Start Guide

Panasonic A6B Series
EtherCAT AC Servo Motor & Driver
CiA402 CSP Mode

86Duino Coding IDE 501

EtherCAT Library

(Version 1.0)

# Revision

Date	Version	Description
2025/1/21	Version1.0	New Release.

Preface

In this guide, we will show you how to use the EtherCAT MDevice QEC-M-01 and the Panasonic A6B Series EtherCAT AC Servo Motor and Driver.

Notes QEC's PoE (Power over Ethernet)

In QEC product installations, users can easily distinguish between PoE and non-PoE: if the RJ45 house is red, it is PoE type, and if the RJ45 house is black, it is non-PoE type.

Non-PoE type	PoE type
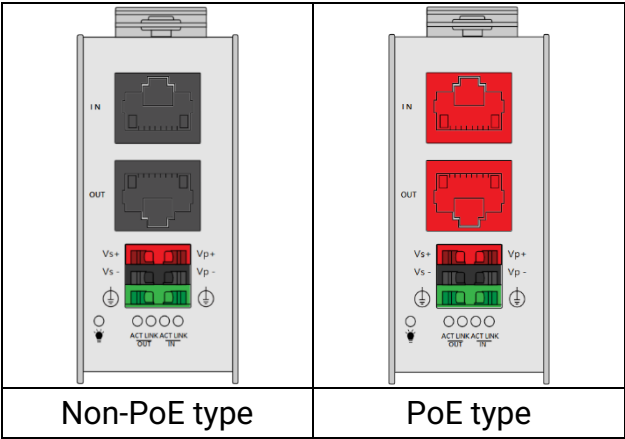
PoE (Power over Ethernet) is a function that delivers power over the network. QEC can be equipped with an optional PoE function to reduce cabling. In practice, PoE is selected based on system equipment, so please pay attention to the following points while evaluating and testing:

1. The PoE function of QEC is different and incompatible with EtherCAT P, and the PoE function of QEC is based on PoE Type B, and the pin functions are as follows:

PINS	10/100 DC on Spares (Mode B)
1	Rx+
2	Rx-
3	Tx+
4	DC+
5	DC+
6	Tx-
7	DC-
8	DC-

T568B Color
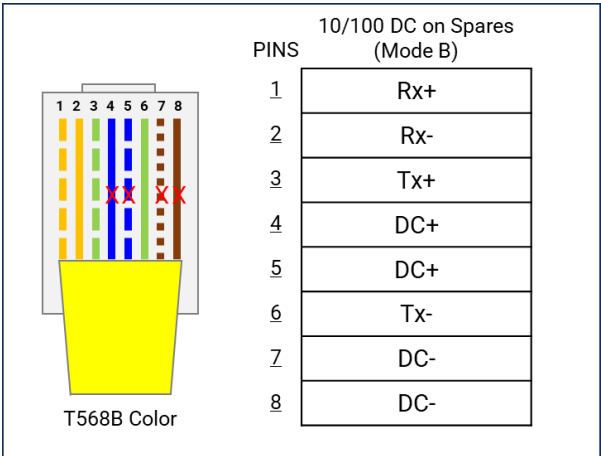
2. When connecting PoE and non-PoE devices, make sure to disconnect Ethernet cables at pins 4, 5, 7, and 8 (e.g., when a PoE-supported QEC EtherCAT MDevice connects with a third-party EtherCAT SubDevice).

3. QEC's PoE power supply is up to 24V/3A.

# 1. Connection and wiring hardware

The following devices are used here:

1. QEC-M-01（EtherCAT MDevice）
2. Panasonic A6B Series (EtherCAT AC Servo Motor and Driver)
3. 24V power supply & EU-type terminal cable & LAN cable
4. 200 to 220V power supply for Panasonic A6B Series

1.1   QEC-M-01

QEC EtherCAT MDevice.

1. Power Supply: Connect to Vs+/Vs- and Vp+/Vp- power supplies via EU terminals for 24V power.
2. EtherCAT Connection: Using the EtherCAT Out port (On the top side) connected to the EtherCAT In port of EtherCAT SubDevice via RJ45 cable.
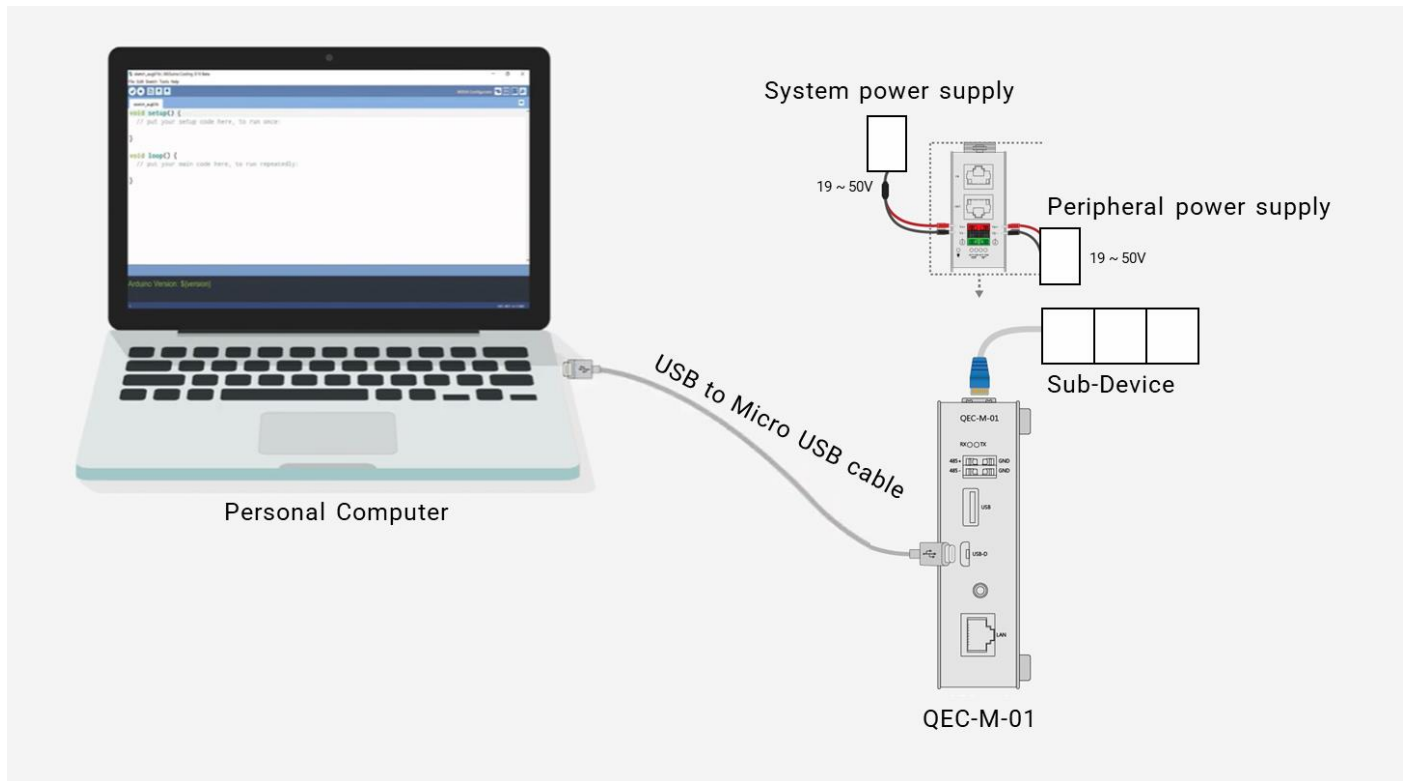
1.2   Panasonic A6B Series

Panasonic A6B Series is a high-performance integrated EtherCAT AC Servo Motor and Driver. This figure shows an example of when the driver wiring guide

* The photo is A6BF series. There are no X3 and X5 connectors in the A6BE series.
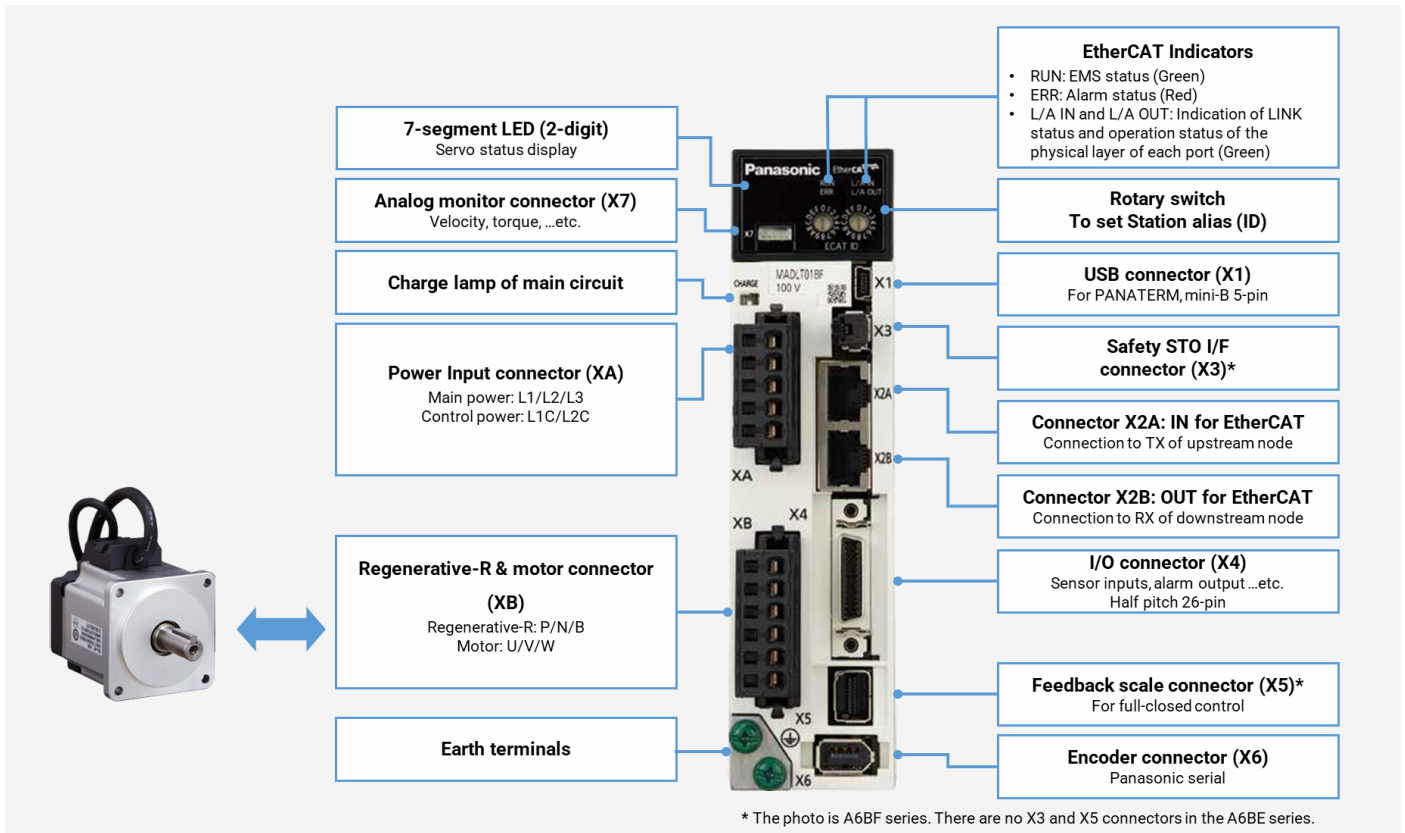
1. EtherCAT Connectivity
   - EtherCAT Input (X2A) and EtherCAT Output (X2B) ports for network communication.
   - Station ID rotary switch for device identification.
2. Signal Connections
   - I/O Signal Interface (X4): For digital inputs and outputs.
   - Encoder Connector (X6): For tracking motor position.
   - Regenerative Resistor & Motor Connector (XB): Provides motor power and control connections.
3. Power Supply
   - Main Power Input (XA): Supplies power to the motor.
   - Control Power Input (XA1/XA2): Provides power for the driver's control circuitry.
4. Safety and Status
   - Safety STO Interface (X3): Supports Safe Torque Off (STO) functionality for emergency stops.
   - LED Indicators: Displays driver status, EtherCAT communication status, and error information.
5. Additional Features
   - Rotary Switch: For setting the station alias (ID).
   - USB Interface (X1): Used for configuration and monitoring.

Wiring Diagram Overview

Below is an example of a typical wiring setup for the Panasonic A6B EtherCAT Servo Driver:

1. EtherCAT MDevice Connection:
   - Connect the EtherCAT output port of the QEC-M-01 to the X2A (EtherCAT IN) port of the servo driver.
   - If additional SubDevices are required, connect the X2B (EtherCAT OUT) port to the next SubDevice's EtherCAT input.
2. Motor Connection:
   - Connect the motor's U, V, W wires to the XB connector.
   - Connect the regenerative resistor to the P, N, B terminals.
3. Power Supply Wiring:
   - Connect the main power supply to the XA connector (L1, L2, L3).
   - Connect the control power supply to XA1 and XA2 terminals.
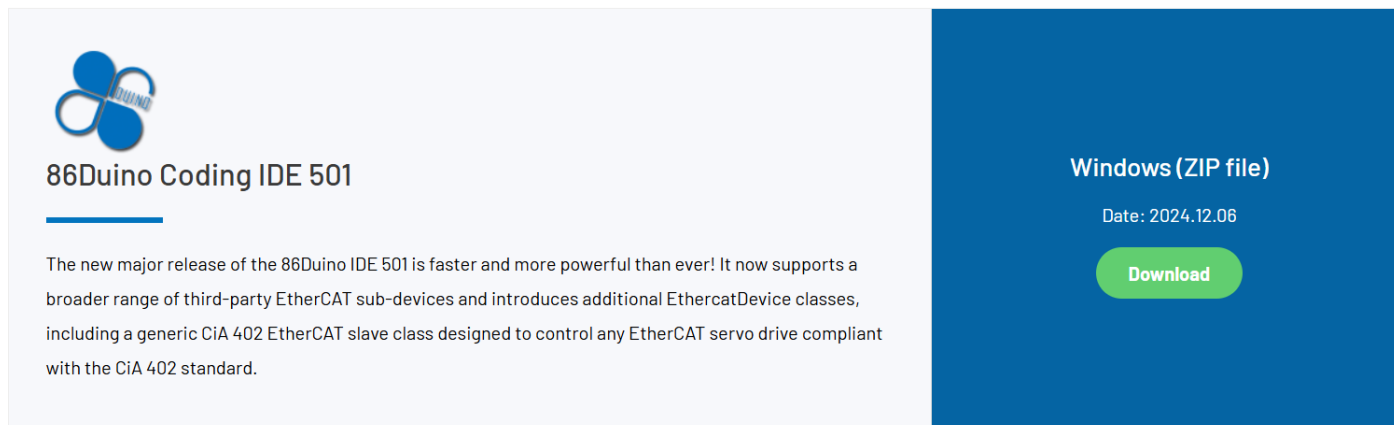4. Signal Wiring:
   - Connect input/output signals such as sensors and alarms to the X4 connector.
   - Connect encoder signal wires to the X6 connector.
5. Additional Settings:
   - Use the rotary switch to set the Station Alias (ID).
   - Configure the servo driver via the X1 USB interface.

2.   Software/Development Environment

Download 86duino IDE from https://www.qec.tw/software/.

After downloading, please unzip the downloaded zip file, no additional software installation is required, just double-click 86duino.exe to start the IDE.

Note:

If Windows displays a warning, click Details once and then click the Continue Run button once.

86Duino Coding IDE 501+ looks like below.

3.   Connect to PC and set up the environment

Follow the steps below to set up the environment:

1. Connect the QEC-M-01 to your PC via a Micro USB to USB cable (86Duino IDE installed).
2. Turn on the QEC power.
3. Open "Device Manager" (select in the menu after pressing Win+X) ->" Ports (COM & LPT)" in your PC and expand the ports; you should see that the "Prolific PL2303GC USB Serial COM Port (COMx)" is detected; if not, you will need to install the required drivers.
    (For Windows PL2303 driver, you can download here)
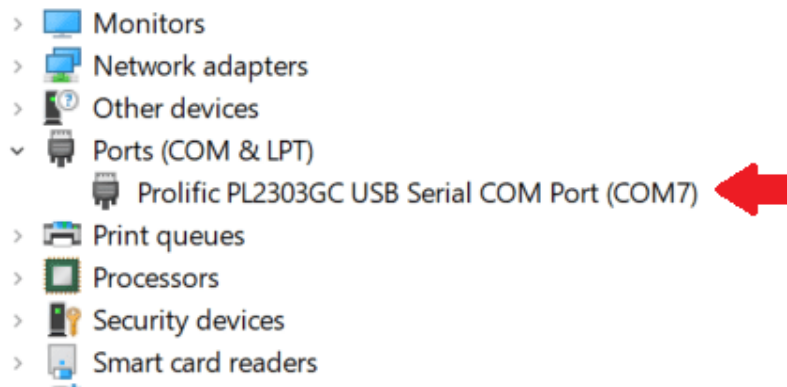
4. Open the 86Duino IDE.
5. Select the correct board: In the IDE's menu, select Tools> Board > QEC-M-01 (or the QEC MDevice model you use).
6. Select Port: In the IDE's menu, select Tools > Port and select the USB port to connect to the QEC MDevice (in this case, COM9 (QEC)).
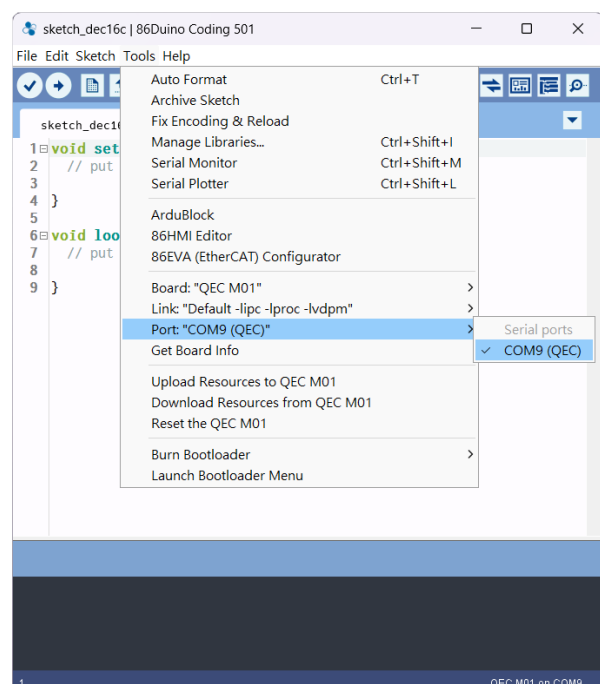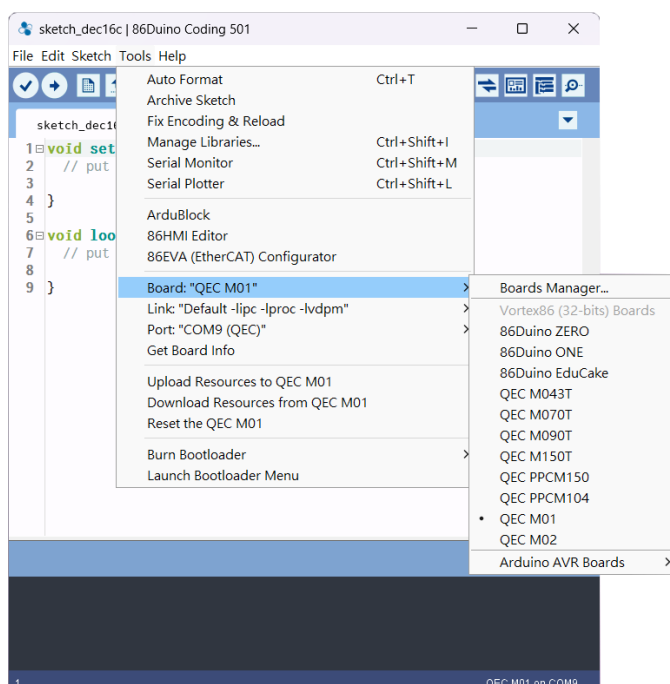
4. Write code

The EtherCAT MDevice (QEC-M-01) and the Panasonic A6B (EtherCAT AC Servo Driver) can be configured and programmed via the EtherCAT library in the 86Duino IDE.

The Arduino development environment has two main parts: setup() and loop(), which correspond to initialization and main programs. Before operating the EtherCAT network, you must configure it once. The process should be from Pre-OP to OP mode in EtherCAT devices.

The following program sets the Panasonic A6B Driver into CiA402 Cyclic Synchronous Position (CSP) mode:

- EtherCAT Cycle Time: 1 millisecond.
- EtherCAT Mode: ECAT_SYNC.
- EtherCAT Distributed Clock (DC) Mode: Enabled.

The EthercatMaster object ("master") represents the QEC-M-01, while the EthercatDevice_CiA402 object ("motor") represents the Panasonic A6B Driver.

A. In Setup Function:

In the setup() function initializes communication and configures the motor for CiA402 CSP mode. Follow the steps below:

1. Initialize Serial Communication
   - Start serial communication at a baud rate of 115200.
2. Start the EtherCAT MDevice
   - Begin the EtherCAT MDevice and set the EtherCAT state machine to the PRE-OPERATIONAL state.
3. Set Cyclic Synchronous Position (CSP) Mode
   - Configure the motor to CSP mode using setCiA402Mode(CIA402_CSP_MODE).
4. Set Distributed Clock (DC) Mode
   - Use the setDc() function to set the A6 Servo Driver to DC mode and synchronize the cycle time with EtherCAT communication.
5. Register a Cyclic Callback function
   - Use the attachCyclicCallback() function to register a callback function named "myCallback".
6. Start the EtherCAT MDevice
   - Use the start() function to switch the EtherCAT state machine to the OPERATIONAL state. Set the cycle time to 1ms and ECAT_SYNC mode.
7. Set the current position
   - Use the setTargetPosition() and getPositionActualValue() functions to set the first target position to the current position in the begin of enable for safety.

8. Enable the Motor
    • Use the enable() function to enable the motor and transition it to
      CIA402_OPERATION_ENABLED.

B. In Callback Function:
Use the setTargetPosition() function to set the target position as the current position to add
1000.

C. In Loop Function:
The loop() function, it prints the motors' current position to the serial monitor.

The example code is as follows:

```cpp
#include "Ethercat.h"


EthercatMaster master;
EthercatDevice_CiA402 motor;


int32_t position = 0;


void myCallback() {
  if (motor.getCiA402State() != CIA402_OPERATION_ENABLED)
    return;
  motor.setTargetPosition(position += 1000);
}



void setup() {
  Serial.begin(115200);

  Serial.print("Begin: "); Serial.println(master.begin());
  Serial.print("Slave: "); Serial.println(motor.attach(0, master));
  motor.setCiA402Mode(CIA402_CSP_MODE);
  motor.setDc(1000000);

  master.attachCyclicCallback(myCallback);
  Serial.print("Start: "); Serial.println(master.start(1000000, ECAT_SYNC));

  motor.setTargetPosition(position = motor.getPositionActualValue());
  Serial.print("Enable: "); Serial.println(motor.enable());
}


void loop() {
```
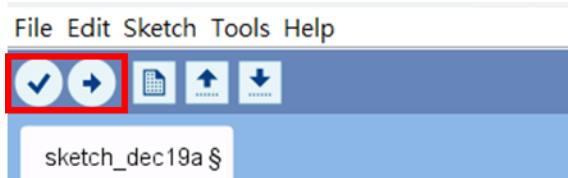
```
  Serial.print("Pos: "); Serial.println(motor.getPositionActualValue());
  // ...
}
```

Note:

Once the code is written, click on the toolbar to ☑ compile, and to confirm that the compilation is complete and error-free, you can click ➡ to upload.
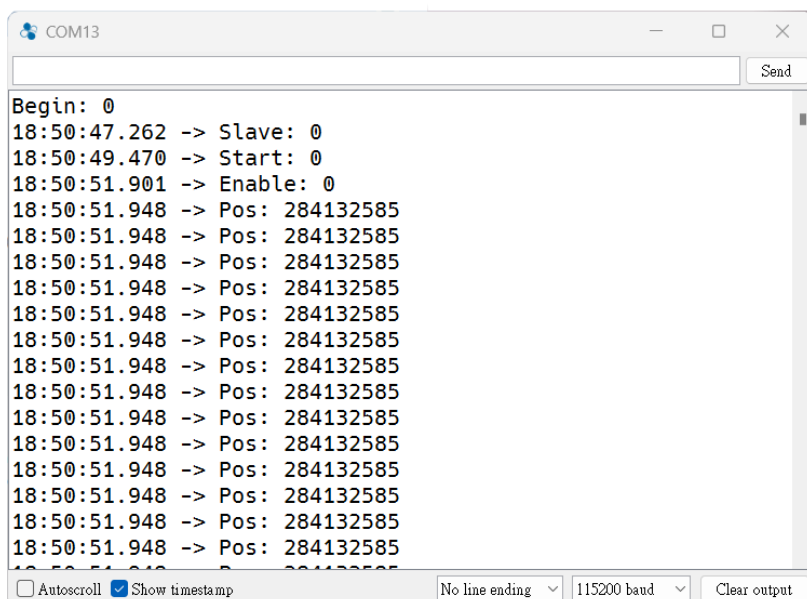
After you successfully upload the program to the QEC-M-01, you can open the Serial Monitor on 86Duino IDE. Please check the Serial baud rate is same as your setting.
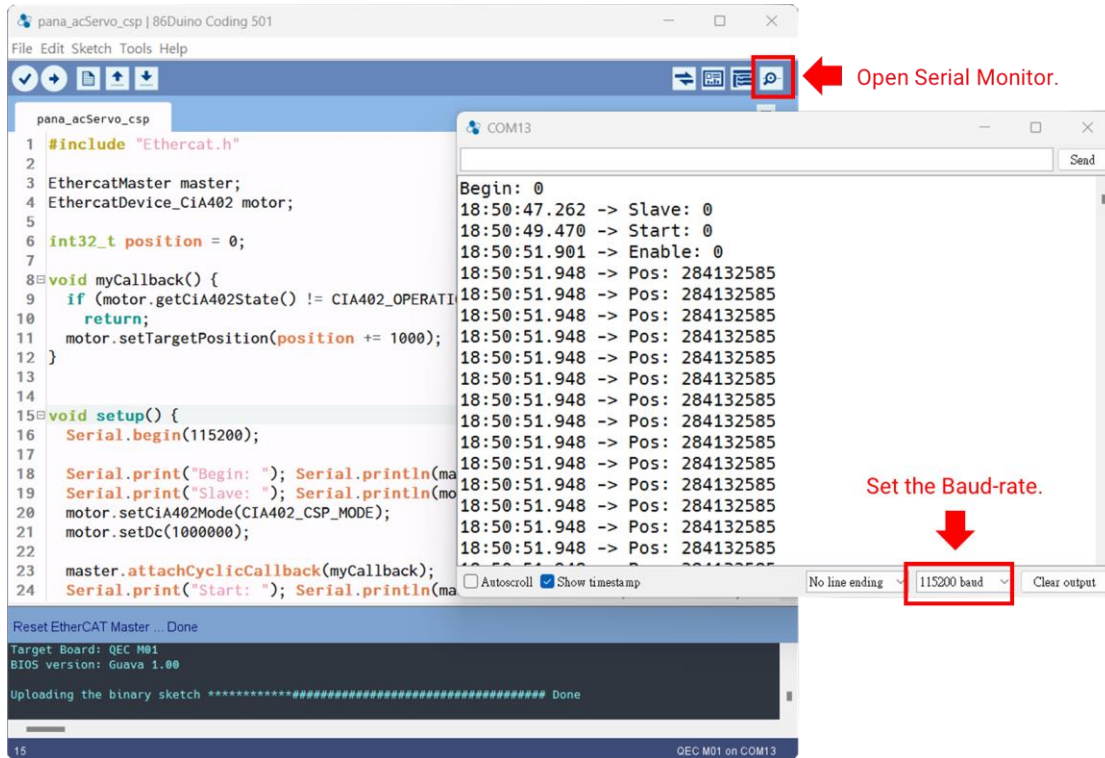
If the EtherCAT communication config successful, Serial Monitor will print "0" and "Enable: 0".
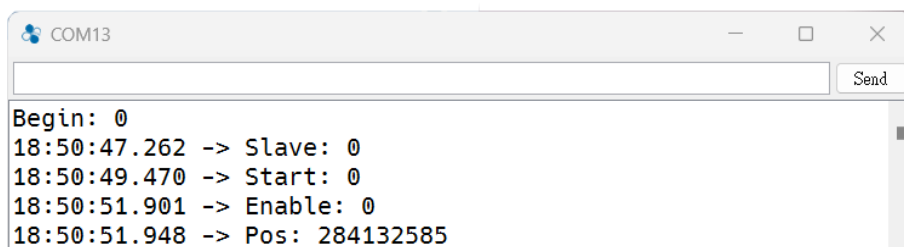
It will print the motor's current position to the serial monitor.

Troubleshooting

QEC-M-01 cannot successfully upload code

When you are unable to successfully upload code, please open 86EVA to check if your QEC EtherCAT MDevice's environment is abnormal. As shown in the figure below, please try updating your QEC EtherCAT MDevice's environment, which will include the following three items: Bootloader, EtherCAT firmware, and EtherCAT tool.

Now, we will further explain how to proceed with the update:

Step 1: Setting up QEC-M

1. Download and install 86Duino IDE 500+ (or a newer version): You can download it from [Software](#).
2. Connect the QEC-M: Use a USB cable to connect the QEC-M to your computer.
3. Open 86Duino IDE: After the installation is complete, open the 86Duino IDE software.
4. Select Board: From the IDE menu, choose "Tools" > "Board" > "QEC-M-01" (or the specific model of QEC-M you are using).
5. Select Port: From the IDE menu, choose "Tools" > "Port" and select the USB port to which the QEC-M is connected.
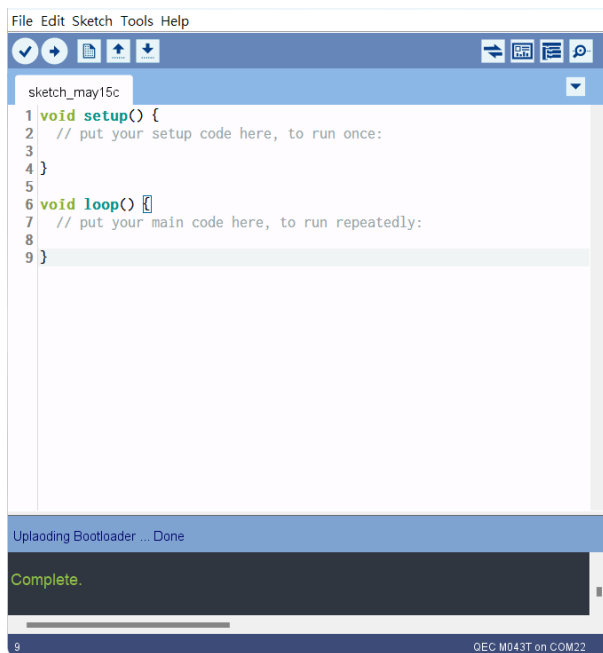
Step 2: Click "Burn Bootloader" button

After connecting to your QEC-M product, go to "Tools"> "Burn Bootloader". The currently selected QEC-M name will appear. Clicking on it will start the update process, which will take approximately 5-20 minutes.

QEC-M-01:
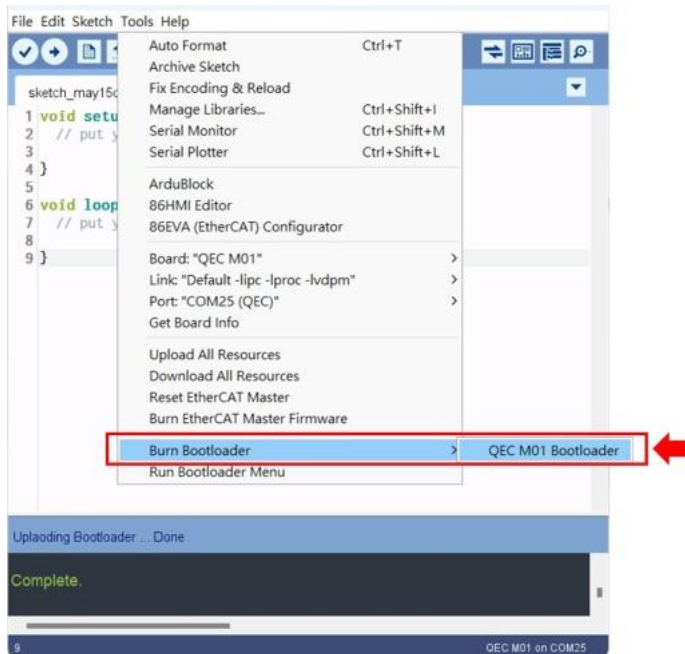
Step 3: Complete the Update

After completing the above steps, your QEC-M has been successfully updated to the latest version of the development environment.

Warranty

This product is warranted to be in good working order for a period of one year from the date of purchase. Should this product fail to be in good working order at any time during this period, we will, at our option, replace or repair it at no additional charge except as set forth in the following terms. This warranty does not apply to products damaged by misuse, modifications, accident or disaster. Vendor assumes no liability for any damages, lost profits, lost savings or any other incidental or consequential damage resulting from the use, misuse of, originality to use this product. Vendor will not be liable for any claim made by any other related party. Return authorization must be obtained from the vendor before returned merchandise will be accepted. Authorization can be obtained by calling or faxing the vendor and requesting a Return Merchandise Authorization (RMA) number. Returned goods should always be accompanied by a clear problem description.