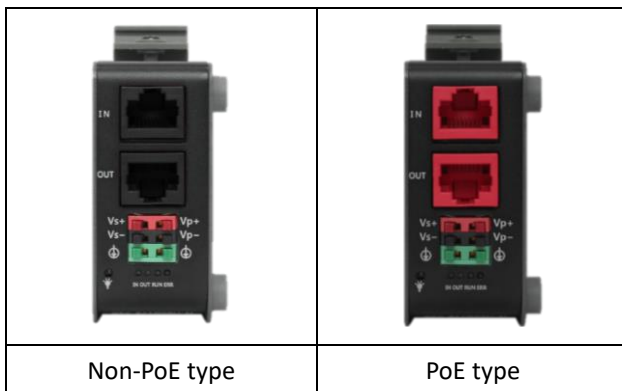# QEC Start Guide 5: MPG

In this guide, we will show you how to use the EtherCAT Master QEC-M-01P and the QEC-RXXHUX Series (EtherCAT HID Slave)。
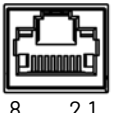
Notes QEC's PoE (Power over Ethernet)

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:

	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)

   * PoE LAN with the Red Housing; Regular LAN with Black Housing.

   * 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 master connects with a third-party EtherCAT slave).
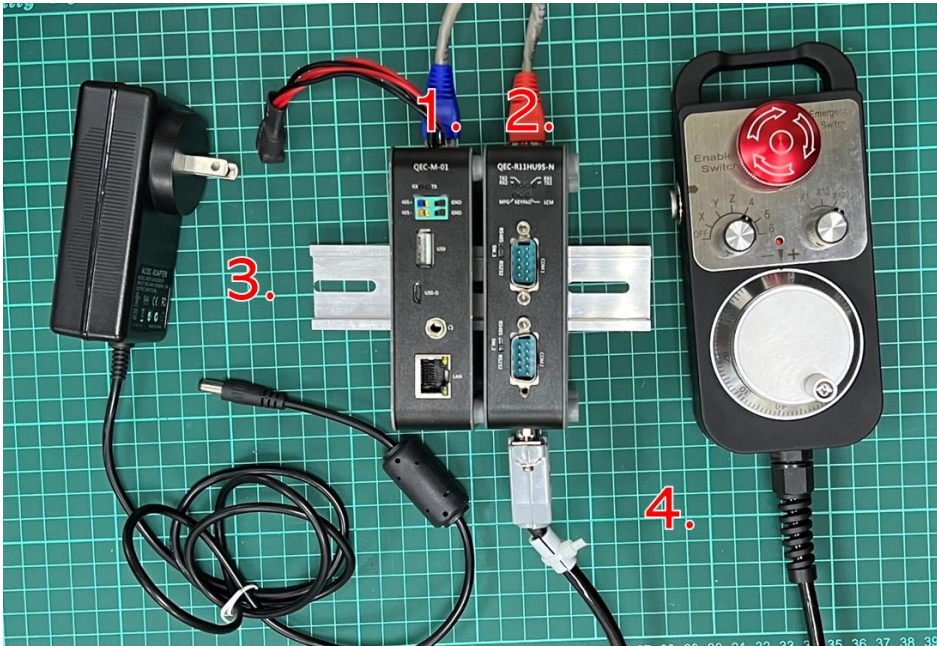

3.   QEC's PoE power supply is up to 24V/3A.

Connection and wiring hardware

The following devices are used here:

1. QEC-M-01P（EtherCAT Master/PoE）
2. QEC-R11HU9S-N (EtherCAT HID Slave, supports 2 UART, 1 MPG, 1 Keypad, 1 LCM) & LAN cable
3. 24V power supply & EU-type terminal cable
4. MPG Hand wheel

QEC-M-01P

The QEC EtherCAT master with PoE functions.

1. Using the EtherCAT Out port (top side) connected to the EtherCAT In port of QEC-R11HU9S via RJ45 cable (powered by PoE).
2. Connect to Vs+/Vs- and Vp+/Vp- power supplies via EU terminals for 24V power. And provide power to QEC-R11HU9S-n via the PoE function.
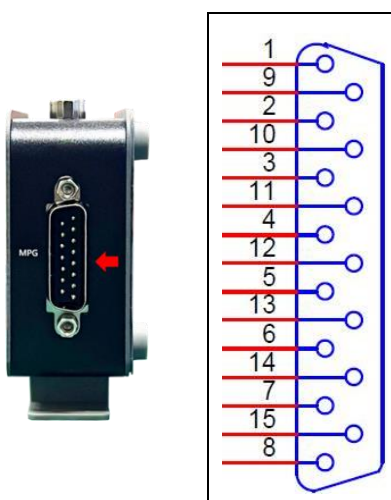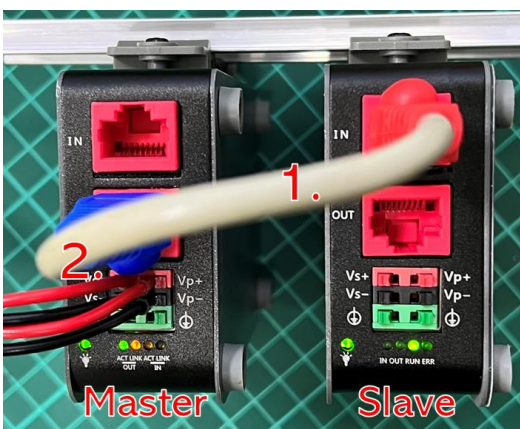
QEC-R11HU9S-N

The QEC-R11HU9S-N with PoE function.

● Use the hand wheel to connect to the MPG interface at the bottom of QEC-R11HU9S-N.
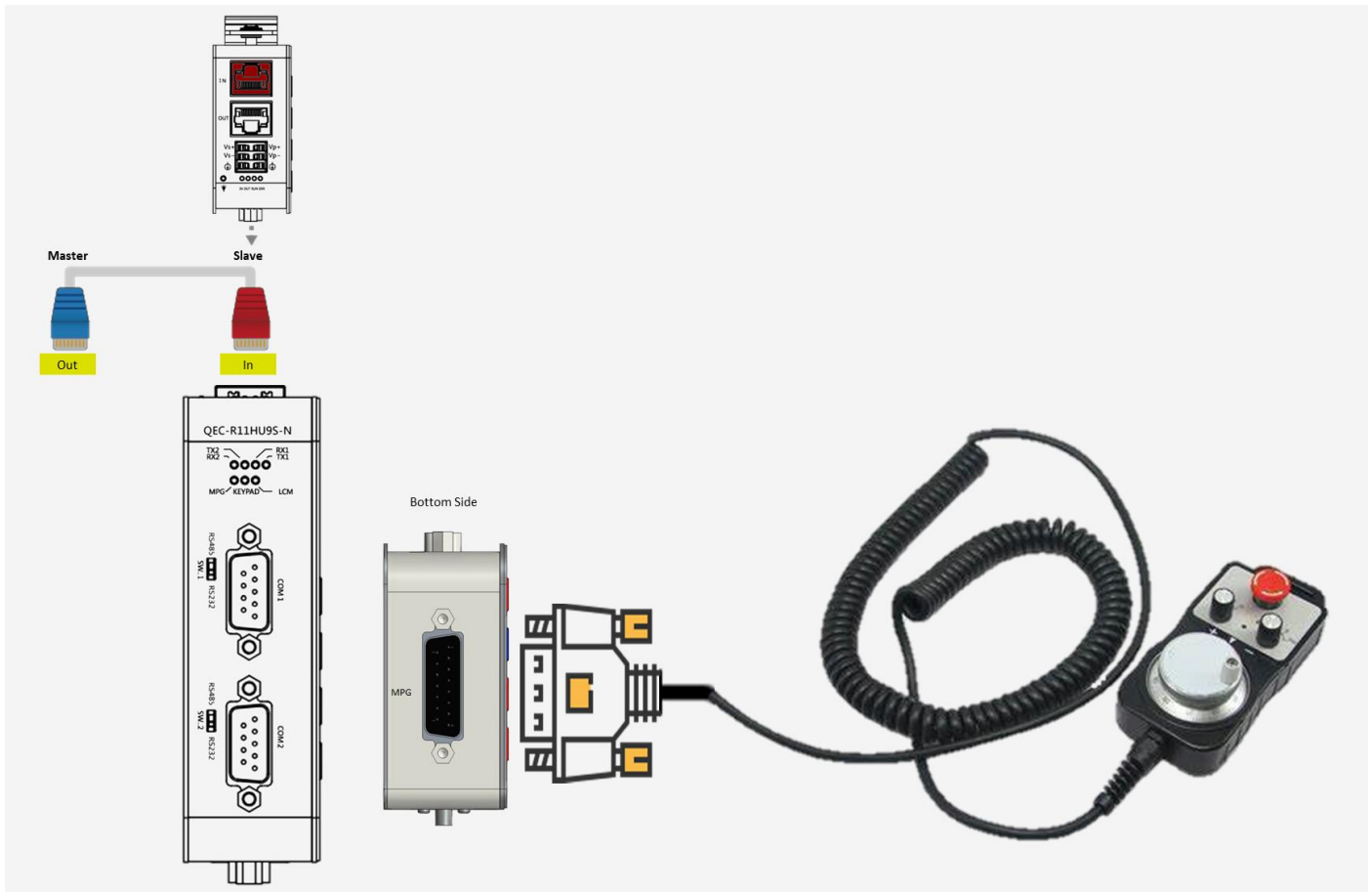
Notes: Regarding the signal assignment of the MPG connector at the bottom of QEC-R11HU9S-N.

No.	Pin Assignment	No.	Pin Assignment
1	VCC	9	AXIS_B0
2	A	10	AXIS_B1
3	A-	11	AXIS_B2
4	B	12	MULTIPLE_B0
5	B-	13	MULTIPLE_B1
6	C	14	EMERGENCY
7	C-	15	LED
8	GND	-	-

Software/Development Environment: 86Duino IDE

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

QEC
Quicker, Easier Control

Download

The open-source 86Duino Software (IDE) makes it easy to write code and upload it to the QEC. Refer to the *Getting Started page* for Installation instructions.
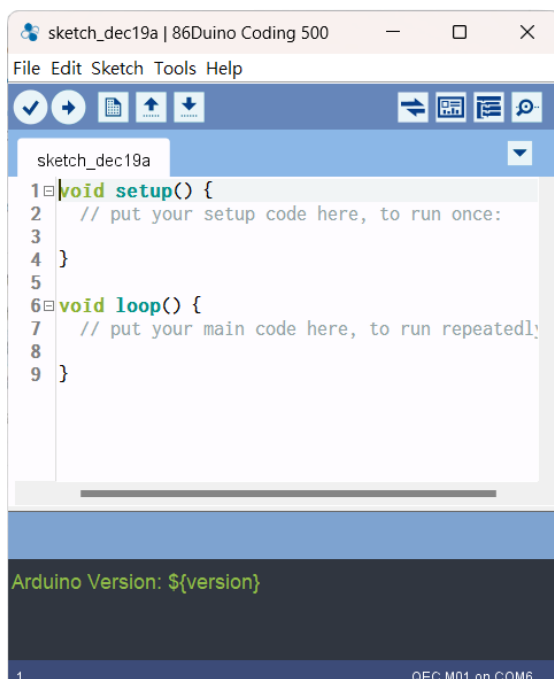
86Duino_Coding_500_Beta_202 30926_13

Download

About how to update the QEC Master (QEC-M series products) with the latest version of the 86Duino IDE, please see **this page**.

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

tools	2023/10/26 14:54
tools-builder	2023/10/26 14:54
86duino.exe	2023/10/26 14:49
86duino_debug.exe	2023/10/26 14:49
arduino.l4j.ini	2023/10/26 14:49

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

86Duino Coding IDE 500+ looks like below.

```
void setup() {
  // put your setup code here, to run once:

}

void loop() {
  // put your main code here, to run repeatedly

}
```

Arduino Version: ${version}

QEC M01 on COM6

# Connect to your 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](#))
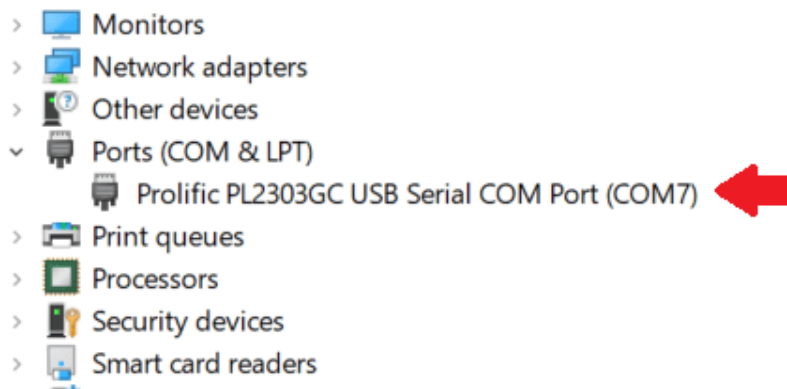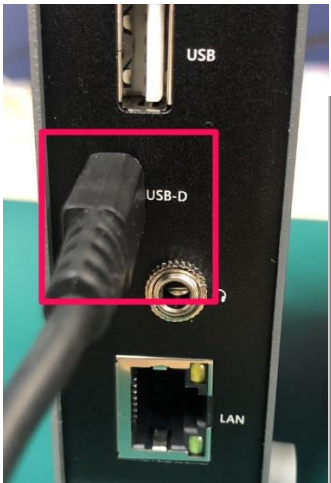
4. Open the 86Duino IDE.

5. Select the correct board: In the IDE's menu, select Tools> Board > QEC-M-01 (or the QEC-M master model you use).

6. Select Port: In the IDE's menu, select Tools > Port and select the USB port to connect to the QEC-M master (in this case, COM4 (QEC)).
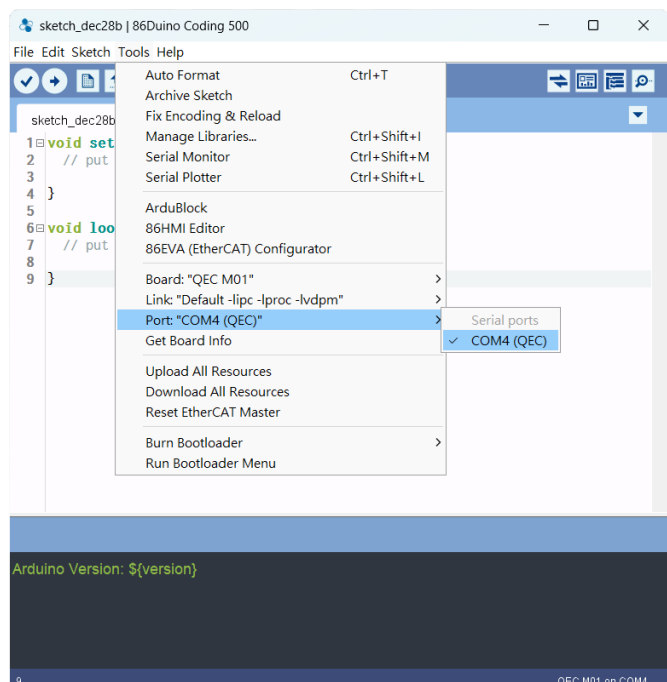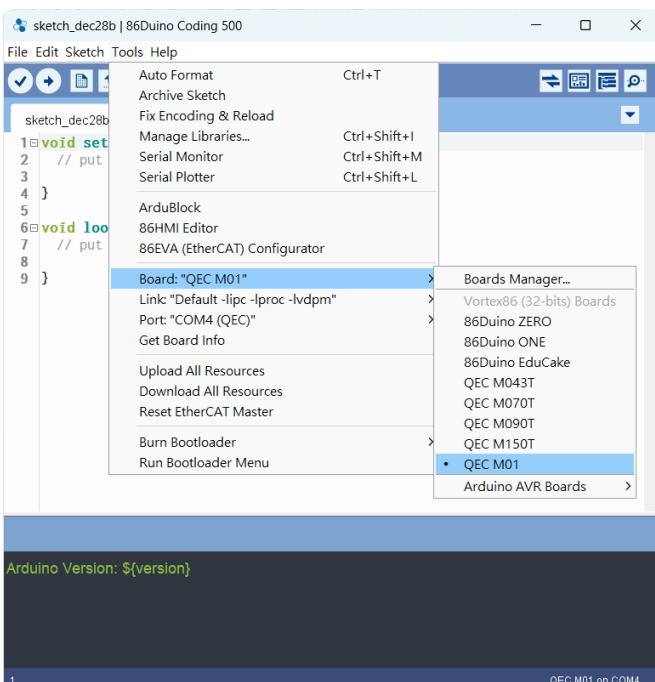
Development Method 1: Write code

The EtherCAT master (QEC-M-01P) and the HID slave (QEC-R11HU9S-N) can be configured and programmed via the EtherCAT library in the 86Duino IDE. The Arduino development environment has two main parts: `setup()` and `loop()`; `setup()` corresponds to initialization which do once, and `loop()` corresponds to main programs which do repeat. Before operating the EtherCAT network, you must configure it once. The process should be from Pre-OP to OP mode in EtherCAT devices.

In the following example, we want to read the data and status of the MPG of QEC-R11HU9S, and print out EMG, Enable, Axis, Ratio, Raw, Logical data through the Serial Monitor of 86Duino IDE.

When using QEC Slave, you can use the dedicated QEC Ethercat Slave Library. For example, QEC-R11HU9S can be used EthercatDevice_QECRXXHU Class. The example code is as follows:

```
#include "Ethercat.h" // Include the EtherCAT Library


EthercatMaster EcatMaster; // Create an EtherCAT Master Object

EthercatDevice_QECR11HU9S Slave1; // Create an EtherCAT Slave Object for QEC R11HU9S


// Callback function for cyclic updates

void myCallback() {

  Slave1.update(); // Update the Ethercat slave

}


void setup() {

  Serial.begin(115200); // Begin serial communication at 115200 baud rate

  while (!Serial); // Wait for the serial port to connect


  // Initialize the EtherCAT Master. If successful, all slaves enter PRE OPERATIONAL state

  EcatMaster.begin();


  // Attach QECR11HU9S slave device to the EtherCAT Master at position 0

  Slave1.attach(0, EcatMaster);


  // Set a cyclic callback for the Ethercat Master

  EcatMaster.attachCyclicCallback(myCallback);


  // Start the EtherCAT Master. If successful, all slaves enter OPERATIONAL state

  // Sync Mode, and the parameter 1000000 sets the cycle time in nanoseconds

  EcatMaster.start(1000000, ECAT_SYNC);

}


void loop() {
```

```
    // Read various parameters from the slave and print them

    Serial.print("EMG: ");

    Serial.print(Slave1.mpgReadEmergencyStop()); // Read and print the emergency stop status

    Serial.print(", Enable: ");

    Serial.print(Slave1.mpgReadEnableSwitch()); // Read and print the enable switch status

    Serial.print(", Axis: ");

    Serial.print(Slave1.mpgReadAxis()); // Read and print the axis information

    Serial.print(", Ratio: ");

    Serial.print(Slave1.mpgReadRatio()); // Read and print the ratio

    Serial.print(", Raw: ");

    Serial.print(Slave1.mpgReadEncoderRaw()); // Read and print the raw encoder value

    Serial.print(", Logical: ");

    Serial.println(Slave1.mpgReadEncoder()); // Read and print the logical encoder value

}
```

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.
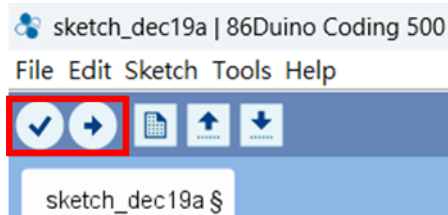
After the upload is completed, you can read the data and status of the MPG of QEC-R11HU9S, and view EMG, Enable, Axis, Ratio, Raw, Logical and other data through the Serial Monitor of 86Duino IDE.
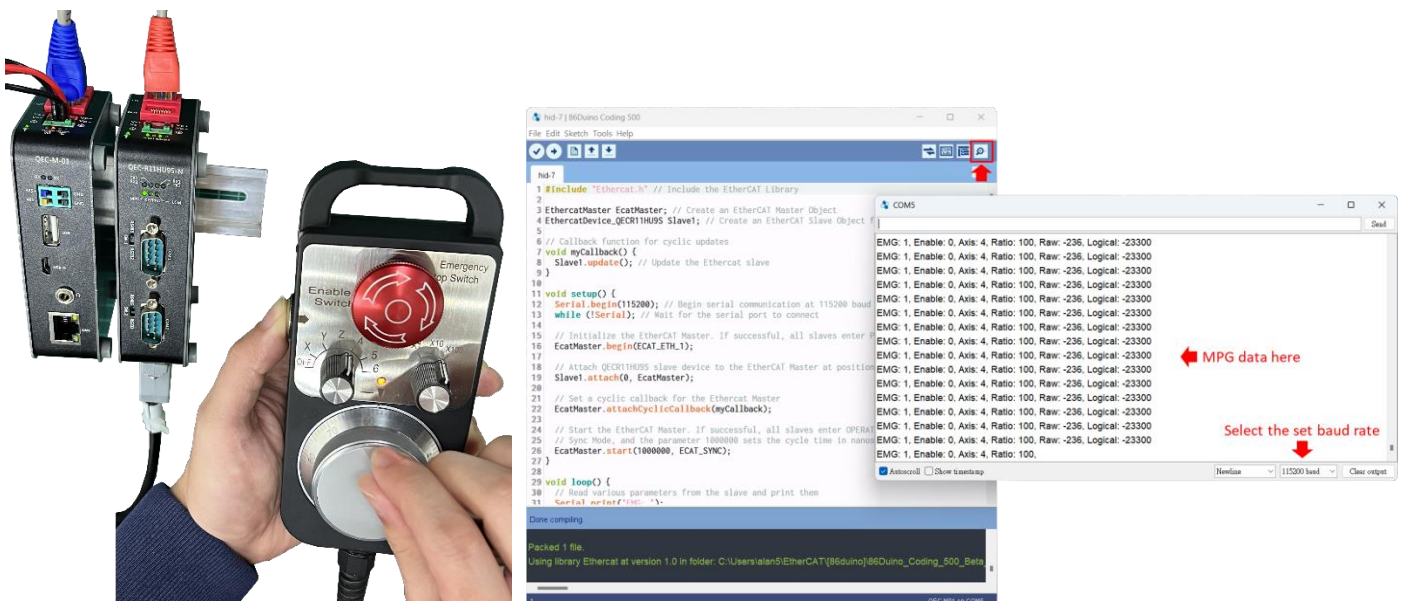
Development Method 2: Use 86EVA with code

86EVA is a graphical EtherCAT configurator based on the EtherCAT Library in the 86Duino IDE and is one of the development kits for 86Duino. The user can use it to configure the EtherCAT network quickly and start programming.

In the following example, we want to read the data and status of the MPG of QEC-R11HU9S, and print out EMG, Enable, Axis, Ratio, Raw, Logical data through the Serial Monitor of 86Duino IDE.

Step 1: Turn on 86EVA and scan

The 86EVA tool can be opened via the following buttons.
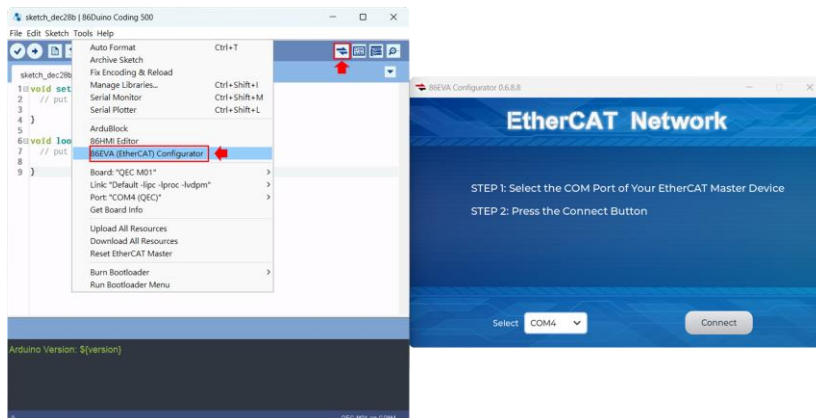

Once you have confirmed that the correct COM port has been selected of QEC-M-01P, press the Connect button to start scanning the EtherCAT network.

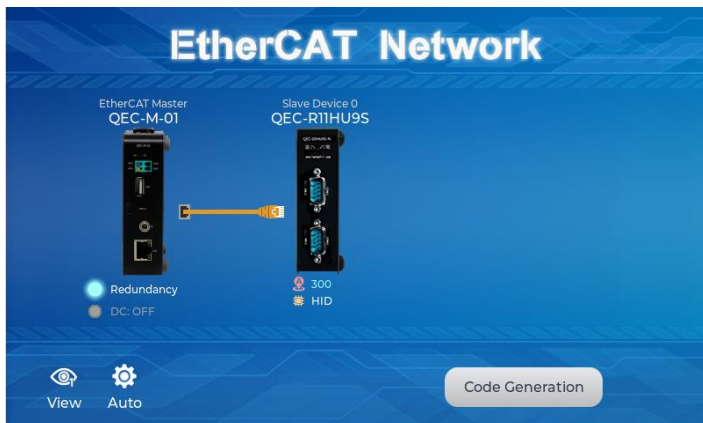The connected devices will be displayed after the EtherCAT network has been scanned.

Press the "View" button in the lower left corner to check the device's status (Voltage, Current, and Temperature; View2) and operating time (Hours; View3).
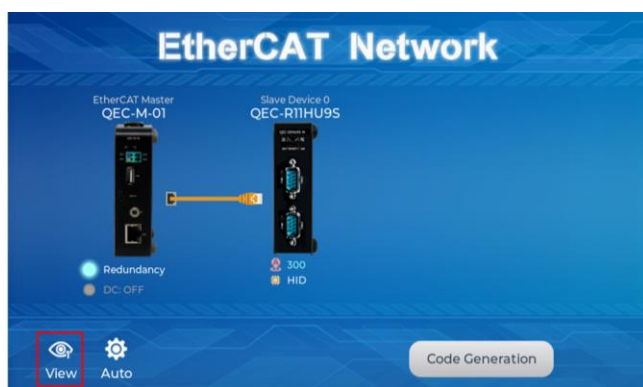
Step 2: Set the parameters

Press twice on the scanned device image to enter the corresponding parameter setting screen.

QEC-M-01

Press twice on the image of the QEC-M-01 to see the parameter settings.

This example will use the default settings and not change any settings; please click "Back" in the upper left corner to return.

QEC-R11HU9S-N

Press twice on the image of the QEC-R11HU9S to see the parameter settings.

Go to the "MPG Port Mapping" area. Among them, we select "Virtual MPG Port1" in the drop-down box of MPG in "MPG Port Mapping".

After finishing, click "Back" in the upper left corner to return.

These actions are to set the QEC-R11HU9S's MPG to virtual MPG port1 of EVA.

Step 3: Generate the code

Once you've set your device's parameters, go back to the home screen and press the "Code Generation" button in the bottom right corner.
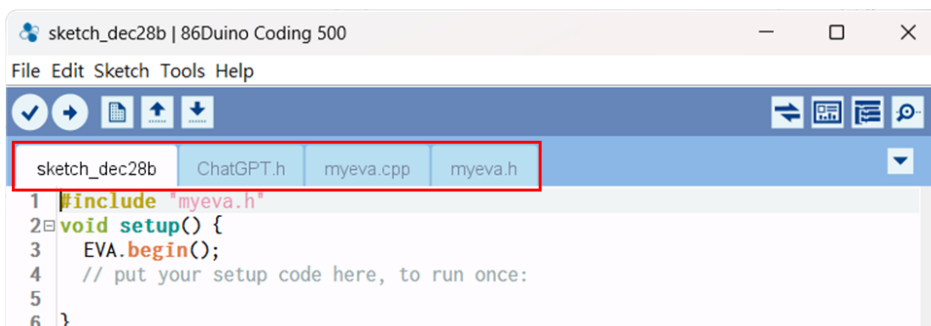

When you're done, double-click the OK button to turn off 86EVA, or it will close in 10 seconds.

The generated code and files are as follows:

- sketch_dec28b: Main Project (.ino, depending on your project name)
- ChatGPT.h: Parameters to provide to ChatGPT referred
- myeva.cpp: C++ program code of 86EVA
- myeva.h: Header file of 86EVA

Additional note: After 86EVA generates code, the following code will be automatically generated in the main program (.ino), and any of them missing will cause 86EVA not to work.

1. `#include "myeva.h"` : Include EVA Header file
2. `EVA.begin()` in `setup();` : Initialize the EVA function

Step 4: Write the code

In the following example, we want to read the data and status of the MPG of QEC-R11HU9S, and print out EMG, Enable, Axis, Ratio, Raw, Logical data through the Serial Monitor of 86Duino IDE.

```
#include "myeva.h" // Include the 86eva header file


void setup() {
    EVA.begin(); // Initialize EVA function
  Serial.begin(115200); // Begin serial communication at 115200 baud rate
  while (!Serial); // Wait for the serial port to connect


  VirtualMpg1.begin(); // Initialize the Virtual handwheel 1
}


void loop() {
  // Read various parameters from the slave and print them
  Serial.print("EMG: ");
  Serial.print(VirtualMpg1.isEmergencyStop()); // Read and print the emergency stop status
  Serial.print(", Enable: ");
  Serial.print(VirtualMpg1.isEnabled()); // Read and print the enable switch status
  Serial.print(", Axis: ");
  Serial.print(VirtualMpg1.currentAxis()); // Read and print the axis information
  Serial.print(", Ratio: ");
  Serial.print(VirtualMpg1.currentRatio()); // Read and print the ratio
  Serial.print(", Raw: ");
  Serial.print(VirtualMpg1.readRaw()); // Read and print the raw encoder value
  Serial.print(", Logical: ");
  Serial.println(VirtualMpg1.read()); // Read and print the logical encoder value
}
```

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.
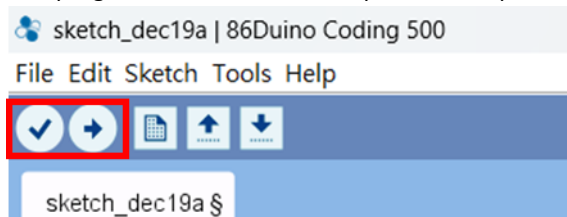
After the upload is completed, you can read the data and status of the MPG of QEC-R11HU9S, and view EMG, Enable, Axis, Ratio, Raw, Logical and other data through the Serial Monitor of 86Duino IDE.
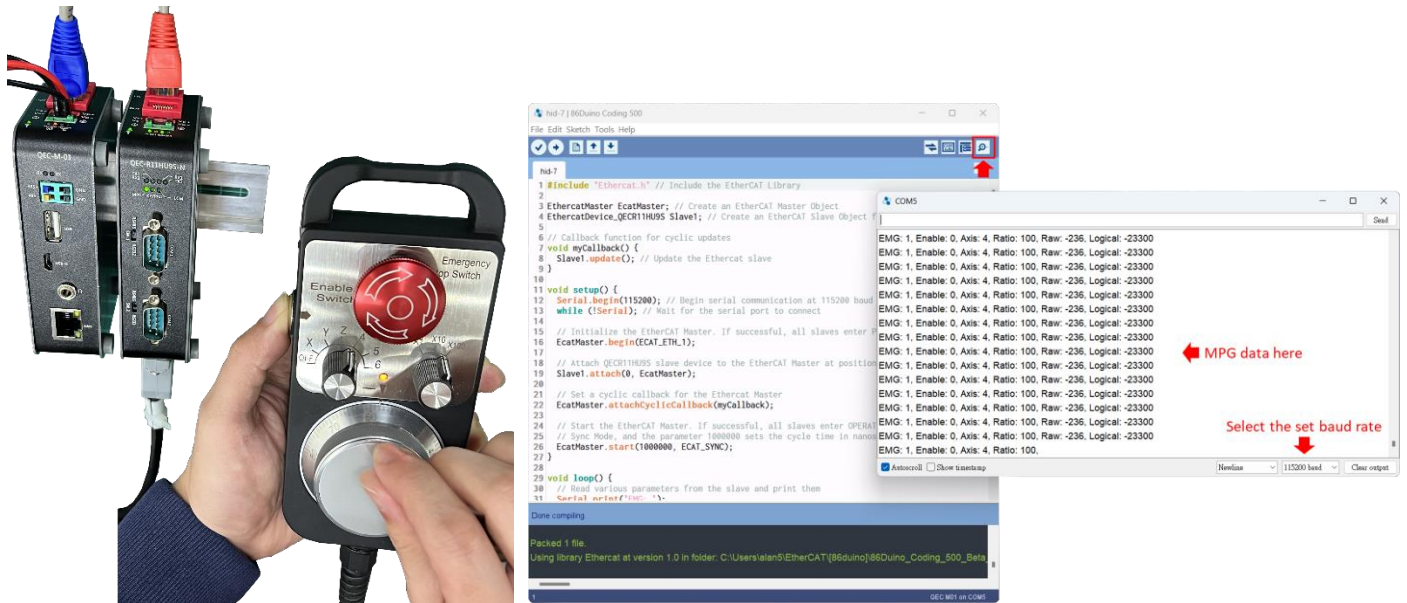
Troubleshooting

Old environment of your QEC-M-01:

If you cannot upload your sketch successfully, please try to update your QEC EtherCAT Master. This update covers the following three updates: Bootloader, EtherCAT Firmware, and EtherCAT Tools.

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.
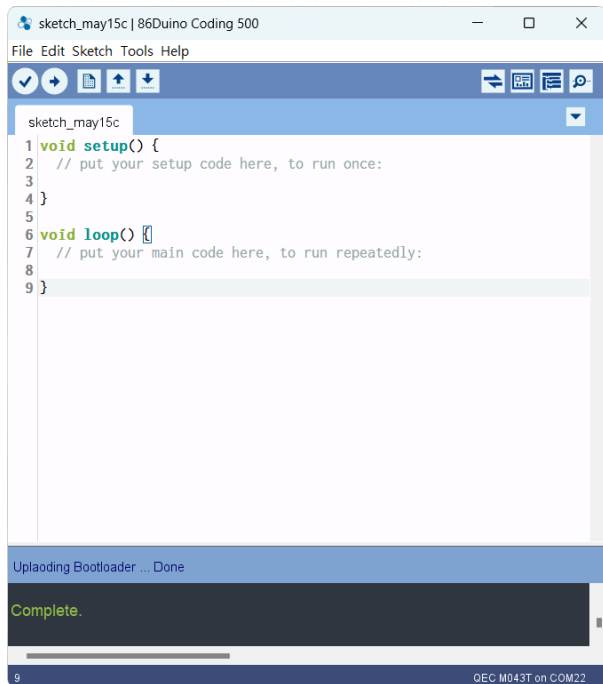
Step 2: Click "Burn Bootloader" button

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:
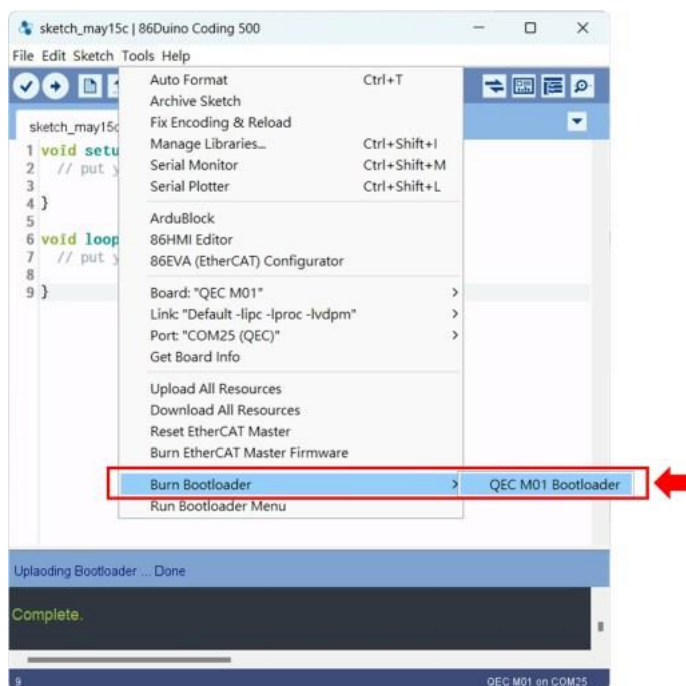
Step 3: Complete the Update

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

For more information and sample requests, please write to info@icop.com.tw, call your nearest ICOP branch, or contact our official global distributor.