

DEBIX Model C User Guide

Version: V1.5 (2025-03)

Compiled by: Polyhex Technology Company Limited (<http://www.polyhex.net/>)

DEBIX Model C is the first DEBIX single board computer to feature the NXP i.MX 93, a low-power processor rating up to 1.7GHz with only 1 watt of power at full load consumption, and the Arm Ethos™-U65 microNPU enables developers to create more capable ML applications.

Engineered to deliver a more energy-efficient and cost-effective solutions for intelligent edge computing, DEBIX Model C provides multiple extensible interfaces for IoT edge, contactless HMI, smart home, building control and industrial applications.

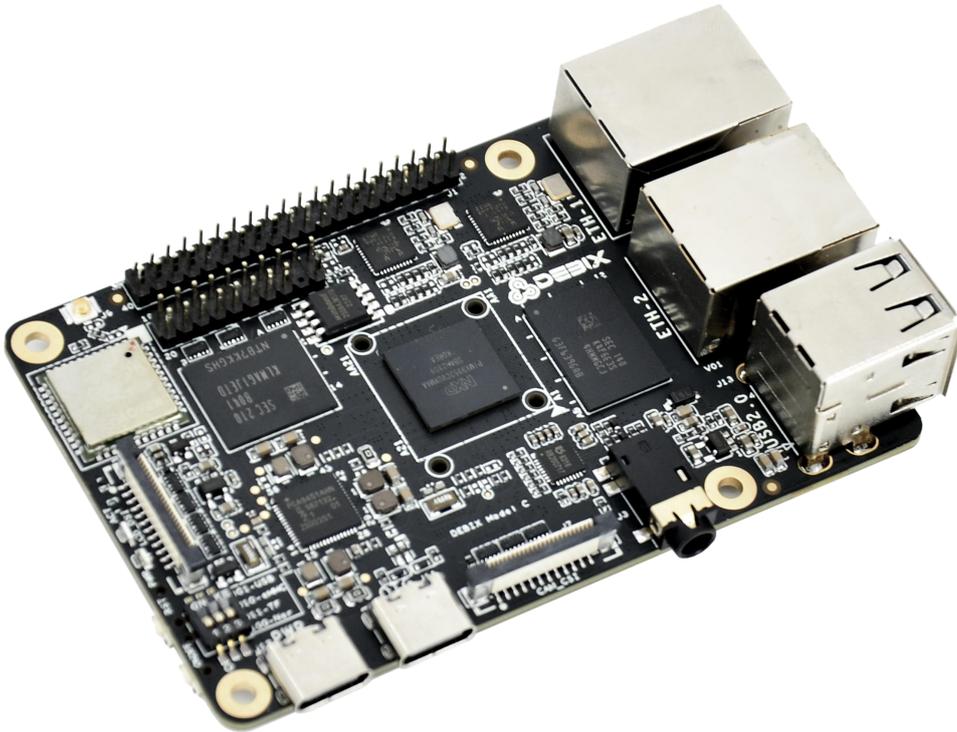


Figure 1 DEBIX Model C

REVISION HISTORY		
Rev.	Date	Description
1.0	2023.08.29	First edition
1.1	2024.04.19	Added 4.7.Usage of Display and 4.8.Usage of Camera .
1.2	2024.08.16	Revised 4.7.Usage of Display : updated the display model, and modified the pin connection instructions.
1.3	2025.01.14	1. Updated the supported OS version in the specifications table 2. Updated the technical support contact information
1.4	2025.01.23	Revised the supply voltage options of LVDS
1.5	2025.03.07	Revised the part of 3.1.2 System Boot and optimized content

INDEX

REVISION HISTORY	2
INDEX	3
Chapter 1 Security	6
1.1. Safety Precaution	6
1.2. Safety Instruction	6
1.3. Declaration of Compliance	7
1.4. Technical Support	8
Chapter 2 DEBIX Model C Introduction	10
2.1. Overview	11
2.2. Composition	14
2.3. Interface	14
2.3.1. Power Interface	14
2.3.2. USB Interface	15
2.3.3. Ethernet Interface	16
2.3.4. Display Interface	17
2.3.4.1. LVDS Interface	17
2.3.4.2. MIPI DSI	19
2.3.5. MIPI CSI	21
2.3.6. Audio	25
2.3.7. GPIO	26

2.3.8. LED & Key	27
2.3.9. DIP Switch	28
2.3.10. Slot	29
2.4. Packing List	30
Chapter 3 Get Started	31
3.1. Software Installation	31
3.1.1. Download Image	31
3.1.2. System Boot	31
3.1.2.1. Flashing Micro SD Image	32
3.1.2.2. Flashing eMMC Image	34
3.2. Hardware Installation	36
Chapter 4 Software Application Examples	37
4.1. Switch Boot Mode	37
4.2. Usage of Ethernet	38
4.3. Usage of WiFi	40
4.4. Usage of BT	42
4.5. Usage of Audio	43
4.6. Usage of USB	44
4.7. Usage of Display	47
4.8. Usage of Camera	51
4.9. ADC IN Verification	52

4.10. LED & Key	53
4.11. Usage of GPIO	53
4.12. Heat Dissipation	56

Chapter 1 Security

1.1. Safety Precaution

This document inform how to make each cable connection. In most cases, you will simply need to connect a standard cable.

Table 1 Terms and conventions

Symbol	Meaning
<p><i>Warning!</i></p> 	<p>Always disconnect the power cord from the chassis whenever there is no workload required on it. Do not connect the power cable while the power is on. Sudden power surges can damage sensitive electronic components. Only experienced electricians should open the chassis.</p>
<p><i>Caution!</i></p> 	<p>Always ground yourself to remove any static electric charge before touching <i>DEBIX</i> product. Modern electronic devices are very sensitive to electric charges. Use a grounding wrist strap at all times. Place all electronic components on a static-dissipative surface or in a static-shielded bag.</p>

1.2. Safety Instruction

To avoid malfunction or damage to this product please observe the following:

1. Disconnect the device from the DC power supply before cleaning. Use a damp cloth. Do not use liquid detergents or spray-on detergents.
2. Keep the device away from moisture.
3. During installation, set the device down on a reliable surface. Drops and bumps will lead to damage.
4. Before connecting the power supply, ensure that the voltage is in the required range, and the way of wiring is correct.
5. Carefully put the power cable in place to avoid stepping on it.
6. If the device is not used for a long time, power it off to avoid damage caused by

sudden overvoltage.

7. Do not pour liquid into the venting holes of the enclosure, as this could cause fire or electric shock.

8. For safety reasons, the device can only be disassembled by professional personnel.

9. If one of the following situations occur, get the equipment checked by service personnel:

- The power cord or plug is damaged.
- Liquid has penetrated into the equipment.
- The equipment has been exposed to moisture.
- The equipment does not work well, or you cannot get it to work according to the user's manual.
- The equipment has been dropped and damaged.
- The equipment has obvious signs of breakage.

10. Do not place the device outside the specified ambient temperature range. This will damage the machine. It needs to be kept in an environment at controlled temperature.

11. Due to the sensitive nature of the equipment, it must be stored in a restricted access location, only accessible by qualified engineer.

DISCLAIMER: Polyhex disclaims all responsibility for the accuracy of any statement of this instructional document.

1.3. Declaration of Compliance

This product has passed the following certifications:

Table 2 Compliance Certification

Symbol	Meaning
	This equipment has passed CE certified.

	This equipment is manufactured in compliance with RoHS regulations.
	This equipment has passed UKCA certified.
	This equipment has passed FCC certified.
	This equipment has passed PSE certified.
	This equipment has passed C-Tick certified.
	This equipment is manufactured in compliance with RCM regulations.

1.4. Technical Support

1. Visit DEBIX website <https://www.debix.io/> where you can find the latest information about the product.

■ Quick Links:

Debix Documentation: <https://debix.io/Document/manual.html>

Debix Blog: <https://debix.io/Software/blog.html>

Debix GitHub: <https://github.com/debix-tech>

2. Contact your distributor, sales representative or Polyhex's customer service center for technical support if you need additional assistance. Please have the following info ready before you call:

- Product name and memory size
- Description of your peripheral attachments

- Description of your software(operating system, version, application software, etc.)
- A complete description of the problem
- The exact wording of any error messages

■ **TechSupport Platforms:**

Discord Community (recommended): <https://discord.com/invite/adaHHaDkH2>

Email: teksupport@debix.io

Chapter 2 DEBIX Model C Introduction

DEBIX Model C is based on an NXP i.MX 93 single-board computer that integrates two Arm® Cortex®-A55 cores, an Arm Cortex-M33 core, and an Arm® Ethos™-U65 Neural Processing Unit (NPU) to provide high performance, low power consumption, multiple power modes, and advanced security. It is widely used in machine vision and machine learning, smart city, IoT gateway, edge computing, and security.

Main features:

- Powerful Dual Core Arm® Cortex® -A55 processor at up to 1.7 GHz with integrated NPU to accelerate machine learning inference;
- General-purpose Arm® Cortex®-M33 at rates up to 250 MHz for real-time and low-power processing;
- Arm Ethos™-U65 microNPU to bring MCU-level ML efficiency;
- Dual 1 Gbps Ethernet controllers drive low latency for gateway applications, one of which supports Time Sensitive Networking (TSN);
- Compatible with DEBIX PoE module, Camera 200A/500A and DEBIX 5"/7"/8"/10.1" LCD monitors;
- Supports system switching between Ubuntu 22.04 Server, Yocto-L6.1.36 and Debian 12 Server.
- Supports cooperative work on FreeRTOS and Linux dual systems.

2.1. Overview

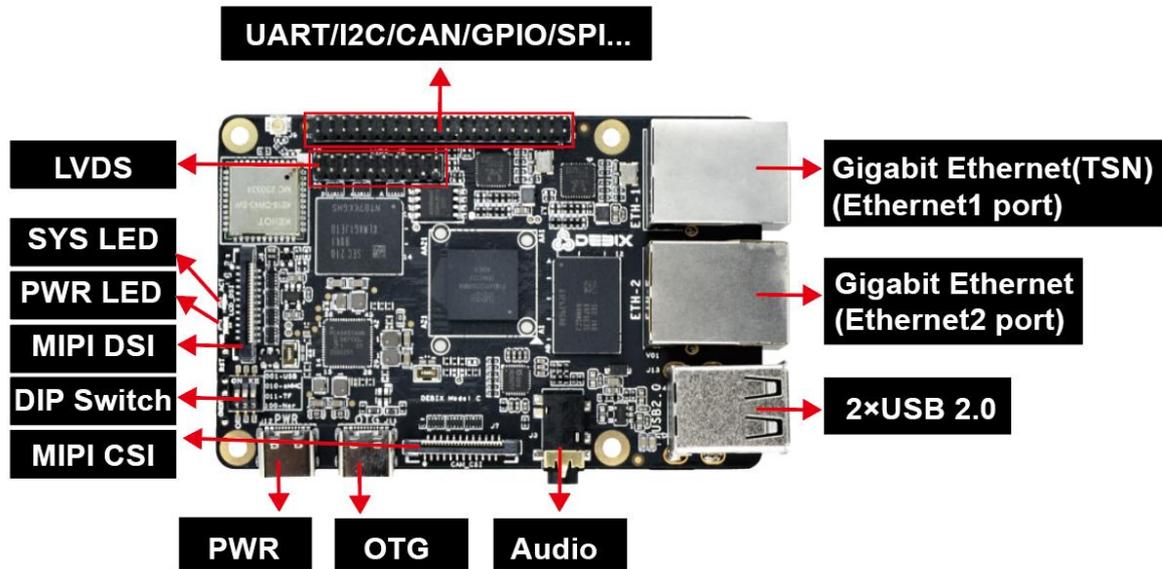


Figure 2

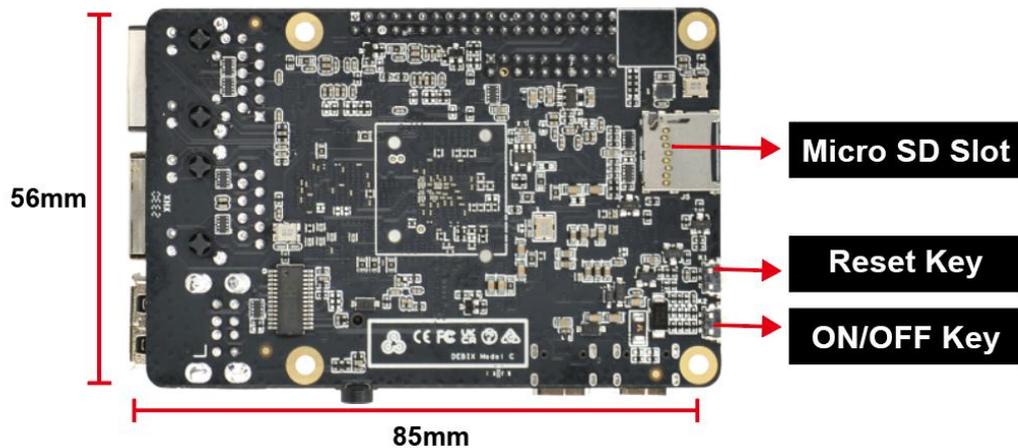


Figure 3

DEBIX Model C uses NXP i.MX 93 Plus based Soc, supports Gigabit Ethernet, 2.4GHz & 5GHz dual-band wireless network and Bluetooth 5.2, etc. The data specifications are as below:

Table 3 DEBIX Model C Specification

System	
CPU	NXP i.MX9352 (i.MX 93 series CPU optional),

	<ul style="list-style-type: none"> ● 2 x Arm® Cortex®-A55 @1.7 GHz, ● 1 x Arm Cortex-M33 @250 MHz, ● Arm® Ethos™ U-65 microNPU @0.5 TOPS
Memory	1GB LPDDR4 (2GB optional)
Storage	<ul style="list-style-type: none"> ● Micro SD Card (8GB/16GB/32GB/64GB/128GB/256GB optional) ● Onboard eMMC (8GB/16GB/32GB/64GB/128GB/256GB optional)
OS	Ubuntu 22.04 Server, Yocto-L6.1.36, Debian 12 Server
Boot Mode	<ul style="list-style-type: none"> ● Nor Flash (Default) ● Micro SD card ● eMMC
Communication	
Gigabit Network	<ul style="list-style-type: none"> ● 2 x 10/100/1000M Ethernet interfaces <ul style="list-style-type: none"> ■ 1 x Gigabit Ethernet port, support TSN and POE power supply (need POE power supply module) ■ 1 x Gigabit Ethernet port (POE power supply is not supported)
Wi-Fi & BT	2.4GHz & 5GHz WiFi IEEE 802.11a/b/g/n, BT 5.2, external Wi-Fi SMA antenna connector
Video & Audio	
LVDS	1 x 720p60 LVDS output, single channel 8 bit, 2 x 10 Pin double-row headers
MIPI DSI	1 x 1080p60 MIPI DSI, support 4-lane, 24Pin 0.5mm Pitch FPC socket
MIPI CSI	1 x 1080p60 MIPI CSI, support 2-lane, 24Pin 0.5mm Pitch FPC socket
Audio	1 x 3.5mm headphone and microphone combo port
External I/O Interface	

USB	<ul style="list-style-type: none"> ● 2 x USB 2.0 Host, the connector is double layer Type-A interface ● 1 x USB 2.0 OTG, the connector is Type-C interface ● 1 x USB 2.0 PWR, the connector is Type-C interface for DC 5V power input
40-Pin Double-Row Headers	<ul style="list-style-type: none"> ● 1 x I2C, 2 x USB 2.0 Host, 4 x 12bit ADC in, 1 x UART Debug ● Default 6 x GPIO, which can be configured to PWM, UART, SPI, I2C, CAN via software ● 5V power input/output, 1.8V/3.3V@300mA power output, system reset, ON/OFF
LED & Key	<ul style="list-style-type: none"> ● 1 x ACT LED (Green) ● 1 x PWR LED (Red and Blue) ● 1 x ON/OFF Key ● 1 x Reset Key
DIP Switch	1 x DIP Switch 3
Slot	1 x Micro SD slot
Power Supply	
Power Input	Default DC 5V/2A power input, the connector is Type-C interface
Mechanical & Environmental	
Size (L x W)	85.0mm x 56.0mm (±0.5mm)
Weight	43g (±0.5g)
Operating Temp.	<ul style="list-style-type: none"> ● Industrial grade: -20°C~70°C ● Industrial grade: -40°C~85°C (Wide temperature optional)

2.2. Composition

DEBIX Model C consists of a range of different computer components. The most important component is the "brain" of the computer, the system-on-chip (SoC) in the center of the motherboard.

The SoC contains most of the components of the computer, often containing both the central processing unit (CPU) and the graphics processing unit (GPU). DEBIX Model C has Random Memory (RAM), eMMC (reserved), WiFi Bluetooth module that contains the wireless communication components, and the PMIC (PCA9451AHN) that manages the power devices of the host machine, as shown in the following figure:

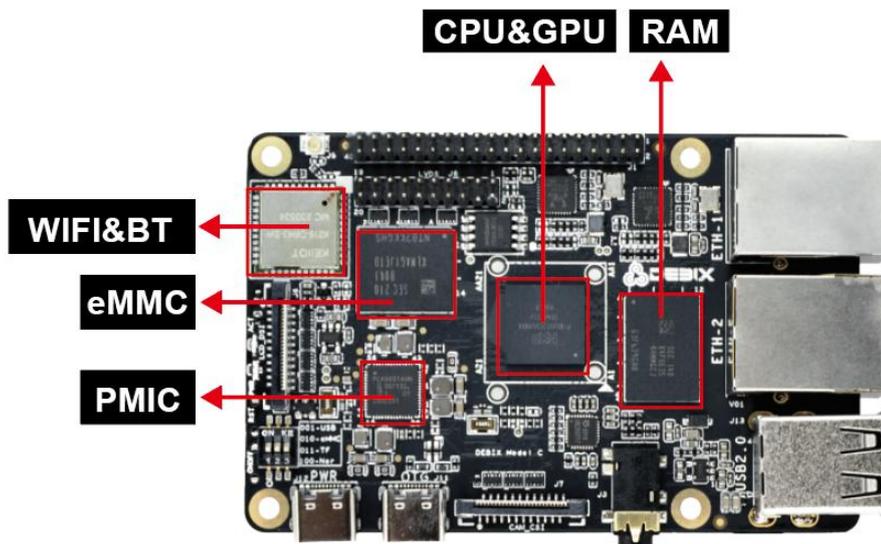


Figure 4 DEBIX Model C

2.3. Interface

2.3.1. Power Interface

DEBIX Model C provides a USB Type-C power interface (J12) with default DC 5V/2A voltage.

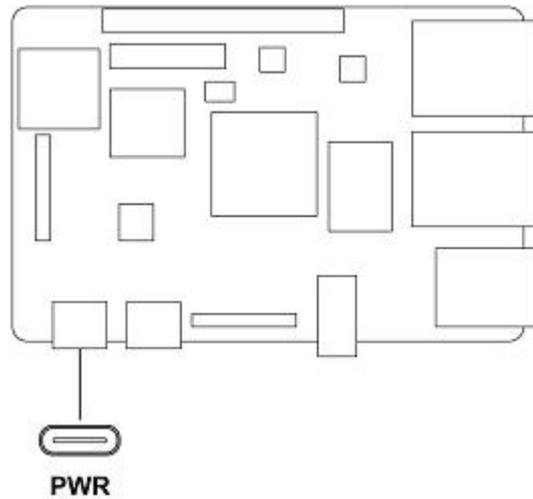


Figure 5 Power Interface

2.3.2. USB Interface

DEBIX Model C has two USB controllers and PHY, supports USB 2.0.

- 2 x USB 2.0 Host with double layer Type-A interface (J13)
- 2 x USB 2.0 with Type-C interface, one is to DC 5V power input, and one is a OTG interface (J11) which can be used for programming, system updating, or USB drive & hard disk connecting etc.

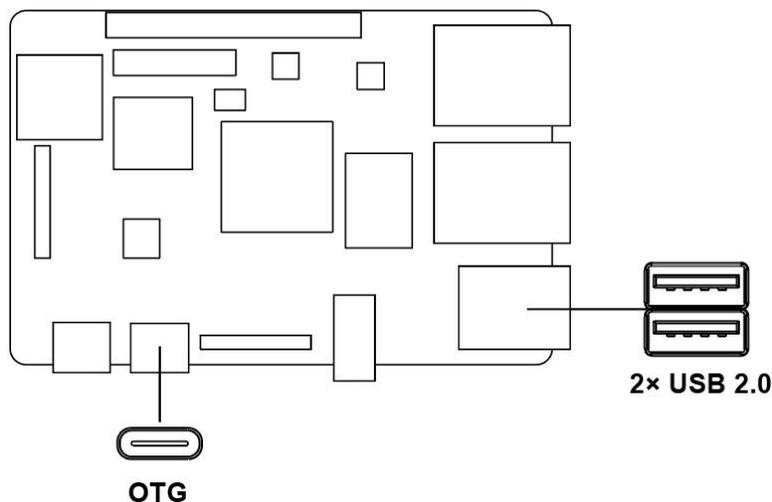


Figure 6 OTG and USB 2.0 Host

2.3.3. Ethernet Interface

DEBIX Model C implements two Ethernet controllers, both of which can operate synchronously. There are two Gigabit Ethernet interfaces onboard, both of which are independent MAC Gigabit Ethernet port:

- One independent MAC Gigabit Ethernet port (J4), supports POE power supply (need POE power supply module): ENET_QOS (Ethernet Quality of Service) (ETH1), based on Synopsys proprietary, supports time-sensitive networking (TSN), EEE, Ethernet AVB (IEEE802.1Qav), IEEE1588.
- One independent MAC Gigabit Ethernet port (J5): ENET1 (ETH2), Gigabit Ethernet controller, supports EEE, Ethernet AVB (IEEE802.1Qav), IEEE1588 time stamp module, the time stamp module is distributed control for industrial automation applications nodes provide accurate clock synchronization.

Connect DEBIX Model C to the network through the network cable of the RJ45 connector, and a set of status indicators below the interface to display the status signal, green indicator is Link and yellow indicator is Active.



Figure 7 Ethernet Interface

Table 4 Description of Gigabit Ethernet Port Status Indicator

LED	Color	Description
Link	Green	Light, the network cable is plugged in, network connection status is good
Active	Yellow	Blinking, network data is being transmitted

2.3.4. Display Interface

2.3.4.1. LVDS Interface

The LVDS display bridge (LDB) connects to an External LVDS Display Interface. The purpose of the LDB is to support flow of synchronous RGB data to external display devices through the LVDS interface.

DEBIX Model C provides one 2 x 10Pin LVDS display output interface (J8) driven by LDB to support single LVDS display.

- Supports FPD link.
- Single channel (4 lanes) 80MHz pixel clock and LVDS clock output. It supports resolutions up to 1366x768p60 or 1280x800p60.
- Supports VESA and JEIDA pixel mapping.
- Supports LVDS Transmitter with four 7-bit channels. Each channel sends the 6 pixel bits and one control signal at 7 times the pixel clock rate. The data and control signals are transmitted over an LVDS link.

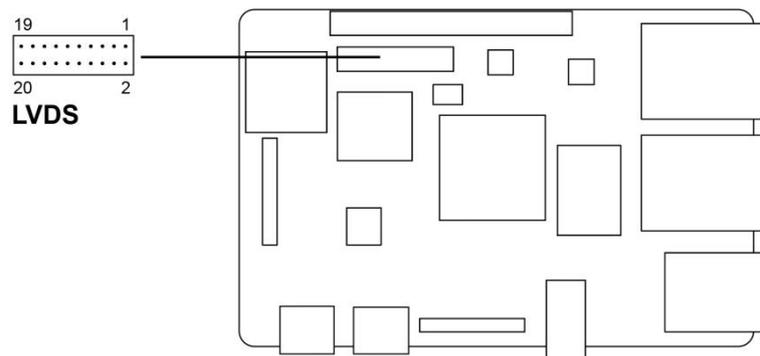


Figure 8 LVDS Interface

The pin sequence is as shown in the figure:

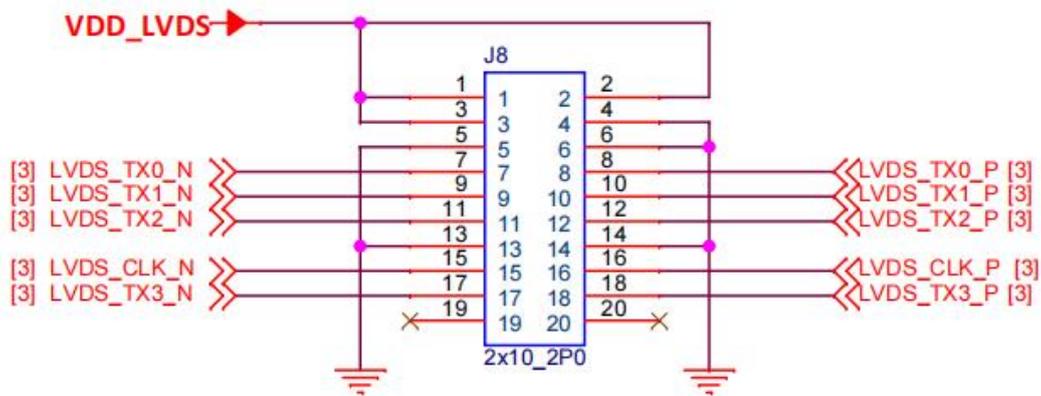


Figure 9 Pin sequence of LVDS

The interface is defined as follows:

Table 5 Pin definition of LVDS

Pin	Definition	Description
1	VDD_LVDS	Default 5V (3.3V,5V optional)
2	VDD_LVDS	Default 5V (3.3V,5V optional)
3	VDD_LVDS	Default 5V (3.3V,5V optional)
4	GND	To Ground
5	GND	To Ground
6	GND	To Ground
7	LVDS_TX0_N	LVDS0 Differential data channel 0 (-)
8	LVDS_TX0_P	LVDS0 Differential data channel 0 (+)
9	LVDS_TX1_N	LVDS0 Differential data channel 1 (-)
10	LVDS_TX1_P	LVDS0 Differential data channel 1 (+)
11	LVDS_TX2_N	LVDS0 Differential data channel 2 (-)
12	LVDS_TX2_P	LVDS0 Differential data channel 2 (+)
13	GND	To Ground
14	GND	To Ground
15	LVDS_CLK_N	LVDS Clock differential signal path (-)
16	LVDS_CLK_P	LVDS Clock differential signal path (+)

17	LVDS_TX3_N	LVDS Differential data channel 3 (-)
18	LVDS_TX3_P	LVDS Differential data channel 3 (+)
19	Not used	-
20	Not used	-

2.3.4.2. MIPI DSI

DEBIX Model C provides a MIPI DSI interface (J6) with a 24Pin/0.5mm Pitch FPC socket connector, which can be used to connect a MIPI display touch screen.

Key features of MIPI DSI include:

- MIPI DSI compliant with MIPI-DSI specification V1.2 and MIPI-DPHY specification v1.2
- Maximum resolution limited to resolutions achievable with a 200MHz pixel clock and active pixel rate of 140Mpixel/s with 24-bit RGB. This includes resolutions such as: 1080p60 or 1920x1200p60
- Support up to 4 Tx data lanes (plus 1 Tx clock lane)
- Support 80Mbps - 1.5Gbps data rate per lane in high speed operation
- Support 10Mbps data rate in low power operation

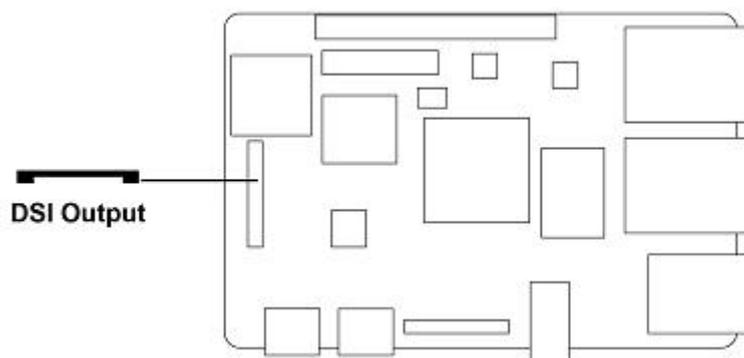


Figure 10

The pin sequence is as shown in the figure:

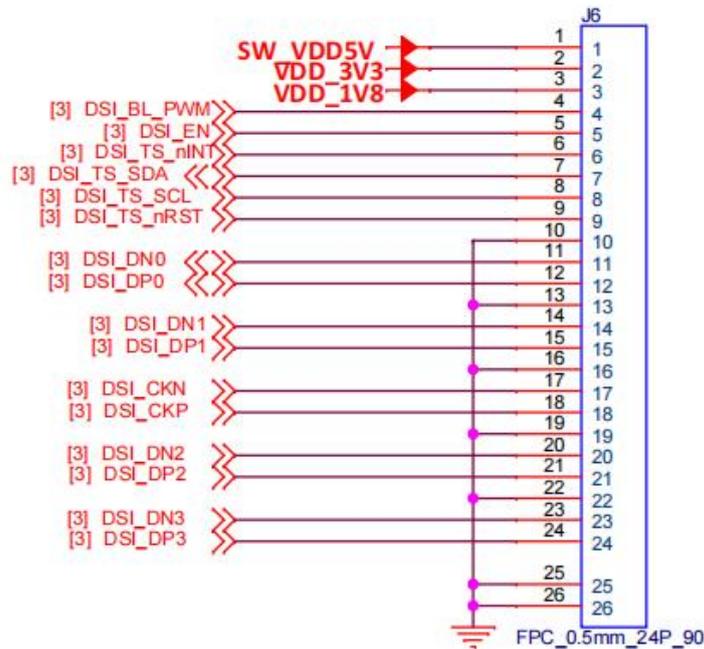


Figure 11 Pin sequence of MIPI DSI

The interface is defined as follows:

Table 6 Pin definition of MIPI DSI

Pin	Definition	Description
1	VDD_5V	5V output
2	VDD_3V3	3.3V output
3	VDD_1V8	1.8V output
4	DSI_BL_PWM	Backlight control signal
5	DSI_EN	LCD enable signal
6	DSI_TP_nINT	touch interrupt pin
7	DSI_I2C_SDA	Touch the clock terminal of I2C (controlled by I2C2)
8	DSI_I2C_SCL	Touch the clock terminal of I2C (controlled by I2C2)
9	DSI_TS_nRST	IO control pin
10	GND	To Ground
11	DSI_DN0	DSI Differential data channel 0 (-)
12	DSI_DP0	DSI Differential data channel 0 (+)

13	GND	To Ground
14	DSI_DN1	DSI Differential data channel 1 (-)
15	DSI_DP1	DSI Differential data channel 1 (+)
16	GND	To Ground
17	DSI_CKN	DSI Differential Clock Channels (-)
18	DSI_CKP	DSI Differential Clock Channels (+)
19	GND	To Ground
20	DSI_DN2	DSI Differential data channel 2 (-)
21	DSI_DP2	DSI Differential data channel 2 (+)
22	GND	To Ground
23	DSI_DN3	DSI Differential data channel 3 (-)
24	DSI_DP3	DSI Differential data channel 3 (+)
25	GND	To Ground
26	GND	To Ground

2.3.5. MIPI CSI

DEBIX Model C has a MIPI CSI-2 Host controller. This controller implements the protocol functions defined in the MIPI CSI-2 specification, allowing camera sensor communication consistent with MIPI CSI-2.

The MIPI CSI-2 controller has the following features:

- PHY-Protocol Interface (PPI) Pattern Generator with programmable packet-to-packet time
- Configurable pipeline interface (1 pipeline stage) between the PHY and MIPI CSI-2 controller
- Support for automatic D-PHY integration in non-automotive configurations
- Programmable value for the number of synchronization stages used for Clock

Crossing Domain (CDC)

- Image Pixel Interface (IPI)
 - Two operating modes:
 - **Camera Timing** - The frame timing signals, and the vertical or horizontal synchronism are generated based on the synchronization of Short Packets received from the sensor.
 - **Controller Timing** - The frame timing signals are generated based on the IPI registers.
 - Generates pixel stream in two different modes:
 - 48-Bit
 - 16-Bit
 - Supports several data formats:
 - RGB
 - YUV
 - RAW
 - User defined
 - Embedded data (when operating in Camera Timing mode and only with RAW image data)
 - Data decoding based on configurable data type
 - Additional pins that provide useful information:
 - End-of-Line indication
 - Number of valid pixels/bytes transmitted per clock cycle
 - First and Last Data Valid Indications
 - End-of-Frame indication
 - Possibility to flush IPI memory (automatically or manually)
 - Possibility to ignore Frame Start as a synchronization event
 - Possibility to select Packets used for IPI Synchronism Events
 - Possibility to reduce memory requirements, down to the minimum FIFO depth

of 32

■ Back-pressure mechanism

There is a MIPI CSI interface (J7) on board, with a 24Pin/0.5mm Pitch FPC socket connector for connecting DEBIX camera module. Data transfer rates up to 1.5 Gbps per channel.

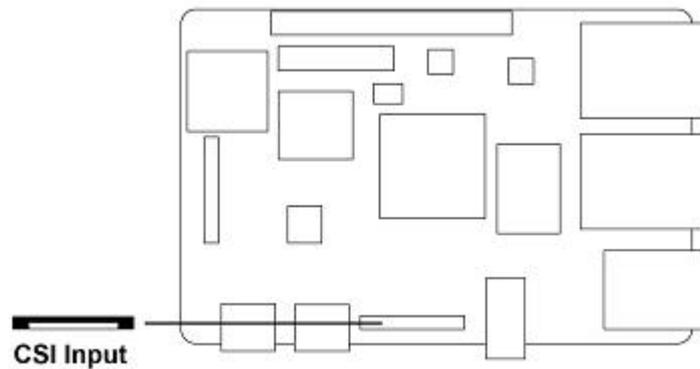


Figure 12 MIPI CSI

The pin sequence is as shown in the figure:

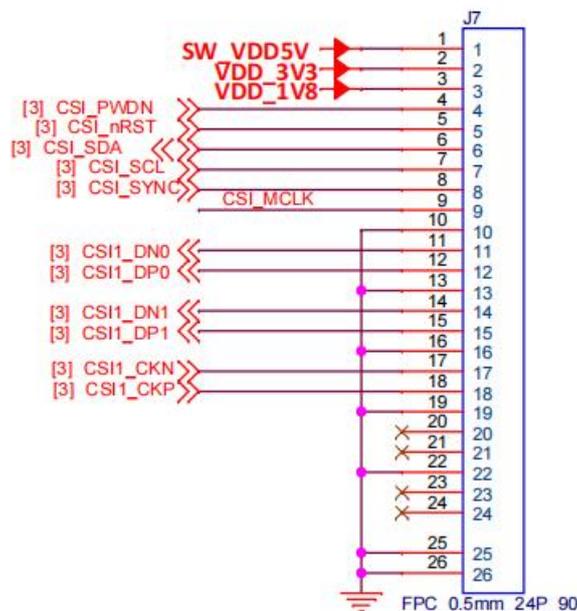


Figure 13 Pin sequence of MIPI CSI

The interface is defined as follows:

Table 7 Pin definition of MIPI CSI

Pin	Definition	Description
1	VDD_5V	5V output
2	VDD_3V3	3.3V output
3	VDD_1V8	1.8V output
4	CSI_PWDN	CSI low power mode
5	CSI_nRST	CSI reset signal
6	CSI_SDA	CSI data signal
7	CSI_SCL	CSI clock signal
8	CSI_SYNC	CSI synchronization signal
9	CSI_MCLK	CSI external clock input
10	GND	To Ground
11	CSI1_DN0	CSI Differential data channel 0 (-)
12	CSI1_DP0	CSI Differential data channel 0 (+)
13	GND	To Ground
14	CSI1_DN1	CSI Differential data channel 1 (-)
15	CSI1_DP1	CSI Differential data channel 1 (+)
16	GND	To Ground
17	CSI1_CKN	CSI Differential Clock Channels (-)
18	CSI1_CKP	CSI Differential Clock Channels (+)
19	GND	To Ground
20	Not used	-
21	Not used	-
22	GND	To Ground
23	Not used	-
24	Not used	-
25	GND	To Ground

26	GND	To Ground
----	-----	-----------

2.3.6. Audio

DEBIX Model C provides a combined headphone and microphone input interface (J3), the connector is 3.5mm socket, with audio in/out function, and supports rated voltage 1.5V MIC audio input.

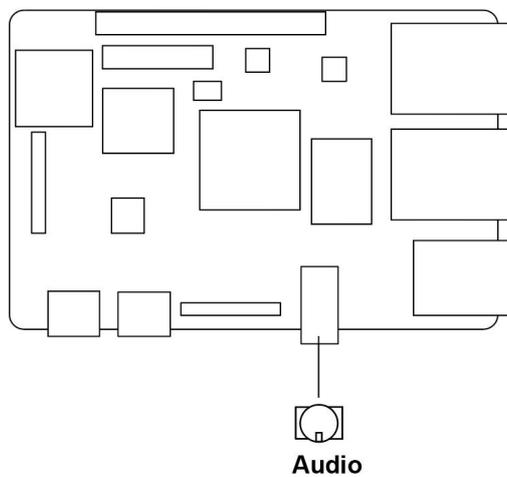


Figure 14 Audio

NOTE

DEBIX uses MIC and only supports four-segment headphones. The definition is shown in the following figure, which includes left channel, right channel, GND, and MIC recording. It is necessary to connect to the DEBIX audio interface according to the GND and MIC connection lines for normal use.



Figure 15 Definition of four-segment headphones

2.3.7. GPIO

DEBIX Model C has a set of 2*20Pin/2.0mm GPIO interface (J1), which can be used for external hardware such as LED, button, sensor, function modules, etc.

- The voltage of I2C, UART (default for Debug), CAN, SPI, GPIO pin is 3.3V.
- The voltage of ADC IN is 1.8V.
- 5V pins (pin6, pin8) can be used to power to DEBIX Model C or peripherals.

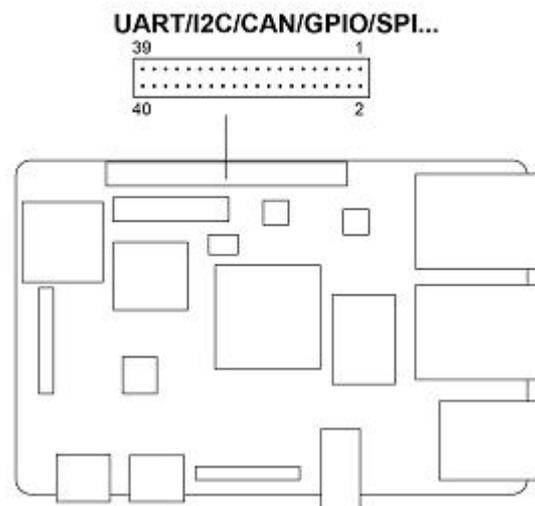


Figure 16 40Pin

The pin sequence is as shown in the figure:

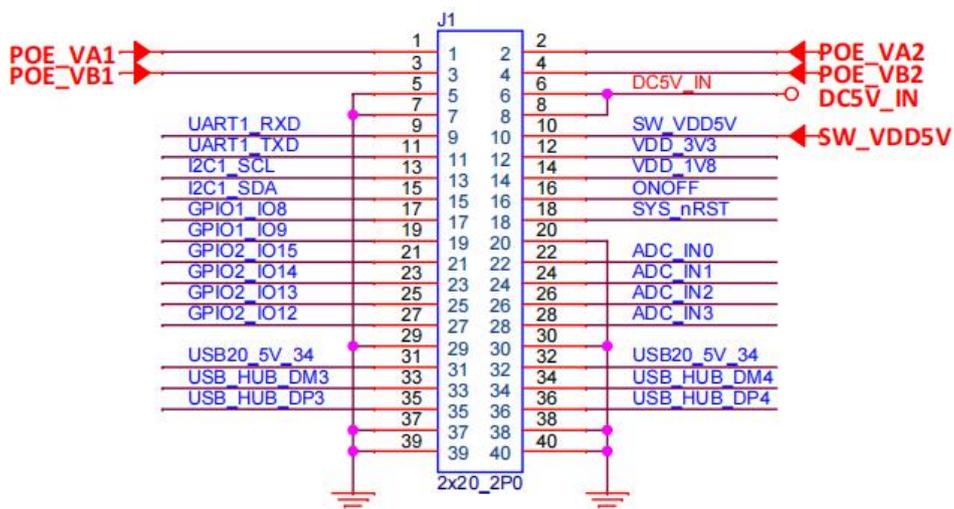


Figure 17 Pin sequence of J1

The interface is defined as follows:

Table 8 Pin definition of GPIO

Pin	Definition	Pin	Definition
1	POE_VA1	2	POE_VA2
3	POE_VB1	4	POE_VB2
5	GND	6	DC5V_IN
7	GND	8	DC5V_IN
9	UART1_RXD	10	SW_VDD5V
11	UART1_TXD	12	VDD_3V3
13	I2C1_SCL	14	VDD_1V8
15	I2C1_SDA	16	ONOFF
17	GPIO1_IO08	18	SYS_nRST
19	GPIO1_IO09	20	GND
21	GPIO2_IO15	22	ADC_IN0
23	GPIO2_IO14	24	ADC_IN1
25	GPIO2_IO13	26	ADC_IN2
27	GPIO2_IO12	28	ADC_IN3
29	GND	30	GND
31	USB20_5V_34	32	USB20_5V_34
33	USB_HUB_DM3	34	USB_HUB_DM4
35	USB_HUB_DP3	36	USB_HUB_DP4
37	GND	38	GND
39	GND	40	GND

For detailed GPIO function MUX, please refer to "[GPIO Pin Multiplexing Function List](#)" from DEBIX Model C Documents.

2.3.8. LED & Key

DEBIX Model C has two LED indicators and two Keys.

- LED

- 1 x ACT LED (Green)
- 1 x Power LED (Red and Blue)
- Key
 - 1 x ON/OFF Key
 - 1 x Reset Key

The specific states are described in the following table:

Table 9 Description of LED & Key

Function Name		Status	Description
LED	Power LED	Lighting	Power is on, and red & blue light
		off	Power is off, and red & blue change to red, until off
	ACT LED	Blinking	System is normal
		off	System fault
Key	ON/OFF Key	Short press	Sleep/Wake
		Long press	Power off/on
	RESET Key	Press	System reset

2.3.9. DIP Switch

There is a dip-switch combination, which is used to determine the BOOT startup mode. There are three switches in total, and each switch has the two states of ON/OFF. By default, the switch is turned ON. Four BOOT startup modes as follows:

- 001-USB burning mode
- 010-eMMC Boot
- 011-Micro SD Card Boot
- 100-SPI Nor Flash boot

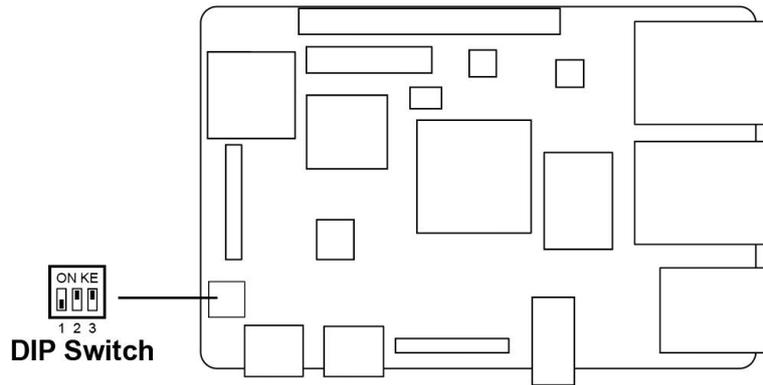


Figure 18 DIP Switch

The selected boot mode is shown in the table below:

Table 10 DIP switch set boot mode

Mod e Switch	USB	eMMC	Micro SD	Nor Flash
SW state setting				
<p>Note: The switch is facing up, it is ON state, the switch is facing down, it is OFF state.</p>				

2.3.10. Slot

DEBIX Model C provides a Micro SD slot (J2), Micro SD card boot mode by default, Micro SD card can be used as a system boot card, insert the Micro SD card with the system installed here, and then power on DEBIX to start the system in the Micro SD card.

The Micro SD card can also be used as a standard memory card to save user data.

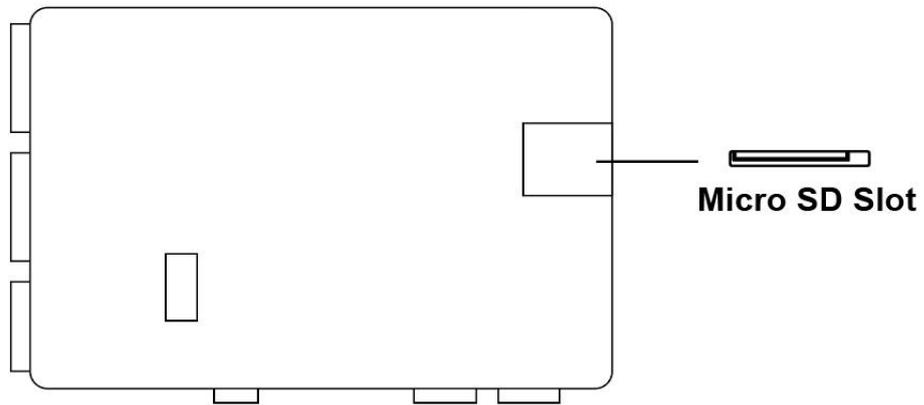


Figure 19 Micro SD Card Slot

2.4. Packing List

- DEBIX Model C (without eMMC by default)

Chapter 3 Get Started

3.1. Software Installation

3.1.1. Download Image

1. Download the latest system image from the [software download page](#) of DEBIX official website;

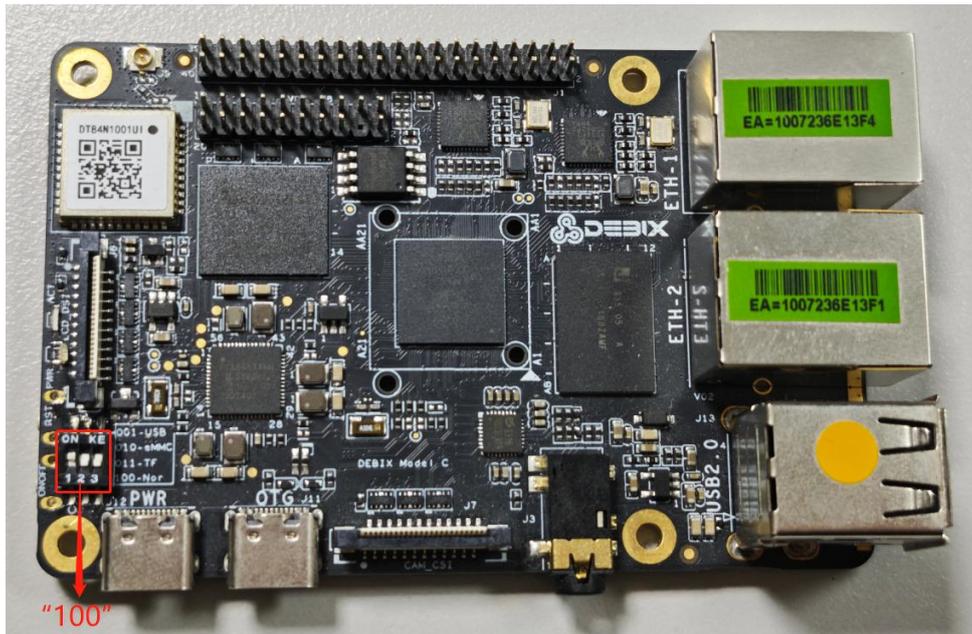
IMPORTANT

The boot type of the image downloaded depends on which boot mode image you choose to install, and whether or not the board contains eMMC, etc. For example, if you need to install an image with eMMC boot mode, and the board has an eMMC module, you can choose the image name with (boot from eMMC).

2. If the downloaded image file is a zip file, you need to decompress it into an .img file;
3. Write the .img file into the Micro SD card by [balenaEtcher](#) tool.

3.1.2. System Boot

The DEBIX Model C motherboard defaults to NOR Flash boot. To enable NOR Flash boot, set the DIP switch to "100". Upon successful NOR Flash initialization, the system loads the U-Boot bootloader. At this point, the user may select the boot modes (SD card boot or eMMC boot).



3.1.2.1. Flashing Micro SD Image

- **Component Preparation**

- ✓ DEBIX Model C
- ✓ Micro SD card, and card reader
- ✓ DC 5V/2A power adapter
- ✓ PC (windows 10/11)

- **Installing Micro SD Card Image**

Click [**Boot from SD Card**] to download the Micro SD card image for DEBIX Model C from [software download page](#) on your PC.

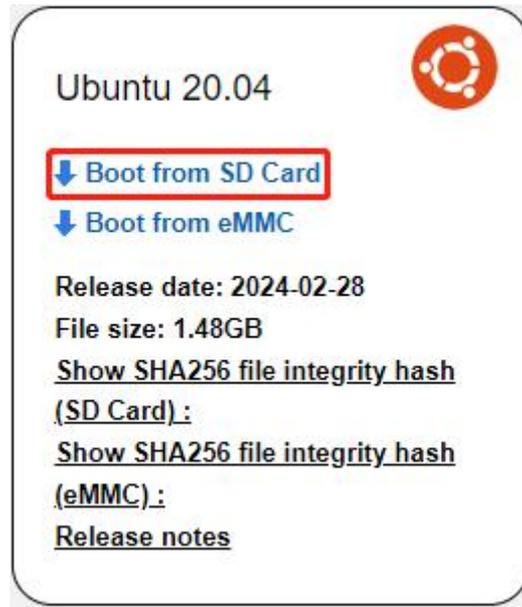


Figure 20

1. Install and open the Etcher tool on your PC, insert the Micro SD card, select the img file to be installed and the disk partition corresponding to the Micro SD card;

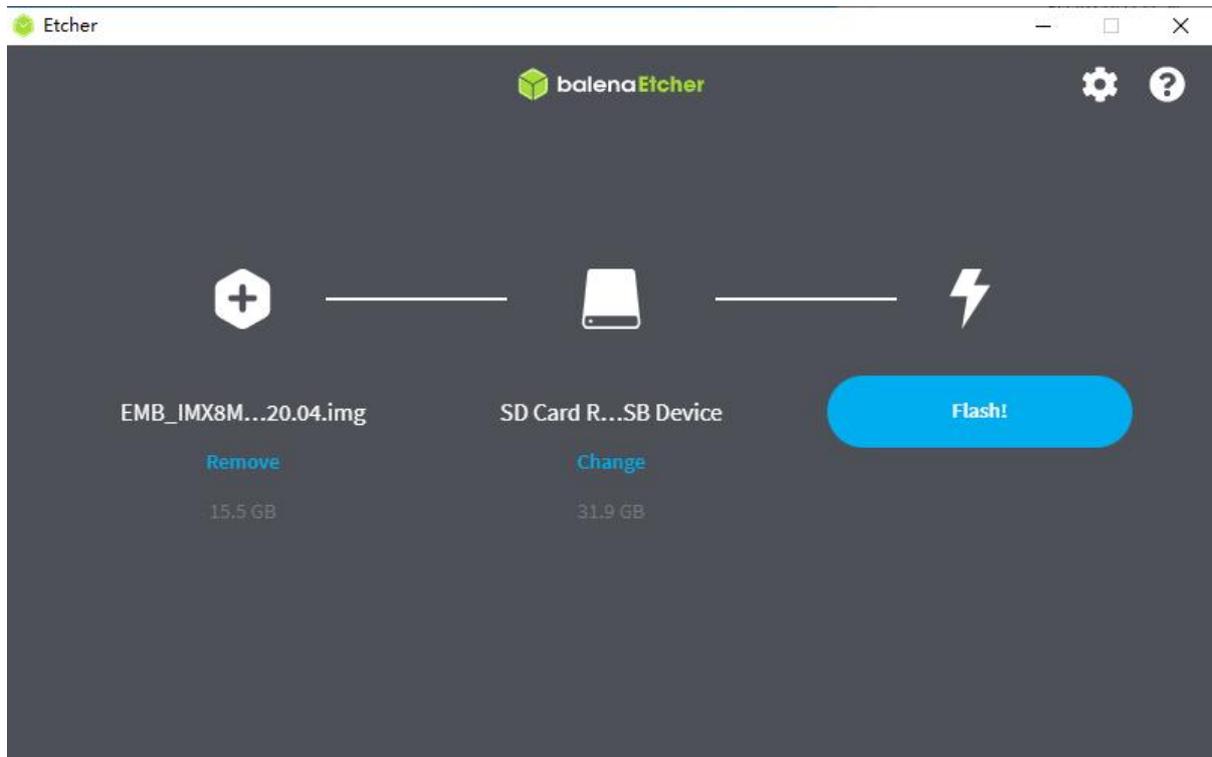


Figure 21

2. Click **[Flash!]** Wait patiently and the program will write the system to the Micro SD

card;

NOTE

The system may prompt you that the disk is unavailable and needs to be formatted, please ignore it, it is not an error!

3. When **Flash Complete!** appears, it means the system has been successfully programmed to the Micro SD card;

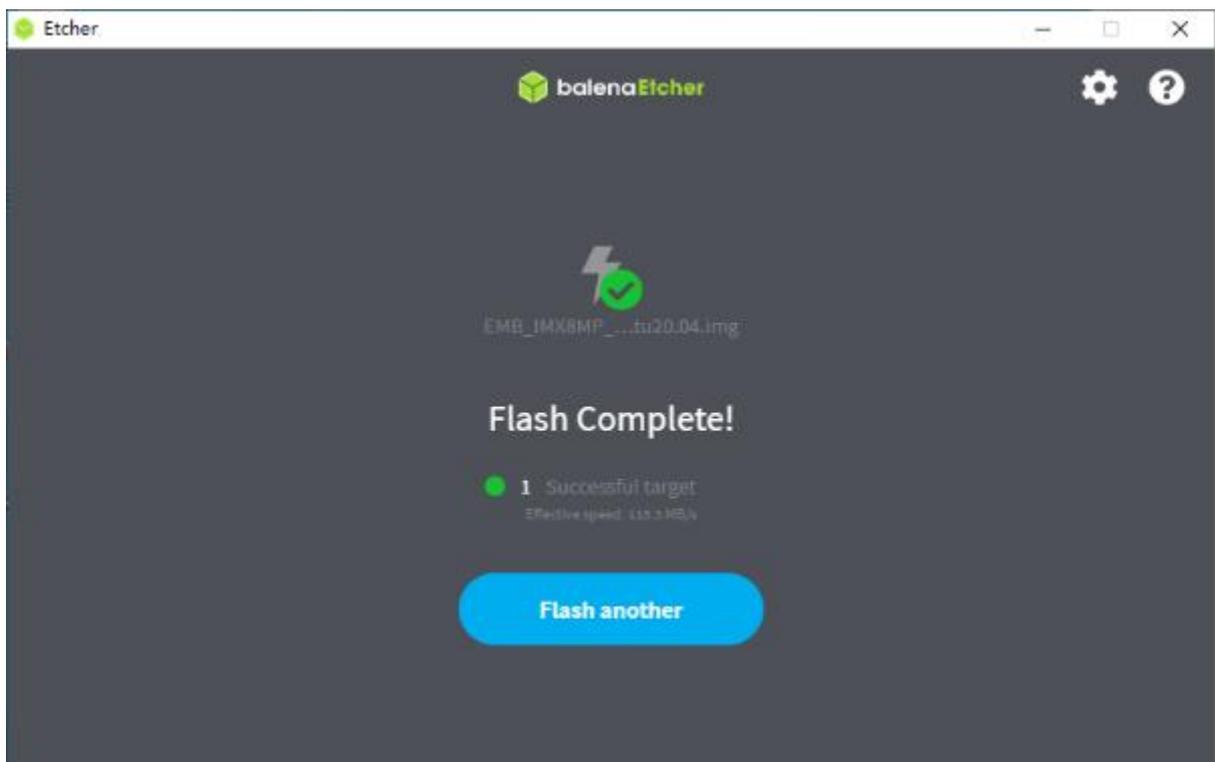


Figure 22

3.1.2.2. Flashing eMMC Image

- **Component Preparation**
 - ✓ DEBIX Model C
 - ✓ Micro SD card, and card reader
 - ✓ DC 5V/2A power adapter

✓ PC (windows 10/11)

- **Installing eMMC Image**

IMPORTANT

For the default configuration, you need to select an eMMC module when purchasing.

Click [**Boot from eMMC**] to download the eMMC image for DEBIX Model C from [software download page](#).

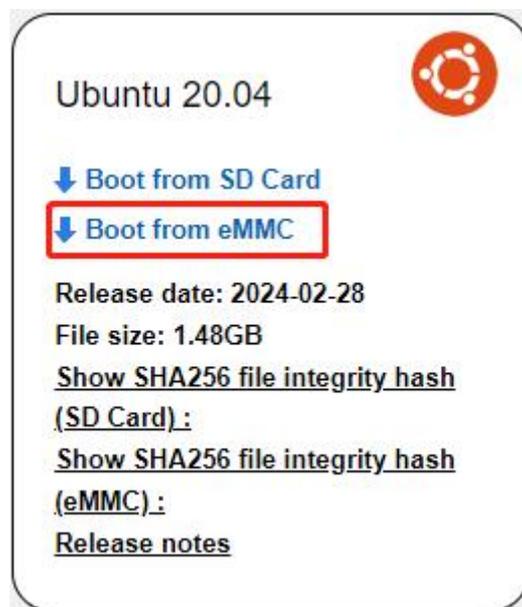


Figure 23

1. Write the downloaded system image to the Micro SD card according to [the steps 1-3 operation of "Installing Micro SD Card Image"](#).
2. Insert the Micro SD card into DEBIX Model C and power on. After booting, the system will automatically write to eMMC through the Micro SD card. When burning, the green LED on the motherboard will flash quickly, please wait. When the green LED changes from fast flash to slow flash, that is, the programming is complete.

3.2. Hardware Installation

Hardware connections are made as shown in the diagram and the steps are as follows:

1. **Insert the Micro SD card with the system installed:** Insert it into the slot on the back of DEBIX Model C; if you need to remove it, just gently pull out the card after power off.
2. **Connect the LVDS screen**
3. **Connect the keyboard**
4. **Connect the mouse**
5. **Connect the network cable**
6. **Connect the power adapter:** Plug in the power supply, DEBIX Model C will power on, and the power indicator light (red and blue) of motherboard will be on, and system indicator light (green) will be blinking.

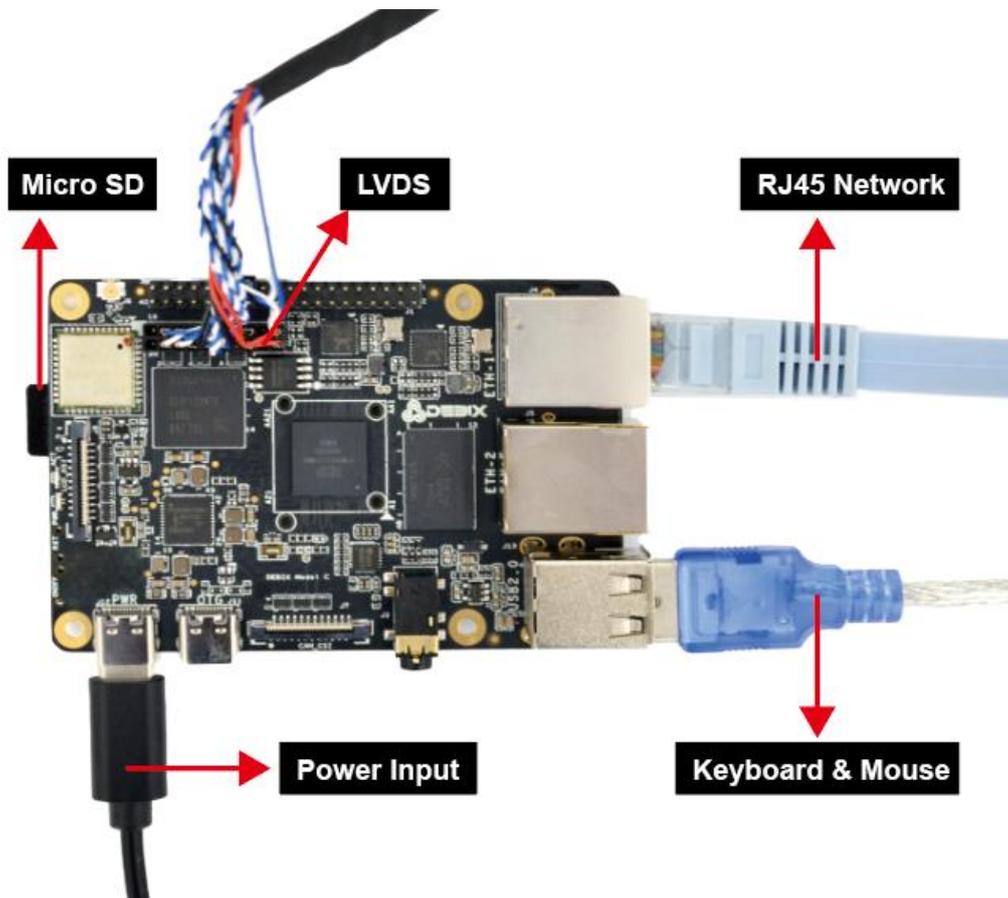


Figure 24 Hardware connection

Chapter 4 Software Application Examples

4.1. Switch Boot Mode

IMPORTANT

The UEFI selection timeout is 3s, if no selection is made within this period, it will automatically enter the last selected system.

When DEBIX Model C has eMMC and Micro SD card, and both contain systems, you can switch the boot mode in the following way:

1. Select the boot mode through the serial port, when “select: SD boot” appears, you can select it via the direction control key and press **Enter** to finish the selection.

There are 4 options:

- select: SD boot
- select: emmc boot
- select: Reboot
- select: About

2. After entering the system, select the boot mode via `Debix_boot` command:

For example, both Micro SD card and eMMC can be installed with yocto system, type the number to select as below:

```
root@DebixModelC:~# Debix_boot
=====
=====
=====
Select Startup Items:
    1 = SD yocto_sd.efi
    2 = Emmc yocto.efi
Waiting for input: █
```

4.2. Usage of Ethernet

- Network port 1 (ENET_QOS), bit number: J4, device node: eth0, device silkscreen: ETH-1
1. Enter the system desktop, open a terminal and type the command to query network port 1;

```
ifconfig eth0
```

```
root@DebixModelC:~# ifconfig eth0
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
    inet 192.168.1.17  netmask 255.255.255.0  broadcast 192.168.1.255
    inet6 fe80::b479:d5ff:fe64:c188  prefixlen 64  scopeid 0x20<link>
    inet6 240e:36d:dda:2400:b479:d5ff:fe64:c188  prefixlen 64  scopeid 0x0<global>
    ether b6:79:d5:64:c1:88  txqueuelen 1000  (Ethernet)
    RX packets 3446  bytes 283890 (277.2 KiB)
    RX errors 0  dropped 376  overruns 0  frame 0
    TX packets 2222  bytes 116792 (114.0 KiB)
    TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0
    device interrupt 106

root@DebixModelC:~#
```

2. Query the speed of network port 1;

```
ethtool eth0
```

```
root@DebixModelC:~# ethtool eth0
Settings for eth0:
    Supported ports: [ TP      MII ]
    Supported link modes:   10baseT/Half 10baseT/Full
                          100baseT/Half 100baseT/Full
                          1000baseT/Full
    Supported pause frame use: Symmetric Receive-only
    Supports auto-negotiation: Yes
    Supported FEC modes: Not reported
    Advertised link modes:  10baseT/Half 10baseT/Full
                          100baseT/Half 100baseT/Full
                          1000baseT/Full
    Advertised pause frame use: Symmetric Receive-only
    Advertised auto-negotiation: Yes
    Advertised FEC modes: Not reported
    Link partner advertised link modes:  10baseT/Half 10baseT/Full
                                         100baseT/Half 100baseT/Full
                                         1000baseT/Full
    Link partner advertised pause frame use: Symmetric
    Link partner advertised auto-negotiation: Yes
    Link partner advertised FEC modes: Not reported
    Speed: 1000Mb/s
    Duplex: Full
    Auto-negotiation: on
```

- Network port 2 (ENET1), bit number: J5, port number: eth1, device silkscreen: ETH-2

1. Type the command to query network port 2;

```
ifconfig eth1
```

```
root@DebixModelC:~# ifconfig eth1
eth1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.1.27 netmask 255.255.255.0 broadcast 192.168.1.255
    inet6 fe80::f454:48ff:fe42:f5a7 prefixlen 64 scopeid 0x20<link>
    inet6 240e:36d:dda:2400:f454:48ff:fe42:f5a7 prefixlen 64 scopeid 0x0<global>
    ether f6:54:48:42:f5:a7 txqueuelen 1000 (Ethernet)
    RX packets 3697 bytes 357082 (348.7 KiB)
    RX errors 0 dropped 393 overruns 0 frame 0
    TX packets 2189 bytes 106811 (104.3 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@DebixModelC:~#
```

2. Query the speed of network port 2;

```
ethtool eth1
```

```
root@DebixModelC:~# ethtool eth1
Settings for eth1:
    Supported ports: [ TP      MII ]
    Supported link modes:   10baseT/Half 10baseT/Full
                           100baseT/Half 100baseT/Full
                           1000baseT/Full

    Supported pause frame use: Symmetric
    Supports auto-negotiation: Yes
    Supported FEC modes: Not reported
    Advertised link modes:  10baseT/Half 10baseT/Full
                           100baseT/Half 100baseT/Full
                           1000baseT/Full

    Advertised pause frame use: Symmetric
    Advertised auto-negotiation: Yes
    Advertised FEC modes: Not reported
    Link partner advertised link modes:  10baseT/Half 10baseT/Full
                                         100baseT/Half 100baseT/Full
                                         1000baseT/Full

    Link partner advertised pause frame use: Symmetric
    Link partner advertised auto-negotiation: Yes
    Link partner advertised FEC modes: Not reported
    Speed: 1000Mb/s
    Duplex: Full
    Auto-negotiation: on
```

3. Check the network connection status via ping command.

```
ping 192.168.1.1
```

```
root@DebixModelC:~# ping 192.168.1.1
PING 192.168.1.1 (192.168.1.1) 56(84) bytes of data:
64 bytes from 192.168.1.1: icmp_seq=1 ttl=64 time=1.07 ms
64 bytes from 192.168.1.1: icmp_seq=2 ttl=64 time=1.08 ms
64 bytes from 192.168.1.1: icmp_seq=3 ttl=64 time=1.07 ms
64 bytes from 192.168.1.1: icmp_seq=4 ttl=64 time=1.07 ms
64 bytes from 192.168.1.1: icmp_seq=5 ttl=64 time=1.08 ms
64 bytes from 192.168.1.1: icmp_seq=6 ttl=64 time=1.08 ms
64 bytes from 192.168.1.1: icmp_seq=7 ttl=64 time=1.08 ms
64 bytes from 192.168.1.1: icmp_seq=8 ttl=64 time=1.08 ms
```

4.3. Usage of WiFi

WiFi device node for DEBIX Model C: wlan0.

1. Unplug the network cable and connect to WiFi (name: polyhex_mi) via the command:

```
connmanctl  
enable wifi  
scan wifi  
services  
agent on  
connect xxx_psk #connect available wifi name, type wifi password
```

```
connmanctl> enable wifi  
wifi is already enabled  
connmanctl> scan wifi  
Scan completed for wifi  
connmanctl> services  
*AR Wired ethernet_1a9427328710 cable  
polyhex_mi1 wifi_ac6aa32c009b_706f6c796865785f6d6931_managed_psk  
wifi_ac6aa32c009b_hidden_managed_psk  
tsc wifi_ac6aa32c009b_747363_managed_psk  
ChinaNet-polyhex wifi_ac6aa32c009b_4368696e614e65742d706f6c79686578_mana  
ged_psk  
polyhex-3 wifi_ac6aa32c009b_706f6c796865782d33_managed_psk
```

```
connmanctl> agent on  
Agent registered  
connmanctl> connect wifi_ac6aa32c009b_706f6c796865785f6d6931_managed_psk  
Agent RequestInput wifi_ac6aa32c009b_706f6c796865785f6d6931_managed_psk  
Passphrase = [ Type=psk, Requirement=mandatory, Alternates=[ WPS ] ]  
PreviousPassphrase = [ Type=psk, Requirement=informational, Value=bohais2021 ]  
WPS = [ Type=wpspin, Requirement=alternate ]  
Passphrase? bohais2021  
connmanctl> [ 5892.058682] IPv6: ADDRCONF(NETDEV_CHANGE): wlan0: link becomes ready  
Connected wifi_ac6aa32c009b_706f6c796865785f6d6931_managed_psk  
connmanctl> █
```

2. Query the WiFi network port

```
ifconfig wlan0
```

```
root@DebixModelC:~# ifconfig wlan0  
wlan0: flags=-28605<UP,BROADCAST,RUNNING,MULTICAST,DYNAMIC> mtu 1500  
inet 192.168.31.121 netmask 255.255.255.0 broadcast 192.168.31.255  
inet6 fe80::ae6a:a3ff:fe2c:9b prefixlen 64 scopeid 0x20<link>  
ether ac:6a:a3:2c:00:9b txqueuelen 1000 (Ethernet)  
RX packets 26 bytes 3180 (3.1 KiB)  
RX errors 0 dropped 0 overruns 0 frame 0  
TX packets 61 bytes 7527 (7.3 KiB)  
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  
root@DebixModelC:~# █
```

3. Check the WiFi network connection status via ping command.

```
ping 192.168.1.1
```

```
root@DebixModelC:~# ping 192.168.1.1
PING 192.168.1.1 (192.168.1.1) 56(84) bytes of data.
64 bytes from 192.168.1.1: icmp_seq=1 ttl=63 time=35.2 ms
64 bytes from 192.168.1.1: icmp_seq=2 ttl=63 time=5.58 ms
64 bytes from 192.168.1.1: icmp_seq=3 ttl=63 time=5.56 ms
64 bytes from 192.168.1.1: icmp_seq=4 ttl=63 time=5.89 ms
64 bytes from 192.168.1.1: icmp_seq=5 ttl=63 time=5.45 ms
64 bytes from 192.168.1.1: icmp_seq=6 ttl=63 time=4.95 ms
64 bytes from 192.168.1.1: icmp_seq=7 ttl=63 time=4.65 ms
```

4.4. Usage of BT

Bluetooth device node for DEBIX Model C: hci0.

1. Enter the system desktop, open a terminal and type the command to query BT device;

```
hciconfig
```

```
root@DebixModelC:~# hciconfig
hci0:  Type: Primary  Bus: UART
      BD Address: AC:6A:A3:2C:00:9C  ACL MTU: 1021:8  SCO MTU: 64:1
      DOWN
      RX bytes:2338 acl:0 sco:0 events:195 errors:0
      TX bytes:37598 acl:0 sco:0 commands:195 errors:0

root@DebixModelC:~# █
```

2. Start bluetooth and match bluetooth.

```
hciconfig hci0 up
bluetoothctl
power on
agent on
default-agent
scan on
pair yourDeviceMAC #Match the Bluetooth MAC address of the device
```

```
# bluetoothctl
Agent registered
[CHG] Controller AC:6A:A3:15:23:40 Pairable: yes
[bluetooth]# power on
Changing power on succeeded
[bluetooth]# agent on
Agent is already registered
[bluetooth]# default-agent
Default agent request successful
[bluetooth]# scan on
Discovery started
[CHG] Controller AC:6A:A3:15:23:40 Discovering: yes
[NEW] Device 6F:77:E4:55:30:6B 6F-77-E4-55-30-6B
[NEW] Device 58:1F:3E:7C:17:CE 58-1F-3E-7C-17-CE
[NEW] Device 61:8D:F0:19:75:3E 61-8D-F0-19-75-3E
[NEW] Device 68:7A:15:E7:AD:CA 68-7A-15-E7-AD-CA
[NEW] Device 78:21:08:79:5C:85 78-21-08-79-5C-85
[NEW] Device 6F:66:07:AC:13:D7 6F-66-07-AC-13-D7
[NEW] Device 68:E4:6A:8E:99:74 68-E4-6A-8E-99-74
[NEW] Device 54:AF:B7:03:4D:69 54-AF-B7-03-4D-69
[NEW] Device 74:5F:D2:47:FC:43 74-5F-D2-47-FC-43

[bluetooth]# pair 4C:02:20:3C:2A:6C
Attempting to pair with 4C:02:20:3C:2A:6C
[CHG] Device 4C:02:20:3C:2A:6C Connected: yes
Request confirmation
[agent] Confirm passkey 381184 (yes/no): yes
[CHG] Device 4C:02:20:3C:2A:6C Modalias: bluetooth:v038Fp1200d1436
[CHG] Device 4C:02:20:3C:2A:6C UUIDs: 00001105-0000-1000-8000-00805f9b34fb
[CHG] Device 4C:02:20:3C:2A:6C UUIDs: 0000110a-0000-1000-8000-00805f9b34fb
[CHG] Device 4C:02:20:3C:2A:6C UUIDs: 0000110c-0000-1000-8000-00805f9b34fb
[CHG] Device 4C:02:20:3C:2A:6C UUIDs: 00001112-0000-1000-8000-00805f9b34fb
[CHG] Device 4C:02:20:3C:2A:6C UUIDs: 00001115-0000-1000-8000-00805f9b34fb
[CHG] Device 4C:02:20:3C:2A:6C UUIDs: 00001116-0000-1000-8000-00805f9b34fb
[CHG] Device 4C:02:20:3C:2A:6C UUIDs: 0000111f-0000-1000-8000-00805f9b34fb
[CHG] Device 4C:02:20:3C:2A:6C UUIDs: 0000112f-0000-1000-8000-00805f9b34fb
[CHG] Device 4C:02:20:3C:2A:6C UUIDs: 00001132-0000-1000-8000-00805f9b34fb
[CHG] Device 4C:02:20:3C:2A:6C UUIDs: 00001200-0000-1000-8000-00805f9b34fb
[CHG] Device 4C:02:20:3C:2A:6C UUIDs: 00001800-0000-1000-8000-00805f9b34fb
[CHG] Device 4C:02:20:3C:2A:6C UUIDs: 00001801-0000-1000-8000-00805f9b34fb
[CHG] Device 4C:02:20:3C:2A:6C UUIDs: 0000fdaa-0000-1000-8000-00805f9b34fb
[CHG] Device 4C:02:20:3C:2A:6C ServicesResolved: yes
[CHG] Device 4C:02:20:3C:2A:6C Paired: yes
Pairing successful
```

4.5. Usage of Audio

- Type the command to record for 10 seconds:

```
arecord -d 10 -f cd -r 44100 -c 2 -t wav test5.wav
```

- Type the command to play audio:

```
aplay test5.wav
```

```
root@DebixModelC:~# arecord -d 10 -f cd -r 44100 -c 2 -t wav test5.wav
Recording WAVE 'test5.wav' : Signed 16 bit Little Endian, Rate 44100 Hz, Stereo
root@DebixModelC:~# ls
test5.wav
root@DebixModelC:~# aplay test5.wav
Playing WAVE 'test5.wav' : Signed 16 bit Little Endian, Rate 44100 Hz, Stereo
```

4.6. Usage of USB

1. Access the U disk in FAT32 format, the system will automatically mount it to the /mnt path.

```
df -h
```

```
root@DebixModelC:~# df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/root        29G   3.0G   24G  11% /
devtmpfs        214M   4.0K  214M   1% /dev
tmpfs           471M     0   471M   0% /dev/shm
tmpfs           189M   8.9M  180M   5% /run
tmpfs           4.0M     0   4.0M   0% /sys/fs/cgroup
tmpfs           471M   16K   471M   1% /tmp
tmpfs           471M  172K   471M   1% /var/volatile
/dev/mmcblk1p1  665M   32M  633M   5% /boot
tmpfs           95M   4.0K   95M   1% /run/user/0
/dev/sda2       29G   3.2G   25G  12% /run/media/sda2
/dev/sda1       500M   34M  467M   7% /run/media/sda1
root@DebixModelC:~#
```

- If the U disk is not mounted, you can mount the U disk with the following command:
 - Query the U disk letter.

```
fdisk -l
```

```
root@DebixModelC:~# fdisk -l
Disk /dev/mtdblock0: 8 MiB, 8388608 bytes, 16384 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes

Disk /dev/mmcblk0: 14.56 GiB, 15634268160 bytes, 30535680 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes

Disk /dev/mmcblk0boot0: 4 MiB, 4194304 bytes, 8192 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes

Disk /dev/mmcblk0boot1: 4 MiB, 4194304 bytes, 8192 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes

Disk /dev/mmcblk1: 29.72 GiB, 31914983424 bytes, 62333952 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0x076c4a2a

Device            Boot  Start      End  Sectors  Size Id Type
/dev/mmcblk1p1  *                16384  1379531  1363148  665.6M c W95 FAT32 (LBA)
/dev/mmcblk1p2                1392640  62333951  60941312  29.1G 83 Linux

Disk /dev/sda: 29.72 GiB, 31914983424 bytes, 62333952 sectors
Disk model: STORAGE DEVICE
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0xda3661a8

Device            Boot  Start      End  Sectors  Size Id Type
/dev/sda1                20480  1044479  1024000  500M c W95 FAT32 (LBA)
/dev/sda2                1228800  62333951  61105152  29.1G 83 Linux
```

- Mount the U disk.

```
mount /dev/sda1 /mnt
```

2. Enter the U disk directory.

```
cd /mnt
```

```
ls
```

```
root@DebixModelC:~# cd /mnt
root@DebixModelC:/mnt# ls
Image
'System Volume Information'
imx8mp-debix-4g-board.dtb
imx8mp-debix-core-HC050IG40029-D58V.C.dtb
imx8mp-debix-core-HC080IY28026-D60V.C.dtb
imx8mp-debix-core-HC101IK25050-D59V.C.dtb
imx8mp-debix-core-JW050R0320I01.dtb
imx8mp-debix-core-JW070R0520B02.dtb
imx8mp-debix-core-JW080R1120B02.dtb
imx8mp-debix-core-JW101HD-X00.dtb
```

3. Clear the cache: run before each read and write test command.

```
sh -c "sync && echo 3 > /proc/sys/vm/drop_caches"
```

4. Test write speed.

```
sh -c "sync && echo 3 > /proc/sys/vm/drop_caches" #Clear the cache
dd if=/dev/zero of=cc bs=400M count=1
```

```
root@DebixModelC:/mnt# sh -c "sync && echo 3 > /proc/sys/vm/drop_caches"
[ 3689.861341] sh (15374): drop_caches: 3
root@DebixModelC:/mnt# dd if=/dev/zero of=cc bs=400M count=1
1+0 records in
1+0 records out
419430400 bytes (419 MB, 400 MiB) copied, 13.4917 s, 31.1 MB/s
root@DebixModelC:/mnt# █
```

5. Test read speed.

```
sh -c "sync && echo 3 > /proc/sys/vm/drop_caches" #Clear the cache
dd if=./cc of=/dev/null bs=400M count=1
```

```
root@DebixModelC:/mnt# sh -c "sync && echo 3 > /proc/sys/vm/drop_caches"
[ 3807.466288] sh (15845): drop_caches: 3
root@DebixModelC:/mnt# dd if=./cc of=/dev/null bs=400M count=1
1+0 records in
1+0 records out
419430400 bytes (419 MB, 400 MiB) copied, 12.5997 s, 33.3 MB/s
root@DebixModelC:/mnt# █
```

4.7. Usage of Display

The three screens supported by DEBIX Model C are as follows:

Table 11 Display Screen supported by DEBIX Model C (Touchscreen with USB interface)

No.	Screen Type	Specification Address
1	DEBIX TD050A 800x480 5-inch LVDS display	https://debix.io/Uploads/Temp/file/20240724/DEBIX%20TD050A.pdf
2	DEBIX TD070A 1024x600 7-inch LVDS display	https://debix.io/Uploads/Temp/file/20240724/DEBIX%20TD070A.pdf

1. Usage of DEBIX TD070A 1024x600 7-inch LVDS screen

- 1) **Component Preparation:** LVDS screen cable, DEBIX Model C, LVDS screen, as shown in the figure below:

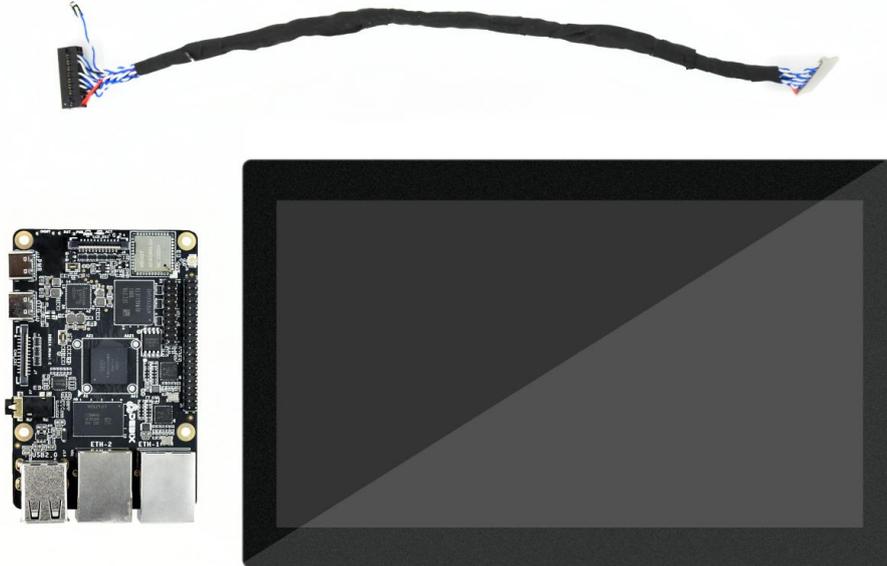


Figure 25

- 2) Plug the double-row female header of LVDS screen cable to LVDS interface (J8) of DEBIX Model C, the red line should be connected to Pin1, Pin2; as for the sole 2Pin blue and white line, the blue line is connected to Pin27 of GPIO (J1), the white line is connected to Pin25 of GPIO (J1).

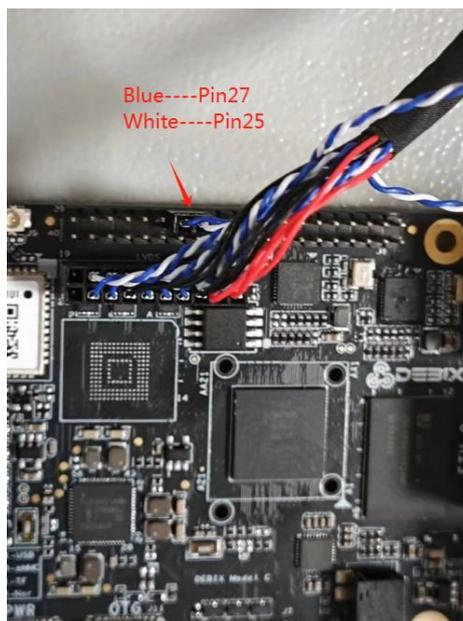


Figure 25 Connect LVDS screen cable to DEBIX Model C

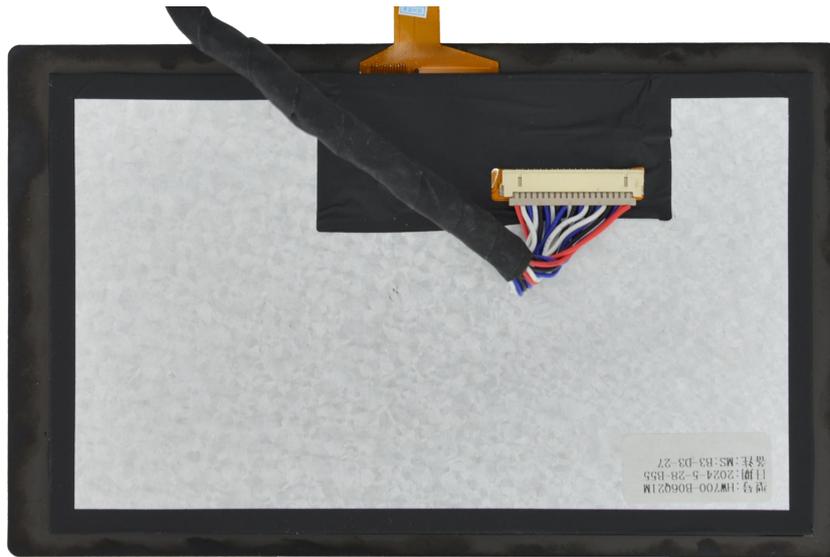


Figure 26 Connect LVDS screen cable to LVDS screen

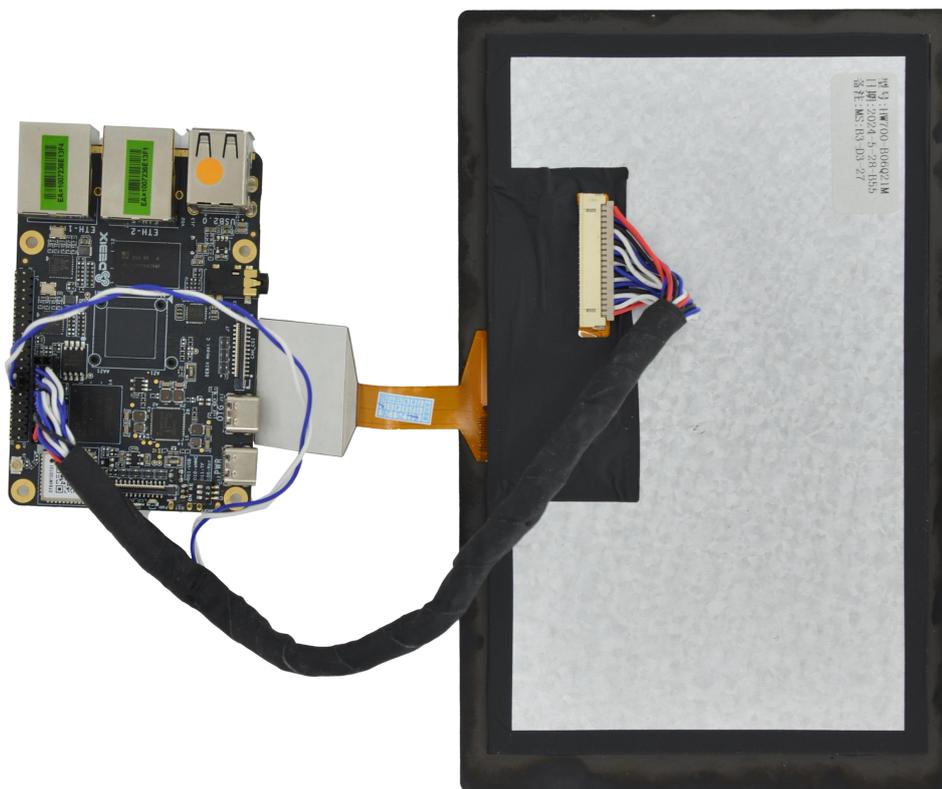


Figure 27 LVDS screen to DEBIX Model C completed

- 3) DEBIX Model C is connected to the power supply, the LVDS screen displays the following figure:

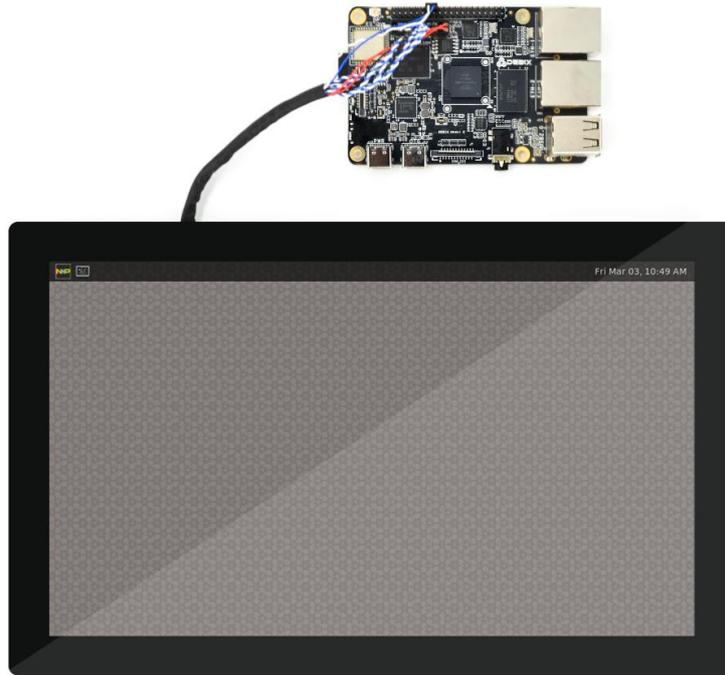


Figure 28

2. Usage of DEBIX TD050A 800x480 5-inch LVDS screen is similar to the one described above.

4.8. Usage of Camera

DEBIX Model C supports two types of camera modules: DEBIX Camera 200A Module, DEBIX Camera 500A Module.

- The **connection method** of DEBIX Model C using DEBIX Camera Module is the same as that of DEBIX Model A. Please refer to [DEBIX Camera Module User Manual](#) for detailed interface and usage information.
- **Preview image commands:**
 - DEBIX Camera 200A Module

```
gst-launch-1.0 v4l2src device=/dev/video0 ! autovideosink
```

- DEBIX Camera 500A Module

```
gst-launch-1.0 v4l2src device=/dev/video0 !  
'video/x-raw,width=1920,height=1080,framerate=(fraction)15/1' ! autovideosink #1080p  
resolution
```

```
gst-launch-1.0 v4l2src device=/dev/video0 !  
'video/x-raw,width=1280,height=720,framerate=(fraction)30/1' ! autovideosink #720p  
resolution
```

```
gst-launch-1.0 v4l2src device=/dev/video0 !  
'video/x-raw,width=640,height=480,framerate=(fraction)30/1' ! autovideosink #640x480  
resolution
```

- It is recommended to use **the USB flash drive switching method**, which is to modify the [Debix_Settings.xml](#) file (stored in USB flash drive and inserted into the device, and the device will switch automatically when powered on, after switching, the device will restart automatically, and the switch will take effect) to switch between the display and camera. As shown in the figure below, select to switch the corresponding display and camera by modifying the value of **enable** to "true" or

```

"false".

<?xml version="1.0" encoding="UTF-8"?>

<Debix_Model_C>
  <panel>
    <select type='10.1 inch' name="HC101IK25050" enable="true" />
    <select type='7 inch' name="JW070R0520B02" enable="false" />
    <select type='8 inch' name="HC080IY28026" enable="false" />
  </panel>
  <camera>
    <select type='ov5640' name="500A" enable="true" />
    <select type='gc2145' name="200A" enable="false" />
  </camera>
</Debix_Model_C>

```

4.9. ADC IN Verification

Power on the device after shorting Pin14 to Pin22 of the GPIO-40Pin using a DuPont cable:

Table 12 ADC IN channel node description

Function	Interface	Pin	Definition	Channel Node
ADC IN	J1	22	ADC_IN0	voltage0
		24	ADC_IN1	voltage1
		26	ADC_IN2	voltage2
		28	ADC_IN3	voltage3

- Query Analogue Conversion Factors via the command:

```
cat /sys/bus/platform/drivers/imx93-adc/44530000.adc/iio:device0/in_voltage_scale
```

- Get the ADC 1 channel voltage via the command:

```
cat /sys/bus/platform/drivers/imx93-adc/44530000.adc/iio:device0/in_voltage0_raw
```

- Query ADC channel 1 again ($4.095 \times 0.439453125 = 1.8V$ in the figure), to get an input of 1.8V.

```
cat /sys/bus/platform/drivers/imx93-adc/44530000.adc/iio:device0/in_voltage0_raw
```

```
root@DebixModelC:~# cat /sys/bus/platform/drivers/imx93-adc/44530000.adc/iio:dev
ice0/in_voltage_scale
0.439453125
root@DebixModelC:~# cat /sys/bus/platform/drivers/imx93-adc/44530000.adc/iio:dev
ice0/in_voltage0_raw
4095
```

4.10. LED & Key

1. LED

- The green indicator is the system LED, the device is running normally, the indicator blinks; otherwise the indicator is off.
- The red and blue indicator is power LED, after power on, the indicator light; after power off, the indicator light turn red until red light is off.

2. Key

- ON/OFF Key
 - Short press: ACT green light is off, the system enters the sleep state.
Short press again: ACT green light blinks to wake up the system.
 - Long press: Long press, red and blue lights turn red until red light is off, and shut down the device.
Long press again, until the red and blue light is on, and the device power on.
- RESET Key
 - Press to reset the system, and green light blinks.

4.11. Usage of GPIO

DEBIX Model C OS has built-in GPIO interface operation command, you can set GPIO by GPIO command.

IMPORTANT

The GPIO voltage input of DEBIX Mode A/B only supports 3.3V. If the input is higher than 3.3V, it may cause damage to the GPIO interface and CPU.

1. In the terminal window, type command `debix-gpio` to print out the use of GPIO as follows:

```
root@DebixModelC:~# debix-gpio
Debix gpio contrl
Usage
  debix-gpio <gpioName> <mode> [value]/[edge]
  gpioName: input gpioName
  mode     : in/out mode
  value    : out mode 0=low 1=high
  edge     : in mode 0=none 1=rising 2=falling 3=both
  eg. debix-gpio GPIO1_IO12 out 1
  eg. debix-gpio GPIO1_IO12 in 3
  debix-gpio <showGpioName>
  showGpioName: list gpio names
root@DebixModelC:~#
```

- Command Format: `debix-gpio <gpioName> <mode> [value]/[edge]`
 - `gpioName`: GPIO interface name, for example: `GPIO1_IO12`
 - `mode`: GPIO mode, respectively out (output) and in (input)
 - `value`: When mode is out (output), the value attribute takes effect; the value can be 0 or 1, 0 means output low level, 1 means output high level
 - `Edge`: When mode is in (input), the edge attribute takes effect; there are 4 GPIO states: 0-none, 1-rising, 2-falling, 3-both
2. Type command `debix-gpio showGpioName` to print out the definition of the GPIO interface and the location on the board, as follows:


```
root@DebixModelC:~# debix-gpio GPIO2_IO12 in 1
Debix gpio contrl
INFO:GPIO_NAME = GPIO2_IO12(44)
INFO:GPIO_MODE = in
INFO:GPIO_FLAG = RISING
INFO:pin:44 value=0

INFO:pin:44 value=1
INFO:pin:44 value=0
INFO:pin:44 value=1
INFO:pin:44 value=0
INFO:pin:44 value=1
```

4.12. Heat Dissipation

When a DEBIX Model C runs for a prolonged period of time, it will result in an increase in its CPU temperature. Therefore, implementations should be considered to cool the CPU and the entire device passively. If the CPU needs to be cooled, it is recommended to use CPU aluminum alloy heatsink: paste aluminum alloy heatsink directly above the CPU for heat dissipation, as shown below:

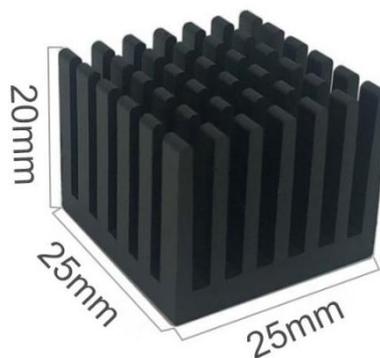


Figure 29 Aluminum alloy heatsink