Start Guide

AMAX-4817 EtherCAT Analog Input



86Duino Coding IDE 501

EtherCAT Library

(Version 1.0)

Revision

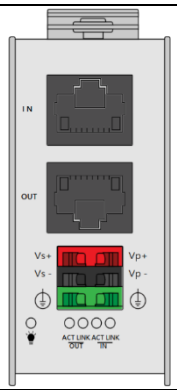
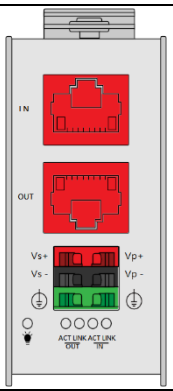
Date	Version	Description
2026/06/03	Version1.0	New Release.

Preface

In this guide, we will show you how to use the EtherCAT MDevice **QEC-M-01** and the **AMAX-4817** module (EtherCAT Analog Input module).

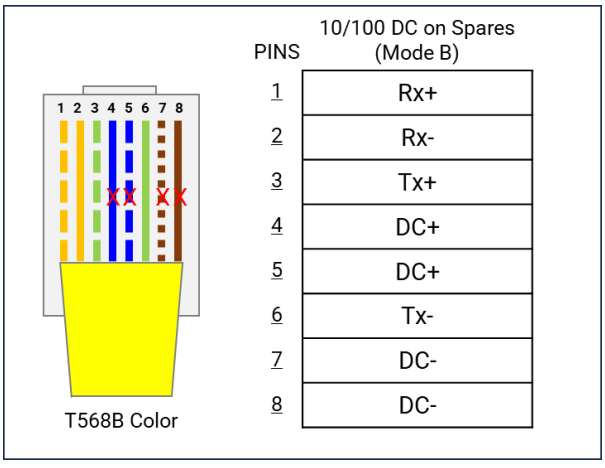
Notes QEC's PoE (Power over Ethernet)

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-

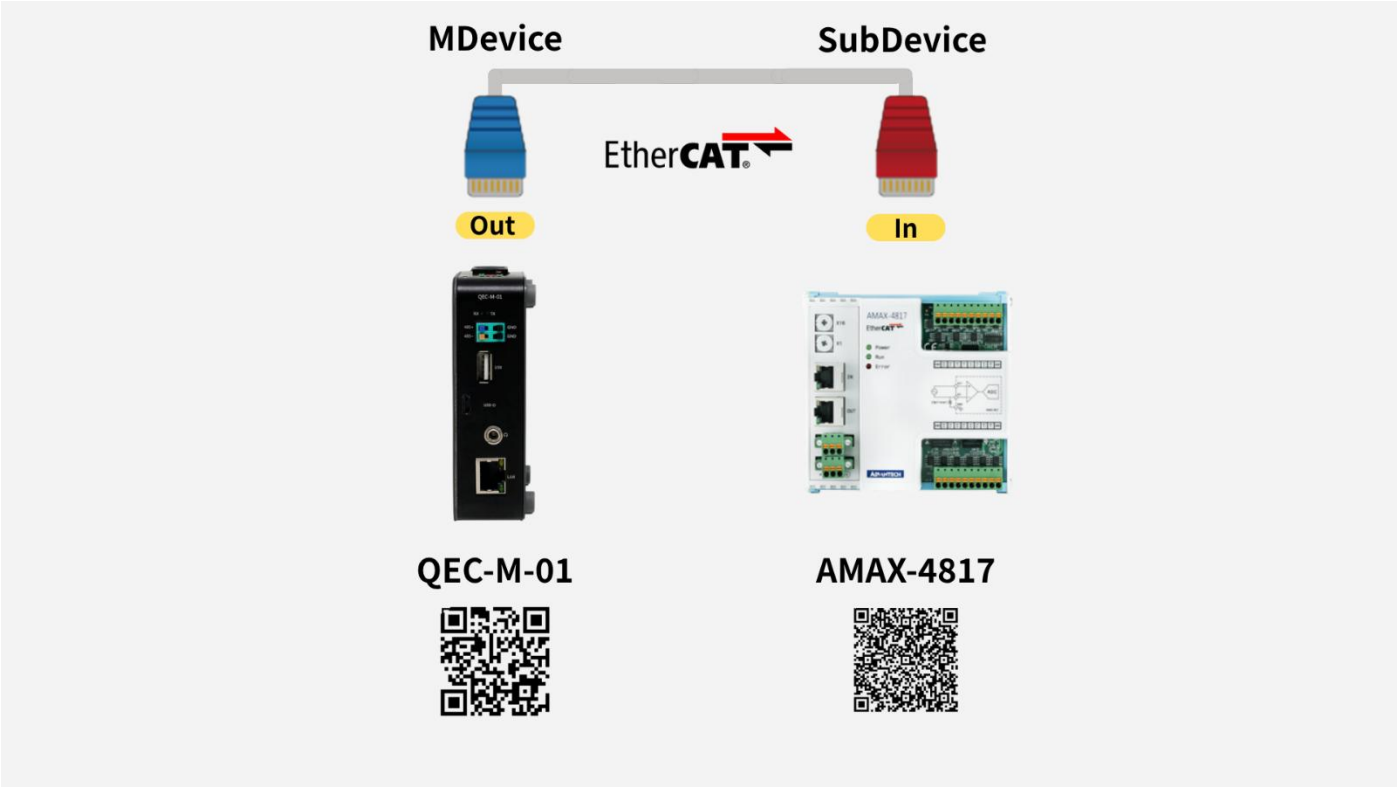
2. When connecting PoE and non-PoE devices, make sure to disconnect Ethernet cables at pins 4, 5, 7, and 8 (e.g., when a PoE-supported QEC EtherCAT MDevice connects with a third-party EtherCAT SubDevice).

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. Advantech AMAX-4817 (EtherCAT Analog Input module)
3. 24VDC power supply & EU-type terminal cable & LAN cable

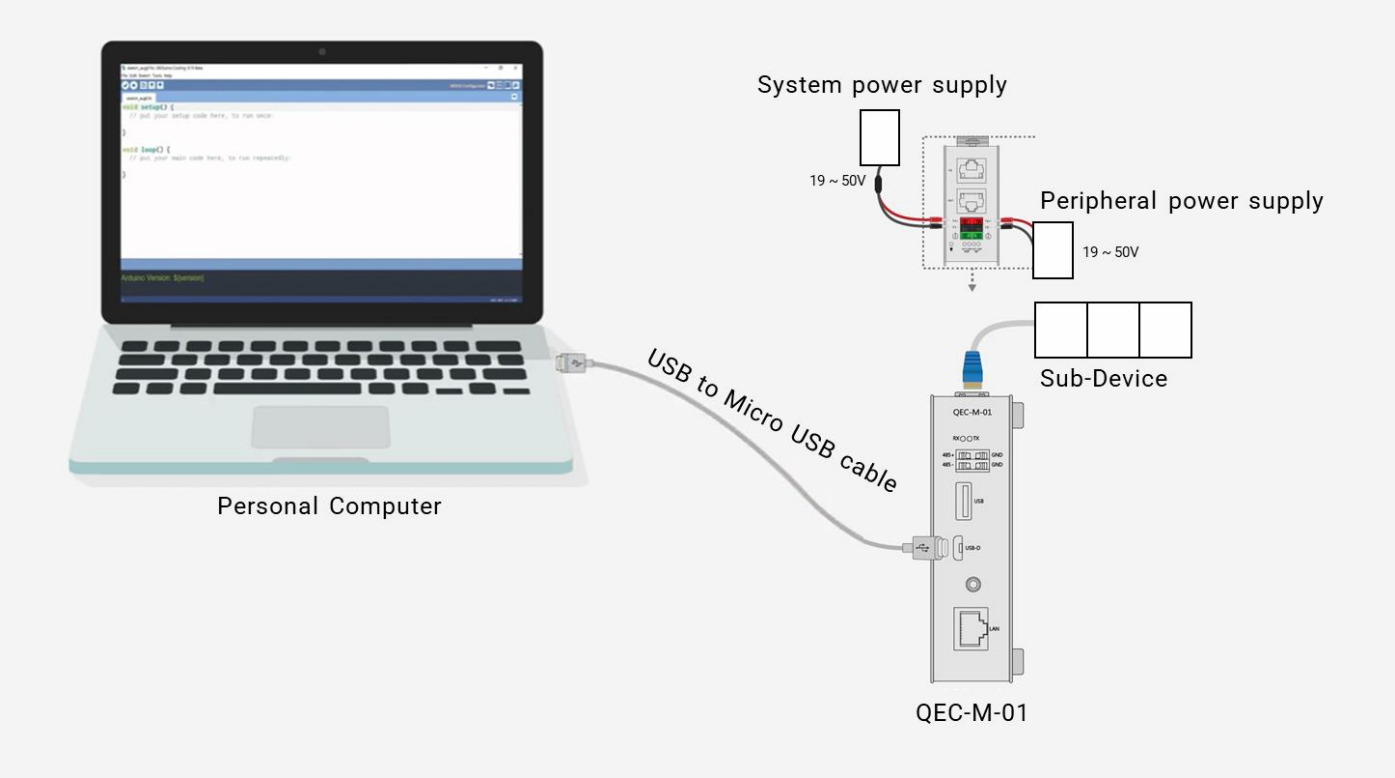


1.1 QEC-M-01

QEC-M-01 is an EtherCAT MDevice with real-time, reliable, and synchronous control. The 86Duino is an industrial Arduino-like software that makes it an easy-to-develop, stable, and economical automation system, allowing users to develop EtherCAT quickly.

The diagram shows a typical wiring example with a QEC MDevice and a Personal Computer.

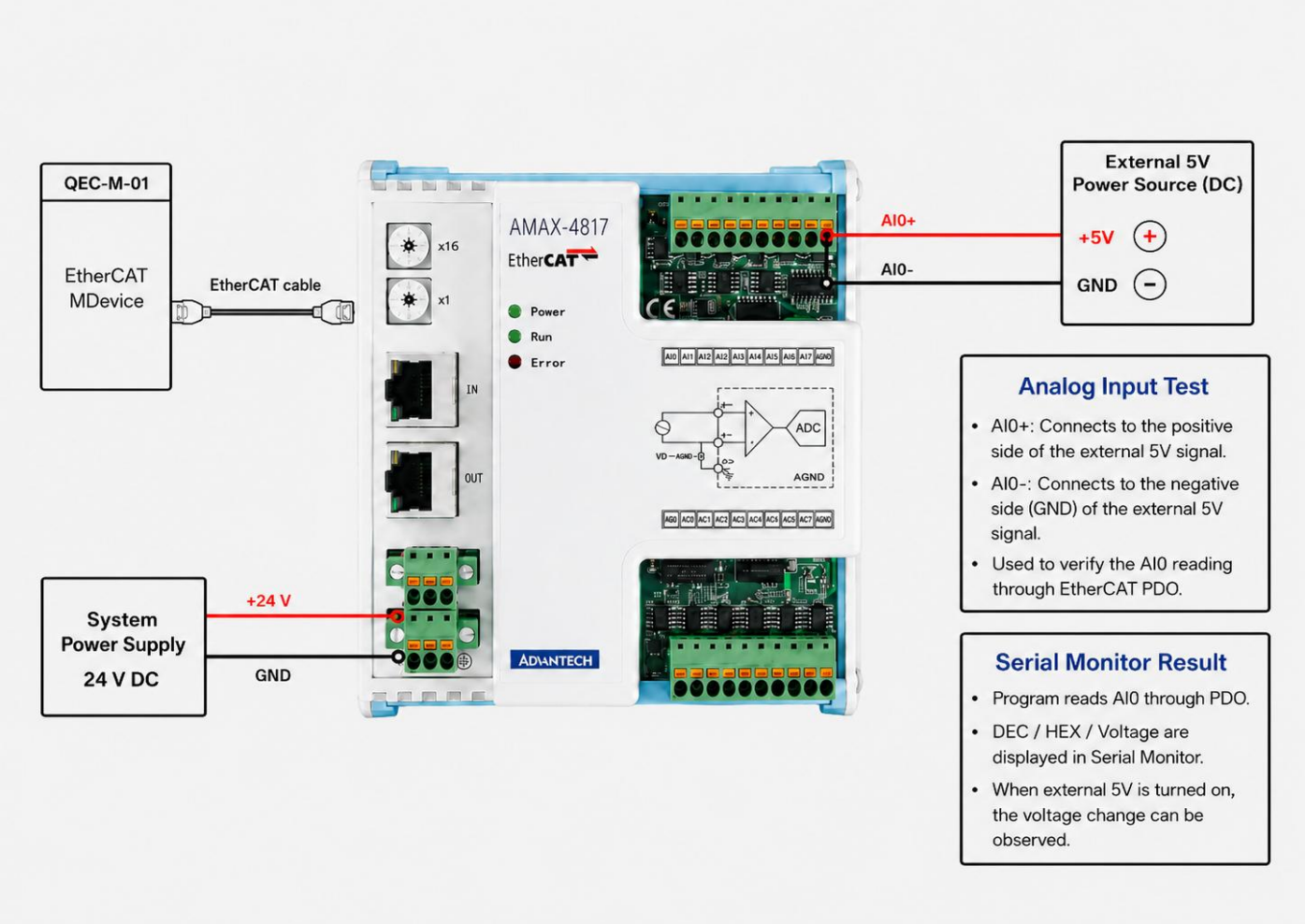
1. Power Supply: Connect to Vs+/Vs- and Vp+/Vp- power supplies via EU terminals for 24V power.
2. EtherCAT Connection: Using the EtherCAT Out port (On the top side) connected to the EtherCAT In port of EtherCAT SubDevice via RJ45 cable.



1.2 Advantech Analog Input Module

The Advantech AMAX-4817 is an EtherCAT remote analog input module with 8-channel, 16-bit isolated analog input. It can be used as an EtherCAT SubDevice in a QEC EtherCAT network to provide voltage/current signal acquisition from field devices.

The diagram below shows a typical wiring example using a **QEC MDevice** (for example, **QEC-M-01**) together with Bore Advantech Analog I/O module.

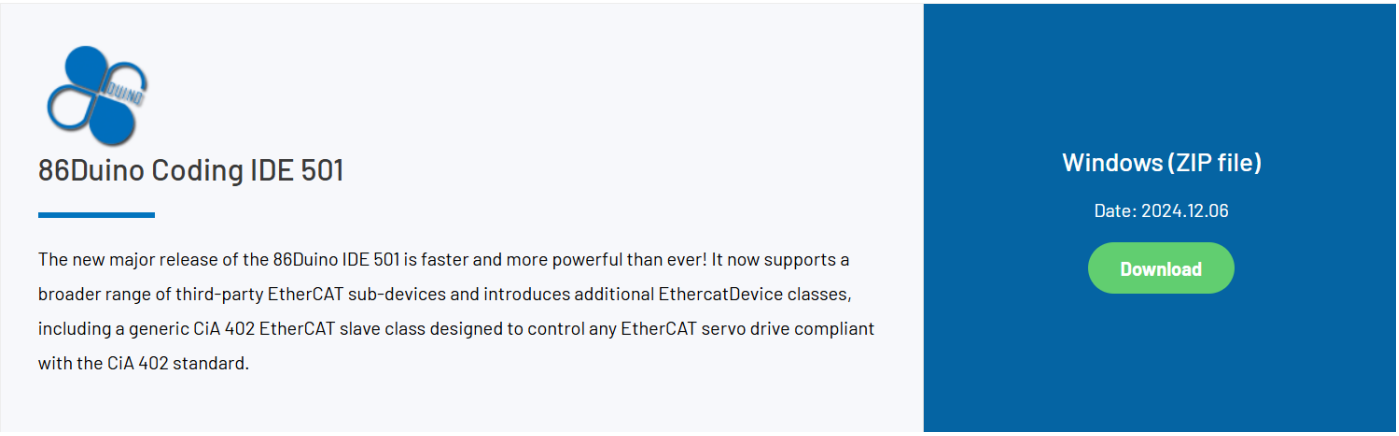


Connections are grouped by function:

- EtherCAT
  - IN: Connect to the EtherCAT port of the QEC MDevice or the previous SubDevice.
  - OUT: Connect to the next EtherCAT SubDevice, if any.
- Power & Grounding
  - Provide 24 VDC power for module operation and field I/O.
  - Connect 0 V (GND) properly to ensure stable communication.
- Analog Input
  - Input 1–8: Connect external analog signals to AI+ and AI-.
  - Select voltage or current input according to the field signal type.
- Indicators: Power / RUN / ERROR / I/O Indicator.

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.

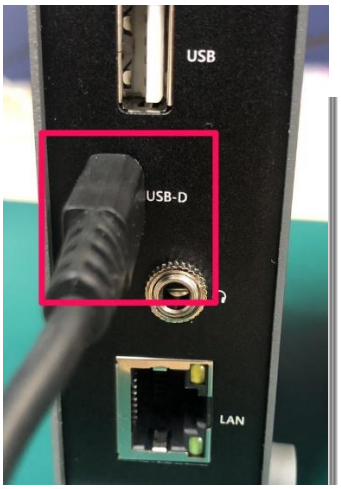
86Duino Coding IDE 501+ looks like below.

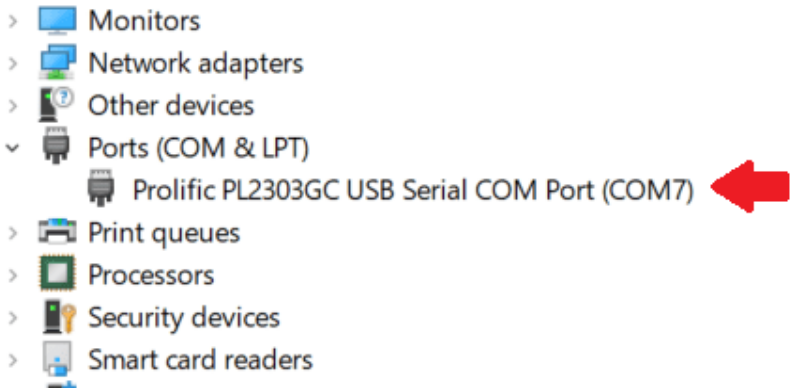


3. Connect to PC and set up the environment

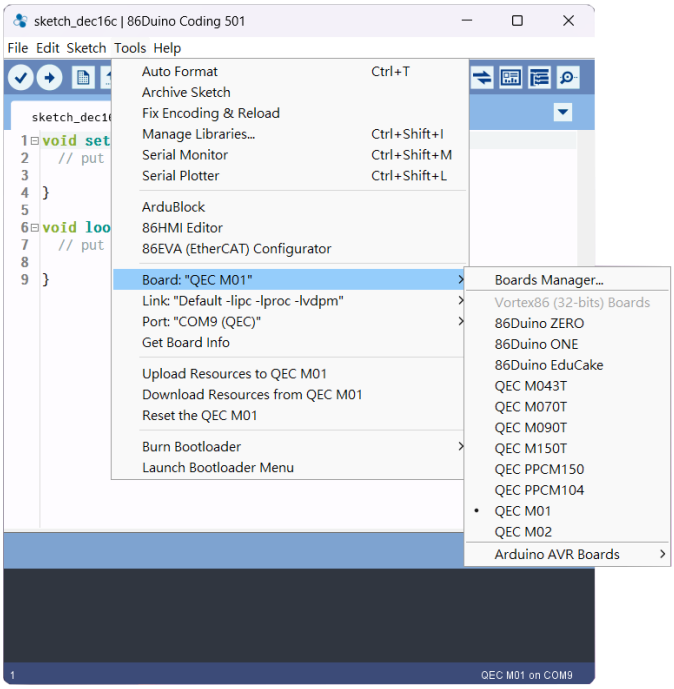
Follow the steps below to set up the environment:

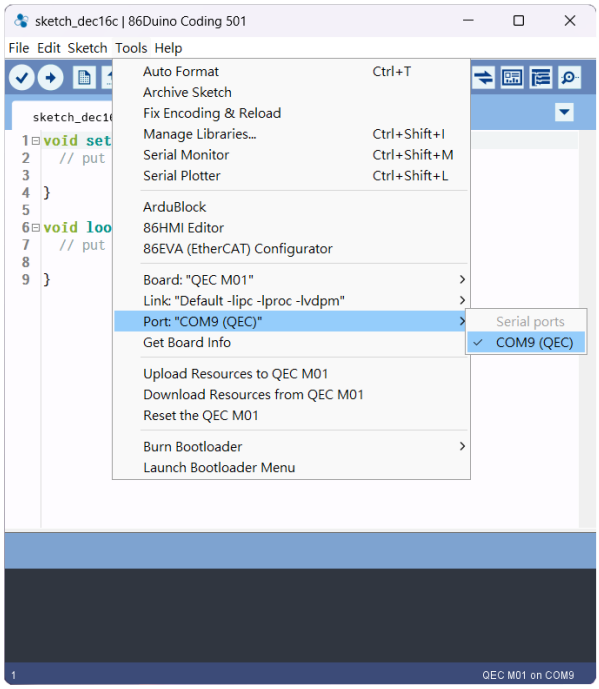
1. Connect the QEC-M-01 to your PC via a Micro USB to USB cable (86Duino IDE installed).
2. Turn on the QEC power.
3. Open “**Device Manager**” (select in the menu after pressing Win+X) ->” **Ports (COM & LPT)**” in your PC and expand the ports; you should see that the “**Prolific PL2303GC USB Serial 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-M01**” (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 Advantech AMAX-4817 module (EtherCAT Analog Input module) 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:

- EtherCAT Cycle Time: 1 millisecond.
- EtherCAT Mode: ECAT_SYNC.

The `EthercatMaster` object (“master”) represents the QEC-M-01, while the `EthercatDevice_Generic` object (“slave0”) represents the Advantech AMAX-4817 module.

We will connect a 5V signal to AI0+ and AI0- of the AMAX-4817, then use the program to read the AI0 value through EtherCAT PDO. The result will be displayed in the Serial Monitor, including the raw DEC value, HEX value, and the converted voltage, so we can check whether the analog input reading is correct.

A. In Setup Function:

In the `setup()` function, initialize communication, enter OP, and configure the analog ranges. Follow the steps below:

1. Initialize Serial Communication
    - Start serial communication at a baud rate of 115200.
2. Start the EtherCAT MDevice
    - Use the `begin()` function to begin the EtherCAT MDevice and set the EtherCAT state machine to the PRE-OPERATIONAL state.
3. Attach the AMAX-4817 EtherCAT Analog Input module
    - Use the `attach()` function to attach to the EtherCAT Network. Set the node number and the specific MDevice.
4. Start the EtherCAT MDevice
    - Use the `start(1000000, ECAT_SYNC)` function to switch the EtherCAT state machine to the OPERATIONAL state. Set the cycle time to 1ms and ECAT_SYNC mode.
5. Analog Input Range Setting
    - For 0–10V input, set AI0 Range to 1 through object 0x2180:01.
    - If the range is not changed, the default setting is 0, which corresponds to ±10V / ±20mA.

B. In Loop Function:

In the `loop()` function, we will read the AI0 value from the AMAX-4817 and print the result to the Serial Monitor every second.

1. Read AI0 PDO Input Data
    - Use `pdoRead16(0)` to read the first 16-bit PDO input data.
    - In this example, PDO offset 0 is used to read AI0.
2. Convert the Raw Value to Voltage
    - Convert the raw AI0 value into an approximate voltage value.
    - When AI0 is configured as 0–10 V input range, use the following formula: `Voltage = Raw Value × 10 / 32767`
3. Print the AI0 Result
    - Print the AI0 raw value in DEC format.
    - Print the same raw value in HEX format.
    - Print the converted voltage value to the Serial Monitor.
4. Update Time
    - Wait 1000 ms (`delay(1000)`) before reading the next value.
    - This allows the AI0 value to be updated once per second.
5. Code Logic Summary
    - `pdoRead16(0)`: reads the 16-bit AI0 value from PDO offset 0.
    - `printHex16(value)`: prints the raw value in 4-digit HEX format.
    - The example reads AI0 every second and displays the raw DEC value, HEX value, and converted voltage.
    - When a 5 V signal is connected to AI0+ and AI0-, the Serial Monitor should show a voltage value close to 5 V.

The example code is as follows:

```
#include "Ethercat.h"

EthercatMaster master;
EthercatDevice_Generic Slave0;

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

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

  Serial.print("Slave0 attach: ");
  Serial.println(Slave0.attach(0, master));

  Serial.print("master start: ");
  Serial.println(master.start(1000000, ECAT_SYNC));
}

void loop() {
  uint16_t ai0_raw = Slave0.pdoRead16(0);
  float ai0_voltage = (float)ai0_raw * 10.0 / 32767.0;

  Serial.print("AI0 DEC = ");
  Serial.print(ai0_raw);

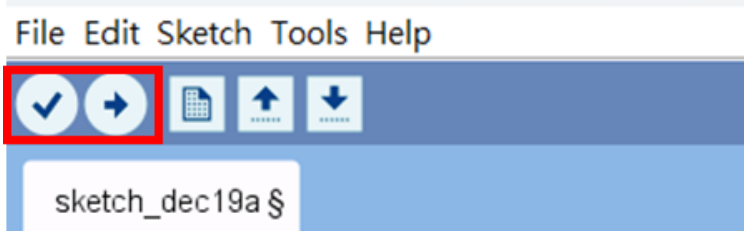
  Serial.print(" , HEX = 0x");
  printHex16(ai0_raw);

  Serial.print(" , Voltage = ");
  Serial.print(ai0_voltage, 3);
  Serial.println(" V");

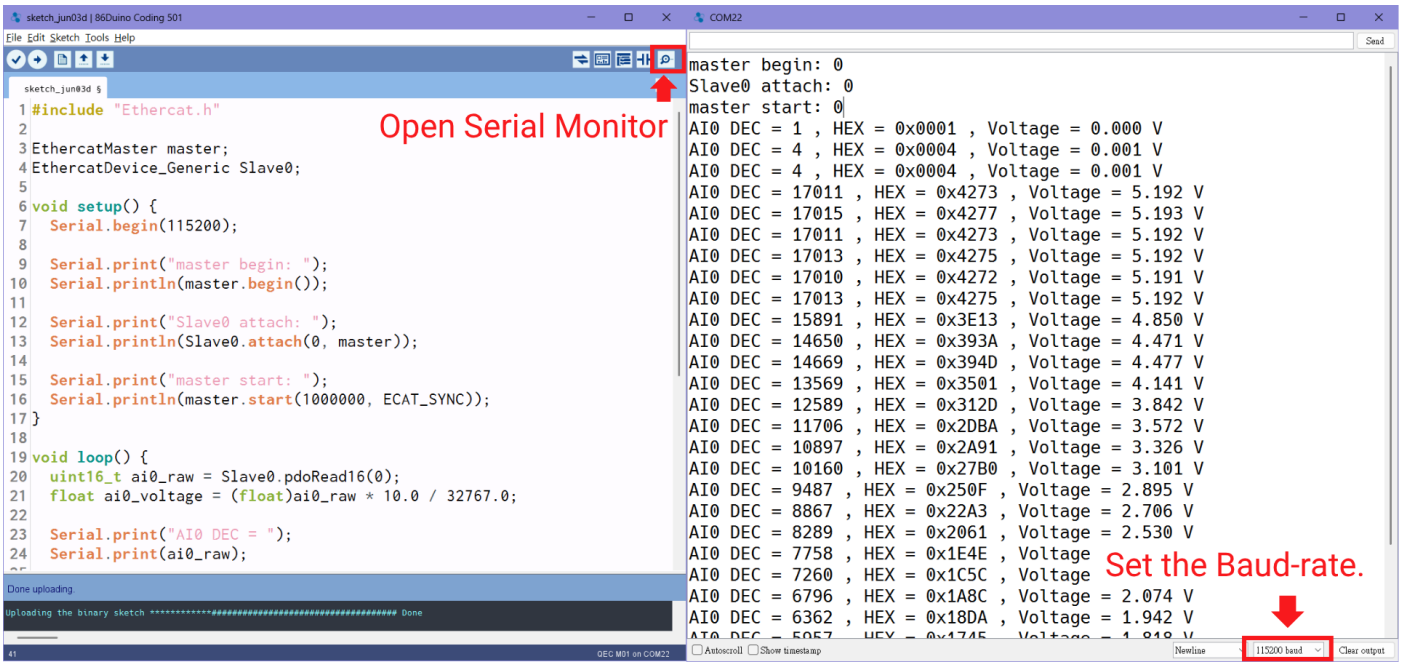
  delay(1000);
}

void printHex16(uint16_t value)  {
  if (value < 0x1000) Serial.print("0");
  if (value < 0x0100) Serial.print("0");
  if (value < 0x0010) Serial.print("0");
  Serial.print(value, HEX);
}
```

***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 “0” for each status for EtherCAT.

```
master begin: 0
Slave0 attach: 0
master start: 0
```

It will print the analog input AI0 value (Voltage) to the serial monitor.

```
AI0 DEC = 1 , HEX = 0x0001 , Voltage = 0.000 V
AI0 DEC = 4 , HEX = 0x0004 , Voltage = 0.001 V
AI0 DEC = 4 , HEX = 0x0004 , Voltage = 0.001 V
```

When the external 5V power supply is turned on, the voltage change can be observed in the Serial Monitor.

```
AI0 DEC = 4 , HEX = 0x0004 , Voltage = 0.001 V
AI0 DEC = 17011 , HEX = 0x4273 , Voltage = 5.192 V
AI0 DEC = 17015 , HEX = 0x4277 , Voltage = 5.193 V
```

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:

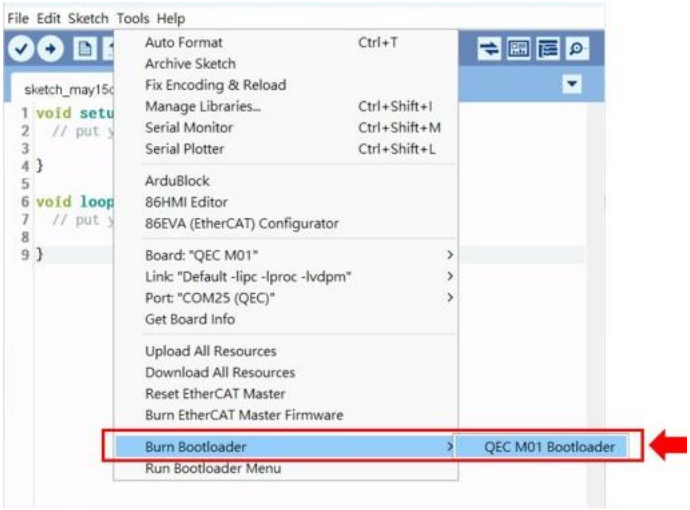
Step 1: Setting up QEC-M

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.

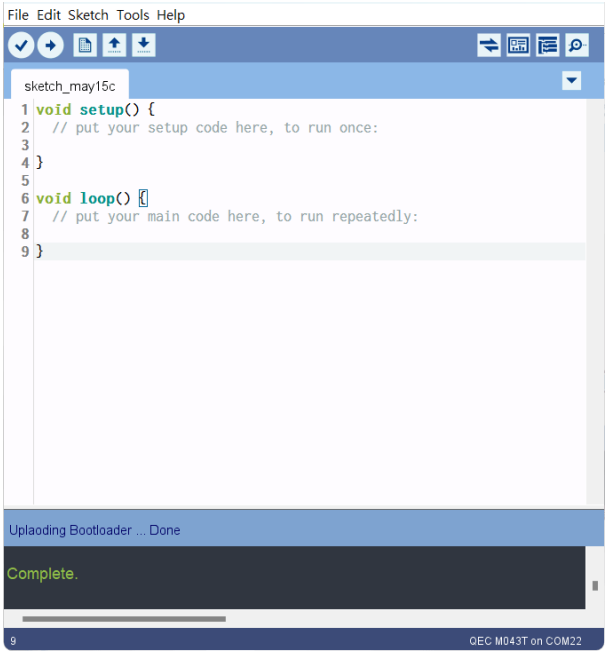
Step 2: Click “Burn Bootloader” button

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.

All Trademarks appearing in this manuscript are registered trademark of their respective owners. All Specifications are subject to change without notice.

©ICOP Technology Inc. 2025