ΔELTA
Smarter. Greener. Together.

Industrial Automation Headquarters
Delta Electronics, Inc.
Taoyuan Technology Center
No.18, Xinglong Rd., Taoyuan City,
Taoyuan County 33068, Taiwan
TEL: 886-3-362-6301 / FAX: 886-3-371-6301

Asia

Delta Electronics (Jiangsu) Ltd.
Wujiang Plant 3
1688 Jiangxing East Road,
Wujiang Economic Development Zone
Wujiang City, Jiang Su Province, P.R.C. 215200
TEL: 86-512-6340-3008 / FAX: 86-769-6340-7290

Delta Greentech (China) Co., Ltd.
238 Min-Xia Road, Pudong District,
ShangHai, P.R.C. 201209
TEL: 86-21-58635678 / FAX: 86-21-58630003

Delta Electronics (Japan), Inc.
Tokyo Office
2-1-14 Minato-ku Shibadaimon,
Tokyo 105-0012, Japan
TEL: 81-3-5733-1111 / FAX: 81-3-5733-1211

Delta Electronics (Korea), Inc.
1511, Byucksan Digital Valley 6-cha, Gasan-dong,
Geumcheon-gu, Seoul, Korea, 153-704
TEL: 82-2-515-5303 / FAX: 82-2-515-5302

Delta Electronics Int'l (S) Pte Ltd.
4 Kaki Bukit Ave 1, #05-05, Singapore 417939
TEL: 65-6747-5155 / FAX: 65-6744-9228

Delta Electronics (India) Pvt. Ltd.
Plot No 43 Sector 35, HSIIDC
Gurgaon, PIN 122001, Haryana, India
TEL : 91-124-4874900 / FAX : 91-124-4874945

Americas

Delta Products Corporation (USA)
Raleigh Office
P.O. Box 12173,5101 Davis Drive,
Research Triangle Park, NC 27709, U.S.A.
TEL: 1-919-767-3800 / FAX: 1-919-767-8080

Delta Greentech (Brasil) S.A.
Sao Paulo Office
Rua Itapeva, 26 - 3° andar Edificio Itapeva One-Bela Vista
01332-000-São Paulo-SP-Brazil
TEL: 55 11 3568-3855 / FAX: 55 11 3568-3865

Europe

Deltronics (The Netherlands) B.V.
Eindhoven Office
De Witbogt 15, 5652 AG Eindhoven, The Netherlands
TEL: 31-40-2592850 / FAX: 31-40-2592851

*We reserve the right to change the information in this manual without prior notice.

V3.0
DELTA_ASDA-A2-E_UM_EN_20150430

ASDA A2-E EtherCAT Interface Servo Drive User Manual

ASDA A2-E EtherCAT Interface Servo Drive User Manual

www.deltaww.com

ΔELTA
Smarter. Greener. Together.

Table of Contents

(This page is intentionally left blank.)

Chapter 1 CoE Drive Overview

1.1 Communication Specification

EtherCAT Communication	Physical layer	100BASE-TX
	Communication connector	RJ45 × 2 (Connector CN6A=IN, CN6B=OUT)
	Network topology	Line connection
	Baud rate	2 x 100 Mbps (full duplex)
	Frame data length	Maximum 1484 bytes
	SyncManager	SM0: Mailbox output SM1: Mailbox input SM2: Process data output SM3: Process data input
	FMMU (Fieldbus Memory Management Units)	FMMU0: Process data output area FMMU1: Process data input area FMMU2: Mailbox status area
	Device profile	CoE: CANopen over EtherCAT
	Synchronization mode	DC synchronization (SYNC0) Non- synchronized (Free Run)
	Communication object	SDO: Service Data Object PDO: Process Data Object EMCY: Emergency Data Object
	LED indicator (On RJ45 Connector)	EtherCAT ERR (ER) × 1 EtherCAT Link/Activity (L/A) × 2 EtherCAT RUN (RN) × 1
	Application layer specifications	IEC61800-7 CiA402 Drive Profile
The supported CiA402 operation modes		■ Profile Position Mode (PP) ■ Profile Velocity Mode (PV) ■ Profile Torque Mode (PT) ■ Homing Mode (HM) ■ Interpolated Position Mode(IP) ■ Cycle Synchronized Position Mode (CSP) ■ Cycle Synchronized Velocity Mode (CSV) ■ Cycle Synchronized Torque Mode (CST)

1.2   The Interface of Delta EtherCAT Servo Drive

Link/Activity of EtherCAT
input port indicator (L/A)

Run indicator (RUN)

Link/Activity of EtherCAT
output port indicator (L/A)

Error indicator (ERR)

EtherCAT input port
(CN6)

EtherCAT output
port (CN6)
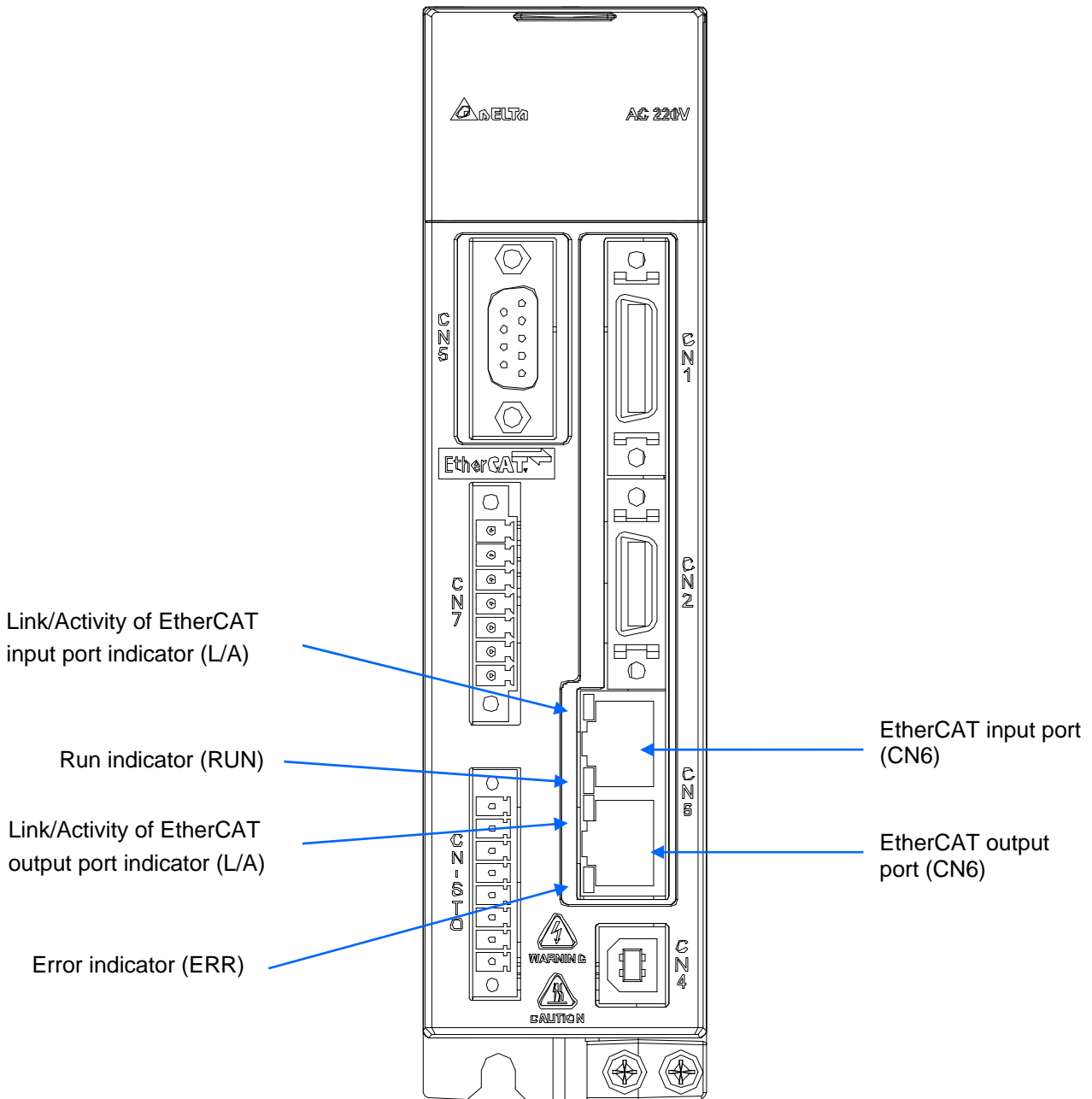
Figure 1 The Interface of Delta EtherCAT Servo Drive

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

1.3 LED Indicators

Indicator state	Indicator pattern
ON	**ON** ────────────────────── **OFF**

Blinking	**ON** ┌──┐ ┌──┐ ┌──┐ ┌──┐
	     │ │ │ │ │ │ │ │
	OFF ┘ 200ms └ 200ms ┘ └──┘ └──┘
	     ←200ms→←200ms→

Single Flash	**ON** ┌──┐ ┌──┐
	     │ │ │ │
	OFF ┘ 200ms └────── 1000ms ──────┘ └
	     ←200ms→←──────1000ms──────→

OFF	**ON** **OFF** ────────────────────────

Figure 2. RJ45 LED indicator pattern

■ ERROR (ERR) LED

The ERR LED indicator shows the error status of EtherCAT communication.

Indicator state	Slave State
Off	No error
Blinking	State change error
Single Flash	Synchronization error SyncManager error
On	PDI Watchdog timeout

State change error:	The state machine does not allow the system to change its state because the wrong parameter settings. Please refer to Figure 29 for its switching conditions.
Synchronization error:	The synchronization of Master Clock and Slave Clock is failed.
SyncManager error:	The data of process data is lost when receiving.
PDI Watchdog timeout:	The hardware failure on slave. Please contact Delta distributors for assistance.

■ RUN LED

The RUN LED indicator shows the status of EtherCAT state machine

Indicator state	Slave State
Off	INIT
Blinking	PRE-OPERATIONAL
Single Flash	SAFE-OPERATIONAL
On	OPERATIONAL

INIT:	After power on, the EtherCAT slave will get into INIT state if there is no error. At INIT state, no communication servo is provided. Accessing slave's register from the host is available at this state.
PRE-OPERATIONAL:	The SDO can be used to communicate with its host controller.
SAVE-OPERATIONAL:	Both SDO and TxPDO, which can send cyclic data from the slave to the host, are workable.
OPERATIONAL:	SDO, TxPDO and RxPDO are working.

■ Link Activity (L/A) LED

The L/A LED indicator shows the physical link status and the link activity.

Indicator state	Slave State
Off	No link
Blinking	Link and activity
On	Link without activity

No link:	The link has not established yet.
Link and activity:	The data is exchanging with its partners.
Link without activity:	The link is established but no data is exchanging now.

1.4 The Topology

The topology is defined by the host controller. Please refer to the host controller's application manual. There are only one input port and one output port on Delta servo drive for EtherCAT communication ports.
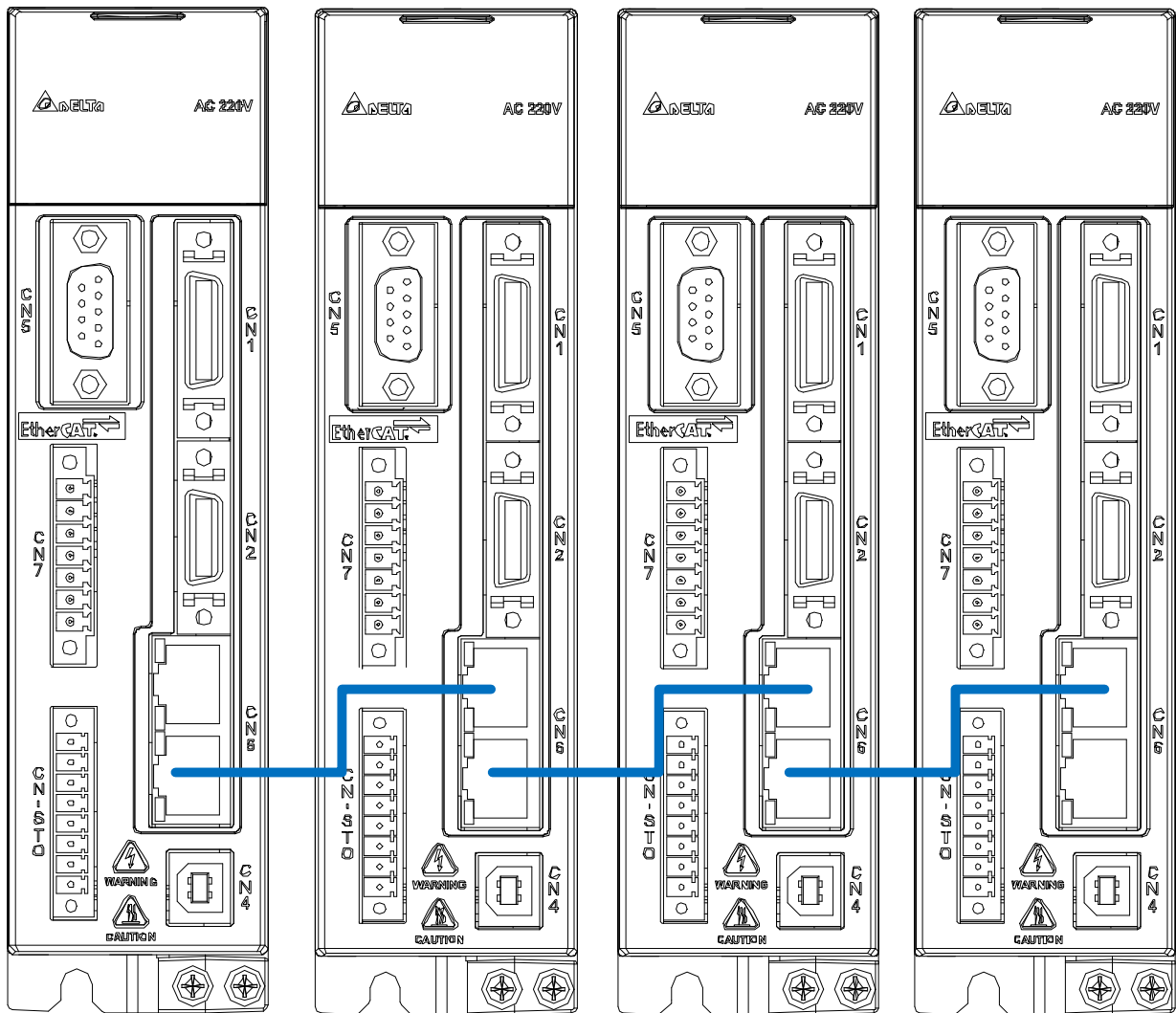
Figure 3 EtherCAT connection topology example

1.5 Wiring

I/O Signal (CN1) Connection and Connector Terminal Layout

In order to have a more flexible communication with the master, 4 programmable Digital Output (DO) and 7 programmable Digital Input (DI) are provided, which are parameter P2-18~P2-21 and P2-10~P2-16 respectively. In addition, the differential output encoder signal A+, A-, B+, B-, Z+ and Z- is also provided. The followings are the pin diagrams.
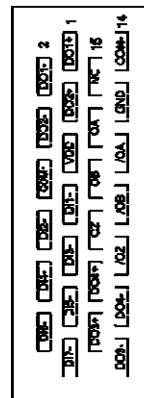
CN1 Connector (female)

CN1 Connector (male)
rear view

2	DO1-	Digital output
4	DO2-	Digital output
6	COM+	Power input (12~24V)
8	DI2-	Digital input
10	DI4-	Digital input
12	DI6-	Digital input

1	DO1+	Digital output
3	DO2+	Digital output
5	VDD	+24V Power output
7	DI1-	Digital input
9	DI3-	Digital input
11	DI5-	Digital input
13	DI7-	Digital input

15	NC	N/A
17	OA	Encoder/ A pulse output
19	OB	Encoder/ B pulse output
21	OZ	Encoder/ Z pulse output
23	DO4+	Digital output
25	DO3+	Digital output

14	COM-	VDD power ground
16	GND	Analog input signal ground
18	/OA	Encoder/ A pulse output
20	/OB	Encoder/ B pulse output
22	/OZ	Encoder/ Z pulse output
24	DO4-	Digital output
26	DO3-	Digital output

> **NOTE**

NC means NO CONNECTION. This terminal is for internal use only. Do not connect it, or it may damage the servo drive.

1.5.1 Explanation of I/O (CN1) Connector Signal

The following details the signals listed in previous section.

General Signals

Signal Name		Pin No	Function	Wiring Method (Refer to 3.4.3)
Position pulse (output)	OA	17	Encoder signal output A, B, Z (Line Driver output)	C13/C14
	/OA	18		
	OB	19		
	/OB	20		
	OZ	21		
	/OZ	22		
Power	VDD	5	VDD is the +24V power provided by the drive and is for Digital Input (DI) and Digital Output (DO) signal. The maximum current is 500mA.	-
	COM+	6	COM+ is the common input of Digital Input (DI) and Digital Output (DO) voltage. When using VDD, VDD should be connected to COM+. If not using, it needs to apply the external power (+12V ~+24V). Its positive end should connect to COM+ and the negative end should connect to COM-.	
	COM-	14		
	GND	16	VCC voltage is based on GND.	
Other	NC	15	NO CONNECTION. This terminal is for internal use only. Do not connect it, or it may damage the servo drive.	

1.5.2 CN2 Connector

CN2 Connector (female)
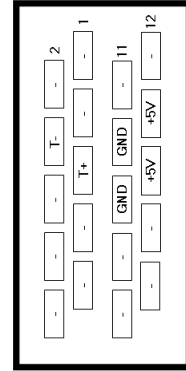
CN2 Connector (male)
Rear view
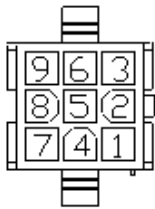
Quick Connector
HOUSING: AMP (1-172161-9)
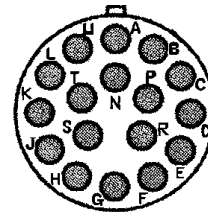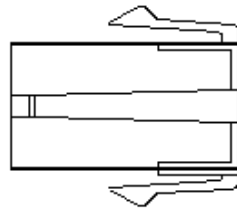
Military Connector
3106A-20-29S

Drive Connector			Motor Connector		
Pin No	Terminal Symbol	Function and Description	Military connector	Quick connector	Color
5	T+	Serial communication signal input / output (+)	A	1	Blue
4	T-	Serial communication signal input / output (-)	B	4	Blue & Black
-	-	Reserved	-	-	-
-	-	Reserved	-	-	-
14,16	+5V	Power +5V	S	7	Red/Red & white
13,15	GND	Power ground	R	8	Black/Black & white
-	-	shielding	L	9	-

1.5.3 CN5 Connector (Full-closed Loop)

Connect linear scale or encoder (A, B, Z format) to the servo and form a full-closed loop. In position mode, the pulse command issued by the controller is based on the control loop of the external linear scale. Please refer to Chapter 6.
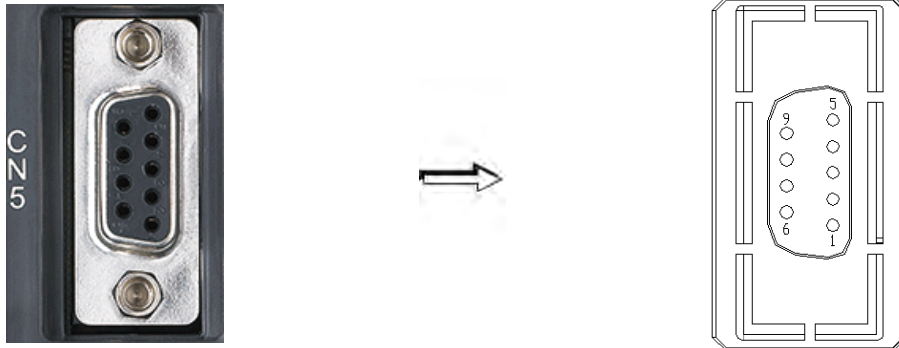
CN5 Connector (female)

Pin No	Signal Name	Terminal Symbol	Function and Description
1	/Z phase input	Opt_/Z	/Z phase
2	/B phase input	Opt_/B	/B phase
3	B phase input	Opt_B	B phase
4	A phase input	Opt_A	A phase
5	/A phase input	Opt_/A	/A phase
6	Encoder grounding	GND	Ground
7	Encoder grounding	GND	Ground
8	Encoder power	+5V	+ 5V power
9	Z phase input	Opt_Z	Z phase

NOTE

1. It only supports the encoder of AB phase and 5V voltage.

2. The application of full-closed loop: it supports the encoder of highest resolution 1280000 pulse/rev (A pulse number per motor revolution for a full-closed loop that corresponds to an optical signal with AB (Quadrature) phase pulses (4x).).

1.5.4 CN6 EtherCAT Terminal

IN

OUT

CN5 Connector (female)
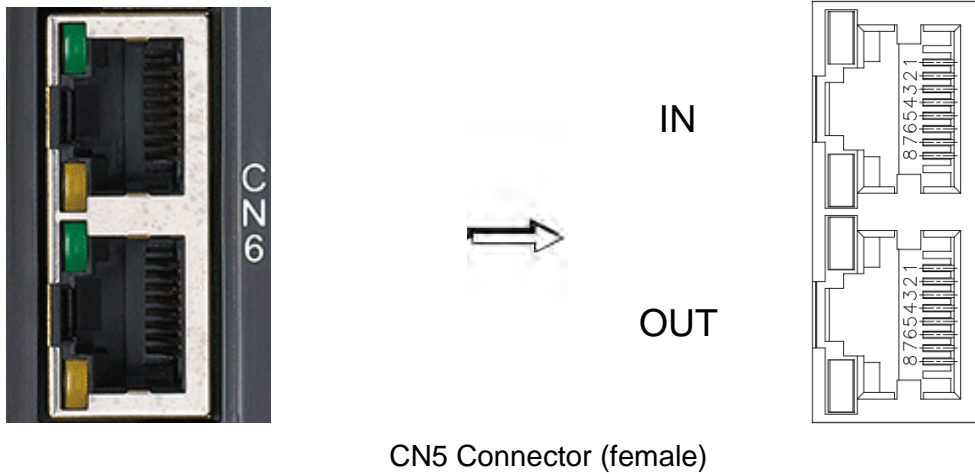
Pin No	Signal Name	Terminal Symbol	Function and Description
1	TX +	TX +	Transmit +
2	TX -	TX -	Transmit -
3	RX +	RX +	Receive +
4	-	-	-
5	-	-	-
6	RX -	RX -	Receive -
7	-	-	-
8	-	-	-

NOTE

1. The maximum distance between two stations should be 50 meters.
2. Please use CAT5e STP Shielding.

1.5.5 CN7 Extension DI
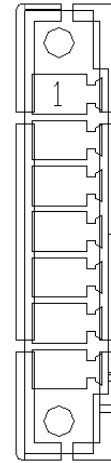
CN7 Connector (male)

Pin No	Signal Name	Terminal Symbol	Function and Description
*1	VDD 24V power	COM+	VDD (24V) power is the same as the voltage of Pin11 in CN1
2	Extension DI9	EDI 9-	Digital input pin 9-
3	Extension DI10	EDI 10-	Digital input pin 10-
4	Extension DI11	EDI 11-	Digital input pin 11-
5	Extension DI12	EDI 12-	Digital input pin 12-
6	Extension DI13	EDI 13-	Digital input pin 13-
7	Extension DI14	EDI 14-	Digital input pin 14-

WARNING

➢ *Caution: Do not apply to dual power or it may damage the servo drive.*

1.5.6 CN-STO
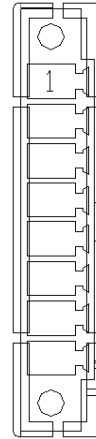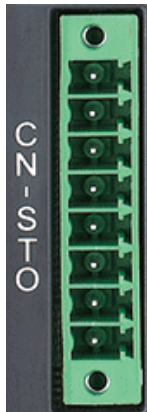
CN-STO Connector (male)

Pin No	Signal Name	Terminal Symbol	Function and Description
*1	VDD24V power	COM+	VDD (24V) power is the same as the voltage of Pin11 in CN1
2	STO_A	STO_A	STO input pin A+
3	/STO_A	/STO_A	STO input pin A-
4	STO_B	STO_B	STO input pin B+
5	/STO_B	/STO_B	STO input pin B-
6	FDBK_A	FDBK_A	STO alarm output pin A, Relay Output **Max. Current : 1A**
7	FDBK_B	FDBK_B	STO alarm output pin B, Relay Output **Max. Current : 1A**
8	COM-	COM-	VDD(24V) power ground

 ➤ *Caution: Do not apply to dual power or it may damage the servo drive.*

1.5.7 STO with Safety Relay
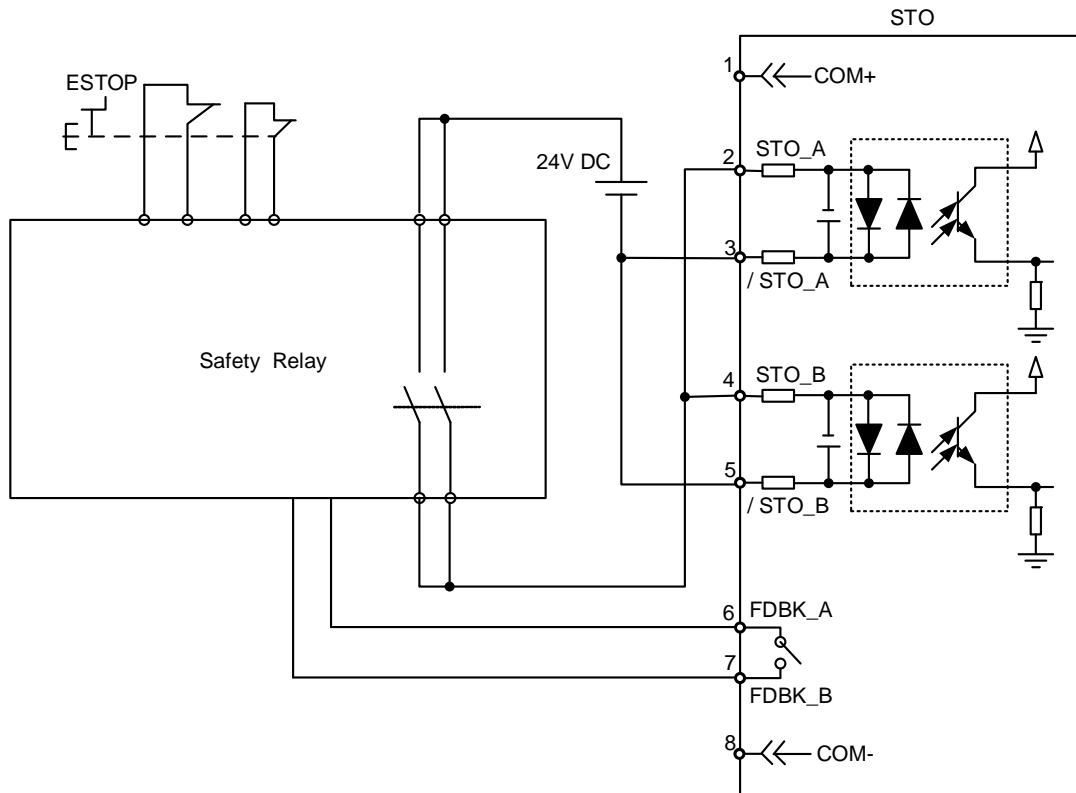
1.5.8 STO Disable
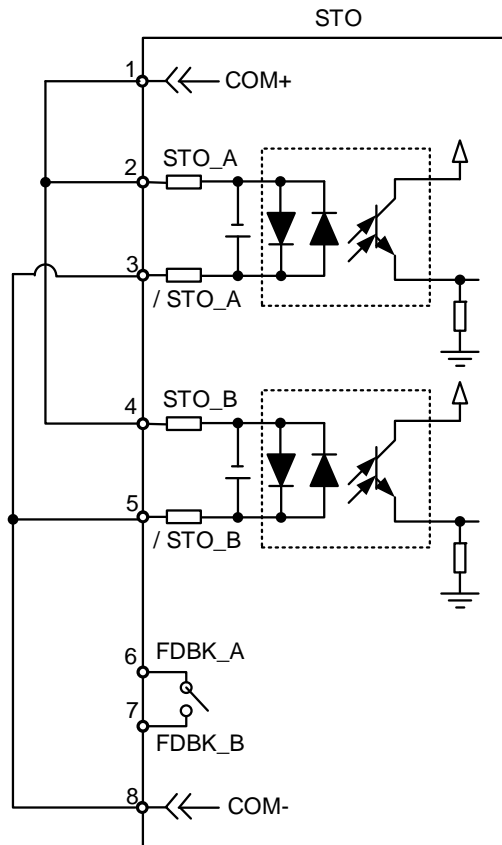
1.6 Dimension

1.6.1 220V Series

100W/200W/400W
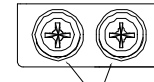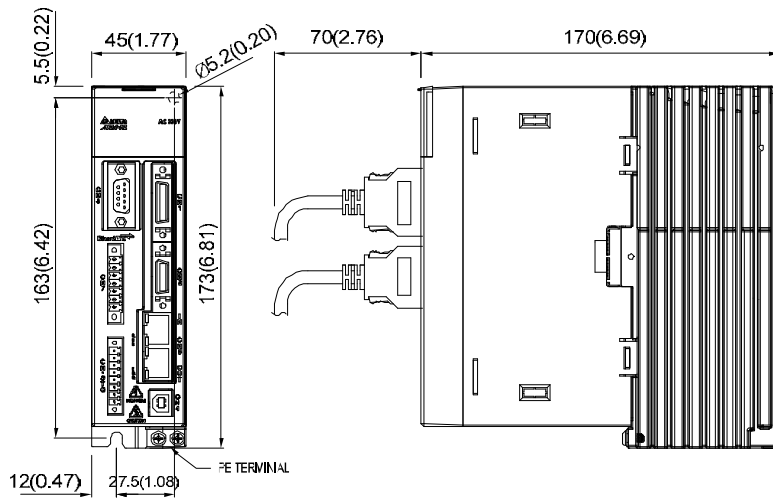
Weight	1.5（3.3）

750W/1kW/1.5kW

Weight	2.0（4.4）

2kW/3kW

SCREW: M4×0.7
MOUNTING SCREW TORQUE:14(kgf-cm)

Weight	2.89（6.36）

PE
TERMINIAL

1.6.2 400V Series
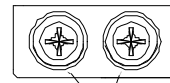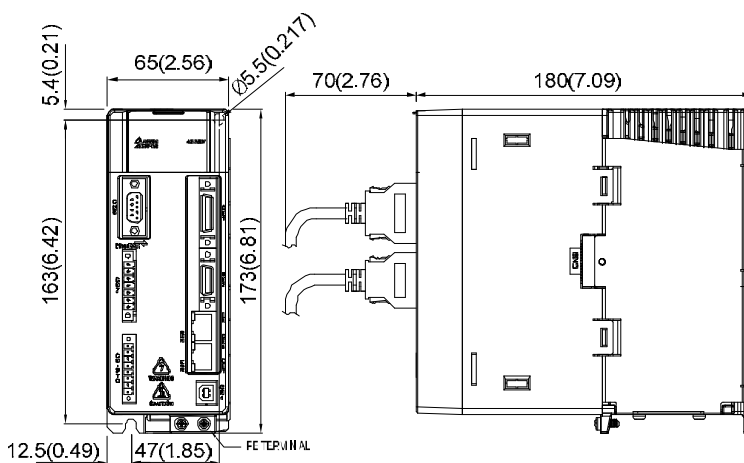
400W/750W/1kW/1.5kW
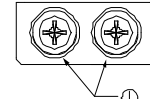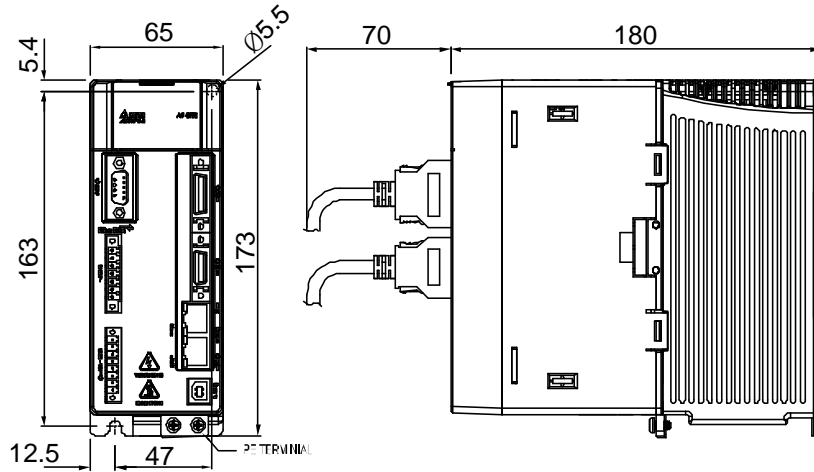
SCREW: M4×0.7
MOUNTING SCREW TORQUE:14(kgf-cm)

Weight	2.0（4.4）

2kW/3kW/4.5kW/5.5kW
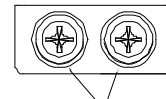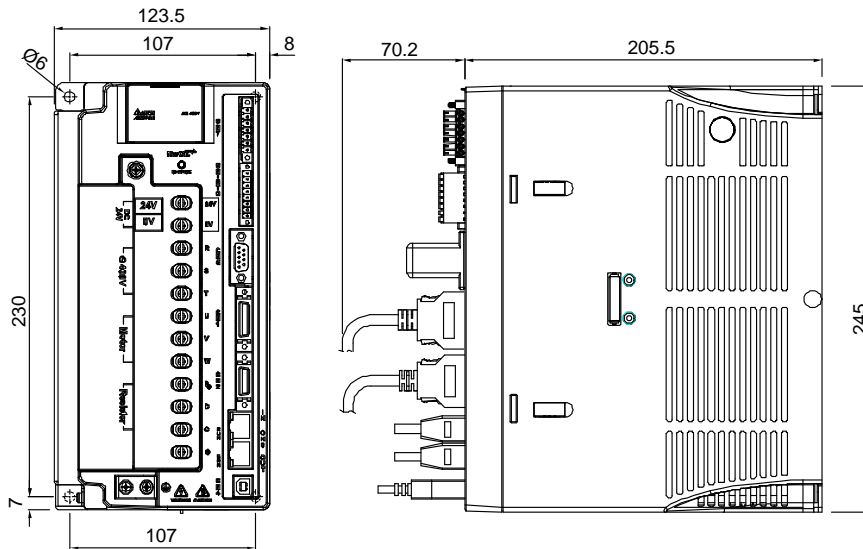
SCREW: M4×0.7
MOUNTING SCREW TORQUE:14(kgf-cm)

Weight	4.6（10.1）

7.5kW
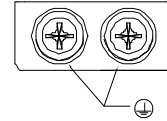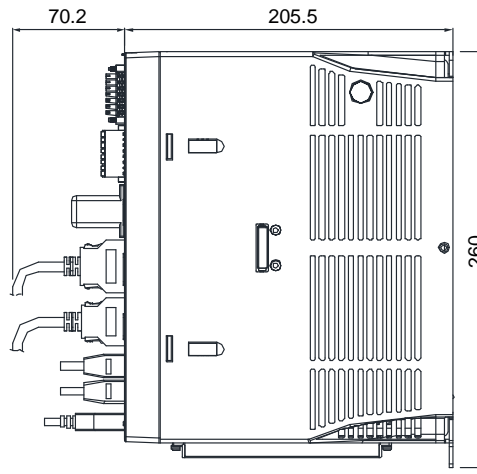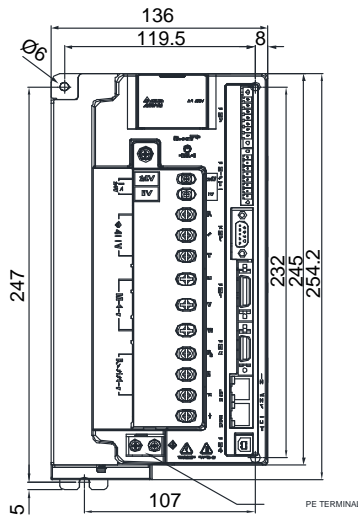
| Weight | 4.6（10.1） |

SCREW: M4×0.7
MOUNTING SCREW TORQUE:14(kgf-cm)

NOTE

1. Dimensions are in millimeters (inches); Weights are in kilograms (kg) and pounds (lbs).
2. Dimensions and weights of the servo drive may be revised without prior notice.

Chapter 2 System Setup

2.1 Parameter Settings of EtherCAT Mode

1. Set parameter **P1-01** to **0x0C$_h$** for EtherCAT communication and CANopen as the application layer.

2. Restart the system of servo drive.

P1-01●	CTL	Control Mode and Output Direction		Address : 0102H 0103H
Interface :	Panel / Software	Communication	Reference :	-
Default :	0		Control Mode :	ALL
Unit :	Pulse (P mode); r/min (S mode); N-m (T mode)		Range :	00 ~ 0x110F
Format :	Hex		Data Size :	16-bit

Settings :

→ Control mode settings
→ Torque output direction settings
→ Discrete I/O Setting
→ not used
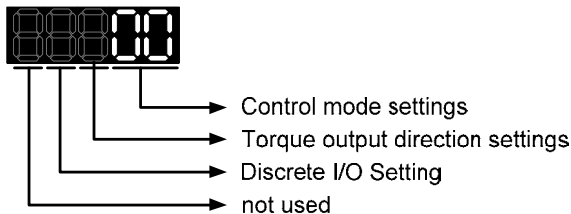
Control mode settings

	PT	PR	S	T	Sz	Tz
Single Mode						
00	▲					
01		▲				
02			▲			
03				▲		
04					▲	
05						▲
Multiple Mode						
0E	▲	▲	▲			
0F	▲	▲		▲		

	PT	PR	S	T	Sz	Tz
Dual Mode						
06	▲		▲			
07	▲			▲		
08		▲	▲			
09		▲		▲		
0A			▲	▲		
0B	N/A					
0C	CANopen Mode					
0D	▲	▲				

PR: Position control mode. The command is from the internal signal. Execution of 64 positions is via DI.POS0 ~ POS5. A variety of homing control is also provided.

S: Speed control mode. The command is from the external signal or internal signal. Execution of the command selection is via DI.SPD0 and DI.SPD1.

T: Torque control mode. The command is from the external signal or internal signal. Execution of the command selection is via DI.TCM0 and DI.TCM1.
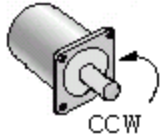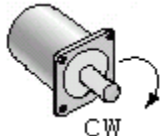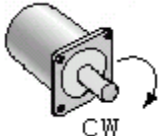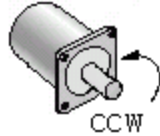
Sz: Zero speed / internal speed command

Tz: Zero torque / internal torque command

Dual Mode: The control mode selection is via DI signals. For example, either PT or S control mode can be selected via DI signal, S-P (see Table A).

Multiple Mode: The control mode selection is via DI signals. For example, PT, PR or S control mode can be selected via DI signals, S-P and PT-PR (see Table A).

Torque output direction settings

Direction	0	1
Forward	CCW	CW
Reverse	CW	CCW

Discrete I/O Setting

1: When switching to different mode, digital inputs/outputs (P2-10 ~ P2-22) will be set to the default value according to the mode you selected.

0: When switching to different mode, the setting value of digital inputs/outputs (P2-10 ~ P2-22) will remain the same and will not be changed.

2.2 TwinCAT Setup

A lot of software can be applied to configure EtherCAT system. The following procedures are the example of TwinCAT of Beckhoff. Please install the software properly before you start to configure the system.

1. Copy Delta XML description to the folder the TwinCAT installed (usually C:\TwinCAT\Io\EtherCAT).

2. Restart the TwinCAT.

3. The configuration procedure can be started by applying TwinCAT manager which shown as below.
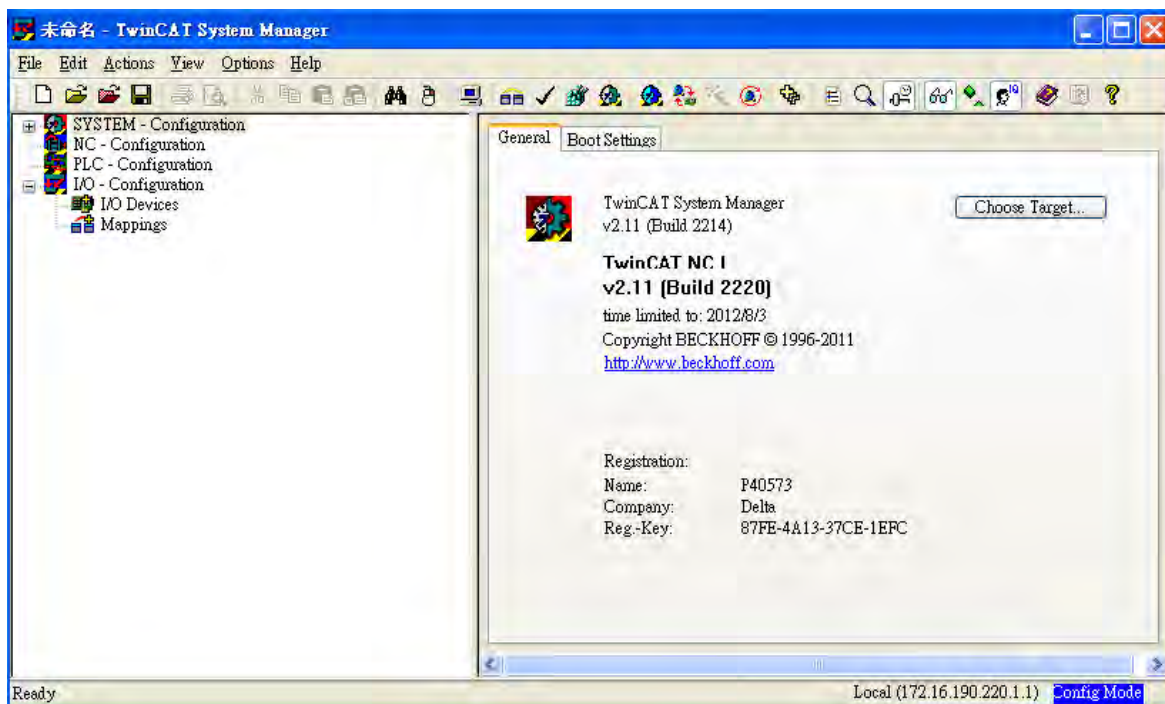
Figure 4

4. Install the Network Interface Card (NIC) for EtherCAT communication.

    ■   Select Options → Show Real Time Ethernet Compatible Devices.
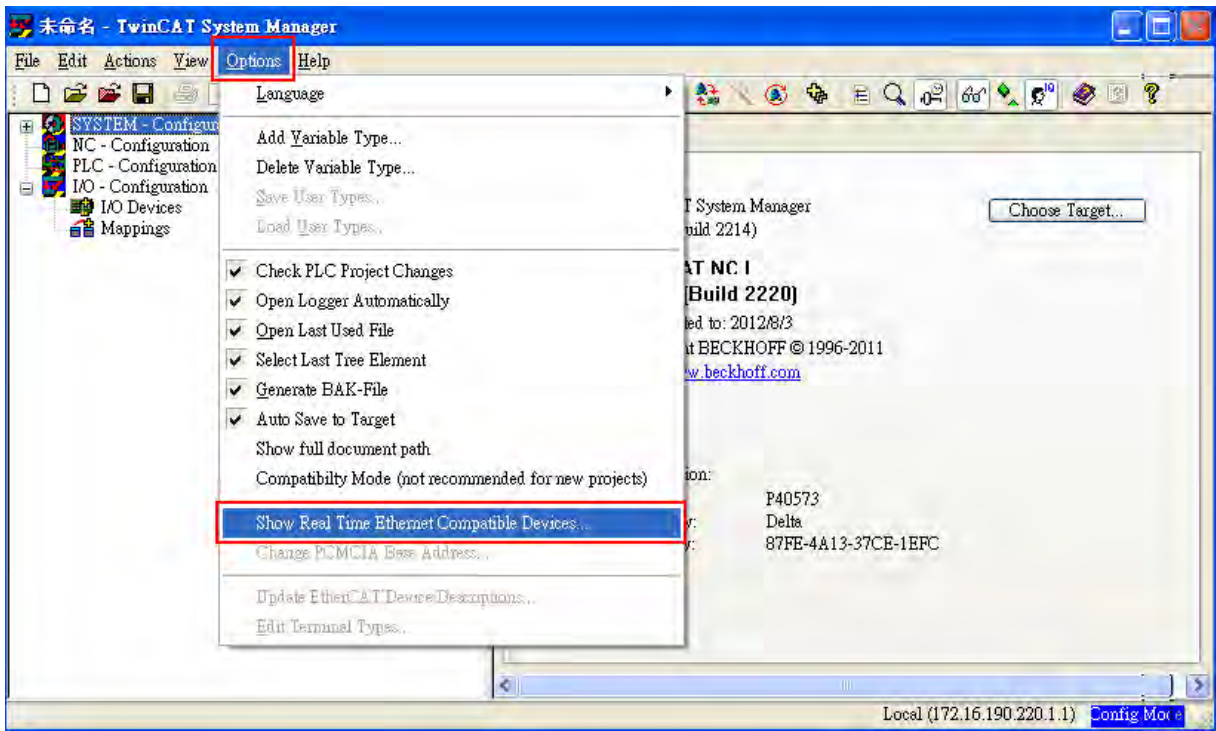
Figure 5

■   Select the correct Adapter from the devices (NICs) installed in the computer for EtherCAT communication and click "**Install**".
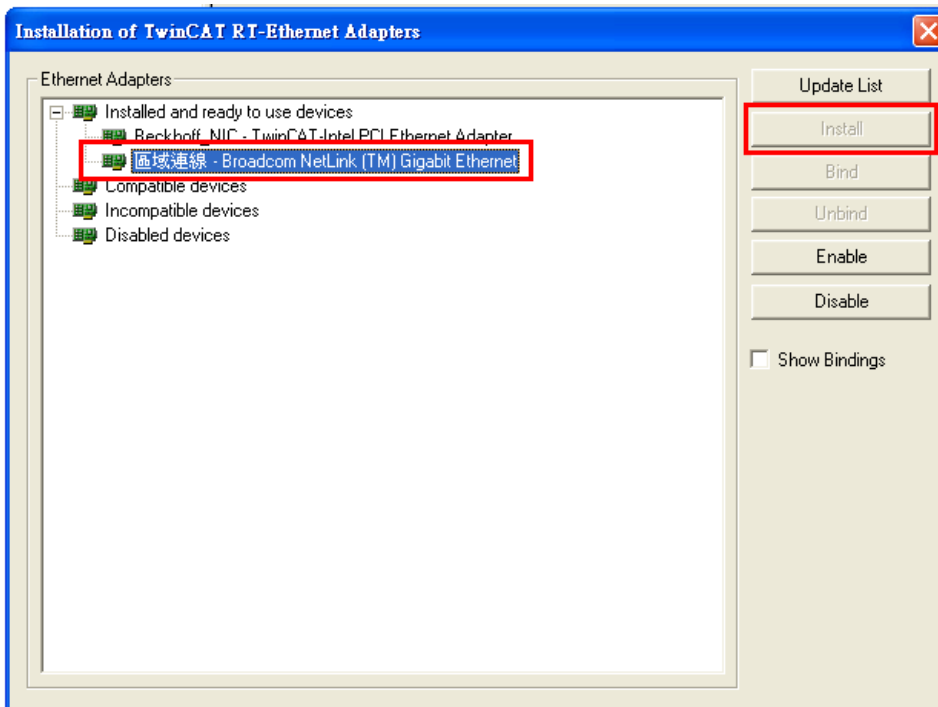
Figure 6

5.   Open a new project from the drop down menu File → new.

6.   Right click I/O Devices and select **Scan Devices** or Press <**F5**> to scan the devices. Click **OK** in the pop-up dialog window to confirm the information.
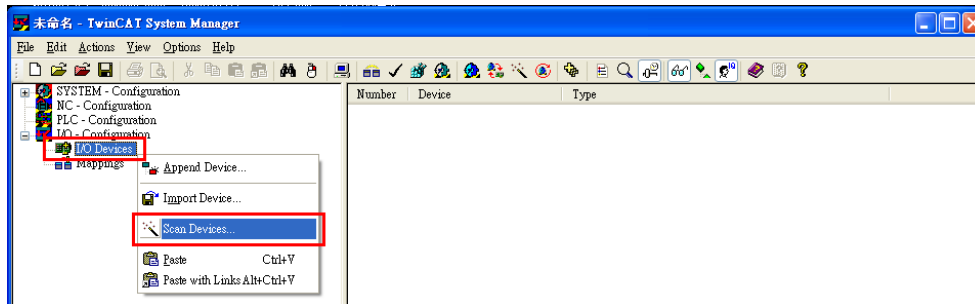
Figure 7

Figure 8

7.   Find Device [n] (EtherCAT), select this device and click **OK**.
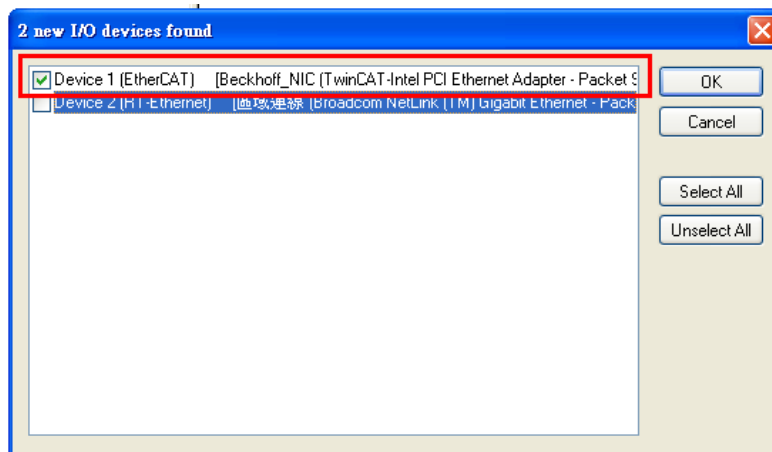
Figure 9

8.   Click **Yes** to scan for boxes.

Figure 10

9.   Click **Yes** to **Add drives** to **NC-Configuration**.

Figure 11

10.   Click **No** and TwinCAT will be switched to **Config mode**.

Figure 12

11.   TwincAT is in Config Mode. In the left panel, it shows **Device (EtherCAT)** and you can find ASDA A2-E CoE Drive.
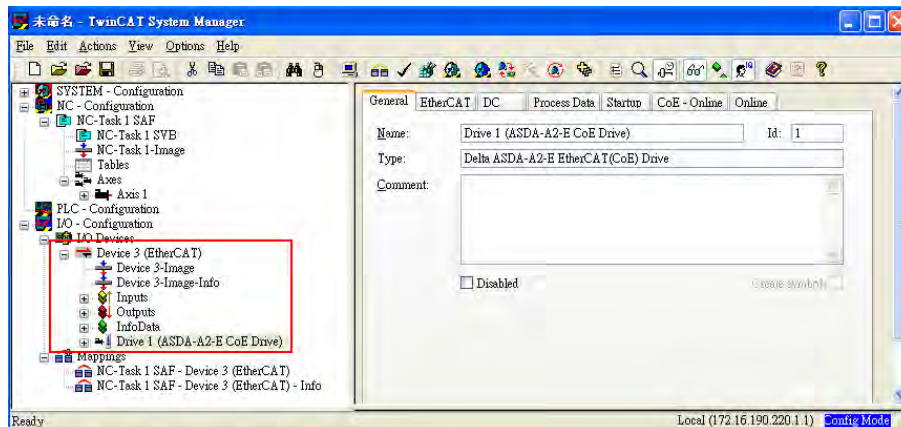
Figure 13

12. Select the Drive (ASDA A2-E) and in **Online** tab you can check if the device's EtherCAT state machine (ESM) is in PREOP state.
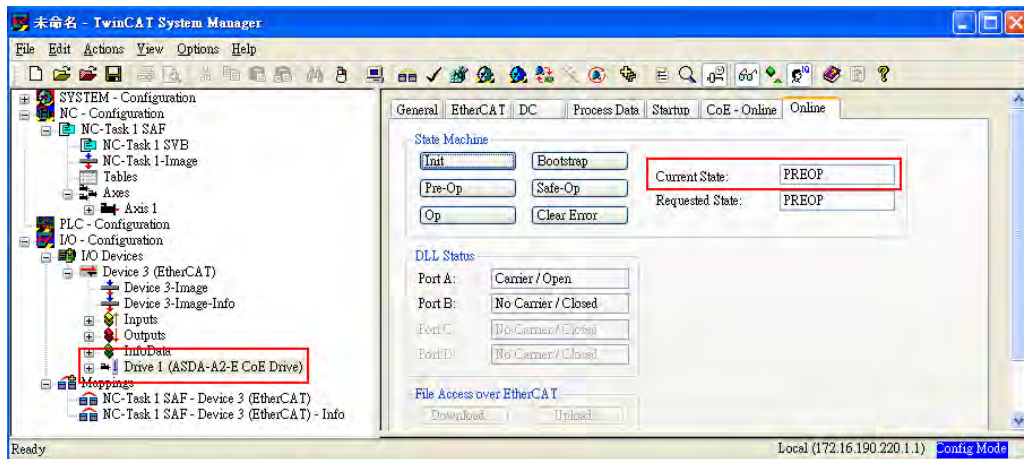
Figure 14

13. Double click on Drive (ASDA A2-E CoE Drive) and it will show:

2nd TxPDO –   CoE Tx PDO mapping
3rd RxPDO –   CoE Rx PDO mapping
WcState
InfoData
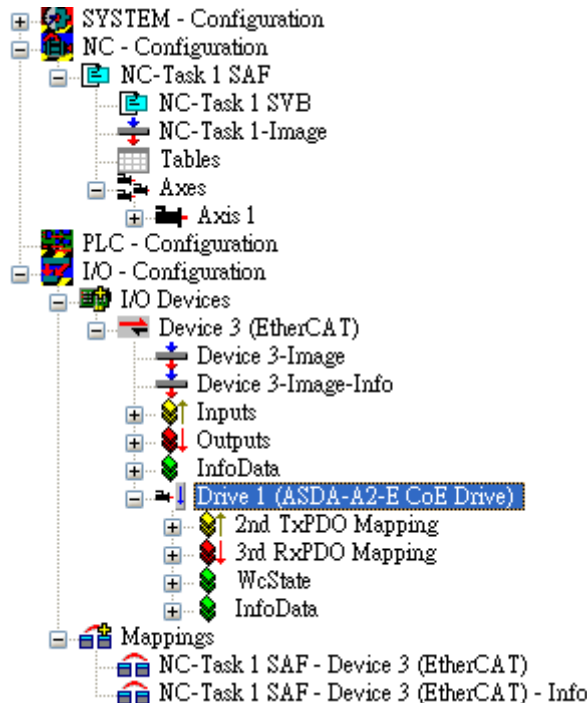
Figure 15

14. Set the communication cycle* and the default value is 2ms.

- Select **NC-Task 1 SAF** in the left window, and set **Cycle ticks** as communication cycle (The minimum value is 1ms) in the right window.
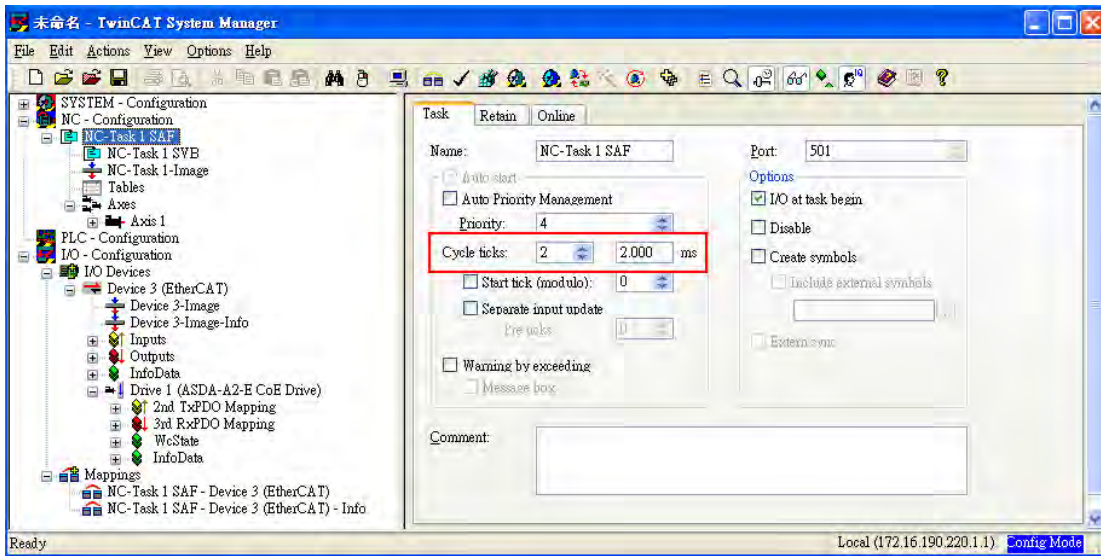
Figure 16

*The communication cycle time, SYNC0 cycle time, and PDO cycle time should be set to the same value.

15. Set **Following Error Calculation** to **Extern**.

- Select **Axis 1_Drive** in the left window → In parameter column of the right window, select **Extern** in **Following Error Calculation** → click **Download** and then click **OK** in pop-up dialog.
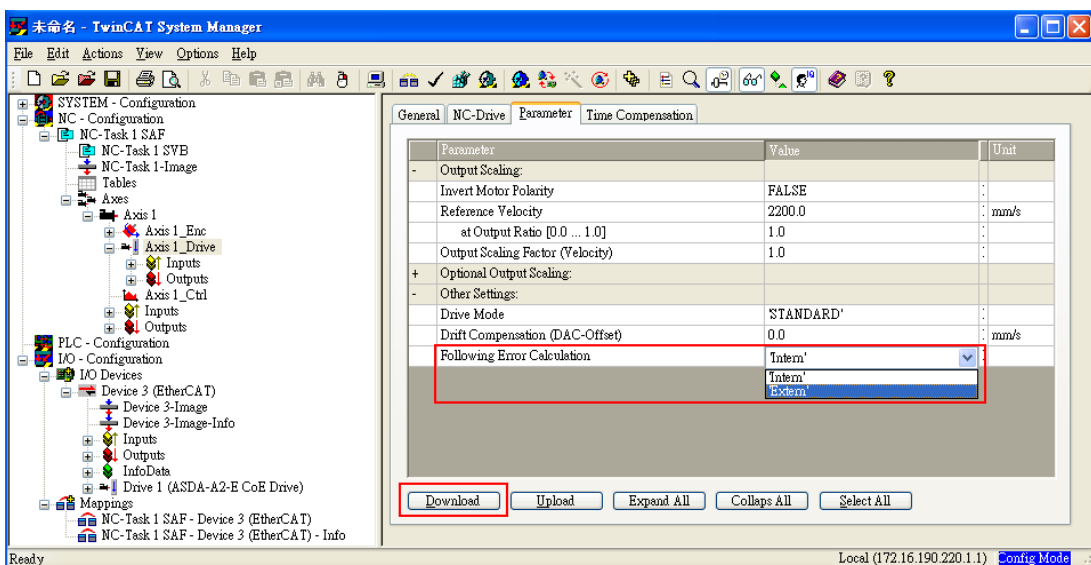
Figure 17

16. Switch TwinCAT to Run Mode.

■ Press ▣▣ to generate Mappings → press ✓ to check confiugration → and
    press 🗗 to activate configuration. TwinCAT will be switched to Run Mode
    and then click OK in pop-up dialog.

Figure 18

17. Enable the axis (Servo On).

■ Under NC-Configuration of the left window, select **Axis 1** → select **Online**
    tab in the right window → click **Set**.
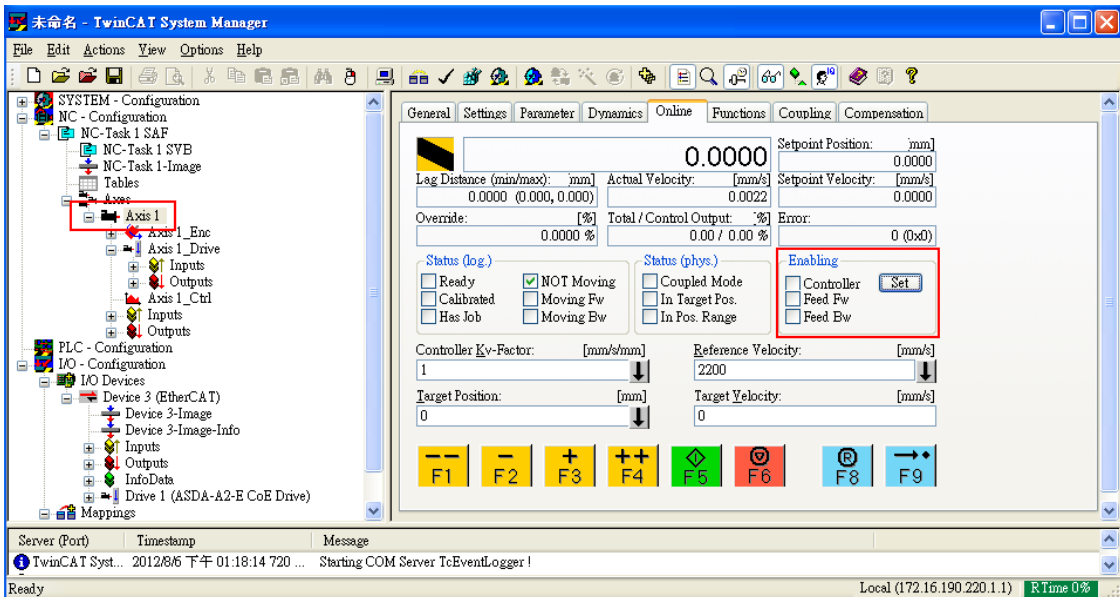
Figure 19

■ In pop-up dialog, click **All** to enable the motor.
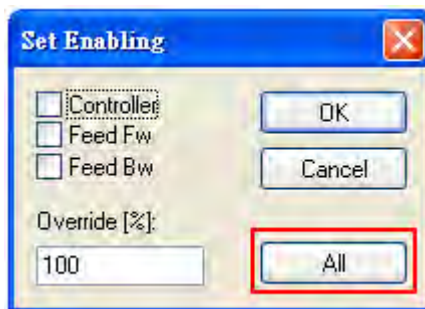
Figure 20

18. In **Online** tab, there are two different speed levels of jogging buttons for forward and backward movement which can test the system. During the operation, please **Be Ensured** that the movement would not damage your system and endanger the personnel safety.

Figure 21

2.3 Synchronization Modes Setting

2.3.1 Two Synchronization Modes of Delta Servo

ASDA A2-E supports two synchronization modes, Free Run mode and DC-Synchronous mode. Please note that the asynchronous Free Run mode is still under the definition of "Synchronization Modes" within EtherCAT specification guide.

■ Free Run Mode (Asynchronous)

The master and slaves are running in an asynchronous manner. The master and the slave both have their own clock to calculate the time. In other words, there is no synchronous clock between the master and the slave. A command sent by the master and a reply from the slave only consists with a sequential order instead of strict clock timing. For example, a master sends a PDO at tick t1 and the slave will receive it at tick t1 or tick t2 and vice versa.
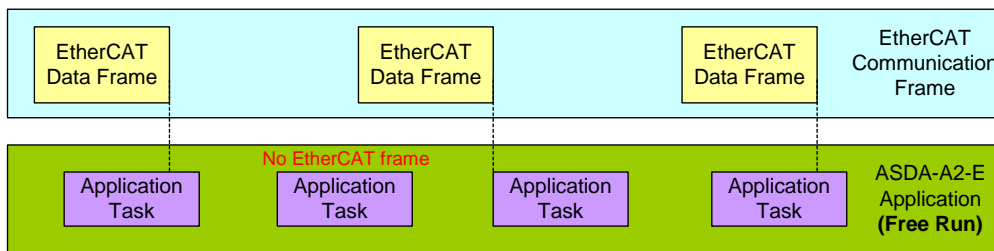
Figure 22 Free Run Mode synchronization

■ DC-Synchronous Mode (SYNC0 synchronization)

There exists a clock tick for the master and all slaves operation. A data sent by the master will be received by slave(s) at the same clock interval. The master will inform all slaves about its clock and ask slaves to align according to the time. A strict clock tick is always running within this system.
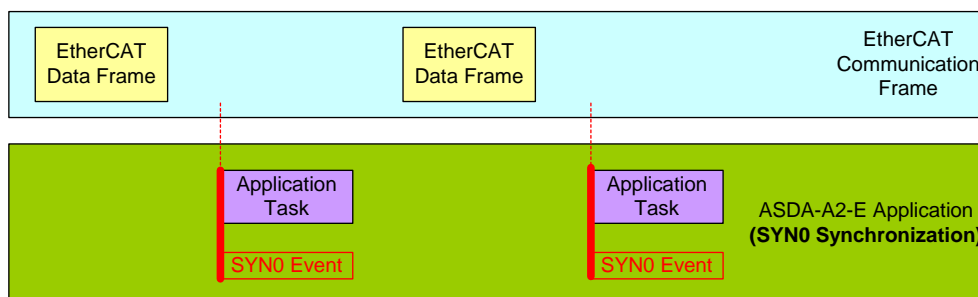
Figure 23 DC-Synchronous mode synchronization

2.3.2 Select the Synchronization Mode

1. Select Drive (ASDA A2-E CoE Drive) in the left window.

2. The DC tab in the right window, users can select DC-Synchronous or Free Run as the Operation Mode. This is for selecting synchronous or asynchronous mode.
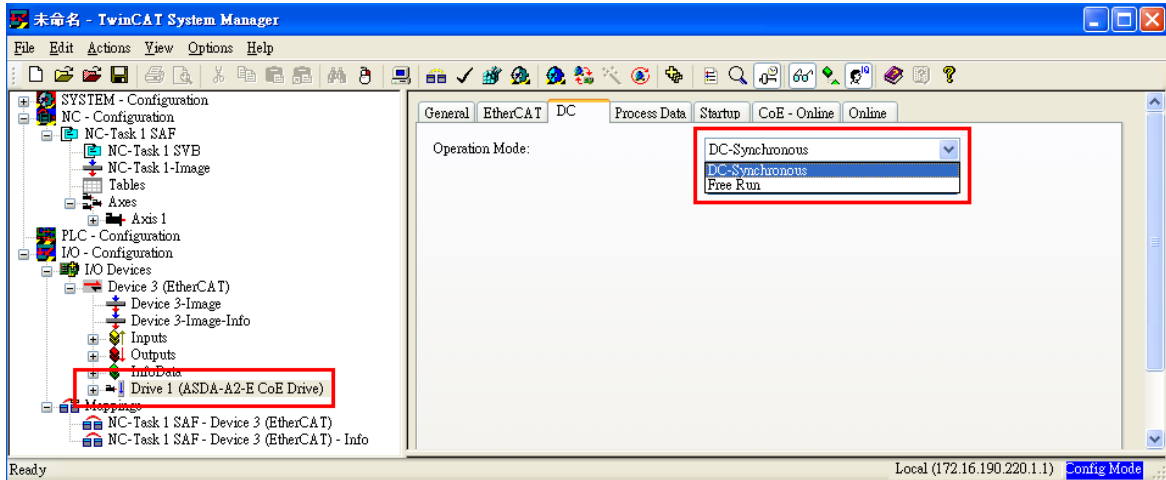
Figure 24

2.3.3 Synchronous Clock Time Setting

1. Select NC-Task 1 SAF in the left window.

2. Click Task in the right window.

3. Cycle ticks are the data exchanging period under the Task tab.
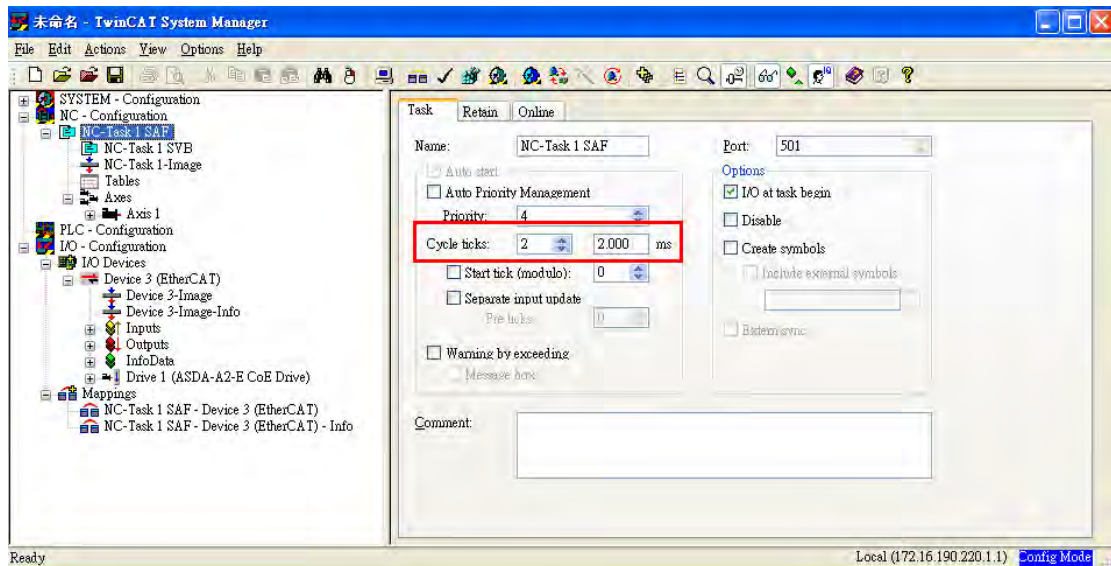
Figure 25

The unit of cycle for SYNC0 cycle time is 1ms.

SYNC0 cycle time supported	1ms (PDO cycle time = 1ms)
	2ms (PDO cycle time = 2ms)
	3ms (PDO cycle time = 3ms)
	…
* **SYNC0 cycle time is used to define PDO cycle time.**	

2.4 PDO Mapping

The PDO mapping Objects are allocated from index 0x1600 to 0x1603 for RxPDOs and 0x1A00 to 0x1A03 for TxPDOs in Object Dictionary.

2.4.1 Default PDO Mappings

The following tables are the default PDO mappings of ASDA A2-E CoE Drive for cyclic data exchange and are also defined in EtherCAT Slave Information file (XML file).

■ 1st PDO Mapping

RxPDO (0x1600)	Control Word (0x6040)	Target Position (0x607A)	Target Velocity (0x60FF)	Target Torque (0x6071)	Mode of Operation (0x6060)

TxPDO (0x1A00)	Status Word (0x6041)	Actual Position (0x6064)	Actual Velocity (0x606C)	Actual Torque (0x6077)	Mode of Operation Display (0x6061)

■ 2nd PDO Mapping (default PDO assignment)

RxPDO (0x1601)	Control Word (0x6040)	Target Position (0x607A)

TxPDO (0x1A01)	Status Word (0x6041)	Actual Position (0x6064)

■ 3rd PDO Mapping

RxPDO (0x1602)	Control Word (0x6040)	Target Velocity (0x60FF)

TxPDO (0x1A02)	Status Word (0x6041)	Actual Position (0x6064)	Actual Velocity (0x606C)

■ 4th PDO Mapping

RxPDO (0x1603)	Control Word (0x6040)	Target Torque (0x6071)	
TxPDO (0x1A03)	Status Word (0x6041)	Actual Position (0x6064)	Actual Torque (0x6077)

2.4.2 Re-define a PDO Mapping

Setup procedure

1. Set 【RxPDO Assignment:0x1C12:0/ TxPDO Assignment: 0x1C13:0】 to 0x0 for disabling the PDO assignment.

2. Set 【RxPDO mapping entry: ex. 0x1601:0/ TxPDO mapping entry: ex. 0x1A01:0】 to 0x0 for disabling the PDO mapping entry setting.

3. Set 【RxPDO mapping entry: ex. 0x1601:0 - 0x1601:7/ TxPDO mapping entry: ex. 0x1A01:0 - 0x1A01:7】.

4. Set 【RxPDO mapping entry: ex. 0x1601:0/ TxPDO mapping entry: ex. 0x1A01:0】 to the number of mapping entries in PDO mapping.

5. Set 【RxPDO Assignment:0x1C12:1/ TxPDO Assignment: 0x1C13:1】 to PDO assignment.

6. Set 【RxPDO Assignment:0x1C12:0/ TxPDO Assignment: 0x1C13:0】 to 0x1 for enabling the PDO assignment.

2.4.3 Using TwinCAT

1. Press 🌀 or **Shift** and **F4** to set/reset TwinCAT to Config Mode (Click OK in pop-up dialog).

2. Select Drive (ASDA A2-E CoE Drive) in the left window. In Process Data field, you can change PDO Assignment for another PDO mapping.

3. Right click the PDO Content Window, and find the PDO mapping that you desire to set, and then you can configure (Insert/Delete/Edit/Move Up/Move Down) the PDO mapping content.

   (8 PDOs is the maximum number of PDO which can be assigned in every PDO mappings.)
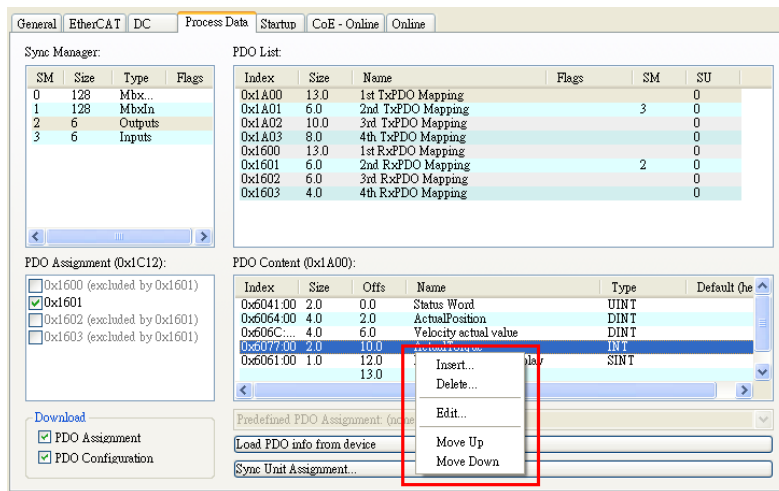
Figure 27
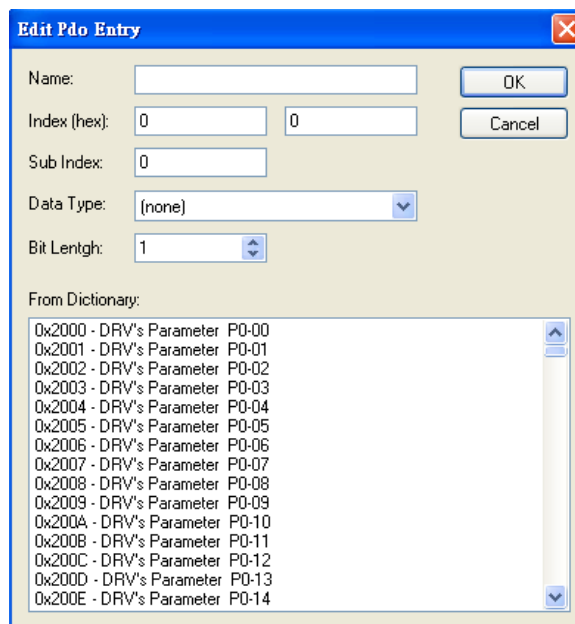
Figure 28 ASD-A2-E CoE drive Object List

4. After changing the PDO Assignment, press or **F4** to reload I/O devices. (Click
   No in pop-up dialog and stay in Config Mode.)

(This page is intentionally left blank.)

Chapter 3 EtherCAT Communication States

ASDA A2-E supports four EtherCAT communication states which are shown as below:

- Init (Initialization)
- Pre-Operational
- Safe-Operational
- Operational
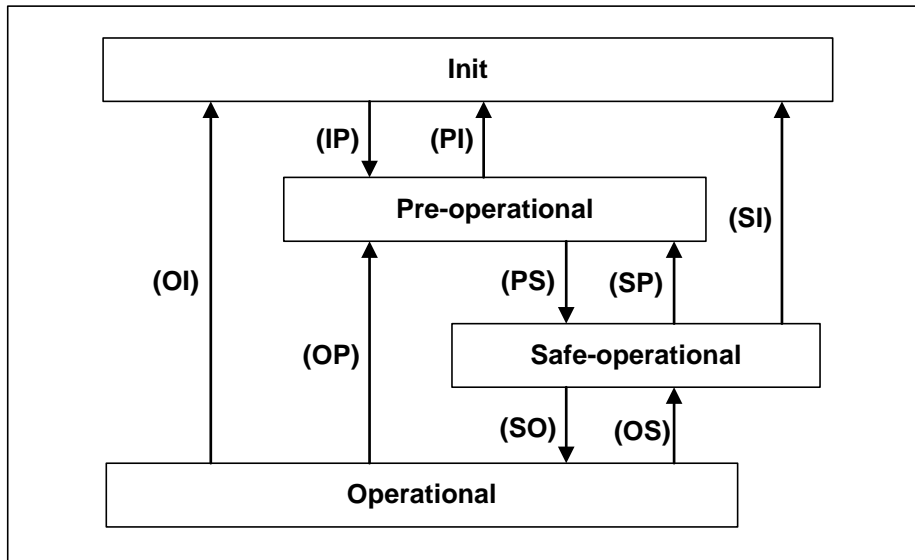
Figure 29 The EtherCAT State machine

EtherCAT host controller can switch the states. Different state provides different service.

State	Description
Init	After power on, the system will be located in this state when hardware is initialized without any error. No communication packet is sent at this stage.
Pre-Operational	The mailbox can be accessed via SDO (Service Data Object). The Emergency message will be sent to the host controller if any alarm occurs.

Safe-Operational	Except SDO, for accessing Mailbox, the PDO (Process Data Object) can only be applied for Process Data Input (TxPDO) at this stage.
Operational	The full function of SDO and PDO (TxPDO and RxPDO) are available now.

3.1 State Transition Operation

The EtherCAT host will send different state transition command for requesting different service.

State Change	Description
IP	● Master will define the slave address and register SyncManager (0/1), and it is possible to access mailbox. ● Master will command the slave to switch to Safe-Operational state.
PS	● SDO from the master will be employed for the settings of PDO mapping. ● Master will define FMMU and register SyncManager (2/3), and slaves keep sending the PDO (TxPDO) packets to the master. ● Master requests the Slave to switch to Operational state.
SO	● Master starts to send PDO (RxPDO). ● The distributed clock synchronization procedure takes place between the master and slaves.
PI, SI, OI	● All communication functions, including SDO and PDO cannot work. ● Switch to Init State.
SP, OP	● Disable PDO function. ● Switch to Pre-Operational state
OS	● Master stops sending Process Data Output (RxPDO). ● Switch to Safe-Operational state

Chapter 4 EtherCAT
Troubleshooting

Q: **Why my TwinCAT cannot find EtherCAT Device from all installed NIC (Network Interface Card) and only shows RT-Ethernet devices?**

A:
1. Please refer to TwinCAT setup procedure and make sure NIC is installed properly.
2. Check if the cable is correctly connected and L/A LED is lit.

Q: **The dialog shows "Unknown device type found" while using TwinCAT Scan boxes.**

A: Copy XML description of the ASDA-A2-E to TwinCAT device description folder (usually in C:\TwinCAT\Io\EtherCAT) and restart TwinCAT System.

Q: **Why does EtherCAT state machine only show INIT in Current State and blank in DLL status when TwinCAT is in Config Mode?**
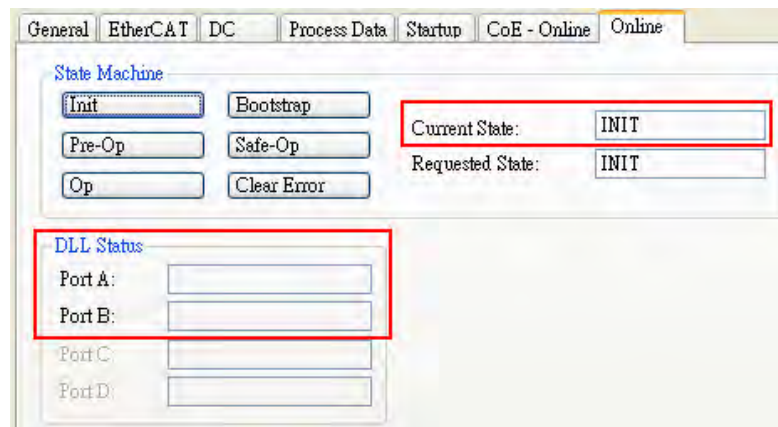
Figure 30

A:
1. Set parameter P1-01 to 0x0C (EtherCAT communication mode).
2. Check the wiring from the host to EtherCAT communication port CN6A for input and CN6B for output on servo drive. If the Link LED lit, it indicates that the physical connection is correct and the drive is connecting.

Q: TwinCAT shows "following error".
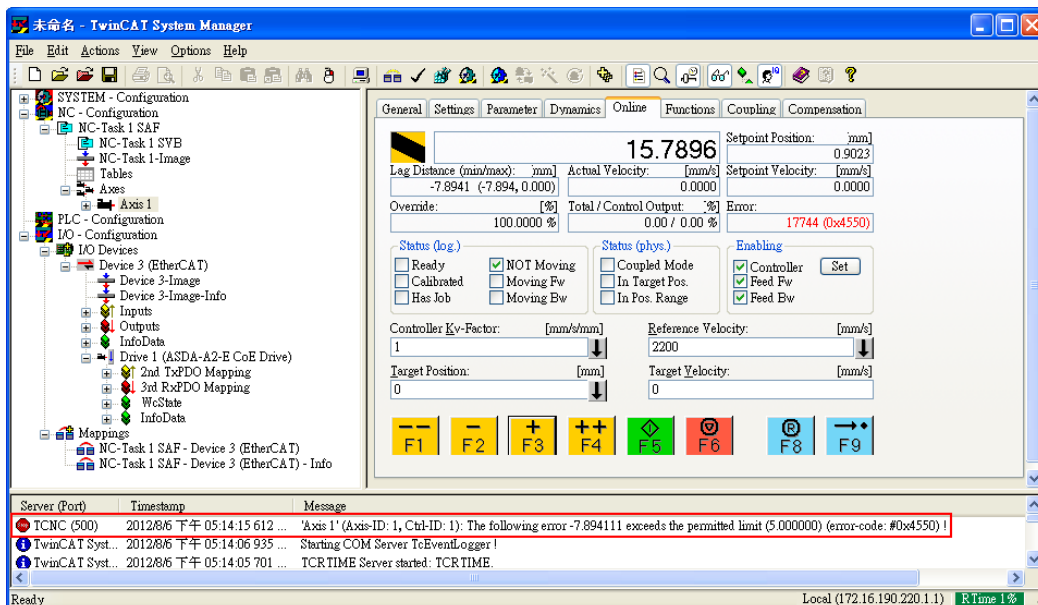
Figure 31

A: Set "Following Error Calculation" to **Extern**:

1. Select **Axis 1_Drive** in the left window.

2. In Parameter tab, select **Extern** in **Following Error Calculation**.

3. **Download** and click **OK** in pop-up dialog.
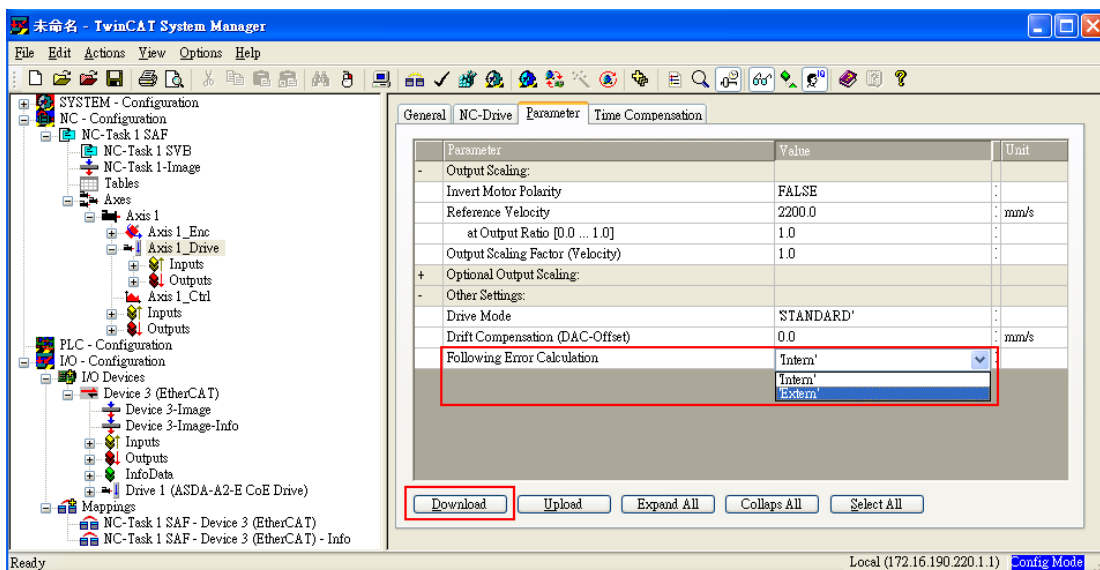
Figure 32

Q: ASDA A2-E servo drive shows AL185

A: This alarm message occurs because of the disconnection of EtherCAT cable between the host and the slave. Please check the wiring. After checking the connection of the cable, it is necessary to re-servo on the drive or set OD 0x6040 to 0x86 for fault reset.

Q: ASDA A2-E servo drive shows AL180

A: Working under Operational state and losing three consecutive PDOs will lead to this alarm.

1. A mechanism inside Delta Servo Drive can be used to monitor the error when receiving PDO by setting P0-02 to 121. If the number keeps increasing, it can be interpreted as the exaggerated jitter of PDO or server interference on the communication cable.

2. Select drive and click **Online**. If the number in the columns of Lost Frames and Rx Errors keeps increasing, it means the system have severe interference.
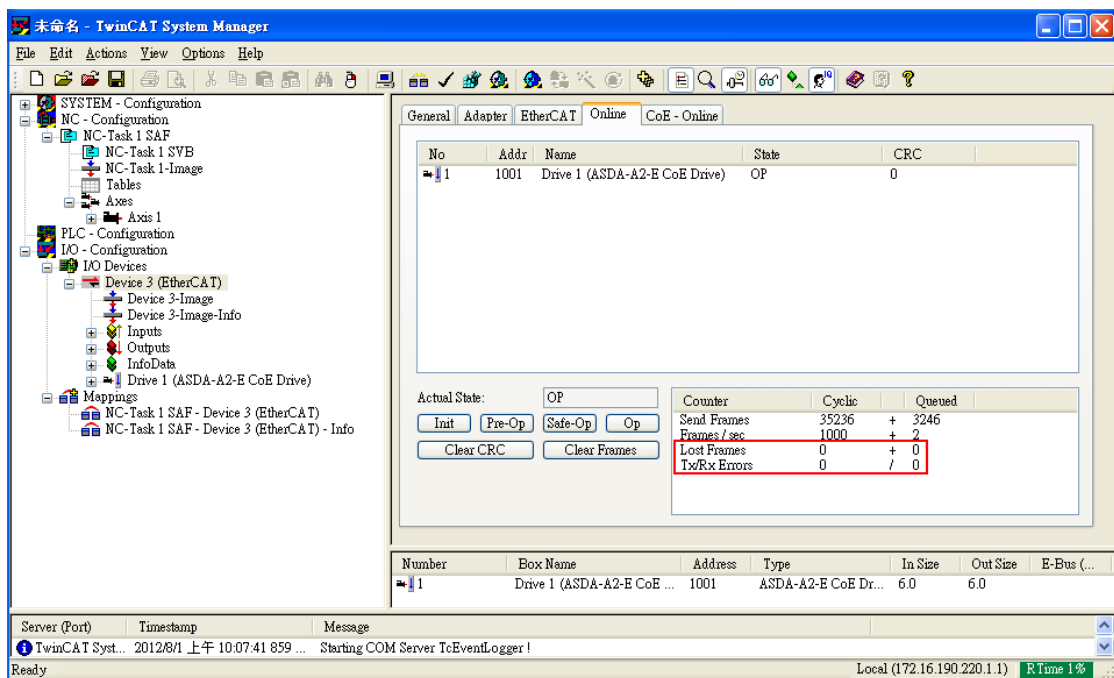
Figure 33

Q: ASDA A2-E servo drive servo drive shows AL3E1

A: Working under Operational state with CANopen CSP/CSV/CST mode and losing two consecutive PDOs will lead to this alarm.

1. Check the reference clock whether with big time jitter

2. Fault reset with control word 0x6040.7 = 1

Q: ASDA A2-E servo drive shows AL3E3

A: Working under Operational state with CANopen CSP/CSV/CST mode and losing two consecutive PDOs will lead to this alarm.

1. Make sure the host controller periodically and stably sends PDO.

2. Make sure grounding and wiring are both correct.

(This page is intentionally left blank.)

Chapter 5 CANopen Operation Mode

5.1 Profile Position Mode

5.1.1 Description

Servo drive (hereinafter referred to as "Drive") receives position command from the host (external) controller (hereinafter referred to as "Host") and then controls servo motor to reach the target position.

Pulse of User-defined Unit Definition:

Pulse of User Unit (PUU): No. of $\frac{PUU}{Rev} = 1280000 \times \frac{0x6093\ Sub2}{OD-6093h\ Sub1}$

5.1.2 Operation Procedures

1. Set【Mode of operations:6060_h】to profile position mode (0x01).

2. Set【Target position:607Ah】to the target position (unit: PUU).

3. Set【Profile velocity:6081h】to the profile velocity (unit: PUU per second).

4. Set【Profile acceleration:6083h】to plan acceleration slope (millisecond from 0rpm to 3000rpm).

5. Set【Profile deceleration:6084h】to plan deceleration slope (millisecond from 0rpm to 3000rpm).

6. Set【Controlword:6040h】as (0x06 > 0x07 > 0x0F) to Servo On the drive and enable the motor.

7. Read【Statusword:6064h】to obtain feedback position of the motor.

8. Read【Statusword:6041h】to obtain the drive status of the following error, set-point acknowledge and target reached.

5.1.3 Advanced Setting Procedures

1. Host could obtain more information about profile position mode.

  ■ Read 【Position demand value:6062_h】to obtain the internal position command.
     (unit: PUU)

  ■ Read 【Position actual value*:6063_h】to obtain the actual position value. (unit:
     increments)

2. Following error

  ■ Set 【Following error window:6065_h】 to define the range of tolerated position
     value, which also defines the range of the position demand value. (unit: PUU)

  ■ Read 【Following error actual value:$60F4_h$】to obtain the actual value of the
     following error. (unit: PUU)

Reference position
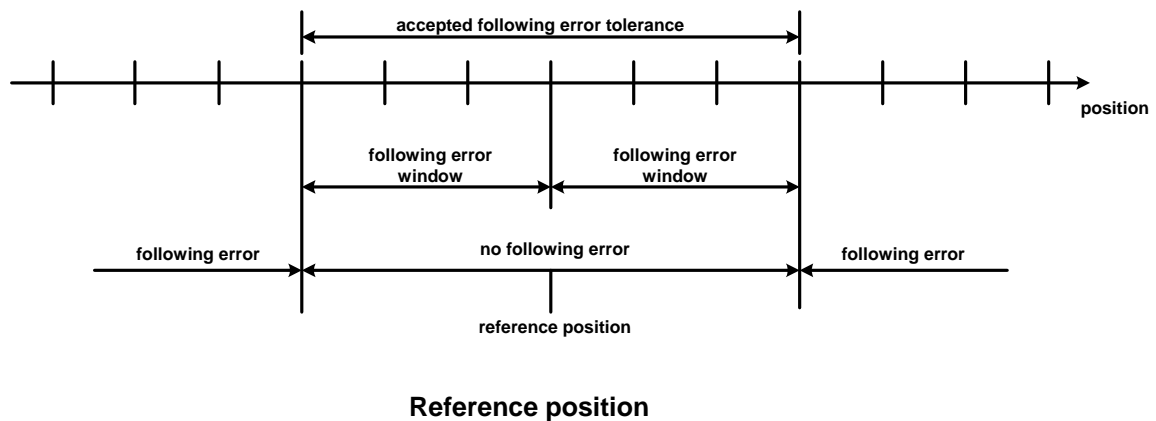
3. Position window

  ■ Set 【Position window:6067_h】to define a symmetrical range of the accepted
     positions which is relative to the target position. (unit: PUU)

  ■ Set 【Position window time:6068_h】to plan activation time of target reached.
     (unit: millisecond)

Position reached
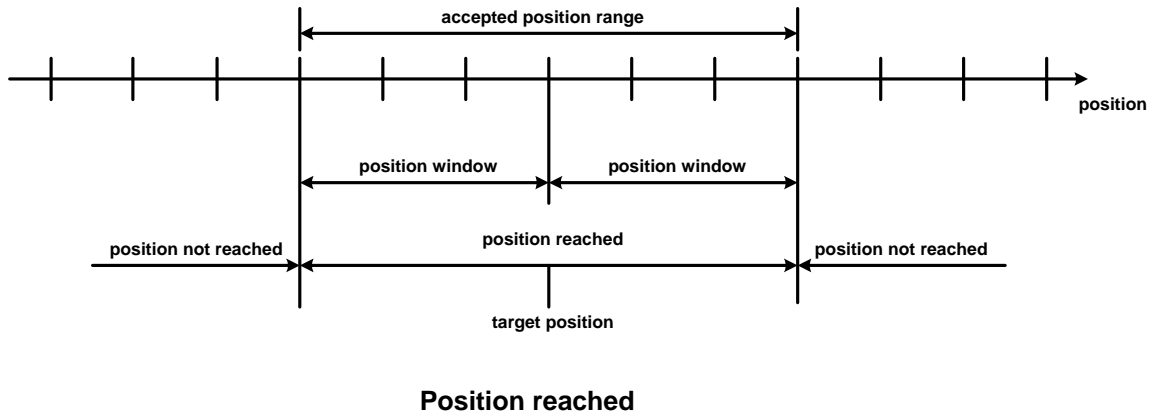
5.1.4 Associated Object List

Index	Name	Type	Attr.
6040$_h$	Controlword	UNSIGNED16	RW
6041$_h$	Statusword	UNSIGNED16	RO
6060$_h$	Modes of operation	INTEGER8	RW
6061$_h$	Modes of operation display	INTEGER8	RO
6062$_h$	Position demand value [PUU]	INTEGER32	RO
6063$_h$	Position actual value [increment]	INTEGER32	RO
6064$_h$	Position actual value	INTEGER32	RO
6065$_h$	Following error window	UNSIGNED32	RW
6067$_h$	Position window	UNSIGNED32	RW
6068$_h$	Position window time	UNSIGNED16	RW
607A$_h$	Target position	INTEGER32	RW
6081$_h$	Profile velocity	UNSIGNED32	RW
6083$_h$	Profile acceleration	UNSIGNED32	RW
6084$_h$	Profile deceleration	UNSIGNED32	RW
6093$_h$	Position factor	UNSIGNED32	RW
60F4$_h$	Following error actual value	INTEGER32	RO
60FC$_h$	Position demand value	INTEGER32	RO

5.2 Interpolation Position Mode

5.2.1 Description

The Host sends PDO periodically. With each PDO, the Host sends the next reference X_i, differece $\triangle X_i$ and controlword to the drive. While the next SYNC0 is receiving, the drive interpolates from X_{i-1} to X_i.

■   Extrapolation, Jitter Compensation

-   When PDO is delayed, the interpolator will predict the speed and position for the next time according to the last acceleration.

-   When PDO delays for 2*cycle, the Drive should stop and send out an error message.

■   PDO Rx/Tx Mapping record

-   The Drive receives PDOs from the Host

·   32 bit reference position [position increment]

·   16 bit symmetrical difference [increments]

$\triangle X_i = (X_{i+1} - X_{i-1})/2$ (It is also the same as velocity.)

·   16 bit controlword

The Drive receives PDOs from the Host (Every PDO contains 8 bytes field which is shown as below.)

32 bit reference position	16 bit difference	16 bit controlword
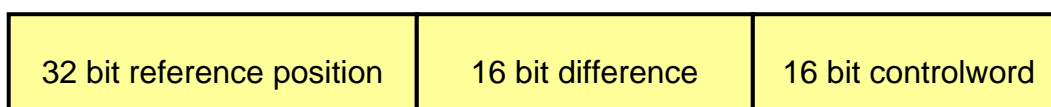
5.2.2 Operation Procedures

1. Set 【Mode of operations:6060$_h$】to interpolation position mode(0x07).

2. Set 【Interpolation sub mode select:60C0$_h$】to Interpolation mode.

  ■ If 60C0$_h$ is [0], the Host does not send [60C1$_h$ Sub-2]. It could save calculating
     time of the host and the Drive could also work.

  ■ If 60C0$_h$ is [-1], the Host needs to send [60C1$_h$ Sub-2] and the Drive works more
     precisely.

3. Set 【Interpolation time period:60C2$_h$】to predict the cycle that SYNC0 receives
   PDO.

  ■ 60C2$_h$ Sub-1 for Interpolation time units. The range is from 1ms to 20ms.

    - 60C2$_h$ Sub-2 for Interpolation time index. The value is always -3, meaning
       the interpolation time unit is 10^{-3} second.

4. Drive PDO Rx:

  ■ 60C1$_h$ Sub-1 for Pos Cmd (32-bit)

  ■ 6040$_h$ Sub-0 for ControlWord.

5.2.3 Associated Object List

Index	Name	Type	Attr.
6040$_h$	Controlword	UNSIGNED16	RW
6041$_h$	Statusword	UNSIGNED16	RO
6060$_h$	Modes of operation	INTEGER8	RW
6061$_h$	Modes of operation display	INTEGER8	RO
6093$_h$	Position factor	UNSIGNED32	RW
60C0$_h$	Interpolation sub mode select	INTEGER16	RW
60C1$_h$	Interpolation data record	ARRAY	RW

(Please refer to the following "Details of Objects" section for more detailed descriptions)

5.3 Cyclic Synchronous Position Mode

5.3.1 Description

The Host plans the path in Cyclic Synchronous Position mode and sends PDO periodically. With each PDO, the Host sends the target position and controlword to the drive. Velocity offset and torque offset can be used as velocity and torque feedforwad.

5.3.2 The Function of CSP Mode
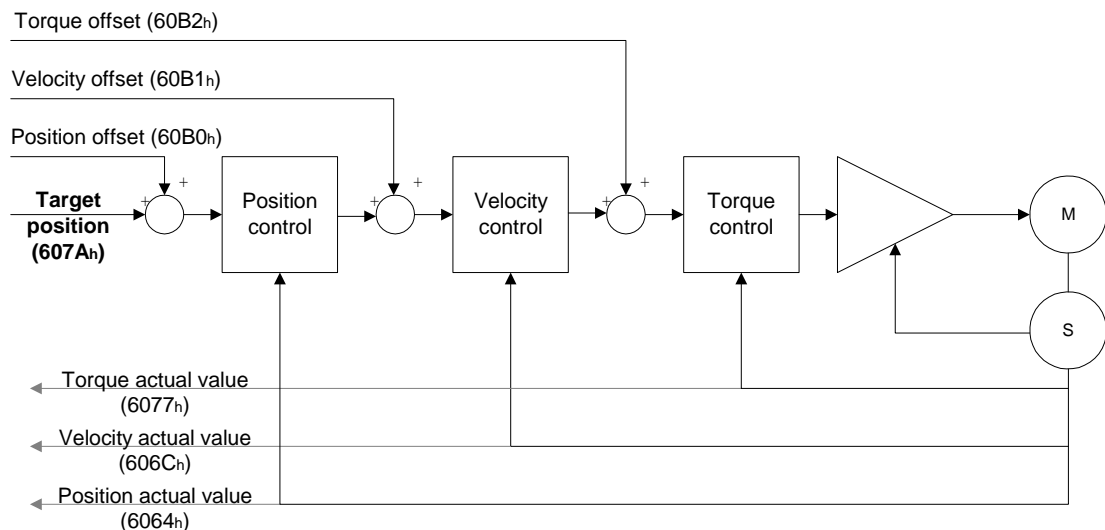
5.3.3 Operation Procedures

1. Set 【Mode of operations:6060_h】 to cyclic synchronous position mode(0x08).

2. Set 【Interpolation time period:$60C2_h$】 to predict the cycle that SYNC0 receives PDO.

- $60C2_h$ Sub-1 for Interpolation time units. The min is from 1ms to 20ms.

  - $60C2_h$ Sub-2 for Interpolation time index. The value is always -3, meaning the interpolation time unit is 10^{-3} second.

3. Drive PDO Rx:

- 607A$_h$ for Target Pos Cmd (32-bit).

- 6040$_h$ Sub-0 for ControlWord.

5.3.4 Associated Object List

Index	Name	Type	Attr.
6040$_h$	Controlword	UNSIGNED16	RW
6041$_h$	Statusword	UNSIGNED16	RO
6060$_h$	Modes of operation	INTEGER8	RW
6061$_h$	Modes of operation display	INTEGER8	RO
607A$_h$	Target position	INTEGER32	RW
60B0$_h$	Position offset	INTEGER32	RW
6064$_h$	Position actual value	INTEGER32	RO
60B1$_h$	Velocity offset	INTEGER32	RW
606C$_h$	Velocity actual value	INTEGER32	RO
60B2$_h$	Torque offset	INTEGER16	RW
6077$_h$	Torque actual value	INTEGER16	RO

(Please refer to the following "Details of Objects" section for more detailed descriptions)

5.4 Homing Mode

5.4.1 Description

This mode could help the drive to find the home position. Users can specify the speed, acceleration and method of homing.

5.4.2 Operation Procedures

1. Set【Mode of operations:6060h】to the homing mode(0x06).

2. Set【Home offset:607Ch】.

3. Set【Homing method:6098h】. The method range is 1 to 35. (Refer to OD-6098h definition which shows below.)

4. Set【Homing speeds:6099h Sub-1】in order to set speed search for switching. (unit: rpm)

5. Set【Homing speeds:6099h Sub-2】in order to set speed during the search for zero. (unit: rpm)

6. Set【Homing acceleration:609Ah】for homing acceleration. (unit: millisecond from 0rpm to 3000rpm)

7. Set【Controlword:6040h】to (0x06 → 0x07 → 0x0F) to Servo ON the drive and enable the motor.

8. Set【Controlword:6040h】to (0x0F → 0x1F) to find Home Switch and do homing.

9. Read【Statusword:6041h】to obtain the drive status.

5.4.3 Associated Object List

Index	Name	Type	Attr.
6040h	Controlword	UNSIGNED16	RW
6041h	Statusword	UNSIGNED16	RO
6060h	Modes of operation	INTEGER8	RW
6061h	Modes of operation display	INTEGER8	RO
607Ch	Home offset	INTEGER32	RW
6093h	Position factor	UNSIGNED32	RW
6098h	Homing method	INTEGER8	RW
6099h	Homing speed	ARRAY	RW
609Ah	Homing acceleration	UNSIGNED32	RW

(Please refer to the following "Details of Objects" section for more detailed descriptions.)

5.5 Profile Velocity Mode

5.5.1 Description

The drive could receive velocity command and plan acceleration and deceleration.

5.5.2 Operation Procedures

1. Set【Mode of operations:6060h】to profile velocity mode(0x03).

2. Set【Controlword:6040h】as (0x06 → 0x07 → 0x0F) to Servo ON the drive and enable the motor.

   (After Servo On, the internal velocity command will be reset and OD-60FFh will be cleared.)

3. Set【Profile acceleration:6083h】to plan acceleration slope. (millisecond from 0rpm to 3000rpm)

4. Set【Profile deceleration:6084h】to plan deceleration slope. (millisecond from 0rpm to 3000rpm)

5. Set【Target velocity:60FFh】. The unit of the target velocity is 0.1rpm.

   (If the drive is already servo-on, it will work immediately while receiving velocity command. OD-60FFh will be cleared to 0 if OD-6060h [Mode] is changed, and Servo-Off or Quick-Stop is activated.)

6. Read【Statusword:6041h】to obtain the drive status.

5.5.3 Advanced Setting Procedures

1. Host could obtain the information of velocity mode.

- ■ Read【Velocity demand value:606B$_h$】to inquire the internal velocity command. (unit: 0.1rpm)

- ■ Read【Velocity actual value:606C$_h$】to obtain the actual velocity value. (unit: 0.1rpm)

2. Host could set velocity monitor threshold.

- ■ Set【Velocity window:606D$_h$】to allocate velocity reached zone. (unit: 0.1rpm)

- ■ Set【Velocity widnow time:606E$_h$】in order to ensure the activation time is before velocity reached. (unit: millisecond)

- ■ Set【Velocity threshold:606F$_h$】to allocate zero speed level. (unit: 0.1rpm)

5.5.4 Associated Object List

Index	Name	Type	Attr.
6040$_h$	Controlword	UNSIGNED16	RW
6041$_h$	Statusword	UNSIGNED16	RO
6060$_h$	Modes of operation	INTEGER8	RW
6061$_h$	Modes of operation display	INTEGER8	RO
606B$_h$	Velocity demand value	INTEGER32	RO
606C$_h$	Velocity actual value	INTEGER32	RO

Index	Name	Type	Attr.
606D$_h$	Velocity window	UNSIGNED16	RW
606E$_h$	Velocity window time	UNSIGNED16	RW
606F$_h$	Velocity threshold	UNSIGNED16	RW
60FF$_h$	Target velocity	INTEGER32	RW

 (Please refer to the following "Details of Objects" section for more detailed descriptions)

5.6 Cyclic Synchronous Velocity Mode

5.6.1 Description

The Host plans the path in Cyclic Synchronous Velocity mode. In this mode, the Host sends PDO periodically including target position and controlword to drive. In addition, velocity offset and torque offset can be used as the velocity and torque feedforwad.

5.6.2 The Function of CSV Mode
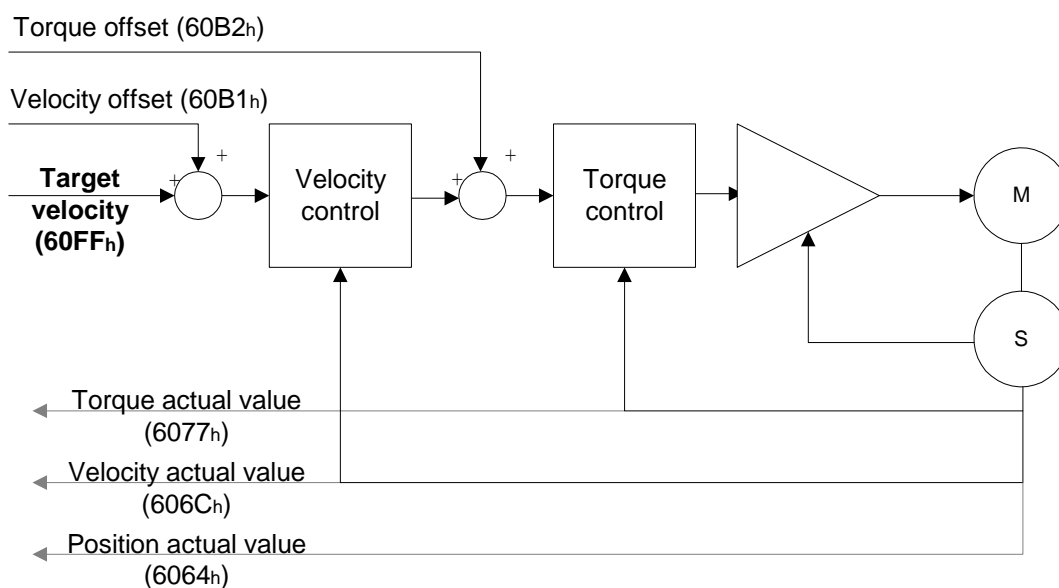
5.6.3 Operation Procedures

1. Set 【Mode of operations:6060_h】to cyclic synchronous velocity mode(0x09).

2. Set 【Interpolation time period:$60C2_h$】 to predict the cycle that SYNC0 receives PDO.

   ■ $60C2_h$ Sub-1 for Interpolation time units. The min is from 1ms to 20ms.

     -   $60C2_h$ Sub-2 for Interpolation time index. The value is always -3, meaning the interpolation time unit is 10^{-3} second.

3. Drive PDO Rx:

■   60F F_h for Target Velocity Cmd (32-bit)

■   6040$_h$ Sub-0 for ControlWord

5.6.4 Associated Object List

Index	Name	Type	Attr.
6040$_h$	Controlword	UNSIGNED16	RW
6041$_h$	Statusword	UNSIGNED16	RO
6060$_h$	Modes of operation	INTEGER8	RW
6061$_h$	Modes of operation display	INTEGER8	RO
60FF $_h$	Target velocity	INTEGER32	RW
60B1 $_h$	Velocity offset	INTEGER32	RW
606C$_h$	Velocity actual value	INTEGER32	RO
6064 $_h$	Position actual value	INTEGER32	RO
60B2$_h$	Torque offset	INTEGER16	RW
6077$_h$	Torque actual value	INTEGER16	RO

5.7 Profile Torque Mode

5.7.1 Description

The drive could receive torque command and plan profile torque slope.

5.7.2 Operation Procedures

1. Set 【Mode of operations:6060h】to profile torque mode(4).

2. Set 【Controlword:6040h】as (0x6 → 0x7 → 0x0F) to Servo ON the drive and enable the motor.

   (After Servo On the drive, the internal torque command will be reset and OD-6071h will be cleared. It means the drive is servo-on and starts to receive the torque command.)

3. Set 【Torque slope:6087h】to plan torque slope time. (unit: millisecond from 0 to 100% rated torque)

4. Set 【Target torque:6071h】to the target torque. The unit is given one rated torque in a thousand. (OD-6071h will be cleared to 0 if OD-6060h [Mode] changed. Servo-Off or Quick-Stop is activated.)

5.7.3 Advanced Setting Procedures

Host could obtain the information of torque mode.

- Read 【Torque demand value:6074_h】 to obtain the output value of the torque limit function. (unit: one rated torque in a thousand)

- Read 【Torque rated current:6075_h】to obtain the rated current determined by the motor and drive type. (unit: multiples of milliamp)

- Read 【Torque actual value:6077_h】to obtain the instantaneous torque in servo motor. (unit: one rated torque in a thousand)

■ Read 【Current actual value:6078$_h$】to obtain the instantaneous current in servo

  motor. (unit: one rated torque in a thousand)

5.7.4 Associated Object List

| Index	Name	Type	Attr.
6040$_h$	Controlword	UNSIGNED16	RW
6041$_h$	Statusword	UNSIGNED16	RO
6060$_h$	Modes of operation	INTEGER8	RW
6061$_h$	Modes of operation display	INTEGER8	RO
6071$_h$	Target torque	INTEGER16	RW
6074$_h$	Torque demand value	INTEGER16	RO
6075$_h$	Motor rated current	UNSIGNED32	RO
6077$_h$	Torque actual value	INTEGER16	RO
6078$_h$	Current actual value	INTEGER16	RO
6087$_h$	Torque slope	UNSIGNED32	RW

(Please refer to the following "Details of Objects" section for more detailed descriptions.)

5.8 Cyclic Synchronous Torque Mode

5.8.1 Description

The Host plans the path in Cyclic Synchronous Torque mode. In this mode, the Host sends PDO periodically including target position and controlword to drive. In addition, velocity offset and torque offset can be used as the velocity and torque feedforwad.

5.8.2 The Function of CST Mode
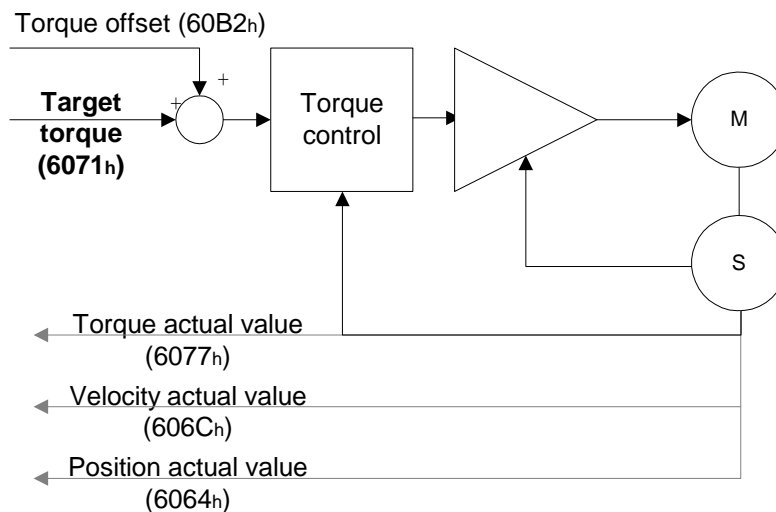
5.8.3 Operation Procedures

1. Set【Mode of operations:6060_h】to cyclic synchronous torque mode (0x0A).

2. Set【Interpolation time period:$60C2_h$】to predict the cycle that SYNC0 receives PDO.

   ■ $60C2_h$ Sub-1 for Interpolation time units. The min is from 1ms to 20ms.

      - $60C2_h$ Sub-2 for Interpolation time index. The value is always -3, meaning the interpolation time unit is 10^{-3} second.

3. Drive PDO Rx:

   ■ 6071_h for Target Torque Cmd (16-bit)

   ■ 6040_h Sub-0 for ControlWord.

5.8.4 Associated Object List

Index	Name	Type	Attr.
6040$_h$	Controlword	UNSIGNED16	RW
6041$_h$	Statusword	UNSIGNED16	RO
6060$_h$	Modes of operation	INTEGER8	RW
6061$_h$	Modes of operation display	INTEGER8	RO
6071$_h$	Target torque	INTEGER16	RW
60B2$_h$	Torque offset	INTEGER16	RW
6077$_h$	Torque actual value	INTEGER16	RO
606C$_h$	Velocity actual value	INTEGER32	RO
6064 $_h$	Position actual value	INTEGER32	RO

(Please refer to the following "Details of Objects" section for more detailed descriptions)

5.9 Limit Position Handling Procedure

5.9.1 Description

Drive will switch to Quick-Stop status while traveling to the position of positive or negative limit sensors, and it can be handled by the following procedures.

5.9.2 Operation Procedures

1. The servo panel will show the alarm while sensors are close to the positive or negative limit. The motor is controlled by a deceleration slope to stop and it is at Quick-Stop status. The drive will keep in servo-on status but will not accept the further motion command.

2. Set 【Controlword:6040h】to 0x8F for fault reset and clearing the alarm displayed on the panel.

3. Set 【Controlword:6040h】to 0x1F/0x0F for Operation Enabled, and then the servo can accept the motion command again.

4. For a motor at its limit position, there must be a command which can drive the motor to the backward direction. Or the alarm will be triggered again while the motor starts moving.

5.10 Touch Probe Function

5.10.1 Description

Touch Probe function can be enabled by the DI on CN7 or the encoder; among that, the feedback position can be latched as positive or negative edge with DI13 on CN7. The time of latch position is shorter than 5 μs and it is used to execute high speed performance in measuring or packaging applications.

5.10.2 Touch Probe Function

The current status of Touch Probe can be obtained by object 60B8h. The definition of each bit is as the followings.

Bit	Value	Definition
0	0	Switch off touch probe 1
	1	Enable touch probe 1
1	0	Trigger first event
	1	Continuous
2	0	Trigger with touch probe 1 input
	1	Trigger with zero impulse signal
3	0	Reserved
4	0	Switch off sampling at positive edge of touch probe 1
	1	Enable sampling at positive edge of touch probe 1
5	0	Switch off sampling at negative edge of touch probe 1
	1	Enable sampling at negative edge of touch probe 1
6、7	0	Reserved
8	0	Switch off touch probe 2
	1	Enable touch probe 2
9	0	Trigger first event
	1	Continuous
10	0	Trigger with touch probe 2 input

	1	Trigger with zero impulse signal
11	0	Reserved
12	0	Switch off sampling at positive edge of touch probe 2
	1	Enable sampling at positive edge of touch probe 2
13	0	Switch off sampling at negative edge of touch probe 2
	1	Enable sampling at negative edge of touch probe 2
14、15		Reserved

5.10.3 Touch Probe Status

The current status of Touch Probe can be obtained by object 60B9h. The definition of each bit is as the followings.

Bit	Value	Definition
0	0	Touch probe 1 is switched off
	1	Touch probe 1 is enabled
1	0	Touch probe 1 has no positive edge value stored
	1	Touch probe 1 has positive edge value stored
2	0	Touch probe 1 has no negative edge value stored
	1	Touch probe 1 has negative edge value stored
3 ~ 5	0	Reserved
6	0	Trigger with touch probe 1 input
	1	Trigger with zero impulse signal
7	0,1	Toggle with every update of Touch probe 1 value Stored
8	0	Touch probe 2 is switched off
	1	Touch probe 2 is enabled
9	0	Touch probe 2 has no positive edge value stored
	1	Touch probe 2 has positive edge value stored

10	0	Touch probe 2 has no negative edge value stored
	1	Touch probe 2 has negative edge value stored
11 ~ 13	0	Reserved
14	0	Trigger with touch probe 2 input
	1	Trigger with zero impulse signal
15	0,1	Toggle with every update of Touch probe 2 value Stored

5.10.4 Associated Object List

Index	Name	Type	Attr.
60B8$_h$	Touch probe function	UNSIGNED16	RW
60B9$_h$	Touch probe status	UNSIGNED16	RO
60BA$_h$	Touch probe pos1 pos value	INTEGER32	RO
60BB$_h$	Touch probe pos1 neg value	INTEGER32	RO
60BC$_h$	Touch probe pos2 pos value	INTEGER32	RO
60BD$_h$	Touch probe pos2 neg value	INTEGER32	RO

(Please refer to the following "Details of Objects" section for more detailed descriptions)

(This page is intentionally left blank.)

Chapter 6 Object Dictionary

Entries

6.1 Specifications for Objects

6.1.1 Object Type

Object Name	Comments
VAR	A single value such as an UNSIGNED8, Boolean, float, INTEGER16 etc.
ARRAY	A multiple data field object where each data field is a sample variable of the SAME basic data type e.g. array of UNSIGNED16 etc. Sub-index 0 is UNSIGNED8 but is not part of the ARRAY data
RECORD	A multiple data field object where the data fields may be any combination of simple variables. Sub-index 0 is UNSIGNED8 but is not part of the RECORD data

6.1.2 Data Type

Please refer to CANopen Standard 301.

6.2 Overview of Object Group 1000$_h$

Index	Object Type	Name	Data Type	Access
1000$_h$	VAR	device type	UNSIGNED32	RO
1001$_h$	VAR	error register	UNSIGNED8	RO
1600$_h$~03$_h$	RECORD	Receive PDO mapping	UNSIGNED32	RW
1A00$_h$~03$_h$	RECORD	Transmit PDO mapping	UNSIGNED32	RW

※ Only 1001$_h$ could be mapped to PDO.

6.3 Overview of Object Group 6000$_h$

Index	Object Type	Name	Data Type	Access	Mappable
603F$_h$	VAR	Error Code	UNSIGNED16	RO	Y
6040$_h$	VAR	Controlword	UNSIGNED16	RW	Y
6041$_h$	VAR	Statusword	UNSIGNED16	RO	Y
605B$_h$	VAR	Shutdown option code	INTEGER16	RW	N
605E$_h$	VAR	Fault reaction option code	INTEGER16	RW	N
6060$_h$	VAR	Modes of operation	INTEGER8	RW	Y
6061$_h$	VAR	Modes of operation display	INTEGER8	RO	Y
6062$_h$	VAR	Position demand value [PUU]	INTEGER32	RO	Y
6063$_h$	VAR	Position actual value [increment]	INTEGER32	RO	Y
6064$_h$	VAR	Position actual value	INTEGER32	RO	Y
6065$_h$	VAR	Following error window	UNSIGNED32	RW	Y
6067$_h$	VAR	Position windows	UNSIGNED32	RW	Y
6068$_h$	VAR	Position window time	UNSIGNED16	RW	Y
606B$_h$	VAR	Velocity demand value	INTEGER32	RO	Y
606C$_h$	VAR	Velocity actual value	INTEGER32	RO	Y
606D$_h$	VAR	Velocity window	UNSIGNED16	RW	Y
606E$_h$	VAR	Velocity window time	UNSIGNED16	RW	Y
606F$_h$	VAR	Velocity threshold	UNSIGNED16	RW	Y
6071$_h$	VAR	Target torque	INTEGER16	RW	Y
6072$_h$	VAR	Max torque	UNSIGNED16	RW	Y
6074$_h$	VAR	Torque demand value	INTEGER16	RO	Y
6075$_h$	VAR	Motor rated current	UNSIGNED32	RO	Y
6076$_h$	VAR	Motor rated torque	UNSIGNED32	RO	Y
6077$_h$	VAR	Torque actual value	UNSIGNED16	RO	Y
6078$_h$	VAR	Current actual value	INTEGER16	RO	Y
607A$_h$	VAR	Target position	INTEGER32	RW	Y

Index	Object Type	Name	Data Type	Access	Mappable
607C$_h$	VAR	Home Offset	INTEGER32	RW	Y
607D$_h$	ARRAY	Software position limit	INTEGER32	RW	Y
607E$_h$	VAR	Polarity	UNSIGNED8	RW	Y
607F$_h$	VAR	Max profile velocity	UNSIGNED32	RW	Y
6080$_h$	VAR	Max motor speed	UNSIGNED32	RW	Y
6081$_h$	VAR	Profile velocity	UNSIGNED32	RW	Y
6083$_h$	VAR	Profile acceleration	UNSIGNED32	RW	Y
6084$_h$	VAR	Profile deceleration	UNSIGNED32	RW	Y
6085$_h$	VAR	Quick stop deceleration	UNSIGNED32	RW	Y
6086$_h$	VAR	Motion profile type	INTEGER16	RW	Y
6087$_h$	VAR	Torque slope	UNSIGNED32	RW	Y
6093$_h$	ARRAY	Position factor	UNSIGNED32	RW	Y
6098$_h$	VAR	Homing method	INTEGER8	RW	Y
6099$_h$	ARRAY	Homing speeds	UNSIGNED32	RW	Y
609A$_h$	VAR	Homing acceleration	UNSIGNED32	RW	Y
60B0$_h$	VAR	Position offset	INTEGER32	RW	Y
60B1$_h$	VAR	Velocity offset	INTEGER32	RW	Y
60B2$_h$	VAR	Torque offset	INTEGER16	RW	Y
60B8$_h$	VAR	Touch probe function	UNSIGNED16	RW	Y
60B9$_h$	VAR	Touch probe status	UNSIGNED16	RO	Y
60BA$_h$	VAR	Touch probe pos1 pos value	INTEGER32	RO	Y
60BB$_h$	VAR	Touch probe pos1 neg value	INTEGER32	RO	Y
60BC$_h$	VAR	Touch probe pos2 pos value	INTEGER32	RO	Y
60BD$_h$	VAR	Touch probe pos2 neg value	INTEGER32	RO	Y
60C0$_h$	VAR	Interpolation sub mode select	INTEGER16	RW	Y
60C1$_h$	ARRAY	Interpolation data record	UNSIGNED16/32	RW	Y
60C2$_h$	RECORD	Interpolation time period	SIGNED8	RW	Y
60C5$_h$	VAR	Max acceleration	UNSIGNED32	RW	Y
60C6$_h$	VAR	Max deceleration	UNSIGNED32	RW	Y

Index	Object Type	Name	Data Type	Access	Mappable
60F2$_h$	VAR	Positioning option code	UNSIGNED16	RW	Y
60F4$_h$	VAR	Following error actual value	INTEGER32	RO	Y
60FC$_h$	VAR	Position demand value	INTEGER32	RO	Y
60FD$_h$	VAR	Digital inputs	UNSIGNED32	RO	Y
60FF$_h$	VAR	Target velocity	INTEGER32	RW	Y
6502$_h$	VAR	Supported drive modes	UNSIGNED32	RO	Y
Delta parameter definition					
2xxx	VAR	Parameter Mapping	INTEGER16/32	RW	Y

6.4 Details of Objects

Object 1000$_h$: Device Type

INDEX	1000$_h$
Name	device type
Object Code	VAR
Data Type	UNSIGNED32
Access	RO
PDO Mapping	No
Value Range	UNSIGNED32
Default Value	04020192 $_h$: A2 Series

Object 1001$_h$: Error Register

INDEX	1001$_h$
Name	error register
Object Code	VAR
Data Type	UNSIGNED8
Access	RO
PDO Mapping	Yes
Value Range	UNSIGNED8
Default Value	0

Object 1600$_h$ ~ 1604$_h$: Receive PDO Mapping Parameter

INDEX	1600$_h$ ~ 1603$_h$
Name	Receive PDO mapping
Object Code	RECORD
Data Type	PDO Mapping
Access	RW
PDO Mapping	No

Sub-Index	0
Description	Number of mapped application objects in PDO
Data Type	UNSIGNED8
Access	RW

PDO Mapping	No
Value Range	0: deactivated 1~8: activated
Default Value	0

Sub-Index	1~8
Description	PDO mapping for the nth application object to be mapped
Data Type	UNSIGNED32
Access	RW
PDO Mapping	No
Value Range	UNSIGNED32
Default Value	0

Object 1A00$_h$ ~ 1A04$_h$: Transmit PDO Mapping Parameter

INDEX	1A00$_h$ ~ 1A03$_h$
Name	Transmit PDO mapping
Object Code	RECORD
Data Type	PDO Mapping
Access	RW
PDO Mapping	No

Sub-Index	0
Description	Number of mapped application objects in PDO
Data Type	UNSIGNED8

Access	RW
PDO Mapping	No
Value Range	0: deactivated 1~8: activated
Default Value	0

Sub-Index	1~8
Description	PDO mapping for the nth application object to be mapped
Data Type	UNSIGNED32
Access	RW
PDO Mapping	No
Value Range	UNSIGNED32
Default Value	0

Object 1C12$_h$: RxPDO assign

INDEX	1C12$_h$
Name	RxPDO assign
Object Code	RECORD
Data Type	PDO Mapping assign
Access	RW
PDO Mapping	No

Sub-Index	0
Description	Number of assigned PDO mapping
Data Type	UNSIGNED8
Access	RW
PDO Mapping	No
Value Range	0: deactivated 1: One PDO mapping be assigned to SycManager2 for RxPDO
Default Value	1

Sub-Index	1
Description	Index of assigned PDO mapping
Data Type	UNSIGNED16
Access	RW
PDO Mapping	No

Value Range	1600 $_h$ to 1603 $_h$
Default Value	1601 $_h$

Object 1C13$_h$: TxPDO assign

INDEX	1C13$_h$
Name	TxPDO assign
Object Code	RECORD
Data Type	PDO Mapping assign
Access	RW
PDO Mapping	No

Sub-Index	0
Description	Number of assigned PDO mapping
Data Type	UNSIGNED8
Access	RW
PDO Mapping	No
Value Range	0: deactivated 1: One PDO mapping be assigned to SycManager3 for TxPDO
Default Value	1

Sub-Index	1
Description	Index of assigned PDO mapping
Data Type	UNSIGNED16
Access	RW
PDO Mapping	No
Value Range	1A00 $_h$ to 1A03 $_h$
Default Value	1A01 $_h$

Object 603F$_h$: Error code (error code of CANopen defined)

INDEX	603F$_h$
Name	Errorcode
Object Code	VAR
Data Type	UNSIGNED16
Access	RO
PDO Mapping	Yes
Value Range	UNSIGNED16

Default Value	0

Object 6040ₕ: Controlword

INDEX	6040ₕ
Name	Controlword
Object Code	VAR
Data Type	UNSIGNED16
Access	RW
PDO Mapping	Yes
Value Range	UNSIGNED16
Default Value	P1-01 = 0x0C, Default is 0x0004
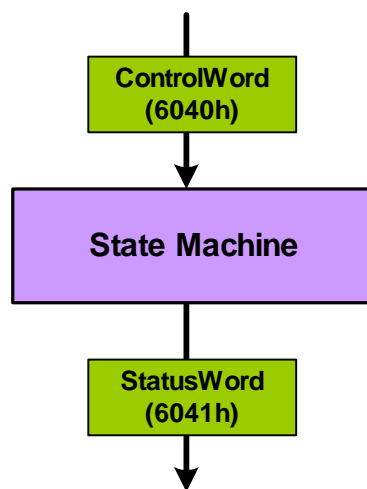
State machine in system context

Bit Definition

15~9	8	7	6~4	3	2	1	0
N/A	Halt	Fault reset	Operation mode specific	Enable operation	Quick Stop (B-contact)	Enable voltage	Switch on

Note：

The user needs to set 6040h to 0x0006->0x0007->0x000F for Servo On step by step.

Bit	Operation mode				
	PP	HM	IP	PV	PT
4	New set-point (positive trigger)	Homing operation start (positive trigger)	N/A	N/A	N/A
5	Change set immediately	N/A	N/A	N/A	N/A
6	Absolute(0) / relative(1)	N/A	N/A	N/A	N/A

Abbreviation:

PP   Profile Position Mode

HM   Homing Mode

IP   Interpolated Position Mode

PV   Profile Velocity Mode

PT   Profile Torque Mode
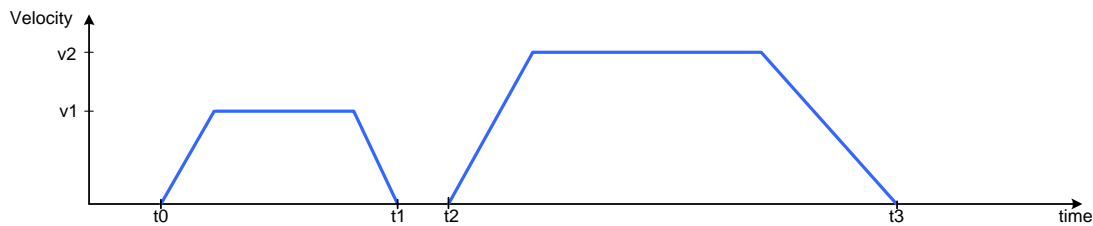
Single set-point
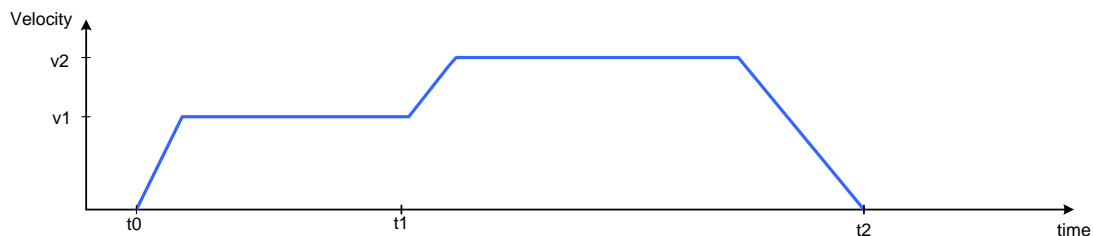
Change settings immediately

Object 6041$_h$: Statusword

INDEX	6041$_h$
Name	Statusword
Object Code	VAR
Data Type	UNSIGNED16
Access	RO
PDO Mapping	Yes
Value Range	UNSIGNED16
Default Value	0

Data Description

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

MSB   LSB

Bit Definition

0	Ready to switch on				
1	Switch on				
2	Operation enabled (status of servo on)				
3	Fault (the drive will servo off)				
4	Voltage enabled				
5	Quick stop				
6	Switch on disabled				
7	Warning (the drive is still servo on)				
8	N/A				
9	Remote				
10	Target reached				
11	Internal limit active (Not supported)				
	PP	HM	IP	PV	PT
12	Set-point acknowledge	Homing attained	IP mode active	Zero Speed	N/A
13	Following error	Homing error	N/A	N/A	N/A
14	N/A	N/A	N/A	N/A	N/A
15	N/A	N/A	N/A	N/A	N/A

Note:

Set-point acknowledge: Trajectory generator has assumed the positioning values

Homing attained: Homing mode carried out successfully

IP mode active: interpolated position mode active – mode is running in IP mode

Object 605B$_h$: Shutdown option code

INDEX	605B$_h$
Name	Shutdown option code
Object Code	VAR
Data Type	INTEGER16
Access	RW
PDO Mapping	Yes
Value Range	INTEGER16
Default Value	0
Comment	0:Disable drive function -1:Dynamic break enable

Object 605E$_h$: Fault reaction option code

INDEX	605E$_h$
Name	Fault reaction option code
Object Code	VAR
Data Type	INTEGER16
Access	RW
PDO Mapping	Yes
Value Range	INTEGER16
Default Value	2
Comment	0:Disable drive, motor is free to rotate 1:slow down on slow down ramp 2:slow down on quick stop ramp

Object 6060$_h$: Modes of operation

INDEX	6060$_h$
Name	Modes of operation
Object Code	VAR
Data Type	INTEGER8
Access	RW
PDO Mapping	Yes
Value Range	INTEGER8
Default Value	0
Comment	0: Reserved 1: Profile position mode 3: Profile velocity mode

	4: Profile torque mode 6: Homing mode 7: Interpolated position mode 8: Cyclic synchronous position mode 9: Cyclic synchronous velocity mode 10: Cyclic synchronous torque mode

Object 6061h: Modes of operation display

INDEX	6061h
Name	Modes of operation display
Object Code	VAR
Data Type	INTEGER8
Access	RW
PDO Mapping	Yes
Value Range	INTEGER8
Default Value	0

Object 6062h: Position demand value

INDEX	6062h
Name	Position demand value
Object Code	VAR
Data Type	INTEGER32
Access	RO
PDO Mapping	Yes
Value Range	INTEGER32
Default Value	0
Comment	Position command is calculated by Interpolation theory Unit: PUU

Object 6063h: Position demand value

INDEX	6063h
Name	Position actual value*
Object Code	VAR
Data Type	INTEGER32
Access	RO
PDO Mapping	Yes
Value Range	INTEGER32

Default Value	0
Comment	Unit: increments

Object 6064ₕ: Position actual value

INDEX	6064ₕ
Name	Position actual value
Object Code	VAR
Data Type	INTEGER32
Access	RO
PDO Mapping	Yes
Value Range	INTEGER32
Default Value	0
Comment	Unit: PUU

Object 6065ₕ: Following error window

INDEX	6065ₕ
Name	Following error window
Object Code	VAR
Data Type	UNSIGNED32
Access	RW
PDO Mapping	Yes
Value Range	UNSIGNED32
Default Value	3840000
Comment	Unit: PUU

Object 6067ₕ: Position window

INDEX	6067ₕ
Name	Position window
Object Code	VAR
Data Type	UNSIGNED32
Access	RW
PDO Mapping	Yes
Value Range	UNSIGNED32
Default Value	100
Comment	Unit: PUU

Object 6068$_h$: Position window time

INDEX	6068$_h$
Name	Position window time
Object Code	VAR
Data Type	UNSIGNED16
Access	RW
PDO Mapping	Yes
Value Range	UNSIGNED16
Default Value	0
Comment	Unit: millisecond

Object 606B$_h$: Velocity demand value

INDEX	606B$_h$
Name	Velocity demand value
Object Code	VAR
Data Type	INTEGER32
Access	RO
PDO Mapping	Yes
Value Range	INTEGER32
Comment	Unit: 0.1rpm

Object 606C$_h$: Velocity actual value

INDEX	606C$_h$
Name	Velocity actual value
Object Code	VAR
Data Type	INTEGER32
Access	RO
PDO Mapping	Yes
Value Range	INTEGER32
Comment	Unit: 0.1rpm

Object 606D_h: Velocity window

INDEX	606D_h
Name	Velocity window
Object Code	VAR
Data Type	INTEGER16
Access	RO
PDO Mapping	Yes
Value Range	0~3000
Default Value	100
Comment	Unit: 0.1rpm

Object 606E_h: Velocity window time

INDEX	606E_h
Name	Velocity window time
Object Code	VAR
Data Type	UNSIGNED16
Access	RW
PDO Mapping	Yes
Value Range	UNSIGNED16
Default Value	0
Comment	Unit: millisecond

Object 606F_h: Velocity threshold

INDEX	606F_h
Name	Velocity threshold
Object Code	VAR
Data Type	UNSIGNED16
Access	RW
PDO Mapping	Yes
Value Range	0~2000
Default Value	100
Comment	Unit: 0.1rpm

Object 6071ₕ: Target torque

INDEX	6071ₕ
Name	Target torque
Object Code	VAR
Data Type	INTEGER16
Access	RW
PDO Mapping	Yes
Value Range	-3000~3000
Default Value	0
Comment	Unit: one rated torque in a thousand

Object 6072ₕ: Max torque

INDEX	6072ₕ
Name	Max torque
Object Code	VAR
Data Type	UNSIGNED16
Access	RW
PDO Mapping	Yes
Value Range	0~3000
Default Value	3000
Comment	Unit: one rated torque in a thousand

Object 6074ₕ: Torque demand value

INDEX	6074ₕ
Name	Torque demand value
Object Code	VAR
Data Type	INTEGER16
Access	RO
PDO Mapping	Yes
Value Range	INTEGER16
Comment	Unit: one rated torque in a thousand

Object 6075$_h$: Motor rated current

INDEX	6075$_h$
Name	Motor rated current
Object Code	VAR
Data Type	UNSIGNED32
Access	RO
PDO Mapping	Yes
Value Range	UNSIGNED32
Comment	Unit: milliamp

Object 6076$_h$: Motor rated torque

INDEX	6076$_h$
Name	Motor rated torque
Object Code	VAR
Data Type	UNSIGNED32
Access	RO
PDO Mapping	Yes
Value Range	UNSIGNED32
Comment	Unit: one rated torque in a thousand

Object 6077$_h$: Torque actual value

INDEX	6077$_h$
Name	Torque actual value
Object Code	VAR
Data Type	INTEGER16
Access	RO
PDO Mapping	Yes
Value Range	INTEGER16
Comment	Unit: one rated torque in a thousand

Object 6078$_h$: Current actual value

INDEX	6078$_h$
Name	Current actual value
Object Code	VAR
Data Type	INTEGER16
Access	RO

PDO Mapping	Yes
Value Range	INTEGER16
Comment	Unit: one rated current in a thousand

Object 607A$_h$: Target position

INDEX	607A$_h$
Name	Target position
Object Code	VAR
Data Type	INTEGER32
Access	RW
PDO Mapping	Yes
Value Range	INTEGER32
Default Value	0
Comment	For Profile position mode 6060$_h$=1 Unit: PUU

Object 607C$_h$: Home offset

INDEX	607C$_h$
Name	Home offset
Object Code	VAR
Data Type	INTEGER32
Access	RW
PDO Mapping	Yes
Value Range	INTEGER32
Default Value	0
Comment	Unit : PUU

Home offset

Object 607D$_h$: Software position limit

INDEX	607D$_h$
Name	Software position limit
Object Code	ARRAY
Data Type	INTEGER32
Access	RW
PDO Mapping	Yes

Sub-Index	0
Description	Number of entries
Data Type	UNSIGNED8
Access	RO
PDO Mapping	Yes
Value Range	2
Default Value	2
Sub-Index	1
Description	Min position limit
Data Type	INTEGER32
Access	RW
PDO Mapping	Yes
Value Range	-2147483648 ~ +2147483647
Default Value	-2147483648
Comment	Unit: PUU

Sub-Index	2
Description	Max position limit
Data Type	INTEGER32
Access	RW
PDO Mapping	Yes
Value Range	-2147483648 ~ +2147483647
Default Value	+2147483647
Comment	Unit: PUU

Object 607F_h: Max profile velocity

INDEX	607F_h
Name	Max profile velocity
Object Code	VAR
Data Type	UNSIGNED32
Access	RW
PDO Mapping	Yes
Value Range	UNSIGNED32
Default Value	P1-55(rpm) * 10
Comment	Unit:0.1rpm

Object 6080_h: Max motor speed

INDEX	6080_h
Name	Max motor speed
Object Code	VAR
Data Type	UNSIGNED32
Access	RW
PDO Mapping	Yes
Value Range	UNSIGNED32
Default Value	P1-55(rpm)
Comment	Unit: rpm

Object 6081_h: Profile velocity

INDEX	6081_h
Name	Profile Velocity
Object Code	VAR
Data Type	UNSIGNED32
Access	RW
PDO Mapping	Yes
Value Range	UNSIGNED32
Default Value	10000
Comment	For Profile position mode 6060_h=1 Unit: PUU per second

Object 6083$_h$: Profile acceleration

INDEX	6083h
Name	Profile acceleration
Object Code	VAR
Data Type	UNSIGNED32
Access	RW
PDO Mapping	Yes
Value Range	1~UNSIGNED32
Default Value	200
Comment	For Profile position mode 6060h=1 & Profile velocity mode 6060h = 3 Unit: millisecond (time from 0rpm to 3000rpm)

Object 6084$_h$: Profile deceleration

INDEX	6084$_h$
Name	Profile deceleration
Object Code	VAR
Data Type	UNSIGNED32
Access	RW
PDO Mapping	Yes
Value Range	1~UNSIGNED32
Default Value	200
Comment	For Profile position mode 6060$_h$=1 & Profile velocity mode 6060h = 3 Unit: millisecond (time from 0rpm to 3000rpm)

Object 6085$_h$: Quick stop deceleration

INDEX	6085$_h$
Name	Quick stop acceleration
Object Code	VAR
Data Type	UNSIGNED32
Access	RW
PDO Mapping	Yes
Value Range	UNSIGNED32
Default Value	0
Comment	Unit: millisecond (time from 0rpm to 3000rpm)

Object 6086$_h$: Motion profile type

INDEX	6086$_h$
Name	Motion profile type
Object Code	VAR
Data Type	INTEGER16
Access	RW
PDO Mapping	Yes
Value Range	INTEGER16
Default Value	0

Object 6087$_h$: Torque slope

INDEX	6087$_h$
Name	Torque slope
Object Code	VAR
Data Type	UNSIGNED32
Access	RW
PDO Mapping	Yes
Value Range	UNSIGNED32
Default Value	0
Comment	Unit: millisecond (time from 0 to 100% rated torque)

Object 6093$_h$: Position factor

INDEX	6093$_h$
Name	Position factor
Object Code	ARRAY
Data Type	UNSIGNED32
Access	RW
PDO Mapping	Yes
Comment	Position factor = Numerator / Feed_constant

Sub-Index	0
Description	Number of entries
Data Type	UNSIGNED8
Access	RO

PDO Mapping	No
Value Range	2
Default Value	2

Sub-Index	1
Description	Numerator
Data Type	UNSIGNED32
Access	RW
PDO Mapping	Yes
Default Value	1
Comment	Same as P1-44

Sub-Index	2
Description	Feed_constant
Data Type	UNSIGNED32
Access	RW
PDO Mapping	Yes
Default Value	1
Comment	Same as P1-45

Object 6098$_h$: Homing method

INDEX	6098$_h$
Name	Homing method
Object Code	VAR
Data Type	INTEGER8
Access	RW
PDO Mapping	Yes
Value Range	0~35
Default Value	0
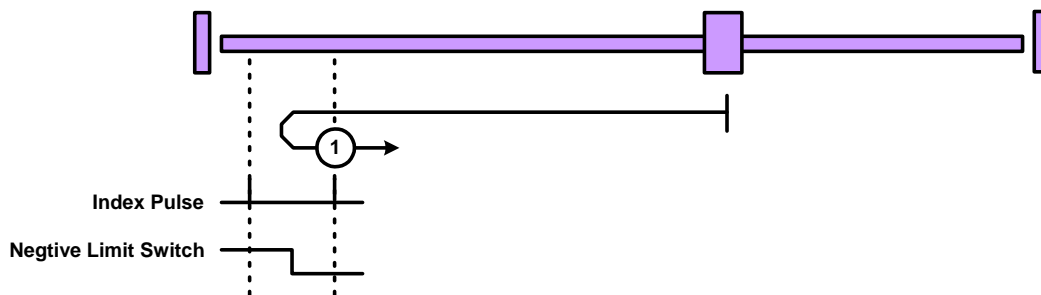
Method1：Homing on negative limit switch and index pulse
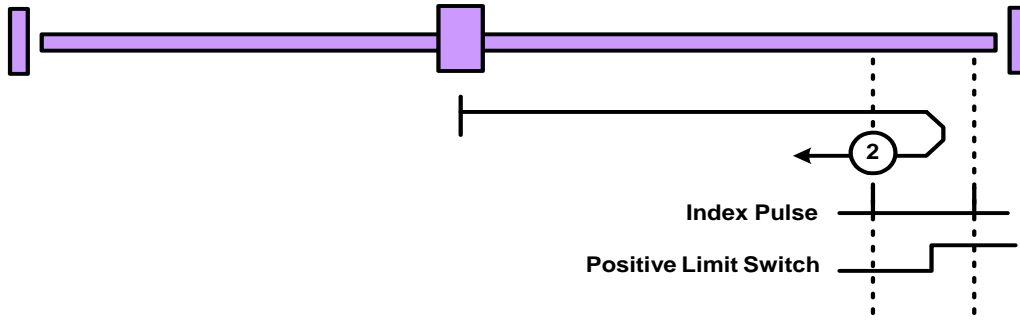
Method2：Homing on positive limit switch and index pulse
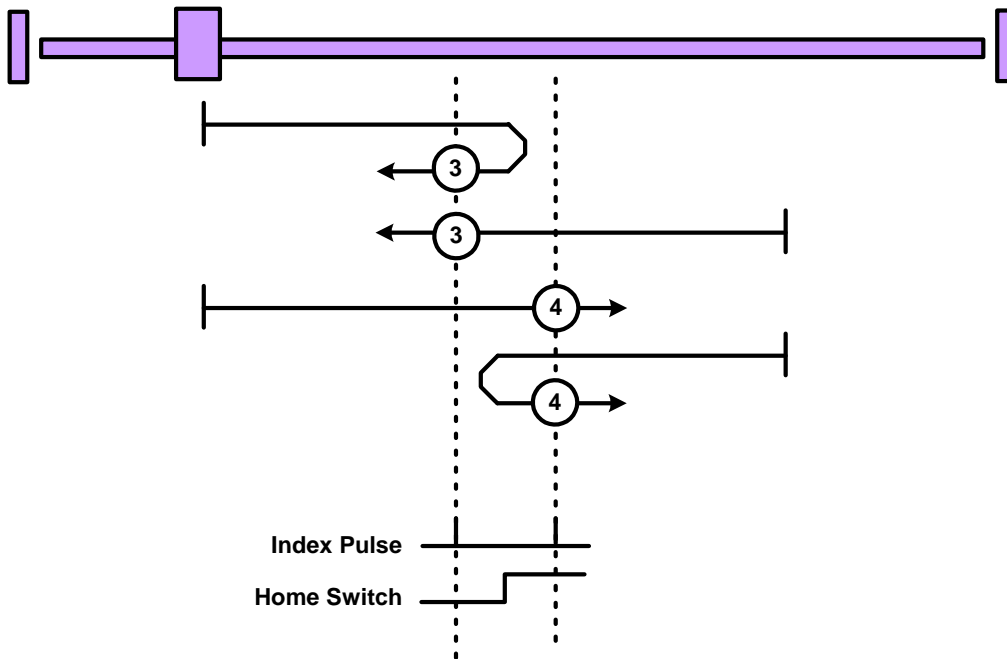
Method 3 and 4：Homing on positive home switch and index pulse
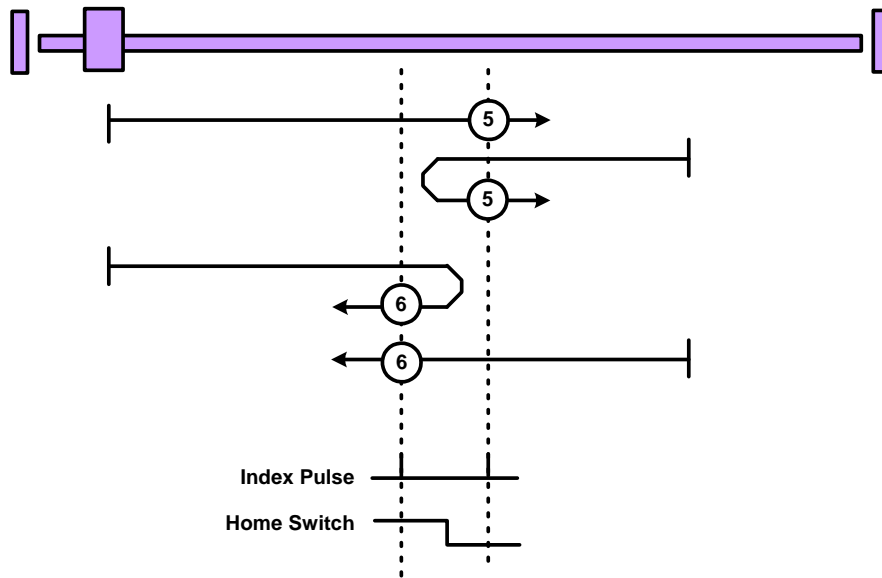
Method 5 and 6：Homing on negative home switch and index pulse
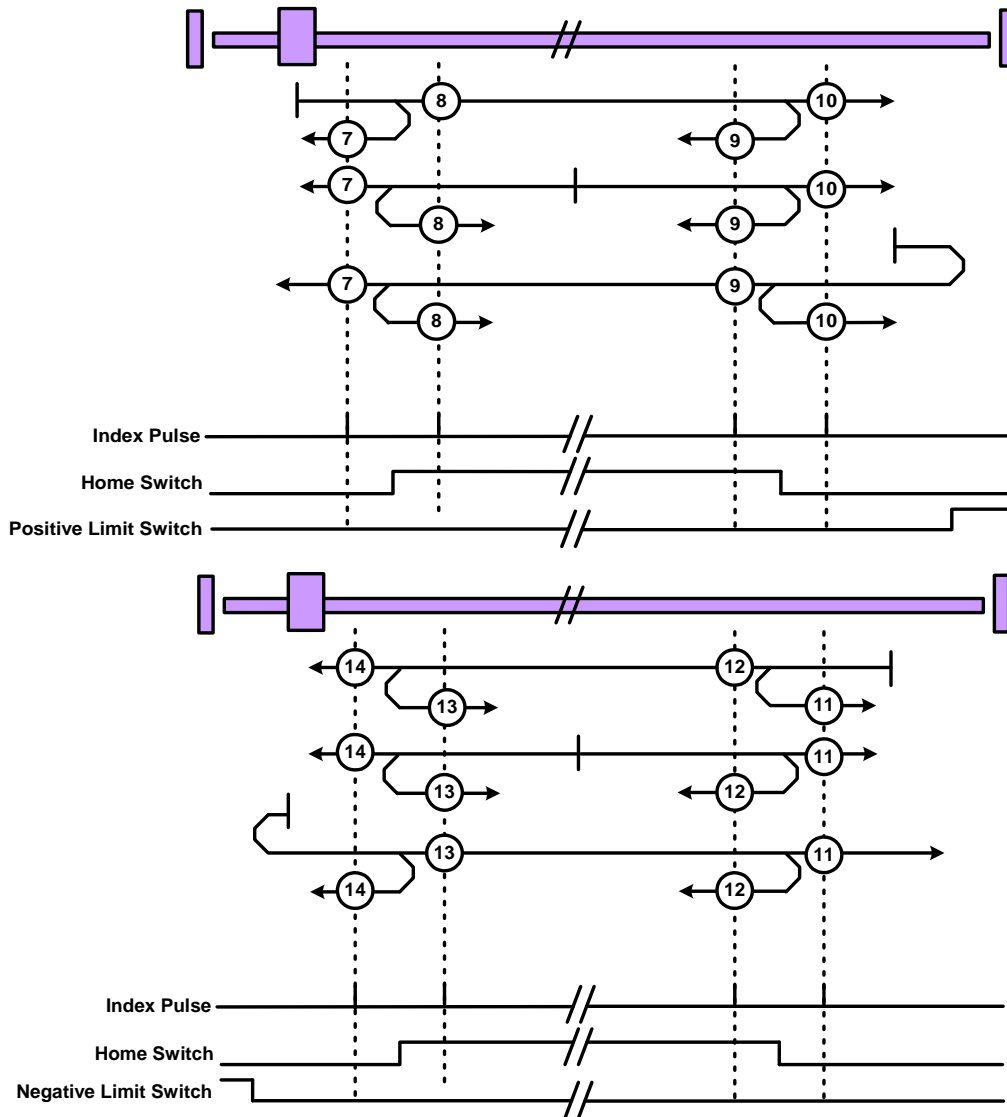
Method 7 to 14：Homing on home switch and index pulse

Method 15 and 16：Reserved (no picture)
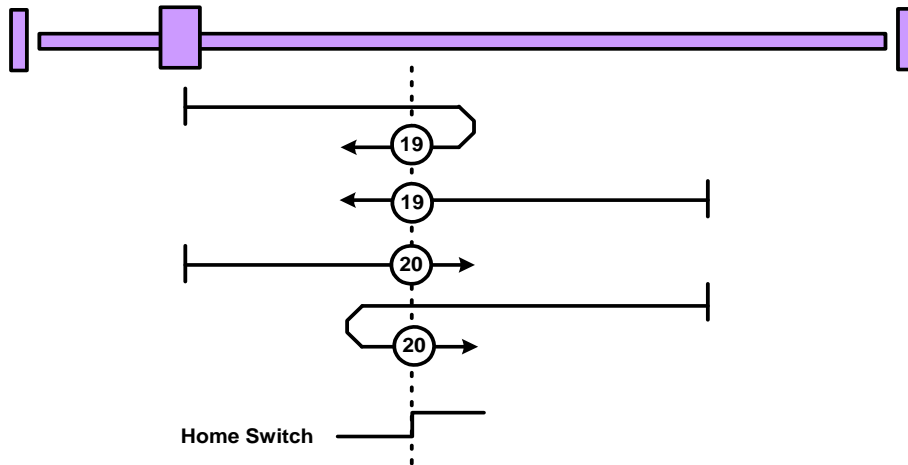
Method 17 to 30：Homing without an index pulse

Method 31 and 32：Reserved (no picture)
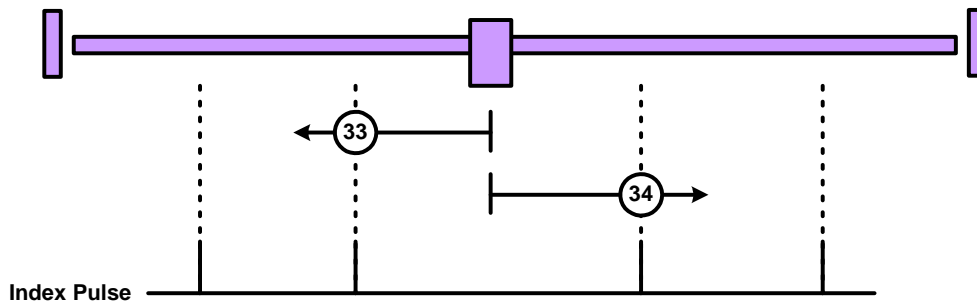
Method 33 to 34：Homing on index pulse

Method 35：Homing on current position (no picture)

Object 6099$_h$: Homing speed

INDEX	6099$_h$
Name	Homing speed
Object Code	ARRAY
Data Type	UNSIGNED32
Access	RW
PDO Mapping	Yes

Sub-Index	0
Description	Number of entries
Data Type	UNSIGNED8
Access	RO
PDO Mapping	Yes
Value Range	2

Default Value	2
Sub-Index	1
Description	Speed during search for switch
Data Type	UNSIGNED32
Access	RW
PDO Mapping	Yes
Value Range	1~2000rpm
Default Value	100
Comment	Unit:0.1rpm

Sub-Index	2
Description	Speed during search for zero
Data Type	UNSIGNED32
Access	RW
PDO Mapping	Yes
Value Range	1~500rpm
Default Value	20
Comment	Unit:0.1rpm

Object 609A$_h$: Homing acceleration

INDEX	609A$_h$
Name	Homing acceleration
Object Code	VAR
Data Type	UNSIGNED32
Access	RW
PDO Mapping	Yes
Value Range	UNSIGNED32
Default Value	100
Comment	Unit: millisecond (time of acc from 0rpm to 3000rpm)

Object 60B0ₕ: Position offset

INDEX	60B0ₕ
Name	Position offset
Object Code	VAR
Data Type	INTEGER32
Access	RW
PDO Mapping	Yes
Value Range	INTEGER32
Default Value	0
Comment	Unit: PUU

Object 60B1ₕ: Velocity offset

INDEX	60B1ₕ
Name	Velocity offset
Object Code	VAR
Data Type	INTEGER32
Access	RW
PDO Mapping	Yes
Value Range	INTEGER32
Default Value	0
Comment	Unit: 0.1rpm

Object 60B2ₕ: Torque offset

INDEX	60B2ₕ
Name	Torque offset
Object Code	VAR
Data Type	INTEGER16
Access	RW
PDO Mapping	Yes
Value Range	3000~-3000
Default Value	0
Comment	Unit: one rated torque in a thousand

Object 60B8$_h$: Touch probe function

INDEX	60B8$_h$
Name	Touch probe function
Object Code	VAR
Data Type	UNSIGNED16
Access	RW
PDO Mapping	Yes
Value Range	UNSIGNED16
Default Value	0
Comment	0

Object 60B9$_h$: Touch probe status

INDEX	60B9$_h$
Name	Touch probe status
Object Code	VAR
Data Type	UNSIGNED16
Access	RO
PDO Mapping	Yes
Value Range	UNSIGNED16
Default Value	0
Comment	0

Object 60BA$_h$: Touch probe pos1 pos value

INDEX	60BA$_h$
Name	Touch probe pos1 pos value
Object Code	VAR
Data Type	INTEGER32
Access	RO
PDO Mapping	Yes
Value Range	INTEGER32
Default Value	0
Comment	Unit: PUU

Object 60BB_h: Touch probe pos1 neg value

INDEX	60BB_h
Name	Touch probe pos1 neg value
Object Code	VAR
Data Type	INTEGER32
Access	RO
PDO Mapping	Yes
Value Range	INTEGER32
Default Value	0
Comment	Unit: PUU

Object 60BC_h: Touch probe pos2 pos value

INDEX	60BC_h
Name	Touch probe pos2 pos value
Object Code	VAR
Data Type	INTEGER32
Access	RO
PDO Mapping	Yes
Value Range	INTEGER32
Default Value	0
Comment	Unit: PUU

Object 60BD_h: Touch probe pos2 neg value

INDEX	60BD_h
Name	Touch probe pos2 neg value
Object Code	VAR
Data Type	INTEGER32
Access	RO
PDO Mapping	Yes
Value Range	INTEGER32
Default Value	0
Comment	Unit: PUU

Object 60C0ₕ: Interpolation sub mode select

INDEX	60C0$_h$
Name	Interpolation sub mode select
Object Code	VAR
Data Type	INTEGER16
Access	RW
PDO Mapping	Yes
Value Range	INTEGER16
Default Value	0
Comment	0: manufacturer specific (Linear interpolation -- **no** need the Pos Difference [OD-60C1sub2]) -1: manufacturer specific (Delta definition -- need pos difference [OD-60C1sub2])

Object 60C1ₕ: Interpolation data record

INDEX	60C1$_h$
Name	Interpolation data record
Object Code	ARRAY
Data Type	INTEGER32
Access	RW
PDO Mapping	Yes
Comment	Set this record by PDO every T msec before SYNC message Where T is specified by 60C2$_h$:01$_h$

Sub-Index	0
Description	Number of entries
Data Type	UNSIGNED8
Access	RO
PDO Mapping	No
Value Range	2
Default Value	2

Sub-Index	1
Description	Pos_Cmd
Data Type	INTEGER32

Access	RW
PDO Mapping	Yes
Value Range	INTEGER32
Default Value	0
Comment	Unit: 32-bit CMD_PUU
Sub-Index	2
Description	Velocity – Pos_Cmd difference
Data Type	INTEGER16
Access	RW
PDO Mapping	Yes
Value Range	INTEGER16
Default Value	0
Comment	$\triangle X_i = (X_{i+1} - X_{i-1})/2$ (It is also the same as velocity.) Unit: PUU

Object 60C2$_h$: Interpolation time period

INDEX	60C2$_h$
Name	Interpolation time period
Object Code	RECORD
Data Type	UNSIGNED8
Access	RW
PDO Mapping	Yes
Comment	The unit of **the interpolation time unit** is given in $10^{interpolation\ time\ index}$ seconds

Sub-Index	0
Description	Number of entries
Data Type	UNSIGNED8
Access	RO
PDO Mapping	No
Value Range	2
Default Value	2

Sub-Index	1
Description	Interpolation time units
Data Type	UNSIGNED8
Access	RW

PDO Mapping	Yes
Value Range	UNSIGNED8
Default Value	1

Sub-Index	2
Description	Interpolation time index
Data Type	INTEGER8
Access	RW
PDO Mapping	Yes
Value Range	-128~63
Default Value	-3

Object 60C5$_h$: Max acceleration

INDEX	60C5$_h$
Name	Max acceleration
Object Code	VAR
Data Type	UNSIGNED32
Access	RW
PDO Mapping	Yes
Value Range	1 ~ 65500
Default Value	200
Comment	Unit: millisecond (min. time from 0 rpm to 3000 rpm)

Object 60C6$_h$: Max deceleration

INDEX	60C6$_h$
Name	Max deceleration
Object Code	VAR
Data Type	UNSIGNED32
Access	RW
PDO Mapping	Yes
Value Range	1 ~ 65500
Default Value	200
Comment	Unit: millisecond (min. time from 3000 rpm to 0 rpm)

Object 60F2$_h$: Positioning option code

INDEX	60F2$_h$
Name	Positioning option code
Object Code	VAR
Data Type	UNSIGNED16
Access	RW
PDO Mapping	Yes
Value Range	UNSIGNED16
Default Value	0

Object 60F4$_h$: Following error actual value

INDEX	60F4$_h$
Name	Following error actual value
Object Code	VAR
Data Type	INTEGER32
Access	RO
PDO Mapping	Yes
Value Range	INTEGER32
Comment	Unit: PUU

Object 60FC$_h$: Position demand value*

INDEX	60FC$_h$
Name	Position demand value*
Object Code	VAR
Data Type	INTEGER32
Access	RO
PDO Mapping	Yes
Value Range	INTEGER32
Comment	Unit: increment

Object 60FD$_h$: Digital inputs

INDEX	60FD$_h$
Name	Digital inputs
Object Code	VAR
Data Type	UNSIGNED32
Access	RO
PDO Mapping	Yes
Value Range	UNSIGNED32
Default Value	0

31 3	2	1	0
reserved	home switch	positive limit switch	negative limit switch
MSB			LSB

Object 60FF$_h$: Target velocity

INDEX	60FF$_h$
Name	Target velocity
Object Code	VAR
Data Type	INTEGER32
Access	RW
PDO Mapping	Yes
Value Range	INTEGER32
Comment	Unit: 0.1rpm

Object 6502$_h$: Supported drive modes

INDEX	6502$_h$
Name	Supported drive modes
Object Code	VAR
Data Type	UNSIGNED32
Access	Ro
PDO Mapping	Yes
Value Range	UNSIGNED32
Default Value	3ED$_h$

31 16	15 7	6	5	4	3	2	1	0
Manufacturer specific	reserved	ip	hm	reserved	tq	pv	vl	pp
MSB								LSB

Object 2xxx_h: Manufacturer parameter

INDEX	2xxx$_h$
Name	Manufacturer parameter
Object Code	VAR
Data Type	INTEGER16/INTEGER32
Access	RW
PDO Mapping	Yes
Value Range	NTEGER16/INTEGER32
Default Value	N/A

Object 2xxx is defined to parameter.

If users desire to use CANopen protocol for simulate Keypad press, they could read and write Keypad parameters via SDO protocol.

Pa-bc <==> 2aBC$_h$

'**BC'** is hexadecimal format of '**bc**'
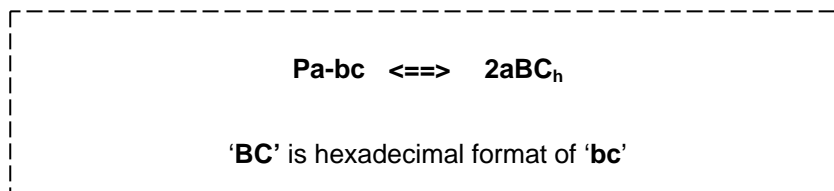
Users could read the Index first for knowing the Length of Parameter and then change the data by SDO or PDO.

Example 1: Object 2309$_h$: EtherCAT Synchronization Setting【P3-09】

INDEX	2309$_h$
Name	EtherCAT Synchronization Setting
Object Code	VAR
Data Type	INTEGER16
Access	RW
PDO Mapping	Yes
Value Range	INTEGER16
Default Value	1512 $_h$

Example 2: Object 212C$_h$: Electronic Gear 【P1-44】

INDEX	212C$_h$
Name	Electronic Gear
Object Code	VAR
Data Type	INTEGER32
Access	RW
PDO Mapping	Yes
Value Range	INTEGER32

(This page is intentionally left blank.)

Chapter 7 Safety Function (Safe Torque Off, STO)

7.1 Description of Terminal Block
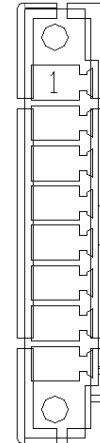
CN-STO Connector (male)

Pin No	Terminal Symbol	Function and Description
*1	COM+	VDD (24V) power is identical to pin 5 of CN1
2	STO_A	STO input pin A+
3	/STO_A	STO input pin A-
4	STO_B	STO input pin B+
5	/STO_B	STO input pin B-
6	FDBK_A	STO alarm output pin A, BJT Output Max. Rating : 80 VDC, 0.5 A
7	FDBK_B	STO alarm output pin B, BJT Output Max. Rating : 80 VDC, 0.5 A
8	COM-	VDD (24V) power ground

⚠ WARNING

➢ *1: Do not apply to dual power or it may damage the servo drive.

STO with safety relay:
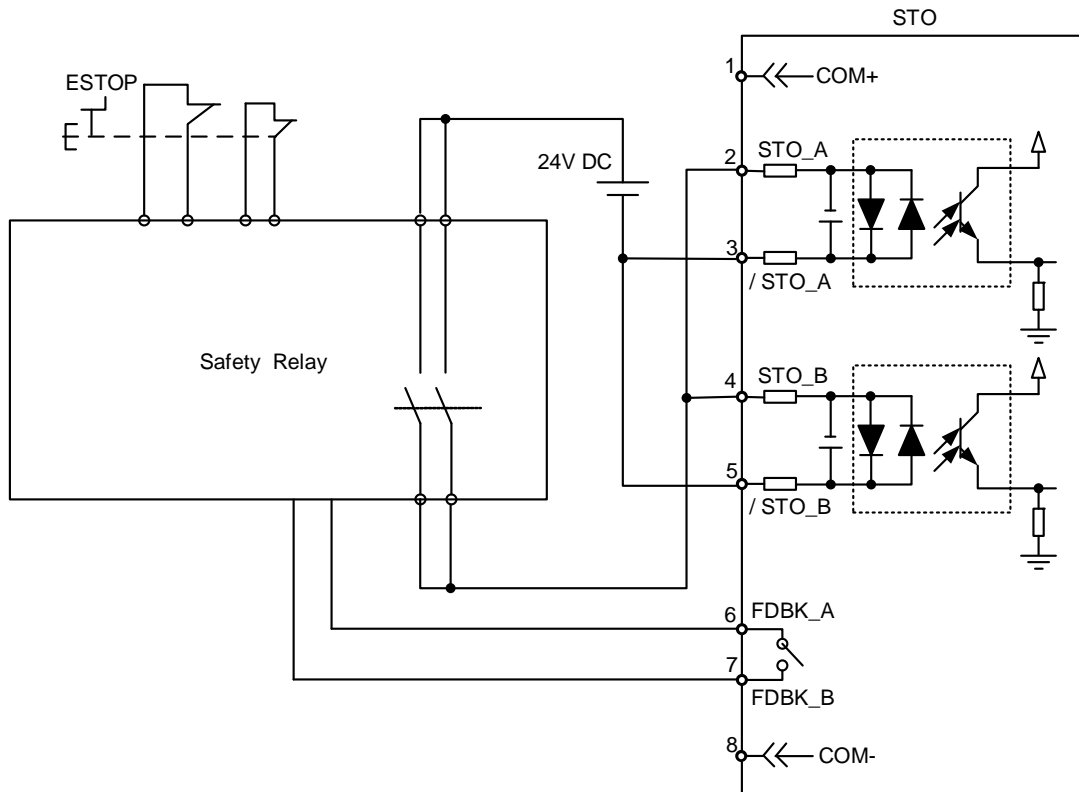
STO Disable:
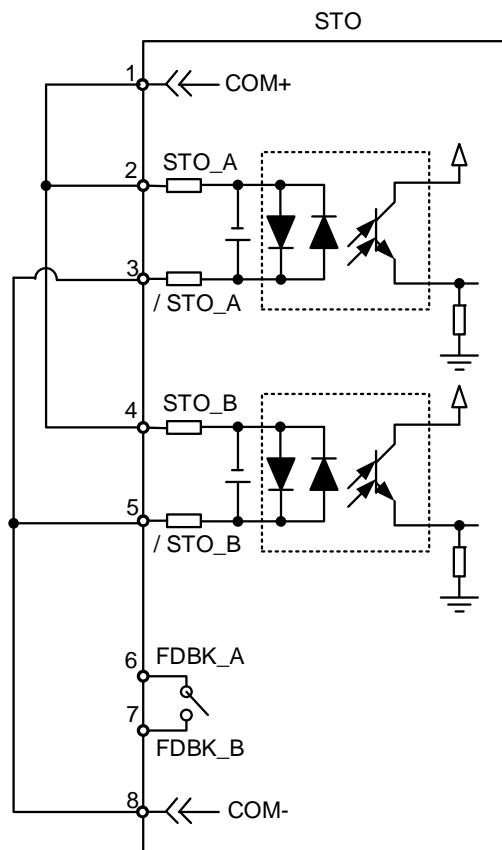
7.1.1 Functional Safety Standard and Certificates

Pleasee refer to Chapter 9.

7.2   STO Safety Function

Fault Rate of Safety Function

Item	Definition	Standard	Features
SFF	Safe Failure Fraction	IEC61508	Channel 1: 80.08% Channel 2: 68.91%
HFT (Type A subsystem)	Hardware Fault Tolerance	IEC61508	1
SIL	Safety Integrity Level	IEC61508	SIL 2
		IEC62061	SILCL 2
PFH	Average frequency of dangerous failure [h-1]	IEC61508	9.56×10^{-10}
PFD_{av}	Probability of Dangerous Failure on Demand	IEC61508	4.18×10^{-6}
Category	Category	ISO13849-1	Category 3
PL	Performance Level	ISO13849-1	d
$MTTF_d$	Mean time to dangerous failure	ISO13849-1	High
DC	Diagnostic Coverage	ISO13849-1	Low

Theorem of Safety Function

Two individual hardware circuits control the drive signal of motor current so as to cut off power supply to avoid generating motor cogging if needed.
See table 1 for the description of theorem
Table 1:   ON =24V   OFF= 0V

Signal	Channel	Status of Opto-Isolator			
STO signal	STO_A ~ / STO_A	ON	ON	OFF	OFF
	STO_B ~ / STO_B	ON	OFF	ON	OFF
Servo Drive Output Status		Ready	STO_B lost (AL502) (Torque off)	STO_A lost (AL501) (Torque off)	STO Mode (Torque off)

(1) Status Description of STO Alarm:

See the figure below. When the motor runs properly (SERVO ON), if STO_A and STO_B signal (which is also called safety signal) is lost for 10 ms at the same time, AL500 occurs. Then, servo drive will be in Servo Off status.
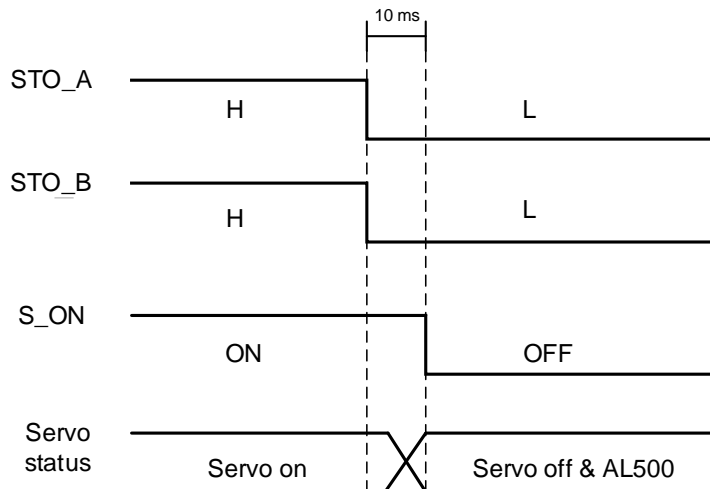
See the figure below. When the motor runs properly (SERVO ON), when one of the safety signal is lost for 1 s, AL501 or AL502 will occur. Then, servo drive will be in SERVO OFF status.
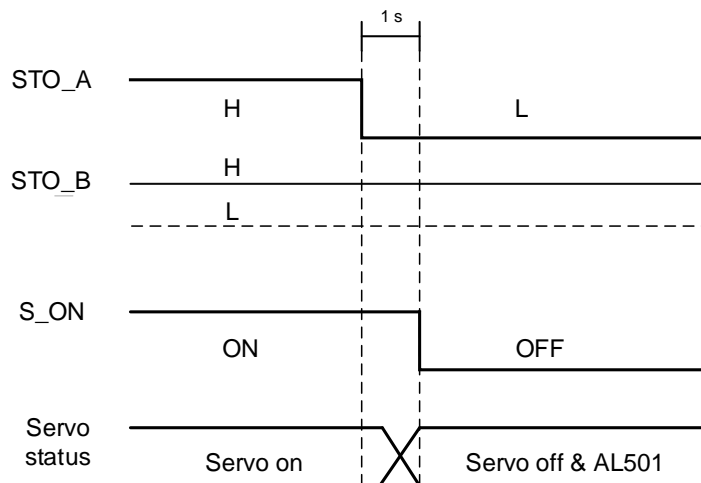
7.3 Related Parameter Descriptions of STO Function

Through the setting of P2-93, users can determine FDBK status and if FDBK will latch when STO alarm occurs. Please refer to the following figure for the setting of P2-93:

P2-93 = X X 1 0

Not in use

1: FDBK no latch
2: FDBK latch

0: Logic A
1: Logic B
2: Logic C
3: Logic D
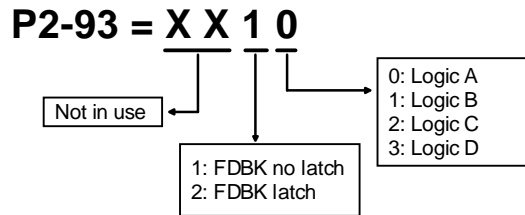
Description of STO Function:
See the table below. Four logics (Logic A, Logic B, Logic C, Logic D) are provided to standardize FDBK status when different STO alarm occurs. Customers can select the corresponding logic according to the demands. (In this table, Open means FDBK+ and FDBK- of CN8 are open circuit. Take Logic C as the example. When AL500 occurs, FDBK+ and FDBK- of CN8 are short circuited.)

| Status of Servo Drive		FDBK_A & FDBK_B Status							
		Logic A		Logic B		Logic C		Logic D	
Parameter P2-93		XX10	XX20	XX11	XX21	XX12	XX22	XX13	XX23
FDBK Behavior		No Latch	Latch	No Latch	Latch	No Latch	Latch	No Latch	Latch
No STO alarm occurs		Open		Close		Open		Close	
Alarm occurs	AL500 occurs	Close		Open		Close		Open	
	AL501 occurs	Close		Open		Open		Close	
	AL502 occurs	Close		Open		Open		Close	
	AL503 occurs	Close		Open		Open		Close	

Open = open circuit; Close = close circuit

If FDBK is latched, when STO alarm occurs, status of FDBK will remain even when the alarm has been cleared. Please note that when more than one alarm occur, the drive panel will only display AL500.

● Example of Latch:
   If Logic C P2-93 = XX22 is set, the FDBK status will be close when safety signal is lost and AL005 occurs.
   1. Since FDBK is selected as Latch, even when the safety signal is back to normal, FDBK status still remains close. Use the approaches below to reset.
      i. Reconnect power supply. FDBK status returns to open.
      ii. Do not reconnect power supply. Instead, set P2-93=XX12 to make FDBK status return to open. Then set P2-93=XX22 again. This step is to set FDBK behavior to Latch.

2. After the FDBK status restores, alarms can be cleared by normal corrective actions. In this case, AL500 can be cleared by DI: Alm Reset.

● Example of No Latch:
If Logic C P2-93 = XX12 is set, the FDBK status will be close when the safety signal is lost and AL005 occurs.

1. Since FDBK is selected as No Latch, safety signals return to normal and the FDBK status automatically changes from short-circuited to normal when AL500 occurs. Do not need to set P2-93 to XX12 again.

2. After the FDBK status restores, alarms can be cleared by normal corrective actions. In this case, AL500 can be solved by DI: Alm Reset.

P2-93	STO	STO FDBK Control		Address︰02BAH 02BCH
Interface︰	Panel / Software	Communication	Reference︰	-
Default︰	0		Control Mode︰	ALL
Unit︰	-		Range︰	-
Format︰	DEC		Data Size︰	16-bit

Settings︰
BIT0: Select the logic for FDBK status
BIT1: Determine if FDBK should be Latched

7.4   Related Alarm Descriptions of STO Function

Display	Alarm Name	Alarm Description	Corrective Actions	Corresponding DO	Servo Status
AL500	STO function is enabled	Safety function (STO) is manually enabled. Please check the causes.	DI.ARST or write 0 into P0-01 or 0x6040.Fault Reset	ALM	Servo Off
AL501	STO_A lost (signal lost or error)	Make sure the wiring of STO_A is correct	DI.ARST or write 0 into P0-01 or 0x6040.Fault Reset	ALM	Servo Off
AL502	STO_B lost (signal lost or error)	Make sure the wiring of STO_B is correct	DI.ARST or write 0 into P0-01 or 0x6040.Fault Reset	ALM	Servo Off
AL503	STO_error	Internal circuit of STO_A ~ / STO_A and STO_B ~ / STO_B is diagnosed as error	DI.ARST or write 0 into P0-01 or 0x6040.Fault Reset	ALM	Servo Off

Causes and Corrective Actions:

AL500: STO Function is enabled

Causes	Checking Method	Corrective Actions
Safety function (STO) is enabled	Safety function (STO) is enabled. Please check the causes.	DI.ARST or write 0 into P0-01 or 0x6040.Fault Reset

AL502: STO_A lost (Signal is lost or error)

Causes	Checking Method	Corrective Actions
STO_A loses enable signal or STO_A signal does not synchronize with STO_B signal for more than 1 second.	Make sure the wiring of STO_A is correct.	DI.ARST or write 0 into P0-01 or 0x6040.Fault Reset

AL502: STO_B lost (Signal is lost or error)

Causes	Checking Method	Corrective Actions
STO_B loses enable signal or STO_A signal does not synchronize with STO_B signal for more than 1 second.	Make sure the wiring of STO_B is correct.	DI.ARST or write 0 into P0-01 or 0x6040.Fault Reset

AL503: STO_error

Causes	Checking Method	Corrective Actions
STO self-diagnostic error	Check if the wiring between STO_A and STO_B is correct.	STO circuit error. Please contact with distributors.

(This page is intentionally left blank.)

Chapter 8 Alarm List

8.1   EtherCAT Communication Fault Messages

Emergency Object

Byte	0	1	2	3	4	5	6	7
Content	Emergency Error Code		Error register	Panel Alarm Code		N/A		

Fault Messages (If ALARM code is not showed here, please refer to the User Manual)			
Display	Fault Name	Fault Description	Clearing Method
AL185	Communication disconnected	EtherCAT link is disconnected	6040h fault reset
AL180	CANopen RxPDO timeout (Servo Off)	The drive dose not receive any RxPDO within three communication cycle times	6040h fault reset
AL122	Sub-index error occurs when accessing CANopen PDO object.	The specified Sub-index in the message does not exist.	6040h fault reset
AL123	Data type (size) error occurs when accessing CANopen PDO object.	The data length in the message does not match the specified object.	6040h fault reset
AL124	Data range error occurs when accessing CANopen PDO object.	The data in the message has exceeded the data range of the specified object.	6040h fault reset
AL125	CANopen PDO object is read-only and write-protected.	The specified object in the message is read-only and write-protected (cannot be changed).	6040h fault reset
AL126	CANopen PDO object does not support PDO.	The specified object in the message does not support PDO.	6040h fault reset

AL127	CANopen PDO object is write-protected when Servo On.	The specified object in the message is write-protected (cannot be changed) when Servo On.	6040h fault reset
AL128	Error occurs when reading CANopen PDO object from EEPROM.	An error occurs when loading the default settings from EEPROM at start-up. All CANopen object returns to the default setting automatically.	6040h fault reset
AL129	Error occurs when writing CANopen PDO object into EEPROM.	An error occurs when writing the current settings into EEPROM.	6040h fault reset
AL130	EEPROM invalid address range	The data amount saved in EEPROM has exceeded the space determined by the firmware. Maybe the firmware version has been upgraded, and the data of old firmware version saved in EEPROM cannot be used.	6040h fault reset
AL131	EEPROM checksum error	The data saved in EEPROM has been damaged and all CANopen objects return to the default settings automatically.	6040h fault reset
AL132	Password error	The parameter is password protected when using CANopen communication to access the parameter. The users must enter the valid password to unlock the parameter.	6040h fault reset
AL201	CANopen load/save 1010/1011 error	Fail while read/write data from/to EEPROM	P2-08 = 10 or P2-08 = 30,28 after firmware upgrade
AL3E1	CANopen SYNC failed (Servo Off)	The synchronous communication with the external controller has failed.	6040h fault reset
AL3E2	CANopen SYNC signal error (Servo Off)	The CANopen SYNC signal is received too early.	6040h fault reset
AL3E3	CANopen SYNC time out (Servo Off)	The CANopen SYNC signal is not received within the specified time.	6040h fault reset
AL3E4	CANopen IP command failed (Servo Off)	Internal command of CANopen IP mode cannot be sent and received.	6040h fault reset

AL3E5	SYNC period error (Servo Off)	Object 0x1006 data error. SYNC period 1006h value is invalid.	6040h fault reset
AL500	Safe torque off (Servo Off)	The safety function (STO) is enabled. STO_A and STO_B change state simultaneously.	6040h fault reset
AL501	STO_A lost (Servo Off)	STO_A is de-energized and STO_B is energized. STO_A signal does not synchronize with STO_B signal for more than 1 second, please check wiring contact or STO safety relay.	6040h fault reset
AL502	STO_B lost (Servo Off)	STO_A is energized and STO_B is de-energized. STO_A signal does not synchronize with STO_B signal for more than 1 second, please check wiring contact or STO safety relay.	6040h fault reset
AL503	STO error (Servo Off)	STO self-diagnostic error	6040h fault reset

8.2   Error Code Table

Display	Description	32bit-ErrorCode (16bit-ErrorCode + 16bit-Additional Info)
AL001	Overcurrent	2310-0001$_h$
AL002	Overvoltage	3110-0002$_h$
AL003	Undervoltage	3120-0003$_h$
AL004	Motor error	7122-0004$_h$
AL005	Regeneration error	3210-0005$_h$
AL006	Overload	3230-0006$_h$
AL007	Overspeed	8400-0007$_h$
AL008	Abnormal pulse control command	8600-0008$_h$
AL009	Excessive deviation	8611-0009$_h$
AL010	Reserved	0000-0010$_h$
AL011	Encoder error	7305-0011$_h$
AL012	Adjustment error	6320-0012$_h$
AL013	Emergency stop activated	5441-0013$_h$
AL014	Reverse limit switch error	5443-0014$_h$
AL015	Forward limit switch error	5442-0015$_h$
AL016	IGBT temperature error	4210-0016$_h$
AL017	Memory error	5330-0017$_h$
AL018	Encoder output error	7306-0018$_h$
AL019	Serial communication error	7510-0019$_h$
AL020	Serial communication time out	7520-0020$_h$
AL021	Reserved	Reserved
AL022	Input power phase loss	3130-0022$_h$
AL023	Early warning for overload	3231-0023$_h$
AL024	Encoder initial magnetic field error	7305-0024$_h$
AL025	Encoder internal error	7305-0025$_h$
AL026	Encoder internal error	7305-0026$_h$
AL027	Encoder data error	7305-0027$_h$

AL030	Motor protection error	7121-0030$_h$
AL031	U,V,W wiring error	3300-0031$_h$
AL040	Full-closed loop excessive deviation	8610-0040$_h$
AL099	DSP firmware upgrade	5500-0099h
AL201	CANopen Data Initial Error	6310-0201$_h$
AL283	Forward software limit	5444-0283$_h$
AL285	Reverse software limit	5445-0285$_h$
AL185	EtherCAT connection error (Servo Off)	8120-0185$_h$
AL180	Node guarding or Heartbeat error (Servo Off)	8130-0180$_h$
AL122	Sub-index error occurs when accessing CANopen PDO object.	8200-0122$_h$
AL123	Data type (size) error occurs when accessing CANopen PDO object.	8200-0123$_h$
AL124	Data range error occurs when accessing CANopen PDO object.	8200-0124$_h$
AL125	CANopen PDO object is read-only and write-protected.	8200-0125$_h$
AL126	CANopen PDO object does not support PDO.	8200-0126$_h$
AL127	CANopen PDO object is write-protected when Servo On.	8200-0127$_h$
AL128	Error occurs when reading CANopen PDO object from EEPROM.	8200-0128$_h$
AL129	Error occurs when writing CANopen PDO object into EEPROM.	8200-0129$_h$
AL130	EEPROM invalid address range.	8200-0130$_h$
AL131	EEPROM checksum error.	8200-0131$_h$
AL132	EEPROM zone error.	8200-0132$_h$
AL201	CANopen load/save 1010/1011 error	6310-0201$_h$
AL3E1	CANopen SYNC failed (Servo Off)	6200-03E1$_h$
AL3E2	CANopen SYNC signal error (Servo Off)	6200-03E2$_h$
AL3E3	CANopen SYNC time out (Servo Off)	6200-03E3$_h$

AL3E4	CANopen IP command failed (Servo Off)	6200-03E4$_h$
AL3E5	SYNC period error (Servo Off)	6200-03E5$_h$
AL500	Safe torque off (Servo Off)	9000-0500$_h$
AL501	STO_A lost (Servo Off)	9000-0501$_h$
AL502	STO_B lost (Servo Off)	9000-0502$_h$
AL503	STO error (Servo Off)	9000-0503$_h$

8.3   SDO Error Message Abort Codes

Abort Code	Description
05040001 h	Client/server command specifier not valid or unknown
06010002 h	Attempt to write a read-only object
06020000 h	Object does not exist in the object dictionary
06040041 h	Object cannot be mapped to PDO
06040042 h	The number and the length of the objects to be mapped would exceed PDO length
06060000 h	Access failed due to a hardware error (store or restore error)
06070010 h	Data type does not match; length of the service parameter does not match
06090011 h	Sub-index does not exist
06090030 h	Value range of parameter exceeded (only for writing access)
08000000 h	General error
080000a1 h	Object error when reading from EEPROM
080000a2 h	Object error when writing to EEPROM
080000a3 h	Invalid Range when accessing EEPROM
080000a4 h	Checksum error when accessing EEPROM
080000a5 h	Password error when writing encryption zone
08000020 h	Data cannot be transferred or stored in the application (store or restore signature error)
08000021 h	Data cannot be transferred or stored in the application because of the local control (store or restore in wrong state)
08000022 h	Object is on the fly

(This page is intentionally left blank.)

Chapter 9 Reference

1. CANopen Application Layer and Communication Profile, CiA Draft Standard 301, Version 4.02, Date: 13 February 2002

2. CANopen Device Profile Drives and Motion Control, CiA Draft Standard Proposal 402, Version 2.0, Date: 26 July 2002

# EC Type-Examination Certificate

Reg.-No.: 01/205/5429.00/15

Product tested	Safety Function "Safe Torque Off" (STO)	**Certificate holder**	Delta Electronics, Inc. 18 Xinglong Road Taoyuan County Taoyuan City 33068 Taiwan, R.O.C.

Type designation
within the drive series VFD-C, VFD-CP, VFD-CT, VFD-CH, VFD-HH, DPD, VFD-ED-S and ASD-A2.
Details see Annex (Version Release List)

Codes and standards

IEC 61800-5-2:2007
IEC 61800-5-1:2007 (in extracts)
IEC 61800-3:2012
IEC 62061:2012

EN ISO 13849-1:2008 + AC:2009
IEC 60204-1:2009 (in extracts)
IEC 61508 Parts 1-7:2010

Intended application
The safety function complies with the requirements of the relevant standards (Cat. 3 / PL d acc. to EN ISO 13849-1, SIL CL 2 acc. to IEC 62061 / IEC 61508) and can be used in applications up to PL d acc. to EN ISO 13849-1 and SIL 2 acc. to IEC 62061 / IEC 61508.

Specific requirements
The instructions of the associated Installation and Operating Manual shall be considered.

It is confirmed, that the product under test complies with the requirements for machines defined in Annex I of the EC Directive 2006/42/EC.

Valid until 2020-01-08

The issue of this certificate is based upon an examination, whose results are documented in Report No. 968/FSP 1074.00/15 dated 2015-01-08.
This certificate is valid only for products which are identical with the product tested. It becomes invalid at any change of the codes and standards forming the basis of testing for the intended application.
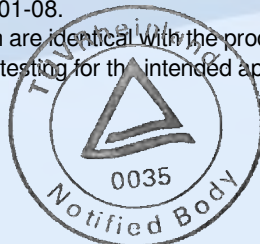
Berlin, 2015-01-08

Certification Body for Machinery, NB 0035

Dipl.-Ing. Eberhard Frejno

www.fs-products.com
www.tuv.com

TÜVRheinland®
Precisely Right.