# Start Guide

## MP3S: CiA402 PP Mode (1-axis)

86Duino Coding IDE 501

EtherCAT Library

(Version 2)

# Revision

| Date | Version | Description |
|------|---------|-------------|
| **2024/2/2** | Version1.0 | New Release. |
| **2024/12/16** | Version1.1 | Split the development steps into two documents. |
| **2024/12/23** | Version2.0 | Using 86Duino IDE 501+ version with the released cia402 library. |

# Preface

In this guide, we will show you how to use the EtherCAT Main-device QEC-M-01P and the QEC-RXXMP3S Series (EtherCAT 3-axis Stepper Motor Controller).
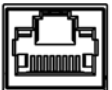
## <span style="color:red">Notes</span> QEC's PoE (Power over Ethernet)

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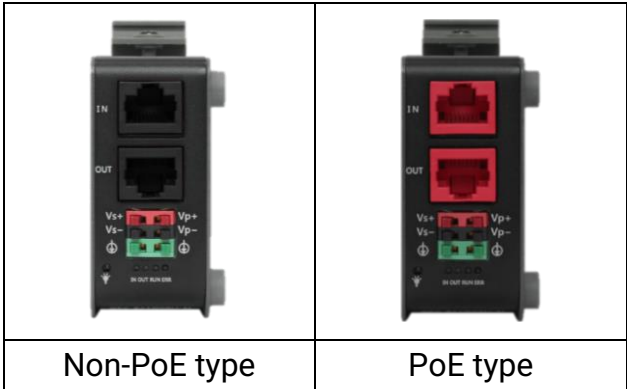
PoE (Power over Ethernet) is a function that delivers power over the network. QEC can be equipped with an optional PoE function to reduce cabling. In practice, PoE is selected based on system equipment, so please pay attention to the following points while evaluating and testing:

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:

| Pin # | Signal Name | Pin # | Signal Name |
|-------|-------------|-------|-------------|
| 1 | LAN1_TX+ | 2 | LAN1_TX- |
| 3 | LAN1_RX+ | 4 | VS+ |
| 5 | VP+ | 6 | LAN1_RX- |
| 7 | VS-(GND) | 8 | VP-(GND) |

   <span style="color:red">*</span> PoE LAN with the Red Housing; Regular LAN with Black Housing.

   <span style="color:red">*</span> L4, L5, L7, L8 pins are option, for RJ45 Power IN/OUT.

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 Main-device connects with a third-party EtherCAT slave).
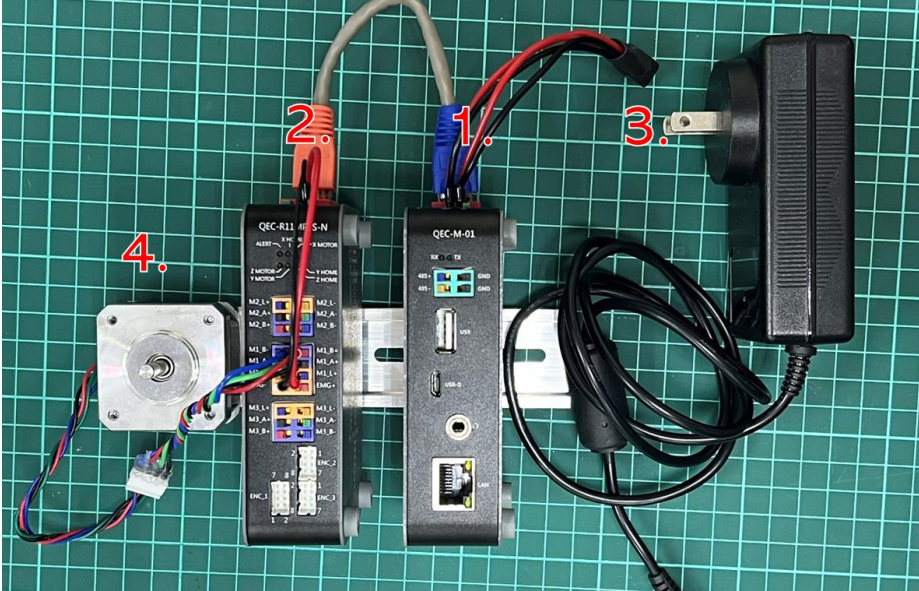


3.  QEC's PoE power supply is up to 24V/3A.

# 1. Connection and wiring hardware

The following devices are used here:

1. QEC-M-01P（EtherCAT Main-device/PoE）
2. QEC-R11MP3S (EtherCAT Slave, 3-axis Stepper Motor Controller)
3. 24V power supply & EU-type terminal cable & LAN cable
4. A 4-wire two-phase bipolar 42 stepper motor (refer to 86STEP | 86Duino)
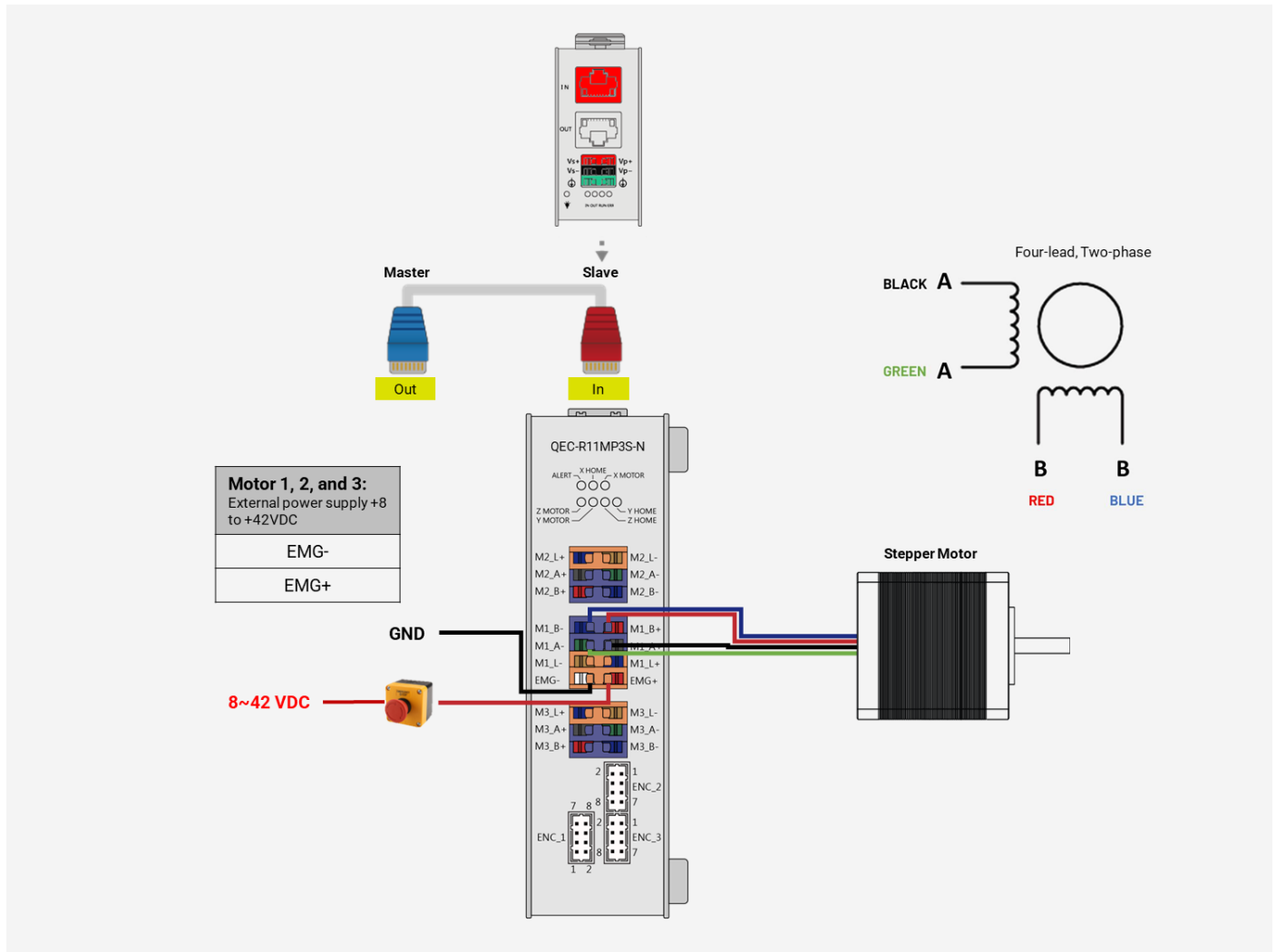


## 1.1 QEC-M-01P

QEC EtherCAT Main-device with PoE function.

1. Using the EtherCAT Out port (top side) connected to the EtherCAT In port of QEC-R11MP3S via RJ45 cable (powered by PoE).
2. Connect to Vs+/Vs- and Vp+/Vp- power supplies via EU terminals for 24V power.

## 1.2 QEC-R11MP3S-N

The QEC-R11MP3S-N with PoE function.



1.    Connecting an 8–42V power supply to EMG+/- is required.
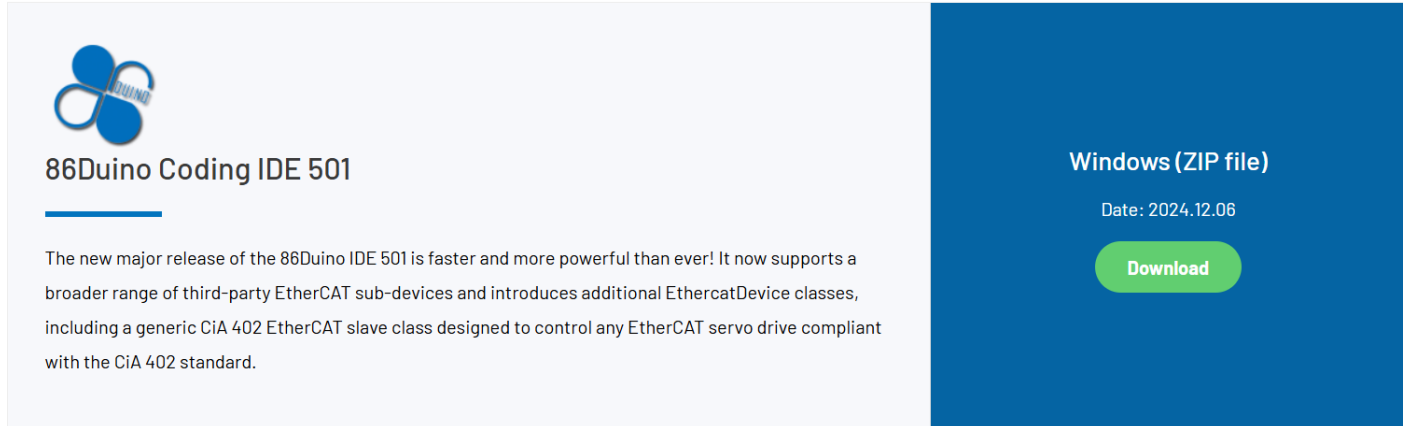
**Note:** The Emergency Stop is a safety mechanism for quickly shutting down machinery in emergencies via a hardware switch.

2.    Connect the 4-wire bidirectional 42-stepper motor to the M1 axis (middle) of the QEC-R11MP3S-N as follows:
   •    A+ (black) connects to M1_A+; A- (green) connects to M1_A-.
   •    B+ (red) connects to M1_B+; B- (blue) connects to M1_B-.

**Note:** The wiring colors of the motor are the same as the terminal colors of the QEC-R11MP3S-N, users can follow the color for wiring reference.

# 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-01P 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])
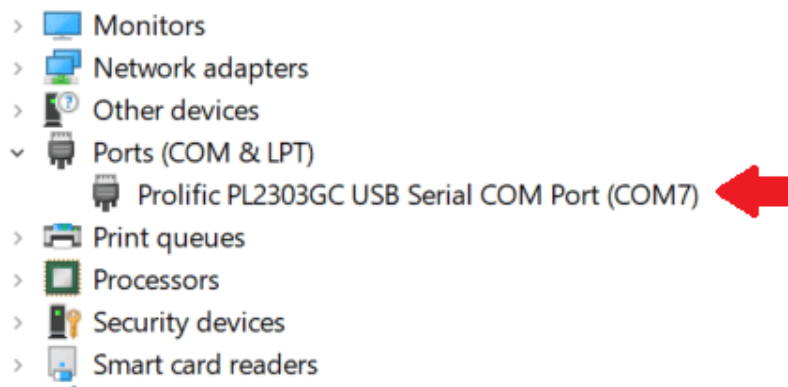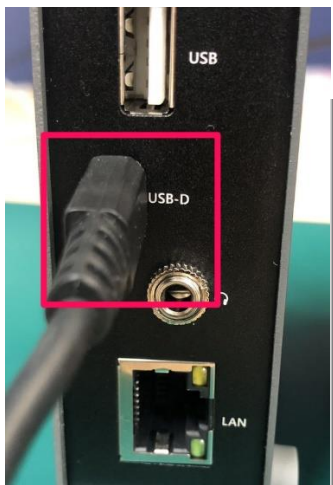


4. Open the 86Duino IDE.
5. Select the correct board: In the IDE's menu, select Tools> Board > QEC-M-01 (or the QEC Main-device model you use).
6. Select Port: In the IDE's menu, select Tools > Port and select the USB port to connect to the QEC Main-device (in this case, COM9 (QEC)).
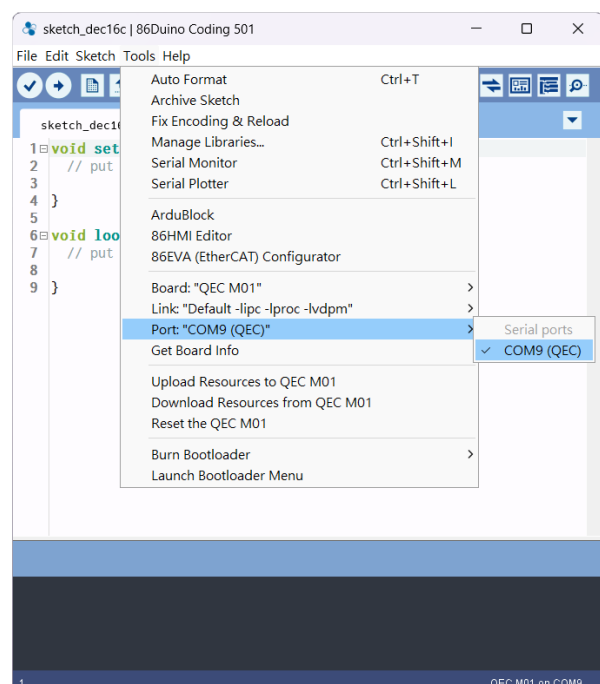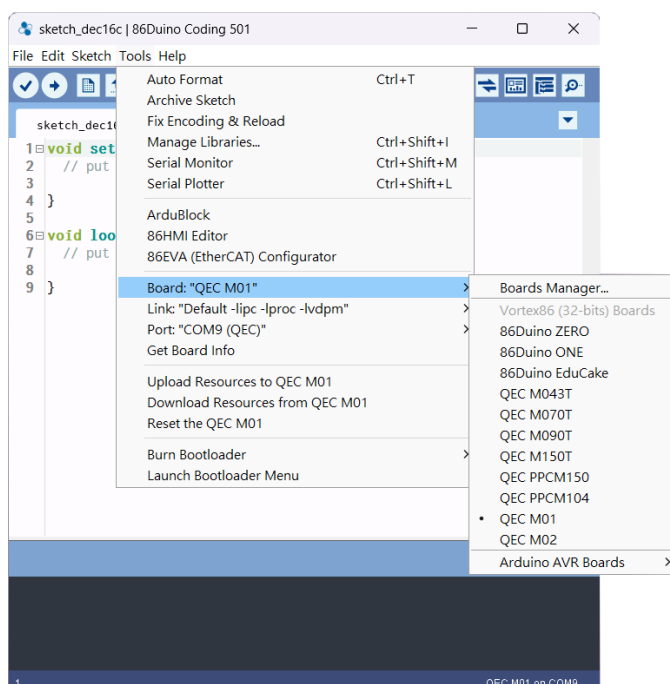
# 4. Write code

The EtherCAT Main-device (QEC-M-01P) and the EtherCAT Stepper Motor controller (QEC-R11MP3S) 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 QEC-R11MP3S into CiA402 Profile Position (PP) mode:
- EtherCAT Cycle Time: 1 millisecond.
- EtherCAT Mode: ECAT_SYNC.

The EthercatMaster object ("mdevice") represents the QEC-M-01, while the EthercatDevice_QECR11MP3S ("sub_device") represents the QEC-R11MP3S, and the EthercatDevice_CiA402 object ("motor") represents a CiA402 identifier to utilize the CiA402 Library for the specified motor driver.

**A. In Setup Function:**

In the setup() function initializes communication and configures the motor for CiA402 Profile Position (PP) mode. Follow the steps below:
1. Initialize Serial Communication
    - Start serial communication at a baud rate of 115200.
2. Start the EtherCAT Main-device
    - Begin the EtherCAT Main-device and set the EtherCAT state machine to the PRE-OPERATIONAL state.
3. Attach the Motor
    - Attach the QECR11MP3S device and retrieve the first motor axis using cia402GetServo(1). *This identifies the motor as a CiA402 object for controlling the stepper motor.*
4. Set Profile Position (PP) Mode
    - Configure the motor to PP mode using setCiA402Mode(CIA402_PP_MODE).
5. Start the EtherCAT Main-device
    - Use the start() function to switch the EtherCAT state machine to the OPERATIONAL state. Set the cycle time to 1ms and ECAT_SYNC mode.
6. Enable the Motor
    - Enable the motor and transition it to CIA402_OPERATION_ENABLED.
7. Configure Profile Parameters
    - Motion Profile Type: Linear Ramp, Profile Velocity: 10,000, Acceleration: 500, Deceleration: 500.

**B. In Loop Function:**

In the loop() function, the current position of the motor is displayed on the Serial Monitor, and the motor alternates its movement back and forth in a repeating cycle:

1. State Machine Logic
   - case 0: Start the motor and move to the target position (100,000 units). Once the command is successfully executed, transition to the next state.
   - case 1: Wait for the motor to reach the target position. Once the target is reached, proceed to the next state.
   - case 2: Start the motor and move back to the original position (-100,000 units). Once the command is successfully executed, transition to the next state.
   - case 3: Wait for the motor to return to the original position. Once the target is reached, reset the state machine back to case 0 to repeat the movement cycle.

2. Code Logic Summary
   - Use the pp_Run() function to initiate position movement.
   - Use the pp_IsTargetReached() function to confirm whether the target position has been reached.
   - The state machine starts at case 0 and resets after completing case 3.

The example code is as follows:

```
#include "Ethercat.h"


EthercatMaster mdevice;
EthercatDevice_QECR11MP3S sub_device;
EthercatDevice_CiA402 *motor;


int pp_state = 0;


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

  Serial.print("Begin: "); Serial.println(mdevice.begin());
  Serial.print("Slave: "); Serial.println(sub_device.attach(0, mdevice));
  motor = sub_device.cia402GetServo(1);
  motor->setCiA402Mode(CIA402_PP_MODE);
  Serial.print("Start: "); Serial.println(mdevice.start(1000000, ECAT_SYNC));

  Serial.print("Enable: "); Serial.println(motor->enable());
  motor->pp_SetMotionProfileType(0); // Linear ramp (trapezoidal profile)
  motor->pp_SetVelocity(10000);
  motor->pp_SetAcceleration(500);
  motor->pp_SetDeceleration(500);
```
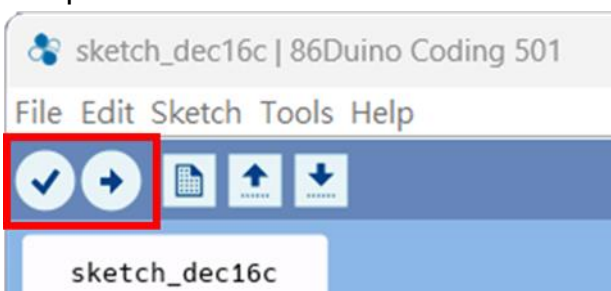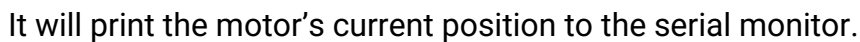
```
}

void loop() {
  Serial.print("Pos: "); Serial.println(motor->getPositionActualValue());
  switch (pp_state)
  {
    case 0:
      if (motor->pp_Run(100000) == 0)
        pp_state++;
      break;
    case 1:
      if (motor->pp_IsTargetReached())
        pp_state++;
      break;
    case 2:
      if (motor->pp_Run(-100000) == 0)
        pp_state++;
      break;
    case 3:
      if (motor->pp_IsTargetReached())
        pp_state = 0;
      break;
  }
}
```

**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. The program will run when the upload is complete.

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".



```
Begin: 0
17:10:39.198 -> Slave: 0
17:10:39.244 -> Start: 0
17:10:41.444 -> Enable: 0
17:10:41.444 -> Pos: -21970
17:10:41.444 -> Pos: -21968
```

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 Main-device's environment is abnormal. As shown in the figure below, please try updating your QEC EtherCAT Main-device's environment, which will include the following three items: Bootloader, EtherCAT firmware, and EtherCAT tool.
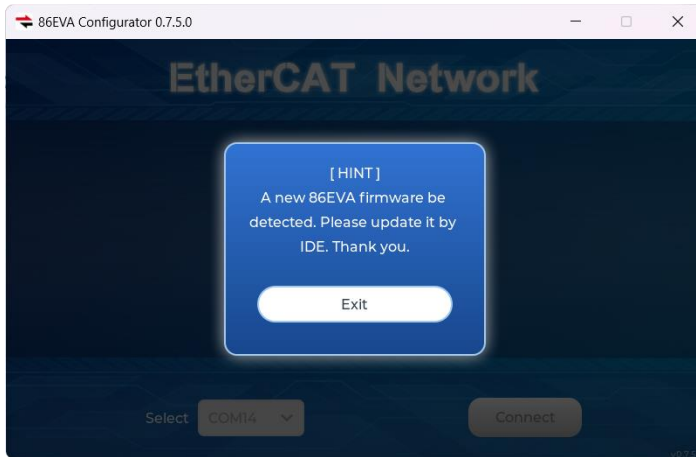


Now, we will further explain how to proceed with the update:

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.
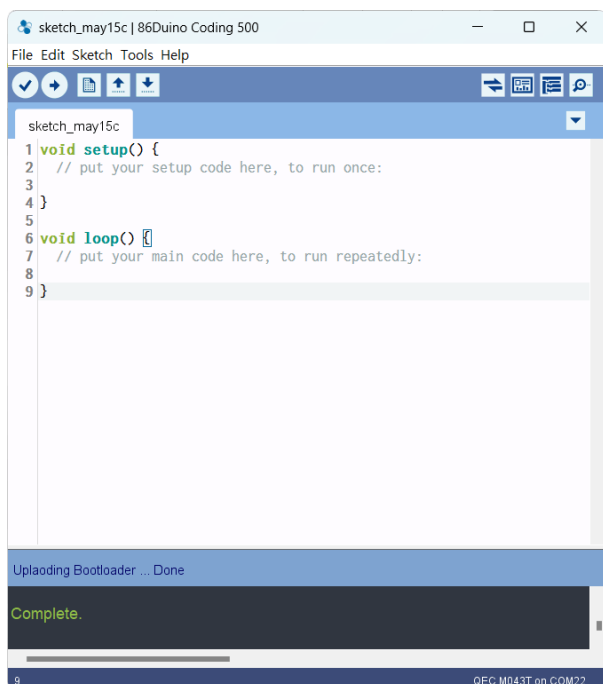
## Step 2: Click "Burn Bootloader" button

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:
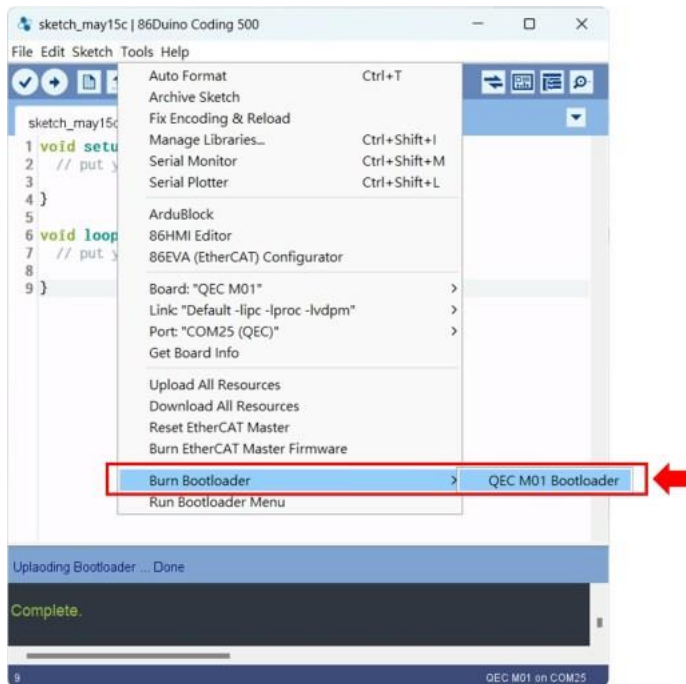


## Step 3: Complete the Update



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.