



User Manual

QEC-M-090T

DM&P Vortex86EX2 Processor

EtherCAT Master System

9" Open Frame Panel PC with 4-wire Resistive Touch Screen

(Revision 1.1)

REVISION

DATE	VERSION	DESCRIPTION
2023/10/26	Version1.0	New Release.
2023/10/31	Version1.1	Updated Arduino Pins and LCD Specifications.

COPYRIGHT

The information in this manual is subject to change without notice for continuous improvement in the product. All rights are reserved. The manufacturer assumes no responsibility for any inaccuracies that may be contained in this document and makes no commitment to update or to keep current the information contained in this manual.

No part of this manual may be reproduced, copied, translated or transmitted, in whole or in part, in any form or by any means without the prior written permission of the ICOP Technology Inc.

©Copyright 2023 ICOP Technology Inc.

Ver.1.1 October 2023

TRADEMARKS ACKNOWLEDGMENT

ICOP® is the registered trademark of ICOP Corporation. Other brand names or product names appearing in this document are the properties and registered trademarks of their respective owners. All names mentioned herewith are served for identification purpose only.

For more detailed information or if you are interested in other ICOP products, please visit our official websites at:

- Global: www.icop.com.tw
- USA: www.icoptech.com
- Japan: www.icop.co.jp
- Europe: www.icoptech.eu
- China: www.icop.com.cn

For technical support or drivers download, please visit our websites at:

- https://www.icop.com.tw/resource_entrance

For EtherCAT solution service, support or tutorials, 86Duino Coding IDE 500+ introduction, functions, languages, libraries, etc. Please visit the QEC website:

- QEC: <https://www.qec.tw/>

This Manual is for the QEC series.

SAFETY INFORMATION

- Read these safety instructions carefully.
- Please carry the unit with both hands and handle it with caution.
- Power Input voltage +19 to +50VDC Power Input (Typ. +24VDC)
- Make sure the voltage of the power source is appropriate before connecting the equipment to the power outlet.
- To prevent the QEC device from shock or fire hazards, please keep it dry and away from water and humidity.
- Operating temperature between -20 to +70°C/-40 to +85°C (Option).
- When using external storage as the main operating system storage, ensure the device's power is off before connecting and removing it.
- Never touch un-insulated terminals or wire unless your power adaptor is disconnected.
- Locate your QEC device as close as possible to the socket outline for easy access and avoid force caused by the entangling of your arms with surrounding cables from the QEC device.
- If your QEC device will not be used for a period of time, make sure it is disconnected from the power source to avoid transient overvoltage damage.

WARNING!



DO NOT ATTEMPT TO OPEN OR TO DISASSEMBLE THE CHASSIS (ENCASING) OF THIS PRODUCT. PLEASE CONTACT YOUR DEALER FOR SERVICING FROM QUALIFIED TECHNICIAN.

Content

Content	iv
Ch. 1 General Information	1
1.1 Introduction	2
1.1.1 QEC-M Systems Diagram	3
1.1.2 Software Support	3
1.2 Specifications	4
1.3 Dimension	6
1.4 Inspection standard for TFT-LCD Panel.....	7
1.5 Ordering Information.....	10
1.5.1 Ordering Part Number:.....	10
Ch. 2 Hardware Installation	11
2.1 CPU Board Outline.....	12
2.2 Connector Summary	13
2.3 I/O Connectors.....	14
J2/J3/J4: USB 2.0 Host.....	14
J5: Micro USB Type-B	14
J6: R6040-LAN1(Secondary EtherCAT).....	14
J7: R6040-LAN2 (Primary EtherCAT).....	14
J8: Giga LAN	14
J9: Power Input Connector	14
J10: I2C0, MCM, GPIO.....	15
J11: MCM, GPIO, COM1 (TTL)	15
J12: GPIO, VCC, GND.....	15
J13: Power source, RESET-	15
J14: ADC/GPIO	15
J15: CAN0 and CAN1 bus.....	15
J16: SPI, RESET-	16
J17: SPI, RESET-, RS485	16
J18: Line-out.....	16
J19: VGA.....	16
J24: MINI PCIe.....	17
J25: SIM Card Holder.....	17
J29A: eMMC Module	17
J29B: eMMC Module	17
2.4 External I/O Overview	18
2.4.1 USB.....	19
2.4.2 Micro USB.....	19
2.4.3 LAN1/LAN2/Giga LAN	20

2.4.4	Arduino pin Assignment	20
2.4.5	10-pin VGA Connector	24
2.4.6	eMMC Module.....	24
2.5	Watchdog Timer.....	24
Ch. 3	Quick Start guide	25
3.1	Package Contents	26
3.2	Hardware Configuration	26
3.3	Software Driver Installation	26
3.4	Set up the QEC-M-090T with 86Duino	27
	Plugin the power supply	27
	Connect the USB cable with PC	27
	Make sure your device is detected	28
	Write your first sketch	28
	Upload your first sketch!	30
3.5	Building an HMI on the QEC-M-090T with 86Duino	31
	Instruction	31
	Uploading the LVGL Example	31
	Using the Graphical HMI editor: 86HMI.....	34
Ch. 4	Software Function.....	36
4.1	Software Description	37
4.2	EtherCAT Function List	37
	EthercatMaster Class Functions	37
	EthercatDevice Class General Functions.....	38
	EthercatDevice_Generic Class Functions	40
4.3	Additional Resources	40
Warranty	41

Ch. 1

General Information

[1.1 Introduction](#)

[1.2 Specifications](#)

[1.3 Dimensions](#)

[1.4 Inspection standard for TFT-LCD Panel](#)

[1.5 Ordering Information](#)

1.1 Introduction

QEC-M-090T is an EtherCAT master with a real-time, reliable, and synchronous control equipped with a 9-inch TFT LCD. The industrial Arduino makes it an easy-to-develop, stable, and economical automation system, allowing users to develop EtherCAT with UI quickly.

The QEC master is highly compatible with third-party EtherCAT devices for communication, such as servo, I/O, etc. For taking full advance of EtherCAT, It supports PDO, CoE, FoE, DC, and EtherCAT cable redundancy to use other EtherCAT slaves flexibly. The QEC master has precise synchronization (min.125 μ s), and its 86Duino IDE provides less than 1 μ s jitter time in the minimum cycle time; it could apply to highly synchronized and precision automatic applications, like motion control and I/O control. (Read More: [EtherCAT Master's Benchmark – QEC](#))

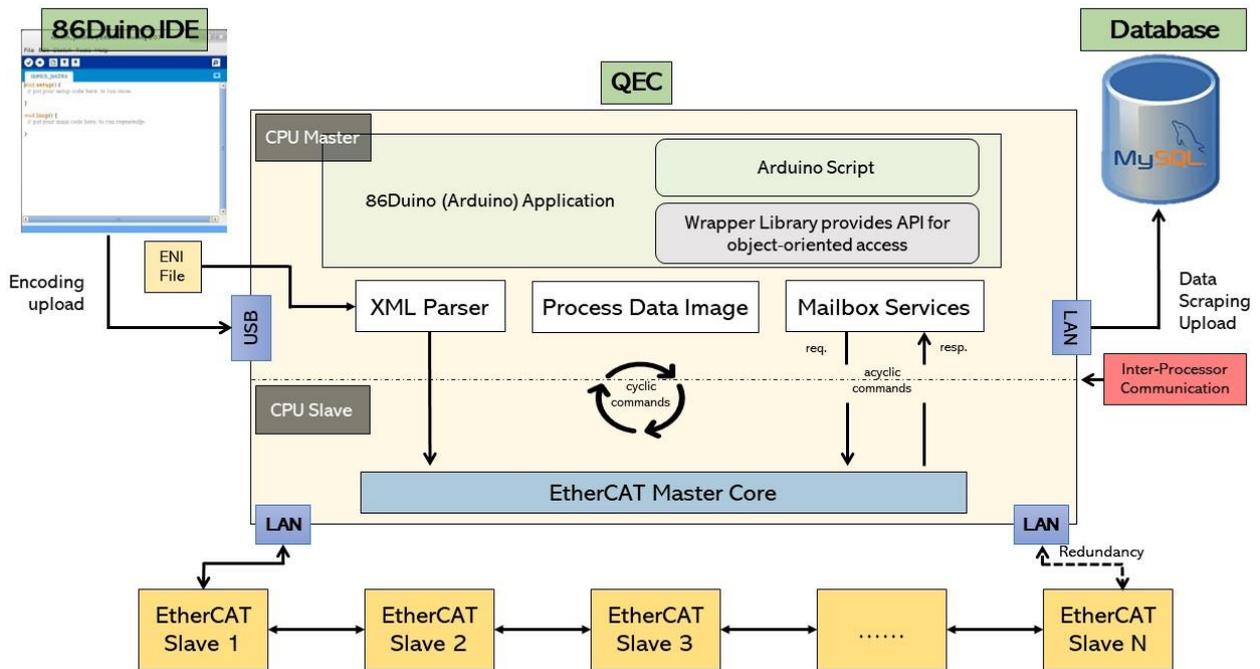
QEC-M-090T has a built-in high endurance 2GB SLC eMMC, designed to provide a stable and reliable operating system. Users can upload the developed executable files and required images or data, such as HMI images, to the QEC-M-090's SLC via the 86Duino IDE without affecting the performance of the master system. Besides, 86Duino IDE also integrates with the LVGL library to provide an advanced and intuitive approach to user interface design on QEC-M-090T.

QEC-M-090T can also monitor hardware information on temperature, voltage, and current. These features allow users to track the system's carbon footprint and estimate its lifespan.

QEC-M-090T's dimension is 245 x 152.2 x 32.55 mm, with its open frame design for easy integration and customization. Operating temperature is from -20°C to +70°C, and with an extended option of -40 to +85°C, this EtherCAT Master remains steadfast even in the most challenging conditions. QEC-M-090T has two networks for EtherCAT Cable redundancy and one Giga LAN for external network connection. It also offers unrivaled connectivity, featuring 3 USB ports, a MicroUSB port (debug/upload only), VGA output, MiniPCIe slot, and full Arduino function Pins (GPIO, PWM, SPI, I2C, CAN, etc.); All provide an off-the-shelf API to use.

These interfaces open up a world of possibilities for integrating various devices and peripherals into your automation network, providing flexibility and adaptability for your unique requirements.

1.1.1 QEC-M Systems Diagram



1.1.2 Software Support

The 86Duino integrated development environment (IDE) software makes it easy to write code and upload it to QEC-M. It runs on Windows, Mac OS X, and Linux. The environment is written in Java and based on Arduino IDE, Processing, DJGPP, and other open-source software.



1.2 Specifications

CPU BOARD SPECIFICATIONS

CPU	DM&P Vortex86EX2 Processor, Master 533MHz/Slave 400MHz	
Memory	512MB/1GB DDRIII Onboard	
Storage	32MB SPI Flash /2GB SLC eMMC	
LAN	1Gbps Ethernet RJ45 x1 10/100Mbps Ethernet RJ45 x2 for EtherCAT	
Expansion	Mini PCIe x 1 with Micro SIM Card Holder	
I/O Connector	2.54mm 2-pin header for Power Connector 1.25mm 4-pin header for EXT I2C TFT Driver 1.25mm 4-pin wafer for Line-Out Power DC Input/Output Connector x1 Micro USB (Type-B) x1 (Upload/Debug only)	VGA Connector (10-pin) x1 Micro SIM Card Holder x1 MiniPCIe slot x1 USB 2.0 Host x3 RJ45 x3
Arduino Compatible Connector	2.54mm 10-pin female header for I2C0, MCM, GPIO 2.54mm 8-pin female header for MCM, GPIO, COM1 (TTL) 2.54mm 8-pin female header for Power source 2.54mm 6-pin female header for ADC/GPIO 2.54mm 6-pin female header for GPIO, VCC and GND 2.54mm 6-pin female header for CAN0 and CAN1 bus 2.54mm 10-pin header for SPI, RESET- 2.54mm 10-pin header for SPI, RESET-, RS485	
Protocol	EtherCAT (EtherCAT Master Functions: PDO, CoE, DC, Cable-redundancy, etc.)	
Ethernet Standard	IEEE 802.3	
Control Cycle Time	125 μ s (min.)	
Software Support	86Duino Coding IDE 500+ (The environment is written in Java and based on Arduino IDE, Processing, DJGPP, and other open-source software)	

*MCM signal is equivalent to Arduino's PWM signal.

MECHANICAL & ENVIRONMENT

Power Connector	6-pin Power Input /Output
Power Requirement	+19 to +50VDC Power Input (Typ. +24VDC)
Power Consumption	8 Watt
Operating Temp.	-20 to +70°C/-40 to +85°C (Option)
Storage Temp.	-30 ~ +85°C
Operating Humidity	0% ~ 90% Relative Humidity, Non-Condensing
Dimension	245 x 152.2 x 32.55mm
Weight	1.07Kg
Internal Monitoring	Temperature, Voltage, Current

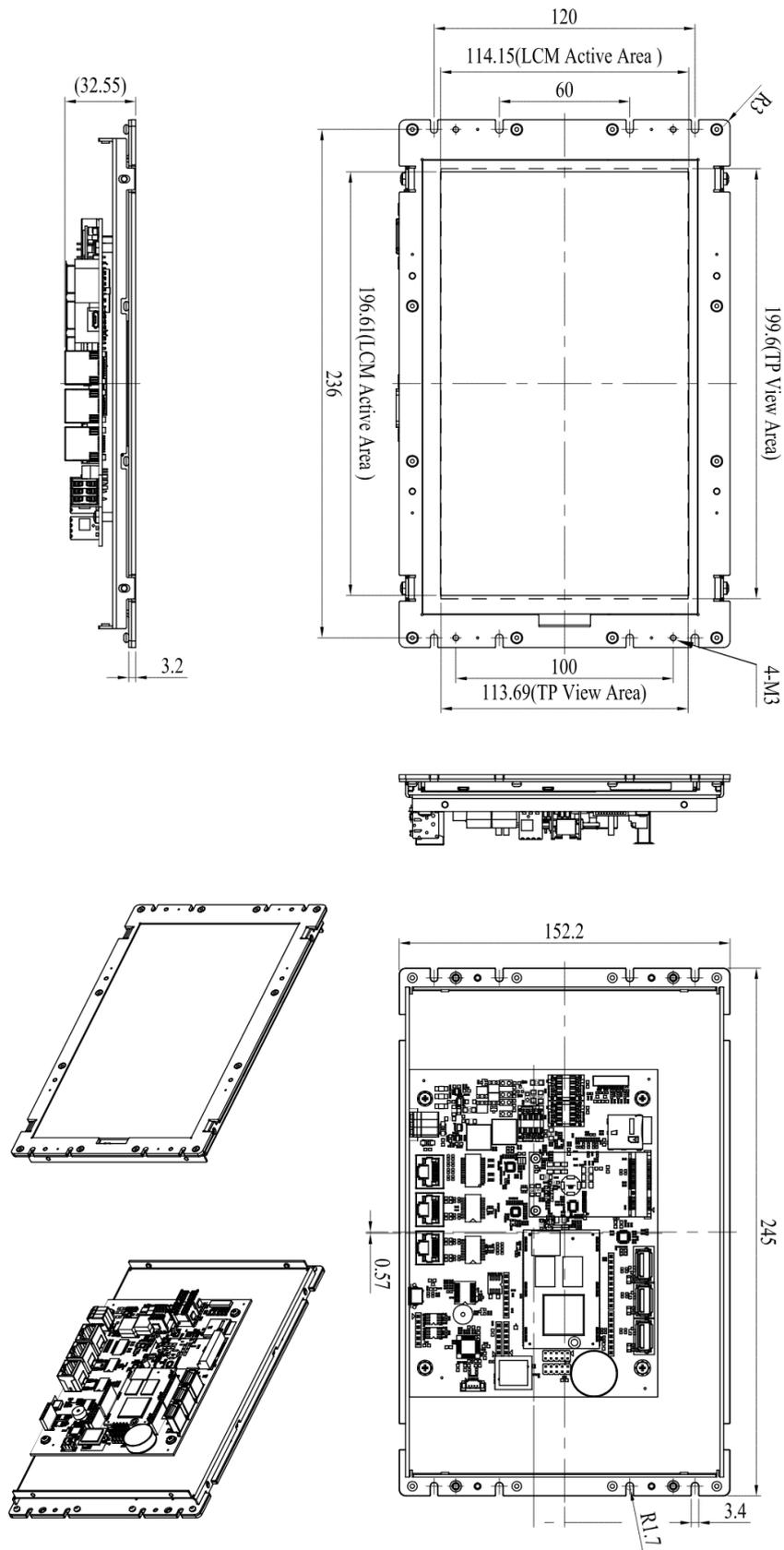
LCD SPECIFICATIONS

Display Type	9" WVGA TFT LCD
Backlight Unit	LED
Display Resolution	800(W) x 480(H)
Contrast Ratio	500: 1
Display Color	16.7M
Active Area	198 (W) x 111.696 (H) mm
Viewing Angle	Vertical 120°, Horizontal 140°
Brightness (cd/m ²)	300 nits
Backlight Lifetime	20,000 hrs

TOUCHSCREEN

Type	Analog Resistive
Resolution	Continuous
Transmittance	80%
Controller	PS/2 interface
Durability	1 million

1.3 Dimension



(Unit: mm)

1.4 Inspection standard for TFT-LCD Panel

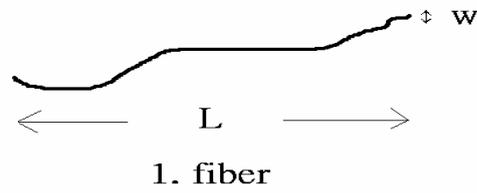
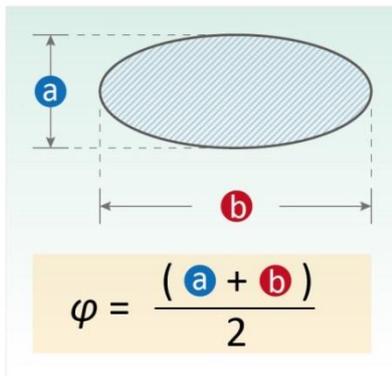
DEFECT TYPE			LIMIT				Note	
VISUAL DEFECT	INTERNAL	SPOT	$\varphi < 0.15\text{mm}$		Ignore		Note1	
			$0.15\text{mm} \leq \varphi \leq 0.5\text{mm}$		$N \leq 4$			
			$0.5\text{mm} < \varphi$		N=0			
		FIBER	$0.03\text{mm} < W \leq 0.1\text{mm}, L \leq 5\text{mm}$		$N \leq 3$		Note1	
			$1.0\text{mm} < W, 1.5\text{mm} < L$		N=0			
		POLARIZER BUBBLE	$\varphi < 0.15\text{mm}$		Ignore		Note1	
			$0.15\text{mm} \leq \varphi \leq 0.5\text{mm}$		$N \leq 2$			
			$0.5\text{mm} < \varphi$		N=0			
		Mura	It' OK if mura is slight visible through 6%ND filter					
ELECTRICAL DEFECT	BRIGHT DOT	A Grade			B Grade			
		C Area	O Area	Total	C Area	O Area	Total	Note3
		$N \leq 0$	$N \leq 2$	$N \leq 2$	$N \leq 2$	$N \leq 3$	$N \leq 5$	Note2
	DARK DOT	$N \leq 2$	$N \leq 3$	$N \leq 3$	$N \leq 3$	$N \leq 5$	$N \leq 8$	
	TOTAL DOT	$N \leq 4$			$N \leq 5$	$N \leq 6$	$N \leq 8$	Note2
	TWO ADJACENT DOT	$N \leq 0$	$N \leq 1 \text{ pair}$	$N \leq 1 \text{ pair}$	$N \leq 1 \text{ pair}$	$N \leq 1 \text{ pair}$	$N \leq 1 \text{ pair}$	Note4
	THREE OR MORE ADJACENT DOT	NOT ALLOWED						
	LINE DEFECT	NOT ALLOWED						

(1) One pixel consists of 3 sub-pixels, including R, G, and B dot. (Sub-pixel = Dot)

(2) Little bright Dot acceptable under 6% ND-Filter.

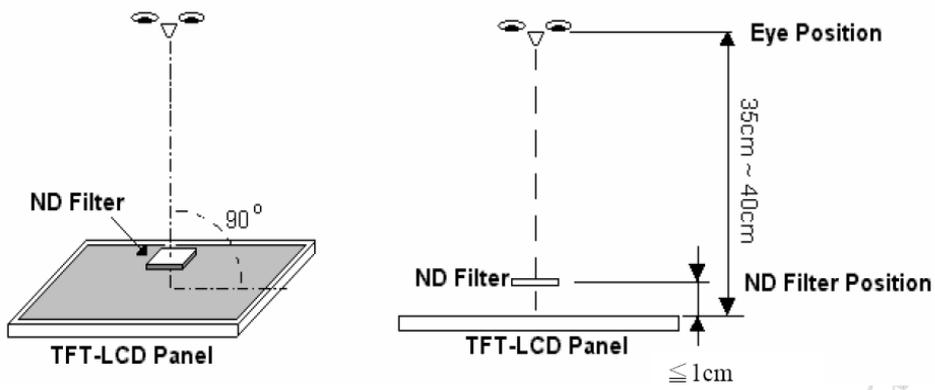
(3) If require G0 grand (Total dot $N \leq 0$), please contact region sales.

[Note 1] W: Width[mm]; L: Length[mm]; N: Number; φ : Average Diameter.

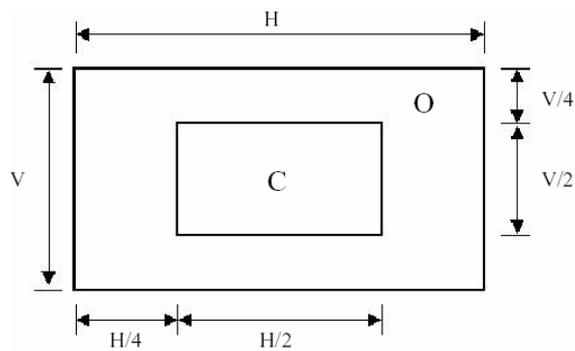


(a) White / Black Spot (b) Polarizer Bubble

[Note 2] Bright dot is defined through 6% transmission ND Filter as following.

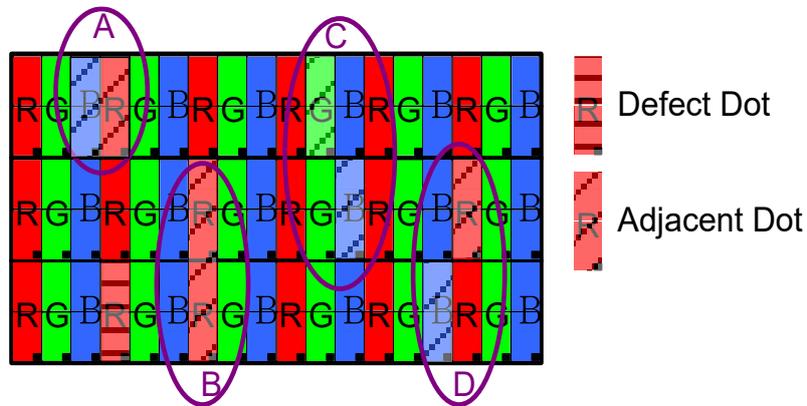


[Note 3] Display area



C Area: Center of display area **O Area:** Outer of display area

[Note 4] Judge the defect dot and the adjacent dot as following. Allow below (as A, B, C and D status) adjacent defect dots, including bright and dark adjacent dot. And they will be counted 2 defect dots in total quantity.



The defects that are not defined above and considered to be problem shall be reviewed and discussed by both parties.

Defects on the Black Matrix, out of Display area, are not considered as a defect or counted.

1.5 Ordering Information

Type	LCD size	(Below is the customization function, the unfilled fields do not need to be filled in; if the customer does not require, it will be directly shipped standard material number, such as QEC-M-090T)							
		PoE	-	Feature		-	Wide Temp.	-	Coating
				Memory	Storage				
QEC-M	090T	<u>X</u>		<u>X</u>	<u>X</u>		<u>X</u>		<u>X</u>

1. Type: Code 1~4

M: EtherCAT Master

2. LCD size: Code 5~8

090T: 9-inch TFT LCD with Restive touchscreen

3. PoE: Code 9

P: RJ45 PoE Device, Red Plastic Housing

None or E: RJ45 w/o power, Black Plastic Housing

(Standard: None)

4. Feature: Code 10~11

X (Memory): G: 1G DDRIII / M: 512M DDRIII (Standard: 512MB)

X (Storage): 1, 2, 4: eMMC size (Standard: 2G)

5. Wide Temp.: Code 12

X: -40 to +85°C / R: -20 to +70°C (Standard: -20 to +70°C)

6. Coating: Code 13

C: Yes / N: Normal

QEC-M-090T X - XX - X - X

1.5.1 Ordering Part Number:

- **QEC-M-090T**: Vortex86EX2 Processor 533MHz-based EtherCAT Master System with 9-inch LCD
- **QEC-M-090TP**: Vortex86EX2 Processor 533MHz-based EtherCAT Master System with 9-inch LCD/PoE
- **QEC-M-090TP-G2-X-C**: Vortex86EX2 Processor 533MHz-based EtherCAT Master System with 9-inch LCD/PoE/1G DDRIII Memory/2G eMMC Storage/Wide Temp./Coating

Ch. 2

Hardware Installation

[2.1 CPU Board Outline](#)

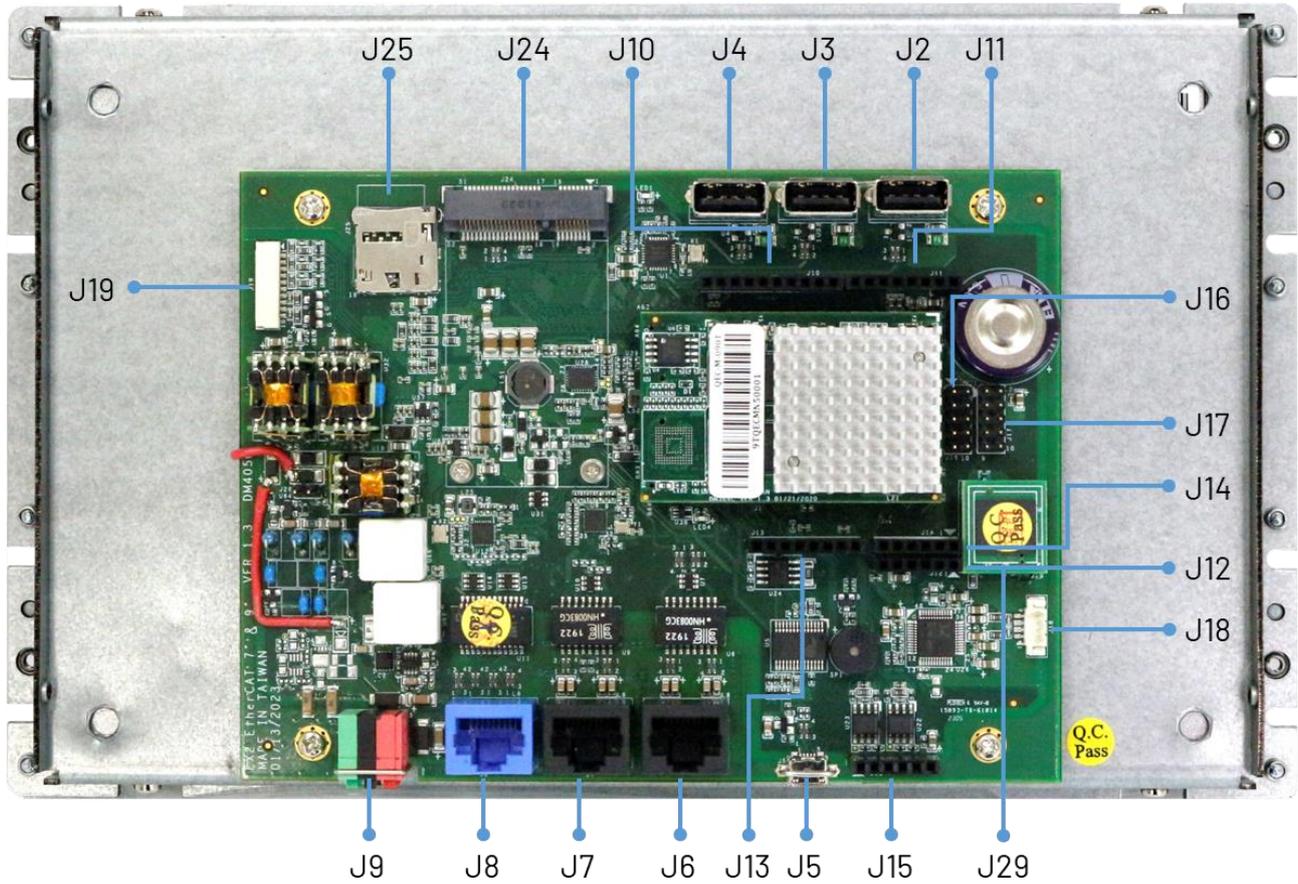
[2.2 Connector Summary](#)

[2.3 Connector Pin Assignments](#)

[2.4 External I/O Overview](#)

[2.5 Watchdog Timer](#)

2.1 CPU Board Outline



2.2 Connector Summary

No.	Description	Type of Connections	Pin #
J2/J3/J4	USB 2.0 Host	1.25mm 5-pin wafer	6-pin
J5	Micro USB	Micro USB Type-B	11-pin
J6	R6040-LAN1	1.25mm 8-pin wafer	8-pin
J7	R6040-LAN2	1.25mm 8-pin wafer	8-pin
J8	Giga LAN	1.25mm 8-pin wafer	8-pin
J9	Power Input Connector	2.00mm 6-pin wafer	6-pin
J10	I2C0, MCM, GPIO	2.54mm female header	10-pin
J11	MCM, GPIO, COM1(TTL)	2.54mm female header	8-pin
J12	GPIO, VCC, GND	2.54mm female header	6-pin
J13	Power source, RESET-	2.54mm female header	8-pin
J14	ADC/GPIO	2.54mm female header	6-pin
J15	CAN0 and CAN1 bus	2.54mm female header	6-pin
J16	SPI, RESET-	2.54mm header	10-pin
J17	SPI, RESET-, RS485	2.54mm header	10-pin
J18	Line-out	1.25mm 4-pin wafer	4-pin
J19	VGA (Reserved and debug used)	1.25mm 10-pin VGA	10-pin
J24	MINI PCIe	Mini PCIe Slot	52-pin
J25	SIM Card Holder	Micro SIM Socket	10-pin
J29A	eMMC TINY MODULE	1.27mm 6-pin header	6-pin
J29B	eMMC TINY MODULE	1.27mm 4-pin header	4-pin

2.3 I/O Connectors

For the external I/O details, visit [Ch2.4](#).

J2/J3/J4: USB 2.0 Host

Pin#	Signal Name
1	VCC
2	MUSBD-
3	MUSBD+
4	GND
5	FGND1

J5: Micro USB Type-B

Pin#	Signal Name
1	USBDET
2	USBDEV2-
3	USBDEV2+
4	-
5	GND

J6: R6040-LAN1 (Secondary EtherCAT)

Pin#	Signal Name
L1	TD+
L2	TD-
L3	RO+
L4	OUT_Us
L5	OUT_Up
L6	RO-
L7	VsGND
L8	VpGND

* L4, L5, L7, L8 pins are option, for RJ45 Power IN/OUT.

J7: R6040-LAN2 (Primary EtherCAT)

Pin#	Signal Name
L1	TD+
L2	TD-
L3	RO+
L4	IN_Us
L5	IN_Up
L6	RO-
L7	VsGND
L8	VpGND

* L4, L5, L7, L8 pins are option, for RJ45 Power IN/OUT.

J8: Giga LAN

Pin#	Signal Name
L1	GTX+
L2	GTX-
L3	GRX+
L4	GTXC+
L5	GTXC-
L6	GRX-
L7	GRXD+
L8	GRXD-

J9: Power Input Connector

Pin#	Signal Name
1	FGND
2	FGND
3	VpGND
4	VsGND
5	Vp
6	Vs

J10: I2C0, MCM, GPIO

Pin#	Signal Name
1	GP30
2	MCM-9
3	MCM-10
4	MCM-11
5	GP31
6	MCM-13
7	GND
8	-
9	I2C0_SDA
10	I2C0_SCL

*MCM signal is equivalent to Arduino's PWM signal.

J11: MCM, GPIO, COM1 (TTL)

Pin#	Signal Name
1	RXD1#
2	TXD1#
3	GP00
4	MCM-3
5	GP02
6	MCM-5
7	MCM-6
8	GP05

*MCM signal is equivalent to Arduino's PWM signal.

J12: GPIO, VCC, GND

Pin#	Signal Name
1	-
2	GP35
3	GP36
4	GP37
5	GND
6	VCC

J13: Power source, RESET-

Pin#	Signal Name
1	VCC
2	GND
3	GND
4	VCC
5	VCC3
6	RESET-
7	VCC3
8	-

J14: ADC/GPIO

Pin#	Signal Name
1	GP57ADC
2	GP56ADC
3	GP43ADC
4	GP42ADC
5	GP41ADC
6	GP40ADC

J15: CAN0 and CAN1 bus

Pin#	Signal Name
1	CAN1_L
2	CAN1_H
3	GND
4	CAN0_L
5	CAN0_H
6	VCC3

J16: SPI, RESET-

Pin#	Signal Name	Pin#	Signal Name
1	SPIO_DI	2	VCC
3	SPIO_CLK	4	SPIO_DO
5	RESET-	6	GND
7	SPIO_CS	8	-
9	-	10	-

J17: SPI, RESET-, RS485

Pin#	Signal Name	Pin#	Signal Name
1	SPI1_DI	2	VCC
3	SPI1_CLK	4	SPI1_DO
5	RESET-	6	GND
7	SPI1_CS	8	-
9	RS485+	10	RS485-

J18: Line-out

Pin#	Signal Name
1	LOUT_R1
2	GND_AUD
3	GND_AUD
4	LOUT_L1

J19: VGA

Reserved and debug used.

Pin#	Signal Name
1	ROUT
2	GND
3	GOUT
4	GND
5	BOUT
6	GND
7	HSYNC_A
8	GND
9	VSYNC_A
10	GND

J24: MINI PCIe

Pin#	Signal Name	Pin#	Signal Name
1	WAKE#	2	+3.3V
3	Reserved / +5V Power-out	4	GND
5	Reserved / +5V Power-out	6	-
7	-	8	SIM-VCC
9	GND	10	SIM-IO
11	REFCLK-	12	SIM-CLK
13	REFCLK+	14	SIM-RST
15	GND	16	SIM-VPP
Mechanical Key			
17	-	18	GND
19	-	20	-
21	GND	22	PERST#
23	PERn0	24	+3.3V
25	PERp0	26	GND
27	GND	28	-
29	GND	30	-
31	PETn0	32	-
33	PETp0	34	GND
35	GND	36	USB_D-
37	GND	38	USB_D+
39	+3.3V	40	GND
41	+3.3V	42	LED_WWAN#
43	GND	44	Reserved / DCD
45	-	46	-
47	-	48	-
49	-	50	GND
51	-	52	+3.3V

J25: SIM Card Holder

Pin#	Signal Name	Pin#	Signal Name
1	SIM-VCC	2	SIM-RST
3	SIM-CLK	4	GND
5	SIM-VPP	6	SIM-IO
7	-	8	-

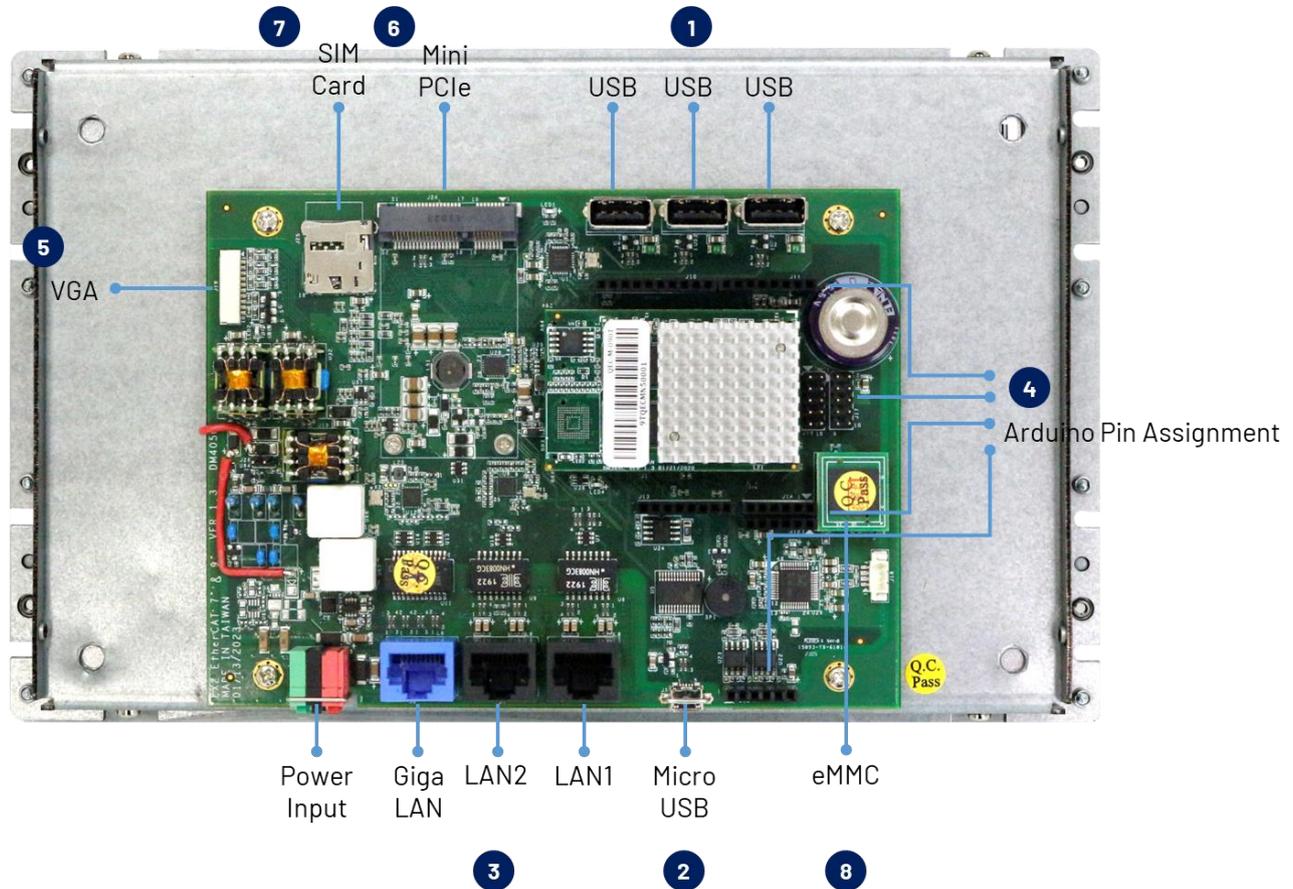
J29A: eMMC Module

Pin#	Signal Name	Pin#	Signal Name
A1	SDA_D0	A2	SDA_D1
A3	SDA_D2	A4	SDA_D3
A5	SDA_CLK	A6	SDA_CMD

J29B: eMMC Module

Pin#	Signal Name	Pin#	Signal Name
B1	VCC3	B2	SDA_CD
B3	VCC3	B4	GND

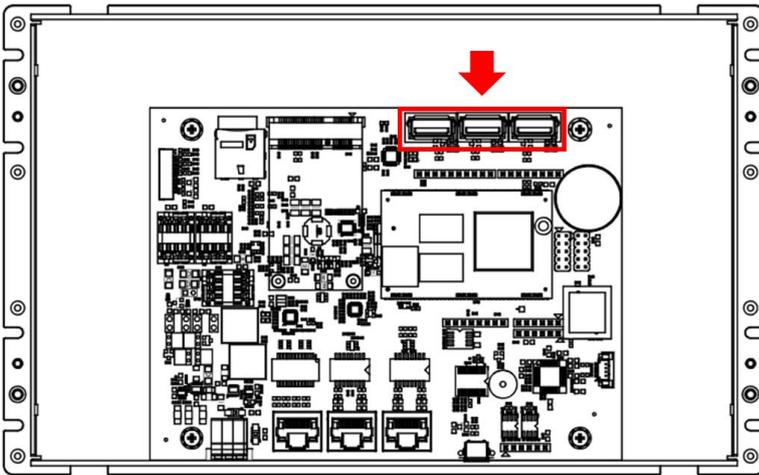
2.4 External I/O Overview



NOTE

1. Standard USB 2.0 Host.
2. Micro USB is mainly for the programming upload.
3. LAN1 and LAN2 are for the EtherCAT communication, and Giga LAN
4. Arduino Standard Pin and QEC additional Arduino Pin.
5. 10 Pin Flat Ribbon VGA.
6. Mini PCIe.
7. SIM Card.
8. eMMC Module.

2.4.1 USB



Standard USB 2.0 with Hot-plug.

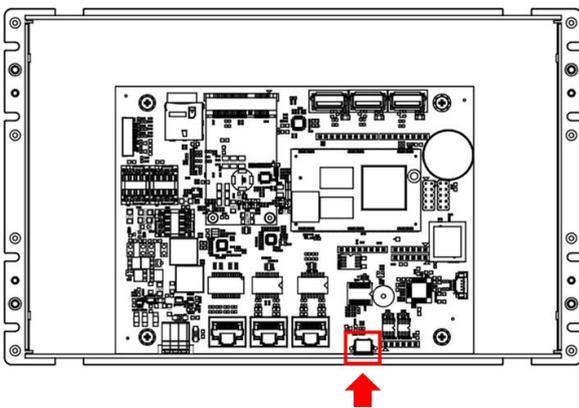
You can plug in the Keyboard, Mouse, or USB dick to control the QEC-M-090T.

For drive USB, you can refer to the following hyperlinks:

- [SD library](#): read USB disk.
- [Keyboard Controller Example](#)
- [Mouse Controller Example](#)

2.4.2 Micro USB

The Micro USB is mainly for programming upload.

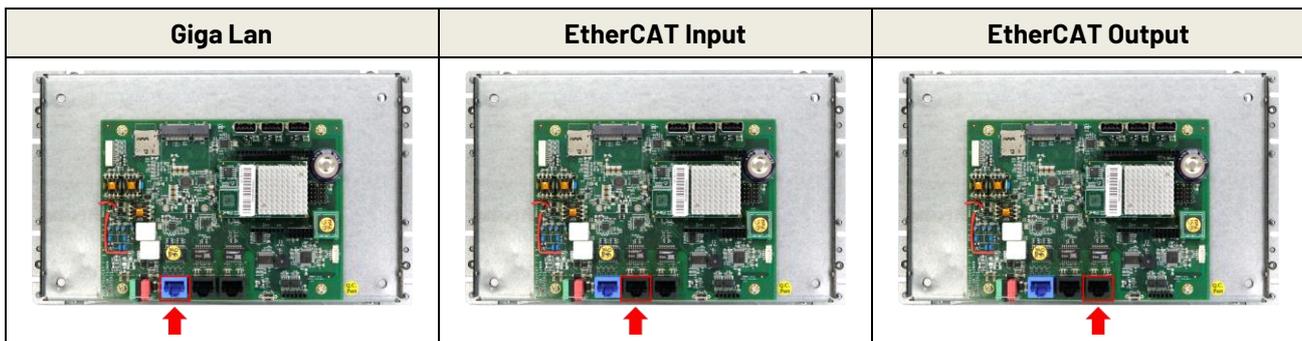


For quick start guide, please see [Ch. 3](#).

2.4.3 LAN1/LAN2/Giga LAN

There are three LAN ports in QEC-M-090T, two for EtherCAT communication and one for external Ethernet work. The EtherCAT Lan on the QEC-M divides into Input and Output for cable redundancy.

To drive GigaLAN, you can refer to [Ethernet library](#) or [Modbus Library](#).



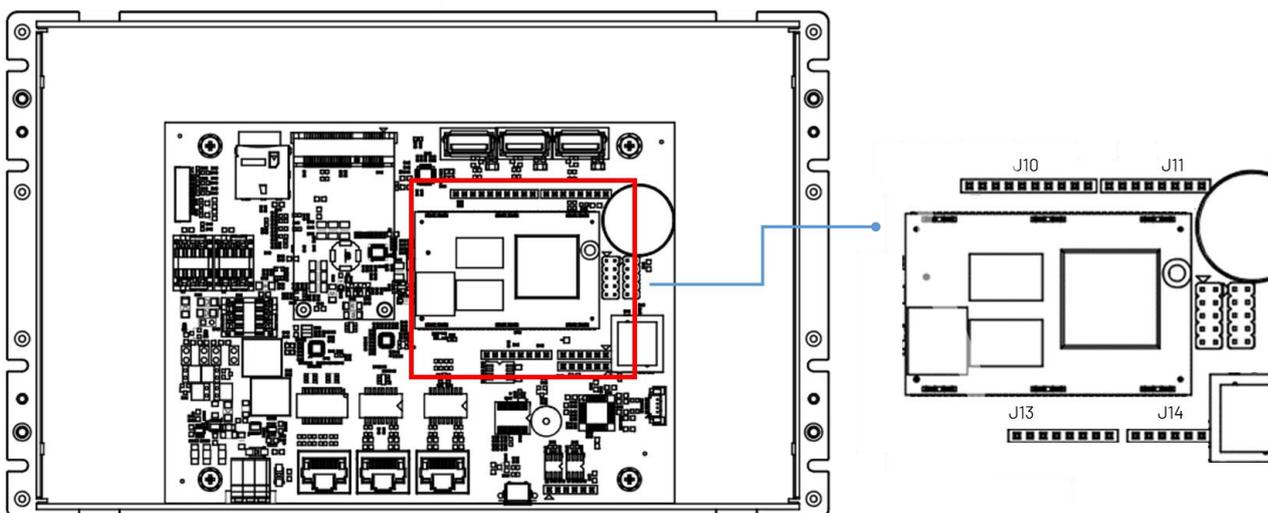
* Giga LAN with the Blue Housing; PoE LAN with the Red Housing; Regular LAN with Black Housing.

2.4.4 Arduino pin Assignment

We have kept the Arduino pin on the QEC-M-090T. Users can easily control these pins via software (86Duino IDE).

Arduino standard pins:

You can use the following pins like 86Duino One board.

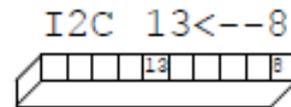


*MCM signal is equivalent to Arduino's PWM signal.

J10: I2C0, MCM, GPIO

Pin#	Signal Name
1	GP30
2	MCM-9
3	MCM-10
4	MCM-11
5	GP31
6	MCM-13
7	GND
8	-
9	I2C0_SDA
10	I2C0_SCL

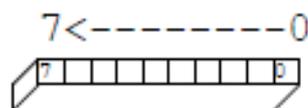
Female 2.54 Function Conn.



J11: MCM, GPIO, COM1(TTL)

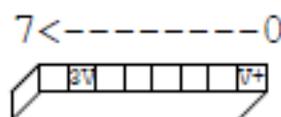
Pin#	Signal Name
1	RXD1#
2	TXD1#
3	GP00
4	MCM-3
5	GP02
6	MCM-5
7	MCM-6
8	GP05

Female 2.54 Function Conn.



J13: Power source, RESET-

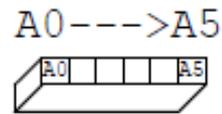
Pin#	Signal Name
1	VCC
2	GND
3	GND
4	VCC
5	VCC3
6	RESET-
7	VCC3
8	-



Female 2.54 Function Conn.

J14: ADC/GPIO

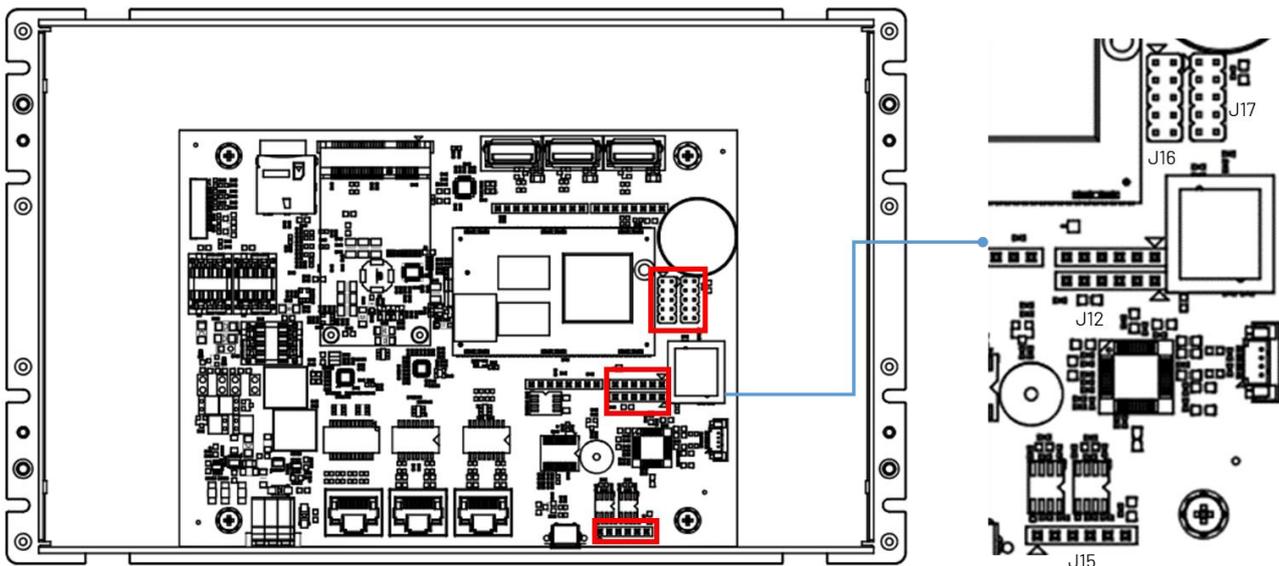
Pin#	Signal Name
1	GP57ADC
2	GP56ADC
3	GP43ADC
4	GP42ADC
5	GP41ADC
6	GP40ADC



Female 2.54 Function Conn.

QEC Arduino pins:

There are other pins on the QEC-M-090T.
Please contact ICOP for the details.



J12: GPIO, VCC, GND

Pin#	Signal Name
1	-
2	GP35
3	GP36
4	GP37
5	GND
6	VCC



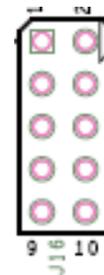
J15: CAN0 and CAN1 bus

Pin#	Signal Name
1	CAN1_L
2	CAN1_H
3	GND
4	CAN0_L
5	CAN0_H
6	VCC3



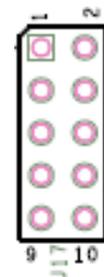
J16: SPI, RESET-

Pin#	Signal Name	Pin#	Signal Name
1	SPI0_DI	2	VCC
3	SPI0_CLK	4	SPI0_DO
5	RESET-	6	GND
7	SPI0_CS	8	-
9	-	10	-



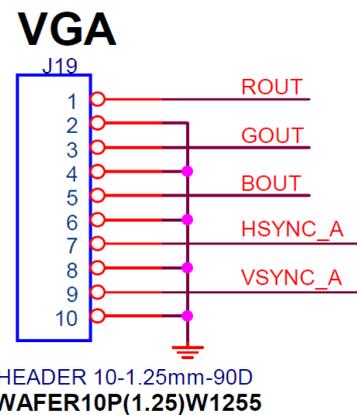
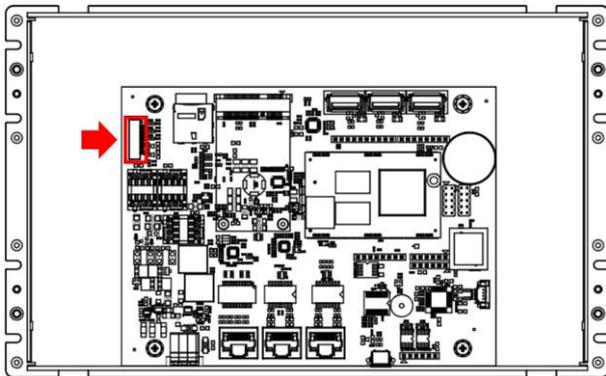
J17: SPI, RESET-, RS485

Pin#	Signal Name	Pin#	Signal Name
1	SPI1_DI	2	VCC
3	SPI1_CLK	4	SPI1_DO
5	RESET-	6	GND
7	SPI1_CS	8	-
9	RS485+	10	RS485-



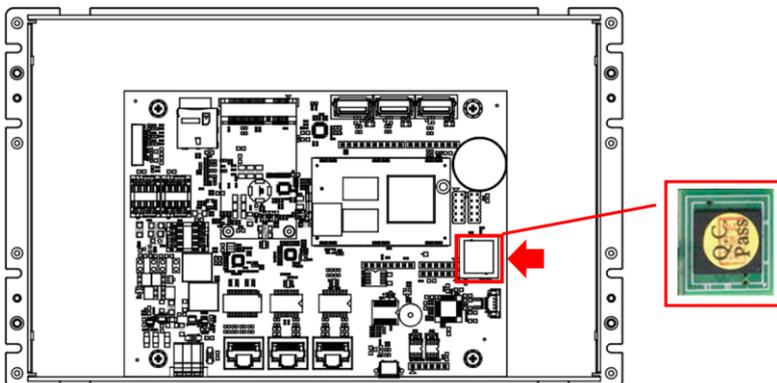
2.4.5 10-pin VGA Connector

Reserved and debug used.



The pin configuration of a VGA Connector includes 10 pins where each pin and its function are discussed right.

2.4.6 eMMC Module



A 2GB eMMC module is onboard by default.

Your 86Duino executable will be uploaded to this eMMC.

To save data to this eMMC, you can refer to [SD library](#) and set `SD.setBank(EMMCDISK);`.

2.5 Watchdog Timer

There are two watchdog timers in [Vortex86EX2 processor](#), we also provide DOS, Linux and WinCE example for your reference. Please contact ICOP for more detail information.

Ch. 3

Quick Start guide

[3.1 Package Contents](#)

[3.2 Hardware Configuration](#)

[3.3 Software Driver Installation](#)

[3.4 Set up the QEC-M for 86Duino](#)

[3.5 Building a HMI on the QEC-M-090T with 86Duino](#)

3.1 Package Contents

The package includes the following items:

- QEC-M-090T
- Cable-set
- Product warranty card

Please contact our sales channels if any of the package items are missing or damaged. Also, feel free to reuse the shipping materials and carton for further storing and shipping needs in the future

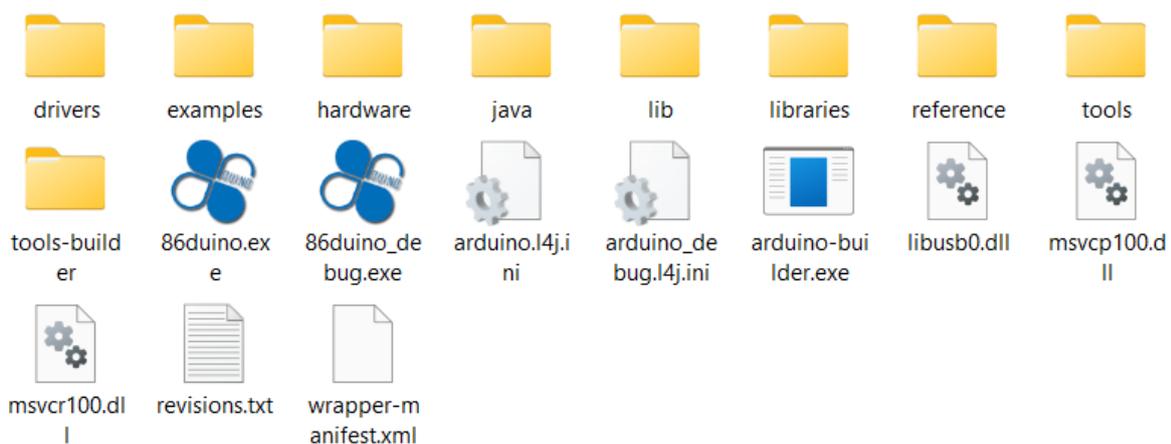
3.2 Hardware Configuration

Before QEC-M Series being shipped to customers, development environment will be pre-installed. All users need to do is to download the software (see [3.3](#)) and follow this user manual to set up the system configuration.

3.3 Software Driver Installation

For QEC-M series software download, please see the following steps:

1. Go to QEC website > Download and scroll down to find the “Download”. Choose the corresponding 86Duino Coding IDE version and download.
2. You will get a zip file, and please unzip it.
3. You will see **86duino.exe** which looks like the picture below.

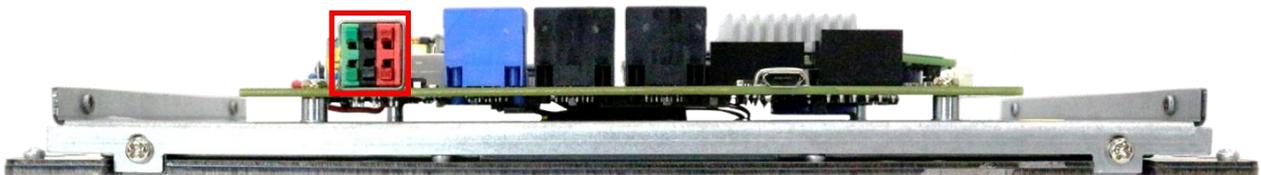


3.4 Set up the QEC-M-090T with 86Duino

For this section, you will need to work with QEC-M-090T products series, and the following instructions is for devices using Window OS system.

Plugin the power supply

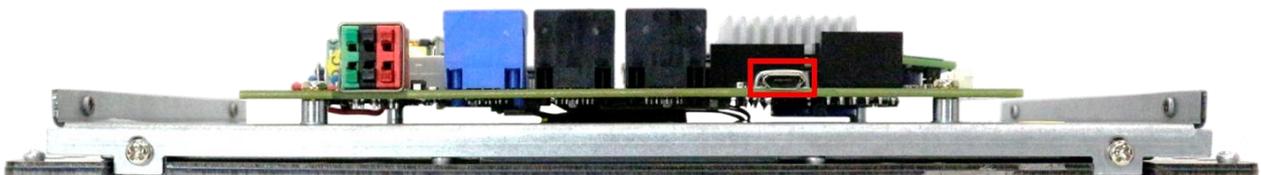
There are two groups of power supplies in QEC-M-090T, Vs and Vp; The voltage requirement for both supplies range from 19V to 50V wide voltage. After powering on, you'll see the power LED light up.



Note: Vs for system power; Vp for peripheral power and backup power

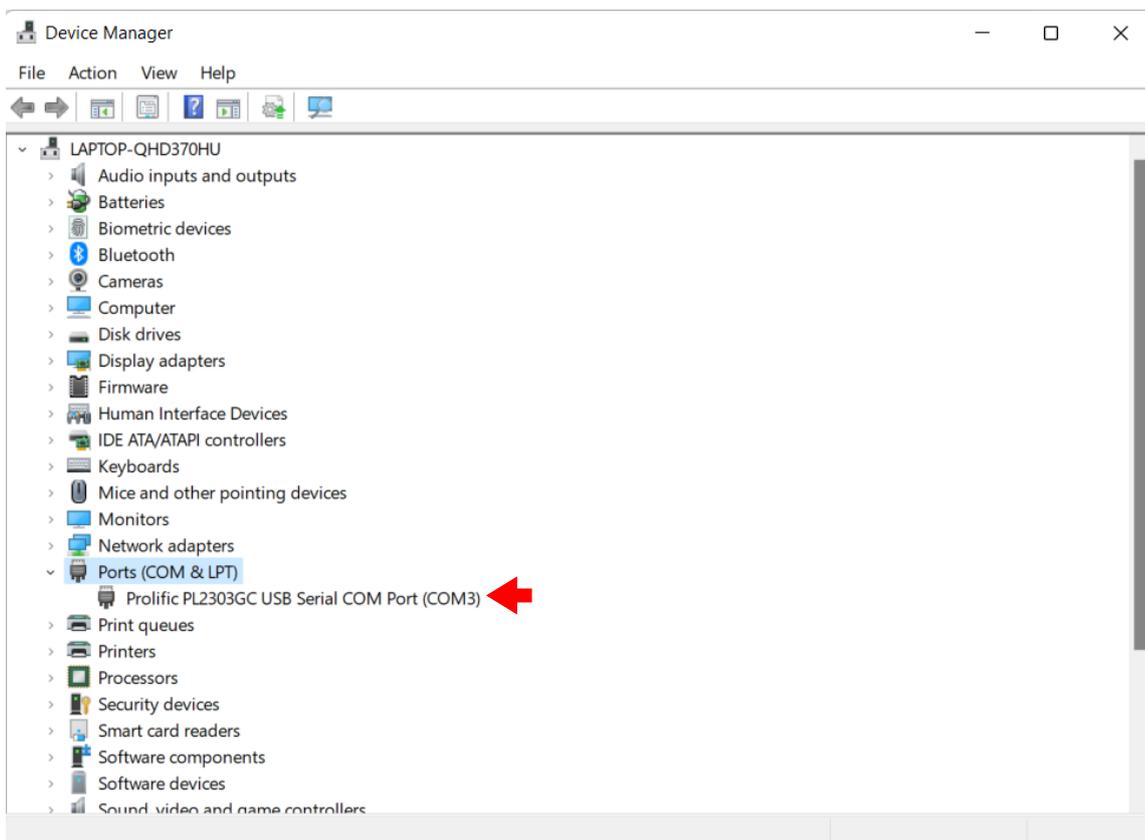
Connect the USB cable with PC

Please connect the Micro USB to USB from QEC-M-090T to your PC which is installed the 86Duino Coding IDE 500+.



Make sure your device is detected

Open Device Manager -> Ports (COM & LPT) in your PC and expand the ports, you should see that the "Prolific PL2303GC USB Serial COM Port (COMx)" detected, if not, you will need to install the required drivers.

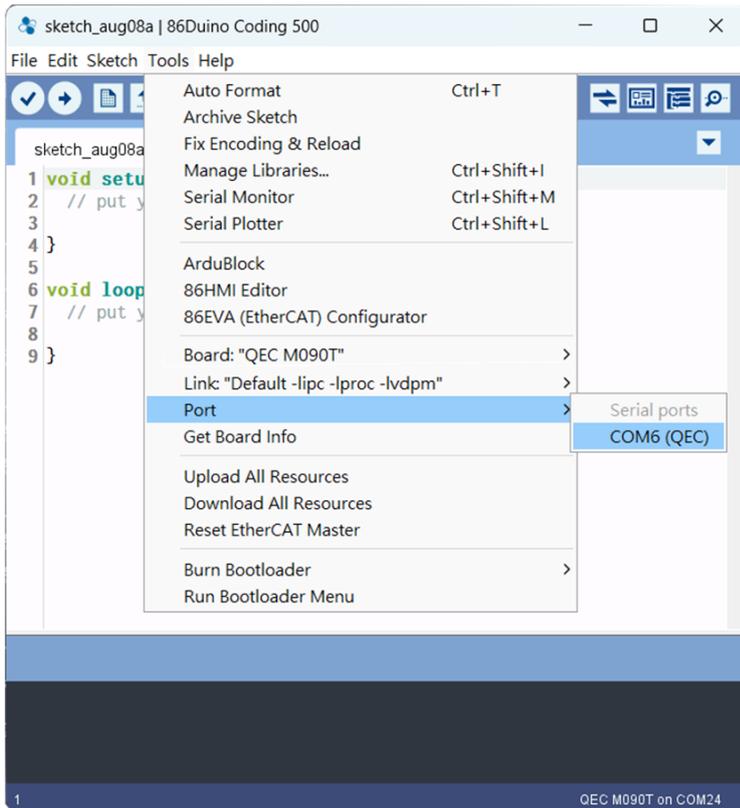


* For Windows PL2303 driver, you can download [here](#).

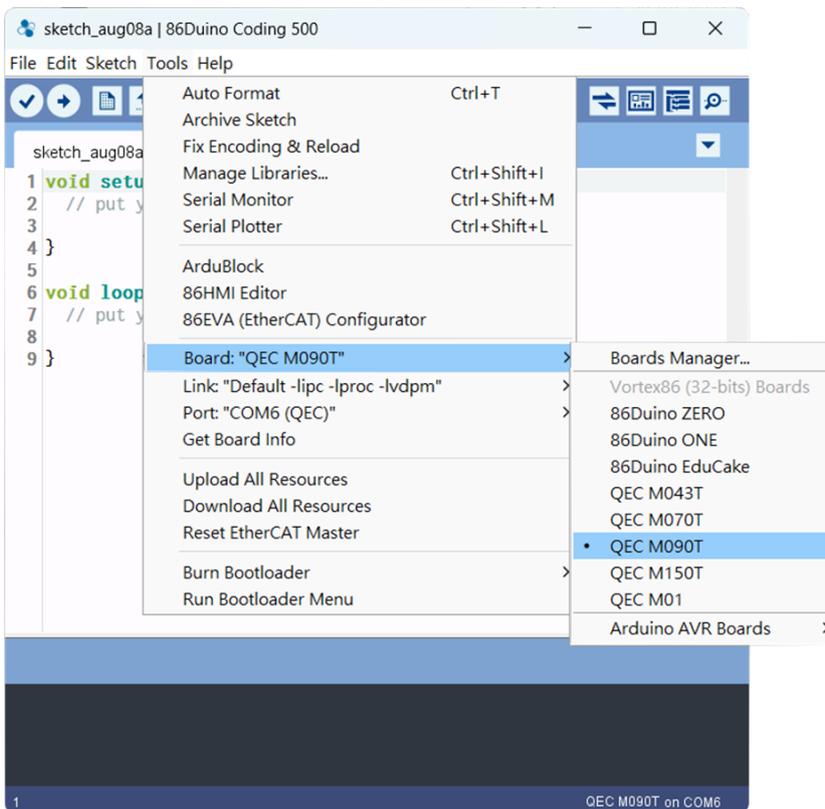
Write your first sketch

- Open the 86Duino Coding IDE 500+
- Open the Tools menu from Menu bar

- Select the assigned COM port for QEC-M-090T. (Here is COM6)



- Select "QEC M090T" for the board.



- Put your code into the editor.

Upload your first sketch!

Hit the right arrow button (upload) next to the upload button. Now while the code uploads, you should see the LEDs next to Tx and Rx blinking indicating data transfer between the board and the computer.

If this is your first time running 86Duino sketch files, we suggest you check out [how to set up the QEC-M-043T for 86Duino](#) before proceeding.

Video: <https://youtu.be/ZMNMtvRCLbs>

3.5 Building an HMI on the QEC-M-090T with 86Duino

This section will demonstrate how to build a basic HMI on the QEC-M-090T using the LVGL library in the 86Duino Coding IDE 500+.

We assume that you have already completed the previous sections of the quick start guide, including Package Contents, Hardware Configuration, Software Driver Installation, and Set up the QEC-M-090T.

Instruction

LVGL (Light and Versatile Graphics Library) is a powerful open-source graphics library that enables the creation of attractive and interactive graphical user interfaces (GUIs) for embedded systems. By utilizing the LVGL library in combination with 86Duino IDE, developers can design and implement HMIs with ease and efficiency.

You can expand and customize the HMI by utilizing other LVGL widgets such as buttons, sliders, graphs, images, and more. LVGL provides a wide range of built-in widgets and customization options to create sophisticated and visually appealing HMIs tailored to your specific application.

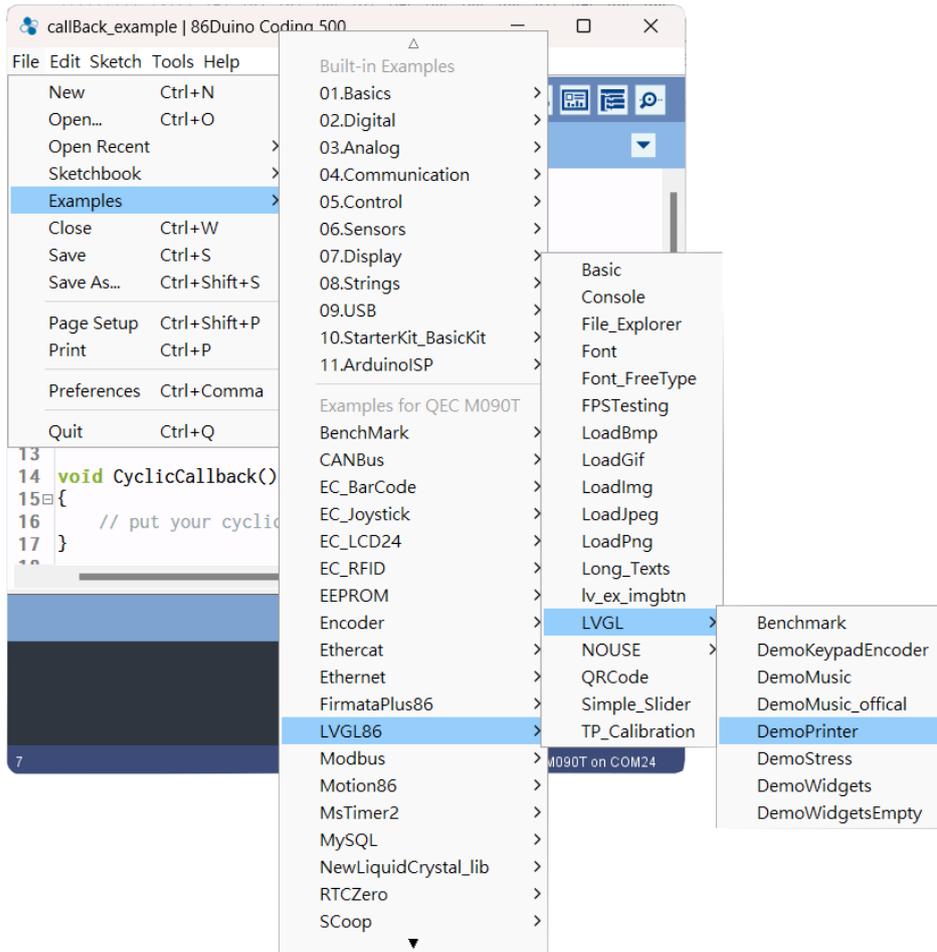
For more details about LVGL, please see [Welcome to the documentation of LVGL! – LVGL documentation!](#)

Uploading the LVGL Example

Use the LVGL example “DemoWidgets” to upload the HMI display on QEC-M-090T. Below are the complete steps for uploading the sketch to QEC-M-090T:

1. Open the 86Duino Coding IDE 500+ and connect the QEC-M-090T.
2. Open the Tools menu from Menu bar -> Select the assigned COM port for QEC-M-090T -> Select “QEC M090T” for the board.

3. Select “Examples” in the File menu, go to LVGL86 -> LVGL and open the “DemoPrinter”.

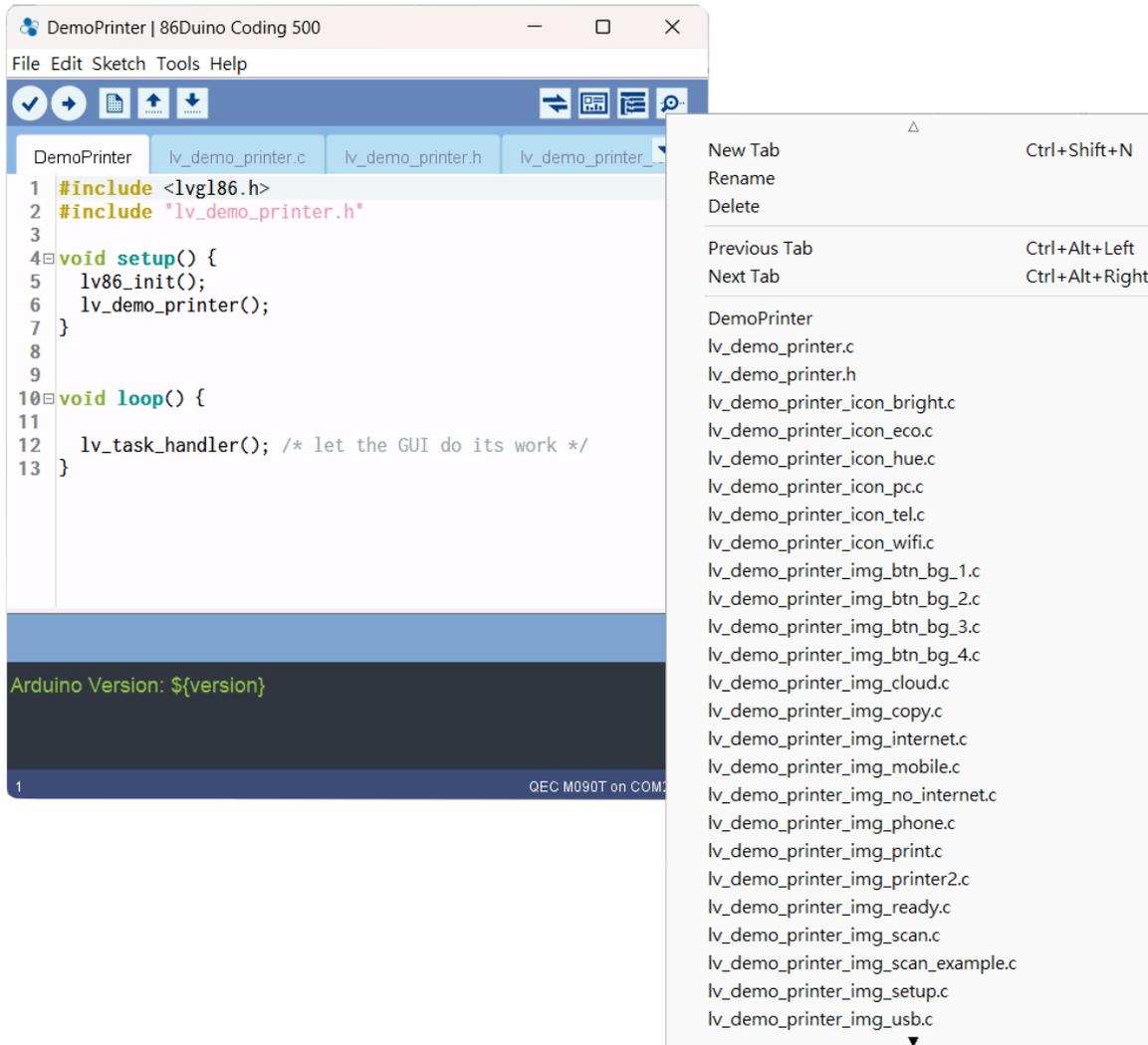


4. Click the “Upload” button to compile your code and upload it to the QEC-M-090T.
5. Then, you can see the DemoPrinter example on QEC-M-090T after the upload is complete.



There are many files in the 'DemoPrinter' example.

1. DemoPrinter:
 - *.ino files are your main application files.
2. .c:
 - *.c files are our C source code files in which our code is written and created by the user.
3. .h:
 - *.h files are header files that are prewritten for our compiler.

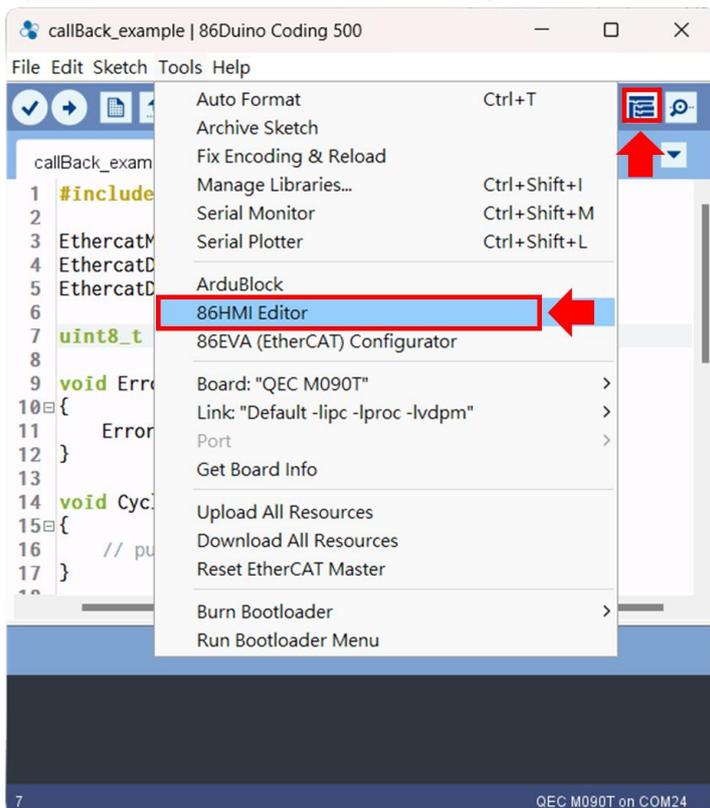


Using the Graphical HMI editor: 86HMI

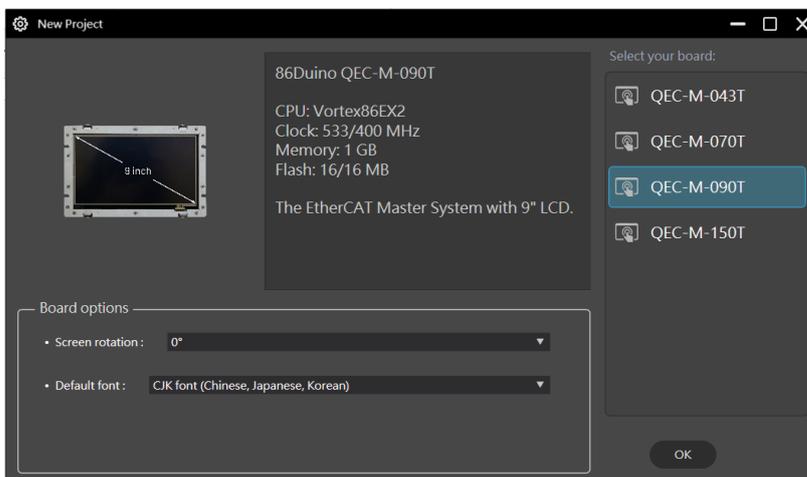
86HMI Editor is an easy-to-use HMI editor that can be used to create a customized HMI quickly. Use the Auto Code Generation function in 86HMI Editor to generate HMI APIs (Application Programming Interface), thus achieving the effect of creating HMIs without writing programs.

Below are the complete steps to design UI to QEC-M-090T using 86HMI:

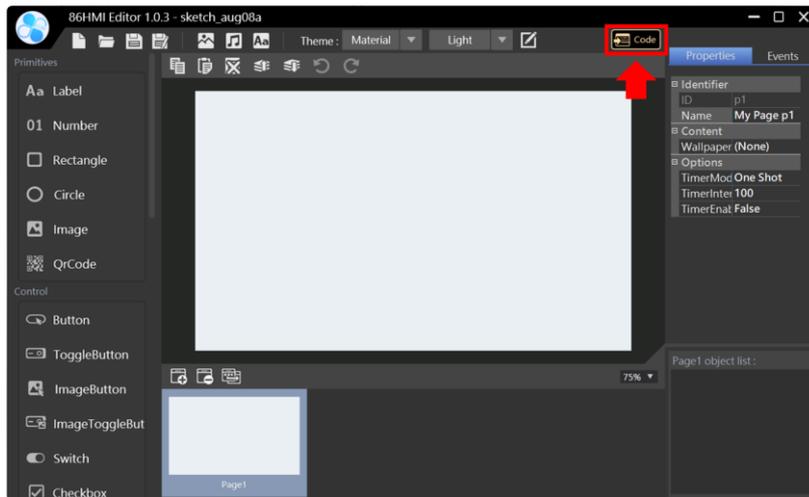
1. Open 86HMI tool via 86Duino Coding IDE 500+.



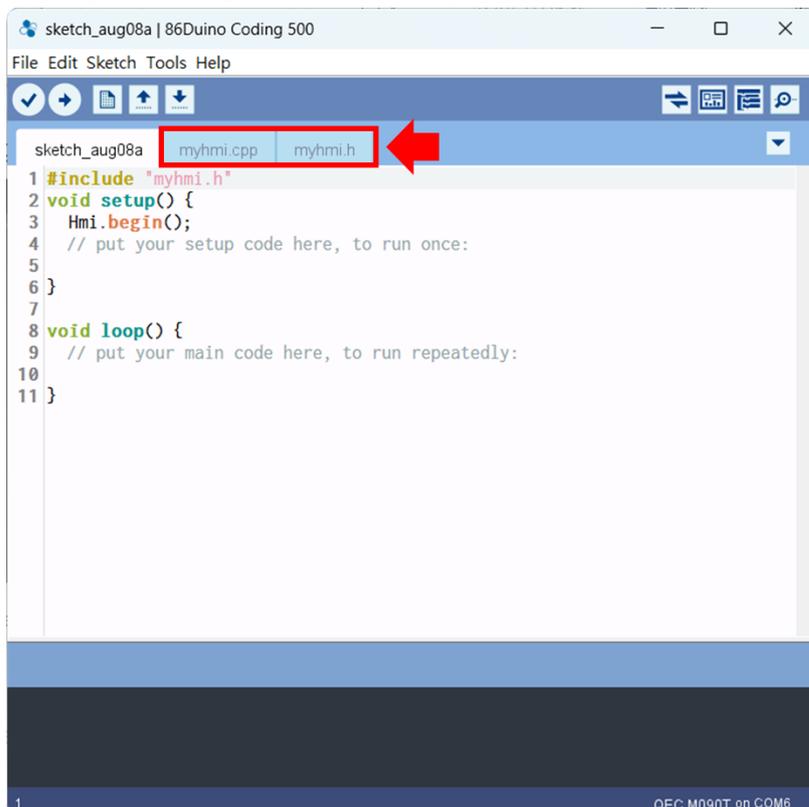
2. Choose QEC-M-090T.



- Then, you can use the left menu to start designing your UI and click the "Code" button to generate the source code back to 86Duino IDE automatically.



- After generating, you can see myhmi.h and myhmi.cpp in 86Duino IDE.



And after you finish uploading, you can see the user interface you just designed on QEC-M-090T.

For more information on how 86HMI can be developed, please write to info@icop.com.tw, call your nearest [ICOP Branch](#), or contact our [Worldwide Official Distributor](#).

Ch. 4

Software Function

[4.1 Software Description](#)

[4.2 EtherCAT Function List](#)

[4.3 Additional Resources](#)

4.1 Software Description

The 86Duino Coding IDE 500+ developed by the QEC team designed specifically for industrial-field control systems, bringing simple and powerful functions into related industrial fields through the open-source Arduino.

Please visit qec.tw for 86Duino Coding IDE 500+ details.

4.2 EtherCAT Function List

The list below introduces the functions of our QEC series products.

Please visit [EtherCAT Master API User Manual](#) for API Function details.

EthercatMaster Class Functions

Initialization Functions:

- `begin()` - EtherCAT Master Initialize.
- `end()` - Shutdown EtherCAT Master.
- `isRedundancy()` - Check EtherCAT uses redundancy or not.
- `libraryVersion()` - The version of EtherCAT Master library.
- `firmwareVersion()` - The version of EtherCAT firmware.

Access to slave information Functions:

- `getSlaveCount()` - Get EtherCAT Slave Count Number on EtherCAT bus.
- `getVendorID()` - Get the EtherCAT Slave Vendor ID on EtherCAT bus.
- `getProductCode()` - Get the EtherCAT Slave Product Code on EtherCAT bus.
- `getRevisionNumber()` - Get the EtherCAT Slave revision Number on EtherCAT bus.
- `getSerialNumber()` - Get the EtherCAT Slave Serial Number on EtherCAT bus.
- `getAliasAddress()` - Get the EtherCAT Slave Alias Address on EtherCAT bus.
- `getSlaveNo()` - Obtain the sequential ID of the specified slave based on the alias address/supplier number/product number/amendment number/serial number.

Control Functions:

- start() - Start EtherCAT Master. All slaves will enter OP state if successful. There are three modes: Auto Freerun / Manually Freerun / Sync.
- stop() - Stop EtherCAT Master.
- getSystemTime() - Get system time of current cycle. (The unit is nanosecond)
- getWorkingCounter() - Get working counter of current cycle.
- getExpectedWorkingCounter() - Get expected working counter. (Fixed value)

EthercatDevice Class General Functions

Access to slave information Functions:

- getVendorID - Get EtherCAT Slave Count Number on EtherCAT bus.
- getProductCode - Get the EtherCAT Slave Product Code on EtherCAT bus.
- getRevisionNumber - Get the EtherCAT Slave revision Number on EtherCAT bus.
- getSerialNumber - Get the EtherCAT Slave Serial Number on EtherCAT bus.
- getAliasAddress - Get the EtherCAT Slave Alias Address on EtherCAT bus.
- getSlaveNo - Get EtherCAT Slave Number on EtherCAT bus.
- readSII - Read EEPROM Info.
- readSII8 - Read EEPROM Info. (uint8_t)
- readSII16 - Read EEPROM Info. (uint16_t)
- readSII32 - Read EEPROM Info. (uint32_t)
- writeSII - Write EEPROM.
- writeSII8 - Write EEPROM. (uint8_t)
- writeSII16 - Write EEPROM. (uint16_t)
- writeSII32 - Write EEPROM. (uint32_t)

Process Data Objects (PDO) Functions:

- pdoBitWrite() - Write Bit of Process Data Output.
- pdoBitRead() - Read Bit of Process Data Input.
- pdoGetOutputBuffer() - Get Slave Process Data Output Pointer.
- pdoGetInputBuffer() - Get Slave Process Data Input Pointer.
- pdoWrite() - Write Slave Process Data Output.
- pdoWrite8() - Write Slave Process Data Output. (unit8_t)
- pdoWrite16() - Write Slave Process Data Output. (unit16_t)
- pdoWrite32() - Write Slave Process Data Output. (unit32_t)
- pdoRead() - Read Slave Process Data Input.
- pdoRead8() - Read Slave Process Data Input. (unit8_t)
- pdoRead16() - Read Slave Process Data Input. (unit16_t)
- pdoRead32() - Read Slave Process Data Input. (unit32_t)

CANopen over EtherCAT (CoE) Functions:

- sdoDownload() - (CoE) Write the object to EtherCAT Slave device.
- sdoDownload8() - (CoE) Write the object to EtherCAT Slave device. (unit8_t)
- sdoDownload16() - (CoE) Write the object to EtherCAT Slave device. (unit16_t)
- sdoDownload32() - (CoE) Write the object to EtherCAT Slave device. (unit32_t)
- sdoUpload() - (CoE) Read the object from EtherCAT Slave device to EtherCAT Master.
- sdoUpload8() - (CoE) Read the object from EtherCAT Slave device to EtherCAT Master. (unit8_t)
- sdoUpload16() - (CoE) Read the object from EtherCAT Slave device to EtherCAT Master. (unit16_t)
- sdoUpload32() - (CoE) Read the object from EtherCAT Slave device to EtherCAT Master. (unit32_t)
- getObjectList() - (CoE) Get the object list from EtherCAT Slave device to EtherCAT Master.
- getObjectDescription() - (CoE) Get the object list's Description from EtherCAT Slave device to EtherCAT Master.
- getEntryDescription() - (CoE) Get the entry description with Slave index.

File over EtherCAT (FoE) Functions:

- readFoE() - Read FoE.
- writeFoE() - Write FoE.

EthercatDevice_Generic Class Functions

Universal objects for various brands of EtherCAT devices.

Initialization Functions:

- `attach()` - Specify the EC-Slave number and mount it on the EC-Master.
- `detach()` - Uninstall the slave object.

4.3 Additional Resources

If you want to learn more about the libraries available in the 86Duino IDE or explore the details of the 86Duino programming language, please visit the following links:

- **86Duino IDE Libraries:** Find an extensive list of libraries supported by 86Duino IDE, along with detailed documentation and examples at <https://www.qec.tw/86duino/libraries/>.
- **86Duino Language Reference:** Learn about the 86Duino programming language, including its syntax, functions, and usage in the official 86Duino Language Reference at <https://www.qec.tw/86duino/86duino-language-reference/>.

These resources provide valuable information for both beginners and experienced developers using the 86Duino platform. By exploring these links, you can harness the full potential of 86Duino IDE and create innovative projects.

Happy coding with 86Duino IDE!

The text of the 86Duino reference is a modification of [the Arduino reference](#) and is licensed under a [Creative Commons Attribution-ShareAlike 3.0 License](#). Code samples in the reference are released into the public domain.

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. 2023