EtherCAT 4-CH Encoder CMP/LTC Slave Module

207-C344F

User Manual

Version: V1.0 2026 Jan. 15

To properly use the product, read this manual thoroughly is necessary.

Part No.: 81-18C344F-010

Revision History

Date	Revision	Description
2023/11/08	1.0	Document creation.
2026/1/15	1.0a	Add Multi-Fixed Interval Trigger descriptions.

The product’s name and version number are both printed on the product itself. Released manual visions for each product design are represented by the digit before and after the period of the manual vision number.
Manual updates are represented by the third digit in the manual vision number.

Trademark

- **Windows 7/8/10**, **Visual Studio** are registered trademarks of Microsoft.
- **Beckhoff®**, **TwinCAT®**, **EtherCAT®**, **Safety over EtherCAT®**, **TwinSAFE®** are registered trademarks of and licensed by Beckhoff Automation GmbH.
- Other product names mentioned herein are used for identification purposes only and may be trademarks and/or registered trademarks of their respective companies.

Electrical safely

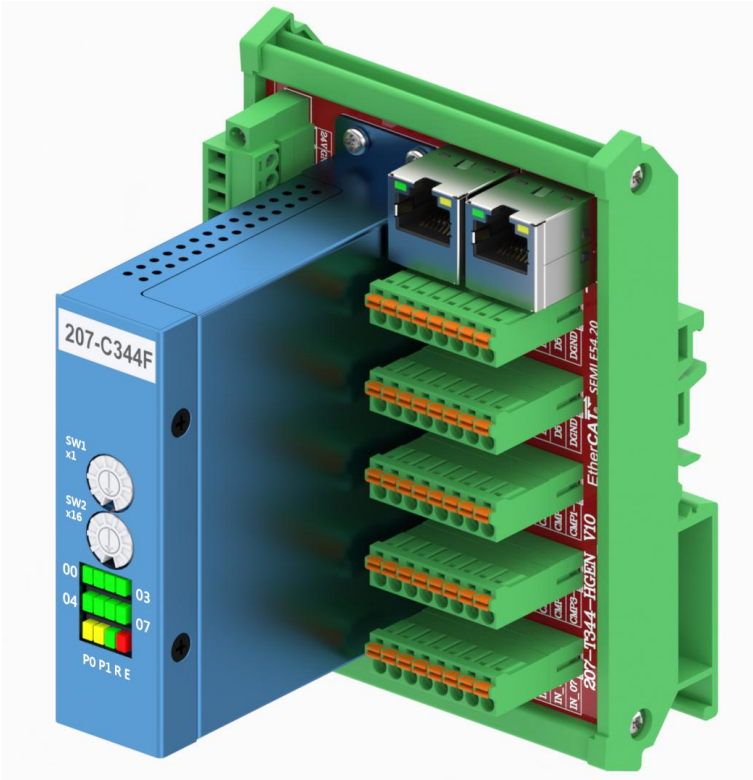
- To prevent electrical shock hazard, disconnect the power cable from the electrical outlet before relocating the system.
- When adding or removing devices to or from the system, ensure that the power cables for the devices are unplugged before the signal cables are connected. Disconnect all power cables from the existing system before you add a device.
- Before connecting or removing signal cables from motherboard, ensure that all power cables are unplugged.
- Seek professional assistance before using an adapter or extension card. These devices could interrupt the grounding circuit.
- Make sure that your power supply is set to the voltage available in your area.
- If the power supply is broken, contact a qualified service technician or your retailer.

Operational safely

- Please carefully read all the manuals that came with the package, before installing the new device.
- Before use the product, ensure all cables are correctly connected and the power cables are not damaged. If the power cables are detected damaged, contact the dealer immediately.
- To avoid short circuits, keep paper clips, screws, and staples away from connectors, slots, sockets and circuitry.
- Avoid dust, humidity, and temperature extremes. Do not place the product in any area where it may become wet.
- If you encounter technical problems with the product, contact a qualified service technician or the dealer.

Contents

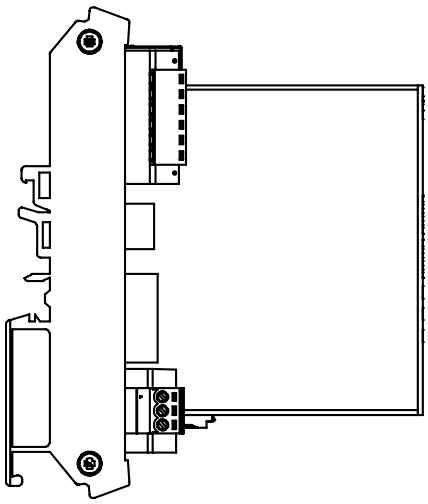
1. Product Overview

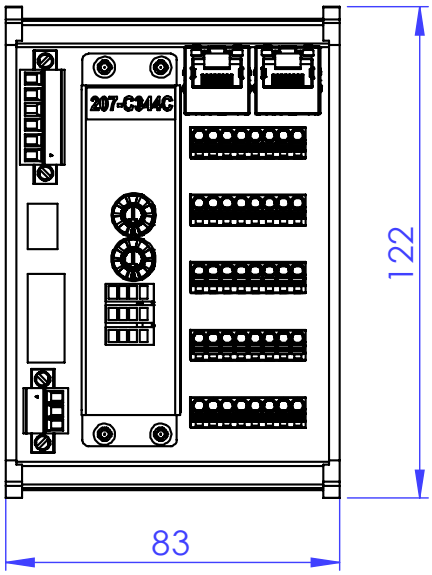


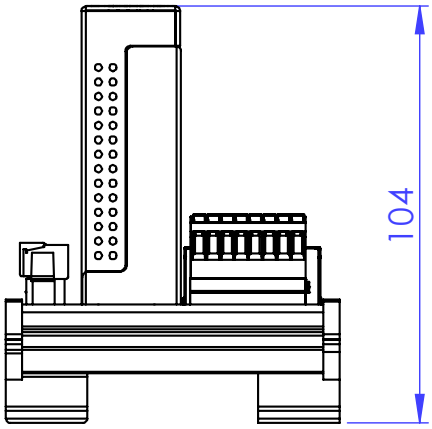
1.1 Naming Rule

2	0	7	-	C	3	4	4	F
EtherCAT Plug-in module				Encoder	EtherCAT series	4 ENC	4 I/O	CMP+LTC

1.2 Dimension







Unit: mm

1.3 Specification

EtherCAT	
Data transfer medium	Ethernet 100BASE-TX, CAT5 Shielded Ethernet cable
Communication type	DC
Distributed Clock	1ms
ID switch	8 bits

Encoder input	
Encoder Ch.	2-Ch. EA/EB/EZ independent 32-bit counter 2-Ch. EA/EB independent 32-bit counter
Encoder input frequency	Max. 4MHz, for CW/CCW and PULSE/DIR Max. 6.5MHz for 4xAB phase mode
Encoder mode	1xAB, 2xAB, 4xAB, CW/CCW, PULSE/DIR

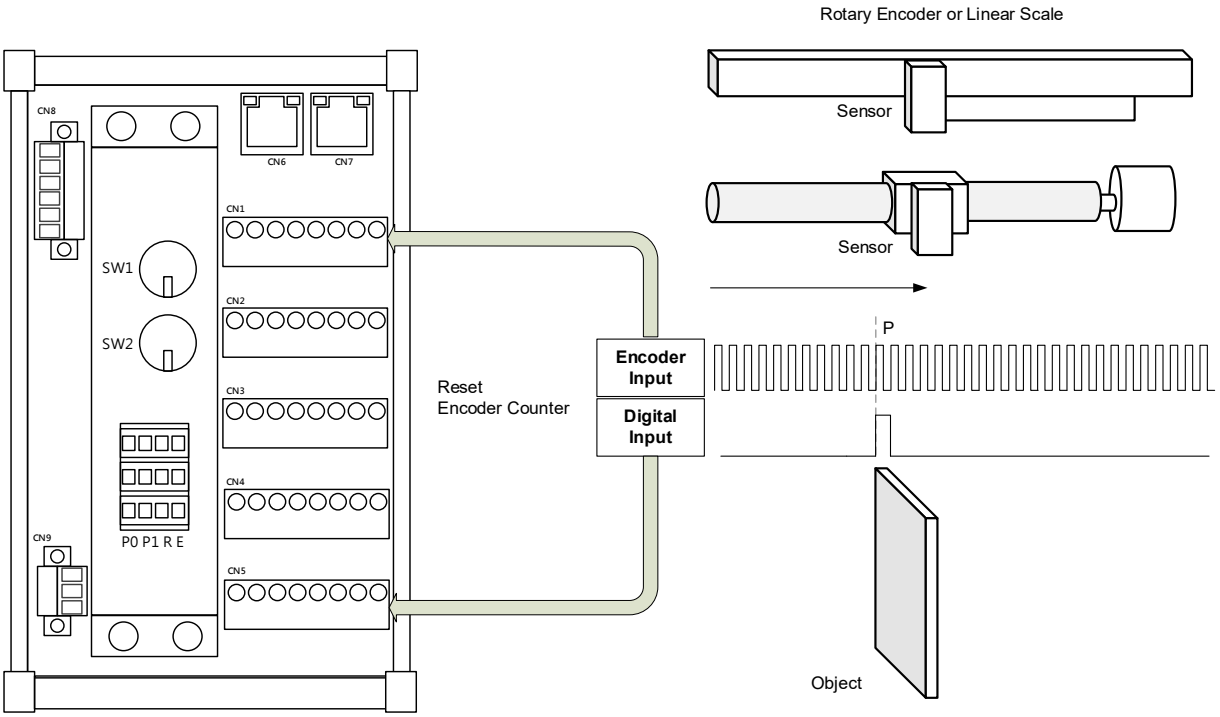
Digital I/O	
Isolated digital input	IN x 8, NPN type
	IN 0~3 as Latch function Max. 10KHz, 256pts for each channel
	IN 0~3 as Encoder reset function
	Digital filter 1~65535us

Compare Trigger Output	
Trigger Output	CMP x 4
	Auto Trigger, Max. trigger frequency 5MHz, 1 ~ unlimited counts
	Table Trigger, 4096pts allocatable for all channels
	Toggle Trigger, 4096pts allocatable for all channels
Trigger Pulse Width	Programmable, 0.1 ~ 6553.5 or 1 ~ 65535 micro-second

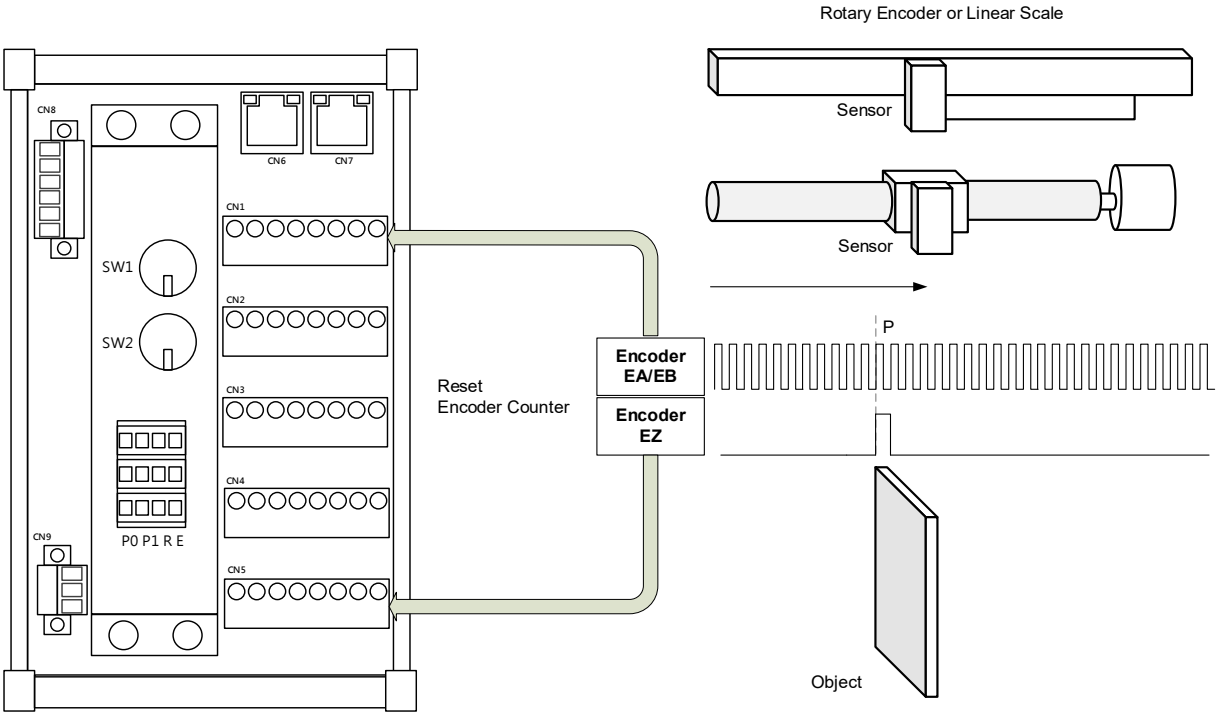
General	
Power input	24VDC±10%
Power consumption	3W typical
Working temperature	0°C~60°C (32°F~140°F) ambient temperature with air flow
Storage temperature	-20°C~80°C (-4°F~176°F)
Humidity	85% (non-condensing @60°C)
Size	L122 x W83 x H104 mm

1.4 Applications

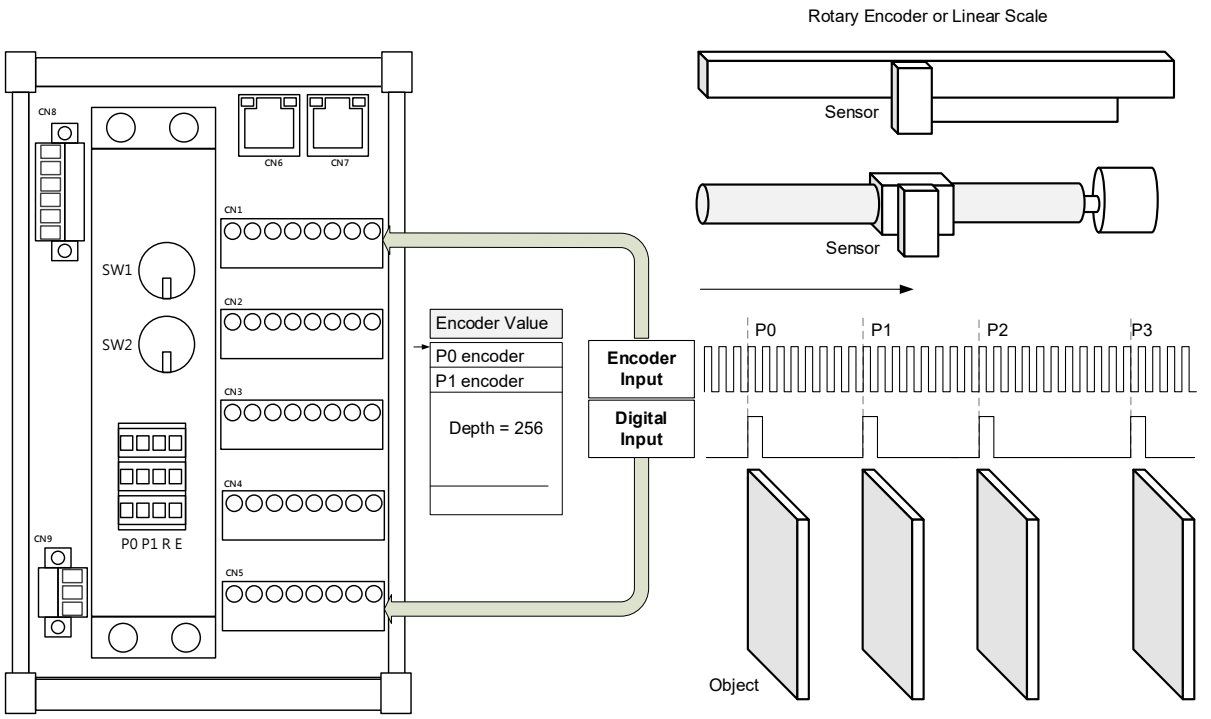
Encoder counter value reset by digital input



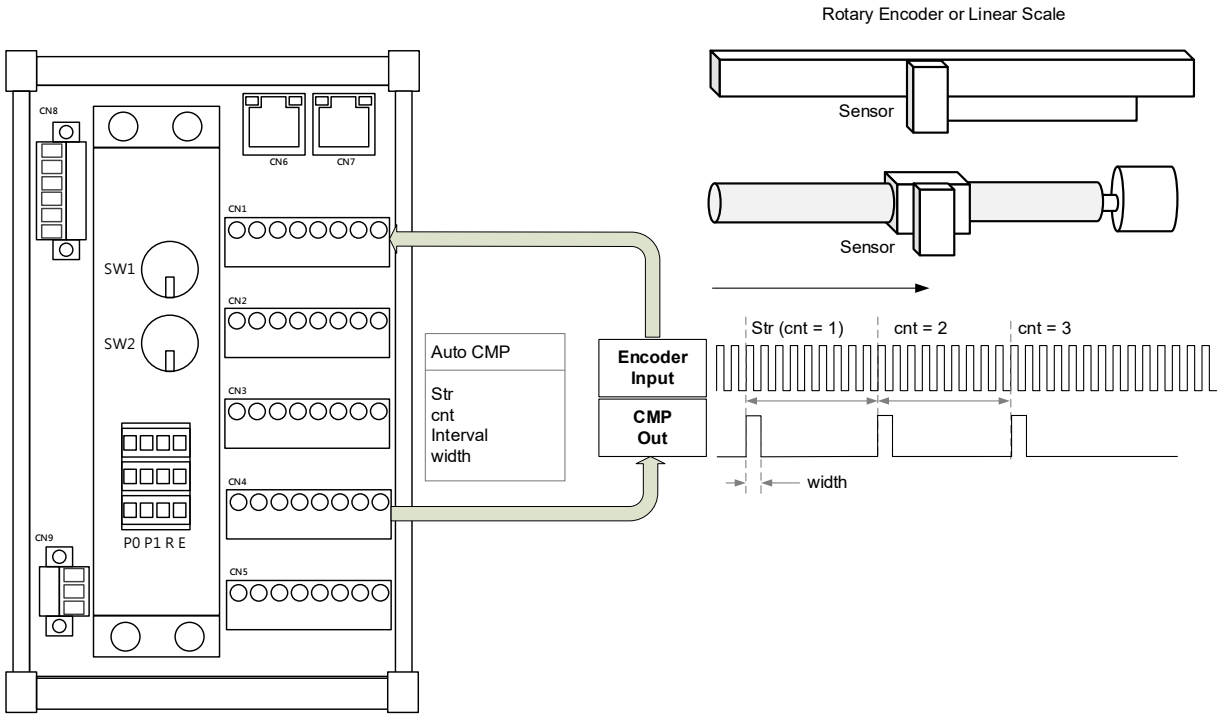
Encoder counter value reset by EZ



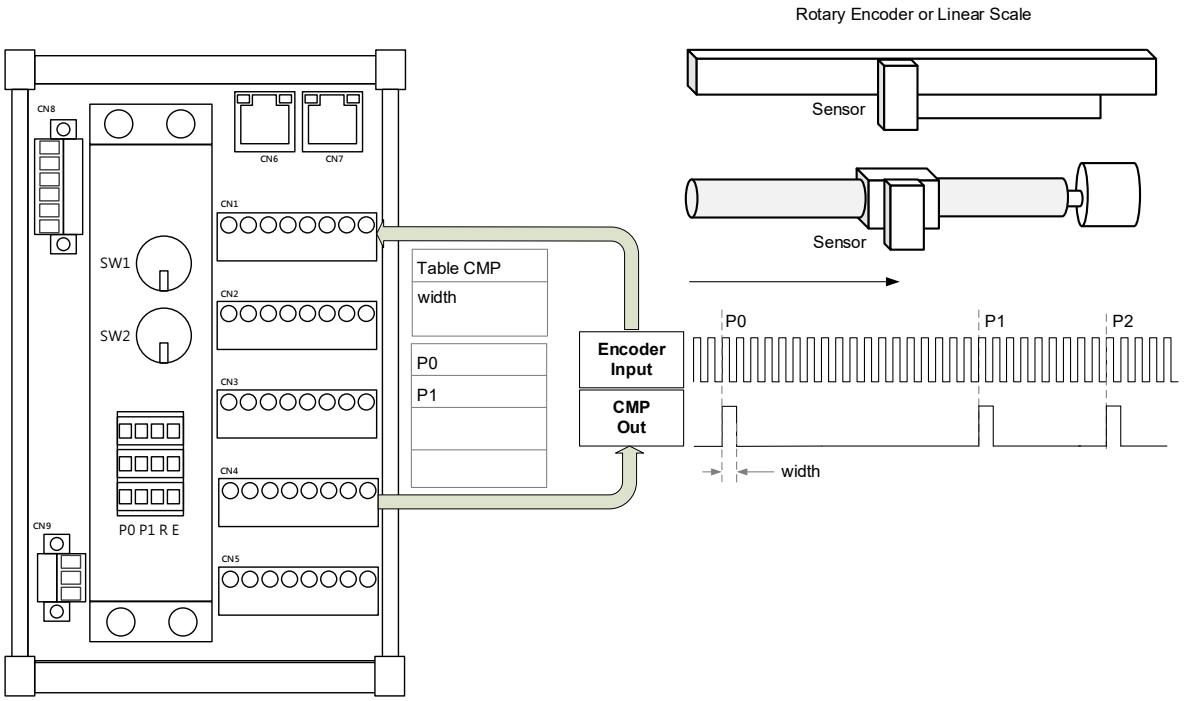
Encoder counter value latch by digital Input



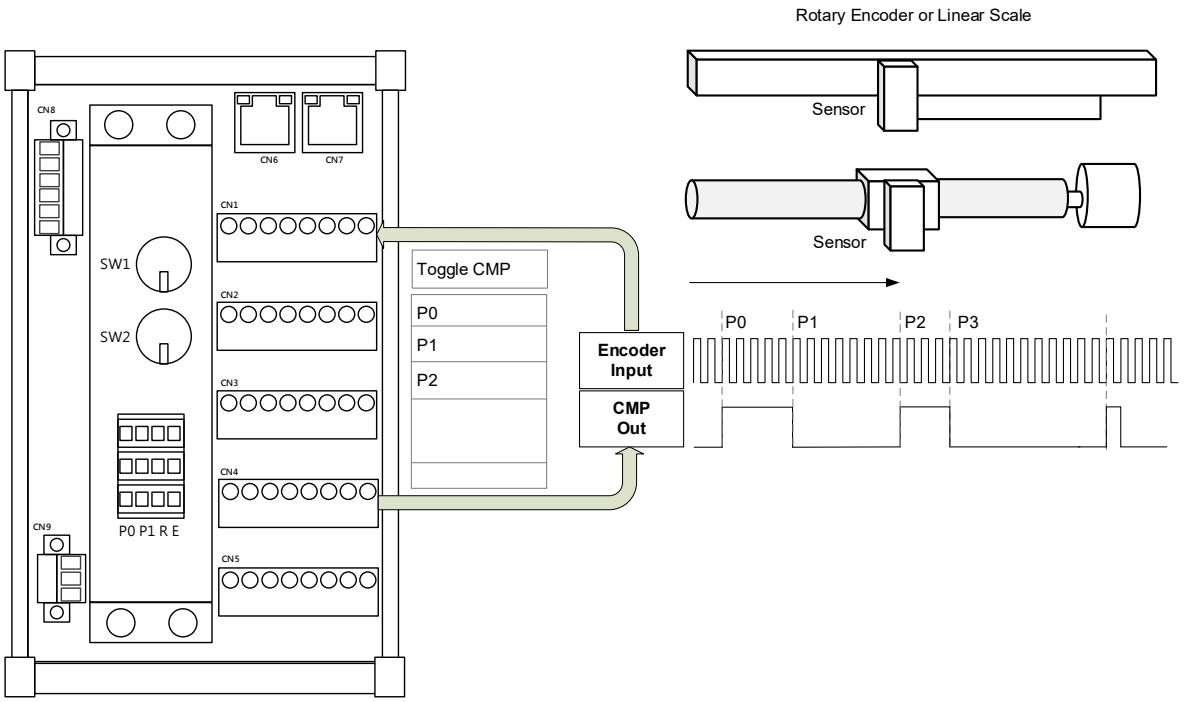
Compare trigger out follow encoder value – Auto CMP



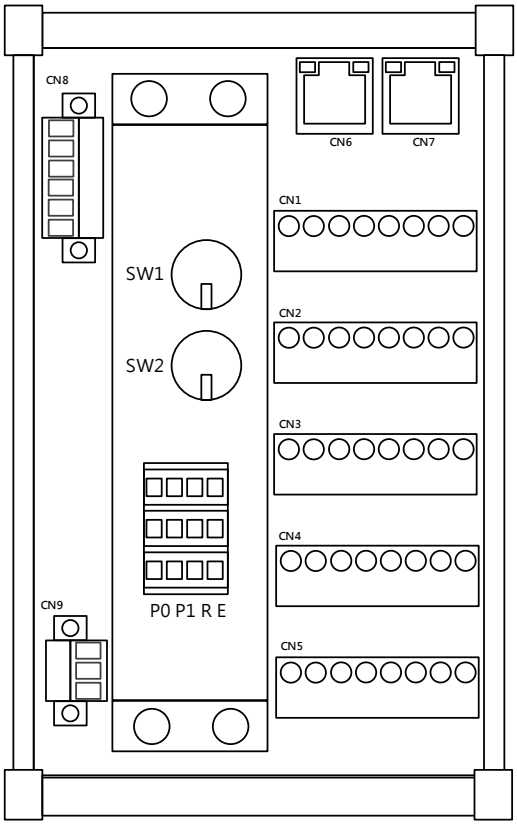
Compare trigger out follow encoder value – Table CMP



Compare trigger out follow encoder value – Toggle CMP

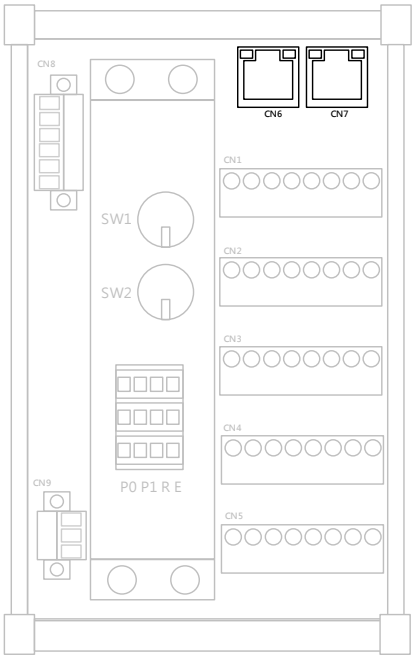


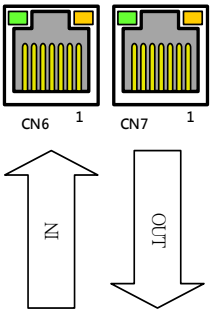
1.5 Placement



Label	Function
CN1	I/O Signal Connector
CN2	I/O Signal Connector
CN3	I/O Signal Connector
CN4	I/O Signal Connector
CN5	I/O Signal Connector
CN6	EtherCAT Communication IN
CN7	EtherCAT Communication OUT
CN8	Power Connector for External Isolation
CN9	Power Connector for Module

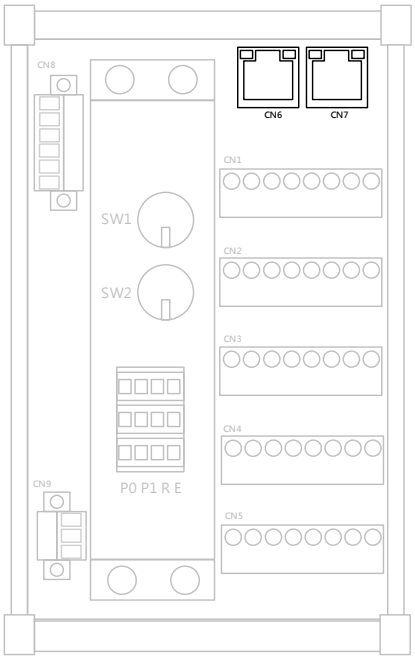
1.5.1 EtherCAT Communication IN and OUT



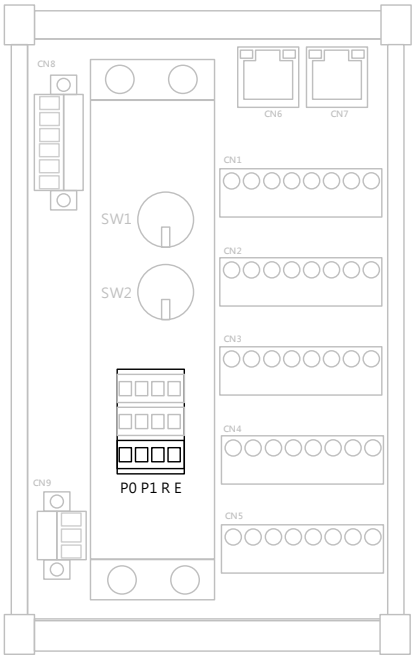


No.	Description
1	TX+
2	TX-
3	RX+
4	-
5	-
6	RX-
7	-
8	-

■ **Status LED**

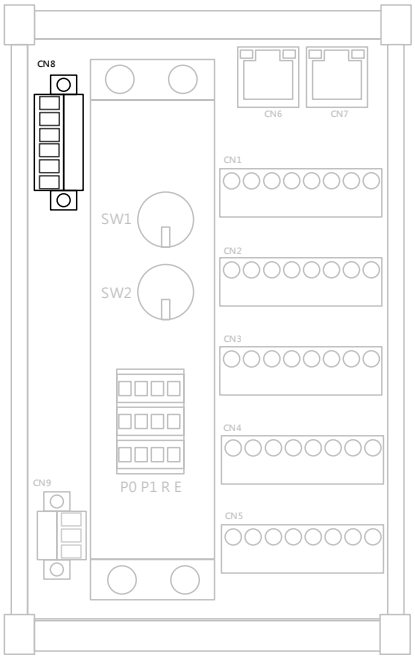


LED	Description
Left (Orange)	Link/Activity indicator: Blinking – There is activity on this port. Off – No link is established.
Right (Green)	Speed indicator: On – Operating as a 100/1000-Mbps connection. Off – Operating as a 10-Mbps connection.

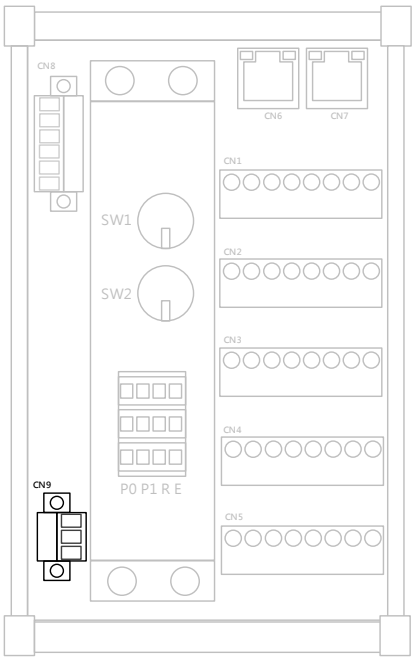


LED	Description
P0 - Yellow	DC +24V Supply
P1 - Yellow	DC +5V Supply for Internal
R - Green	In Normal Communication Off - INIT Status Slow Flash - PRE OP Status Single Flash - SAFE OP Status Last On - OP Status Quick Flash - BOOTSTRAP Status
E - Red	Error Communication Lights on when EtherCAT error occurs.

1.5.2 Power connector



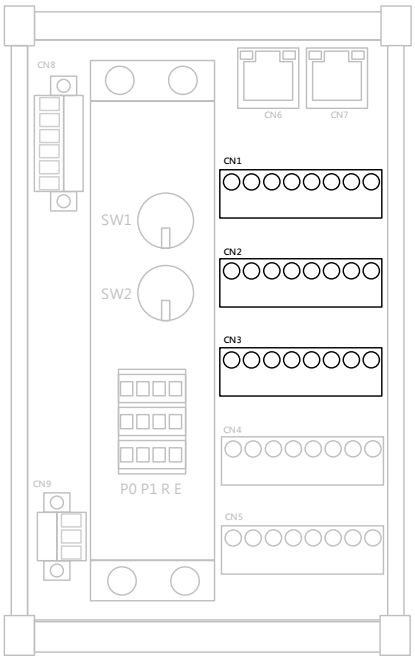
Pin	Label	Function
1	24V	DC 24V input
2	GND	DC 24V ground
3	FG	Field ground
5	24V	DC 24V input
6	GND	DC 24V ground
7	FG	Field ground
CN8 power input is used to supply power for CN5, high speed digital input.		



Pin	Label	Function
1	24Vs	DC 24V input for module internal
2	GNDs	DC 24V ground for module internal
3	FG	Field ground
CN9 power input is used to supply power for CN1~CN4, encoder input and compare trigger out.		

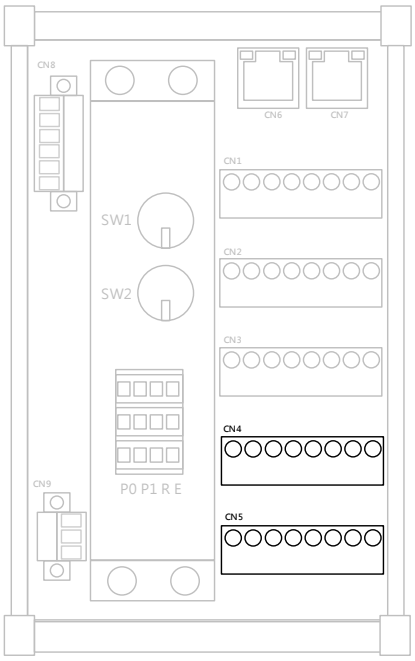
Frame Ground helps to shield internal circuits and equipment from external electromagnetic interference. Please note that grounding design should be based on the specific requirements of the device and the applicable electrical safety standards, ensuring compliance with relevant design and installation guidelines.

1.5.3 Encoder, Compare Trigger and Digital Input



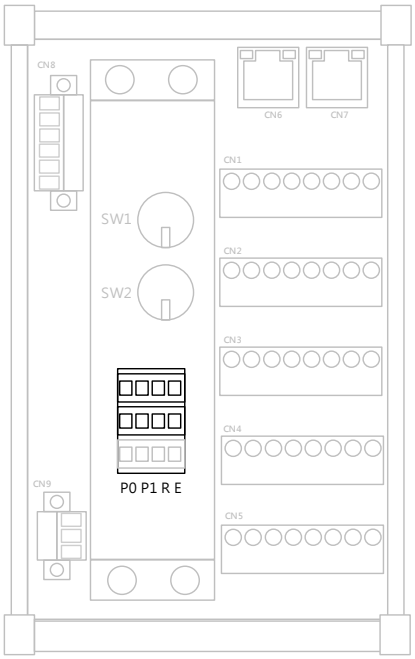
	Pin	Label	Function
CN1	1	EA0+	ENC_0 A phase (+)
	2	EA0-	ENC_0 A phase (-)
	3	EB0+	ENC_0 B phase (+)
	4	EB0-	ENC_0 B phase (-)
	5	EZ0+	ENC_0 Z phase (+)
	6	EZ0-	ENC_0 Z phase (-)
	7	D5V	DC 5V Output
	8	DGND	DC 5V Ground
CN2	1	EA1+	ENC_1 A phase (+)
	2	EA1-	ENC_1 A phase (-)
	3	EB1+	ENC_1 B phase (+)
	4	EB1-	ENC_1 B phase (-)
	5	EZ1+	ENC_1 Z phase (+)
	6	EZ1-	ENC_1 Z phase (-)
	7	D5V	DC 5V Output
	8	DGND	DC 5V Ground
CN3	1	EA2+	ENC_2 A phase (+)
	2	EA2-	ENC_2 A phase (-)
	3	EB2+	ENC_2 B phase (+)
	4	EB2-	ENC_2 B phase (-)
	5	CMP0+	CMP_0 Trigger out (+)
	6	CMP0-	CMP_0 Trigger out (-)
	7	CMP1+	CMP_1 Trigger out (+)
	8	CMP1-	CMP_1 Trigger out (-)

Note Note DC 5V output capacity is 150mA.



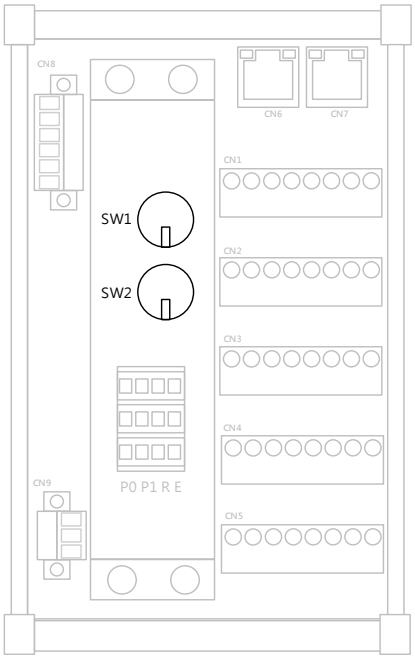
	Pin	Label	Function
CN4	1	EA3+	ENC_3 A phase (+)
	2	EA3-	ENC_3 A phase (-)
	3	EB3+	ENC_3 B phase (+)
	4	EB3-	ENC_3 B phase (-)
	5	CMP2+	CMP_2 Trigger out (+)
	6	CMP2-	CMP_2 Trigger out (-)
	7	CMP3+	CMP_3 Trigger out (+)
	8	CMP3-	CMP_3 Trigger out (-)
CN5	1	IN_00	Port#0 Bit0 Input
	2	IN_01	Port#0 Bit1 Input
	3	IN_02	Port#0 Bit2 Input
	4	IN_03	Port#0 Bit3 Input
	5	IN_04	Port#0 Bit4 Input
	6	IN_05	Port#0 Bit5 Input
	7	IN_06	Port#0 Bit6 Input
	8	IN_07	Port#0 Bit7 Input

■ Status LED



LED	Description
00 ~ 03 – Green	Digital Input bit 0 ~ 3 Lights on when digital input on.
04 ~ 07 - Green	Digital Input bit 4 ~ 7 Lights on when digital input on.

1.5.4 Station Address



Label	Description	Value
SW1	node number_L	0 ~ 15
SW2	node number_H	0 ~ 15

Note that node number = 16*SW2 + 1*SW1.

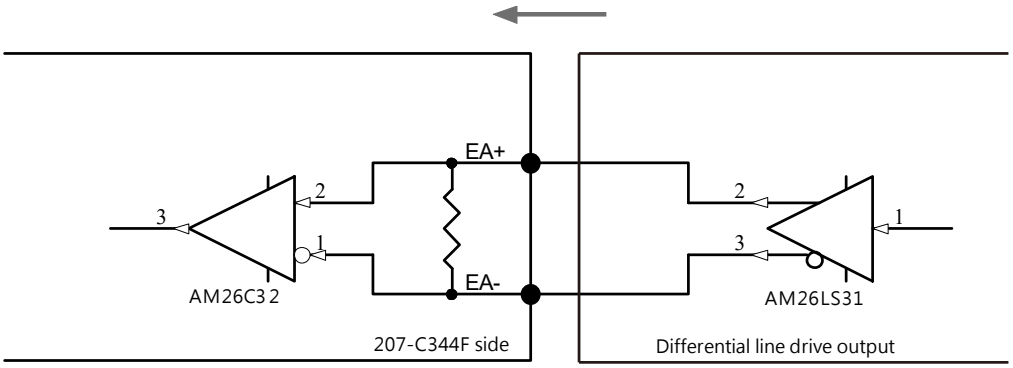
E.g. SW1 = 10, SW2 =2.

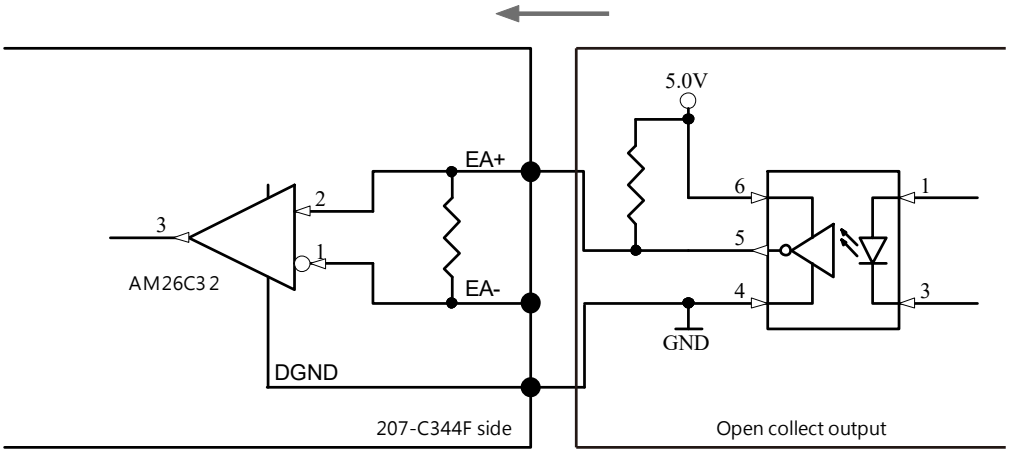
The node number will be set as “2 x 16 + 10 x 1= 42”.

Default values are all 0.

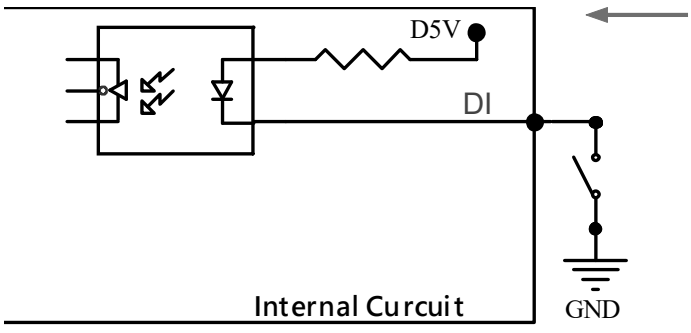
1.6 Signal Circuit

- Encoder Input Circuit



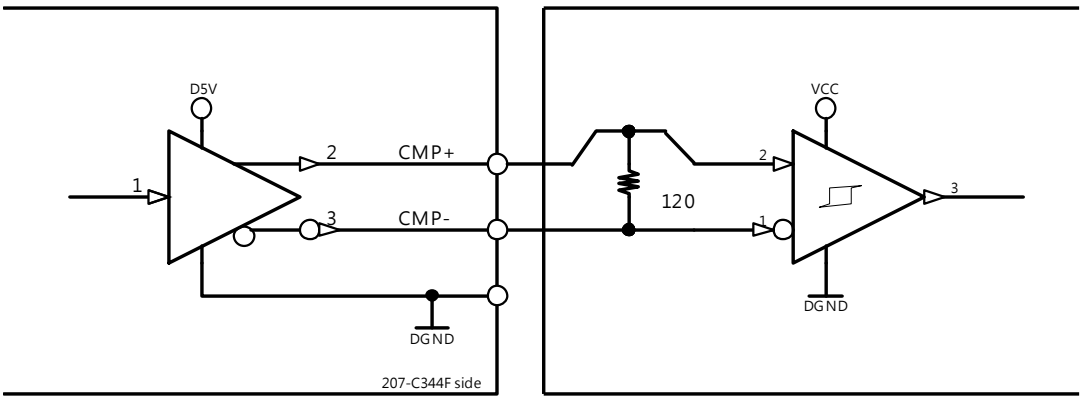


- Digital Input Signal Circuit

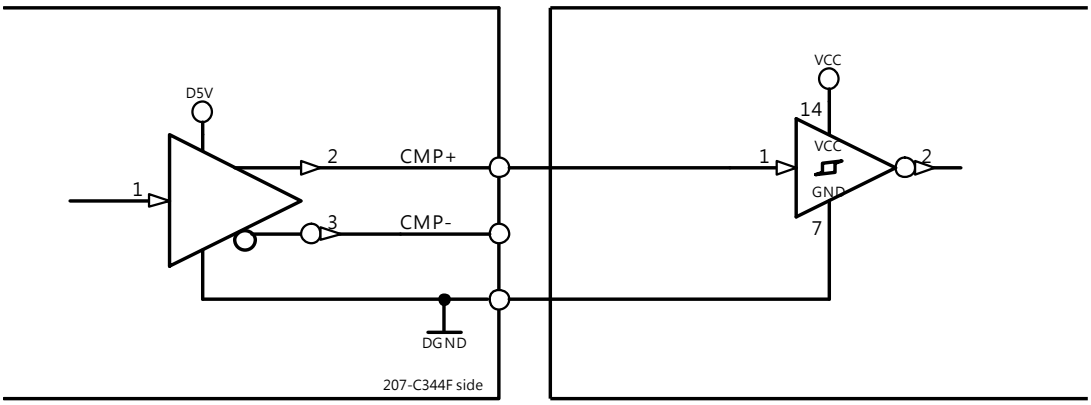


■ Compare Trigger Output Signal Circuit

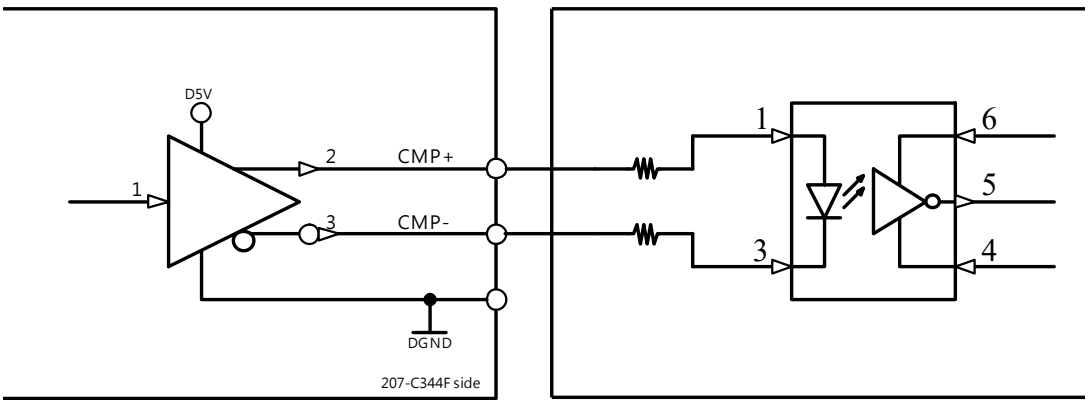
Differential output (Line Driver to Receiver)



Differential output (Line Driver to TTL)



Differential output (Line Driver to Photocoupler)



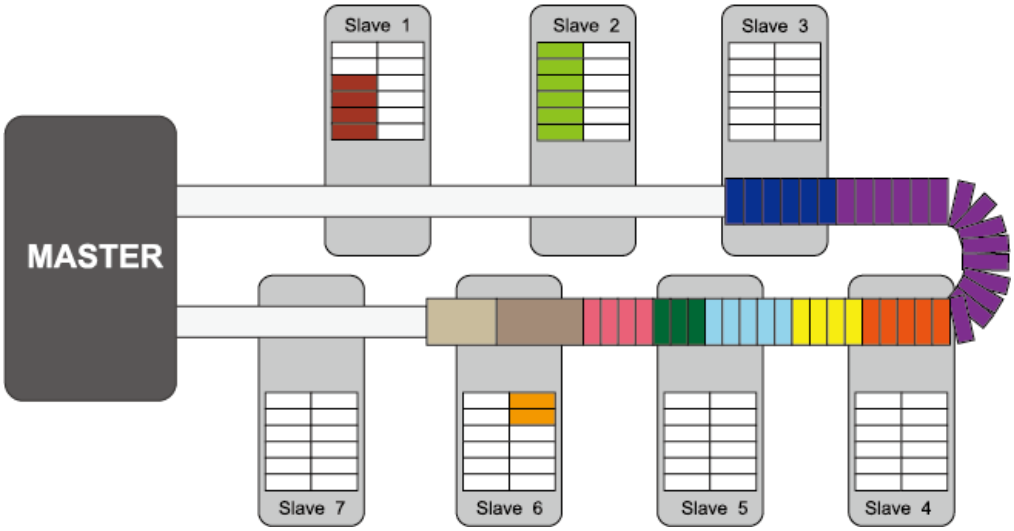
2. EtherCAT Introduction

2.1 Introduction

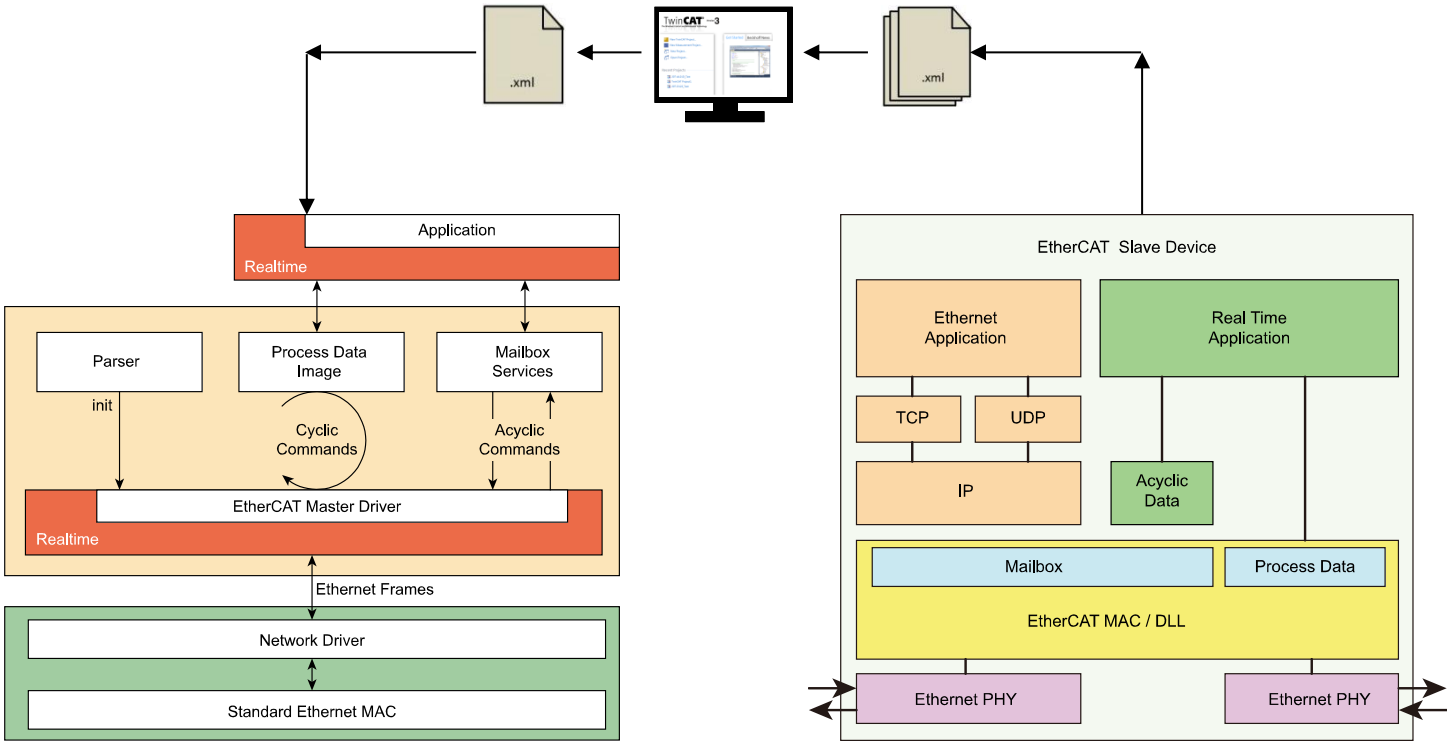
EtherCAT® is an ultra-high-speed serial communication system. This technology is widely applied in factory and machinery automation industries. EtherCAT® is real-time down to the I/O level. The transmission rate of EtherCAT® is 2 x 100 Mbit/s, which makes it the fastest Ethernet. Each EtherCAT® slave device reads and writes the data by the function of "on the fly". One can extract or insert bits or bytes without suspending the system. Each EtherCAT® segment can connect up to 65,535 nodes. With 100BASE-TX, the distance between two nodes is up to 100M with EtherCAT®. With 100BASE-FX (fiber optics), the distance between two nodes is longer than 100M.

Precise synchronization is one of the features of EtherCAT®. The Distributed Clocks (DC) can adjust the time of Master and Slaves to achieve the synchronization. The time of synchronization is less than 1μs. EtherCAT® also leads to lower solution costs because of the low cost slave controller with FPGA, small volume with EtherCAT® instead of IPC, and so on. EtherCAT® is IEC, ISO, and SEMI standard protocol. The slave controller can provide interoperability. The master stacks are suitable for various Real-time Operating System (RTOS).

2.2 System Configurations



2.3 Data Transition



2.4 EtherCAT Tool: TwinCAT

TwinCAT® is the EtherCAT® tool which is developed by Beckhoff. The TwinCAT® (The Windows Control and Automation Technology) automation suite forms the core of the control system. The TwinCAT® software system turns almost any PC-based system into a real-time control with multiple PLC, NC, CNC and/or robotics runtime systems.

All TPM modules can be tested with TwinCAT® easily. With the RJ45 cable, EtherCAT® Master and EtherCAT® slaves can connect to achieve the control system. EZE-xxx model names will be displayed on TwinCAT® for users to operate system conveniently. Carrier specific model name will not be listed.

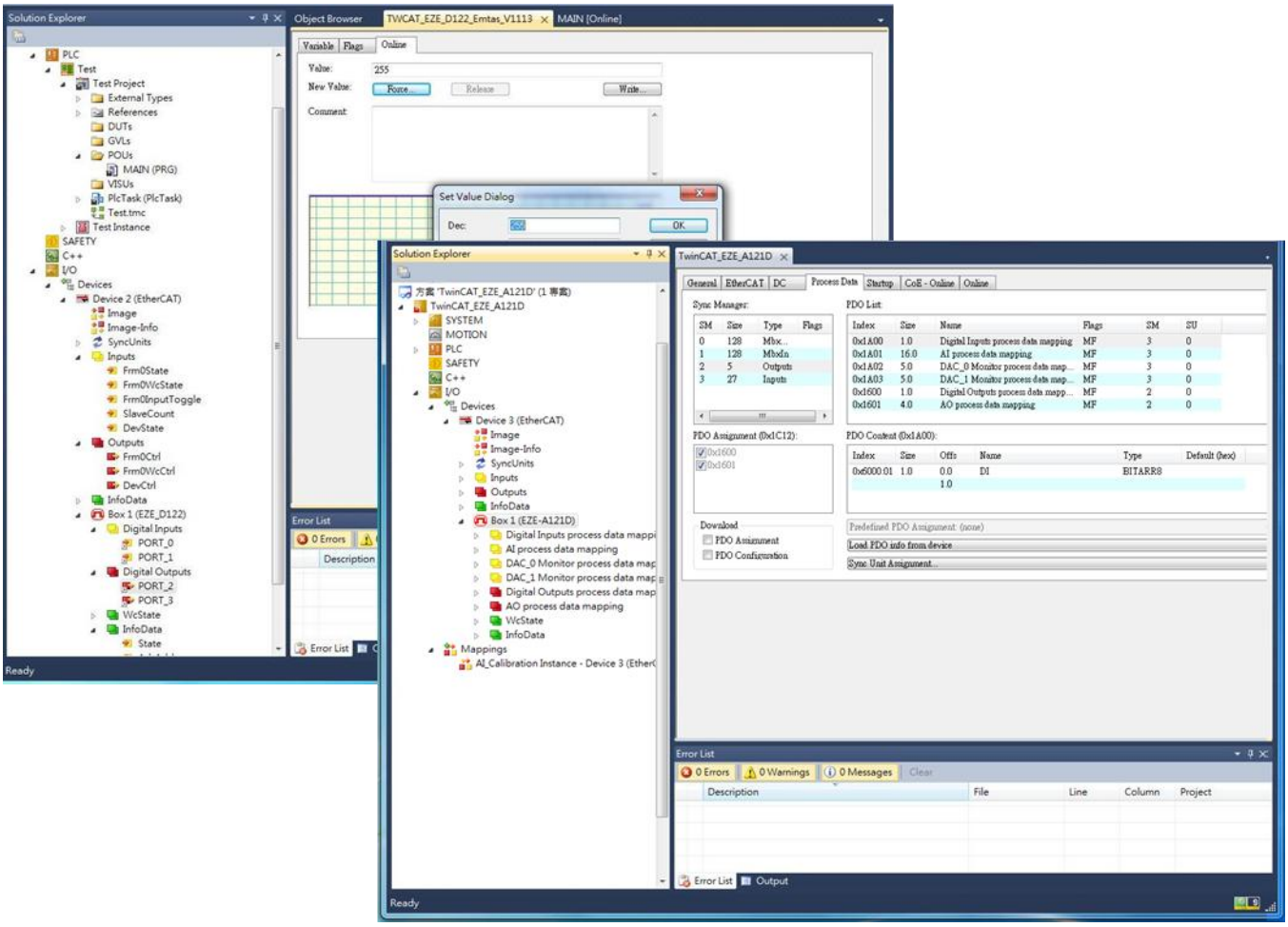


Figure 2-1: illustration of TwinCAT operation

3. EtherCAT Communication

The index ranges depend on the used EtherCAT profile. The basic index ranges used in the device are listed in following table.

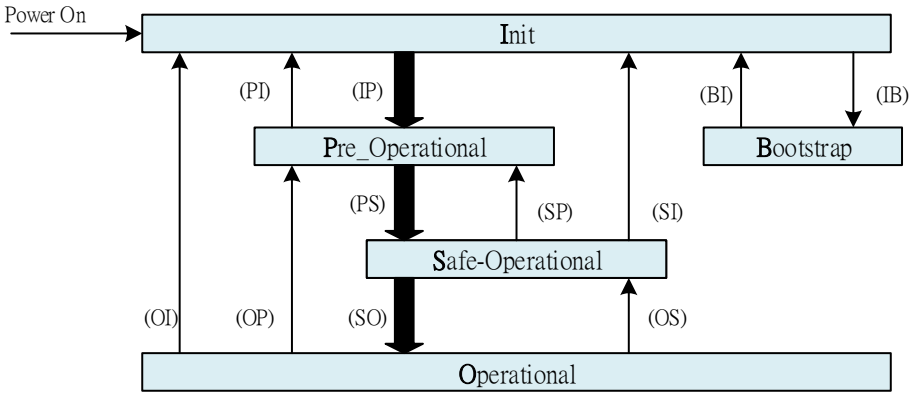
Index Range	Description
0x0000-0x0FFF	Data Type Area
0x1000-0x1FFF	Communication Area
0x1600-0x19FF	RxPDO Mapping
0x1A00-0x1BFF	TxPDO Mapping
0x1C10-0x1C2F	Sync Manager PDO Assignment
0x1C30-0x1C4F	Sync Manager Parameters
0x2000-0x5FFF	Manufacturer Specific Area
0x6000-0x6FFF	Input Area
0x7000-0x7FFF	Output Area
0x8000-0x8FFF	Configuration Area
0x9000-0x9FFF	Information Area
0xA000-0xAFFF	Diagnosis Area
0xB000-0xBFFF	Service Transfer Area
0xC000-0xEFFF	Reserved Area
0xF000h-0xFFFF	Device Area

3.1 Communication data

PDO (Process Data Object) and SDO (Service Data Object) are two different communication mechanisms used for data exchange between the EtherCAT master and slave devices.

3.2 ESM (EtherCAT State Machine)

The EtherCAT State machine (ESM) is used to manage the communications states between the master and slave applications when EtherCAT communications are started and during operation, as show in the following figure.
Normally, the requests of state changes are from the master. The master requests the change by writing the ESM with the request to be changed in the AL control register of the slaves. The slave confirms the result of the state change as either successful or failed and then responds to the master with the local AL status. If the requested state change fails, the slave responds with an error flag.



■ ESM contains states

Symbol	Name	Communication Operation	Description
INIT	Init	The communication part is initializing and the transmission and reception with both SDO (Mailbox) and PDO are impossible	INIT state defines basic communication relation between the master and slave in the application layer. Direct communication between the master and slaves is not possible in the application layer. The master uses the INIT state to initialize the setting for the configuration of the slaves. When the slaves support the mailbox service, the corresponding SM settings will also be executed in INIT state.
PREOP	Pre-Operational	Possible to send and receive data through SDO (Mailbox)	The mailbox communication can be performed in the PREOP state when the slaves support the optional mailbox. Both master and slaves can use the mailbox to initialize application specifications and to change parameters. Process data communication cannot be executed in this state.

SAFEOP	Safe-Operational	The transmission (from slave to master) with PDO as well as the transmission and reception over SDO (Mailbox) are possible.	In SAFEOP state, Slave applications transfer the actual input data, but not the output data that may not be available for processing. The output must be set in this state.
OP	Operational	Possible to send and receive both SDO (Mailbox) and PDO.	In OP state, slave applications transfer the actual input data and the master application transfers the actual output data.
BOOT	Bootstrap	Impossible to send and receive both SDO and PDO, in this state.	In BOOT state, slave applications can receive new firmware downloaded to the FoE (File access Over EtherCAT).

■ State transition and local Management Service

Transition Symbol	Direction	Local Management Service
IP	INIT => PREOP	Start Mailbox Communication
PI	PREOP => INIT	Stop Mailbox Communication
PS	PREOP => SAVEOP	Start Input Update
SP	SAVEOP => PREOP	Stop Input Update
SO	SAVEOP => OP	Start Output Update
OS	OP => SAVEOP	Stop Output Update
OP	OP => PREOP	Stop Input Update, Stop Output Update
SI	SAVEOP => INIT	Stop Input Update, Stop Mailbox Communication
OI	OP => INIT	Stop Input Update, Stop Output Update, Stop Mailbox Communication
IB	INIT => BOOT	Start Firmware Update(FoE), Start Bootstrap Mode
BI	BOO => INIT	Start Firmware Update(FoE), Restart Device

3.3 PDO Properties

Process Data Object (PDO):
PDOs are used for cyclic, real-time data exchange between the master and the slave devices. They allow for fast and deterministic exchange of input and output data.

This device supports the PDO (Process Data Object).

The CANOpen over EtherCAT protocol allows the user to map objects to process data objects (PDOs) in order to use the PDOs for real-time data transfer. The PDO mappings define which objects will be included in the PDOs. PDO is composed of RxPDO transferring from master to slave and TxPDO transferring from slave to master.

PDO Type	Sender	Receiver
RxPDO	Master	Slave
TxPDO	Slave	Maser

3.4 SDO Properties

Service Data Object (SDO):
SDOs are used for configuration, parameterization, and remote access to the internal data of the EtherCAT slave devices. SDOs enable read and write access to the slave's object dictionary.

The data exchange of SDO uses the Mailbox communication. Therefore, be aware that update timing of SDO will be indefinite. The object setting and various sate monitoring of the slave are enabled by reading/writing data from/into the entry of the object dictionary in the master.
So, you need to keep in mind these:

1. It may take some time to read and write operations SDO response.
2. Objects that are updated in the PDO do not update the SDO. They are always overwritten the values of PDO.

3.5 PDO and SDO Usage

Most functions have 2 kinds of objects, PDO and SDO. PDO is used to read value, read status, or write status and SDO is used to configure parameters, such as encoder mode or CMP positions array.

Function	PDO	SDO
Encoder Current Value	Read Value	Configuration
Encoder Preset Value	Read Status or Write	Configuration
Encoder Auto Rest	Read Status or Write	Configuration
Latch (Last one)	Read Status or Write	Configuration
FIFO Latch	Read Status or Write	Read or Configuration
CMP	Read Status or Write	Configuration

3.5.1 Object Lists

PDOs:

Index (hex)	Name
6000:0	DI status
60n3:0	Encoder N inputs
60n4:0	CMP N inputs
60n5:0	Latch N inputs
70n3:0	Encoder N outputs
70n4:0	CMP N outputs
70n5:0	Latch N outputs

SDOs:

Index (hex)	Name
1000:0	Device type
1008:0	Device name
1009:0	Hardware version
100A:0	Software version
1010:0	Store parameters
1011:0	Restore default parameters
1018:0	Identity
20n0:0	Latch N position table index 0 - 199
20n1:0	Latch N position table index 200 - 255
20nF:0	Latch N info

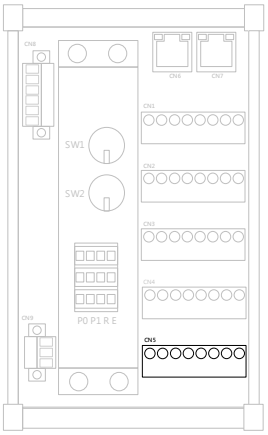
21n0:0	Latch N signal table index 0 - 199
21n1:0	Latch N signal table index 200 - 255
3000:0	CMP table index 0 - 199
3001:0	CMP table index 200 - 399
3002:0	CMP table index 400 - 599
3003:0	CMP table index 600 - 799
3004:0	CMP table index 800 - 999
3005:0	CMP table index 1000 - 1199
3006:0	CMP table index 1200 - 1399
3007:0	CMP table index 1400 - 1599
3008:0	CMP table index 1600 - 1799
3009:0	CMP table index 1800 - 1999
300A:0	CMP table index 2000 - 2199
300B:0	CMP table index 2200 - 2399
300C:0	CMP table index 2400 - 2599
300D:0	CMP table index 2600 - 2799
300E:0	CMP table index 2800 - 2999
300F:0	CMP table index 3000 - 3199
3010:0	CMP table index 3200 - 3399
3011:0	CMP table index 3400 - 3599
3012:0	CMP table index 3600 - 3799
3013:0	CMP table index 3800 - 3999
3014:0	CMP table index 4000 - 4095
3FFF:0	CMP table info
4000:0	Station alias
4007:0	FPGA version
8000:0	DI filter
80n1:0	DI N settings
80n3:0	Encoder N settings
80n4:0	CMP N settings
80n5:0	Latch N settings

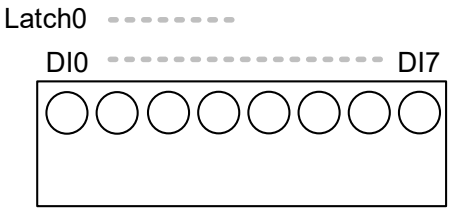
4. Function Description

4.1 DI Function Objects

CN5 provides 8 digital input channels labeled DI0 through DI7.

The key feature is that DI0 ~ DI3 replaced by Latch0 ~ Latch3 could be used as latch function.





4.1.1 Index 6000: DI Status

Index (hex)	Name	Description	Data Type	Flags	Default	Store
6000:0	DI status	Digital input status	UINT8	RO	0	No

Bit	Signal	Bit Off	Bit On
0	DI 0	Signal OFF	Signal ON
1	DI 1		
2	DI 2		
3	DI 3		
4	DI 4		
5	DI 5		
6	DI 6		
7	DI 7		

4.1.2 Index 8000: DI Filter

Index (hex)	Name	Description	Data Type	Flags	Default	Store
8000:0	DI filter	DI0 ~ 3 input signal filter for "Auto Reset Counter" and "Latch" Value: 0 — Disable Value: other — Signal shorter than value will be ignored (unit microsecond)	UINT16	RW	0x0000	Yes

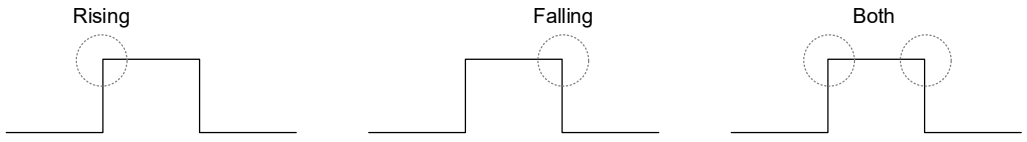
Note DI filter is available for DI 0~3 by one configuration.
Set the time-frequency filter to reduce noise.
Filter range is from 1 ~ 65535 microsecond.

4.1.3 Index 80n1: DI N Settings (N = 0 ~ 3)

Index (hex)	Name	Description	Data Type	Flags	Default	Store
80n1:0	DI N settings	DI active timing configuration	UINT8	RO	1	No
80n1:01	Active Edge	DI N input signal active edge for "Auto Reset Counter" and "Latch" Value: 0 — Rising edge Value: 1 — Falling edge Value: 2 — Both edge	UINT8	RW	0x00	Yes

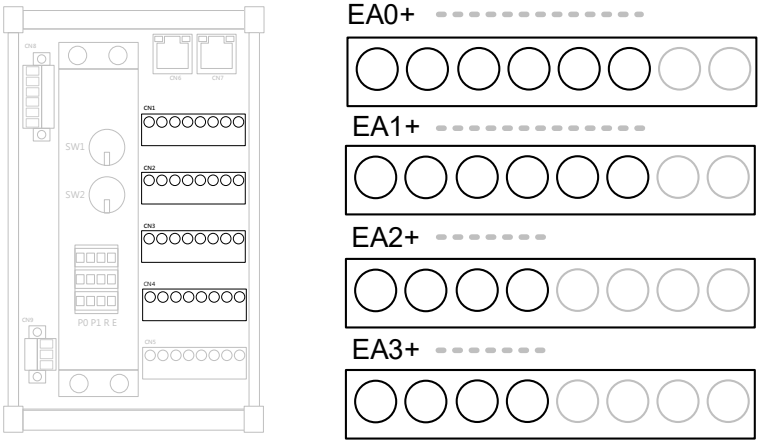
Note DI N means input channels, N and n = 0, 1, 2, 3.

Auto Reset Counter and Latch Active timing:



4.2 Encoder Input Objects

CN1~CN4 provide 4 encoder input channels labeled EAn+, EAn-, EBn+ and EBn-, n = 0, 1, 2, 3. Especially, first two channels with EZ signal.



4.2.1 Index 80n3: Encoder N Settings (N = 0 ~ 3)

Index (hex)	Name	Description	Data Type	Flags	Default value	Store
80n3:0	Encoder N settings	Encoder input configuration	UINT8	RO	4	No
80n3:01	Mode	Set encoder input mode Value: 0 = 4xA/B; 1 = CW/CCW; 2 = P/D; 3 = 1xA/B; 4 = 2xA/B; 5 = P/D reverse	UINT8	RW	0x00	Yes
80n3:02	Ring Counter	The maximum value of ring counter Set 0x00000000 or 0xFFFFFFFF to disable	UINT32	RW	0x00000000	Yes
80n3:03	Select Auto Reset Counter Signal	Set auto reset signal source Bit: 0 = DI 0; 1 = DI 1; 2 = DI 2; 3 = DI 3; 4 ~ 7 = Reserved; 8 = EZ N	UINT16	RW	0x0001<<N	Yes

<table>
<tr><td></td><td></td><td><table><tr><td>9 ~ 15</td><td>Reserved</td></tr></table></td><td></td><td></td><td></td><td></td></tr>
<tr><td>80n3:04</td><td>Auto Reset Counter Mode</td><td>Set auto reset counter mode<table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Once (auto disable)</td></tr><tr><td>1</td><td>Infinite</td></tr></table></td><td>UINT8</td><td>RW</td><td>0x00</td><td>Yes</td></tr>
</table>

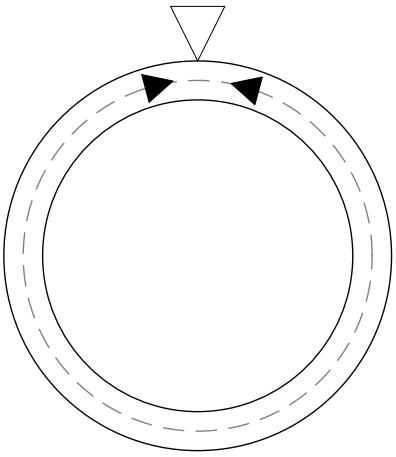
Note Encoder N means encoder channels, N and n = 0, 1, 2, 3.

Note Set the maximum ring counter value so that the maximum value would be in the range from 1 to 4,294,967,295 (FFFFFFFF hex).

If the encoder value exceeds the maximum value, it will return to 0 to continue counting.

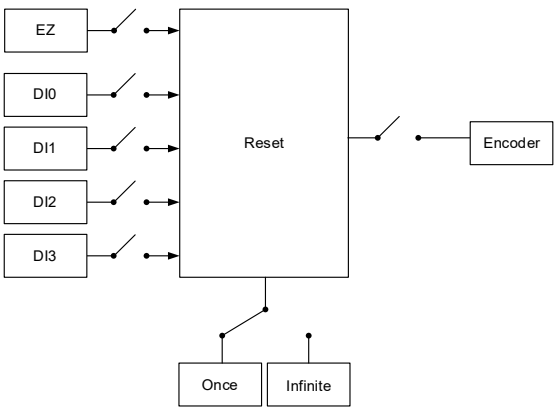
If the encoder value becomes less than 0, it will return to the maximum value to continue counting.

1 ≤ maximum value ≤ 4,294,967,295



Note Auto reset counter function could be set active for multiple signals. i.g. EZ and DI2.

The auto reset counter function has two behaviors: once mode and infinite mode. In once mode, the counter is reset only once, even if the active signal is triggered repeatedly. In infinite mode, the counter is reset repeatedly each time the active signal is triggered.

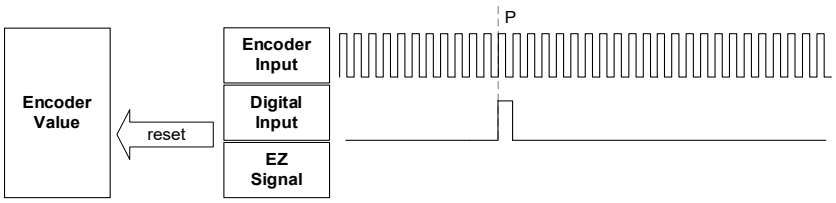


4.2.2 Index 60n3: Encoder N Inputs

<table>
<tr><th>Index (hex)</th><th>Name</th><th>Description</th><th>Data Type</th><th>Flags</th><th>Default</th><th>Store</th></tr>
<tr><td>60n3:0</td><td>Encoder N inputs</td><td>Display encoder value and status</td><td>UINT8</td><td>RO</td><td>4</td><td>No</td></tr>
<tr><td>60n3:01</td><td>Set Counter Done</td><td>Display if set encoder value is completed.<table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Nothing</td></tr><tr><td>1</td><td>The encoder value was set</td></tr></table></td><td>UINT8</td><td>RO</td><td>0</td><td>No</td></tr>
<tr><td>60n3:02</td><td>Counter Value</td><td>Current encoder counter value</td><td>INT32</td><td>RO</td><td>0</td><td>No</td></tr>
<tr><td>60n3:03</td><td>Auto Reset Counter Status</td><td>Display auto reset counter status.<table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Disable</td></tr><tr><td>1</td><td>Enable</td></tr></table></td><td>UINT8</td><td>RO</td><td>0</td><td>No</td></tr>
<tr><td>60n3:04</td><td>Last Auto Reset Counter Signal</td><td>Display which signal is the latest source of auto reset counter signal.<table><tr><th>Bit</th><th>Description</th></tr><tr><td>0</td><td>DI 0</td></tr><tr><td>1</td><td>DI 1</td></tr><tr><td>2</td><td>DI 2</td></tr><tr><td>3</td><td>DI 3</td></tr><tr><td>4 ~ 7</td><td>Reserved</td></tr><tr><td>8</td><td>EZ N</td></tr><tr><td>9 ~ 15</td><td>Reserved</td></tr></table></td><td>UINT16</td><td>RO</td><td>0</td><td>No</td></tr>
</table>

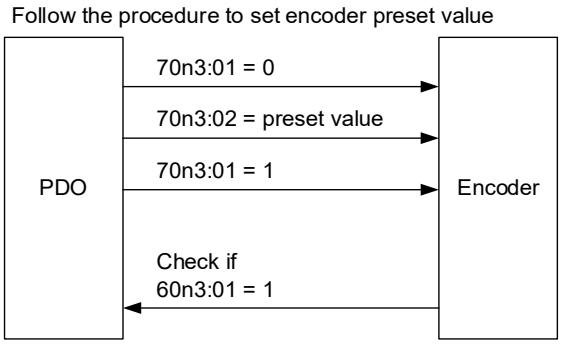
Note Encoder N means encoder channels, N and n = 0, 1, 2, 3.
EZ N means EZ signals, N = 0, 1, because only encoder 0 and 1 have EZ.
The encoder value of each channel could be reset by digital input and EZ signal.
Any one of digital input can reset the encoder value of any one channel.
EZ signal can reset the encoder value of its own channel.

Note "Auto Reset Counter" is a function to reset encoder value by external signal automatically.
The external signal source could be DI or EZ signal.
Once the encoder value is reset and the signal source is also record in Last Auto Reset Counter Signal.



Note "Set Counter Done" is used to check if set encoder preset value is completed.

Once "Set Counter Done" is turned to 1 and it will remain 1 until Set Counter (70n3:01) is switched into 0.



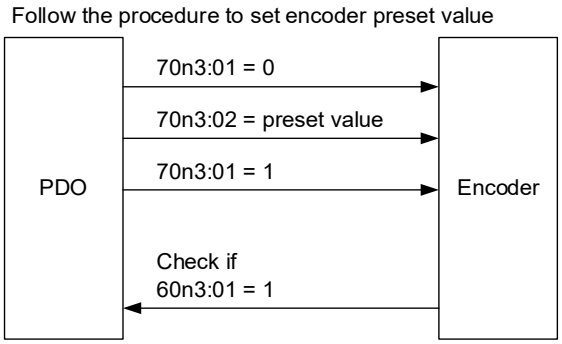
See Chapter 4.2.4 for more details.

4.2.3 Index 70n3: Encoder N Outputs

<table>
<tr><th>Index(hex)</th><th>Name</th><th>Description</th><th>Data Type</th><th>Flags</th><th>Default</th><th>Store</th></tr>
<tr><td>70n3:0</td><td>Encoder N outputs</td><td>Configure the encoder input function.</td><td>UINT8</td><td>RO</td><td>3</td><td>No</td></tr>
<tr><td>70n3:01</td><td>Set Counter</td><td>The switch to set encoder value.
<table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Nothing</td></tr><tr><td>1</td><td>Set encoder counter value</td></tr></table></td><td>UINT8</td><td>RW</td><td>0</td><td>No</td></tr>
<tr><td>70n3:02</td><td>Set Counter Value</td><td>This is the encoder counter value to be set via "Set Counter (index 70n3:01)"</td><td>INT32</td><td>RW</td><td>0</td><td>No</td></tr>
<tr><td>70n3:03</td><td>Enable Auto Reset Counter</td><td>To enable auto reset encoder function.
<table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Disable</td></tr><tr><td>1</td><td>Enable</td></tr></table></td><td>UINT8</td><td>RW</td><td>0</td><td>No</td></tr>
</table>

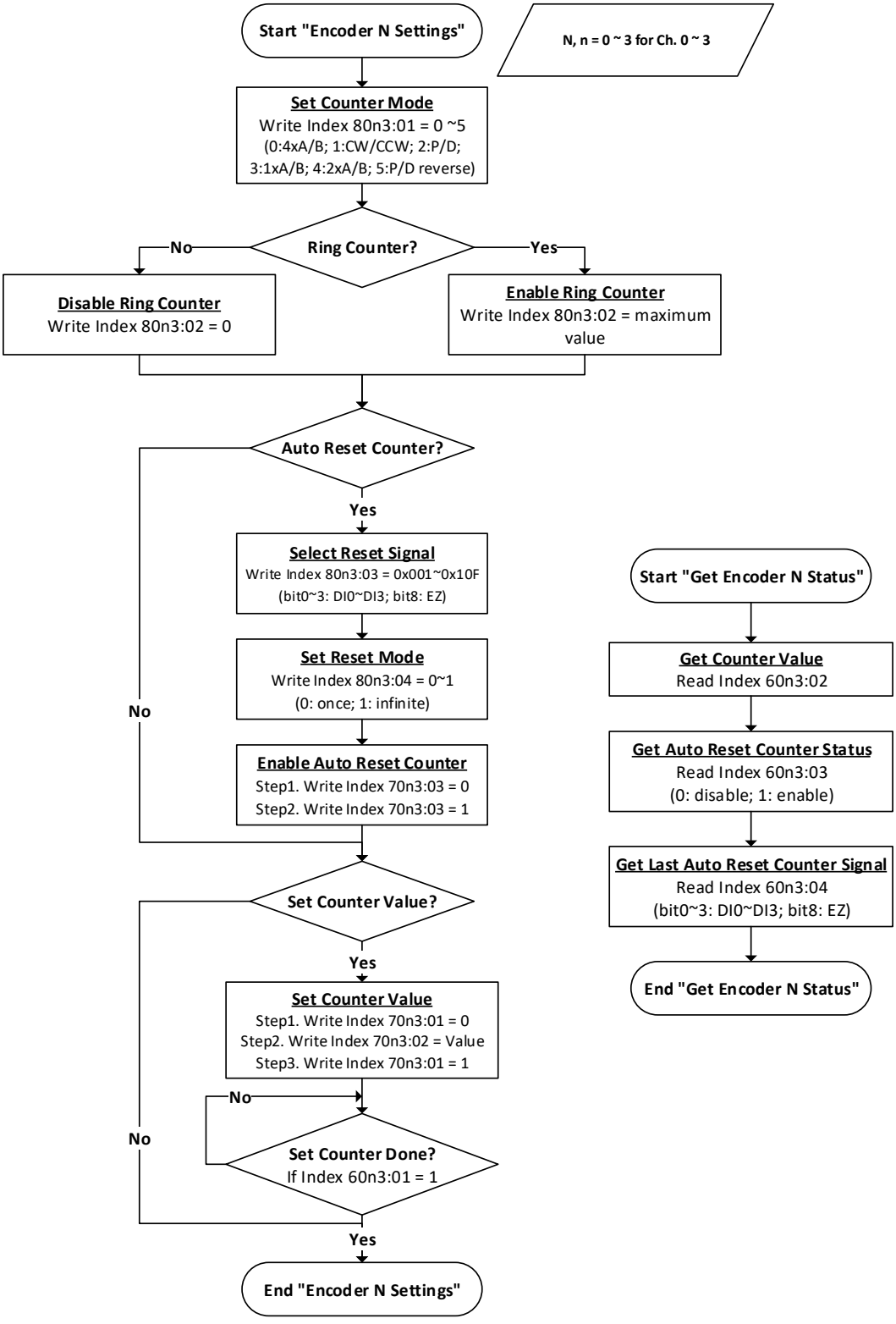
Note Encoder N means encoder channels, N and n = 0, 1, 2, 3.
PDO objects here are merely used for rapid switching on/off.
More parameter configurations are assigned as SDOs.

Note "Auto Reset Counter" is a function to reset encoder value by external signal automatically.
The external signal source could be DI or EZ signal.



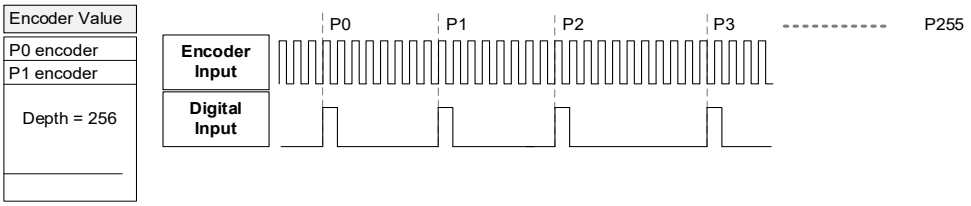
See Chapter 4.2.4 for more details.

4.2.4 Encoder Operation Process



4.3 Latch Function Objects

"FIFO Latch" is a function to latch encoder value by external signal automatically and latch the encoder value continuously up to 256 points for each channel. In other words, the FIFO Latch can store up to 256 encoder values, which allows it to track the position of the encoder over time.



The procedure to operation:

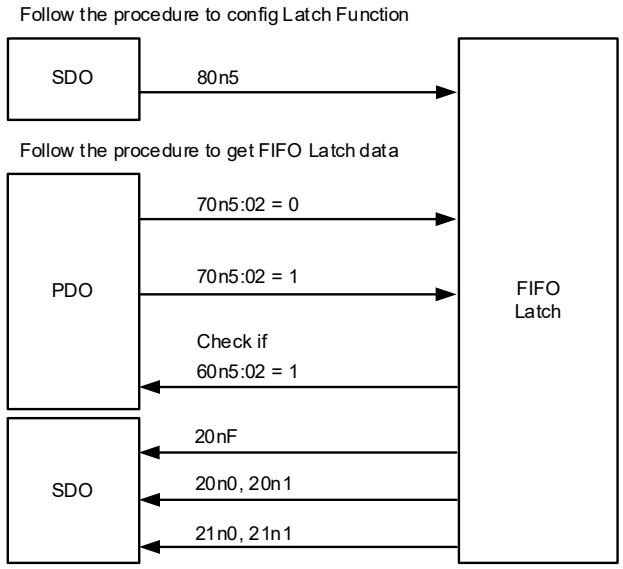
Step1. Latch function configuration

Step2. Enable FIFO Latch procedure if need

Step3. Receiving signals…

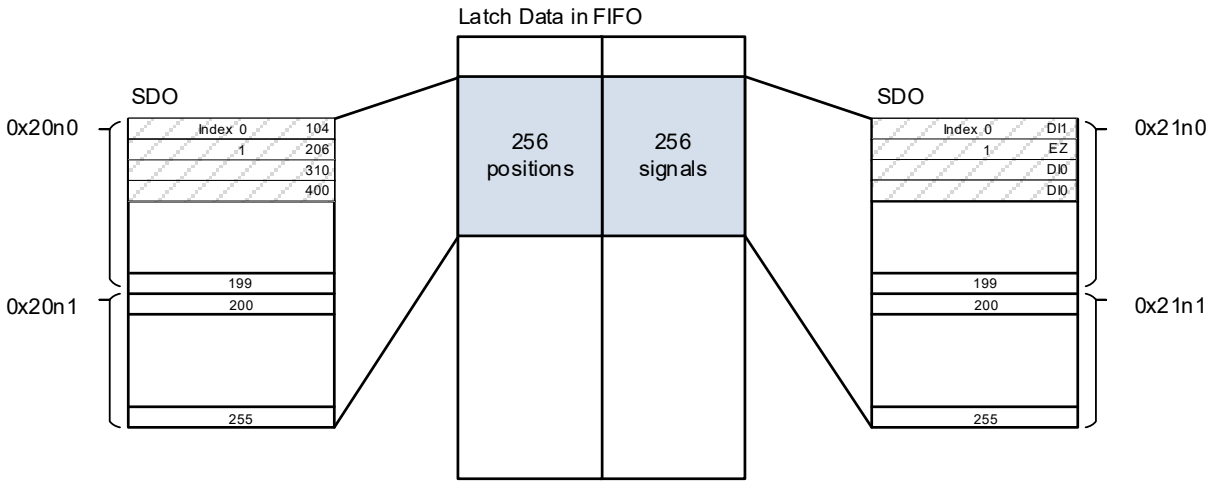
Step4. Check if "FIFO Latch Data Length" > 0 and signal receiving is finished.

Step5. Start to transmit FIFO Latch data from FIFO to SDO table.



See Chapter 4.3.9 for more details.

For example,
Encoder CH0 is latched 4 points and signals source are DI1, EZ, DI0 and DI0 by sequential.
Thus, Table Data Length 0x200F:01 = 4
Latch position table 0x2000 = {104, 206, 310, 400, …, 0} and Latch position table 0x2001 = {0}
Latch signal table 0x2100 = {0x0002(DI1), 0x0100(EZ), 0x0001(DI0), 0x0001(DI0), …, 0} and Latch signal table 0x2101 = {0}



4.3.1 Index 60n5: Latch N Inputs

<table>
<tr><th>Index (hex)</th><th>Name</th><th>Description</th><th>Data Type</th><th>Flags</th><th>Default</th><th>Store</th></tr>
<tr><td>60n5:0</td><td>Latch N inputs</td><td>Display latch status</td><td>UINT8</td><td>RO</td><td>5</td><td>No</td></tr>
<tr><td>60n5:01</td><td>FIFO Latch Status</td><td>Display if FIFO latch is enabled<table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Disable</td></tr><tr><td>1</td><td>Enable</td></tr></table></td><td>UINT8</td><td>RO</td><td>0</td><td>No</td></tr>
<tr><td>60n5:02</td><td>Get FIFO Latch Data Done</td><td>Display if FIFO latch data transmission is completed<table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Disable</td></tr><tr><td>1</td><td>Enable</td></tr></table></td><td>UINT8</td><td>RO</td><td>0</td><td>No</td></tr>
<tr><td>60n5:03</td><td>FIFO Latch Data Length</td><td>The amounts of the latch data since start recording</td><td>UINT16</td><td>RO</td><td>0</td><td>No</td></tr>
<tr><td>60n5:04</td><td>Last Latch Position</td><td>Last latch position</td><td>INT32</td><td>RO</td><td>0</td><td>No</td></tr>
<tr><td>60n5:05</td><td>Last Latch Signal</td><td>Display which signal is the latest source of latch input signal<table><tr><th>Bit</th><th>Description</th></tr><tr><td>0</td><td>DI 0</td></tr><tr><td>1</td><td>DI 1</td></tr><tr><td>2</td><td>DI 2</td></tr><tr><td>3</td><td>DI 3</td></tr><tr><td>4 ~ 7</td><td>Reserved</td></tr><tr><td>8</td><td>EZ N</td></tr><tr><td>9 ~ 15</td><td>Reserved</td></tr></table></td><td>UINT16</td><td>RO</td><td>0</td><td>No</td></tr>
</table>

Note Latch N means latch input channels, N and n = 0, 1, 2, 3.
EZ N means EZ signals, N = 0, 1, because only encoder 0 and 1 have EZ.

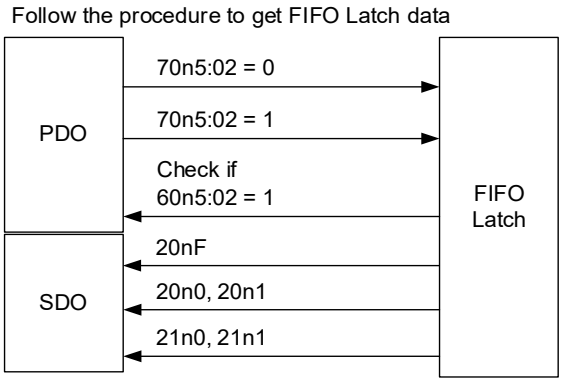
Note The latch value of each channel could be latched by digital input or EZ signal.
Any one of digital input can reset the encoder value of any one channel.

Note Whether FIFO Latch function is enabled or not, Last Latch Position and Last Latch Signal always record the latest one data.

4.3.2 Index 70n5: Latch N Outputs

<table>
<tr><th>Index(hex)</th><th>Name</th><th>Description</th><th>Data Type</th><th>Flags</th><th>Default</th><th>Store</th></tr>
<tr><td>70n5:0</td><td>Latch N outputs</td><td>Transmit FIFO Latch data to SDO</td><td>UINT8</td><td>RO</td><td>2</td><td>No</td></tr>
<tr><td>70n5:01</td><td>Enable FIFO Latch</td><td>To enable FIFO Latch function and clear FIFO
<table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Disable</td></tr><tr><td>1</td><td>Enable</td></tr></table></td><td>UINT8</td><td>RW</td><td>0</td><td>No</td></tr>
<tr><td>70n5:02</td><td>Get FIFO Latch Data</td><td>The switch to start FIFO latch data transmission
<table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Nothing</td></tr><tr><td>1</td><td>Copy data from FIFO to table (index 20nX, 21nX, X=0~1)</td></tr></table></td><td>UINT8</td><td>RW</td><td>0</td><td>No</td></tr>
</table>

Note Latch N means latch input channels, N and n = 0, 1, 2, 3.
PDO objects here are merely used for rapid switching on/off.
More parameter configurations are assigned as SDOs.



See Chapter 4.3.9 for more details.

4.3.3 Index 80n5: Latch N Settings (N = 0 ~ 3)

<table>
<tr><th>Index (hex)</th><th>Name</th><th>Description</th><th>Data Type</th><th>Flags</th><th>Default value</th><th>Store</th></tr>
<tr><td>80n5:0</td><td>Latch N settings</td><td>Configure latch function.</td><td>UINT8</td><td>RO</td><td>3</td><td>No</td></tr>
<tr><td>80n5:01</td><td>Encoder Source</td><td>Select encoder source<table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Encoder 0</td></tr><tr><td>1</td><td>Encoder 1</td></tr><tr><td>2</td><td>Encoder 2</td></tr><tr><td>3</td><td>Encoder 3</td></tr></table></td><td>UINT8</td><td>RW</td><td>N</td><td>Yes</td></tr>
<tr><td>80n5:02</td><td>Select Latch Signal</td><td>Select latch input source<table><tr><th>Bit</th><th>Description</th></tr><tr><td>0</td><td>DI 0</td></tr><tr><td>1</td><td>DI 1</td></tr><tr><td>2</td><td>DI 2</td></tr><tr><td>3</td><td>DI 3</td></tr><tr><td>4 ~ 15</td><td>Reserved</td></tr></table></td><td>UINT16</td><td>RW</td><td>0x01<<N</td><td>Yes</td></tr>
<tr><td>80n5:03</td><td>FIFO Latch Mode</td><td>Set latch mode<table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Once (auto disable)</td></tr><tr><td>1</td><td>Infinite</td></tr></table></td><td>UINT8</td><td>RW</td><td>1</td><td>Yes</td></tr>
</table>

Note Latch N means latch input channels, N and n = 0, 1, 2, 3.

For example, Encoder CH0 is latched 4 points and signals source are DI1, EZ, DI0 and DI0 by sequential.
Table Data Length 0x200F:01 = 4
Latch position table 0x2000 = {104, 206, 310, 400, …, 0} and Latch position table 0x2001 = {0}
Latch signal table 0x2100 = {0x0002(DI1), 0x0100(EZ), 0x0001(DI0), 0x0001(DI0), …, 0} and Latch signal table 0x2101 = {0}

4.3.4 Index 20nF: Latch N Info (N = 0 ~ 3)

Index (hex)	Name	Description	Data Type	Flags	Default	Store
20nF:0	Latch N info	Latch data information	UINT8	RO	3	No
20nF:01	Table Data Length	Data length 0 ~ 256 points that are put into latch N position/signal table.	UINT16	RO	0	No
20nF:02	Table Size	Size of latch N position/signal table Constant 256 points for each latch channel	UINT16	RO	256	No
20nF:03	FIFO Size	Size of latch N FIFO Constant 256 points for each latch channel	UINT16	RO	256	No

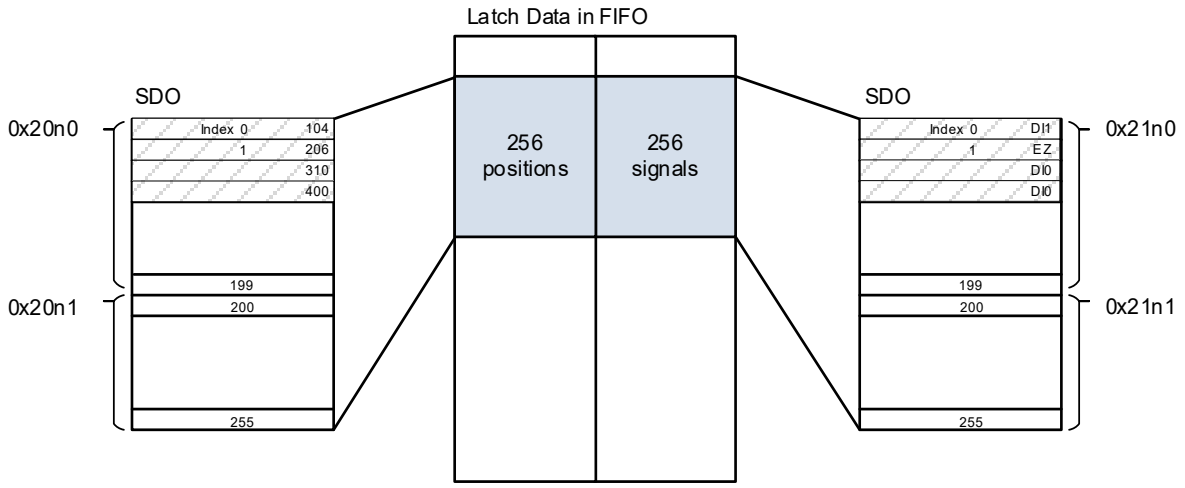
Note Latch N means latch input channels, N and n = 0, 1, 2, 3.

For example, Encoder CH0 is latched 4 points and signals source are DI1, EZ, DI0 and DI0 by sequential.

Table Data Length 0x200F:01 = 4

Latch position table 0x2000 = {104, 206, 310, 400, …, 0} and Latch position table 0x2001 = {0}

Latch signal table 0x2100 = {0x0002(DI1), 0x0100(EZ), 0x0001(DI0), 0x0001(DI0), …, 0} and Latch signal table 0x2101 = {0}



4.3.5 Index 20n0: Latch N Position Table Index 0 - 199 (N = 0 ~ 3)

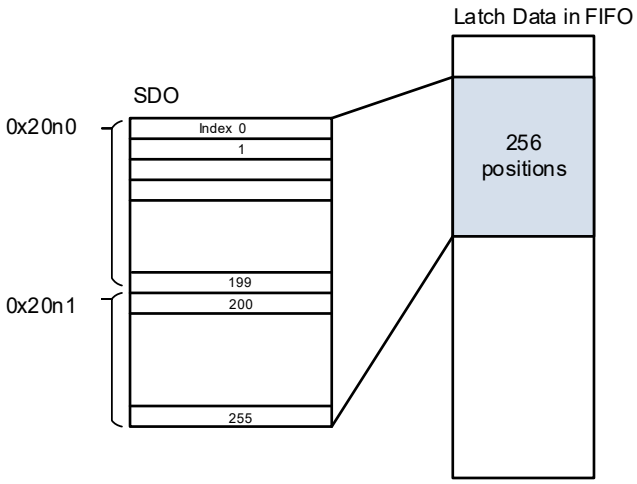
Index (hex)	Name	Description	Data Type	Flags	Default value	Store
20n0:0	Latch N position table index 0 - 199	Latch position table low part.	UINT8	RO	200	No
20n0:01	Subindex 001	Latch N position table index 0	INT32	RO	0	No
…	…	…	INT32	RO	0	No
20n0:C8	Subindex 200	Latch N position table index 199	INT32	RO	0	No

4.3.6 Index 20n1: Latch N Position Table Index 200 - 255 (N = 0 ~ 3)

Index (hex)	Name	Description	Data Type	Flags	Default value	Store
20n1:0	Latch N position table index 200 - 255	Latch position table high part.	UINT8	RO	56	No
20n1:01	Subindex 001	Latch N position table index 200	INT32	RO	0	No
…	…	…	INT32	RO	0	No
20n1:38	Subindex 056	Latch N position table index 255	INT32	RO	0	No

Note Latch N means latch input channels, N and n = 0, 1, 2, 3.

PDO Latch N Outputs 0x70n5 is used to activate data transmission from FIFO to SDO.



4.3.7 Index 21n0: Latch N Signal Table Index 0 - 199 (N = 0 ~ 3)

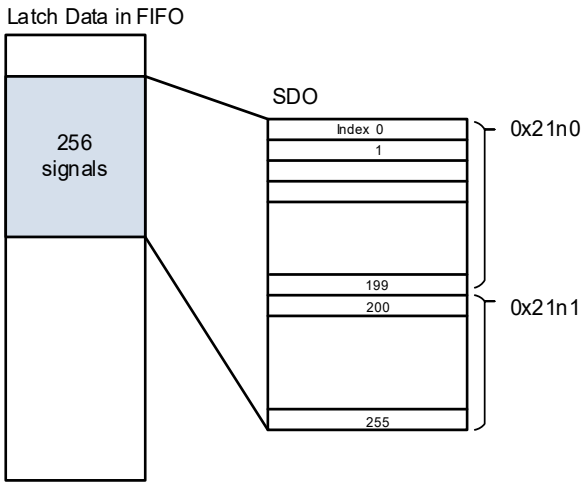
Index (hex)	Name	Description	Data Type	Flags	Default	Store
21n0:0	Latch N signal table index 0 - 199	Latch signal table low part.	UINT8	RO	200	No
21n0:01	Subindex 001	Latch N signal table index 0	UINT16	RO	0	No
…	…	…	UINT16	RO	0	No
21n0:C8	Subindex 200	Latch N signal table index 199	UINT16	RO	0	No

4.3.8 Index 21n1: Latch N Signal Table Index 200 - 255 (N = 0 ~ 3)

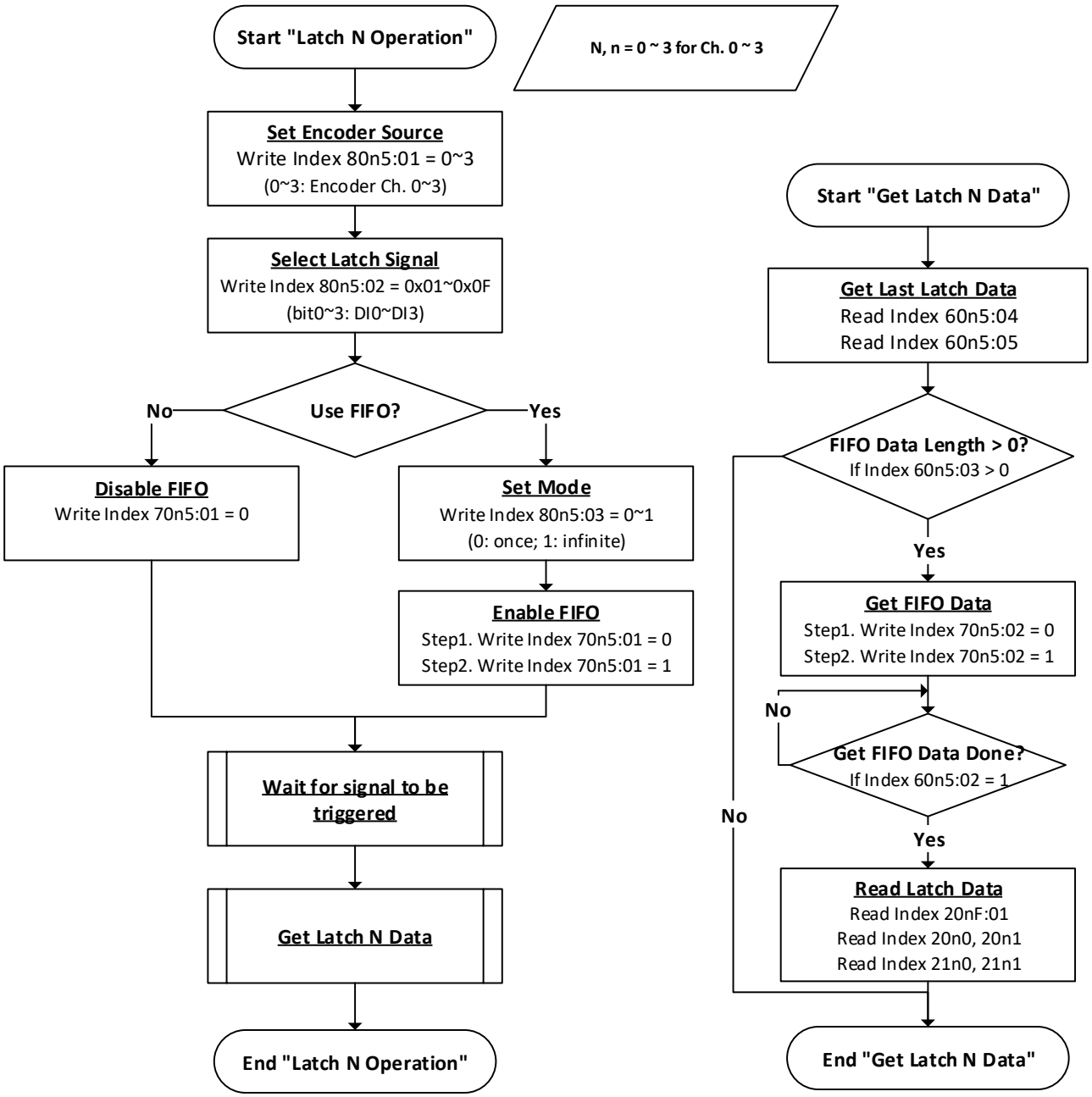
Index (hex)	Name	Description	Data Type	Flags	Default	Store
21n1:0	Latch N signal table index 200 - 255	Latch signal table high part.	UINT8	RO	56	No
21n1:01	Subindex 001	Latch N signal table index 200	UINT16	RO	0	No
…	…	…	UINT16	RO	0	No
21n1:38	Subindex 056	Latch N signal table index 255	UINT16	RO	0	No

Note Latch N means latch input channels, N and n = 0, 1, 2, 3.

PDO Latch N Outputs 0x70n5 is used to activate data transmission from FIFO to SDO.



4.3.9 Latch Operation Process



4.4 CMP Trigger Objects

The module provides 4 different purpose CMP trigger modes, Fixed Interval (Auto), Table, Toggle and Multi-Fixed Interval.

Compare trigger is used to generate an output pulse when an encoder counter reaches a predefined compare position. This operation is typically executed by hardware and is designed to be virtually instantaneous without any time delay, ensuring precise position control or measurement.

Auto Trigger: Perform fixed interval or equidistant compare trigger with preset starting position.

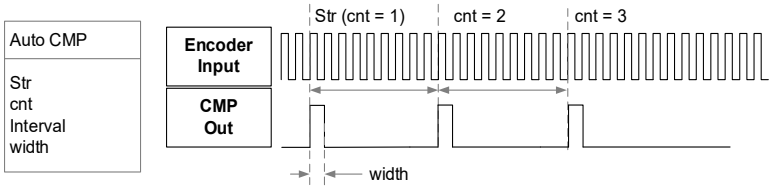
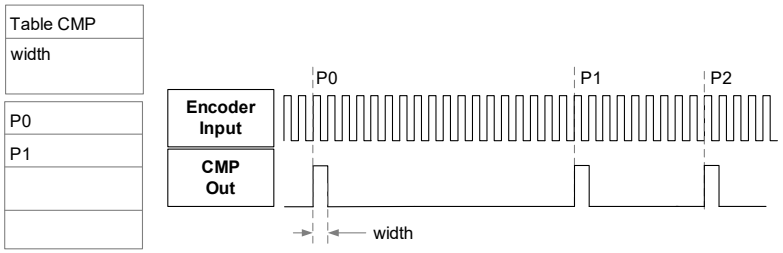
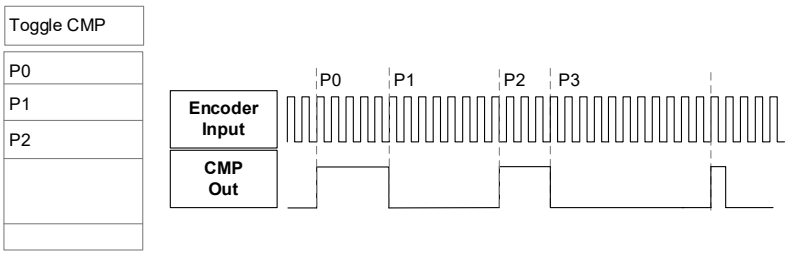


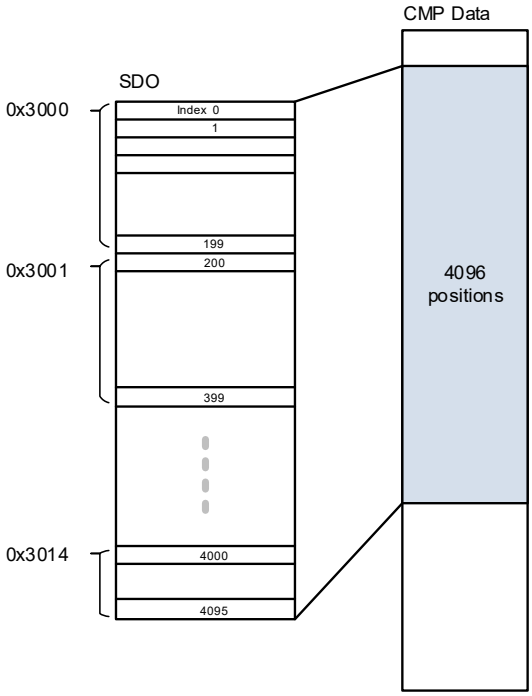
Table Trigger: Perform arbitrary position or preset table array compare trigger.



Toggle Trigger: Perform arbitrary position and arbitrary pulse width compare trigger.

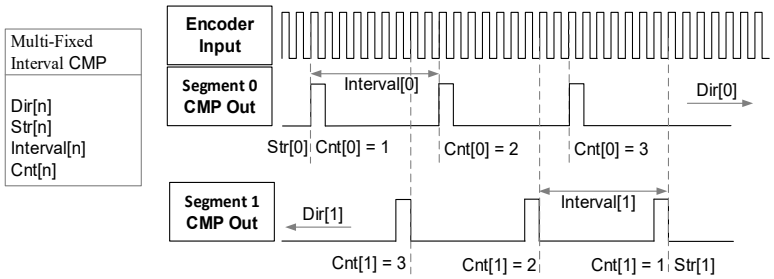


Assign the array position value into SDO for Table Trigger and Toggle Trigger and the SDO objects are CMP Table Index 0 ~ 4095. It means that there are 4096 points shared by all CMP channels.



Multi-Fixed Interval Trigger: Perform sequential fixed interval triggers automatically. Users can define multiple "Segments" in the CMP table, and the module will execute them in order. Segment structure in CMP Table: In this mode, every 4 entries in the CMP table form one Segment.

Entry 1: Direction
Entry 2: Start Position
Entry 3: Interval Distance
Entry 4: Trigger Count



4.4.1 Index 60n4: CMP N Inputs

<table>
<tr><th>Index (hex)</th><th>Name</th><th>Description</th><th>Data Type</th><th>Flags</th><th>Default</th><th>Store</th></tr>
<tr><td>60n4:0</td><td>CMP N inputs</td><td>Display CMP status</td><td>UINT8</td><td>RO</td><td>2</td><td>No</td></tr>
<tr><td>60n4:01</td><td>CMP Status</td><td>Display if CMP is enabled.<table><tr><th>Bit</th><th>Description</th></tr><tr><td>0</td><td>CMP Enabled</td></tr><tr><td>1</td><td>Sequence Busy (valid for trigger type 4)</td></tr><tr><td>2 ~ 15</td><td>Reserved</td></tr></table></td><td>UINT8</td><td>RO</td><td>0</td><td>No</td></tr>
<tr><td>60n4:02</td><td>Actual Trigger Count</td><td>Current actual trigger counter value</td><td>UINT32</td><td>RO</td><td>0</td><td>No</td></tr>
</table>

Note CMP N means compare output channels, N and n = 0, 1, 2, 3.

4.4.2 Index 70n4: CMP N Outputs

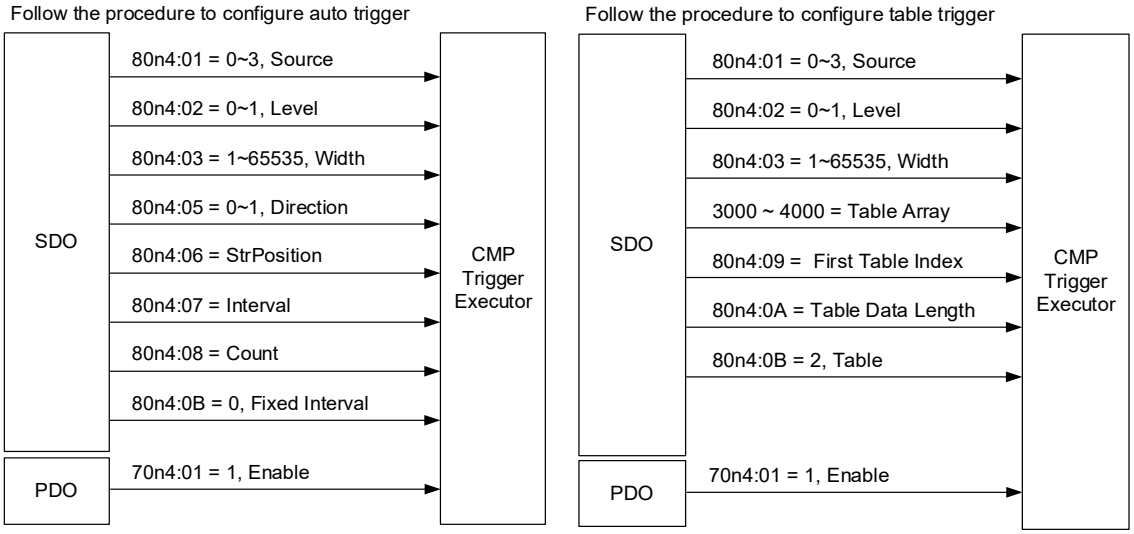
Index(hex)	Name	Description	Data Type	Flags	Default	Store
70n4:0	CMP N outputs	Configure CMP function.	UINT8	RO	1	No
70n4:01	Enable CMP	To enable CMP function. Value: 0 = Disable; 1 = Enable	UINT8	RW	0	No

Note CMP N means compare output channels, N and n = 0, 1, 2, 3.

PDO objects here are merely used for rapid switching on/off.

More parameter configurations are assigned as SDOs.

Do not modify parameters after CMP has been switched to “Enable.”



See Chapter 4.4.26 for more details.

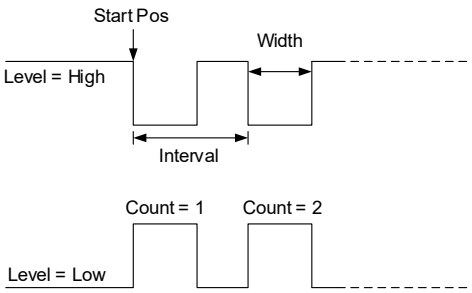
4.4.3 Index 80n4: CMP N Settings (N = 0 ~ 3)

<table>
<tr><th>Index (hex)</th><th>Name</th><th>Description</th><th>Data Type</th><th>Flags</th><th>Default value</th><th>Store</th></tr>
<tr><td>80n4:0</td><td>CMP N settings</td><td>Configure CMP function.</td><td>UINT8</td><td>RO</td><td>11</td><td>No</td></tr>
<tr><td>80n4:01</td><td>Encoder Source</td><td>Select encoder source
<table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Encoder 0</td></tr><tr><td>1</td><td>Encoder 1</td></tr><tr><td>2</td><td>Encoder 2</td></tr><tr><td>3</td><td>Encoder 3</td></tr></table></td><td>UINT8</td><td>RW</td><td>N</td><td>Yes</td></tr>
<tr><td>80n4:02</td><td>Trigger Level</td><td>Set trigger level
<table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Normal low</td></tr><tr><td>1</td><td>Normal high</td></tr></table></td><td>UINT8</td><td>RW</td><td>1</td><td>Yes</td></tr>
<tr><td>80n4:03</td><td>Pulse Width</td><td>Width of the triggering duration
Duration range is 1 ~ 65535, so that pulse width = duration * unit
Unit: 1us or 0.1us (see index 80n4:04)</td><td>UINT16</td><td>RW</td><td>10</td><td>Yes</td></tr>
<tr><td>80n4:04</td><td>Pulse Width Unit</td><td>Set pulse width unit
<table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>1 us</td></tr><tr><td>1</td><td>0.1 us</td></tr></table></td><td>UINT8</td><td>RW</td><td>0</td><td>Yes</td></tr>
<tr><td>80n4:05</td><td>Direction</td><td>Motion direction which enables auto compare trigger (valid when index 80n4:0B = 1)
<table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Negative direction</td></tr><tr><td>1</td><td>Positive direction</td></tr></table></td><td>UINT8</td><td>RW</td><td>1</td><td>Yes</td></tr>
<tr><td>80n4:06</td><td>Start Position</td><td>The start position which auto compare trigger begins (valid when index 80n4:0B = 1)</td><td>INT32</td><td>RW</td><td>10</td><td>Yes</td></tr>
<tr><td>80n4:07</td><td>Interval</td><td>The distance interval for regularly auto compare trigger (valid when index 80n4:0B = 1)</td><td>UINT32</td><td>RW</td><td>10</td><td>Yes</td></tr>
<tr><td>80n4:08</td><td>Trigger Count</td><td>The amounts of the trigger since start auto compare triggering (valid when index 80n4:0B = 1)
0 means unlimited counts until call</td><td>UINT32</td><td>RW</td><td>0</td><td>Yes</td></tr>
</table>

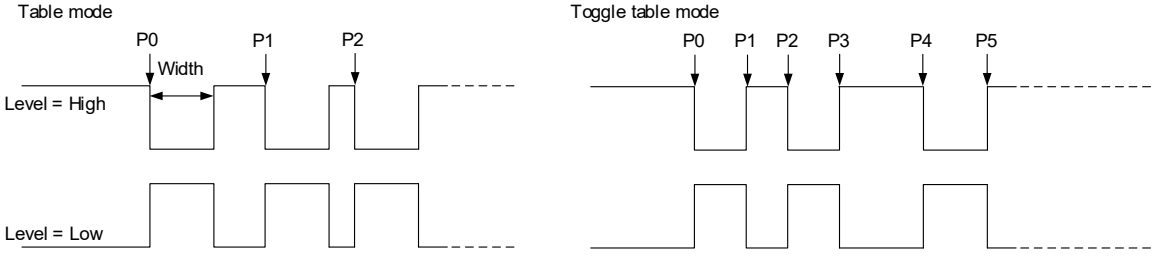
		stop				
80n4:09	First Table Index	The first index of the programmed entries in the CMP table. (valid when index 80n4:0B = 2, 3 or 4)	UINT32	RW	1024*N	Yes
80n4:0A	Table Data Length	The total length (number of entries) of the programmed data in the CMP table. (valid when index 80n4:0B = 2, 3 or 4) Note for type 4: The value must be the total number of parameters across all segments. Length = Number of Segments x 4	UINT32	RW	10	Yes
80n4:0B	Trigger Type	Set CMP Trigger Type **Value** / **Description** 0 Force Trigger 1 Fixed Interval 2 Table 3 Toggle Table 4 Multi-Fixed Interval	UINT8	RW	0	Yes

Note CMP N means compare output channels, N and n = 0, 1, 2, 3.
PDO CMP N Outputs 0x70n4 is used to activate CMP function.

Note Start Position, Interval and Trigger Count are available for Fixed Interval mode.

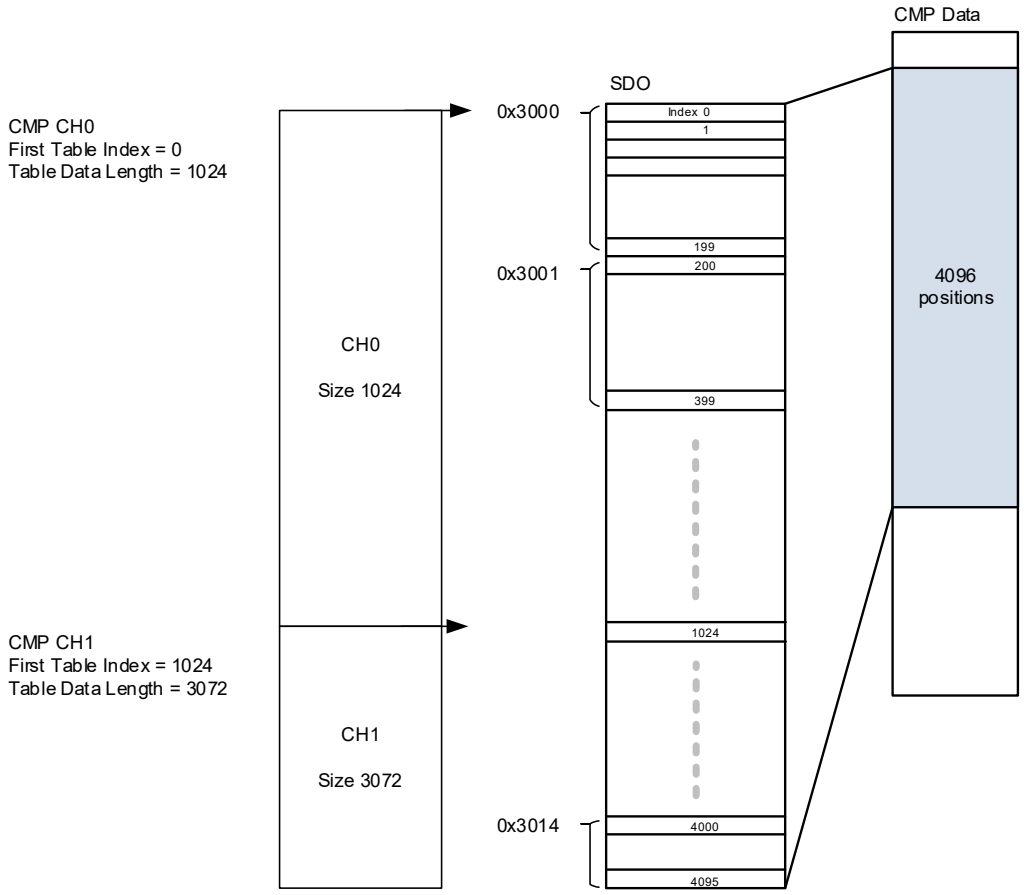


Note First Table Index Table Data Length are available for Table and Toggle Table.



Note CMP table is shared for four channels so set First Table Index and Table Data Length to allocate table size for each channel.

For example, allocate 1024 points to CH0 and 3072 points to CH1.



Note When the Trigger Type is set to 4 (Multi-Fixed Interval), the data in the CMP table is interpreted as groups of parameters called "Segments". The firmware reads 4 entries sequentially for each segment.

Segment Data Structure: Users must define the segments following the 4-entry sequence below:

1st Entry: Direction

2st Entry: Start Position

3st Entry: Interval

4st Entry: Trigger Count

Example: If First Table Index (0x80n4:09) is set to 0, the first segment (Segment 0) occupies Index 3000:01 to 3000:04. The second segment (Segment 1) occupies Index 3000:05 to 3000:08, and so on.

Important: The Table Data Length (Index 80n4:0A) must be set to the total number of entries (Number of Segments × 4).

4.4.4 Index 3FFF: CMP Table Info

Index (hex)	Name	Description	Data Type	Flags	Default	Store
3FFF:0	CMP table info	CMP table information	UINT8	RO	1	No
3FFF:01	Table Size	Size of CMP table Constant 4096 points for all channels	UINT32	RO	4096	No

4.4.5 Index 3000: CMP Table Index 0 - 199

Index (hex)	Name	Description	Data Type	Flags	Default value	Store
3000:0	CMP table index 0 - 199	Number of entries	UINT8	RO	200	No
3000:01	Subindex 001	CMP table index 0	INT32	RW	0	No
…	…	…	INT32	RW	0	No
3000:C8	Subindex 200	CMP table index 199	INT32	RW	0	No

4.4.6 Index 3001: CMP Table Index 200 - 399

Index (hex)	Name	Description	Data Type	Flags	Default value	Store
3001:0	CMP table index 200 - 399	Number of entries	UINT8	RO	200	No
3001:01	Subindex 001	CMP table index 200	INT32	RW	0	No
…	…	…	INT32	RW	0	No
3001:C8	Subindex 200	CMP table index 399	INT32	RW	0	No

4.4.7 Index 3002: CMP Table Index 400 - 599

Index (hex)	Name	Description	Data Type	Flags	Default value	Store
3002:0	CMP table index 400 - 599	Number of entries	UINT8	RO	200	No
3002:01	Subindex 001	CMP table index 400	INT32	RW	0	No
…	…	…	INT32	RW	0	No
3002:C8	Subindex 200	CMP table index 599	INT32	RW	0	No

4.4.8 Index 3003: CMP Table Index 600 - 799

Index (hex)	Name	Description	Data Type	Flags	Default value	Store
3003:0	CMP table index 600 - 799	Number of entries	UINT8	RO	200	No
3003:01	Subindex 001	CMP table index 600	INT32	RW	0	No
…	…	…	INT32	RW	0	No
3003:C8	Subindex 200	CMP table index 799	INT32	RW	0	No

4.4.9 Index 3004: CMP Table Index 800 - 999

Index (hex)	Name	Description	Data Type	Flags	Default value	Store
3004:0	CMP table index 800 - 999	Number of entries	UINT8	RO	200	No
3004:01	Subindex 001	CMP table index 800	INT32	RW	0	No
…	…	…	INT32	RW	0	No
3004:C8	Subindex 200	CMP table index 999	INT32	RW	0	No

4.4.10 Index 3005: CMP Table Index 1000 - 1199

Index (hex)	Name	Description	Data Type	Flags	Default value	Store
3005:0	CMP table index 1000 - 1199	Number of entries	UINT8	RO	200	No
3005:01	Subindex 001	CMP table index 1000	INT32	RW	0	No
…	…	…	INT32	RW	0	No
3005:C8	Subindex 200	CMP table index 1199	INT32	RW	0	No

4.4.11 Index 3006: CMP Table Index 1200 - 1399

Index (hex)	Name	Description	Data Type	Flags	Default value	Store
3006:0	CMP table index 1200 - 1399	Number of entries	UINT8	RO	200	No
3006:01	Subindex 001	CMP table index 1200	INT32	RW	0	No
…	…	…	INT32	RW	0	No
3006:C8	Subindex 200	CMP table index 1399	INT32	RW	0	No

4.4.12 Index 3007: CMP Table Index 1400 – 1599

Index (hex)	Name	Description	Data Type	Flags	Default value	Store
3007:0	CMP table index 1400 - 1599	Number of entries	UINT8	RO	200	No
3007:01	Subindex 001	CMP table index 1400	INT32	RW	0	No
…	…	…	INT32	RW	0	No
3007:C8	Subindex 200	CMP table index 1599	INT32	RW	0	No

4.4.13 Index 3008: CMP Table Index 1600 - 1799

Index (hex)	Name	Description	Data Type	Flags	Default value	Store
3008:0	CMP table index 1600 - 1799	Number of entries	UINT8	RO	200	No
3008:01	Subindex 001	CMP table index 1600	INT32	RW	0	No
…	…	…	INT32	RW	0	No
3008:C8	Subindex 200	CMP table index 1799	INT32	RW	0	No

4.4.14 Index 3009: CMP Table Index 1800 - 1999

Index (hex)	Name	Description	Data Type	Flags	Default value	Store
3009:0	CMP table index 1800 - 1999	Number of entries	UINT8	RO	200	No
3009:01	Subindex 001	CMP table index 1800	INT32	RW	0	No
…	…	…	INT32	RW	0	No
3009:C8	Subindex 200	CMP table index 1999	INT32	RW	0	No

4.4.15 Index 300A: CMP Table Index 2000 - 2199

Index (hex)	Name	Description	Data Type	Flags	Default value	Store
300A:0	CMP table index 2000 - 2199	Number of entries	UINT8	RO	200	No
300A:01	Subindex 001	CMP table index 2000	INT32	RW	0	No
…	…	…	INT32	RW	0	No
300A:C8	Subindex 200	CMP table index 2199	INT32	RW	0	No

4.4.16 Index 300B: CMP Table Index 2200 – 2399

Index (hex)	Name	Description	Data Type	Flags	Default value	Store
300B:0	CMP table index 2200 - 2399	Number of entries	UINT8	RO	200	No
300B:01	Subindex 001	CMP table index 2200	INT32	RW	0	No
…	…	…	INT32	RW	0	No
300B:C8	Subindex 200	CMP table index 2399	INT32	RW	0	No

4.4.17 Index 300C: CMP Table Index 2400 - 2599

Index (hex)	Name	Description	Data Type	Flags	Default value	Store
300C:0	CMP table index 2400 - 2599	Number of entries	UINT8	RO	200	No
300C:01	Subindex 001	CMP table index 2400	INT32	RW	0	No
…	…	…	INT32	RW	0	No
300C:C8	Subindex 200	CMP table index 2599	INT32	RW	0	No

4.4.18 Index 300D: CMP Table Index 2600 - 2799

Index (hex)	Name	Description	Data Type	Flags	Default value	Store
300D:0	CMP table index 2600 - 2799	Number of entries	UINT8	RO	200	No
300D:01	Subindex 001	CMP table index 2600	INT32	RW	0	No
…	…	…	INT32	RW	0	No
300D:C8	Subindex 200	CMP table index 2799	INT32	RW	0	No

4.4.19 Index 300E: CMP Table Index 2800 - 2999

Index (hex)	Name	Description	Data Type	Flags	Default value	Store
300E:0	CMP table index 2800 - 2999	Number of entries	UINT8	RO	200	No
300E:01	Subindex 001	CMP table index 2800	INT32	RW	0	No
…	…	…	INT32	RW	0	No
300E:C8	Subindex 200	CMP table index 2999	INT32	RW	0	No

4.4.20 Index 300F: CMP Table Index 3000 – 3199

Index (hex)	Name	Description	Data Type	Flags	Default value	Store
300F:0	CMP table index 3000 - 3199	Number of entries	UINT8	RO	200	No
300F:01	Subindex 001	CMP table index 3000	INT32	RW	0	No
…	…	…	INT32	RW	0	No
300F:C8	Subindex 200	CMP table index 3199	INT32	RW	0	No

4.4.21 Index 3010: CMP Table Index 3200 - 3399

Index (hex)	Name	Description	Data Type	Flags	Default value	Store
3010:0	CMP table index 3200 - 3399	Number of entries	UINT8	RO	200	No
3010:01	Subindex 001	CMP table index 3200	INT32	RW	0	No
…	…	…	INT32	RW	0	No
3010:C8	Subindex 200	CMP table index 3399	INT32	RW	0	No

4.4.22 Index 3011: CMP Table Index 3400 - 3599

Index (hex)	Name	Description	Data Type	Flags	Default value	Store
3011:0	CMP table index 3400 - 3599	Number of entries	UINT8	RO	200	No
3011:01	Subindex 001	CMP table index 3400	INT32	RW	0	No
…	…	…	INT32	RW	0	No
3011:C8	Subindex 200	CMP table index 3599	INT32	RW	0	No

4.4.23 Index 3012: CMP Table Index 3600 - 3799

Index (hex)	Name	Description	Data Type	Flags	Default value	Store
3012:0	CMP table index 3600 - 3799	Number of entries	UINT8	RO	200	No
3012:01	Subindex 001	CMP table index 3600	INT32	RW	0	No
…	…	…	INT32	RW	0	No
3012:C8	Subindex 200	CMP table index 3799	INT32	RW	0	No

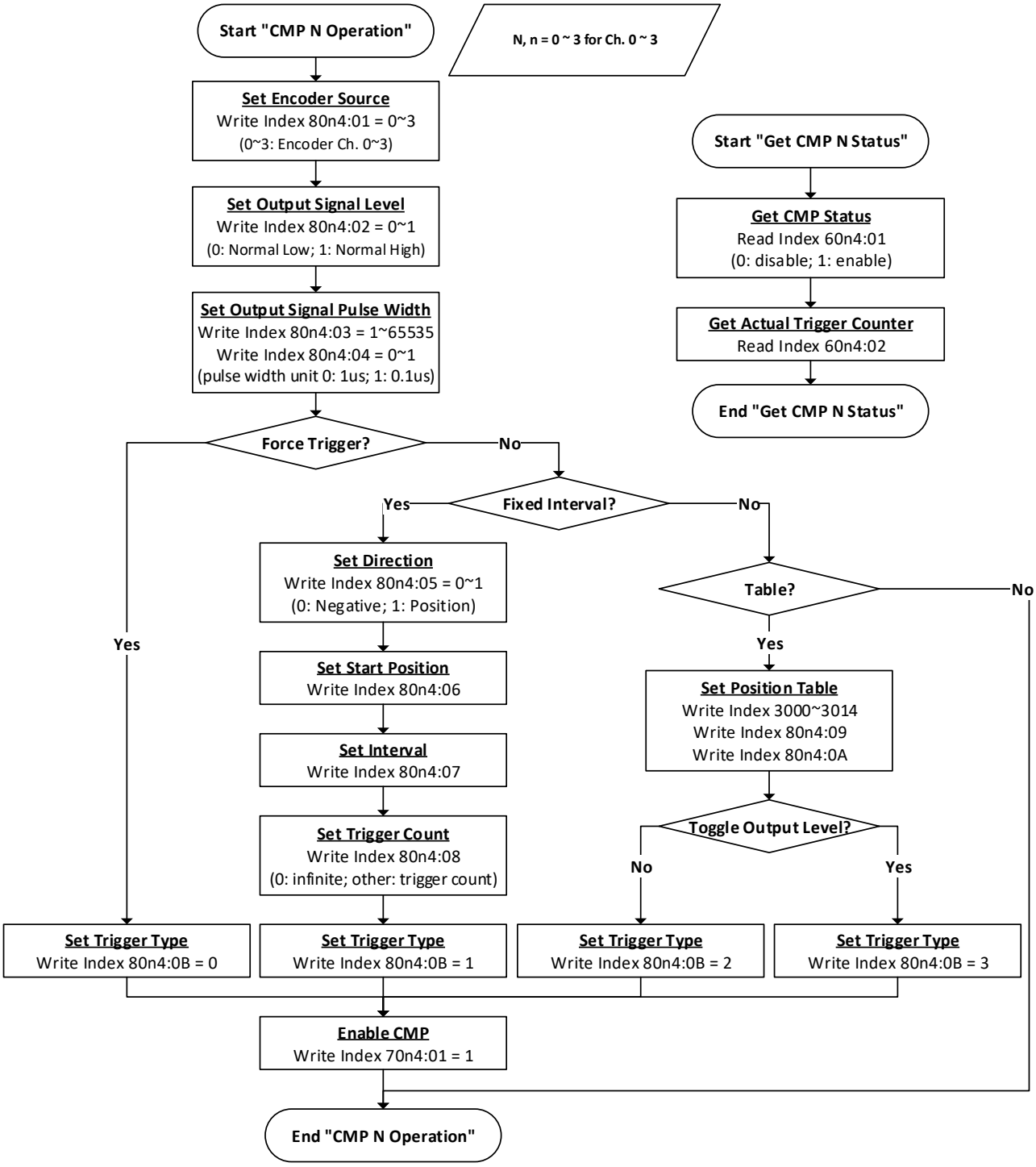
4.4.24 Index 3013: CMP Table Index 3800 – 3999

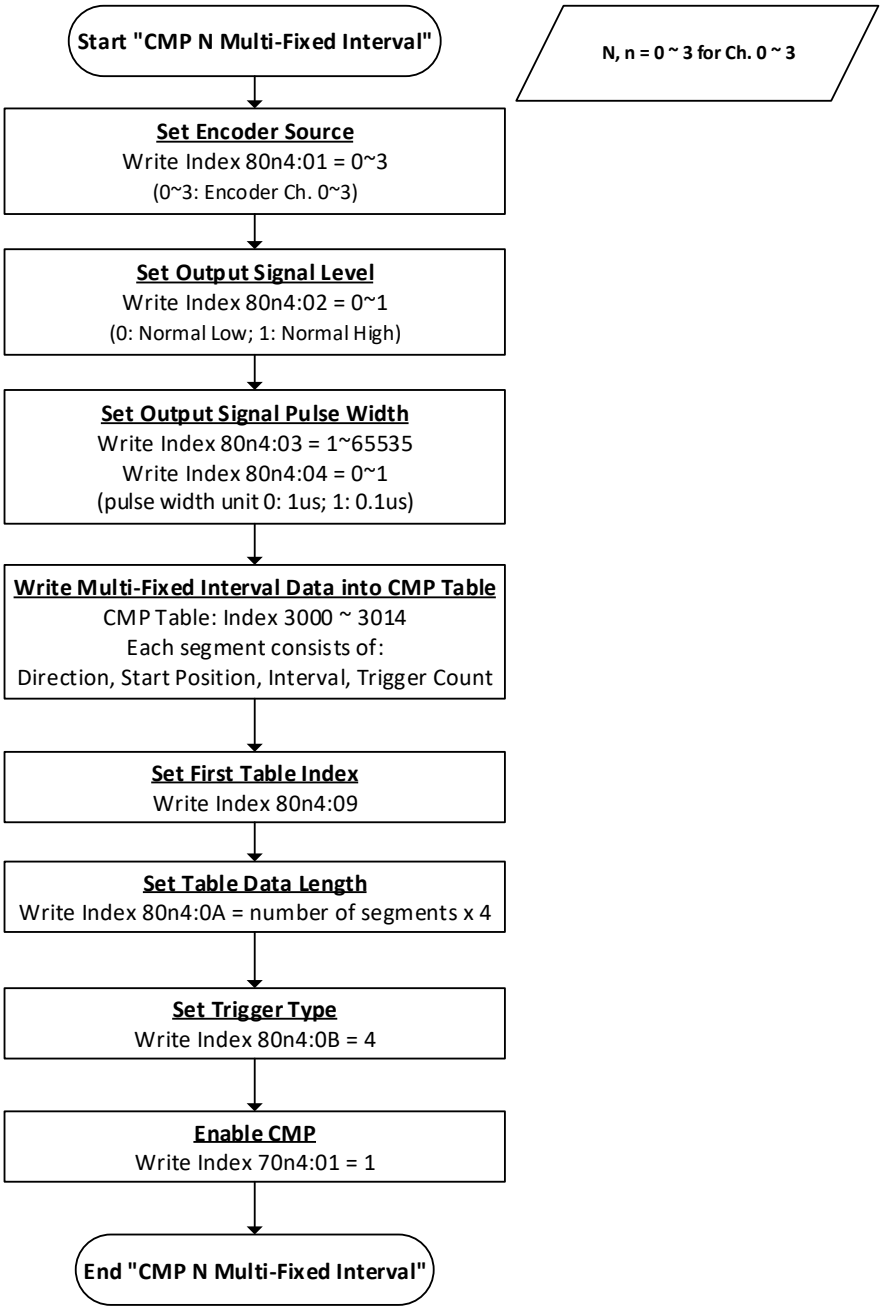
Index (hex)	Name	Description	Data Type	Flags	Default value	Store
3013:0	CMP table index 3800 - 3999	Number of entries	UINT8	RO	200	No
3013:01	Subindex 001	CMP table index 3800	INT32	RW	0	No
…	…	…	INT32	RW	0	No
3013:C8	Subindex 200	CMP table index 3999	INT32	RW	0	No

4.4.25 Index 3014: CMP Table Index 4000 - 4095

Index (hex)	Name	Description	Data Type	Flags	Default value	Store
3014:0	CMP table index 4000 - 4095	Number of entries	UINT8	RO	96	No
3014:01	Subindex 001	CMP table index 4000	INT32	RW	0	No
…	…	…	INT32	RW	0	No
3014:60	Subindex 096	CMP table index 4095	INT32	RW	0	No

4.4.26 Compare Operation Process





Note After enabling the function, writing SDO is inhibited.
So must disable the function to write new parameters.

5. Standard Objects

SDO (Service Data Object) objects can be accessed or managed within the context of the EtherCAT Status Machine (ESM) by following a specific set of rules.

ESM State	SDO Communication	Transmit PDO (TxPDO)	Receive PDO (RxPDO)
Init	Not Available	Not Available	Not Available
Pre-operational	Available	Not Available	Not Available
Safe-operational	Available	Available	Not Available
Operational	Available	Available	Available

5.1 Index 1000: Device Type

Index (hex)	Name	Description	Data Type	Flags	Default value	Store
1000:0	Device type	Device type	UINT32	RO	0x00001389	No

5.2 Index 1008: Device Name

Index (hex)	Name	Description	Data Type	Flags	Default value	Store
1008:0	Device name	Device name	STRING	RO	EZE-C344F	No

5.3 Index 1009: Hardware Version

Index (hex)	Name	Description	Data Type	Flags	Default value	Store
1009:0	Hardware version	Hardware version	STRING	RO	0.0.0.0	No

5.4 Index 100A: Software Version

Index (hex)	Name	Description	Data Type	Flags	Default value	Store
100A:0	Software version	Firmware version (the value is same as index 1018:04)	STRING	RO	0.0.0.0	No

5.5 Index 1010: Store Parameters

Index(hex)	Name	Description	Data Type	Flags	Default value	Store
1010:0	Store parameters	Number of entries	UINT8	RO	1	No
1010:01	SubIndex 001	This object is used to write (back up) the objects data into flash memory. When user writes **0x65766173** ("save") into index 1010:01, system will back up the whole target objects with retain function into flash memory.	UINT32	RW	0x00000000	No

The parameters are stored in ram or flash of the module. The parameters stored in RAM will be discarded once the power is off. The parameters stored in flash are saved even if the control power is off. Hence, Store parameter is used to save parameter in flash and replace the default value.

ESM State	SDO Communication	Store Parameters
Init	Not Available	Not Available
Pre-operational	Available	Available
Safe-operational	Available	Available
Operational	Available	Available

5.6 Index 1011: Restore Default Parameters

Index(hex)	Name	Description	Data Type	Flags	Default value	Store
1011:0	Restore default parameters	Number of entries	UINT8	RO	1	No
1011:01	SubIndex 001	This object is used to restore the objects to default values from flash memory. When user writes **0x64616F6C** ("load") into index 1010:01, system will start to restore the whole target objects with retain function from flash memory.	UINT32	RW	0x00000000	No

5.7 Index 1018: Identity

Index(hex)	Name	Description	Data Type	Flags	Default value	Store
1018:0	Identity	Number of entries	UINT8	RO	4	No
1018:01	Vendor ID	Vendor ID of TPM	UINT32	RO	0x000006AB	No
1018:02	Product code	Product code	UINT32	RO	0x43344103	No
1018:03	Revision	ESI file version	UINT32	RO	0x00000000	No
1018:04	Serial number	Firmware version (the value is same as index 100A:0)	UINT32	RO	0x00000000	No

5.8 Index 4000: Station Alias

<table>
<tr><th>Index (hex)</th><th>Name</th><th>Description</th><th>Data Type</th><th>Flags</th><th>Default value</th><th>Store</th></tr>
<tr><td>4000:0</td><td>Station alias</td><td>Number of entries</td><td>UINT8</td><td>RO</td><td>4</td><td>No</td></tr>
<tr><td>4000:01</td><td>Station Alias Selection</td><td>How to set a station alias
<table>
<tr><th>Value</th><th>Description</th></tr>
<tr><td>0</td><td>The value saved at 0004h in the SII is set as station alias</td></tr>
<tr><td>1</td><td>The value made of index 4000:02 and dip switch of amplifier is set as station alias*</td></tr>
<tr><td>other</td><td>Reserved</td></tr>
</table>
*If setting values for both the dip switch and index 4000:02 are 0, the value of the SII area (0004h) is regard as station alias</td><td>UINT8</td><td>RW</td><td>0x01</td><td>Yes</td></tr>
<tr><td>4000:02</td><td>Station High Byte</td><td>High byte of station alias
<table>
<tr><th colspan="2">Station Alias (16 bits)</th></tr>
<tr><th>High byte</th><th>Low byte</th></tr>
<tr><td>Value set by index 4000:02</td><td>Value set by dip switch</td></tr>
</table></td><td>UINT8</td><td>RW</td><td>0x00</td><td>Yes</td></tr>
<tr><td>4000:03</td><td>Station Switch</td><td>This parameter is to show the value of station ID switch which is in front of device
This value will be refreshed when the switch is changed</td><td>UINT8</td><td>RO</td><td>0x00</td><td>No</td></tr>
<tr><td>4000:04</td><td>Station Alias</td><td>This parameter is to show the station alias which will be refreshed After power on</td><td>UINT16</td><td>RO</td><td>0x0000</td><td>No</td></tr>
</table>

5.8.1 Device Addressing

The device can be addressed via "device position address (auto increment address)", "node address (configured station address/configured station alias)" or "a broadcast".

- Position Addressing (auto increment addressing)

In this mode, the datagram holds the position address of the addressed slave as a negative value. Each slave increments the address. The slave which reads the address equal zero is addressed and will execute the appropriate command at receives.

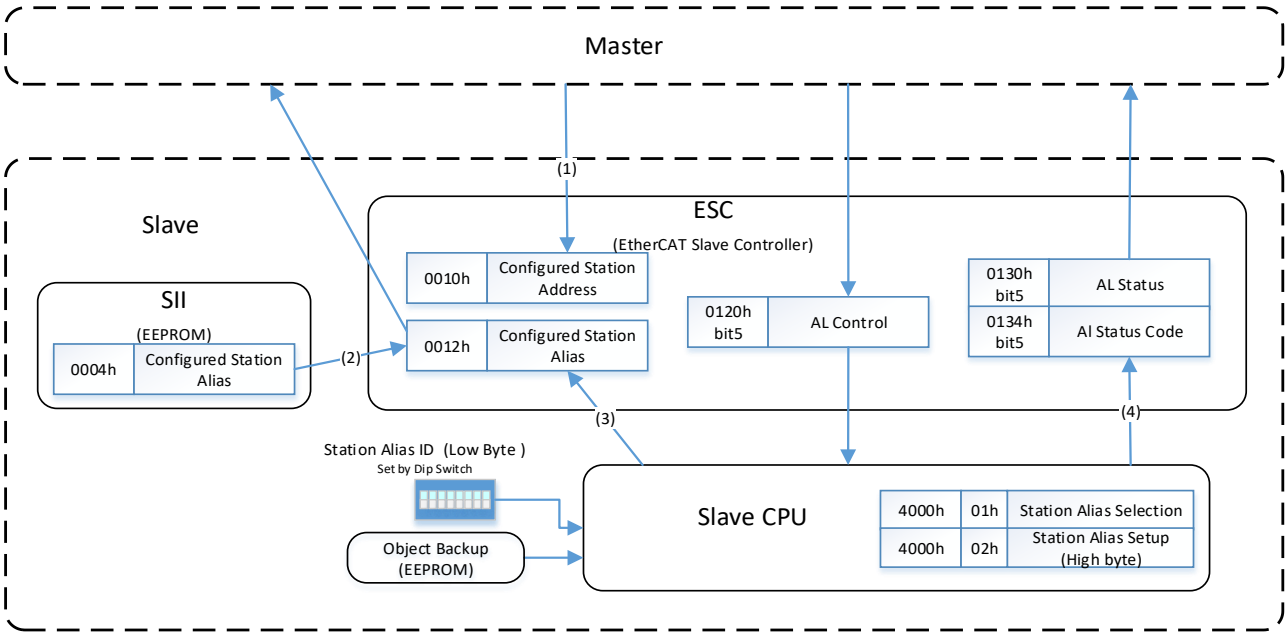
Position addressing should only be used during starting up of the EtherCAT system to scan the fieldbus and later only occasionally to detect newly attached slaves.

- Node Addressing (fixed addressing)

The configured station address is assigned by the master during start up and cannot be changed by the EtherCAT slave. The configured station alias address is stored in the ESI EEPROM. The configured station alias must be enabled by the master. The appropriate command action will be executed if node address matches with either configured station address or configured station alias.

The slave matched to the address set at station register (0x0010) from the master by position addressing is normally addressed in node addressing. This enables access without fail even when a device is added, the segment topology has changed and/or the slave has been removed.

The respective slave node address is set with the dip switch at the front of the device and CoE object dictionary index 4000. 0 - 65535 axes addresses can be set using the 8 dip switch (0x00 - 0xFF: bit7 - 0) at the front of the device and with a set value of bit 15 - 8, previously written in the non-volatile memory (index 4000:02) inside the device. When the alias selection (index 4000:01) is set to 1, the setting values will be written in the station alias setting register (0x0012) in an address space after the control power has been turned on. When the device address has changed under the control power on status, re-input the power to enable the change in axis address.



(1.) Set the position address by the master

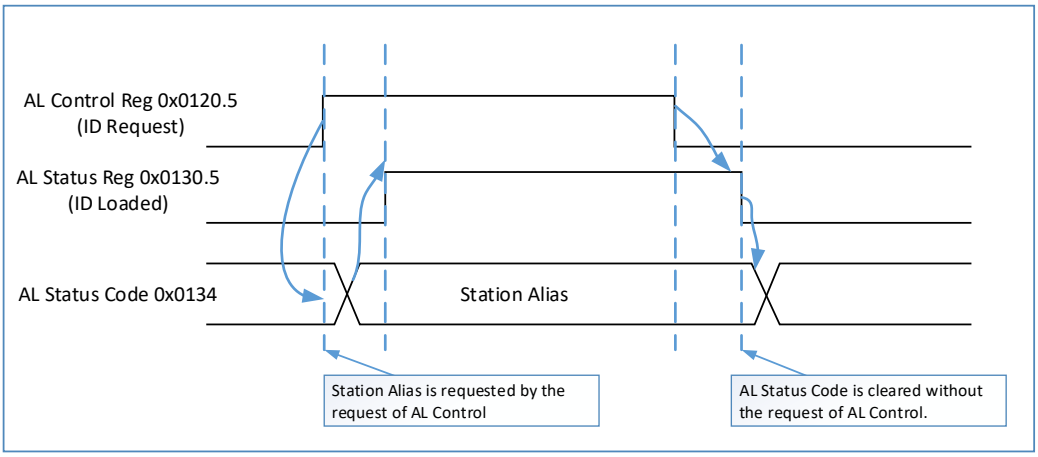
The slave matched to the address set at station register (0x0010) from the master by position addressing is normally addressed in node addressing.

(2.) Reading the value of SII from configured station alias (index 4000:01 = 0)

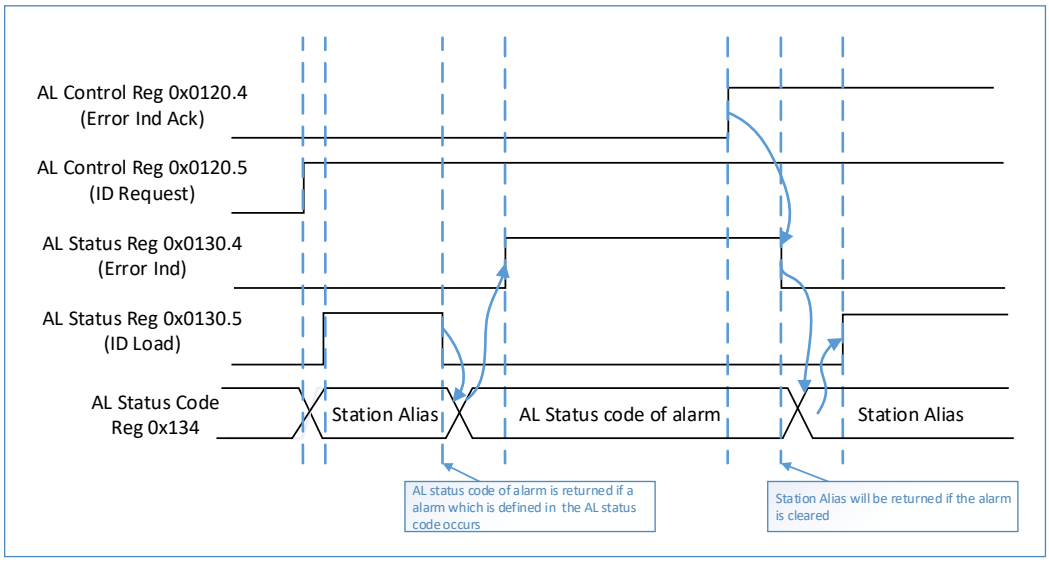
Setting the value of CoE object index 4000:01 to 0 and reading the value of 0004h (configured station alias) in the SII from 0012h (configured station alias) of ESC register. The device reads the value of object index 4000:01 (configured station selection) from backup EEPROM at the control power-on. If the value is 0, the value saved at 0004h (configured station alias) in the SII into 0012h (configured station alias) of ESC register and master reads this value.

(3.) Reading the value of dip switch from configured station alias (index 4000:01 = 1)

Setting the value of CoE object index 4000:01 to 1 and reading the value which is combined by object index 4000:02 (station alias setup (high byte)) and dip switch on the front of device from 0012h (configured station alias) of ESC register. The device reads the value of the object index 4000:01 (station alias selection) from backup EEPROM at the control power-on. If the value is 1, the value made of object index 4000:02 (station alias setup(high)) and dip switch on the front of device from 0012h (configured station alias) of ESC register. Master reads this value.



(1.) Bit5 (ID request) of AL control (0120h) is set to 1.
(2.) The station alias set up by dip switch (low byte) and index 4006:02 (high byte) returns to AL status code (0134h).
(3.) To put bit5 (ID loaded) of AL status (0130h) from 0 to 1.
(4.) When bit5 (ID request) of control register is set from 1 to 0, the bit5 (ID loaded) of AL status register (0x130) will change to 0.
(5.) AL status code (0134h) is clear.



In the period of returning station alias, if an alarm which is defined in the AL status code occurs, AL status code of the alarm is returned. When the alarm is cleared, station alias will return again.

5.9 Index 4007: FPGA Version

Index (hex)	Name	Description	Data Type	Flags	Default value	Store
4007:0	FPGA version	FPGA version	UINT8	RO	0x00	No

- END -