Start Guide

NPM PULSERVO II EtherCAT Driver CiA402 CSP Mode (1-axis)

86Duino Coding IDE 501

EtherCAT Library

(Version 1.0)

Revision

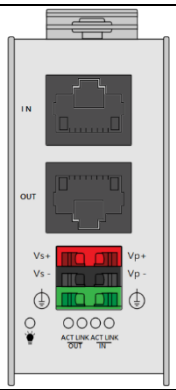
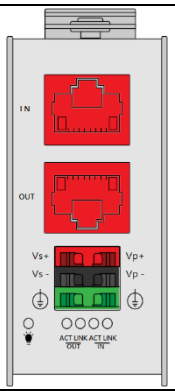
Date	Version	Description
2025/08/29	Version1.0	New Release.

Preface

In this guide, we will show you how to use the EtherCAT MDevice QEC-M-01 and the NPM Pulservo II (Closed-loop stepping motor Driver).

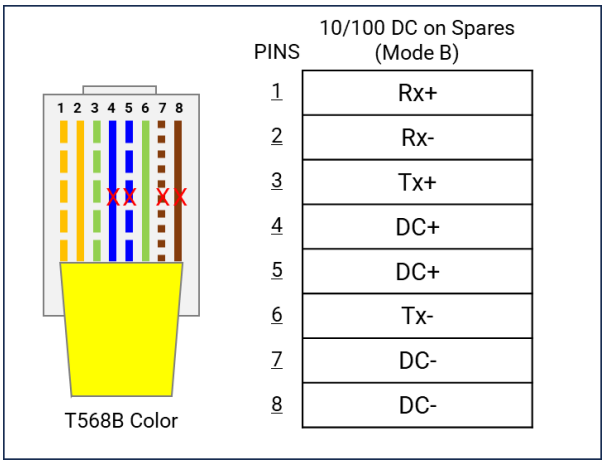
Notes QEC's PoE (Power over Ethernet)

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-

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

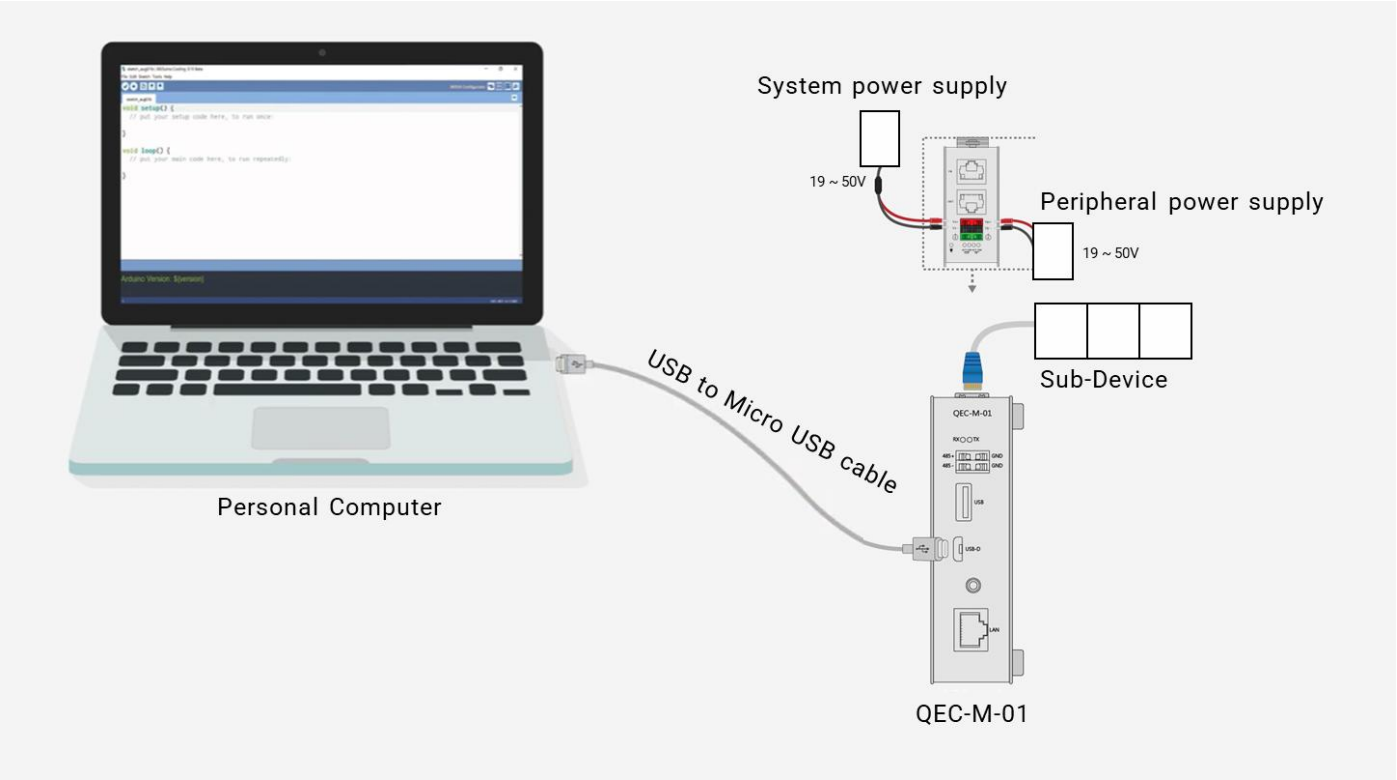
The following devices are used here:

1. QEC-M-01 (EtherCAT MDevice)
2. NPM Pulservo II (Closed-loop stepping motor Driver)
3. 24V power supply & EU-type terminal cable & LAN cable
4. PSM2-28 (Standard type Motor, frame size 28 mm square)

1.1 QEC-M-01

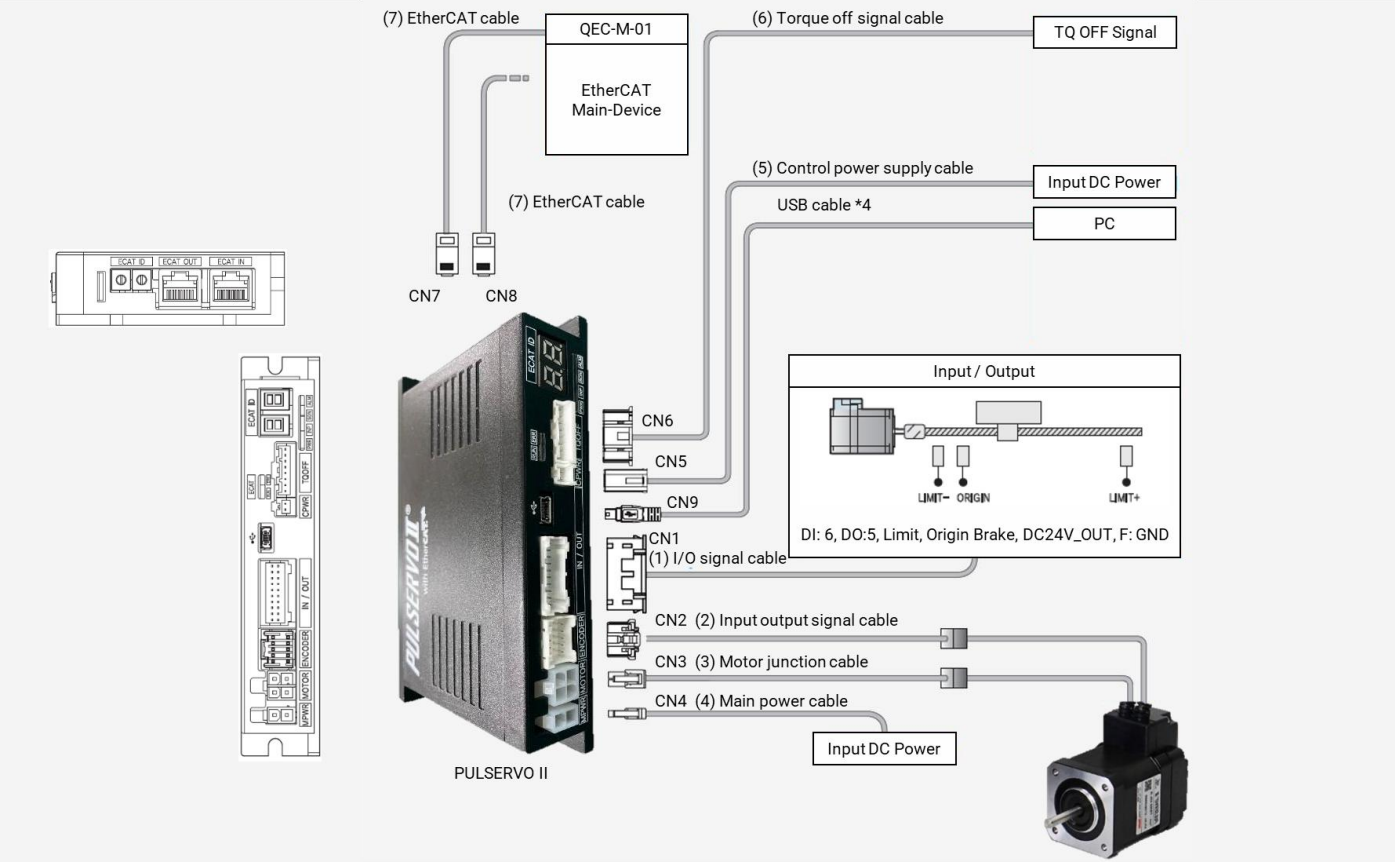
QEC EtherCAT MDevice.

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.



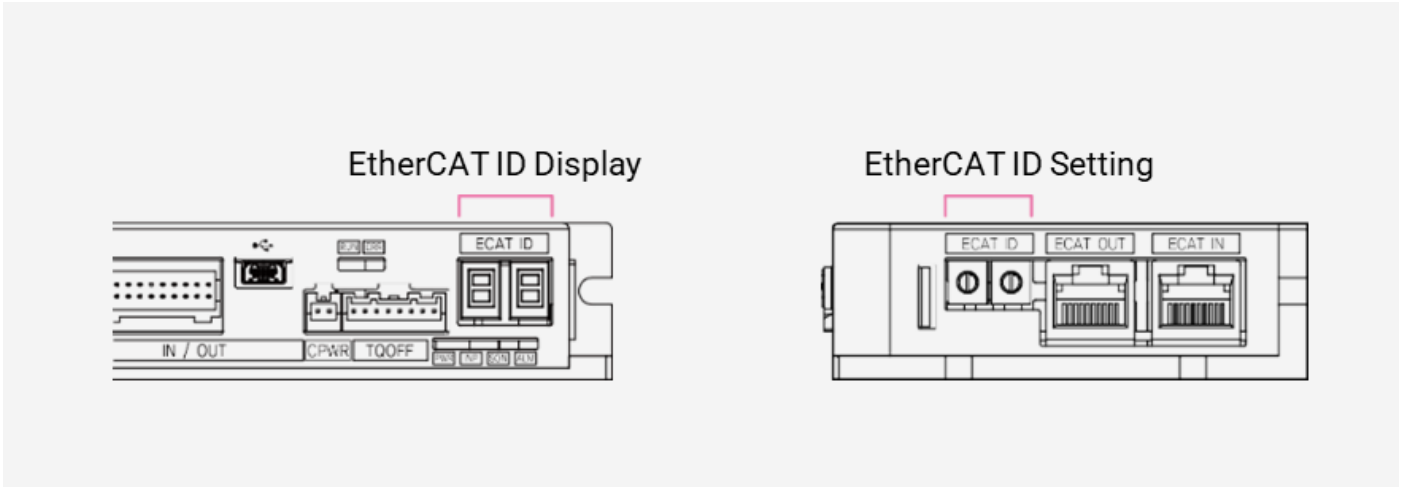
1.2 NPM Pulservo II

NPM Pulservo II, a PLS2-EC series EtherCAT interface closed-loop stepping motor driver (NPM Step-Servo Driver). This figure shows an example of when the **PSM2-28** motor is connected.



1. EtherCAT Connectivity:
   - Two EtherCAT ports (CN7, CN8) for network communication.
   - EtherCAT ID setting switches (SW1, SW2) for device identification.
2. Signal Connections:
   - Input/output signal interface (CN1) for digital inputs and outputs.
   - Encoder connection (CN2) to track motor position.
   - Motor connection (CN3) for power and control.
3. Power Supply:
   - Main power input (CN4) for operating the motor.
   - Control power input (CN5) for driver control circuitry.
4. Safety and Status:
   - Torque-Off signal input (CN6) for emergency stop functionality.
   - LED indicators for driver status, EtherCAT status, and EtherCAT ID display.
5. USB Port:
   - USB interface (CN9) for configuration and monitoring.

ID setting for Pulservo II Driver:



Change the EtherCAT ID (Configured Alias ID) value by configuring the rotary switch. The switch on the left shows 10 digits and the switch on the right shows 1 digit.
The setting range is 0 to 99.

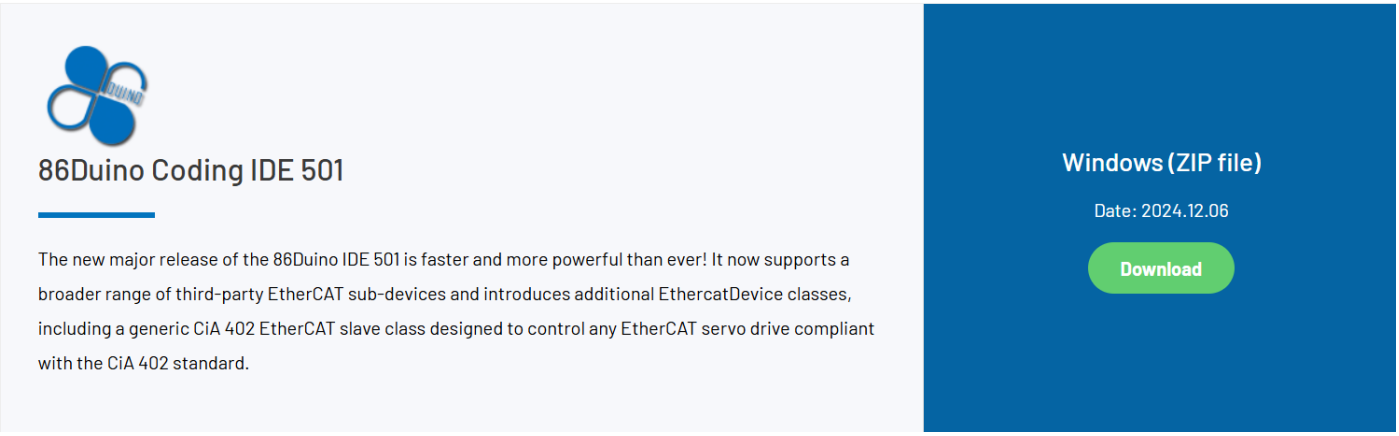
Note: The ID Value (Configured Alias ID) set on the rotary switch is applied when the product is turned ON.

ID Settings:

- The 7-segment LED indicates the Physical Address or EtherCAT ID (EtherCAT Configured Alias) value. The display value conditions are as follows:
- When all rotary switches are set to "0", the 7-segment LED indicates the EtherCAT Physical Address. Since there is no connection between the SubDeivce (this product) and the MDevice, it indicates 0 (zero) until the Physical Address is assigned. When the MDevice assigns a physical address to each SubDeivce (this product), its value is displayed.
- If the rotary switch is set to any other value other than "0", the 7-segment LED indicates the corresponding set value (EtherCAT Configured Alias).
- If the 7-segment LED of the ID is blinking, it indicates that the ID value is not set. It is set when the power is turned ON.

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.

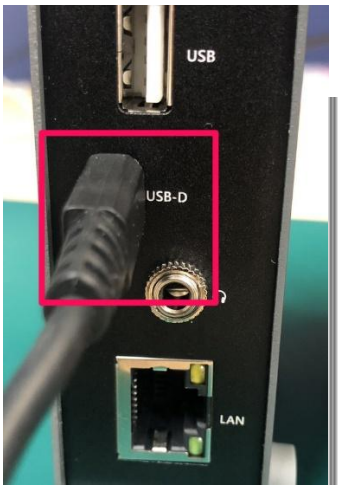
*__Note__: If Windows displays a warning, click Details once and then click the Continue Run button once.

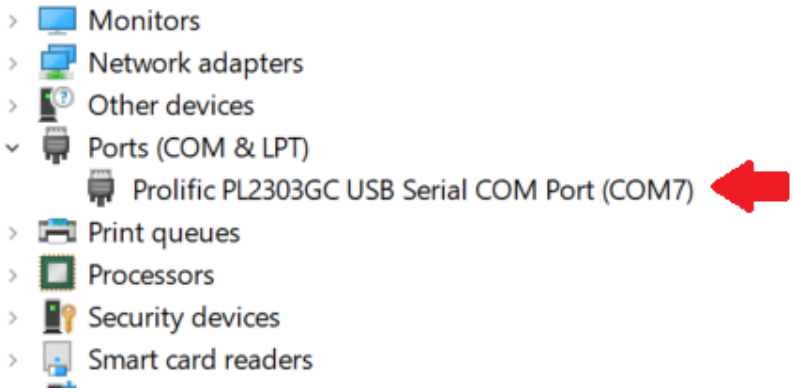
86Duino Coding IDE 501+ looks like below.

3. Connect to PC and set up the environment

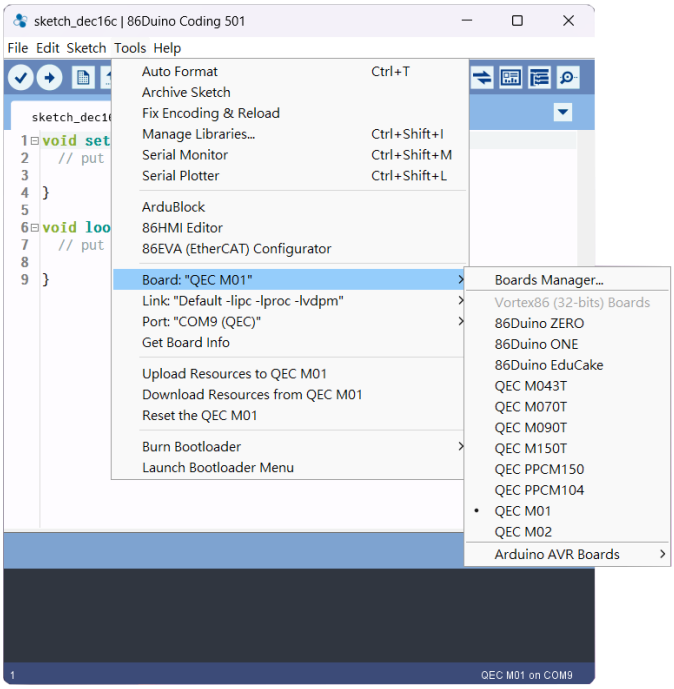
Follow the steps below to set up the environment:

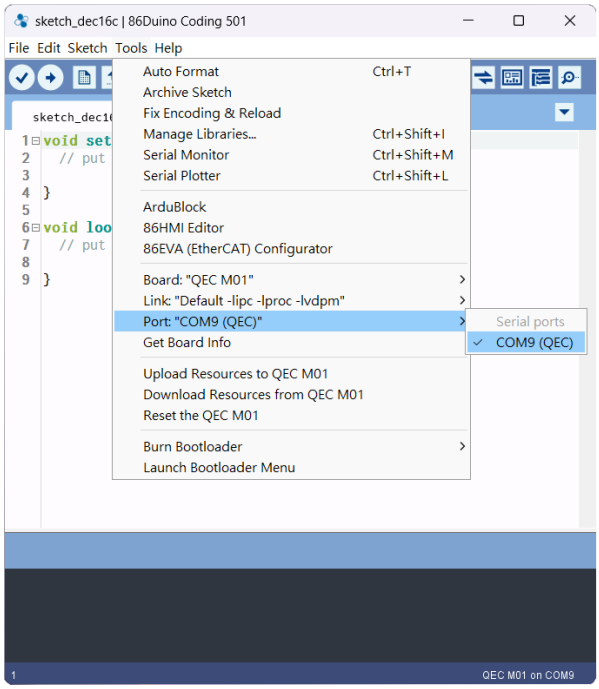
1. Connect the QEC-M-01 to your PC via a Micro USB to USB cable (86Duino IDE installed).
2. Turn on the QEC power.
3. Open “Device Manager” (select in the menu after pressing Win+X) ->” Ports (COM & LPT)” in your PC and expand the ports; you should see that the “Prolific PL2303GC USB Serial COM Port (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 NPM Pulservo II (Closed-loop stepping motor 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 NPM Pulservo II into CiA402 Cyclic-Synchronous Position (CSP) mode:

- EtherCAT Cycle Time: 1 millisecond.
- EtherCAT Mode: ECAT_SYNC.
- Motion is generated inside a 1 ms **cyclic callback** by stepping a command position at a fixed rate (constant velocity) and flipping direction at the travel limits.

Objects:

- `EthercatMaster master` represents QEC-M-01.
- `EthercatDevice_CiA402 motor` represents the NPM Pulservo II driver.

A. In Setup Function:

The `setup()` function initializes communication and configures the motor for CiA402 **CSP** mode.
Steps:

1. Initialize Serial Communication
   Start serial at 115200 baud for status printing.
2. Begin the EtherCAT MDevice & attach the CiA402 slave
   Call `master.begin()` and `motor.attach(0, master)` to bind the driver.
3. Configure Distributed Clocks (cycle = 1 ms)
   `motor.setDc(1000000);` to match a 1 ms control period.
4. Set CiA402 Mode to CSP
   `motor.setCiA402Mode(CIA402_CSP_MODE);`
5. Register the cyclic callback & start OP in ECAT_SYNC
   `master.attachCyclicCallback(MyCyclicCallback);` then
   `master.start(1000000, ECAT_SYNC);` to switch the network to OPERATIONAL at 1 ms.
6. Initialize command position & enable the motor
   Set the first CSP target to the actual position for safety
   `motor.setTargetPosition(position = motor.getPositionActualValue());`
   and call `motor.enable()` to reach CIA402_OPERATION_ENABLED.

B. In Loop Function:

The `loop()` prints the actual position for monitoring. All motion logic runs in the cyclic callback.

C. In Callback Function

This function emits a new CSP target every 1 ms to achieve constant-velocity ping-pong motion between −100,000 and +100,000 counts.

1. State Guard
   Exit early unless `motor.getCiA402State() == CIA402_OPERATION_ENABLED`.
2. Direction Memory
   static int dir = +1; holds direction across calls (+1 forward, −1 backward).
3. Travel Limits
   Upper bound: +100,000 counts
   Lower bound: −100,000 counts
   Flip direction at the bounds:
   - If position >= 100000 ⇒ dir = −1
   - If position <= −100000 ⇒ dir = +1
4. Constant-Velocity Step
   Per tick: position += dir * 100;
   With 1 ms period, speed ≈ 100 × 1000 = 100,000 counts/s.
   Command it with `motor.setTargetPosition(position);`.
5. Code logic summary
   CSP targets are produced every 1 ms in the callback (synchronous with DC).
   Travel range and speed are trivially tunable by changing the limits (±100,000) and the step size (counts per 1 ms).
   Behavior mimics a simple PP back-and-forth, but is generated continuously in CSP.

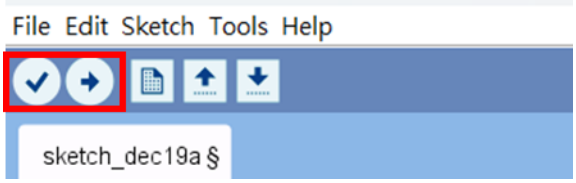
The example code is as follows:

```
#include "Ethercat.h"

EthercatMaster master;
EthercatDevice_CiA402 motor;

int32_t position = 0;

void MyCyclicCallback()
{
  if (motor.getCiA402State() != CIA402_OPERATION_ENABLED) return;

  // Set the target position under csp.
  static int dir = +1;
  if (position >= 100000)  dir = -1;
  else if (position <= -100000) dir = +1;
```

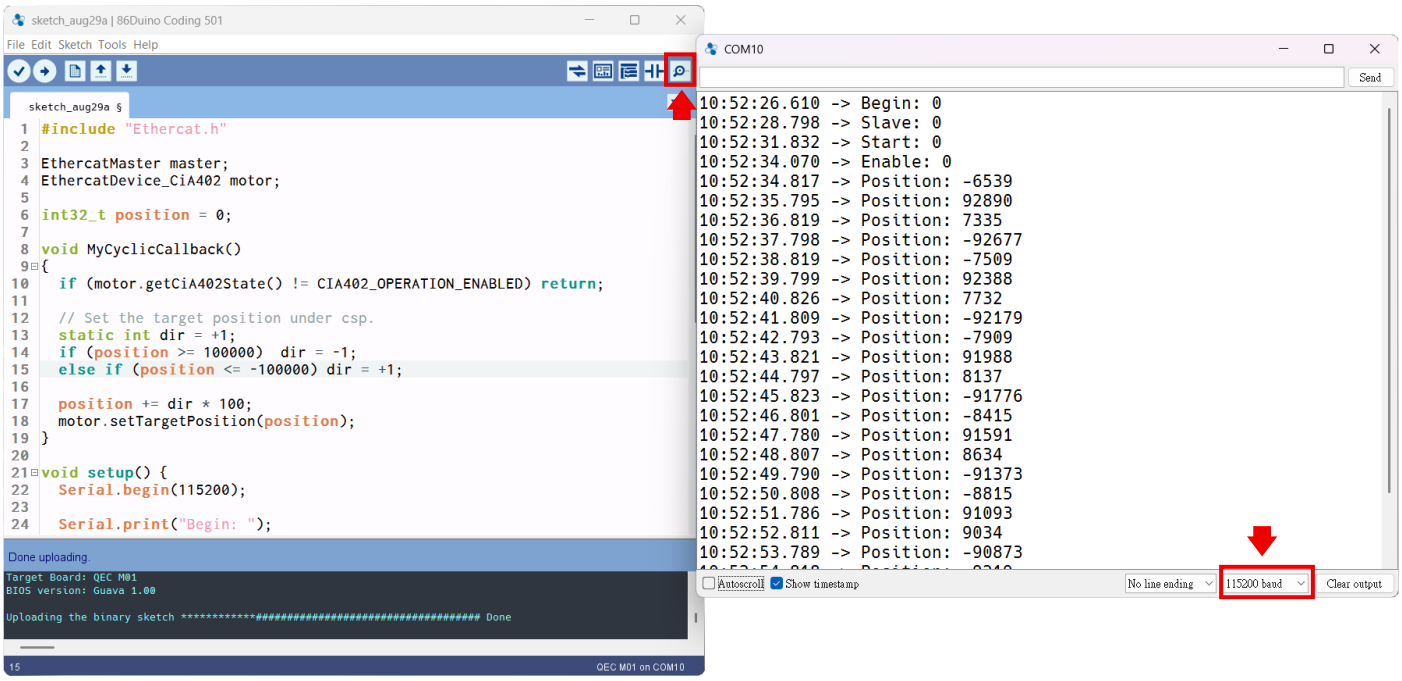
```
  position += dir * 100;
  motor.setTargetPosition(position);
}

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

  Serial.print("Begin: ");
  Serial.println(master.begin());

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

  // Register a callback function.
  master.attachCyclicCallback(MyCyclicCallback);
  Serial.print("Start: ");
  Serial.println(master.start(1000000, ECAT_SYNC));

  // Set target position to current position for safe.
  motor.setTargetPosition(position = motor.getPositionActualValue());
  Serial.print("Enable: ");
  Serial.println(motor.enable());
}

void loop() {
  // Print Position.
  Serial.print("Position: ");
  Serial.println(motor.getPositionActualValue());
  delay(1000);
}
```

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.

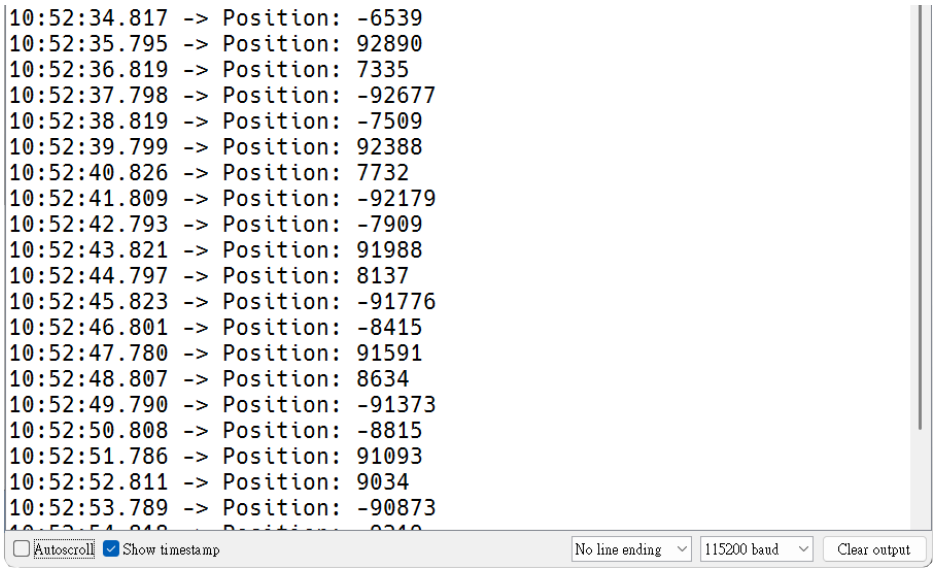


After you successfully upload the program to the QEC-M-01, you can open the Serial Monitor on the 86Duino IDE. Please check that the Serial baud rate is the same as your setting.



If the EtherCAT communication configuration is successful, the Serial Monitor will print each status in “0” and “Enable: 0”.

It will print the motor’s current position to the serial monitor every second.



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:

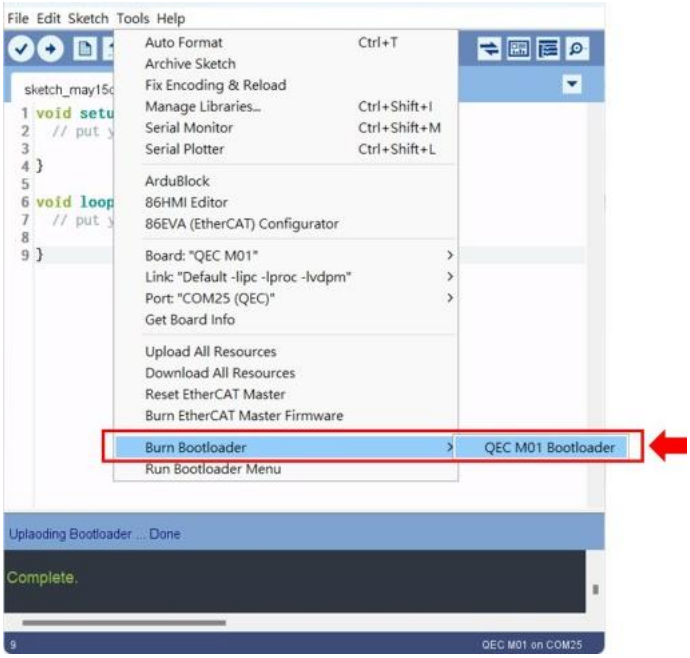
Step 1: Setting up QEC-M

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.

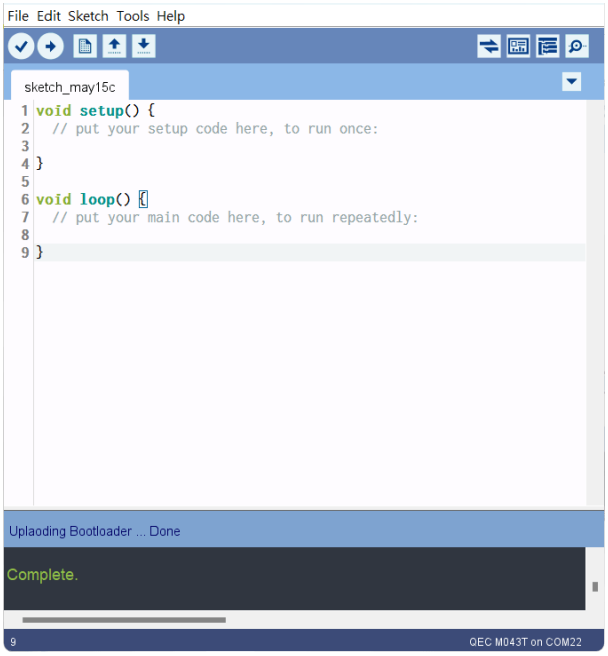
Step 2: Click “Burn Bootloader” button

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.