DEBIX Model C & Model D User Guide

Version: V1.11 (2026-05)

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

DEBIX Model C and Model D are low-power single board computers designed for intelligent edge computing and embedded applications. Built on the same hardware platform with pin-compatible NXP i.MX 9 series processors, the system delivers an optimized balance of performance, power efficiency, and scalability for evolving embedded designs.

To address different application requirements, DEBIX provides two processor options while maintaining the same carrier board architecture and extensible interfaces. The platform supports flexible integration for IoT edge, contactless HMI, smart home, building control, and industrial applications.

Applicable Models:

- DEBIX Model C (NXP i.MX9352)
- DEBIX Model D (NXP i.MX9131)

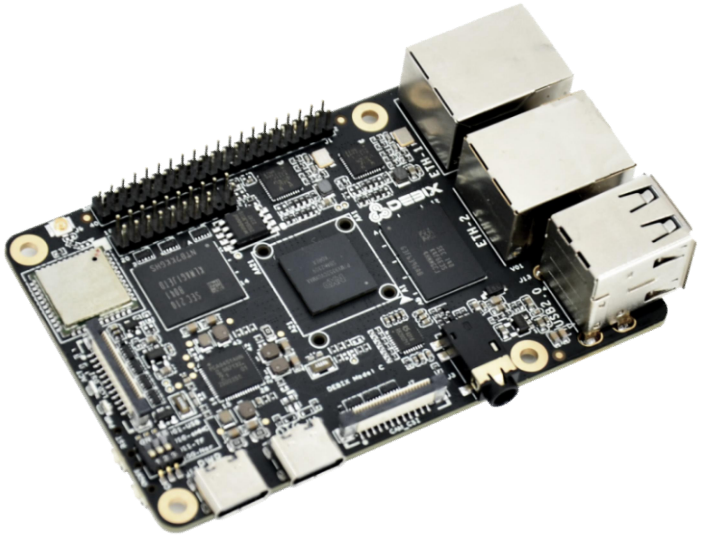


Figure 1 DEBIX Model C/D

REVISION HISTORY		
Rev.	Date	Description
1.0	2023.08.29	First edition
1.1	2024.04.19	Added Section 4.7.Usage of Display and 4.8.Usage of Camera
1.2	2024.08.16	Updated Section 4.7.Usage of Display to add display model information and pin connection details
1.3	2025.01.14	1. Updated the supported OS version in Table 3 2. Updated the technical support contact information
1.4	2025.01.23	Modified the supply voltage options of LVDS
1.5	2025.03.07	Updated Section 3.1.2.System Boot and related content
1.6	2025.05.20	1. Modified the OS support information in the Table 3 2. Modified Section 3.1.Software Installation 3. Updated the display product brief links in Section 4.7.Usage of Display
1.7	2025.05.26	Added the NOR Flash reprogramming method in Section 3.1.2.System Boot.
1.8	2025.07.23	1. Removed the Compliance Statement section. For detailed product compliance information, please refer to the product page 2. Added Section 4.10. Usage of Debug
1.9	2025.09.04	Modified Section 4.7.Usage of Display to add the usage method for the USB touch of the LVDS display
1.10	2026.01.13	Modified Section 4.7.Usage of Display to add instructions for switching the display on the device
1.11	2026.05.20	Updated the user manual to support DEBIX Model C and Model D, and added model-specific differences and processor information.

INDEX

Chapter 1 Security

1.1. Safety Precaution

This document describes the peripheral connection method for the device. In most cases, the device will operate by simply connecting the peripheral cables.

Table 1 Terms and conventions

Symbol	Meaning
Warning! 	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.
Caution! 	Always ground yourself to remove any static electric charge before touching ***DEBIX*** 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.

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 assumes no liability for the accuracy of any statement of this instructional document.

1.3. 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 Introduction

DEBIX Model C and Model D are low-power single board computers designed for embedded and intelligent edge computing applications. Both models are built on the same hardware platform and carrier board architecture while utilizing pin-compatible NXP i.MX 9 series processors to meet different performance and application requirements.

DEBIX Model C is powered by the NXP i.MX9352 processor, providing enhanced processing capability and Arm Ethos™-U65 microNPU support for machine learning applications. DEBIX Model D is based on the NXP i.MX9131 processor and is optimized for low-power embedded applications with efficient computing performance.

The two models share the same hardware interfaces, mechanical design, and peripheral expandability, enabling a consistent development experience across different application scenarios. Unless otherwise specified, the hardware interfaces and operating procedures described in this manual apply to both models.

Main features:

- Based on pin-compatible NXP i.MX 9 series processors, delivering an optimized balance of performance, power efficiency, and scalability for embedded applications
- Low-power architecture designed for energy-efficient intelligent edge computing and industrial embedded systems
- Rich peripheral connectivity, including dual Gigabit Ethernet and multiple USB interfaces for flexible system integration
- Multiple extensible interfaces for IoT edge, contactless HMI, smart home, building control, and industrial applications
- Provides compatibility with DEBIX expansion accessories

2.1. Overview

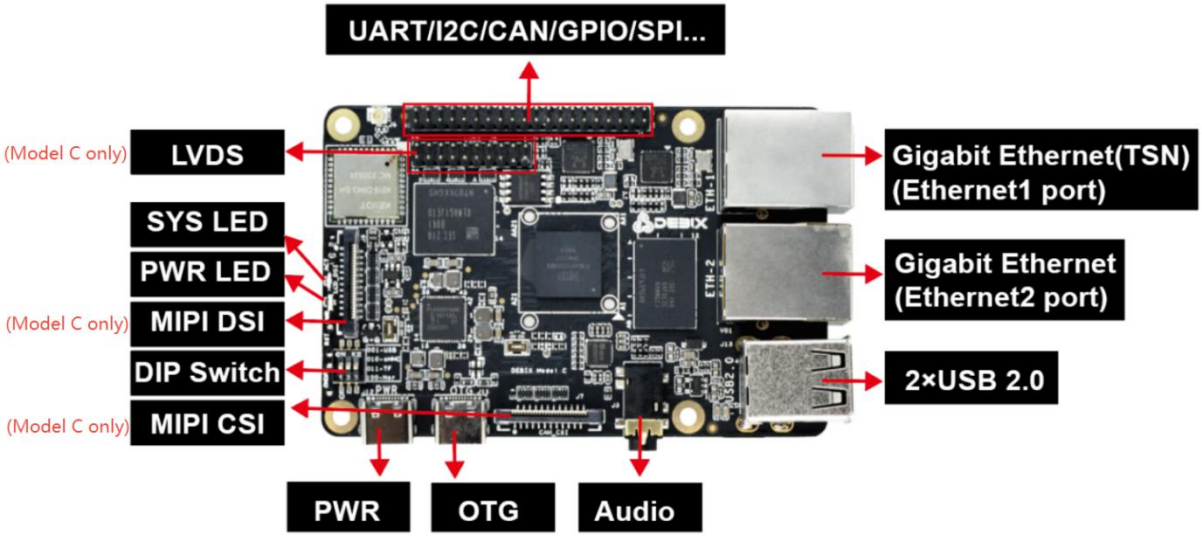


Figure 2

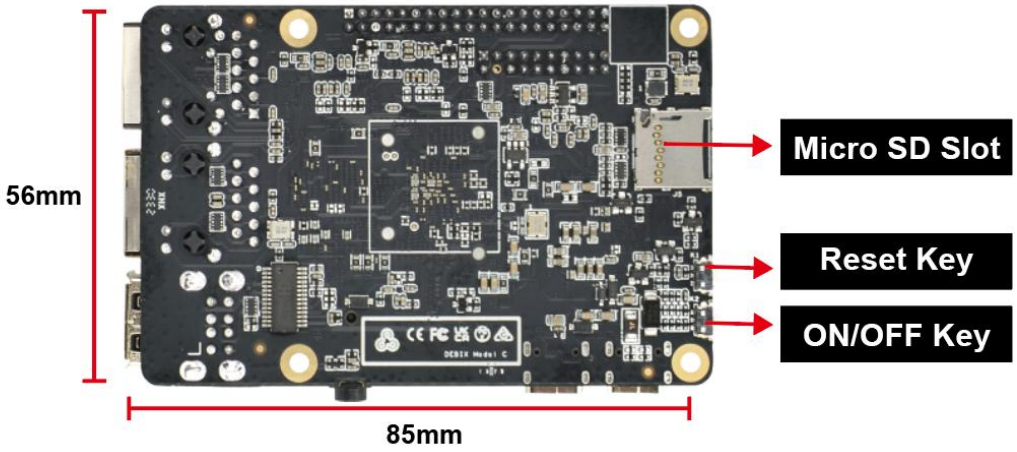


Figure 3

DEBIX Model C and Model D share the same hardware platform while providing different processor configurations to meet various application requirements. The detailed specifications of each model are listed below.

Table 2 Product Specifications of DEBIX Model C and DEBIX Model D

Feature	DEBIX Model C	DEBIX Model D
System		
CPU	NXP i.MX9352 2 x Arm® Cortex®-A55 @1.7 GHz The maximum power consumption is 1W.	NXP i.MX9131 1 x Arm® Cortex®-A55 @1.4 GHz The maximum power consumption is 1.24W.
MCU	1 x Arm Cortex-M33 @250 MHz	None
GPU	2D Graphics Accelerator (PXP, Pixel Pipeline)	None
NPU	Arm Ethos™-U65 microNPU, up to 0.5 TOPS	None
Security	Arm® TrustZone®, TrustZone Resource Domain Controller (TRDC), EdgeLock® Secure Enclave, Secure RTC	Arm® TrustZone®, TrustZone Resource Domain Controller (TRDC), EdgeLock® Secure Enclave, Secure RTC and Secure Storage
Watchdog	Hardware Watchdog	
Memory	1GB LPDDR4X (2GB optional)	
Storage	Default: Onboard 8MB NOR Flash + MicroSD Card (user-provided) Optional: Onboard 8-256GB eMMC	
OS	Ubuntu 22.04 Server, Yocto-L6.1.36, Debian 12 Server (also supports OpenWRT and FreeRTOS)	Yocto, Zephyr
Boot Mode	• NOR Flash (Default) • MicroSD card • eMMC	
Communication		
Gigabit	2 x 10/100/1000M Ethernet interfaces	

Network	● 1 x Gigabit Ethernet port, support TSN and POE power supply (POE power device module required) ● 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	None
MIPI DSI	1 x 1080p60 MIPI DSI, support 4-lane, 24Pin 0.5mm Pitch FPC socket	None
MIPI CSI	1 x 1080p60 MIPI CSI, support 2-lane, 24Pin 0.5mm Pitch FPC socket	None
Audio	1 x 3.5mm headphone and microphone combo port	
External I/O Interface		
USB	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 Headers	● 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	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-bit)	

Slot	1 x MicroSD 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)
Net Weight	43g (±0.5g)
Operating Temp.	• Industrial grade: -20°C~70°C • Industrial grade: -40°C~85°C (Wide temperature optional)

2.2. Interface

2.3.1. Power Interface

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

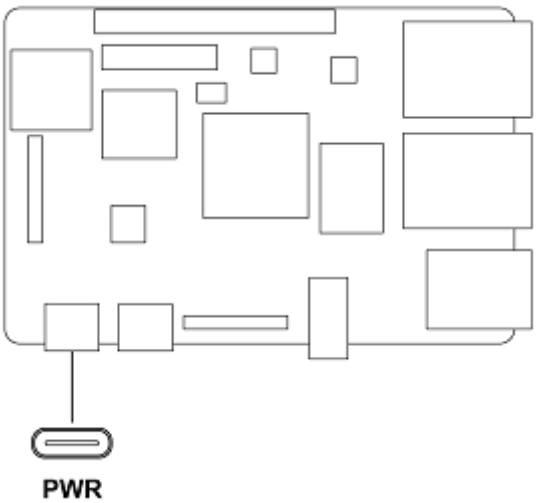


Figure 4 Power Interface

2.3.2. USB Interface

The product provides four USB interfaces, supporting USB 2.0.

- 2 x USB 2.0 Host with dual-stacked Type-A interface (J13).
- 2 x USB 2.0 with Type-C interface: one is used for 5V DC power input, while the other serves as an OTG interface (J11). The OTG interface supports system flashing and upgrading, and can also connect external devices such as USB drives and portable hard disks.

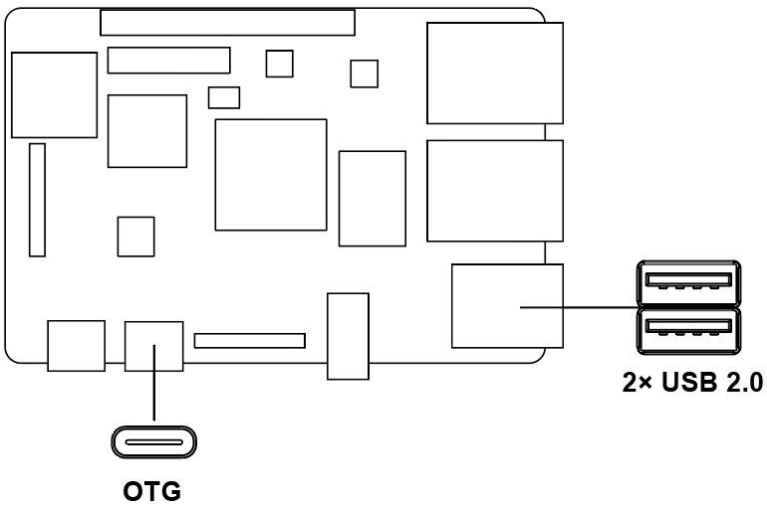


Figure 5 OTG and USB 2.0 Host

2.3.3. Ethernet Interface

DEBIX Model C/D is equipped with two Gigabit Ethernet controllers that can operate simultaneously. It provides two independent MAC Gigabit Ethernet ports:

- Gigabit Ethernet port1 (J4, ETH1): supports POE power supply (POE power supply module required). ENET_QOS (Ethernet Quality of Service), based on Synopsys proprietary, supports time-sensitive networking (TSN), EEE, Ethernet AVB (IEEE802.1Qav), IEEE1588.
- Gigabit Ethernet port2 (J5, ETH2): ENET1, 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.

Use an RJ45 Ethernet cable to connect DEBIX to the network. A set of status indicators

below each Ethernet port displays the network status: the green indicator shows the Link status (connection status), while the yellow indicator shows the Active status (data transmission activity).

Figure 6 Ethernet Interface

Table 3 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

This section applies to DEBIX Model C only. Display and camera interfaces are not supported on DEBIX Model D.

2.3.4.1. LVDS Interface

[Applies to DEBIX Model C only]

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.

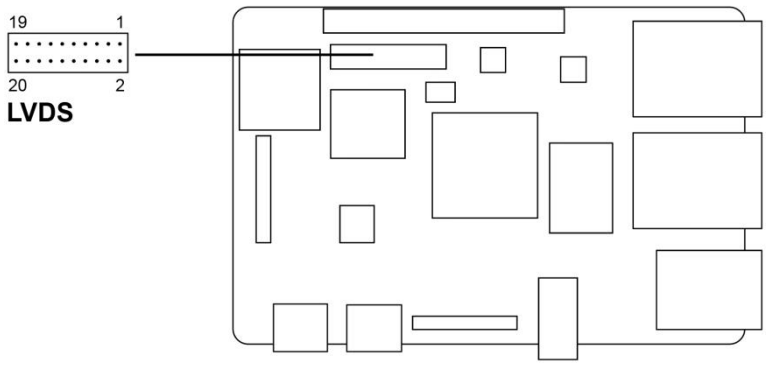


Figure 7 LVDS Interface

The pin sequence is as shown in the figure:

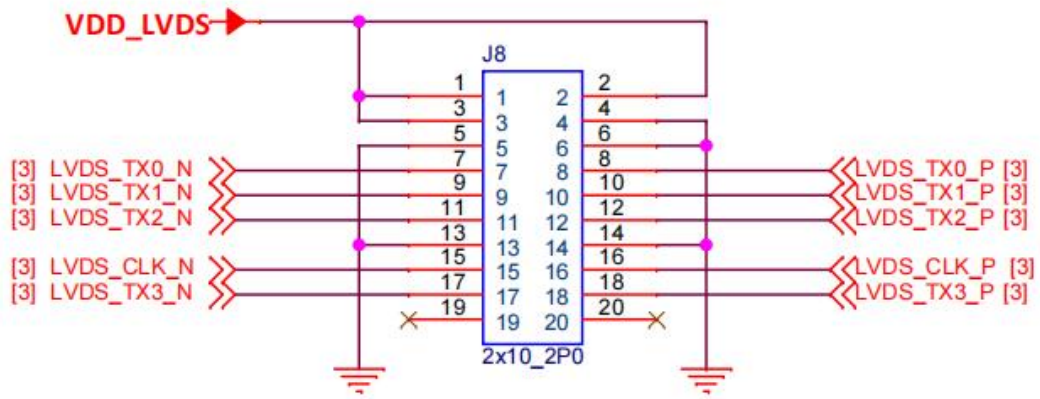


Figure 8 Pin sequence of LVDS

The interface is defined as follows:

Table 4 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

[Applies to DEBIX Model C only]

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

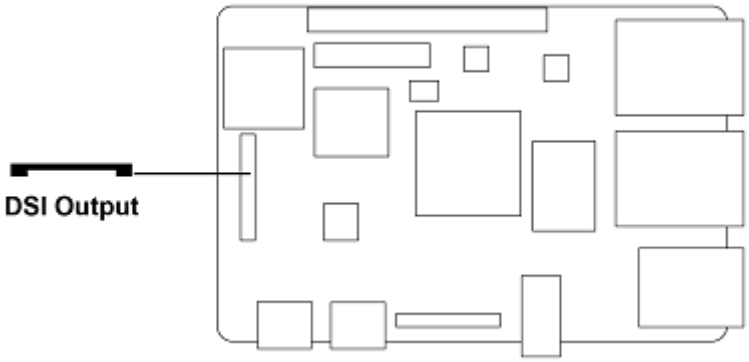


Figure 9

The pin sequence is as shown in the figure:

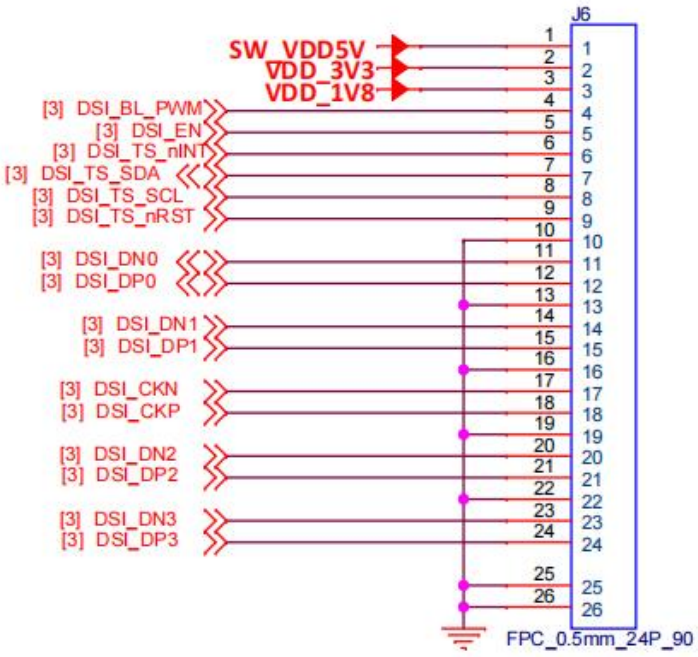


Figure 10 Pin sequence of MIPI DSI

The interface is defined as follows:

Table 5 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

[Applies to DEBIX Model C only]

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.

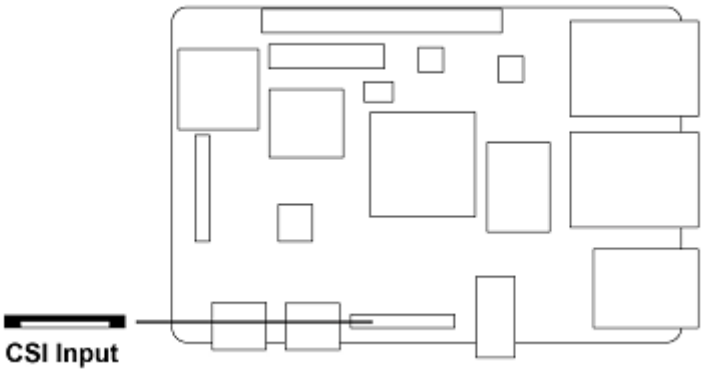


Figure 11 MIPI CSI

The pin sequence is as shown in the figure:

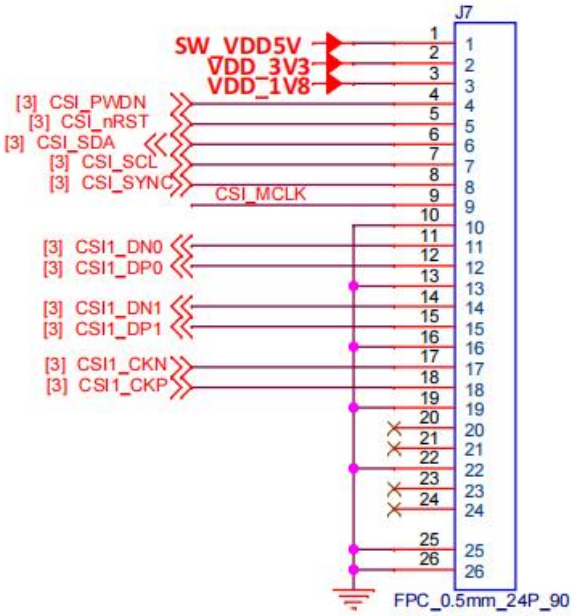


Figure 12 Pin sequence of MIPI CSI

The interface is defined as follows:

Table 6 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

The product 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.

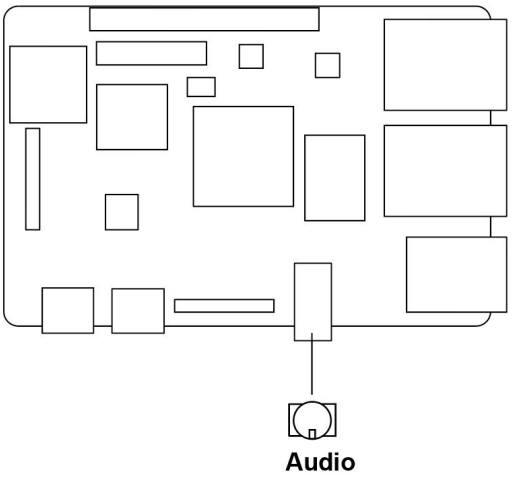


Figure 13 Audio

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 14 Definition of four-segment headphones

2.3.7. GPIO

The product 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.

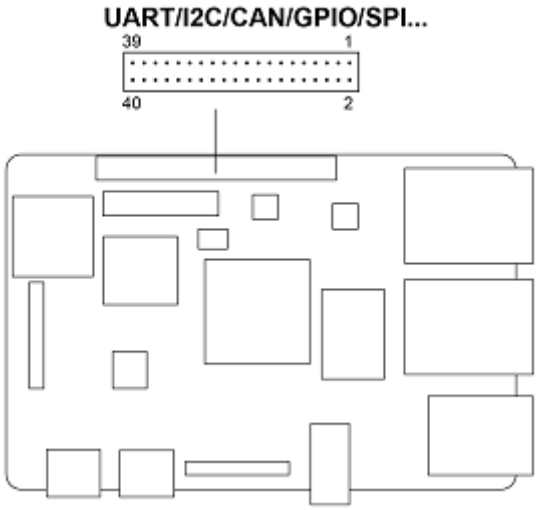


Figure 15 40Pin

The pin sequence is as shown in the figure:

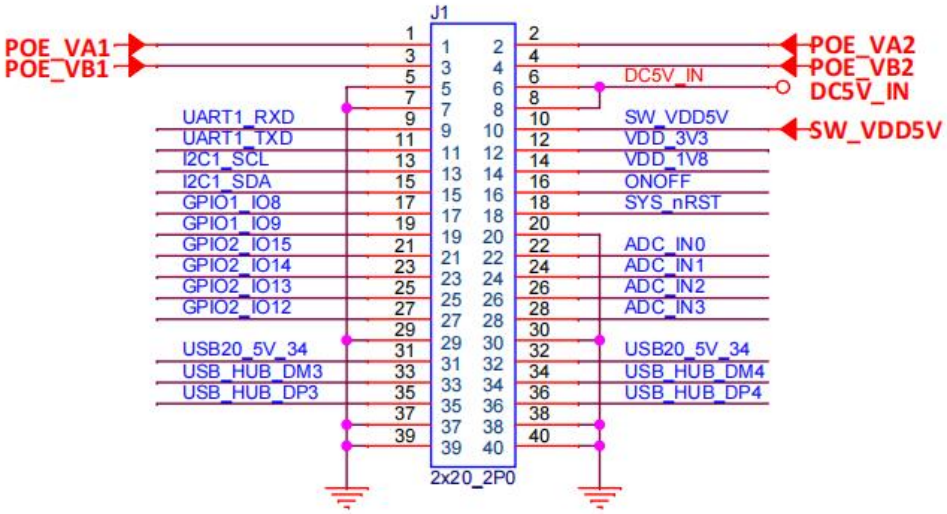


Figure 16 Pin sequence of J1

The interface is defined as follows:

Table 7 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	Device Node: /dev/ttyLP0	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 pin multiplexing functions, please refer to "GPIO Pin Multiplexing Function List" on the product page.

2.3.8. LED & Key

The product 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 8 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 3-bit DIP switch for boot mode selection. Each switch bit has two states: ON and OFF. Four boot modes are shown as follows:

- 100-SPI NOR Flash Boot (Default)
- 001-USB Flashing Mode
- 010-eMMC Boot
- 011-MicroSD Card Boot

IMPORTANT

For Yocto and Ubuntu systems, the mainboard must be configured to SPI NOR Flash boot mode (set the DIP switch to "100"). After U-Boot in the NOR Flash starts, the system can be loaded from either the MicroSD card or eMMC as required (see Section 4.1 for switching methods). If the DIP switch is incorrectly set to other modes such as "011" or "010", the mainboard will fail to boot.

This restriction does not apply to other systems (such as Debian). For these systems, the DIP switch can be configured according to the actual boot mode.

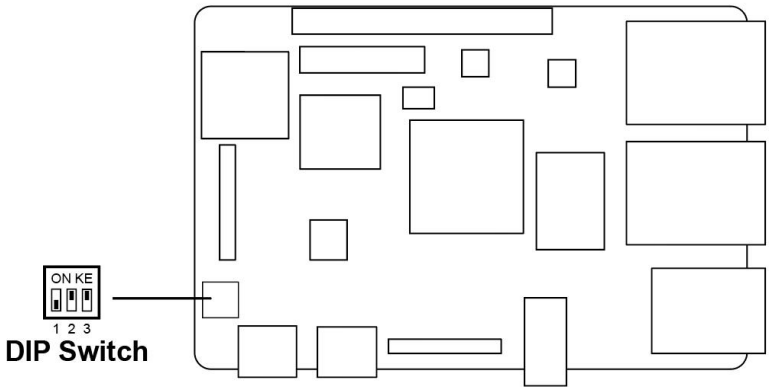


Figure 17 DIP Switch

The selected boot modes are shown in the table below:

Table 9 Boot modes configured via DIP switch

Mode / Switch	NOR Flash (Default)	USB	eMMC	MicroSD
SW state setting				

Note: When the switch points upward, it indicates ON; when pointing downward, it indicates OFF.

2.3.10. Slot

The product provides a MicroSD card slot (J2).

- When the device boots from the MicroSD card (default boot mode), the installed MicroSD card can be used as the system boot device. Insert a MicroSD card with a system image into the slot, and then power on DEBIX to boot the system from the MicroSD card.
- When the device boots from other storage media, the MicroSD card can be used as a standard storage device for saving user data after the system is powered on.

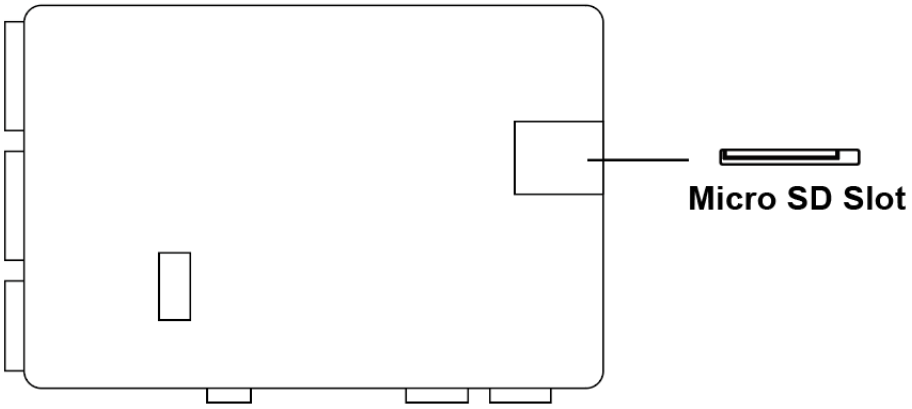


Figure 18 MicroSD Card Slot

Chapter 3 Getting 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.

NOTE

> The image to be installed depends on the selected boot mode and whether the board is equipped with eMMC storage. For example, if you need to boot from eMMC and the board includes an eMMC module, please select an image labeled “eMMC Flashing”.

2. If the downloaded image file is in .zip format, please extract it first to obtain the .img file.
3. Use the balenaEtcher tool to flash the .img file to the MicroSD card.

3.1.2. System Boot

The mainboard defaults to NOR Flash boot. To enable NOR Flash boot, set the DIP switch to ”100” (See the figure 19 below). For detailed DIP switch setting, please refer to 2.3.9.DIP Switch. 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).

IMPORTANT

- For Yocto and Ubuntu operating system, the SD card boot image and eMMC boot image do not contain Uboot files by default. Therefore, if the DIP switch is set to “011” or “010”, the mainboard cannot be started.
- For other system images (eg. Debian), set the DIP switch according to the corresponding boot mode (non-“100” NOR Flash mode).

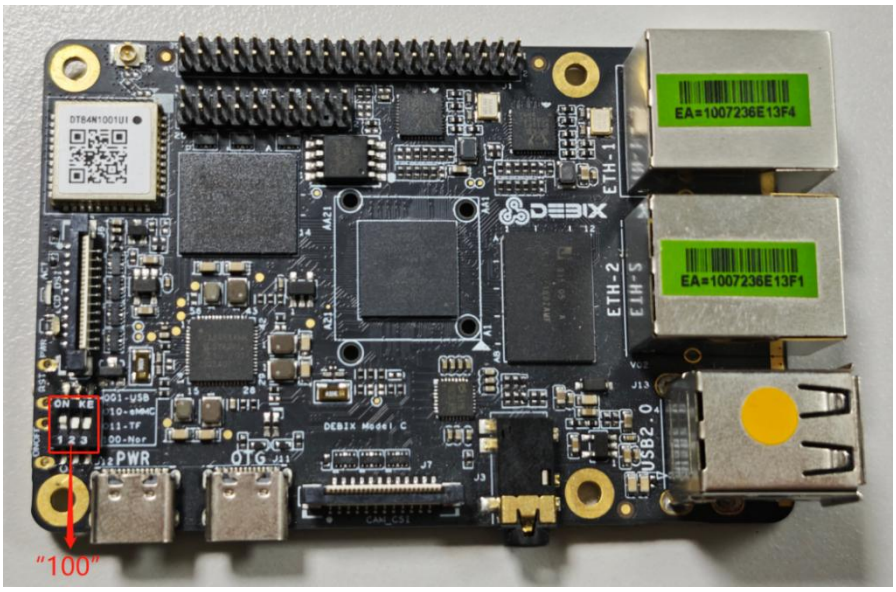
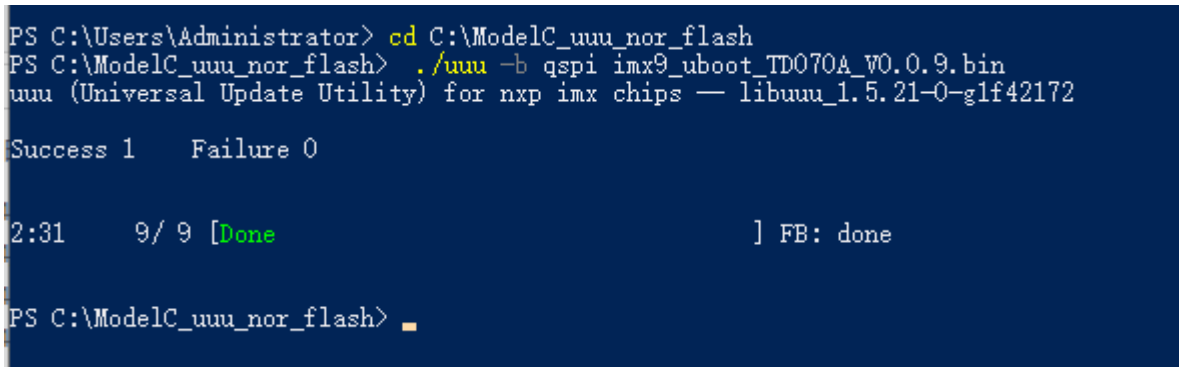


Figure 19 NOR Flash Boot Mode

NOTE

If the Nor Flash is damaged and needs to be reprogrammed, please download the firmware file from: https://we.tl/t-R4YunuTJg9. Then follow the steps:

1. Set the mainboard DIP switch to ”001” (USB Flash mode);
2. Connect the PC to the mainboard's OTG port using a Type-C data cable;
3. Run the following command: **./uuu -b qspi imx9_uboot_TD070A_V0.0.9.bin**
4. Power on the mainboard to start the programming process (See the figure below).



3.1.2.1. Flashing MicroSD Image

- **Component Preparation**

✓ DEBIX Model C/D

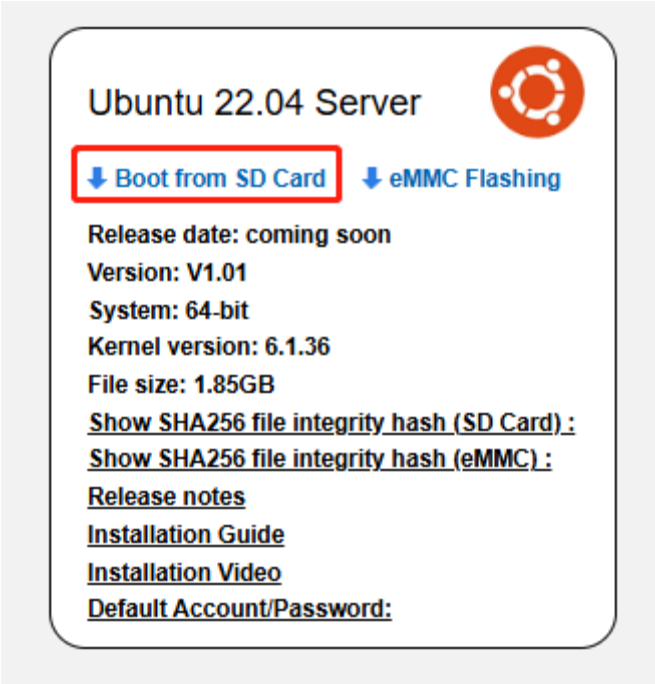
✓ MicroSD card, and card reader

✓ DC 5V/2A power adapter

✓ PC (windows 10/11)

- **Installing the Boot from MicroSD Card Image**

Click **[Boot from SD Card]** to download the MicroSD card image for DEBIX Model C/D from software download page on your PC.



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

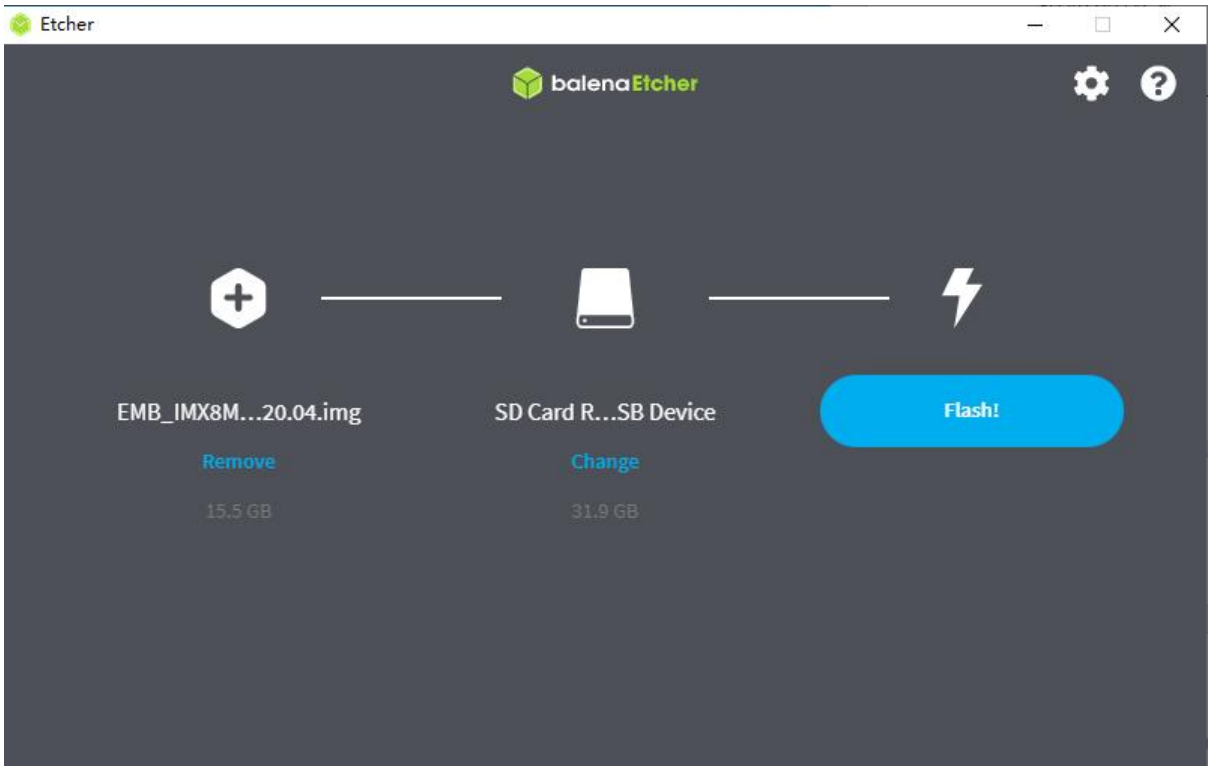


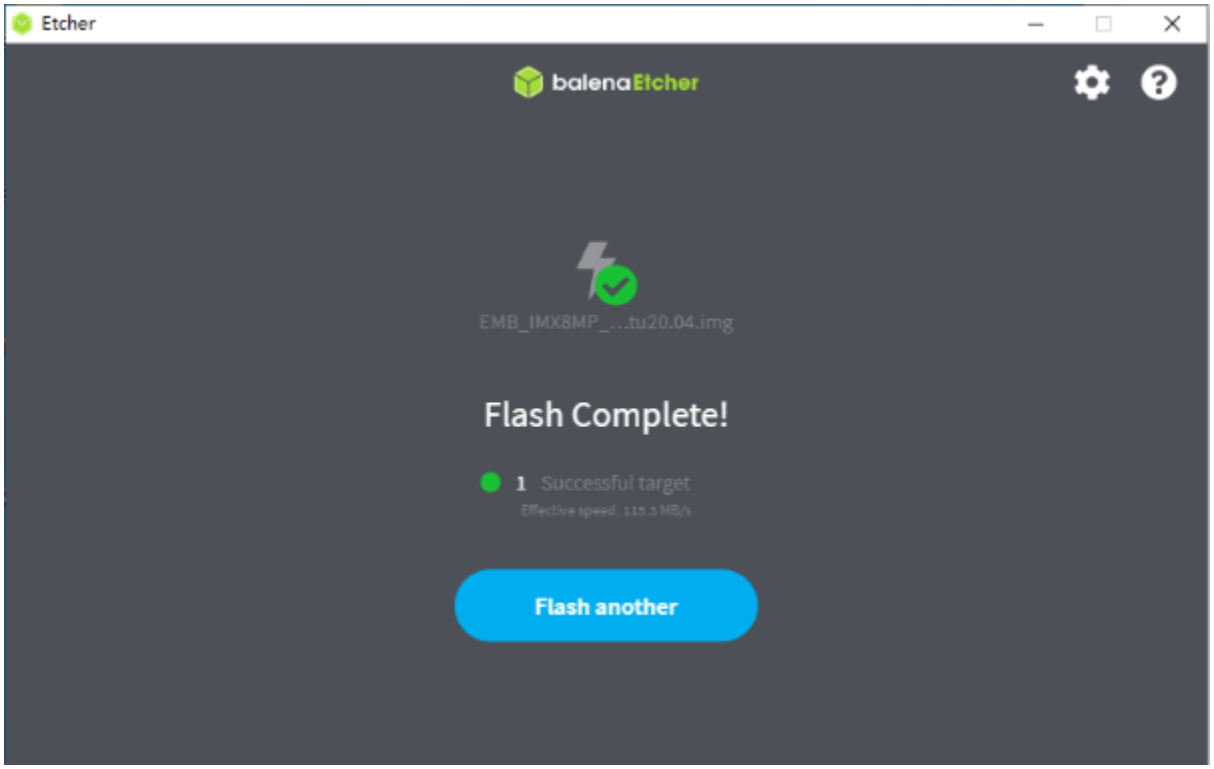
Figure 20

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

NOTE

> The system may prompt that the disk needs to be formatted. Please ignore this message, as this is expected behavior and not an error.

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



3.1.2.2. Flashing eMMC Image via MicroSD Card

- **Component Preparation**

✓ DEBIX Model C/D

✓ MicroSD card, and card reader

✓ DC 5V/2A power adapter

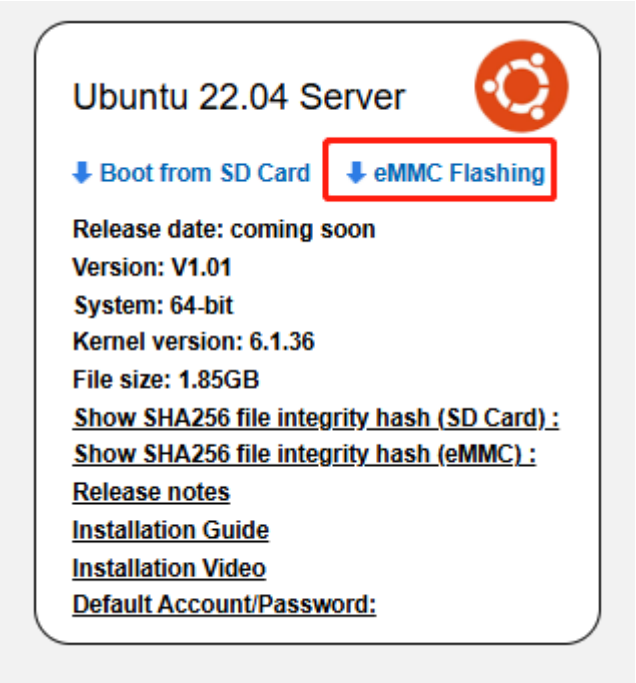
✓ PC (windows 10/11)

- **Installing the Boot from eMMC Image on the MicroSD Card**

IMPORTANT

To flash an eMMC image, an eMMC module must be selected when purchasing the product.

Click **[eMMC Flashing]** to download the eMMC image from software download page.



1. Write the downloaded system image to the MicroSD card according to the steps 1-3 operation of "Installing MicroSD Card Image".

2. Insert the MicroSD card into the slot and power on. After booting, the system will be automatically written to eMMC through the MicroSD card. When burning, the green LED on the mainboard will flash quickly, please wait. When the green LED changes from fast flash to slow flash, that is, the programming is complete.

3.1.2.3. Flashing eMMC Image via USB

- **Component Preparation**

✓ DEBIX Model C/D

✓ USB Type-C data cable

✓ DC 5V/2A power adapter

✓ PC (windows 10/11)

- **Flashing eMMC Image via USB**

IMPORTANT

- To flash an eMMC image, an eMMC module must be selected when purchasing the product.
- The following steps are applicable to Debian image flashing only.

1. Download the system installation package and UUU tool we provided to DEBIX, check the MD5 match after downloading, and then unzip it to PC.

2. Use USB cable to connect the OTG port of DEBIX to the USB port of PC, set the DIP switch to "001" USB flashing mode, connect the power supply, the system will enter the USB Flashing mode.

3. Run **Windows PowerShell** as administrator.

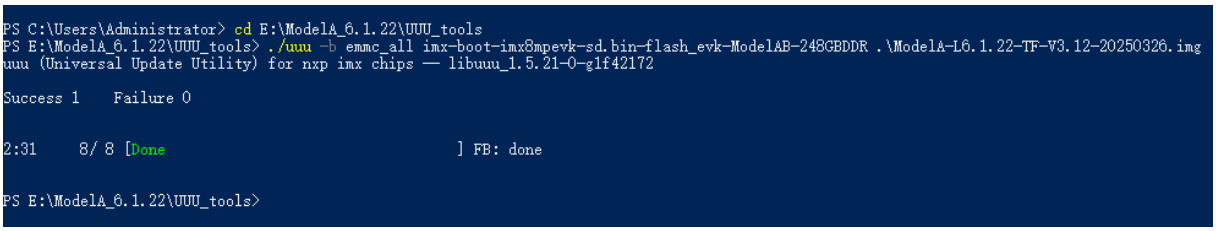
4. Type cd command to enter the root directory of the system installation package, for example:

```
cd E:\ModelC\UUU_tools
```

5. Run the following command to download the file and flash the system image to the eMMC.

```
./uuu -b emmc_all imx9_uboot.bin  iMX93-Debian12-Model-C_V1.1_20241203.wic
```

6. Wait for the system flashing to finish. When the terminal shows green "Done", it means the process is finished.



7. After flashing, disconnect the power supply and OTG USB cable, make sure the DEBIX is completely powered off, and then connect the power supply to start.

3.2. Hardware Installation

Make the hardware connections as shown in the diagram. The connection steps are as follows:

1. **Insert the MicroSD card with the system image installed:** Insert the card into the slot on the back of the mainboard. To remove it, power off the device first, then gently pull out the card.
2. **Connect the LVDS display.**
3. **Connect the keyboard.**
4. **Connect the mouse.**
5. **Connect the Ethernet cable.**
6. **Connect the power adapter:** Plug in the power supply. The mainboard power indicators (red and blue) will turn on, and the system status indicator (green) will blink.

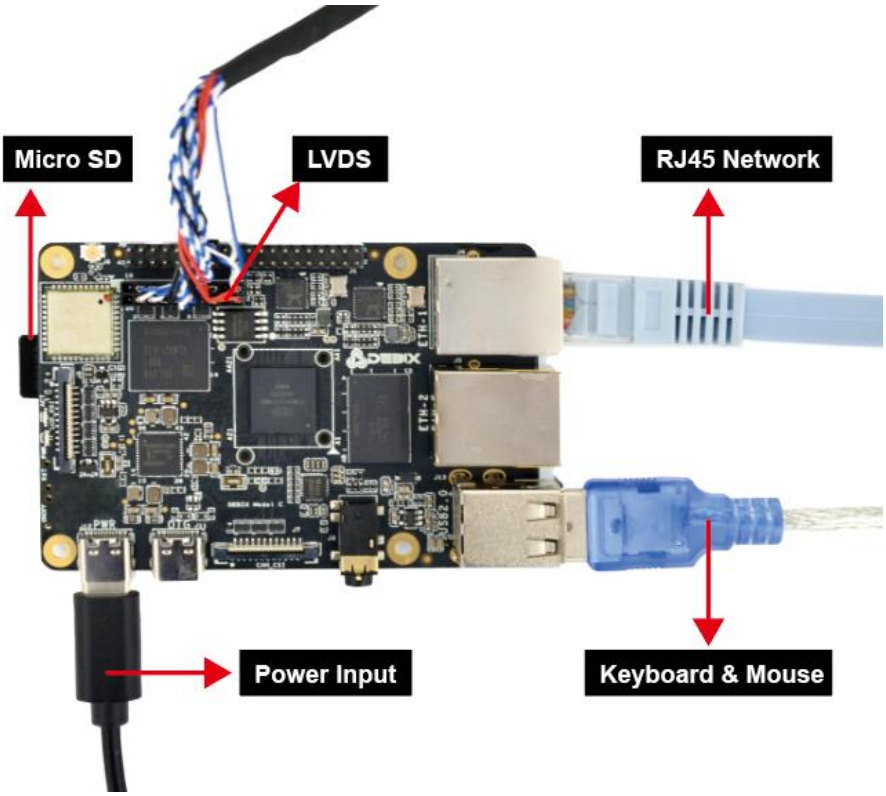


Figure 21 Hardware connection

Chapter 4 Software Application Examples

4.1. Switch Boot Mode

NOTE

> 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/D has eMMC and MicroSD card, and both contain systems, you can switch the boot mode in the following way:

1. **Select the Boot Mode Through the Serial Port**

After entering the serial console interface, when the `select:` prompt appears, use the arrow keys to select an option and press **Enter** to confirm. The following four options are available:

- `select: SD boot` — Boot from MicroSD card
- `select: eMMC boot` — Boot from eMMC
- `select: Reboot` — Reboot the device
- `select: About` — View related information

2. **Select the Boot Mode Ussing the `debix_boot` command After System Startup**

After entering the system, you can run the `debix_boot` command, then enter the corresponding number as shown in the figure below to select the boot mode:

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

4.2. Usage of Ethernet

- Ethernet port 1 (ENET_QOS): (Refdes.: J4. Device Node: eth0. 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<g
lobal>
        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
```

- Ethernet port 2 (ENET1): (Refdes.: J5. Device Node: eth1. 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<g
lobal>
        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/D: wlan0.

1. Disconnect the Ethernet cable, then use the following commands to connect the device to the WiFi network (polyhex_mi):

```
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=bohai2021 ]
  WPS = [ Type=wpspin, Requirement=alternate ]
Passphrase? bohai2021
connmanctl> [ 5892.058682] IPv6: ADDRCONF(NETDEV_CHANGE): wlan0: link becomes re
ady
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/D: 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. Enable Bluetooth and pair with the Bluetooth device.

```
hciconfig hci0 up
bluetoothctl
power on
agent on
default-agent
scan on
pair yourDeviceMAC          #Pair with the target device using its Bluetooth MAC address.
```

```
# 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 UUIDs: 98b97136-36a2-11ea-8467-484d7e99a198
[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

- Run the command to record for 10 seconds:

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

- Run 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 using the following command:
  - Check the U disk device node.

```
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

[Applies to DEBIX Model C only.]

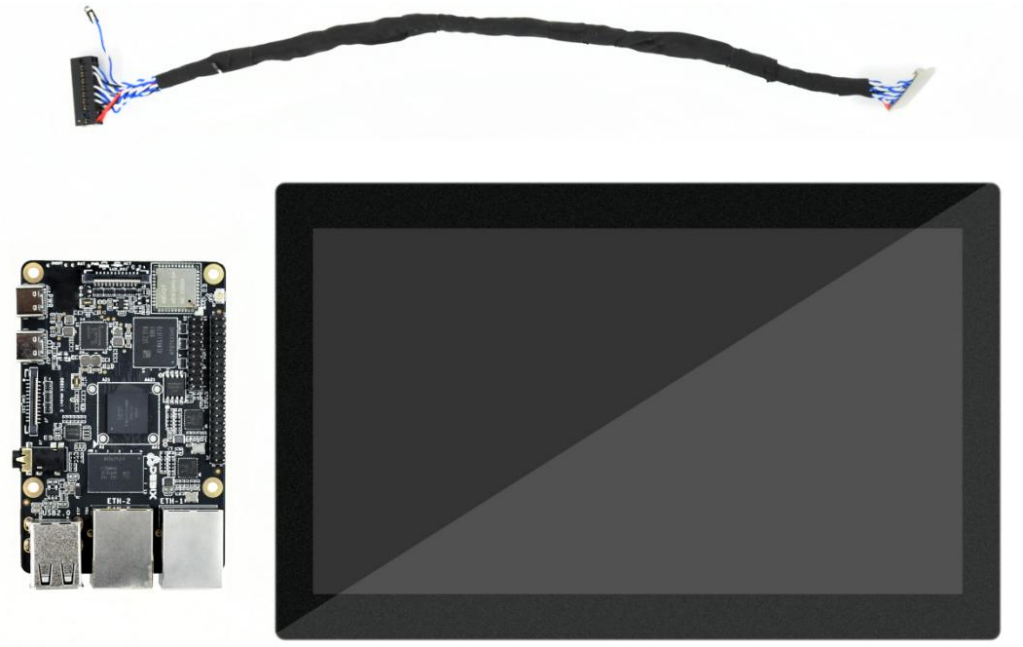
The four screens supported by DEBIX Model C are as follows:

Table 10 Display Screen supported by DEBIX Model C

No.	Screen Type	Product brief Link
1	DEBIX TD050A 800x480 5-inch LVDS screen	https://debix.io/hardware/5inch-tft-lcd-monitor.html
2	DEBIX TD070A 1024x600 7-inch LVDS screen	https://debix.io/hardware/7inch-tft-lcd-monitor.html
3	DEBIX TD101A 1280x800 10.1-inch LVDS screen	https://debix.io/hardware/10inch-tft-lcd-monitor.html
4	DEBIX TD080B 8-inch MIPI DSI display	For further information, please contact DEBIX representatives (Email Address: info@debix.io).

The following steps use the DEBIX TD070A LVDS screen as an example.

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



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).

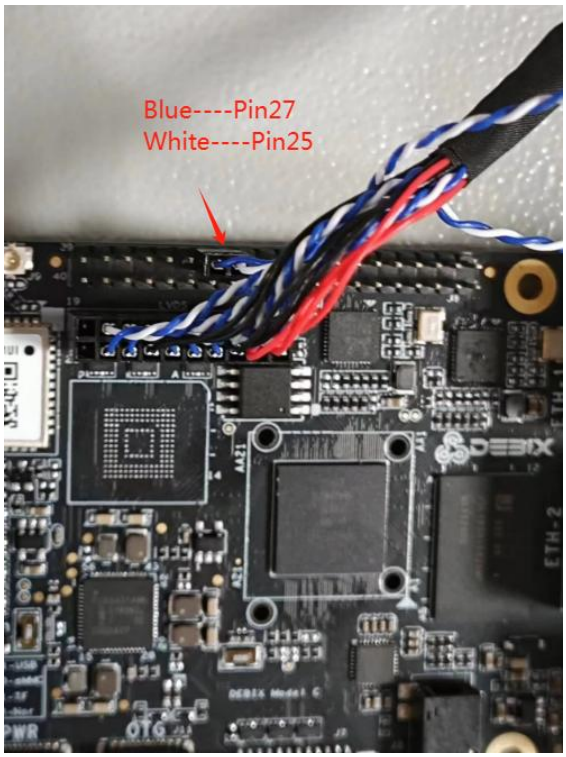


Figure 25 Connect LVDS screen cable to DEBIX Model C

Figure 26 Connect LVDS screen cable to LVDS screen

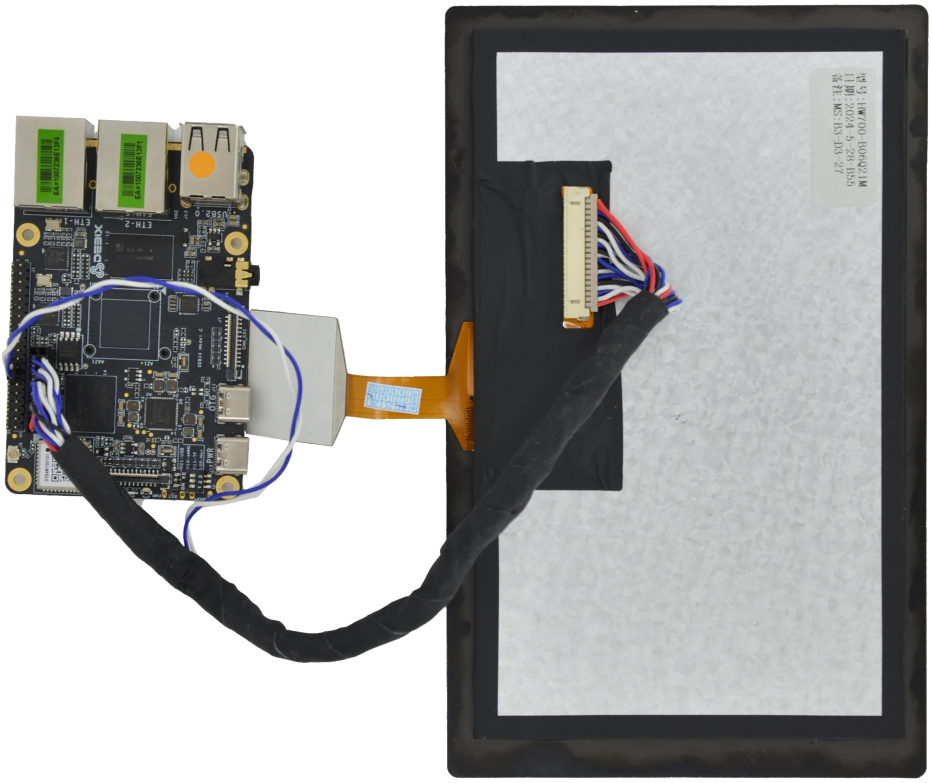
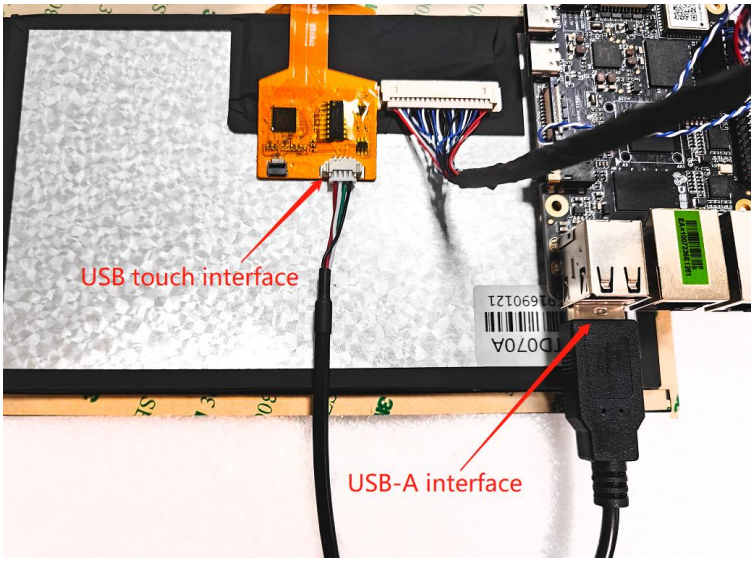


Figure 27 LVDS screen to DEBIX Model C completed

NOTE:

- **Usage of the USB Touch Interface on the LVDS Display:**

Connect the USB-A end of the USB-A to 4Pins data cable to the mainboard, and connect the 4Pins end to the Touch interface of the LVDS display, as shown in the figure below:



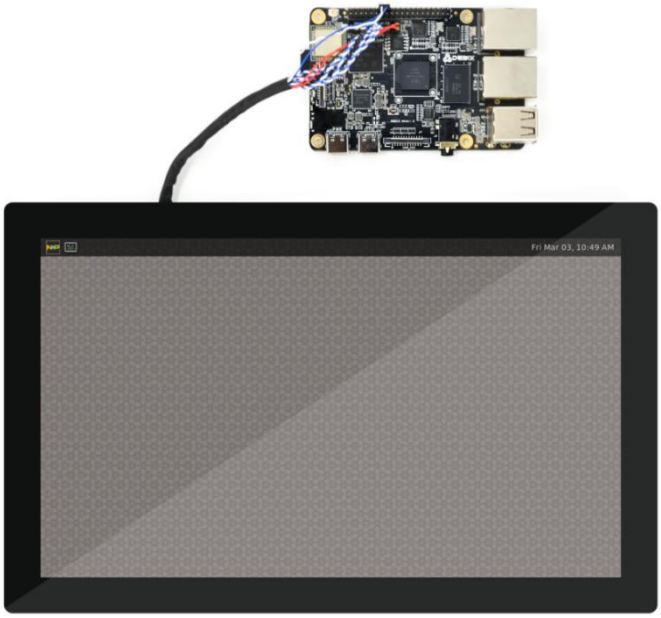
3) **Switching the Display Configuration:** After connecting the display, you need to select the corresponding configuration on the device. It is recommended to use the **USB flash drive switching method**, which is to modify the Debix_Settings.xml file to switch between the display and camera. ***(Save the modified file to a USB flash drive and insert it into the device. The device will automatically switch the configuration on boot and reboot to apply the changes.)*** As shown in the figure below, 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) After the configuration is switched and the device reboots successfully, the LVDS display will appear as shown below.



4.8. Usage of Camera

[Applies to DEBIX Model C only.]

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

- **Connection Instruction**

The camera module connection method for DEBIX Model C is the same as that of DEBIX Model A. For detailed instructions, please refer to the DEBIX Camera Module User Manual.

- **Image Preview Command**:
  - 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
```

- **Switching the Display Configuration:**

It is recommended to use **the USB flash drive switching method**, which is to modify the Debix_Settings.xml file to switch between the display and camera. ***(Save the modified file to a USB flash drive and insert it into the device. The device will automatically switch the configuration on boot and reboot to apply the changes.)*** As shown in the figure below, 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 11 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 x 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. Usage of Debug

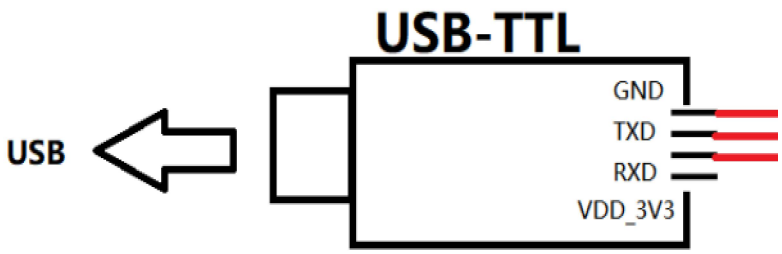
NOTE

The IO level of the debug serial port is 3.3V.

(1) **Hardware connection:** connect the debug interface **(Pin7, Pin9 and Pin11 on the J1 Interface)** to the USB-TTL module, as shown in the figure below:

- Connect the RXD of the debug serial port **(J1 Pin9)** to the TXD port of the USB-TTL module;
- Connect the TXD of the debug serial port **(J1 Pin11)** to the RXD port of the USB-TTL module;

- Connect the GND of the debug serial port **(J1 Pin7)** to that of the USB-TTL module.



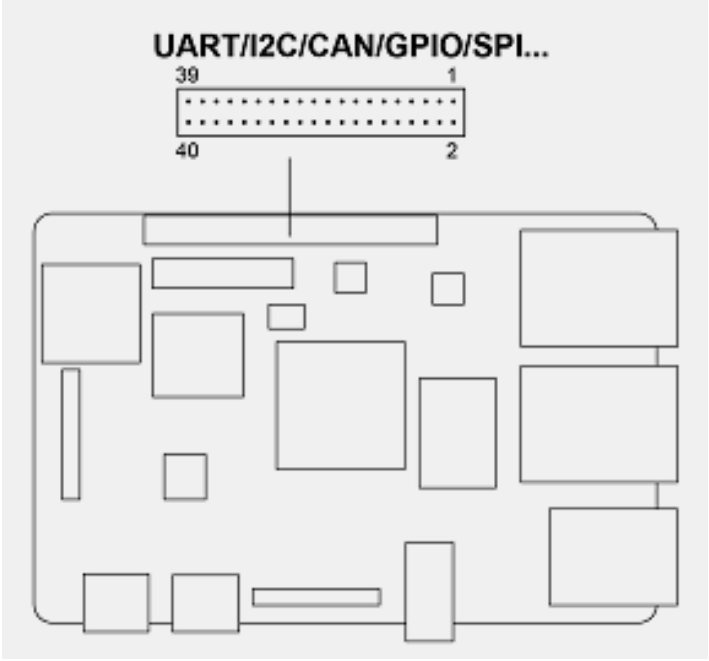


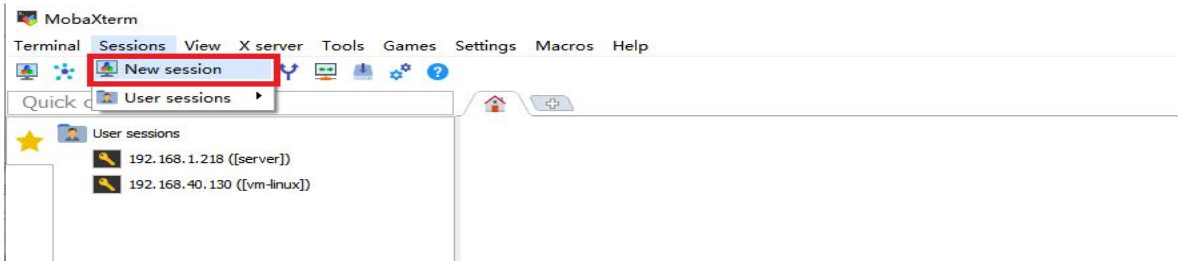
Table 12 The description of the debug interface

Function	Interface	Pin	Definition
Debug	J1	7	GND
		9	UART1_RXD
		11	UART1_TXD

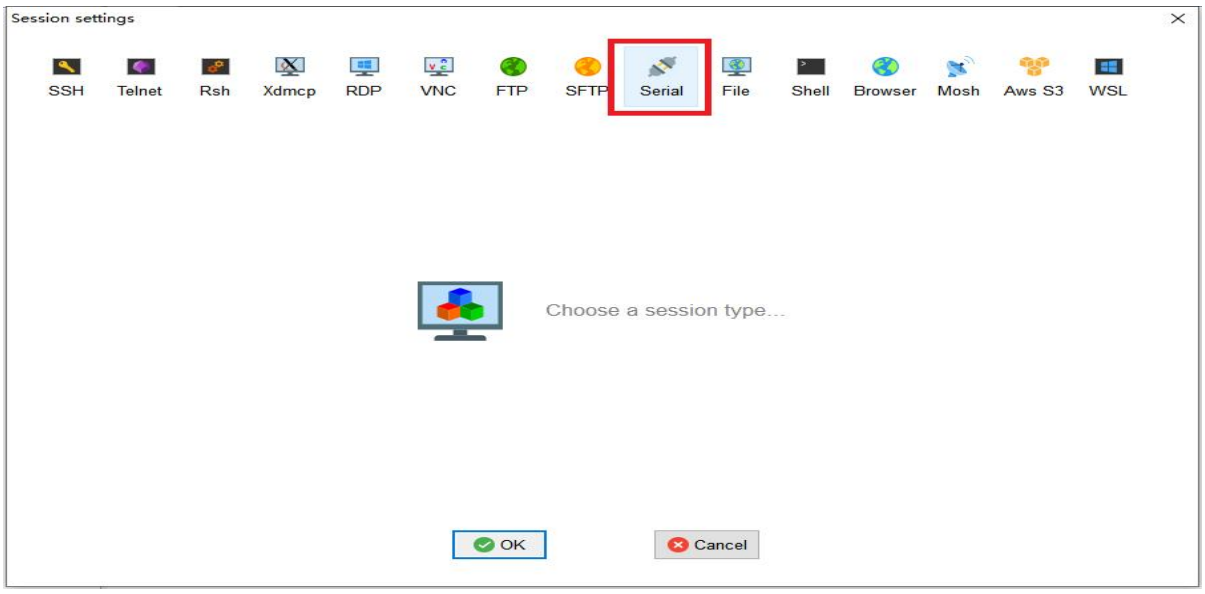
(2) Open the **Windows Device Manager** and check the serial port number of the

USB-TTL 3.3V device.

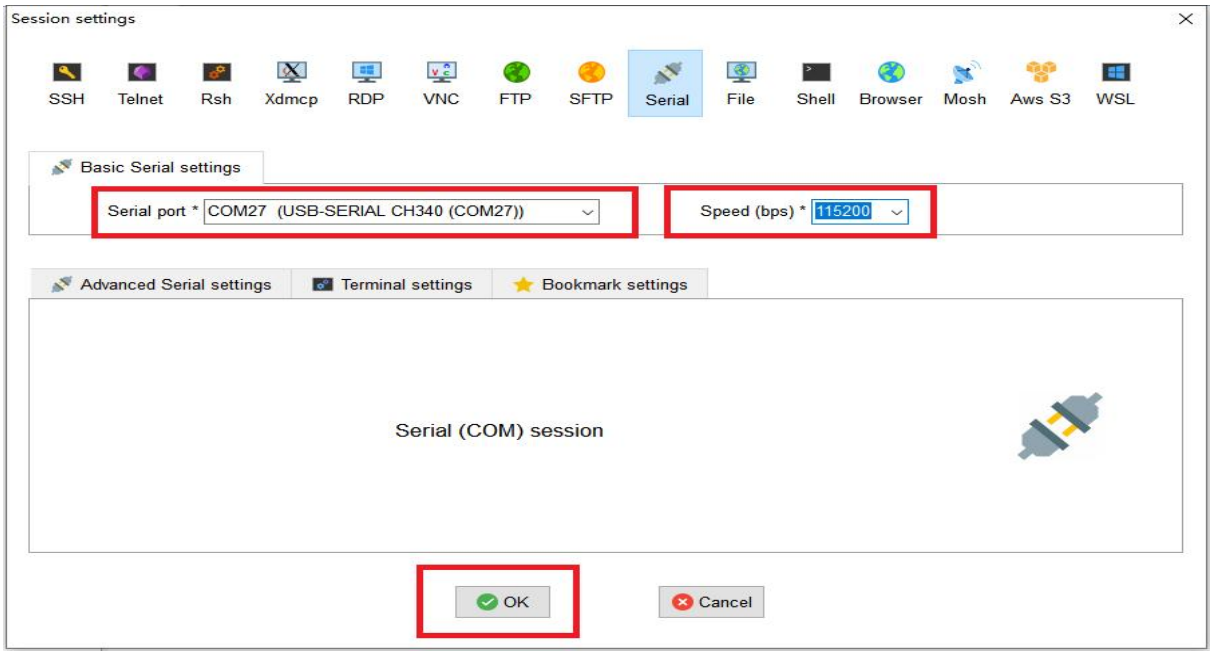
(3) Open **MobaXterm**, click **Sessions** on the menu bar, select **New session**.



(4) Select **Serial** in the pop-up "Session settings" dialog box.



(5) Change the port number to the COM port found in the device manager, set the Speed(bps) to **115200**, and click **OK**.



(6) On the terminal, you can see the boot process logs output by Uboot, Kernel, and System. After the system boot is complete,it will enter the serial console.

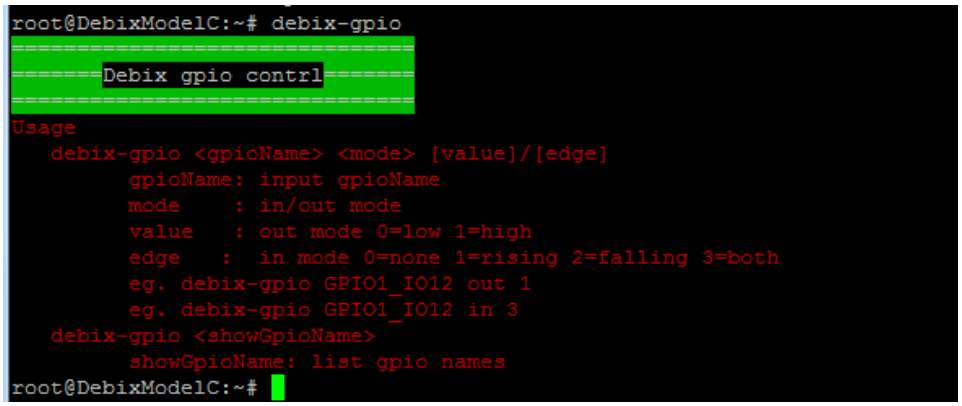
4.11. Usage of GPIO

DEBIX Model C/D OS provides built-in GPIO commands for GPIO control and configuration.

IMPORTANT

The GPIO voltage input of DEBIX Mode C/D 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:



- 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 showGpioName
J1
LVDS
Wi
Fi
DEBIX
SoC
RAM
ETH-1
ETH-2
USB2.0
pwr
usb
CSI
H
P
SoC : i.MX93
USB ports : 2
Ethernet ports : 2 (1000Mbps max. speed)
Wi-fi : True
Bluetooth : True
Camera (CSI) : 1
Display (LVDS+DSI) : 2
J1:
POE_VA1 (1) (2) POE_VA2
POE_VB1 (3) (4) POE_VB2
GND (5) (6) DC5V_IN
GND (7) (8) DC5V_IN
UART1_RXD (9) (10) SW_VDD5V
UART1_TXD (11) (12) VDD_3V3
I2C1_SCL (13) (14) VDD_1V8
I2C1_SDA (15) (16) ONOFF
GPIO1_IO08 (17) (18) SYS_nRST
GPIO1_IO09 (19) (20) GND
GPIO2_IO15 (21) (22) ADC_IN0
GPIO2_IO14 (23) (24) ADC_IN1
GPIO2_IO13 (25) (26) ADC_IN2
GPIO2_IO12 (27) (28) ADC_IN3
GND (29) (30) GND
USB20_5V_34 (31) (32) USB20_5V_34
USB_HUB_DM3 (33) (34) USB_HUB_DM4
USB_HUB_DP3 (35) (36) USB_HUB_DP4
GND (37) (38) GND
GND (39) (40) GND
For further information, please refer to https://www.debix.io/Document/index.html
Available gpio names:
GPIO1_IO08 GPIO1_IO09
GPIO2_IO15 GPIO2_IO14
GPIO2_IO13 GPIO2_IO12
root@DebixModelC:~#

3. Example: Set GPIO1_IO08 to output high, run command debix-gpio GPIO1_IO08 out 1, GPIO1_IO08 will output 3.3V.

```
root@DebixModelC:~# debix-gpio GPIO1_IO08 out 1
================================
=======Debix gpio contrl=======
================================
INFO:GPIO_NAME = GPIO1_IO08(8)
INFO:GPIO_MODE = out
INFO:GPIO_SET  = HIGH
root@DebixModelC:~# debix-gpio GPIO1_IO08 out 0
================================
=======Debix gpio contrl=======
================================
INFO:GPIO_NAME = GPIO1_IO08(8)
INFO:GPIO_MODE = out
INFO:GPIO_SET  = LOW
root@DebixModelC:~#
```

4. Example: Set GPIO2_IO12 to input rising edge, type command debix-gpio GPIO2_IO12 in 1, if Pin34 (GPIO2_IO12) detects power, the message INFO: pin:131 value=1; if the power is disconnected, the message INFO: pin:131 value=0.

```
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

Long-term operation of the DEBIX Model C/D may cause the CPU temperature to rise. Passive cooling is recommended to reduce the temperature of the CPU and device. An aluminum heatsink may be attached to the top of the CPU for cooling. An example heatsink block is shown below.

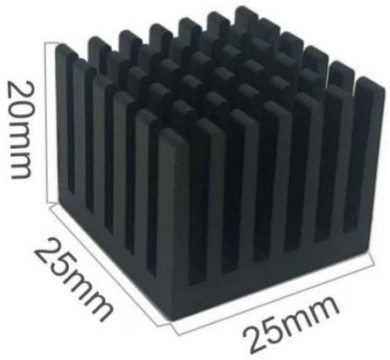


Figure 29 Aluminum alloy heatsink