Start Guide

QEC-R11D88K : EtherCAT Remote Digital I/O with 86EVA and ArduBlock

86Duino Coding IDE 501

EtherCAT Library

(Version 1.2)

Revision

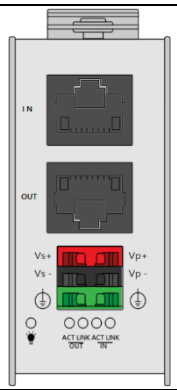
Date	Version	Description
2024/8/23	Version1.0	New release.
2025/3/19	Version1.1	• Change document's title from 'Digital Output' to 'QEC Digital Output – QEC-RXXD88H'. • Change Master to MDevice, Slave to SubDevice.
2025/10/29	Version1.2	Change device to QEC-R11D88K-N.

Preface

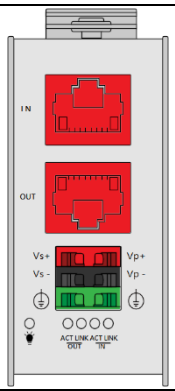
In this guide, we will show you how to use the EtherCAT MDevice **QEC-M-01** and the **QEC-R11D88K** series (EtherCAT Digital I/O module, supports PNP/NPN configuration).

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.



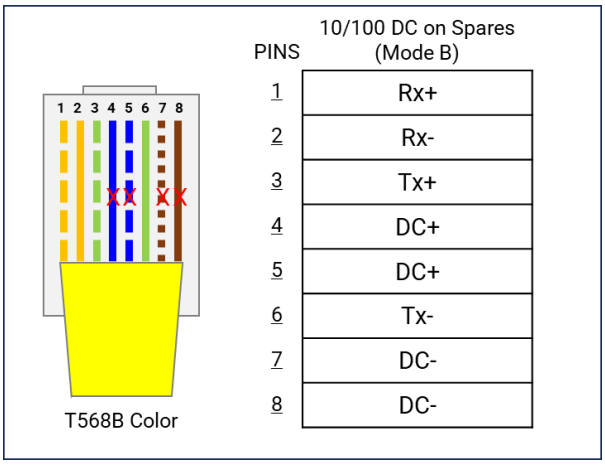
Non-PoE type



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

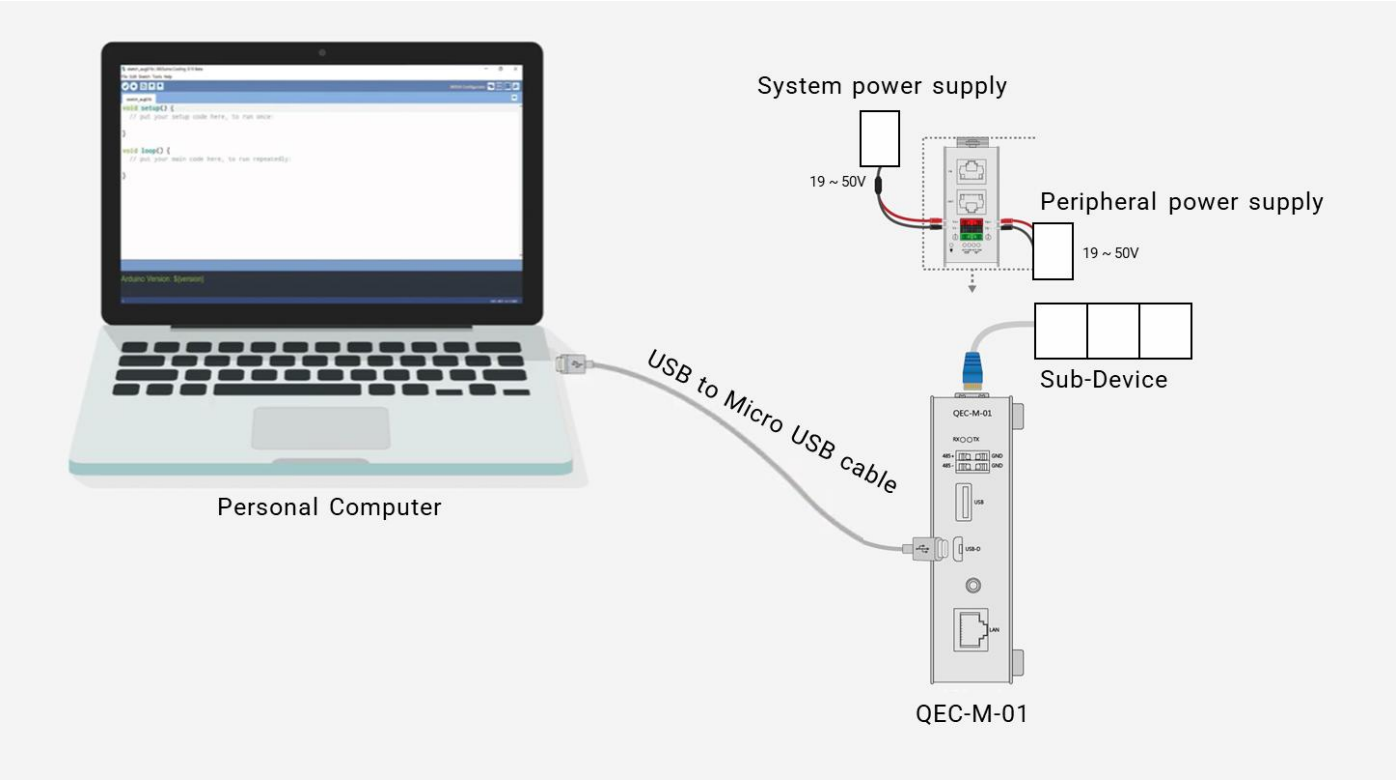
The following devices are used here:

1. QEC-M-01 (EtherCAT MDevice)
2. QEC-R11D88K series (EtherCAT Digital I/O module, supports PNP/NPN configuration).
3. 24VDC power supply & EU-type terminal cable & LAN cable

1.1 QEC-M-01

QEC EtherCAT MDevice.

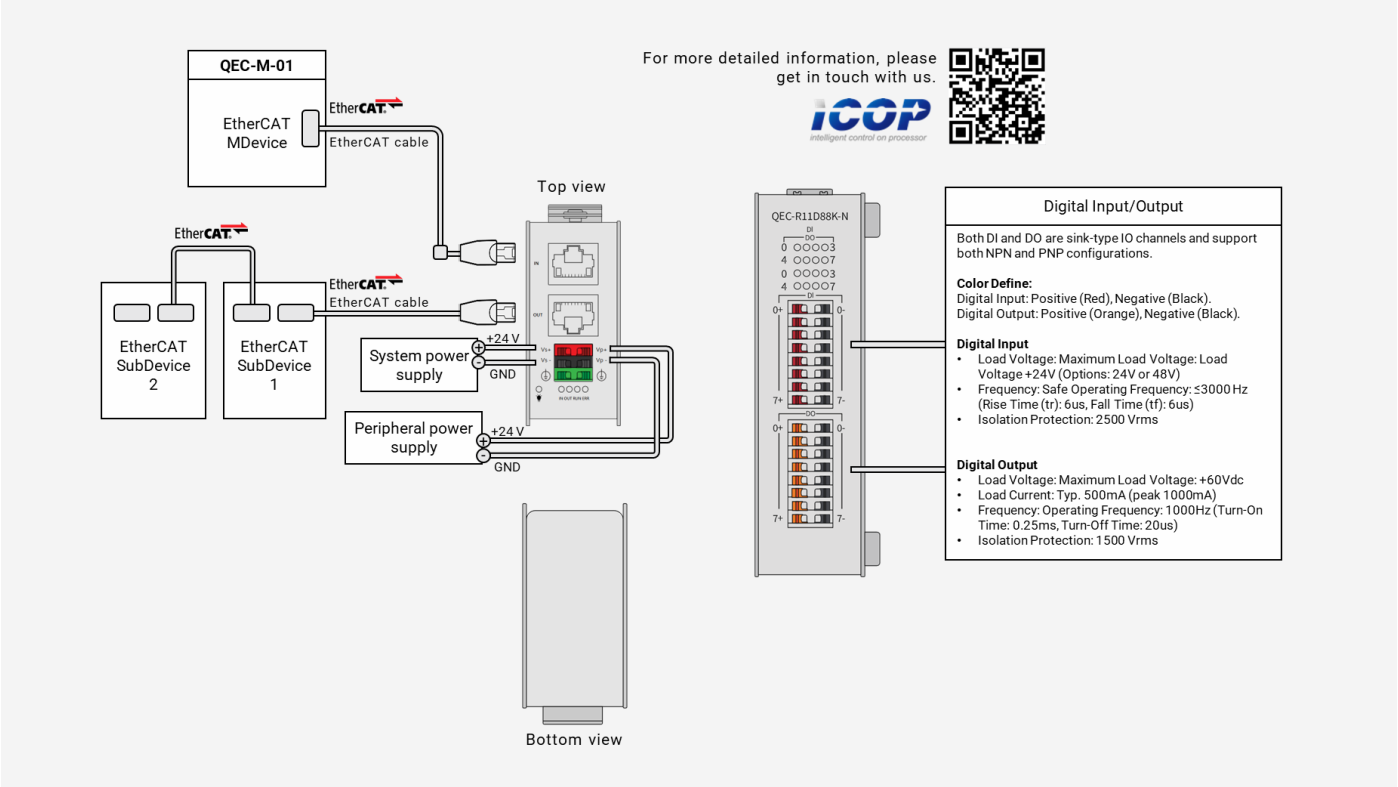
1. Power Supply: Connect to Vs+/Vs- and Vp+/Vp- power supplies via EU terminals for 24V power.
2. EtherCAT Connection: Using the EtherCAT Out port (On the top side) connected to the EtherCAT In port of EtherCAT SubDevice via RJ45 cable.



1.2 QEC-R11D88K

The **QEC-R11D88K** is an EtherCAT SubDevice module with isolated 16-ch Digital I/O (DI8/DO8) and supports PNP/NPN configuration.

The diagram shows a typical wiring example with a QEC MDevice and an EtherCAT network.

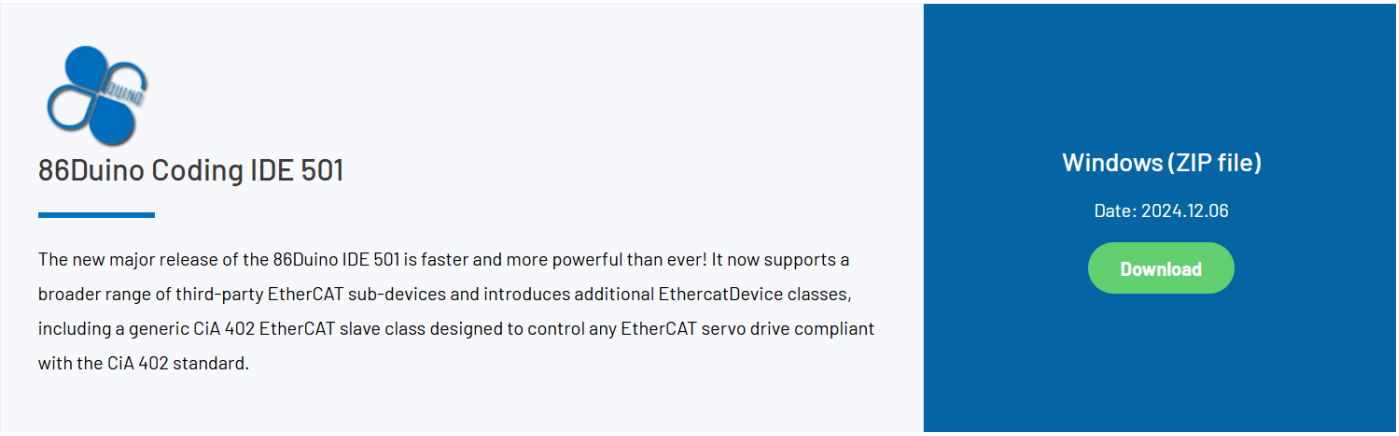


Connections are grouped by function:

- EtherCAT: MDevice → IN; OUT for daisy-chain.
- Power & Grounding:
  - VS+/VS−: system power +24 V/GND
  - VP+/VP−: field I/O power +24 V/GND
- Digital I/O:
  - DI 8 (sink type): supports NPN/PNP; safe op ≤ 3 kHz, tr/tf 6 µs; input +24 V (options 24/48 V).
  - DO 8: up to +60 Vdc, typ. 500 mA (peak 1 A); 1 kHz (Ton 0.25 ms / Toff 20 µs). Add flyback/snubber for inductive loads.
  - Terminal groups: DI00–07, DO00–07, each with + / −.
  - Isolation: DI 2500 Vrms; DO 1500 Vrms.
- Indicators: PWR / RUN / ERR / L/A (see later section).
- Color legend: DI + Red / − Black; DO + Orange / − Black.

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.

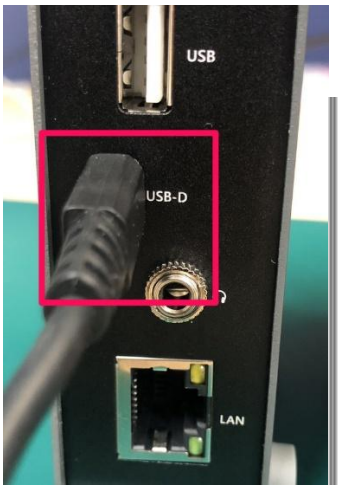
*__Note__: If Windows displays a warning, click Details once and then click the Continue Run button once.

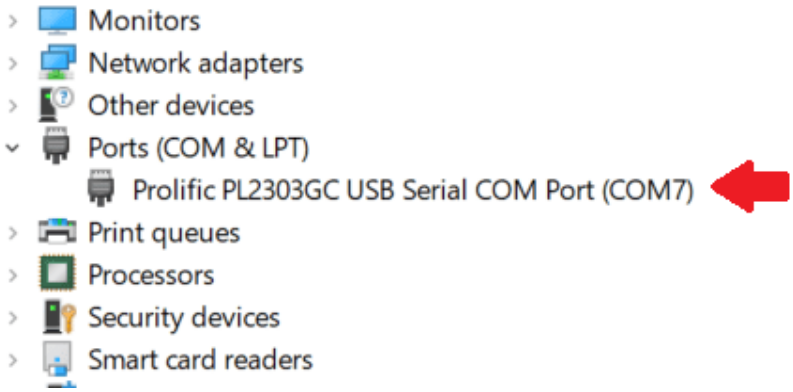
86Duino Coding IDE 501+ looks like below.

3. Connect to PC and set up the environment

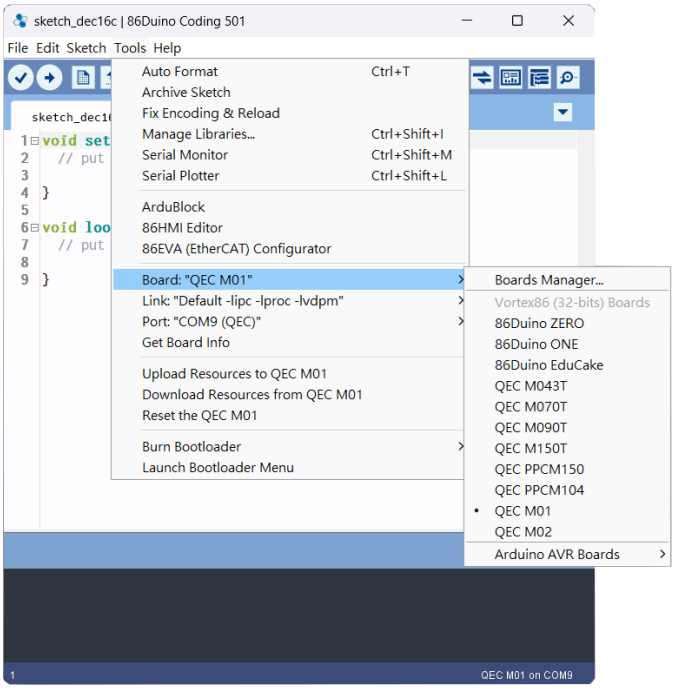
Follow the steps below to set up the environment:

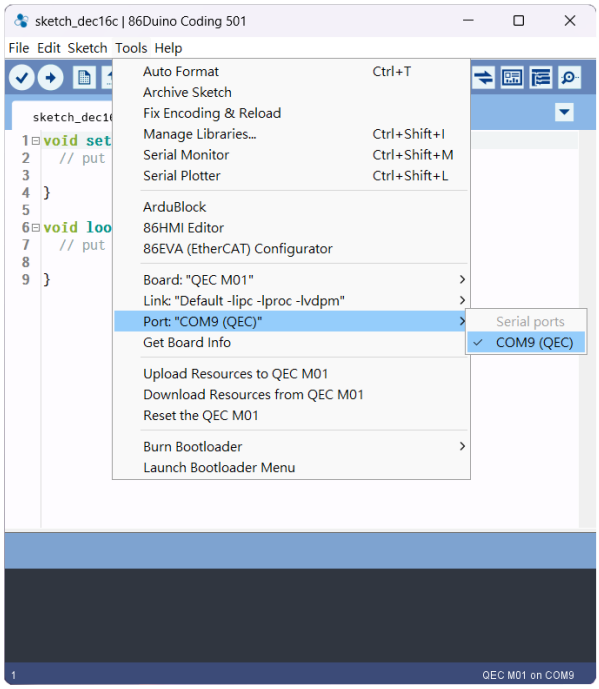
1. Connect the QEC-M-01 to your PC via a Micro USB to USB cable (86Duino IDE installed).
2. Turn on the QEC power.
3. Open “**Device Manager**” (select in the menu after pressing Win+X) ->” **Ports (COM & LPT)**” in your PC and expand the ports; you should see that the “**Prolific PL2303GC USB Serial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

This example shows how to operate the EtherCAT MDevice (QEC-M-01) and the QEC-R11D88K (EtherCAT Digital I/O module, supports PNP/NPN configuration) through the 86Duino IDE's graphical low-code programming tool, **86EVA** and **ArduBlock**.

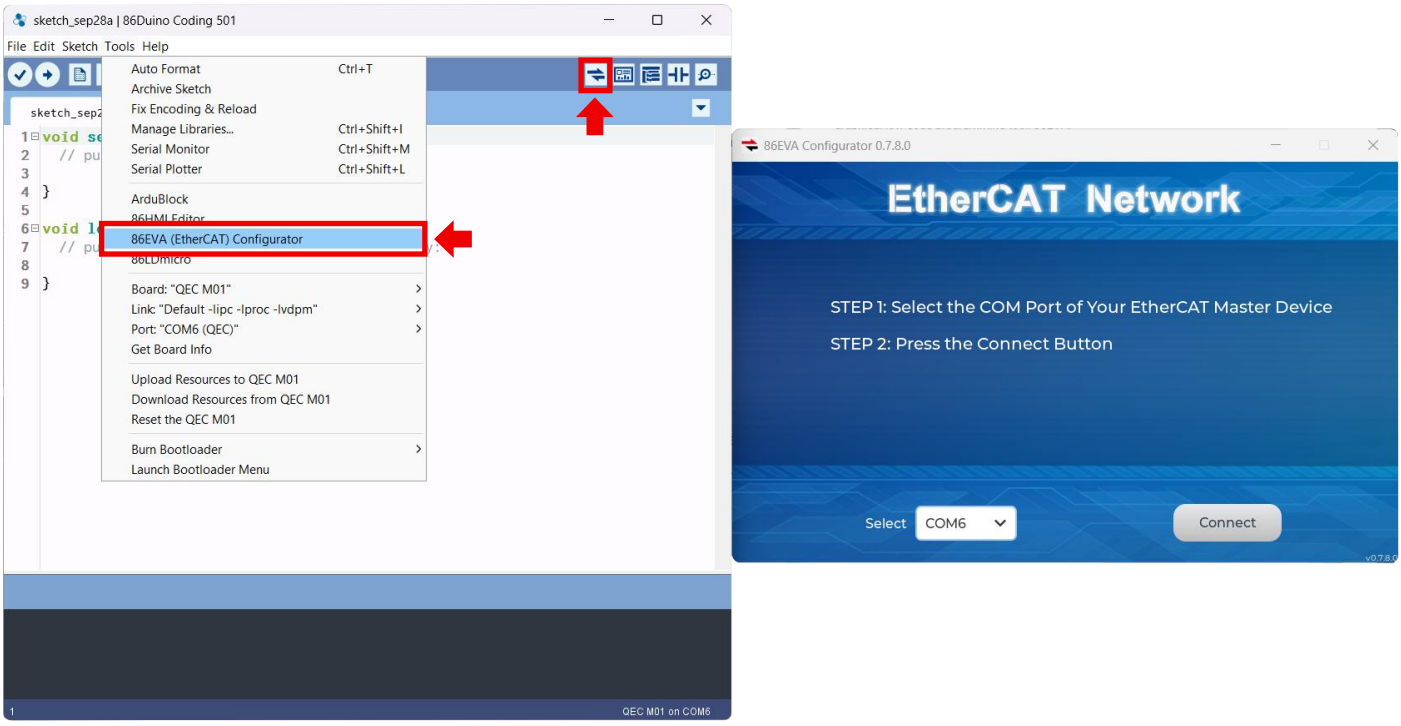
Software Tools Description:

- **86EVA (EVA, EtherCAT-Based Virtual Arduino):**
  is a graphical EtherCAT configuration tool based on the EtherCAT Library in the 86Duino IDE and is one of the development kits for 86Duino.
- **ArduBlock:**
  is a graphical interface for programming and IO control. It is third-party software that belongs to Arduino IDE, developed by David Li, a Shanghai-based creator, and must be attached to the IDE to operate. ArduBlock is a software that converts graphical blocks into code and eventually generates the main program to 86Duino Coding IDE, then compiles and uploads it.

We will present the 86EVA usage in the following section, step by step.

Step 1: Turn on 86EVA and scan

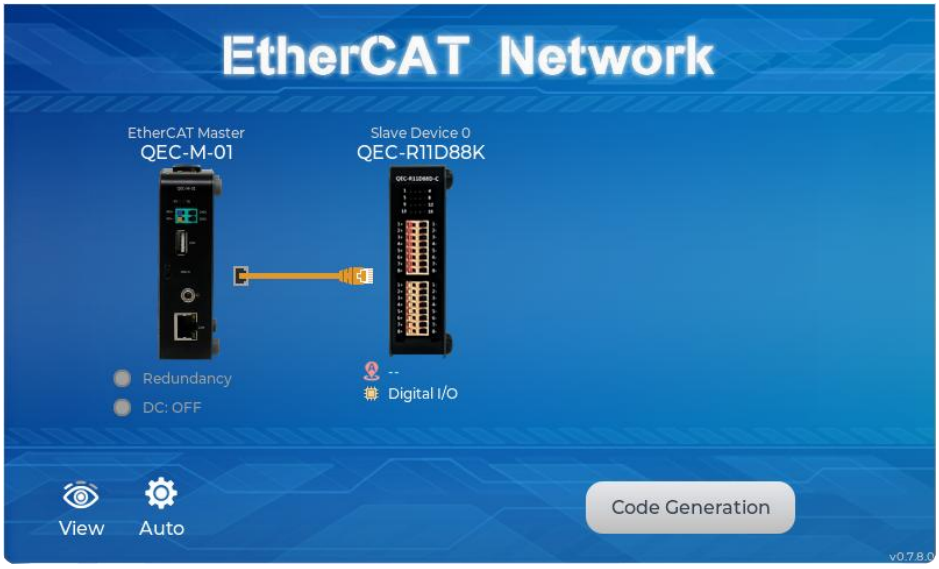
The 86EVA tool can be opened via the following buttons.



Please select the correct COM port and then click the “**Connect**” button.

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

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



Step 2: Set the parameters

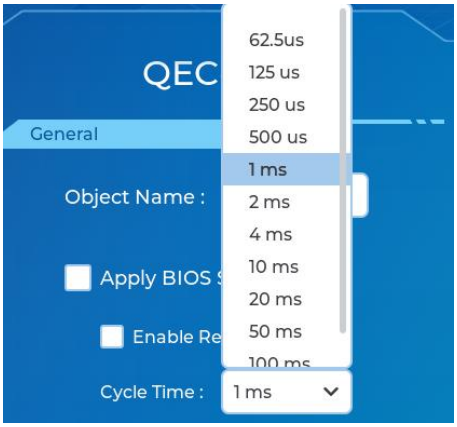
Press twice on the scanned device image to enter the corresponding parameter setting screen.

Step 2.1: QEC-M-01

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

Please check the following configures.

1. Turn off the "**Apply BIOS Settings**".
2. Select "**1ms**" to the Cycle Time.



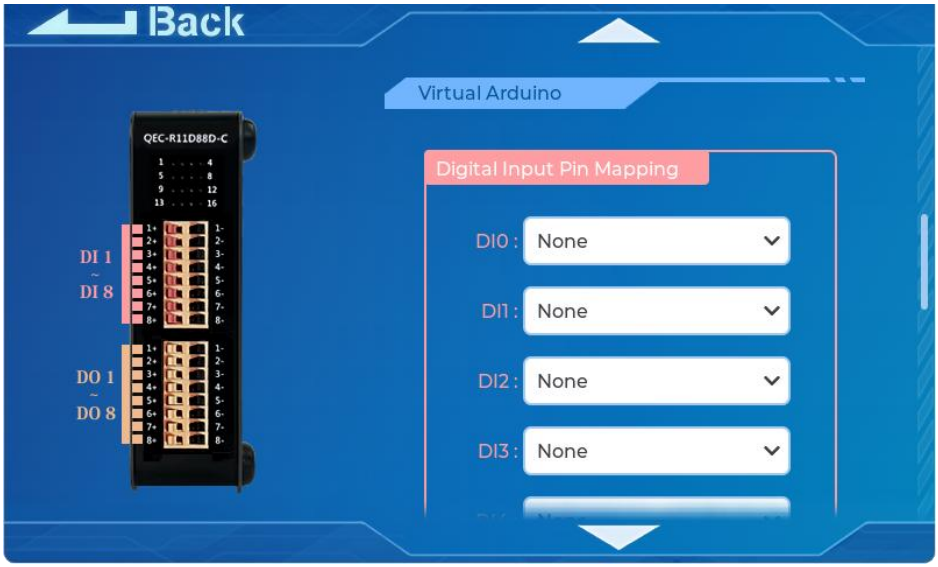
Click "**Back**" in the upper left corner to return.

Step 2.2: QEC-R11D88K

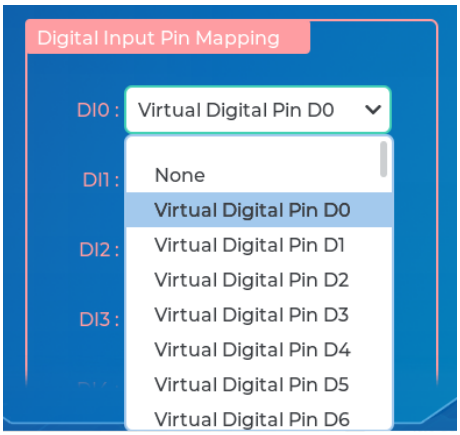
Press twice on the image of the QEC-R11D88K to see the parameter settings.

The page will show the Object Name, Alias Address, Vendor ID, Product Code, Virtual Arduino Mapping, and Virtual Servo Configuration parameters.

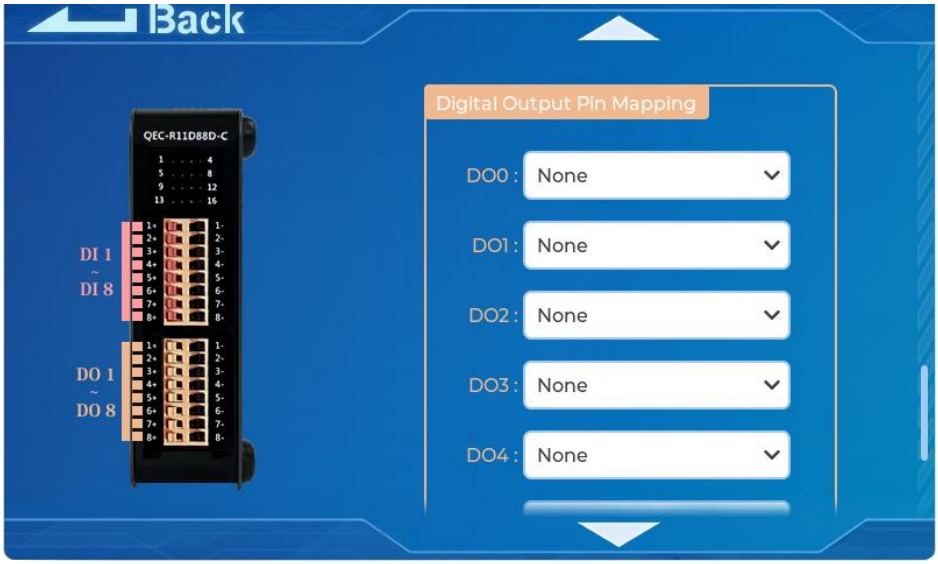
Continue down to the **“Virtual Arduino”** area.
First, we’re in the digital input pin mapping area.



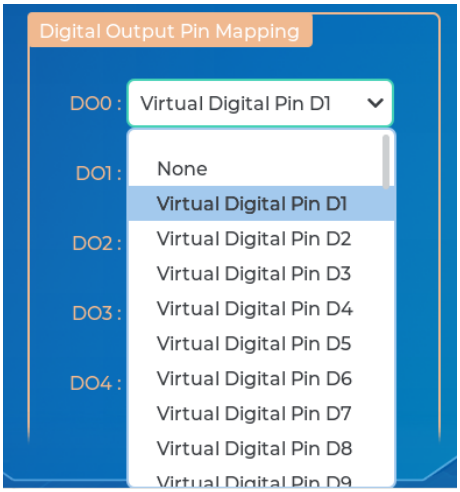
We select “**Virtual Digital Pin D0**” from the “**DI00**” drop-down in “**Digital Input Pin Mapping**”.



Next, we go down to the digital output pin mapping area.

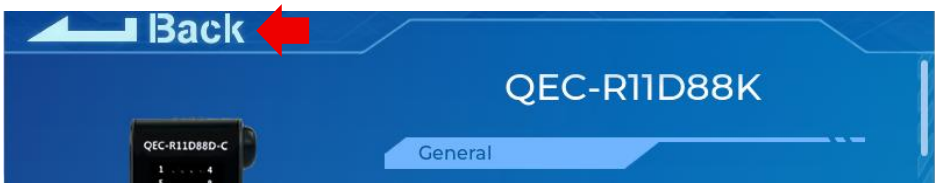


We select “**Virtual Digital Pin D1**” from the “**DO00**” drop-down in “**Digital Output Pin Mapping**”.



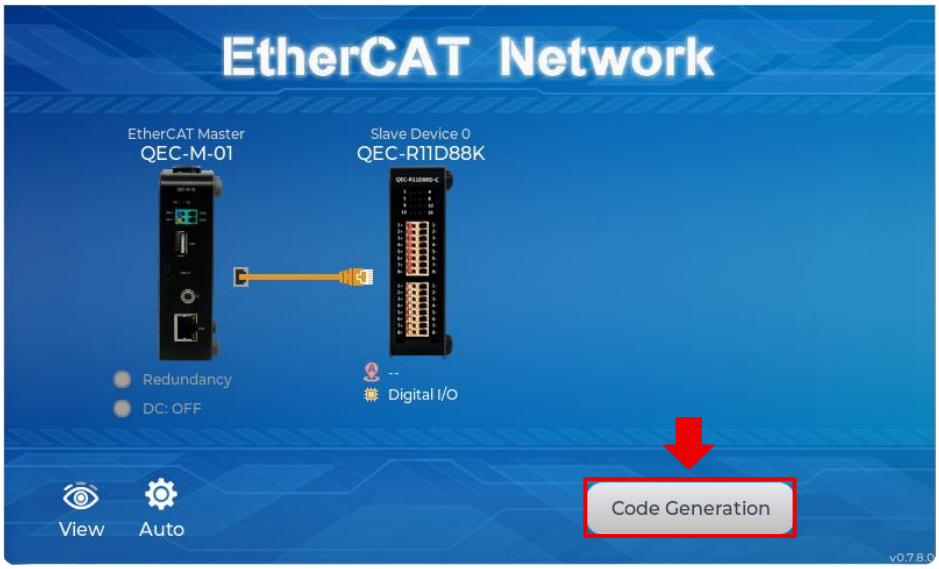
This setting is for ArduBlock tool configuration.

Click "**Back**" in the upper left corner to return.

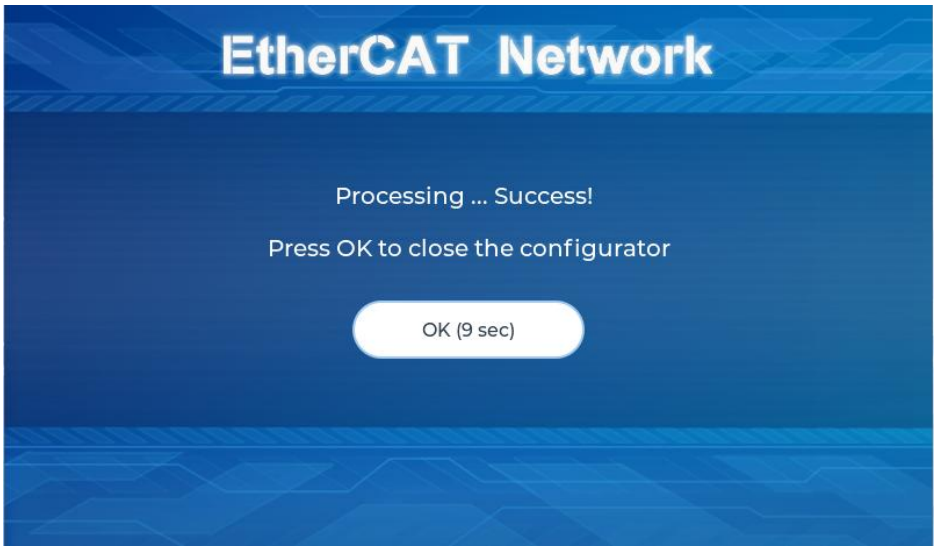


Step 3: Generate the code

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.

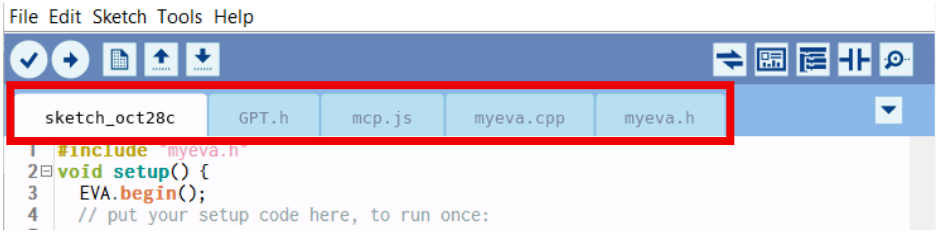


When you're done, double-click the OK button to turn off 86EVA, or it will close in 10 seconds.



The generated code and files are as follows:

- sketch_oct20c: Main Project (.ino, depending on your project name)
- myeva.cpp: C++ program code of 86EVA
- myeva.h: Header file of 86EVA



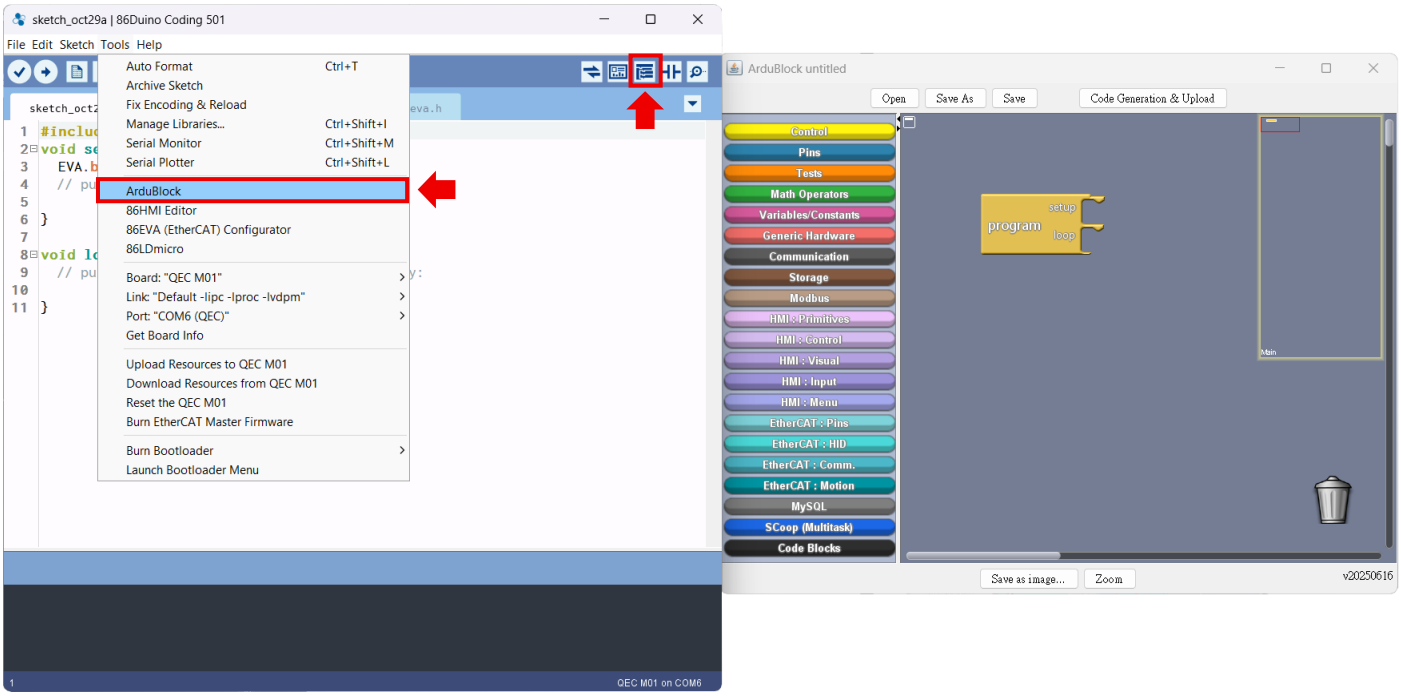
***Additional note:**

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

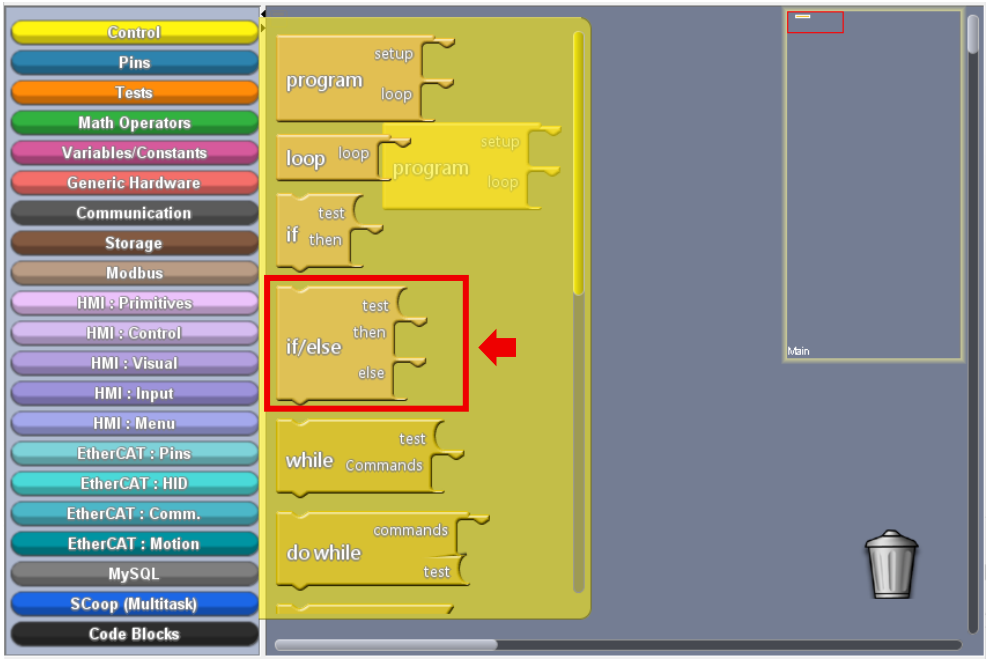
Step 4: Turn on ArduBlock and Setup

Open ArduBlock.

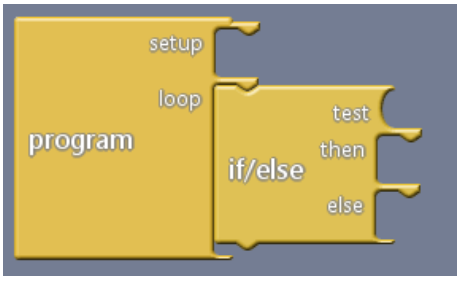


We already configured the EtherCAT DI/DO to the Virtual Digital Pins.
Next, we will read the Virtual Digital Pins DO (which maps to D88K's DI00) and mirror its state to the Virtual Digital Pins D1 (which maps to D88K's DO00).

First, we put the "**if/else**" block from the "**Control**" class into the "**program**" block's "**loop**" area.

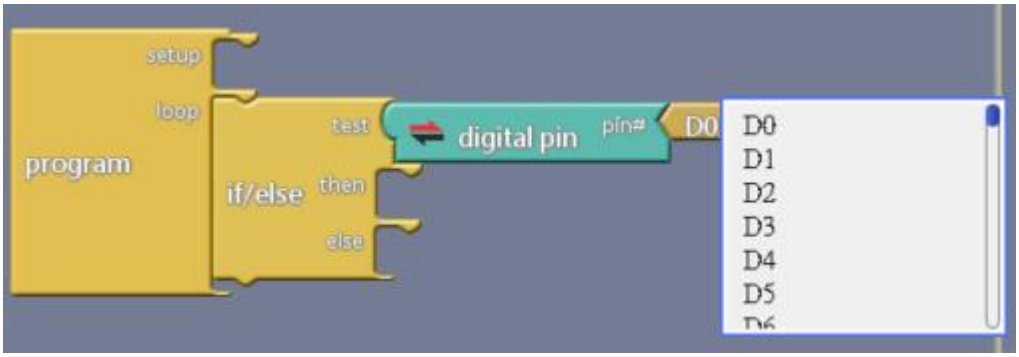


As shown in the following image.



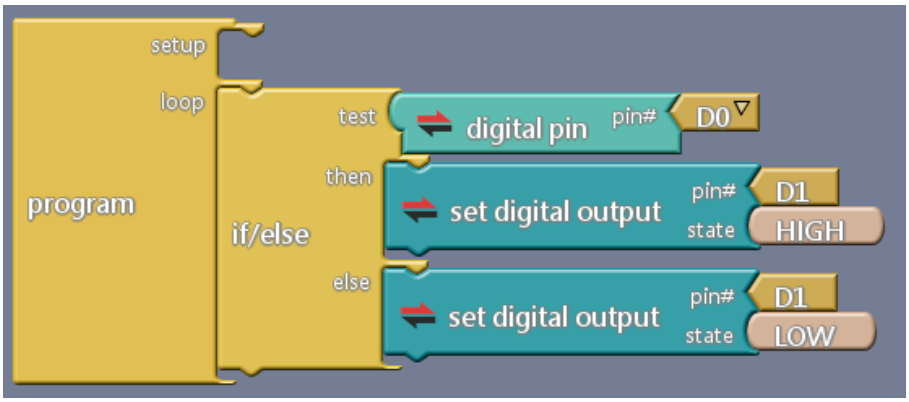
In this example, we want to read and judge the Virtual Digital Pins DO state, so we put the "**digital pin**" block from the "**EtherCAT : Pins**" class in the "**if/else**" block's "**test**" area.

We can select the pin we need in the appropriate blocks, such as D0, D1, or D2, which correspond to the Virtual Digital Pins.
As shown in the following image.



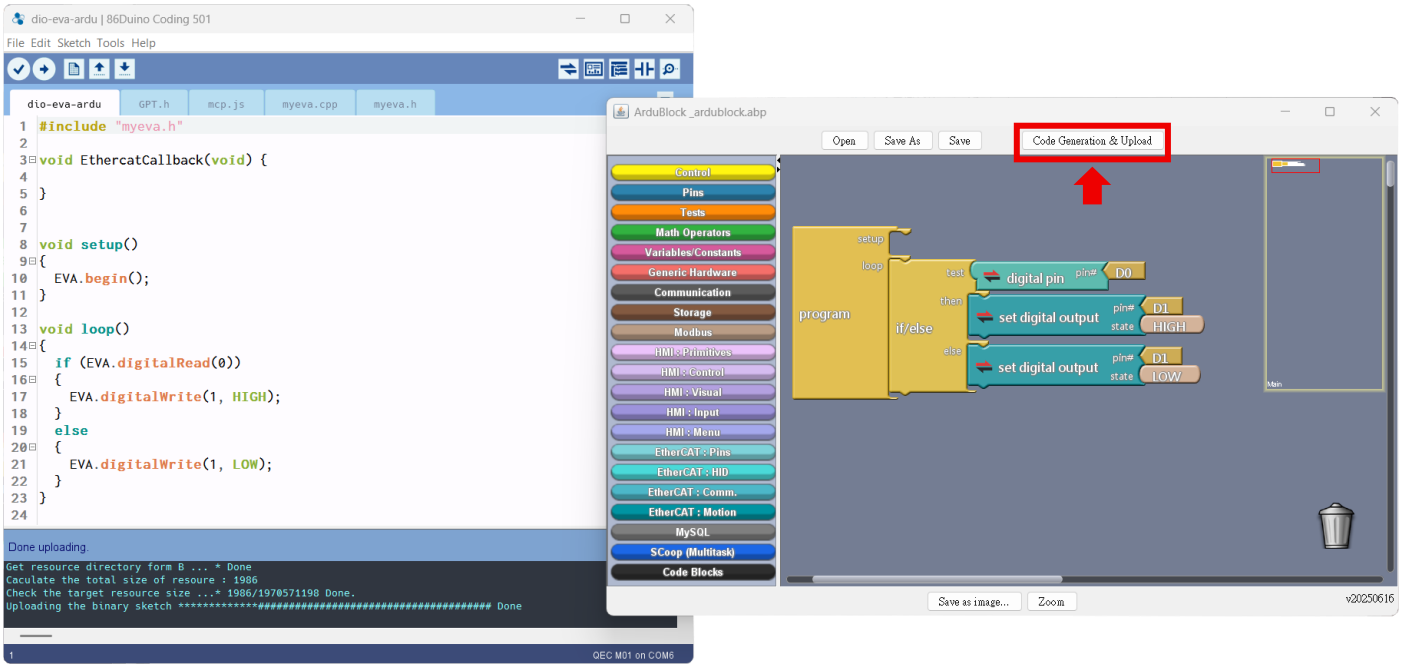
Then, we want to **mirror Virtual Digital Pins D0 state to Virtual Digital Pins D1**.
So, we put the "**set digital output**" block from the "**EtherCAT : Pins**" class in the "**if/else**" block's "**then**" and "**else**" area.

If the state is HIGH, set the Virtual Digital Pins D1 to **HIGH**; otherwise, set it to **LOW**.
As shown in the following image.



This programming block reads the Virtual Digital Pins DO (which map to D88K's DI00) and mirrors their state to the Virtual Digital Pins D1 (which map to D88K's DO00).

After finishing, you can click the "**Code Generate & Upload**" button to generate and upload the code to the QEC-M-01.



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:

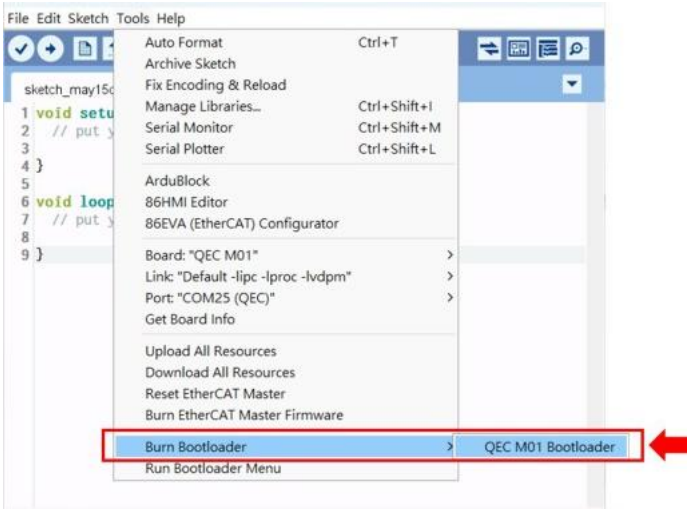
Step 1: Setting up QEC-M

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.

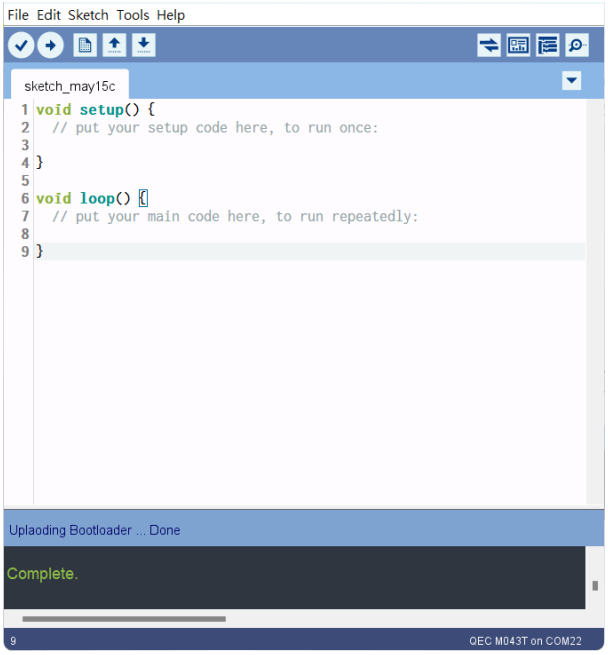
Step 2: Click “Burn Bootloader” button

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.