



ED-IPC1220

User Manual

by EDA Technology Co., Ltd

built: 2026-04-09

1 Hardware Manual

This chapter introduces the product overview, packing list, appearance, button, indicator, and interface.

1.1 Overview

The ED-IPC1220 is an industrial computer based on the Raspberry Pi CM0, featuring 512MB RAM as standard, with optional 8GB or 16GB eMMC. It provides common interfaces such as USB, Ethernet, RS232, RS485, DI/DO, and ADC. It supports network access via Wi-Fi, Ethernet, and 4G, and integrates an RTC. It is primarily used in industrial control and IoT applications.



1.2 Packing List

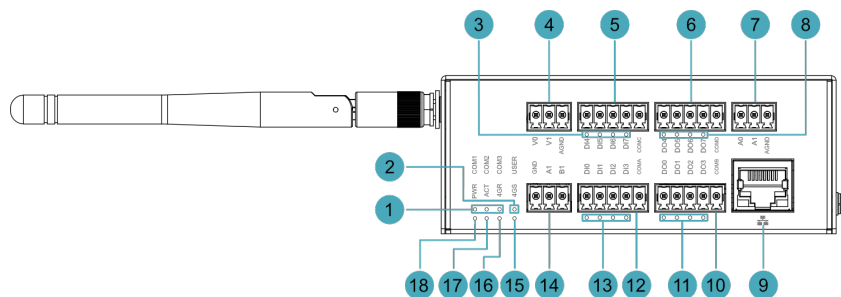
- 1 x ED-IPC1220 Unit
- [Optional Wi-Fi/BT Version] 1 x 2.4GHz Wi-Fi/BT Antenna
- [Optional 4G Version] 1 x 4G Antenna
- 3 x 3-Pin Phoenix Terminals
- 1 x 2-Pin Phoenix Terminal with Screw Holes
- 1 x 6-Pin Phoenix Terminal
- 4 x 5-Pin Phoenix Terminals

1.3 Appearance

Introduce the functions and definitions of the interfaces on each panel.

1.3.1 Front Panel

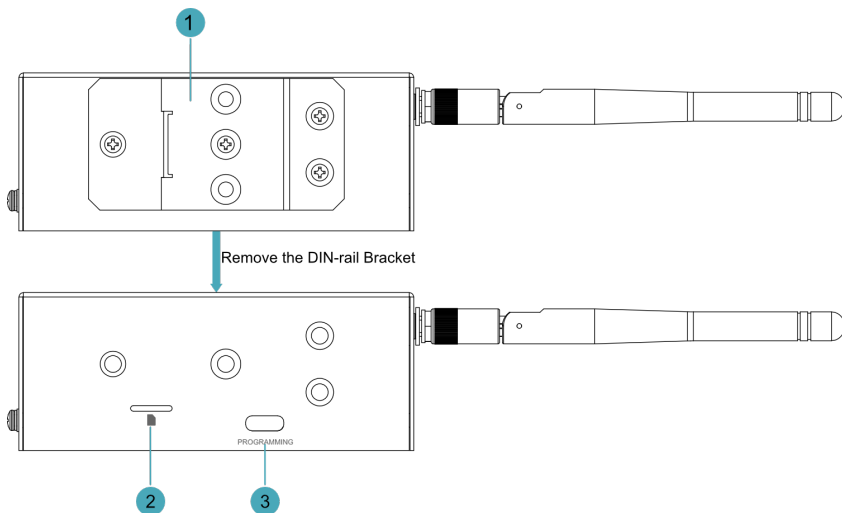
Introduce the types and definitions of the front panel interfaces.



NO.	Function Definition
1	3 x Serial port indicators, green, used to view the communication status of the serial ports. <ul style="list-style-type: none"> • COM1: Corresponds to serial port RS485-1 (GND/A1/B1) • COM2: Corresponds to serial port RS485-2 (GND/A2/B2) • COM3: Corresponds to serial port RS232 (GND/TX3/RX3)
2	1 x User indicator, green, users can customize its status according to actual needs.
3	4 x DI indicators (DI4~DI7), green, used to view the input status of DI ports (DI4~DI7).
4	2 x ADV interfaces, 3-Pin 3.5mm pitch Phoenix terminals, for connecting external sensors.
5	4 x DI interfaces (DI4~DI7), 5-Pin 3.5mm pitch Phoenix terminals, for connecting external sensors.
6	4 x DO interfaces (DO4~DO7), 5-Pin 3.5mm pitch Phoenix terminals, for connecting external loads.
7	2 x ADA interfaces, 3-Pin 3.5mm pitch Phoenix terminals, for connecting external sensors.
8	4 x DO indicators (DO4~DO7), green, used to view the output status of DO ports (DO4~DO7).
9	1 x 100M Ethernet Interface, RJ45 connector, with LED indicator, 10/100M auto-negotiation interface, for connecting to Ethernet.
10	4 x DO interfaces (DO0~DO3), 5-Pin 3.5mm pitch Phoenix terminals, for connecting external loads.
11	4 x DO indicators (DO0~DO3), green, used to view the output status of DO ports (DO0~DO3).
12	4 x DI interfaces (DI0~DI3), 5-Pin 3.5mm pitch Phoenix terminals, for connecting external sensors.
13	4 x DI indicators (DI0~DI3), green, used to view the input status of DI ports (DI0~DI3).
14	1 x RS485 interface (RS485-1), 3-Pin 3.5mm pitch Phoenix terminal, for connecting third-party controllers.
15	1 x 4G Signal indicator, red/green dual-color, used to view the status of the 4G signal.
16	1 x 4G Network registration indicator, red/green dual-color, used to view the network registration status of the 4G module.
17	1 x System status indicator, green, used to view the system's read/write data status.
18	1 x Power indicator, red, used to view the device's power status.

1.3.2 Rear Panel

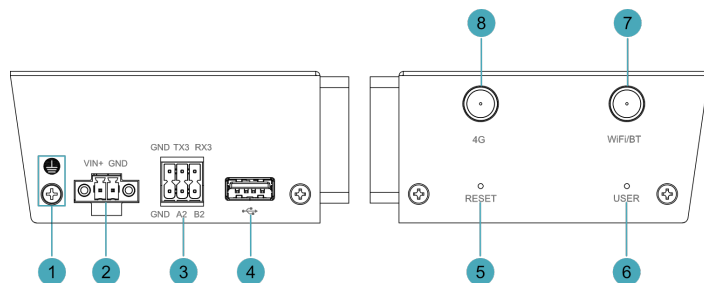
Introduces the interface types and definitions on the rear panel.



NO.	Function Definition
1	1 x Rail mounting bracket, used to install the ED-IPC1220 unit on a DIN rail via the bracket.
2	1 x Nano SIM card slot (optional) for installing a Nano SIM card to obtain 4G signal.
3	1 x Type-C USB interface, supports eMMC flashing via this interface.

1.3.3 Side Panel

Introducing the types and definitions of side panel interfaces.



NO.	Function Definition
1	1 x Grounding point, for connecting to the earth ground of the external power supply.
2	1 x DC input, 2-Pin 3.5mm pitch Phoenix terminal with screw holes, supports 9V~28V power input.
3	1 x RS485 interface (RS485-2) + 1 x RS232 Interface, 2x3-Pin 3.5mm pitch phoenix terminal, for connecting third-party controllers.
4	1 x USB 2.0 interface, Type-A connector, supports up to 480Mbps transmission rate.
5	1 x Reset Button, recessed button, pressing the button restarts the device.
6	1 x User Button, recessed button, users can customize its action according to actual needs.
7	1 x Wi-Fi/BT antenna interface (optional), SMA connector, for connecting the Wi-Fi/BT antenna.
8	1 x 4G antenna interface (optional), SMA connector, for connecting the 4G antenna.

1.4 Button

The ED-IPC1220 device contains 2 recessed buttons, namely the RESET button and the USER button, marked as "RESET" and "USER" on the housing.

- RESET Button: Pressing the RESET button resets the device, causing it to restart.
- USER Button: A user-defined button. No action is configured by default. The corresponding GPIO pin is GPIO25, supporting user-defined actions.

1.5 Indicator

Introduces the various states and meanings of the indicators on the ED-IPC1220 device.


Indicator	Status	Description
PWR	On	The device has been powered on.
	Blink	Power supply of the device is abnormal, please stop the power supply immediately.
	Off	The device is not powered on.
ACT	Blink	The system started successfully and is reading and writing data.
	Off	The device is not powered on or does not read and write data.
USER	On	User-defined
	Off	The device is not powered on or is not user-defined. The default state is off.
4GR	Red Steady On	SIM card failure
	Red Slow Blink	4G module initializing
	Red Fast Blink	4G module disconnected from network
	Green Steady On	4G module successfully registered to network
	Green Slow Blink	4G module data traffic normal
	Off	4G module not working
4GS	Red Steady On	4G module has no signal
	Green Steady On	4G module signal normal
	Red/Green Dual-color Steady On	4G module signal abnormal
	Off	4G module not working

Indicator	Status	Description
Green indicator of Ethernet port	On	The Ethernet connection is abnormal
	Blink	Data is being transmitted over the Ethernet port
	Off	The Ethernet connection is not set up
Yellow indicator of Ethernet port	On	The Ethernet connection is in the normal state
	Blink	The Ethernet connection is abnormal
	Off	The Ethernet connection is not set up
COM1~COM3	On/Blink	Data is being transmitted
	Off	The device is not powered on or there is no data transmission
DI0~DI7	On/Blink	Input signal detected
	Off	Device not powered on or no data transmission
DO0~DO7	On/Blink	Output signal detected
	Off	Device not powered on or no data transmission

1.6 Interface

Introduce the definitions and functions of the various interfaces in the product.

1.6.1 Nano SIM Card Slot (Optional)

The ED-IPC1220 device contains 1 Nano SIM card slot. The slot interface is marked with "", used for installing a Nano SIM card to obtain 4G signal.

TIP

If the customer's selected product does not include the 4G function, the device will not include the Nano SIM card slot.

1.6.2 Power Interface

The ED-IPC1220 device contains 1 power input interface, using a 2-Pin 3.5mm pitch Phoenix terminal with screw holes. The interface is marked as "VIN+/GND" and supports 9V~28V power input.

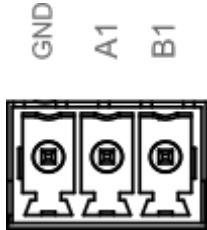
1.6.3 RS485/RS232 Interface

The ED-IPC1220 device contains 2 RS485 interfaces and 1 RS232 interface, using one 3-Pin 3.5mm pitch Phoenix terminal (1 x RS485) and one 2x3-Pin 3.5mm pitch Phoenix terminal (1 x

RS485 + 1 x RS232). The interfaces are marked as "GND/A1/B1", "GND/A2/B2", and "GND/TX3/RX3" respectively, supporting connection to third-party controllers.

1.6.3.1 3-Pin 3.5mm Pitch Phoenix Terminal (1 x RS485)

The terminal pin definitions are as follows:

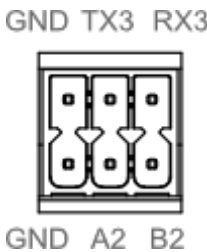
	Pin ID	Pin Name
	1	GND
	2	A1 (RS485-1_A)
	3	B1 (RS485-1_B)

The corresponding CM0 GPIO pin names for the RS485 interface are as follows:

Signal	CM0 GPIO Name
RS485-1_A	GPIO14
RS485-1_B	GPIO15

1.6.3.2 2x3-Pin 3.5mm Pitch Phoenix Terminal (1 x RS485 + 1 x RS232)

The terminal pin definitions are as follows:

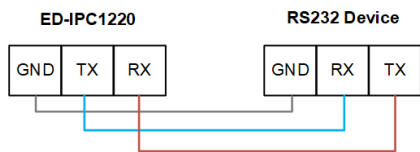
	Pin ID	Pin Name
	1	GND
	2	A2 (RS485-2_A)
	3	B2 (RS485-2_B)
	4	GND
	5	TX3
	6	RX3

The corresponding SPI pin names for the RS485 and RS232 interfaces are as follows:

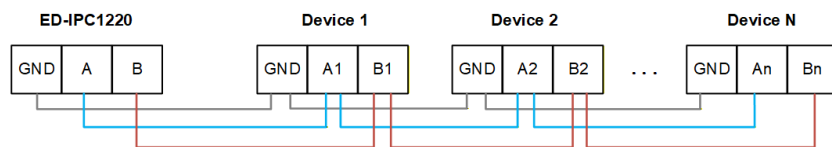
Signal	SPI Pin Out
RS485-2_A	TX1
RS485-2_B	RX1
TX3	TX2
RX3	RX2

1.6.3.3 Connecting Cables

The RS232 connection diagram is as follows:



The RS485 connection diagram is as follows:



RS485 Termination Resistor Configuration

The ED-IPC1220 device contains 2 RS485 interfaces. A 120Ω termination resistor is reserved between lines A and B on the RS485 circuit. Inserting a jumper cap enables this termination resistor. By default, no jumper cap is connected, and the 120Ω termination resistor function is disabled. The locations of the termination resistors on the PCBA are J11 (for RS485-1) and J13 (for RS485-2).

TIP

The device housing needs to be opened to view the location of the 120Ω termination resistors.

1.6.4 DI Interface

The ED-IPC1220 device contains 8 DI interfaces, using two 5-Pin 3.5mm pitch Phoenix terminals, supporting connection to external sensors.

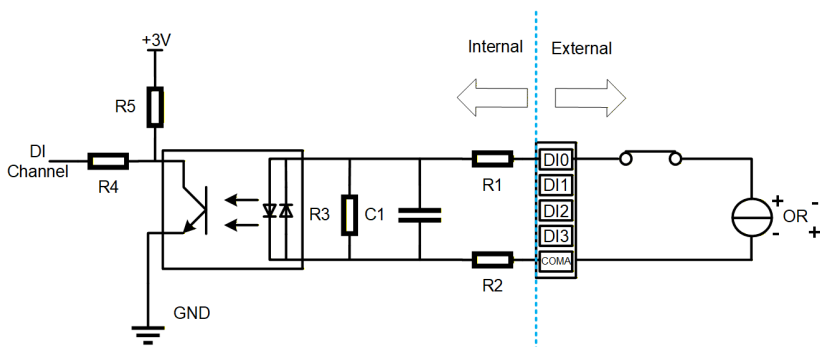
The terminal pin definitions are as follows:

	Pin ID	Pin Name
	1	DI0
	2	DI1
	3	DI2
	4	DI3
	5	COMA
	6	DI4
	7	DI5
8	DI6	

9	DI7
10	COMC

Connecting Cables

The connection diagram for a single DI interface is as follows:



Parameter	Description
Input Type	Wet Contact (NPN)
Isolation Protection	3.75kV
COM Terminal	Every 4 DI channels share one common COM terminal • DI0, DI1, DI2, and DI3 share COMA • DI4, DI5, DI6, and DI7 share COMC
DI to COM	• ON State: 5~30 VDC or -30~-5 VDC • OFF State: 0~2 VDC or -2~0 VDC

1.6.5 DO Interface

The ED-IPC1220 device contains 8 DO interfaces, using two 5-Pin 3.5mm pitch Phoenix terminals, supporting connection to external loads.

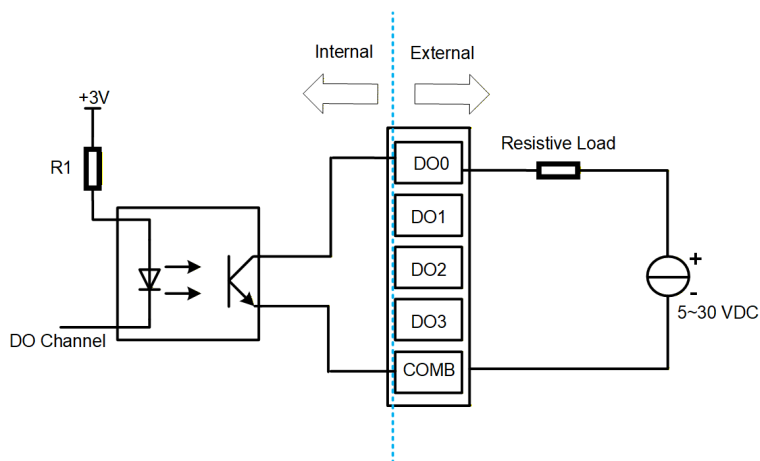
The terminal pin definitions are as follows:

Pin ID	Pin Name
1	DO0
2	DO1
3	DO2
4	DO3
5	COMB
6	DO4
7	DO5

	8	DO6
	9	DO7
	10	COMD

Connecting Cables

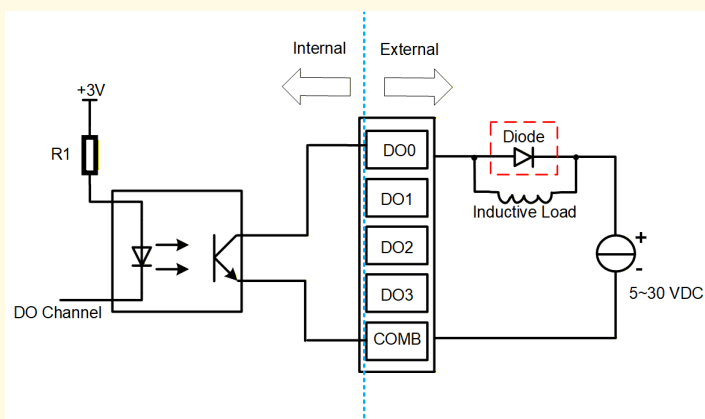
The connection diagram for a single DO interface is as follows:



Parameter	Description
Sensor Type	NPN
Isolation Protection	3.75kV
COM Terminal	Every 4 DO channels share one common COM terminal: • DO0, DO1, DO2, and DO3 share COMB • DO4, DO5, DO6, and DO7 share COMD
Output	5~30 VDC, 24 VDC recommended, maximum current 0.2A (per channel)

WARNING

If an inductive load is connected to a DO channel, it is recommended to add a freewheeling diode (as shown below) to the circuit for protection. Please select an appropriate freewheeling diode according to the specifications of the inductive load.



1.6.6 ADV Interface

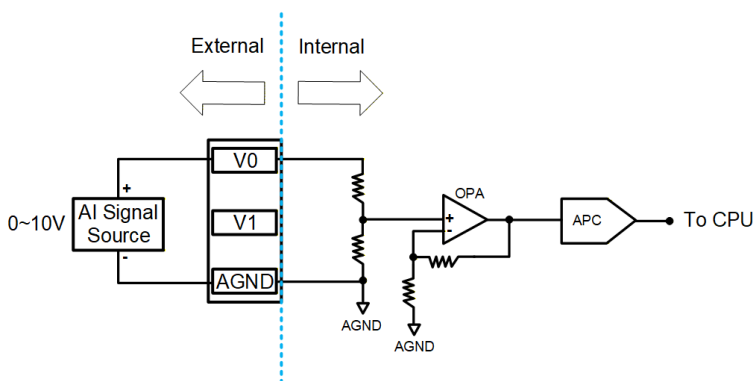
The ED-IPC1220 device contains 2 Analog Voltage Input (ADV) interfaces, using 3-Pin 3.5mm pitch Phoenix terminals, supporting connection to external sensors.

The terminal pin definitions are as follows:

	Pin ID	Pin Name
	1	V0
	2	V1
	3	AGND

Connecting Cables

The connection diagram for a single ADV interface is as follows:



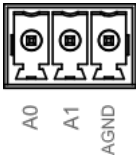
Parameter	Description
Signal Type	Voltage Input (Differential)
Input Voltage Range	0~10V
Input Impedance	10MΩ
Resolution	12 bit
Accuracy	±0.1% of full scale at 25°C; ±0.3% of full scale from -20°C to 60°C
Sampling Rate	3KHz

1.6.7 ADA Interface

The ED-IPC1220 device contains 2 Analog Current Input (ADA) interfaces, using 3-Pin 3.5mm pitch Phoenix terminals, supporting connection to external sensors.

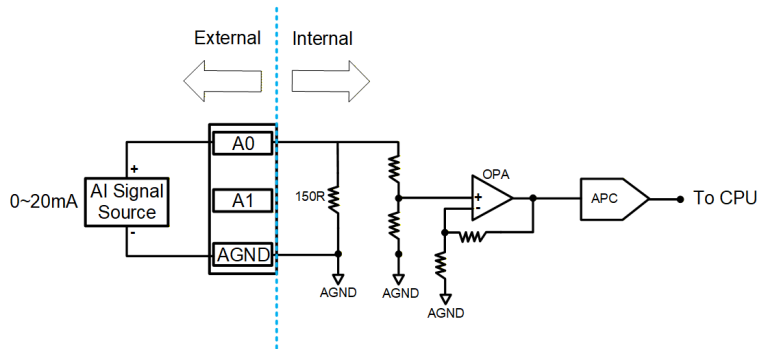
The terminal pin definitions are as follows:

Pin ID	Pin Name
--------	----------

	1	A0
	2	A1
	3	AGND

Connecting Cables

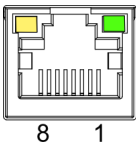
The connection diagram for a single ADA interface is as follows:



Parameter	Description
Signal Type	Current Input (Differential)
Input Current Range	0~20mA
Current Sampling Resistor	120Ω
Input Impedance	10MΩ
Resolution	12 bit
Accuracy	±0.1% of full scale at 25°C; ±0.3% of full scale from -20°C to 60°C
Sampling Rate	3KHz


1.6.8 100M Ethernet Interface

The ED-IPC1220 device contains 1 auto-negotiating 10/100M Ethernet interface, marked with "E". It uses an RJ45 connector. When connecting to Ethernet, it is recommended to use a Cat6 or higher specification network cable. The corresponding pin definitions for the connector are as follows:

	Pin ID	Pin Name
	1	TX+
	2	TX-
	3	Rx+
	4	-

5	-
6	RX-
7	-
8	-

1.6.9 USB 2.0 Interface

The ED-IPC1220 device contains 1 USB 2.0 interface, marked with "", a standard Type-A interface. It supports connecting standard USB 2.0 peripherals, with a maximum transmission rate of 480Mbps.

1.6.10 Type-C USB Interface

The ED-IPC1220 device contains 1 Type-C USB interface, marked as "PROGRAMMING". It supports programming the device's eMMC by connecting to a PC.

1.6.11 Antenna Interface (Optional)

The ED-IPC1220 device contains 2 SMA antenna interfaces, marked as "4G" and "WiFi/BT" respectively, for connecting the 4G antenna and the Wi-Fi/BT antenna.

TIP

If the customer's selected product does not include the 4G function and Wi-Fi function, the device will not include the antenna interfaces.

2 Installing Components (Optional)

This chapter describes how to install optional components.

2.1 Install Antenna

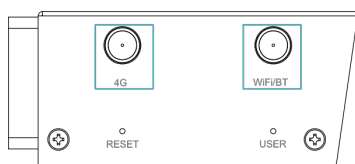
If the purchased ED-IPC1220 device includes 4G and Wi-Fi functions, you need to install the antennas before using the device.

Preparation:

The corresponding antennas have been obtained from the packing box. If multiple antennas are included, distinguish them by the labels on the antennas.

Steps:

1. Identify the location of the antenna interfaces on the side of the device, as shown in the marked positions in the figure below.



2. Align the interfaces on the device and the antenna, and tighten the antenna clockwise until it is secure and will not fall off.

2.2 Install Nano SIM Card

If the purchased ED-IPC1220 device includes the 4G function, you need to install a Nano SIM card before using the 4G function.

NOTE

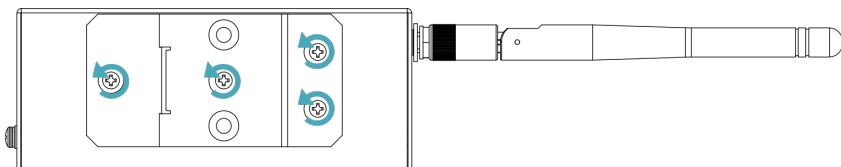
Please turn off the power before inserting or removing the Nano SIM card.

Preparation:

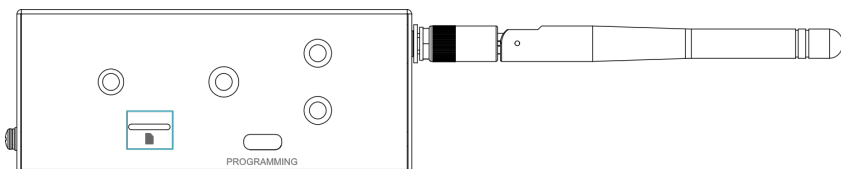
- The 4G Nano SIM card to be used has been obtained.
- The device power is disconnected.

Steps:

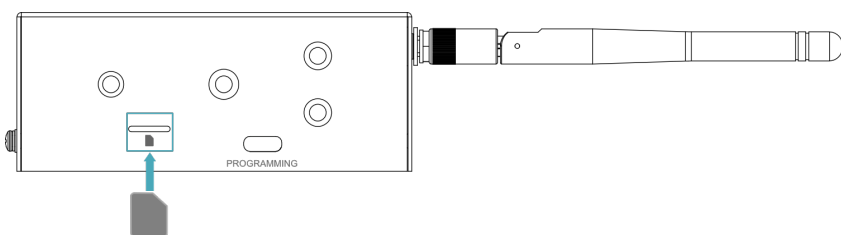
1. Use a Phillips screwdriver to unscrew the 4 screws on the DIN rail bracket counterclockwise, and remove the default DIN rail bracket.



2. Identify the location of the Nano SIM card slot on the side of the device, as shown in the marked position in the figure below.



3. Insert the Nano SIM card with its contact side facing up into the corresponding slot. A click indicates successful installation.



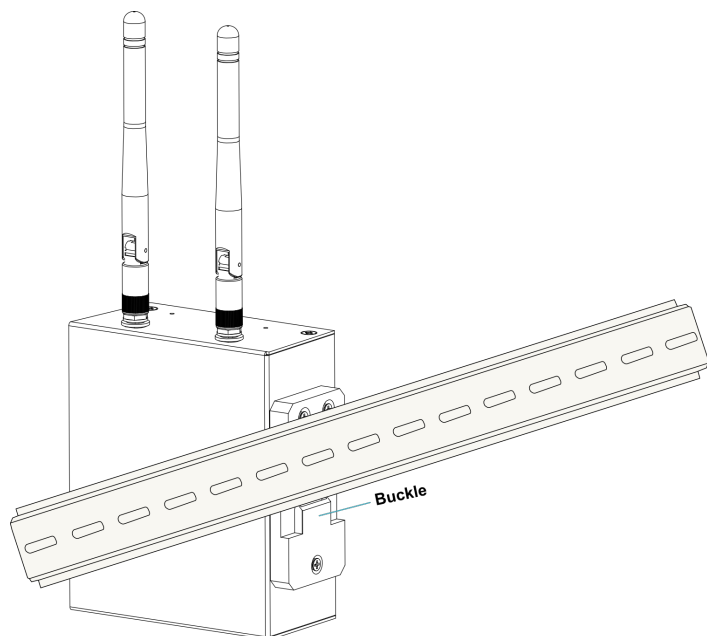
4. Install the DIN rail bracket back onto the device.

3 Installing Device

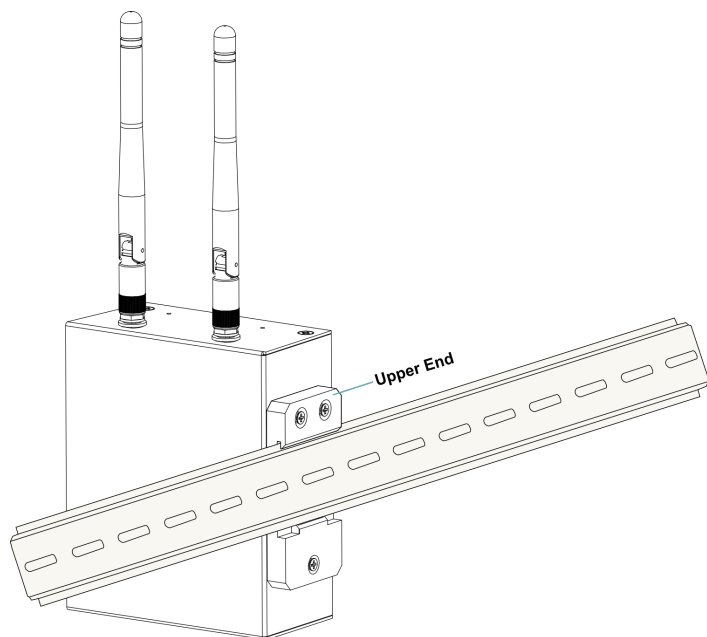
The ED-IPC1220 supports DIN rail mounting. A DIN rail bracket is included as standard and pre-installed.

Steps:

1. Position the side of the device with the rail bracket against the target rail. Hook the retractable buckle of the bracket onto the lower edge of the DIN rail from bottom to top.



2. Hold the device and press the buckle firmly upward until the top of the bracket fully snaps onto the upper edge of the DIN rail.



4 Starting Up and Logging In to the Device

This chapter describes the specific operations for connecting cables, starting up the device, and logging in.

4.1 Connecting Cables

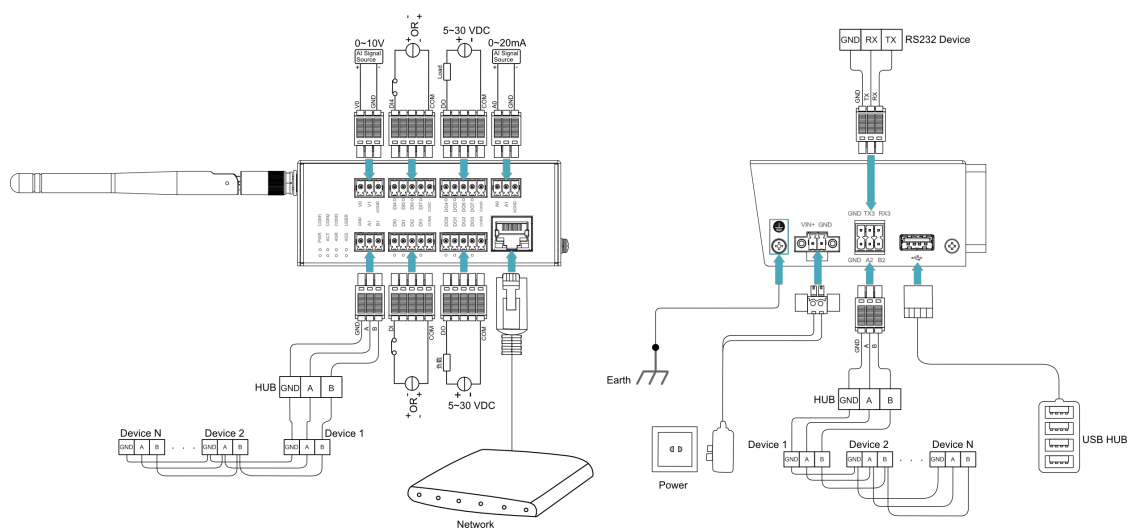
Describes the method for connecting the cables.

Preparation:

- Have obtained accessories that are functioning correctly, such as a USB hub, a voltage/current signal source, a router, and a power adapter.
- Have access to a functioning network.
- Have obtained functioning network cables and other signal connection cables.

Schematic diagram of connecting cables:

For the pin definitions of each interface and the specific wiring methods, please refer to [1.6 Interface](#).



TIP

The RS485-1 (GND/A1/B1) port has the serial console login mode enabled by default. If it needs to be used as a regular serial port, you must first disable the serial console login mode using the `raspi-config` command.

4.2 Starting Up the System

The ED-IPC1220 device does not include a power switch. The system will start up once power is connected.

- The red PWR LED lights up, indicating the device is receiving power normally.

- The green ACT LED blinks, indicating the system is starting up normally.

4.3 Logging In to the System

The ED-IPC1220 device includes two RS485 ports. The RS485-1 (GND/A1/B1) port has the serial console login mode enabled by default, allowing users to log in to the device through this port. The following describes the detailed procedure.

TIP

Default username: `pi` ; Default password: `raspberrypi` .

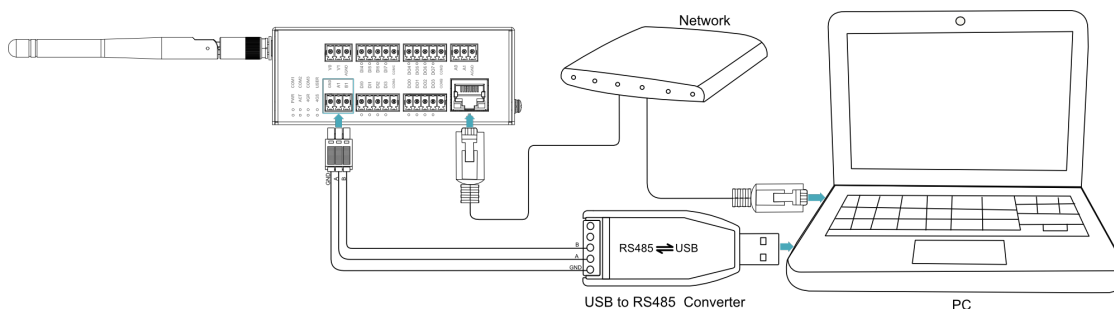
Preparation:


- Have a functioning Windows PC, and a serial terminal tool (such as MobaXterm) is installed on the Windows PC.
- Have connected both the ED-IPC1220 and the PC to the same router (with the router in DHCP mode for automatic IP assignment), so that the ED-IPC1220 and the PC are on the same IP subnet.
- Have a USB-to-RS485 converter.
- Have obtained necessary connection cables (for connecting the device's RS485-1 port to the USB-to-RS485 converter).

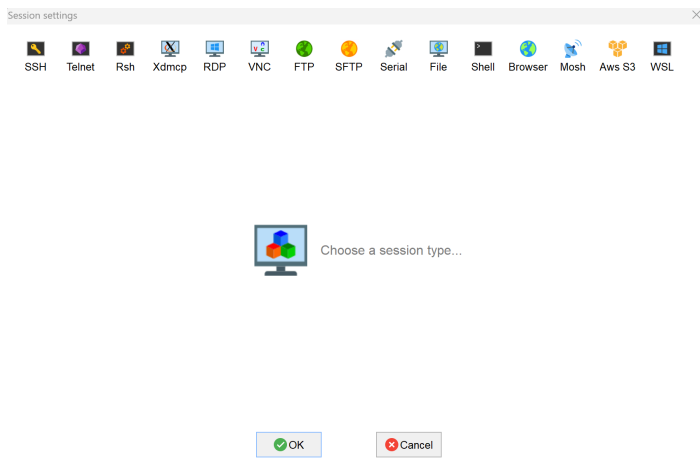
Steps:


The following procedure uses a Windows system as an example.

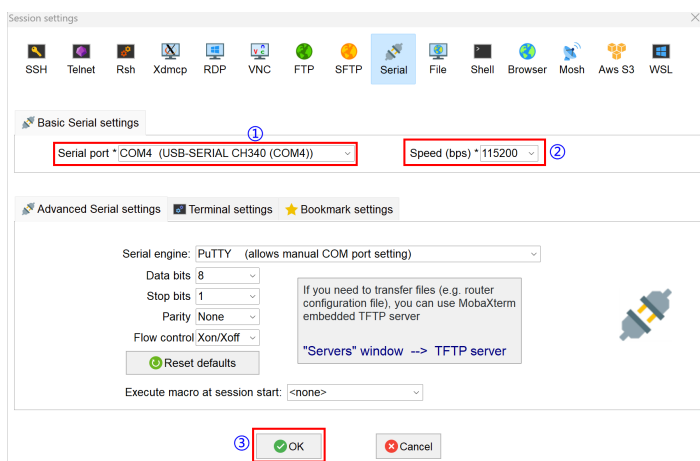
1. Connect the ED-IPC1220 and the PC to the same router (so they are on the same IP subnet). Then, connect the RS485-1 port on the ED-IPC1220 to the Windows PC via the USB-to-RS485 converter, as shown in the figure below.



2.  Open `MobaXterm` , click the `Session` button to open the session creation window, as shown below.

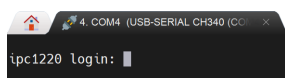


3. Click the  icon to open the serial connection interface. Set the "Serial port" and "Speed", then click "OK".

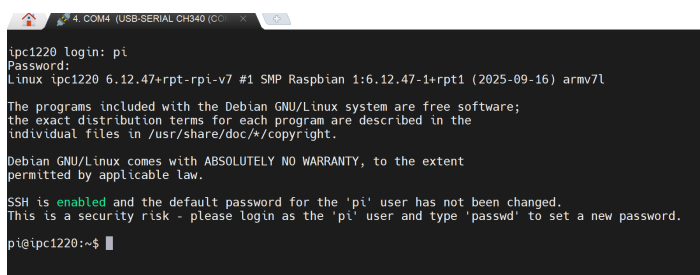


- "Serial port": Select the serial port corresponding to the USB-to-RS485 converter.
- "Speed(115200)": Select "115200".
- "Terminal settings" and "Advanced Serial settings" can be kept as default.

4. In the opened terminal pane, press "Enter" to enter login mode.



5. Enter the username and password as prompted to log in to the system.



TIP

Default username: `pi` ; Default password: `raspberrypi` .

6. Execute the following command to query the device's IP address.

```
ifconfig
```

```
sh
```

5 Configuring System

This chapter describes the specific operations for system configuration.

5.1 Finding Device IP

Since the ED-IPC1220 does not include an HDMI interface, remote login is required for device management in practical applications, necessitating obtaining the device's IP address.

5.1.1 Querying via Local Serial Port Login

Logging In to the System via Local Serial Port

5.1.2 Querying by Logging into the Router

After the device starts up normally, you can log into the router to view the current device IP.

Preparation:

- The device is connected to the network via a router.
- You have obtained the IP address (e.g., 192.168.X.X) and network password of the router in your network.

Steps:

1. Open a browser, enter the router's IP address (e.g., 192.168.X.X) in the address bar, and press `Enter` to access the router login interface.
2. Input the network password as prompted to access the router management interface.
3. Find the device's IP address by its hostname in the list of connected devices within the management interface.

TIP

The default hostname for the ED-IPC1220 device is `ipc1220` .

5.1.3 Querying via Ping in Windows Terminal

Preparation:

- Have a functioning Windows PC.
- Have connected both the ED-IPC1220 and the PC to the same router, so they are on the same IP subnet.

Steps:

1. Open the Windows PC's terminal pane as an administrator.
2. Execute the following command in the terminal pane to ping the ED-IPC1220 device.

```
ping -4 ipc1220.local
```

sh

- ipc1220 represents the device's hostname.

TIP

This method is only applicable when there is only one ED-IPC1220 device on the same subnet.


5.2 Connecting to the Device via SSH

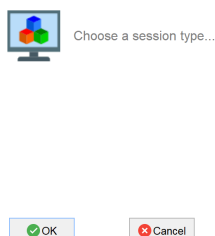
After the device starts up normally, you can choose to connect to it remotely via SSH for configuration or debugging. Users can choose their own remote login tool. The following uses **MobaXterm** as an example.


Preparation:

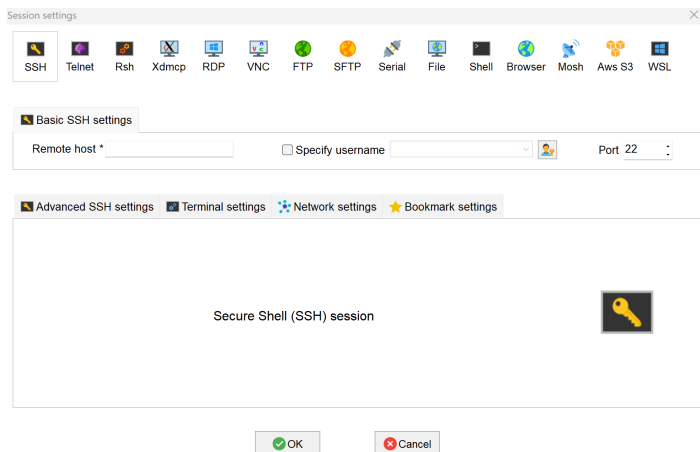
- Have a functioning Windows PC with the **MobaXterm** tool installed.
- Have connected both the ED-IPC1220 and the PC to the same router, so they are on the same IP subnet.
- Have obtained the IP address of the ED-IPC1220.

Steps:

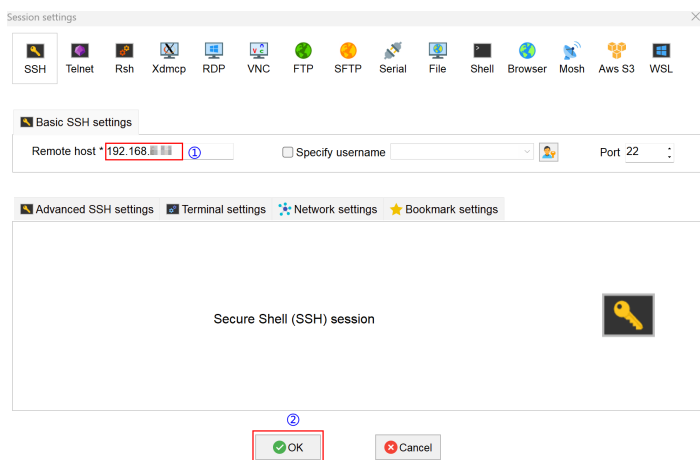
1.  Open **MobaXterm**, click the **Session** button to open the session creation window, as shown below.



2.  Click the **SSH** icon in the top-left corner to open the SSH connection interface.



3. Input the obtained device IP address, then click "OK".



4. Click "Accept" in the pop-up prompt box to enter the system login interface.
5. Input the username and password as prompted. After successful login, you will enter the system.

TIP

Default username: `pi` , Default password: `raspberrypi` .

```

login as: pi
pi@192.168.1.100's password:
• MobaXterm Personal Edition v23.0 •
  (SSH client, X server and network tools)
  > SSH session to pi@192.168.1.100
    • Direct SSH      : ✓
    • SSH compression : ✓
    • SSH-browser     : ✓
    • X11-forwarding  : ✓ (remote display is forwarded through SSH)
  > For more info, ctrl+click on help or visit our website.

Linux ipc1220 6.12.47+rpt-rpi-v7 #1 SMP Raspbian 1:6.12.47-1+rpt1 (2025-09-16) armv7l

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Tue Mar 24 11:58:47 2026 from 192.168.1.100

SSH is enabled and the default password for the 'pi' user has not been changed.
This is a security risk - please login as the 'pi' user and type 'passwd' to set a new password.

pi@ipc1220:~$

```

5.3 Configuring Storage Devices

Configuring Storage Devices

5.4 Configuring Ethernet IP

Configuring Ethernet IP

5.5 Configuring Wi-Fi

Configuring Wi-Fi

5.6 Configuring Bluetooth

Configuring Bluetooth

5.7 Configuring 4G

The 4G function is enabled by default on the ED-IPC1220 device. After inserting the SIM card and powering on the device, wait a few minutes for the 4G network to connect automatically.

5.7.1 Scenarios Without APN Configuration

If the user's 4G network does not require APN configuration, you can follow the steps below to check the 4G network status.

Preparation:

- The ED-IPC1220 device has started up normally, and you are logged into the system.
- A Nano SIM card with 4G service is correctly installed in the ED-IPC1220's SIM card slot.

NOTE

Please turn off the power before inserting or removing the Nano SIM card.

Steps:

1. Open a command pane and execute the following command to check if the 4G network is connected.

```
ifconfig
```

```
sh
```

The return information is as shown below (the usb0 interface represents the 4G interface):

```

pi@raspberrypi:~$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.1.1 netmask 255.255.255.0 broadcast 192.168.1.255
    inet6 fe80::732e:31bc:ae26:f477 prefixlen 64 scopeid 0x20<link>
    ether 00:e0:9a:1b:05:24 txqueuelen 1000 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

usb0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.191.34.79 netmask 255.0.0.0 broadcast 10.255.255.255
    inet6 fe80::c704:40cc:227f:72b6 prefixlen 64 scopeid 0x20<link>
    ether ae:0c:29:a3:9b:6d txqueuelen 1000 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

wlan0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    ether 8c:1f:64:34:a0:04 txqueuelen 1000 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

```

- If the usb0 interface in the returned information displays a specific IP address, it indicates that the 4G network is connected.
- If the usb0 interface in the returned information does not display a specific IP address, it indicates that the 4G network is not connected.

2. Execute the following command to query the status of the 4G service.

```
sudo systemctl status ed-lte-daemon.service
```

sh

The returned information is as shown in the figure below:

```

pi@ipc1220:~$ sudo systemctl status ed-lte-daemon.service
● ed-lte-daemon.service - EDATEC QML Reconnect service
   Loaded: loaded (/usr/lib/systemd/system/ed-lte-daemon.service; enabled; preset: enabled)
   Active: active (running) since Thu 2026-03-19 09:33:52 GMT; 16min ago
     Invocation: 8d2a2fd88cf5421caac488d27e3e9c94
   Main PID: 1541 (ed-lte-tool)
      Tasks: 3 (limit: 259)
     CPU: 17.191s
   Group: /system.slice/ed-lte-daemon.service
          └─1541 /usr/local/bin/ed-lte-tool --daemon

Mar 19 09:33:53 ipc1220 ed-lte-tool[1541]: Sending AT command: AT+CFUN=4
Mar 19 09:33:55 ipc1220 ed-lte-tool[1541]: [2026-03-19 09:33:55][INFO] RUN: AT+CFUN=1
Mar 19 09:33:55 ipc1220 ed-lte-tool[1541]: Sending AT command: AT+CFUN=1
Mar 19 09:33:57 ipc1220 ed-lte-tool[1541]: [2026-03-19 09:33:57][INFO] RUN: reset 4G.
Mar 19 09:33:57 ipc1220 ed-lte-tool[1541]: [2026-03-19 09:33:57][INFO] Resetting None, Some("4G_RST"), None
Mar 19 09:33:57 ipc1220 ed-lte-tool[1541]: [2026-03-19 09:33:57][INFO] Using GPIO name: 4G_RST
Mar 19 09:33:58 ipc1220 ed-lte-tool[1541]: [2026-03-19 09:33:58][INFO] LTE reset successful via GPIO gpiochip2: line 3 (initial: LOW)
Mar 19 09:33:58 ipc1220 ed-lte-tool[1541]: [2026-03-19 09:33:58][ERROR] LTE module not ready.
Mar 19 09:33:59 ipc1220 ed-lte-tool[1541]: [2026-03-19 09:33:59][INFO] RUN: AT+CFUN=4
Mar 19 09:33:59 ipc1220 ed-lte-tool[1541]: Sending AT command: AT+CFUN=4

```

- If the information at the marked position in the returned message shows "Active: active (running)", it indicates that the 4G status is normal.
- If the information at the marked position in the returned message shows "Active: inactive (dead)", it indicates that the 4G status is abnormal.

5.7.2 Scenarios With APN Configuration

If the user's 4G network requires APN configuration, it can be configured by following the steps below.

Preparation:

- The ED-IPC1220 device has started up normally.
- A Nano SIM card with 4G service is correctly installed in the ED-IPC1220's SIM card slot.

- You have obtained the APN name, username, and password. The following uses the example information below.
 - APN Name: APN1
 - Username: admin
 - Password: admin

NOTE

Please turn off the power before inserting or removing the Nano SIM card.

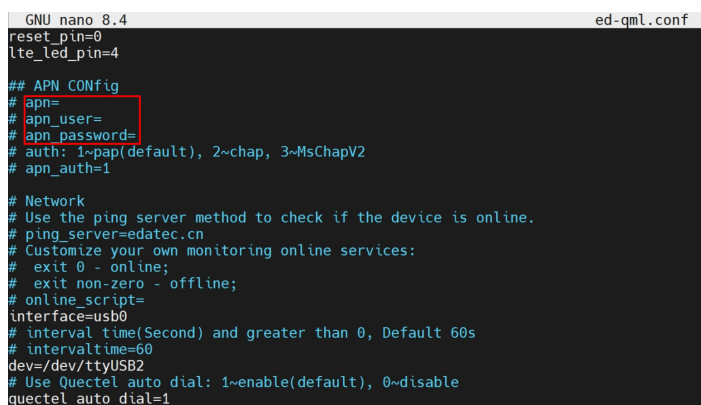
Steps:

1. Open a command pane and execute the following commands in sequence to open the `ed-qml.conf` configuration file.

```
cd /etc/
sudo nano ed-qml.conf
```

sh

2. Set "apn", "apn_user", and "apn_password" in the "APN Config" section as needed.



```
GNU nano 8.4 ed-qml.conf
reset_pin=0
lte_led_pin=4

## APN CONFIG
# apn=
# apn_user=
# apn_password=
# auth: 1~pap(default), 2~chap, 3~MsChapV2
# apn_auth=1

# Network
# Use the ping server method to check if the device is online.
# ping_server=edatec.cn
# Customize your own monitoring online services:
# exit 0 - online;
# exit non-zero - offline;
# online_script=
interface=usb0
# interval time(Second) and greater than 0, Default 60s
# intervaltime=60
dev=/dev/ttyUSB2
# Use Quectel auto dial: 1~enable(default), 0~disable
quectel_auto_dial=1
```

If the `ed-qml.conf` file does not contain the default configuration example, you can add the following information at the end of the file to customize the APN configuration, "ping_server" in the "Network" section, "online_script", etc.

- APN configuration example:

```
apn=custom_value
apn_user=custom_value
apn_password=custom_value
auth: 1~pap(default), 2~chap, 3~MsChapV2
apn_auth=1
```

sh

- Network configuration, default is "edatec.cn".

```
ping_server=custom_value
```

sh

- Configure custom script, specify the custom path.

```
online_script=custom_path
```

sh

3. Press **Ctrl+O** to save the file, then press **Enter**, and finally press **Ctrl+X** to exit the file editing mode.
4. Open a command pane and execute the following command to check if the 4G network is connected.

```
ifconfig
```

sh

The return information is as shown below (the usb0 interface represents the 4G interface):

```
pi@raspberrypi:~$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.1.1 netmask 255.255.255.0 broadcast 192.168.1.255
    inet6 fe80::732e:31bc:ae26:f477 prefixlen 64 scopeid 0x20<link>
    ether 00:e0:9a:1b:05:24 txqueuelen 1000 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

usb0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.191.34.79 netmask 255.0.0.0 broadcast 10.255.255.255
    inet6 fe80::c704:40cc:227f:72b6 prefixlen 64 scopeid 0x20<link>
    ether ae:0c:29:a3:9b:6d txqueuelen 1000 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

wlan0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    ether 8c:1f:64:34:a0:04 txqueuelen 1000 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

- If the usb0 interface in the returned information displays a specific IP address, it indicates that the 4G network is connected.
- If the usb0 interface in the returned information does not display a specific IP address, it indicates that the 4G network is not connected.

5. Execute the following command to query the status of the 4G service.

```
sudo systemctl status ed-lte-daemon.service
```

sh

The return information is as shown below:

```
pi@ipc1220:~$ sudo systemctl status ed-lte-daemon.service
● ed-lte-daemon.service - EDITEC 4G Reconnect service
   Loaded: loaded (/usr/lib/systemd/system/ed-lte-daemon.service; enabled; preset: enabled)
   Active: active (running) since Thu 2026-03-19 09:18:32 GMT; 16min ago
  Invocation: 8d292f488c45421caac480d27e369c94
    Main PID: 1541 (ed-lte-tool)
      Tasks: 3 (limit: 359)
         CPU: 17.191s
   CGroup: /system.slice/ed-lte-daemon.service
           └─1541 /usr/local/bin/ed-lte-tool --daemon

Mar 19 09:33:53 ipc1220 ed-lte-tool[1541]: Sending AT command: AT+CFUN=4
Mar 19 09:33:55 ipc1220 ed-lte-tool[1541]: [2026-03-19 09:33:55][INFO] RUN: AT+CFUN=1
Mar 19 09:33:55 ipc1220 ed-lte-tool[1541]: Sending AT command: AT+CFUN=1
Mar 19 09:33:57 ipc1220 ed-lte-tool[1541]: [2026-03-19 09:33:57][INFO] RUN: reset 4G
Mar 19 09:33:57 ipc1220 ed-lte-tool[1541]: [2026-03-19 09:33:57][INFO] Resetting None, Some("4G_RST"), None
Mar 19 09:33:57 ipc1220 ed-lte-tool[1541]: [2026-03-19 09:33:57][INFO] Using GPIO name: 4G_RST
Mar 19 09:33:58 ipc1220 ed-lte-tool[1541]: [2026-03-19 09:33:58][INFO] LTE reset successful via GPIO gpiochip2: line 3 (initial: LOW)
Mar 19 09:33:58 ipc1220 ed-lte-tool[1541]: [2026-03-19 09:33:58][ERROR] LTE module not ready.
Mar 19 09:35:29 ipc1220 ed-lte-tool[1541]: [2026-03-19 09:35:29][INFO] RUN: AT+CFUN=4
Mar 19 09:35:29 ipc1220 ed-lte-tool[1541]: Sending AT command: AT+CFUN=4
```

- If the information at the marked position in the returned message shows "Active: active (running)", it indicates that the 4G status is normal.
- If the information at the marked position in the returned message shows "Active: inactive (dead)", it indicates that the 4G status is abnormal.

5.7.3 Basic Configuration Commands

If the 4G network cannot connect, you can use the following commands for querying and configuration.

Command	Description
ifconfig	<p>Check if the 4G network is connected. The usb0 interface represents the 4G interface, as shown below.</p> <pre>pi@raspberrypi:~\$ ifconfig eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500 inet 192.168.1.1 netmask 255.255.255.0 broadcast 192.168.1.255 inet6 fe80::732e:31bc:aed6:4177 prefixlen 64 scopeid 0x20<link> ether 00:a0:9a:1b:05:24 txqueuelen 1000 (Ethernet) RX packets 0 bytes 0 (0.0 B) RX errors 0 dropped 0 overruns 0 frame 0 TX packets 0 bytes 0 (0.0 B) TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0 lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536 inet 127.0.0.1 netmask 255.0.0.0 inet6 ::1 prefixlen 128 scopeid 0x10<host> loop txqueuelen 1000 (Local Loopback) RX packets 0 bytes 0 (0.0 B) RX errors 0 dropped 0 overruns 0 frame 0 TX packets 0 bytes 0 (0.0 B) TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0 usb0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500 inet 10.191.34.79 netmask 255.0.0.0 broadcast 10.255.255.255 inet6 fe80::c704:40cc:227f:72b6 prefixlen 64 scopeid 0x20<link> ether 9c:3c:29:a3:0b:0d txqueuelen 1000 (Ethernet) RX packets 0 bytes 0 (0.0 B) RX errors 0 dropped 0 overruns 0 frame 0 TX packets 0 bytes 0 (0.0 B) TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0 wlan0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500 ether 8c:1f:64:34:a0:04 txqueuelen 1000 (Ethernet) RX packets 0 bytes 0 (0.0 B) RX errors 0 dropped 0 overruns 0 frame 0 TX packets 0 bytes 0 (0.0 B) TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0</pre> <ul style="list-style-type: none"> • If the usb0 interface shows a specific IP address, it indicates 4G is connected. • If the usb0 interface does not show a specific IP address, it indicates 4G is not connected.
sudo systemctl status ed-lte-daemon.service	<p>Query the status of the 4G service, as shown below.</p> <pre>pi@ipc1220:~\$ sudo systemctl status ed-lte-daemon.service ● ed-lte-daemon.service - EDITEC 4G Reconnect service Loaded: loaded (/usr/lib/systemd/system/ed-lte-daemon.service; enabled; preset: enabled) Active: active (running) since Thu 2026-03-19 09:18:32 GMT; 16min ago Invocation: 8d292f488c45421caac480d27e369c94 Main PID: 1541 (ed-lte-tool) Tasks: 3 (limit: 359) CPU: 17.191s CGroup: /system.slice/ed-lte-daemon.service └─1541 /usr/local/bin/ed-lte-tool --daemon Mar 19 09:33:53 ipc1220 ed-lte-tool[1541]: Sending AT command: AT+CFUN=4 Mar 19 09:33:55 ipc1220 ed-lte-tool[1541]: [2026-03-19 09:33:55][INFO] RUN: AT+CFUN=1 Mar 19 09:33:55 ipc1220 ed-lte-tool[1541]: Sending AT command: AT+CFUN=1 Mar 19 09:33:57 ipc1220 ed-lte-tool[1541]: [2026-03-19 09:33:57][INFO] RUN: reset 4G Mar 19 09:33:57 ipc1220 ed-lte-tool[1541]: [2026-03-19 09:33:57][INFO] Resetting None, Some("4G_RST"), None Mar 19 09:33:57 ipc1220 ed-lte-tool[1541]: [2026-03-19 09:33:57][INFO] Using GPIO name: 4G_RST Mar 19 09:33:58 ipc1220 ed-lte-tool[1541]: [2026-03-19 09:33:58][INFO] LTE reset successful via GPIO gpiochip2: line 3 (initial: LOW) Mar 19 09:33:58 ipc1220 ed-lte-tool[1541]: [2026-03-19 09:33:58][ERROR] LTE module not ready. Mar 19 09:35:29 ipc1220 ed-lte-tool[1541]: [2026-03-19 09:35:29][INFO] RUN: AT+CFUN=4 Mar 19 09:35:29 ipc1220 ed-lte-tool[1541]: Sending AT command: AT+CFUN=4</pre> <ul style="list-style-type: none"> • If the information at the marked location shows "Active: active (running)", it indicates the 4G status is normal. • If the information at the marked location shows "Active: inactive (dead)", it indicates the 4G status is abnormal.
sudo systemctl enable ed-lte-daemon.service	<p>Enable the 4G service</p>
sudo systemctl start ed-lte-daemon.service	<p>Start the 4G service</p>
	<p>Stop the 4G service</p>

Command	Description
sudo systemctl stop ed-lte- daemon.service	
sudo ed-lte-tool -r	Reset the 4G module

TIP

If the 4G port connection is normal but the 4G service status query shows an abnormality, you can enable and then start the 4G service in sequence.

5.8 Configuring RTC

Configuring RTC

5.9 Configuring Serial Port

Describes the configuration methods for RS485 and RS232.

TIP

Serial port communication supports baud rates from 2400 to 115200.

5.9.1 Installing PicoCom Tool

In a Linux environment, you can use the picocom tool to debug the RS232 and RS485 serial ports.

Execute the following command to install the picocom tool.

```
sudo apt-get install picocom
```

sh

5.9.2 Configuring RS485

The ED-IPC1220 includes two RS485 interfaces. Their corresponding COM ports and device files are as follows:

RS485 Port Number	Corresponding COM Port	Corresponding Device File
1	RS485-1	/dev/com1
2	RS485-2	/dev/com2

Preparation:

The connection between the ED-IPC1220's RS485 port and the external device is completed.

Steps:

1. (Optional) Disable the serial port login mode.

TIP

- The RS485-1 (GND/A1/B1) port on the ED-IPC1220 device defaults to serial port login mode. If it needs to be configured as a regular serial port, please disable the serial console login mode first.
- The RS485-2 (GND/A2/B2) port defaults to a normal serial port with the serial port login mode disabled, so this step can be skipped.

- a. Open a command pane and execute the following command to open the configuration tool interface.

```
sudo raspi-config
```

sh

```
Raspberry Pi Software Configuration Tool (raspi-config)
1 System Options      Configure system settings
2 Display Options     Configure display settings
3 Interface Options   Configure connections to peripherals
4 Performance Options Configure performance settings
5 Localisation Options Configure language and regional settings
6 Advanced Options    Configure advanced settings
8 Update              Update this tool to the latest version
9 About raspi-config  Information about this configuration tool

<Select>              <Finish>
```

- b. Select "Interface Options" from the menu, then press "Enter".

```
Raspberry Pi Software Configuration Tool (raspi-config)
1 System Options      Configure system settings
2 Display Options     Configure display settings
3 Interface Options   Configure connections to peripherals
4 Performance Options Configure performance settings
5 Localisation Options Configure language and regional settings
6 Advanced Options    Configure advanced settings
8 Update              Update this tool to the latest version
9 About raspi-config  Information about this configuration tool

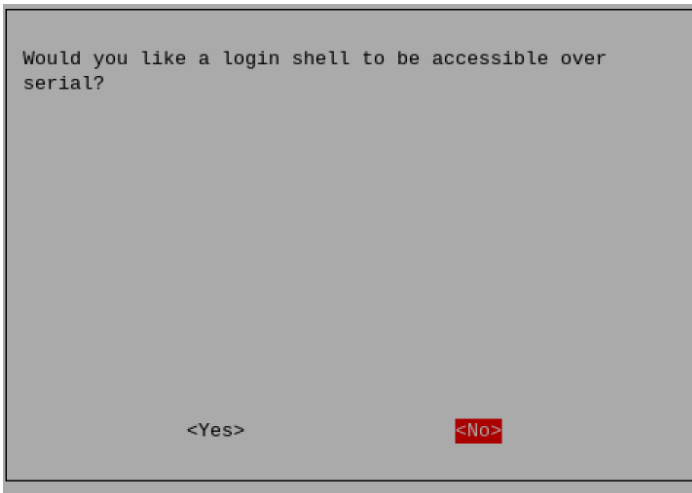
<Select>              <Finish>
```

- c. Select "Serial Port" from the menu, then press "Enter".

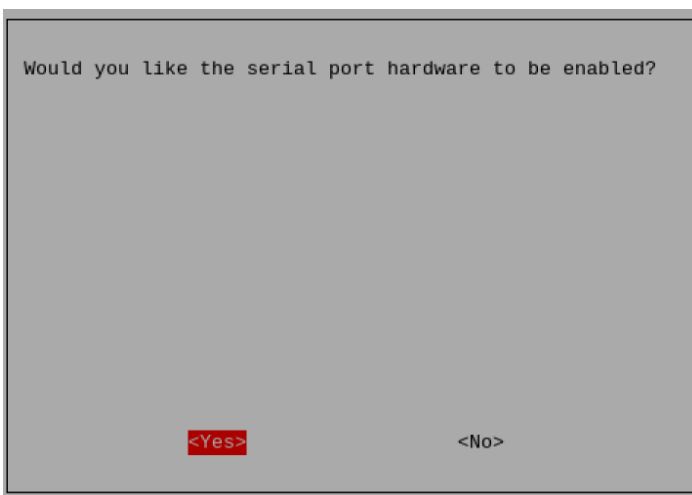
```
Raspberry Pi Software Configuration Tool (raspi-config)
I1 SSH                Enable/disable remote command line access using SSH
I2 RPi Connect        Enable/disable Raspberry Pi Connect
I3 VNC                Enable/disable graphical remote desktop access
I4 SPI                Enable/disable automatic loading of SPI kernel module
I5 I2C                Enable/disable automatic loading of I2C kernel module
I6 Serial Port        Enable/disable shell messages on the serial connection
I7 1-Wire              Enable/disable one-wire interface

<Select>              <Back>
```

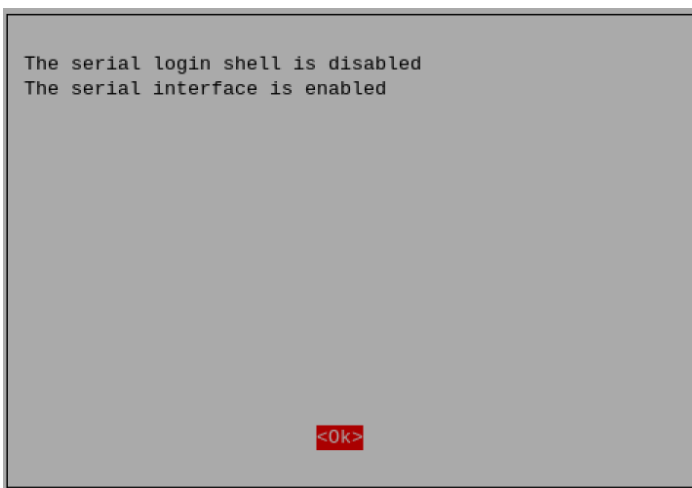
- d. Select "No" in the prompt box, then press "Enter".



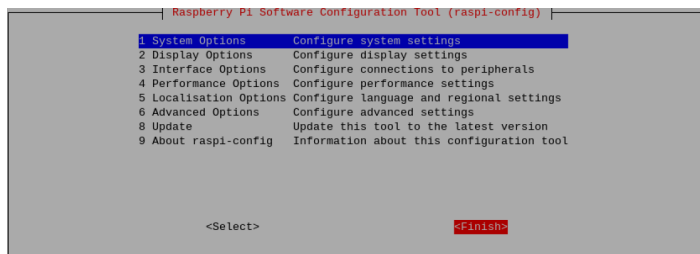
e. Select "Yes" in the prompt box, then press "Enter".



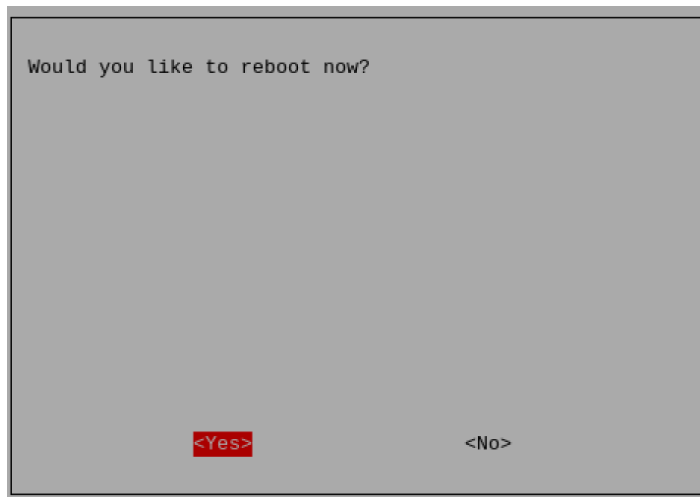
f. Press "Enter" to confirm the settings.



g. Select "Finish" in the configuration tool interface, then press "Enter".



h. Select "Yes" in the prompt box, then press "Enter" to restart the device.



TIP

If the RS485-1 (GND/A1/B1) port needs to be restored to serial console login mode after being disabled, you can reconfigure it in the `raspi-config` tool.

2. Open the serial port and configure the baud rate.

- Execute the following command to open the RS485-1 port and configure the serial baud rate to 115200.

```
picocom -b 115200 /dev/com1
```

sh

- Execute the following command to open the RS485-2 port and configure the serial baud rate to 115200.

```
sudo picocom -b 115200 /dev/com2
```

sh

3. Enter commands as needed to control the external device.

5.9.3 Configuring RS232

The ED-IPC1220 includes one RS232 interface. Its corresponding COM port and device file are as follows:

RS232 Port Number	Corresponding COM Port	Corresponding Device File
1	RS232	/dev/com3

Preparation:

The connection between the ED-IPC1220's RS232 port and the external device is completed.

Steps:

1. Execute the following command to open the RS232 serial port and configure the serial baud rate to 115200.

```
sudo picocom -b 115200 /dev/com3
```

sh

2. Enter commands as needed to control the external device.

5.10 Configuring DI

The ED-IPC1220 includes 8 DI interfaces. DI0~DI3 are native GPIO ports of the CM0, and DI4~DI7 are extended GPIO ports. The corresponding GPIO ports and names for each DI interface are listed in the table below. Users can configure them according to actual needs.

DI Interface	Corresponding GPIO of CM0	Name
DI0	GPIO17	-
DI1	GPIO18	-
DI2	GPIO19	-
DI3	GPIO20	-
DI4	-	DI4
DI5	-	DI5
DI6	-	DI6
DI7	-	DI7

Preparation:

The connection between the ED-IPC1220's DI interface and the external sensor is completed.

Steps:

- If the interface is DI0~DI3, execute the following command to obtain the interface data, where X represents the GPIO number corresponding to the DI interface, e.g., 17.

```
pinctrl get X
```

sh

A low level (lo) in the returned result indicates an active state.

- If the interface is DI4~DI7, execute the following command to obtain the interface data, where Y represents the name corresponding to the DI interface, e.g., DI4.

```
sudo ed-gpio get Y
```

sh

A low level (0) in the returned result indicates an active state.

5.11 Configuring DO

The ED-IPC1220 includes 8 DO interfaces. DO0~DO3 are native GPIO ports of the CM0, and DO4~DO7 are extended GPIO ports. The corresponding GPIO ports and names for each DO interface are listed in the table below. Users can configure them according to actual needs.

DO Interface	Corresponding GPIO of CM0	Name
DO0	GPIO12	-
DO1	GPIO21	-
DO2	GPIO22	-
DO3	GPIO23	-
DO4	-	DO4
DO5	-	DO5
DO6	-	DO6
DO7	-	DO7

Preparation:

The connection between the ED-IPC1220's DO interface and the external load is completed.

Steps:

- If the interface is DO0~DO3, execute the following commands to set the output to high or low level, where X represents the GPIO number corresponding to the DO interface, e.g., 12.

Set to high level:

```
pinctrl set X op dh
```

sh

Set to low level:

```
pinctrl set X op dl
```

sh

- If the interface is DO4~DO7, execute the following commands to set the output, where Y represents the name corresponding to the DO interface, e.g., DO4.

Set to high level:

```
sudo ed-gpio set Y 1
```

sh

Set to low level:

```
sudo ed-gpio set Y 0
```

sh

5.12 Configuring USER Indicator

The ED-IPC1220 includes one USER LED. The LED is a dual-color (red/green) LED supporting mixed colors and is controlled via extended I/O. The corresponding extended I/O names are listed in the table below:

USER LED	GPIO Name
Red LED	USER_LED_RED
Green LED	USER_LED_GED

- Query the LED status:

```
sudo ed-gpio get USER_LED_RED
sudo ed-gpio get USER_LED_GED
```

sh

- Turn on the red LED:

```
sudo ed-gpio set USER_LED_RED 1
```

sh

- Turn off the red LED:

```
sudo ed-gpio set USER_LED_RED 0
```

sh

- Turn on the green LED:

```
sudo ed-gpio set USER_LED_GED 1
```

sh

- Turn off the green LED:

```
sudo ed-gpio set USER_LED_GED 0
```

sh

TIP

The USER LED supports mixing red and green to produce orange.

5.13 Configuring USER Button

The ED-IPC1220 includes one USER button, controlled via GPIO. The corresponding GPIO pin is listed in the table below:

USER Button	Corresponding GPIO of CM0
USER	GPIO25

By default, GPIO25 is at a high level. Pressing the button sets GPIO25 to a low level. You can query the current state using the following command.

```
pinctrl get 25
```

sh