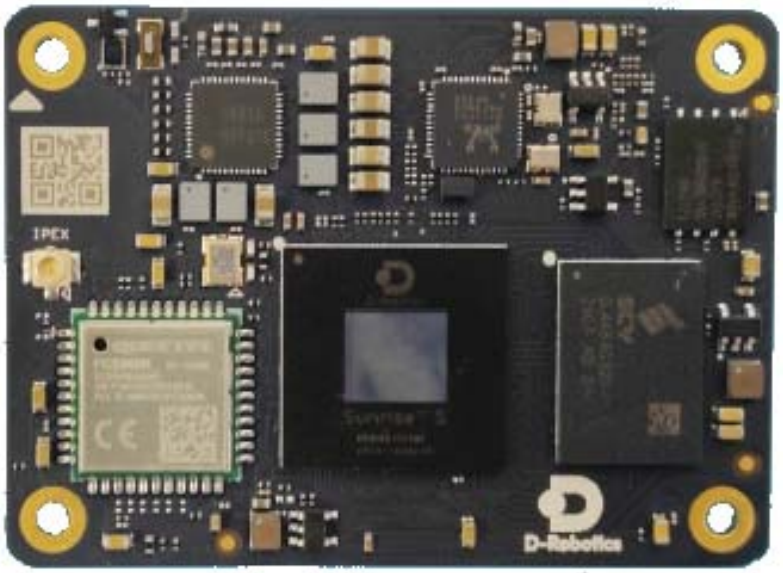
RDK X5H Module User manual

Core Module Interface

All on-board design, no core module.

The RDK X5H Module carrier board provides a set of 300pin board-to-board connectors for core module installation. During installation, the correct direction and positioning must first be confirmed to avoid damage to the core module and carrier board connectors.



The module installation method is as follows:

1. Check the pin orientation of the core module to confirm the correct installation direction.
2. Place the core module directly above the carrier board and ensure the four positioning holes around are aligned.
3. Press down from the center of the core module. When the module clicks, it indicates proper installation.

Power Interface

The development board provides one USB Type C interface (interface 1) as the power supply interface, requiring a power adapter that supports **5V/5A** to power the development board. After connecting the power adapter to the development board, the **green power indicator light turns on**, indicating normal power supply to the development board. After version 3.1.0, the **orange status indicator light flashes**, indicating normal system operation.

The development board provides one USB Type C interface (interface 1) as the power supply interface, requiring a power adapter that supports **5V/5A** to power the development board. After connecting the power adapter to the development board, the **green 5V indicator PWR indicator light turns on**, indicating

normal power supply to the development board, and the **green ACT indicator light flashes**, indicating normal system operation.

Do not use the computer's USB interface to power the development board, otherwise, insufficient power supply may cause the development board to experience **unexpected power loss and repeated restarts**.

Debug Serial Port{#debug_uart}

The development board provides one debug serial port (interface 4) to achieve serial port login and debugging functions. The parameter configuration of the computer's serial port tool is as follows:

- Baud rate: 115200
- Data bits: 8
- Parity: None
- Stop bits: 1
- Flow Control: None

For serial port connection, connect the development board interface 4 to the PC using a Micro-USB cable.

The development board provides one debug serial port (rear interface 22) to achieve serial port login and debugging functions. The parameter configuration of the computer's serial port tool is as follows:

- Baud rate: 921600
- Data bits: 8
- Parity: None
- Stop bits: 1
- Flow Control: None

For serial port connection, connect the development board interface 22 to the PC using a Micro-USB cable.

Generally, when users use this interface for the first time, they need to install the CH340 driver on the computer. Users can search for the keyword `CH340 serial port driver` to download and install it.

Wired Network Port

The development board provides one Gigabit Ethernet interface (interface 6), supporting 1000BASE-T and 100BASE-T standards, with static IP mode enabled by default, and the IP address is `192.168.127.10`. To confirm the development board's IP address, you can log in to the device via the serial port and use the `ifconfig` command to view the `eth0` network port configuration.

Additionally, this interface supports PoE (Power over Ethernet) functionality, allowing simultaneous data and power transmission through the network cable without the need for additional power cables, making device installation more convenient and flexible.

HDMI Display Interface{#hdmi_interface}

The development board provides one HDMI (interface 10) display interface, supporting up to 1080P resolution. The development board outputs the Ubuntu system desktop (Ubuntu Server version displays logo icon) on the monitor through the HDMI interface. Additionally, the HDMI interface supports real-time display of camera and network stream images.

The development board provides one HDMI (interface 21) display interface, supporting up to 1080P resolution. The development board outputs the Ubuntu system desktop (Ubuntu Server version displays logo icon) on the monitor through the HDMI interface. Additionally, the HDMI interface supports real-time display of camera and network stream images.

USB Display Interface

The development board achieves multi-channel USB interface expansion through hardware circuits to meet user demands for connecting multiple USB devices. The interface descriptions are as follows:

Interface Type	Interface Number	Quantity	Description
USB 2.0 Type C	Interface 3	1 channel	USB Device mode, used to connect to the host to implement ADB, Fastboot, system burning, and other functions
USB 3.0 Type A	Interface 7	4 channels	USB Host mode, expanding into 4 USB ports through HUB, used to connect USB 3.0 peripherals

The development board achieves multi-channel USB interface expansion through hardware circuits to meet user demands for connecting multiple USB devices. The interface descriptions are as follows:

Interface Type	Interface Number	Quantity	Description
USB 2.0 Type C	Interface 3	1 channel	USB Device mode, used to connect to the host to implement ADB, Fastboot, system burning, and other functions
USB 3.0 Type A	Interface 4 & Interface 5	4 channels	USB Host mode, expanding into 4 USB ports through HUB, used to connect USB 3.0 peripherals

USB 2.0 Switch to HOST

The development board can short-circuit interface 2 to switch USB 2.0 to HOST mode.

Connecting a USB Drive

The development board's USB Type A interface supports USB drive functionality, capable of automatically detecting and mounting connected USB drives. The default mount directory is `/media/sda1`.

Connecting a USB to Serial Adapter Board

The development board's USB Type A interface supports USB to serial adapter board functionality, capable of automatically detecting the connection and creating device nodes `/dev/ttyUSB*` or `/dev/ttyACM*` (the asterisk represents a number starting from 0). Users can refer to the Using Serial Ports chapter for usage instructions.

USB Camera

The development board's USB Type A interface supports USB camera functionality, capable of automatically detecting connected USB cameras and creating device node `/dev/video0`.

MIPI Camera Interface{#mipi_port}

The development board provides two 22pin MIPI CSI interfaces (interface 5), enabling the connection of two MIPI cameras and supporting dual-camera input. Currently, the development board is compatible with various camera modules. The module models and specifications are as follows:

No.	Sensor	Resolution	FOV	I2C Device Address
1	IMX219	800W		
2	OV5647	500W		
3	IMX477	1230W		

The camera module connects to the development board via a 22pin same-direction flat cable. Insert the metal side of the cable facing away from the black latch into the connector.

After installation, users can confirm if the module's I2C address can be detected normally using the i2cdetect command.

To check the I2C device address of the Camera Sensor on the mipi_host0 interface near the network port:

```
echo 353 > /sys/class/gpio/export
echo out > /sys/class/gpio/gpio353/direction
echo 0 > /sys/class/gpio/gpio353/value
sleep 0.1
echo 1 > /sys/class/gpio/gpio353/value

i2cdetect -y -r 6
```Query the I2C device address of the Camera Sensor on the mipi_host2
interface away from the network port:
```shell
echo 351 > /sys/class/gpio/export
echo out > /sys/class/gpio/gpio351/direction
echo 0 > /sys/class/gpio/gpio351/value
sleep 0.1
echo 1 > /sys/class/gpio/gpio351/value

i2cdetect -y -r 4
```

When the I2C device address of the Camera Sensor is successfully detected, you can see the printout as shown below (taking the detection of IMX219 on the mipi_host2 interface as an example, you can find that the address 10 is printed out):

```
root@ubuntu:~# i2cdetect -y -r 4
     0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:                         -- -- -- -- -- -- -- --
10: 10 -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
20: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
30: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
```

```
40: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
50: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
60: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
70: -- -- -- -- -- -- -- --
```

The development board provides two 22 - pin MIPI CSI interfaces CAM1 (interface 10) and CAM2 (interface 9), which can realize the access of two MIPI cameras and support the access of binocular cameras. Currently, the development board has adapted to a variety of camera modules, and the module models and specifications are as follows:

No.	Sensor	Resolution	FOV	I2C Device Address
1	IMX219	800W		
2	OV5647	500W		
3	IMX477	1230W		

The camera module is connected to the development board through a 22 - pin same - direction flat cable. The metal surface of the flat cable is inserted into the connector with the black latch facing back.

After the installation is completed, users can confirm whether the I2C address of the module can be normally detected through the i2cdetect command.

Query the I2C device address of the Camera Sensor on the CAM1 (interface 10):

```
echo 353 > /sys/class/gpio/export
echo out > /sys/class/gpio/gpio353/direction
echo 0 > /sys/class/gpio/gpio353/value
sleep 0.1
echo 1 > /sys/class/gpio/gpio353/value

i2cdetect -y -r 6
```

Query the I2C device address of the Camera Sensor on the CAM2 (interface 9):

```
echo 351 > /sys/class/gpio/export
echo out > /sys/class/gpio/gpio351/direction
echo 0 > /sys/class/gpio/gpio351/value
sleep 0.1
echo 1 > /sys/class/gpio/gpio351/value

i2cdetect -y -r 4
```

When the I2C device address of the Camera Sensor is successfully detected, you can see the printout as shown below (taking the detection of IMX219 on CAM2 (interface 9) as an example, you can find that the address 10 is printed out):

```
root@ubuntu:~# i2cdetect -y -r 4
     0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:                         -- -- -- -- -- -- -- --
10: 10 -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
20: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
30: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
40: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
50: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
60: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
70: -- -- -- -- -- -- -- --
```

Important Note: It is strictly forbidden to plug or unplug the camera while the development board is not powered off, otherwise the camera module will be easily burned out.

LCD Display Interface

RDK X5 provides a MIPI DSI LCD display interface (interface 14), which can be used for the access of LCD displays. The interface is 22 - pin and can be directly connected to many Raspberry Pi LCD displays using the DSI - Cable - 12cm cable.

RDK X5 Module provides a MIPI DSI LCD display interface (interface 19), which can be used for the access of LCD displays. The interface is 22 - pin and can be directly connected to many Raspberry Pi LCD displays using the DSI - Cable - 12cm cable.

Micro SD Interface

The development board provides one Micro SD card interface (interface 13). It is recommended to use a storage card with a capacity of at least 16GB to meet the installation requirements of the Ubuntu operating system and related function packages.

The development board provides one Micro SD card interface (back interface 22). It is recommended to use a storage card with a capacity of at least 16GB to meet the installation requirements of the Ubuntu operating system and related function packages.

It is prohibited to hot - plug the TF storage card during the use of the development board, otherwise it will cause abnormal operation of the system and even damage the file system of the storage card.

CANFD Interface

The RDK X5H development board provides a CANFD interface (interface 8) and a CAN terminal resistor access switch (the 2 - pin socket behind interface 8. For high - speed communication, it is necessary to enable the terminal resistors at both ends to prevent signal reflection and improve anti - interference ability), which can be used for CAN and CAN FD communication. For more information, please refer to the CAN Usage chapter.

The RDK X5H Module development board provides a CANFD interface (interface 15) and a CAN terminal resistor access switch (interface 14. For high - speed communication, it is necessary to enable the terminal resistors at both ends to prevent signal reflection and improve anti - interference ability), which can be used for CAN and CAN FD communication. For more information, please refer to the CAN Usage chapter.

40PIN Interface

The RDK X5 development board provides a 40PIN interface, and the IO signal adopts a 3.3 V level design. The pin definition is compatible with products such as Raspberry Pi. For detailed pin definitions and multiplexing relationships, please refer to the hardware development chapter.

---

sidebar_position: 2

1.2.2 RDK X5H

```
import Tabs from '@theme/Tabs';
import TabItem from '@theme/TabItem';
```

Before using the RDK X5H development board, you need to do the following preparations.

Burning Preparation

Power Supply

The RDK X5H development board is powered through the USB Type C interface, and a power adapter that supports **5V/5A** is required to power the development board.

The RDK X5H Module is powered through the USB Type C interface on the carrier board, and a power adapter that supports **5V/5A** is required to power the development board.

Do not use the computer's USB interface to power the development board, otherwise it will cause abnormal situations such as **abnormal power - off and repeated restarts** of the development board due to insufficient power supply.

For handling more problems, you can refer to the Common Problems chapter.

Storage

The RDK X5H development board uses a Micro SD card as the system boot medium. It is recommended to use a storage card with a capacity of at least 16GB to meet the storage space requirements of the Ubuntu

system and application software.

The RDK X5H Module core board comes with eMMC, and there is an SD card slot on the carrier board, supporting system boot in both eMMC and SD card modes.

The system boots from eMMC by default. The boot mode can be configured via srpi-config. For details, please refer to srpi-config Advanced Options Boot Order.

When using a Micro SD card as the system boot medium, it is recommended to use a storage card with a capacity of at least 16GB to meet the storage space requirements of the Ubuntu system and application software.

Display

The RDK X5H development board supports HDMI display interface. Connect the development board to a monitor via an HDMI cable to enable graphical desktop display.

The RDK X5H Module development board supports HDMI display interface. Connect the development board to a monitor via an HDMI cable to enable graphical desktop display.

Network Connection

The RDK X5H development board supports two types of network interfaces: Ethernet and Wi-Fi. Users can achieve network connectivity through either interface.

The RDK X5H Module development board supports two types of network interfaces: Ethernet and Wi-Fi. Users can achieve network connectivity through either interface.

System Image Flashing

The RDK kit currently provides Ubuntu 22.04 system images, available in two versions: desktop and server;

desktop: An Ubuntu system with a desktop environment, allowing connection of external monitors and mouse operation;

server: An Ubuntu system without a desktop environment, allowing remote operation via serial port or network.

Image Download {#img_download}

Click **Download Image** to enter the version selection page, choose the latest version directory, and proceed to the system download page.

After downloading, extract the Ubuntu system image file, such as `ubuntu-preinstalled-desktop-arm64.img`.

1. The verification process using balenaEtcher may report errors, but the system can still boot normally;
2. It is highly recommended by the official team to use the Rufus flashing tool for image installation, as it is relatively more stable;

The **RDK X5H Module** comes pre-installed with a test version system image. To ensure the use of the latest system version, it is recommended to follow this document to flash the latest version system image.

Click **Download Image** to enter the version selection page, choose the latest version directory, and proceed to the system download page.

After downloading, extract the Ubuntu system image file, such as `ubuntu-preinstalled-desktop-arm64.img`.

1. The verification process using balenaEtcher may report errors, but the system can still boot normally;
2. It is highly recommended by the official team to use the Rufus flashing tool for image installation, as it is relatively more stable;
3. The RDK X5H Module can only use system versions 3.2.0 and later;

System Image Flashing

Before flashing the Ubuntu system image, the following preparations are needed:

- Prepare a Micro SD card with a capacity of at least 16GB
- SD card reader
- Download the image flashing tool Rufus (can be downloaded here)

Rufus is a bootable drive creation tool that supports the Windows platform. The steps to create an SD boot drive using Rufus are as follows:

1. Open the Rufus tool and select the corresponding Micro SD card as the target device from the "Device" dropdown menu.
2. Click the "Select" button to choose the extracted `rdk-x5-ubuntu-preinstalled-desktop-arm64.img` file as the flashing image.
3. Leave other parameters at their default settings, click the "Start" button, and wait for the flashing process to complete. Once completed, you can close Rufus and remove the storage card.

On-board Flashing

Insert the SD card into the development board, connect the USB 2.0 Type-C to the PC, press and hold the Sleep button (next to the headphone jack), power on the development board, wait for 5 seconds, and the development board will enter flashing mode.

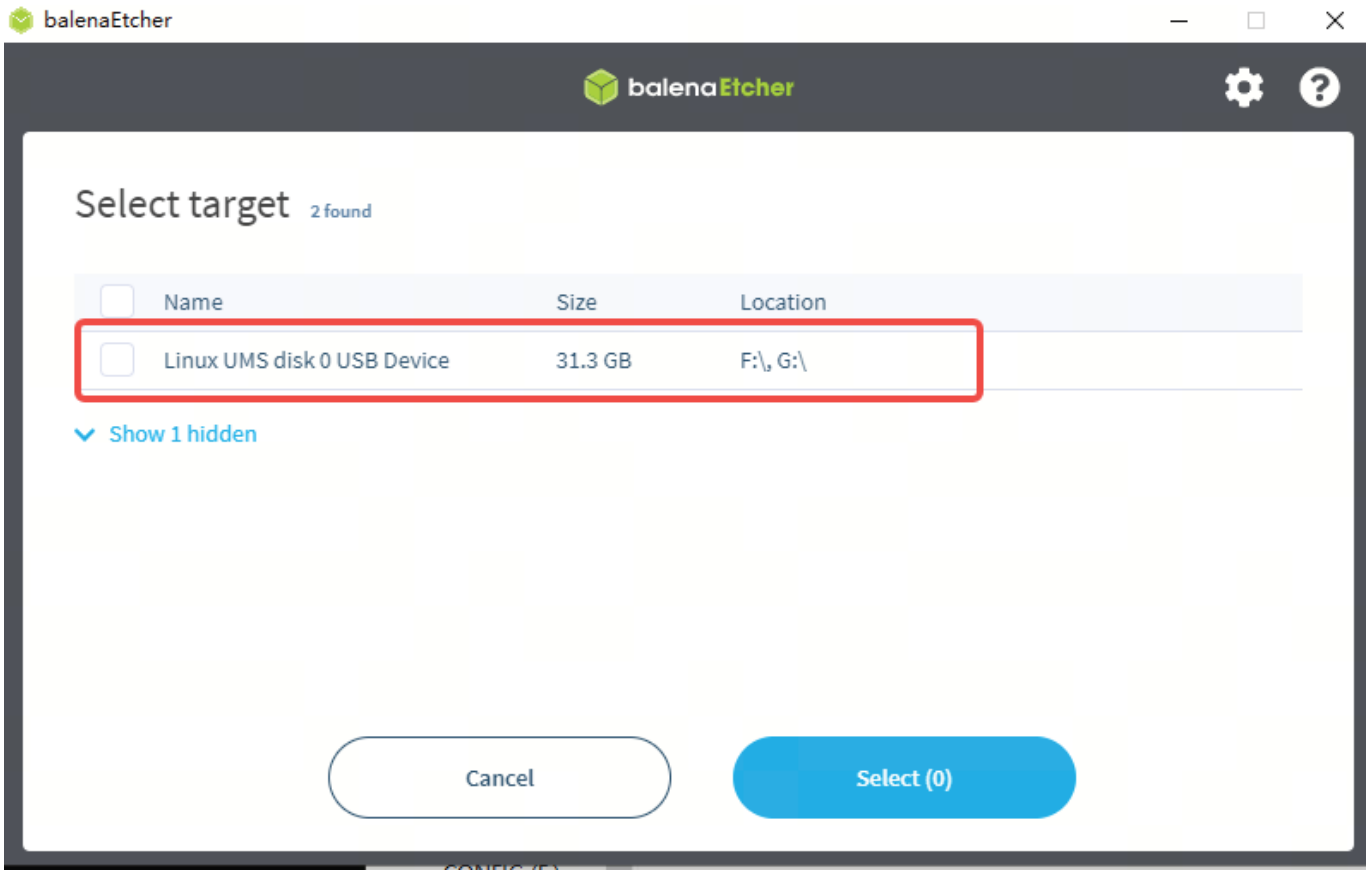
The PC can detect the SD card mapped as a USB drive, then follow the system flashing section to complete the system flashing.

Insert the SD card into the development board, connect the USB 2.0 Type-C to the PC, press and hold the Sleep button (interface 23), power on the development board, wait for 5 seconds, and the development board will enter flashing mode.

The PC can detect the SD card mapped as a USB drive;

If the SD card is not inserted into the development board, the eMMC built into the core board will be mapped as a USB drive;

Then follow the system flashing section to complete the system flashing.



System Boot

First, ensure the development board is powered off. Then insert the prepared storage card into the Micro SD card slot of the development board and connect the development board to a monitor via an HDMI cable.

Finally, power on the development board.

During the first system boot, default environment configuration will be performed, which takes about 45 seconds. After the configuration is completed, the Ubuntu system desktop will be displayed on the monitor.

- **Green** indicator light: Lit indicates normal hardware power-on.

If the development board does not display output for a long time (more than 2 minutes) after powering on, it indicates abnormal startup. Debugging via a serial port cable is required to check if the development board is functioning properly.

After the Ubuntu Desktop version system completes booting, it will output the system desktop on the monitor via the HDMI interface, as shown in the figure below:

Common Issues

Common issues when using the development board for the first time are as follows:

- **Power on but no boot**: Please ensure using the adapter recommended in the Power Supply section; ensure the Micro SD card of the development board has been flashed with the Ubuntu system image.
- **Hot-plugging the storage card during use**: The development board does not support hot-plugging the Micro SD storage card. If a misoperation occurs, please restart the development board.

Precautions

- Do not plug or unplug any devices while powered on except for USB, HDMI, and network cables.
- The Type-C USB interface of the RDK X5 is only used for power supply.
- Use a reputable brand USB Type-C power cable; otherwise, power supply abnormalities may occur, leading to unexpected system shutdown issues.

For handling more issues, you can refer to the Common Issues chapter, and also visit the D-Robotics Developer Official Forum for assistance.

---

sidebar_position: 3 ---# 1.3 Getting Started Configuration

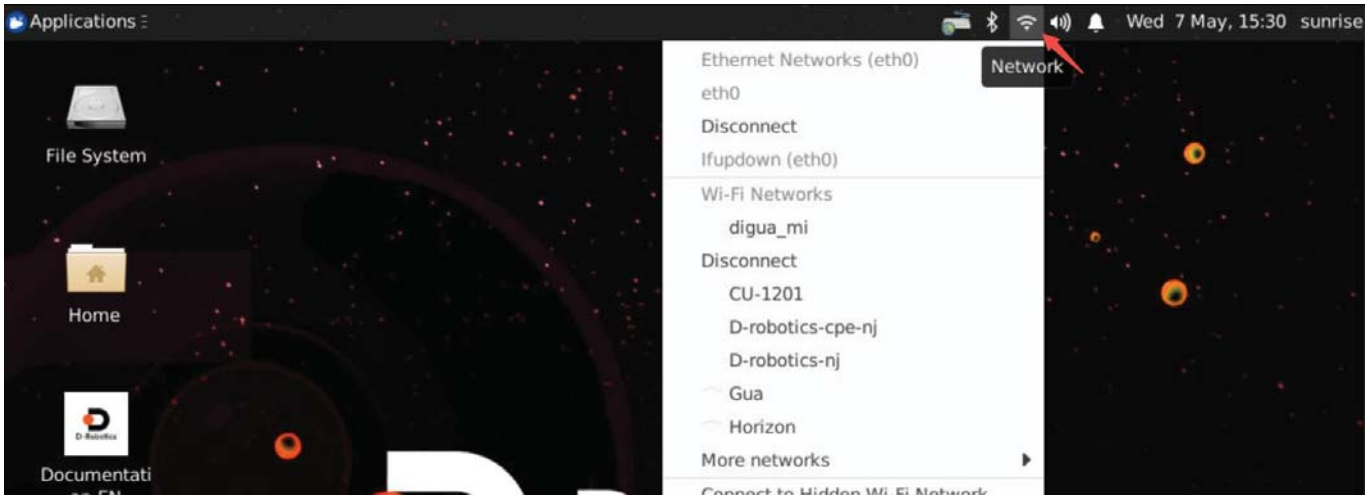
The introductory configuration method introduced in this chapter is only supported on the RDK X3, RDK X5, and RDK X3 Module models of development boards;

The system version is no less than `2.1.0`.

```
import Tabs from '@theme/Tabs';
import TabItem from '@theme/TabItem';
```

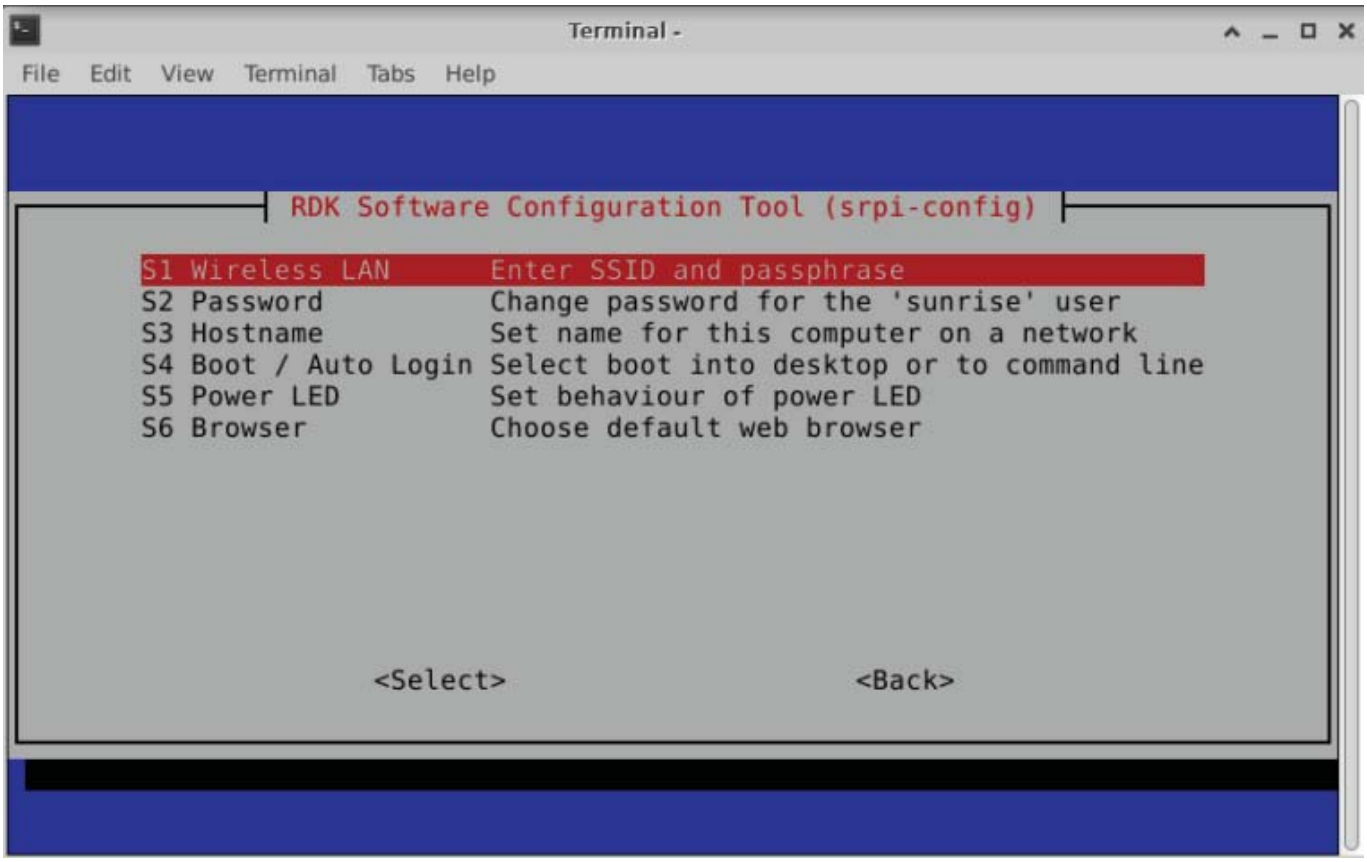
Connect to Wi-Fi

Use the Wi-Fi management tool in the upper right corner of the menu bar to connect to Wi-Fi. As shown in the figure below, click on the Wi-Fi name you want to connect to, then enter the Wi-Fi password in the pop-up dialog box.



Use the srpi-config tool to connect to Wi-Fi.

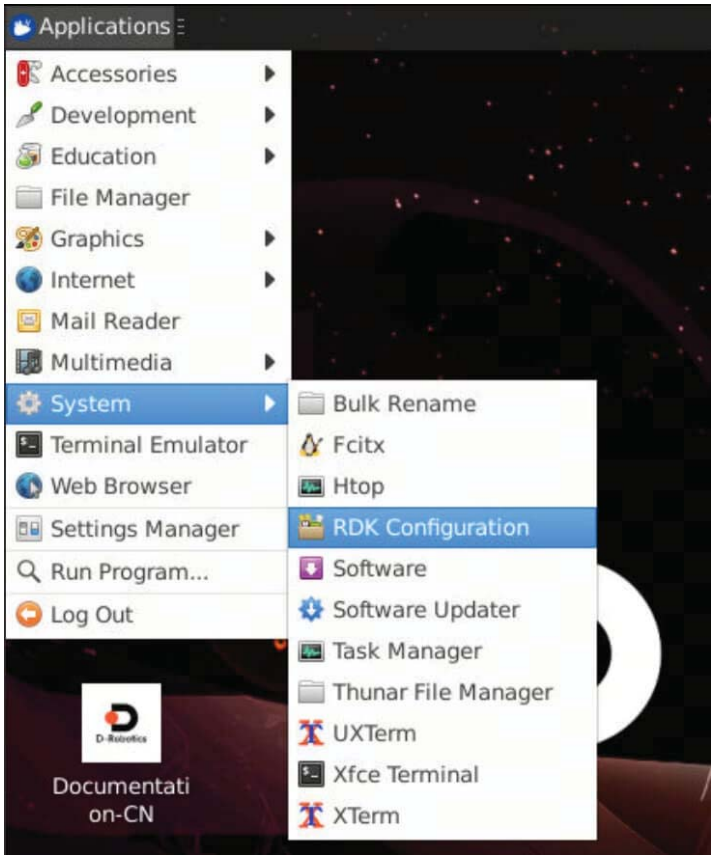
Execute the `sudo srpi-config` command, select System Options -> Wireless LAN, and input the Wi-Fi name (`SSID`) and password (`passwd`) according to the prompts.



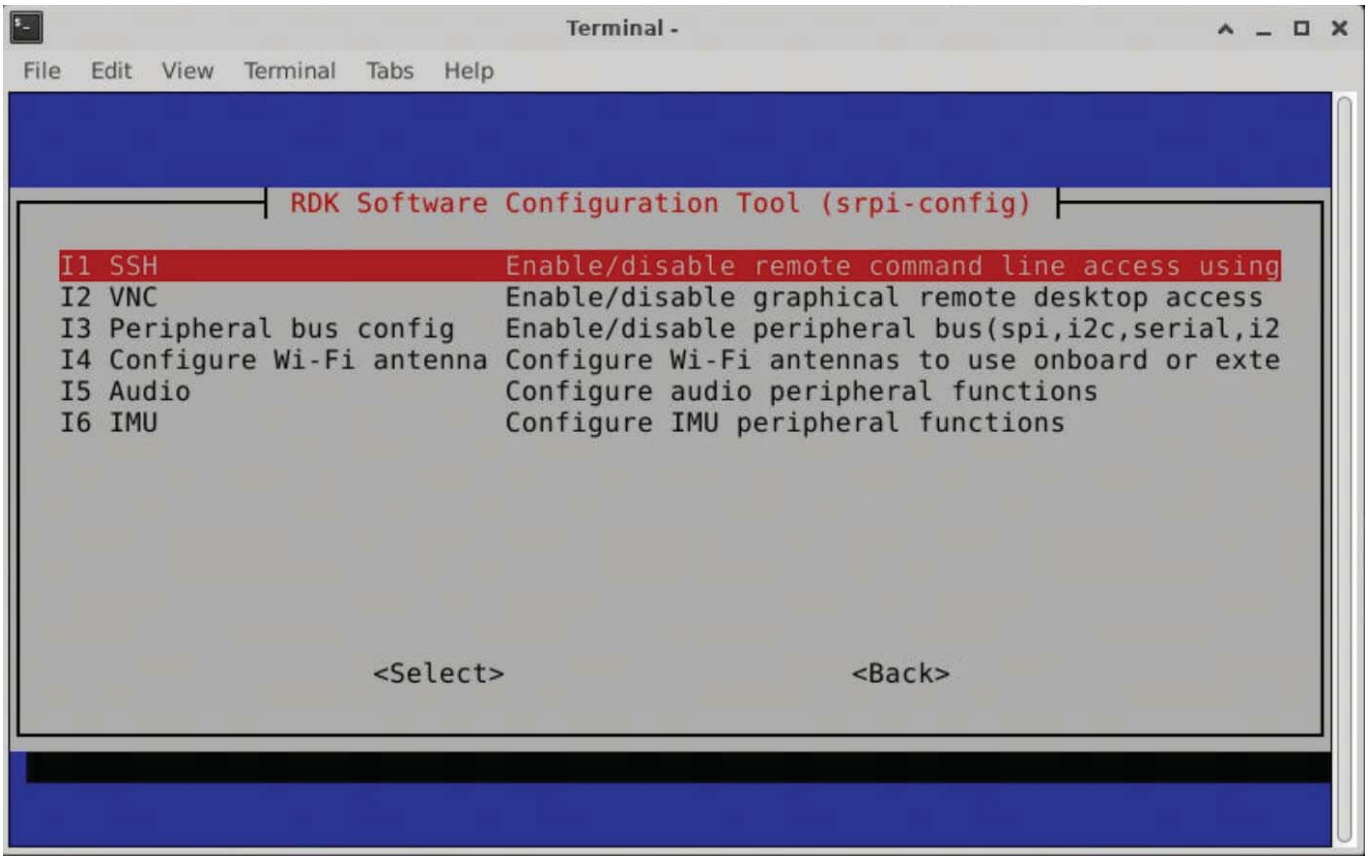
Enable SSH Service

The current system version enables SSH login service by default. Users can use this method to turn the SSH service on or off.

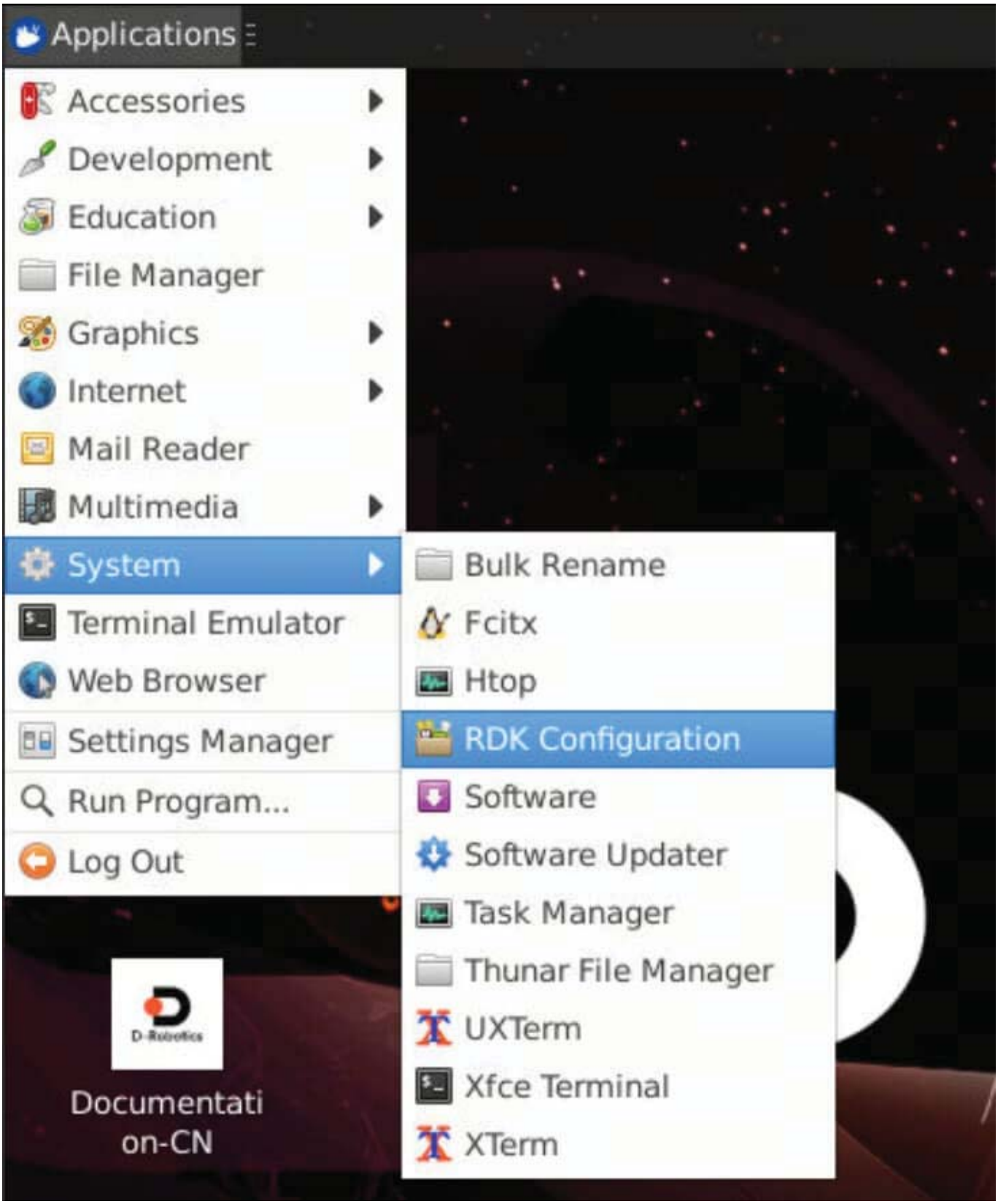
Find the `RDK Configuration` item through the menu bar and click to open.



Select Interface Options -> SSH item, and enable or disable the SSH service according to the prompts



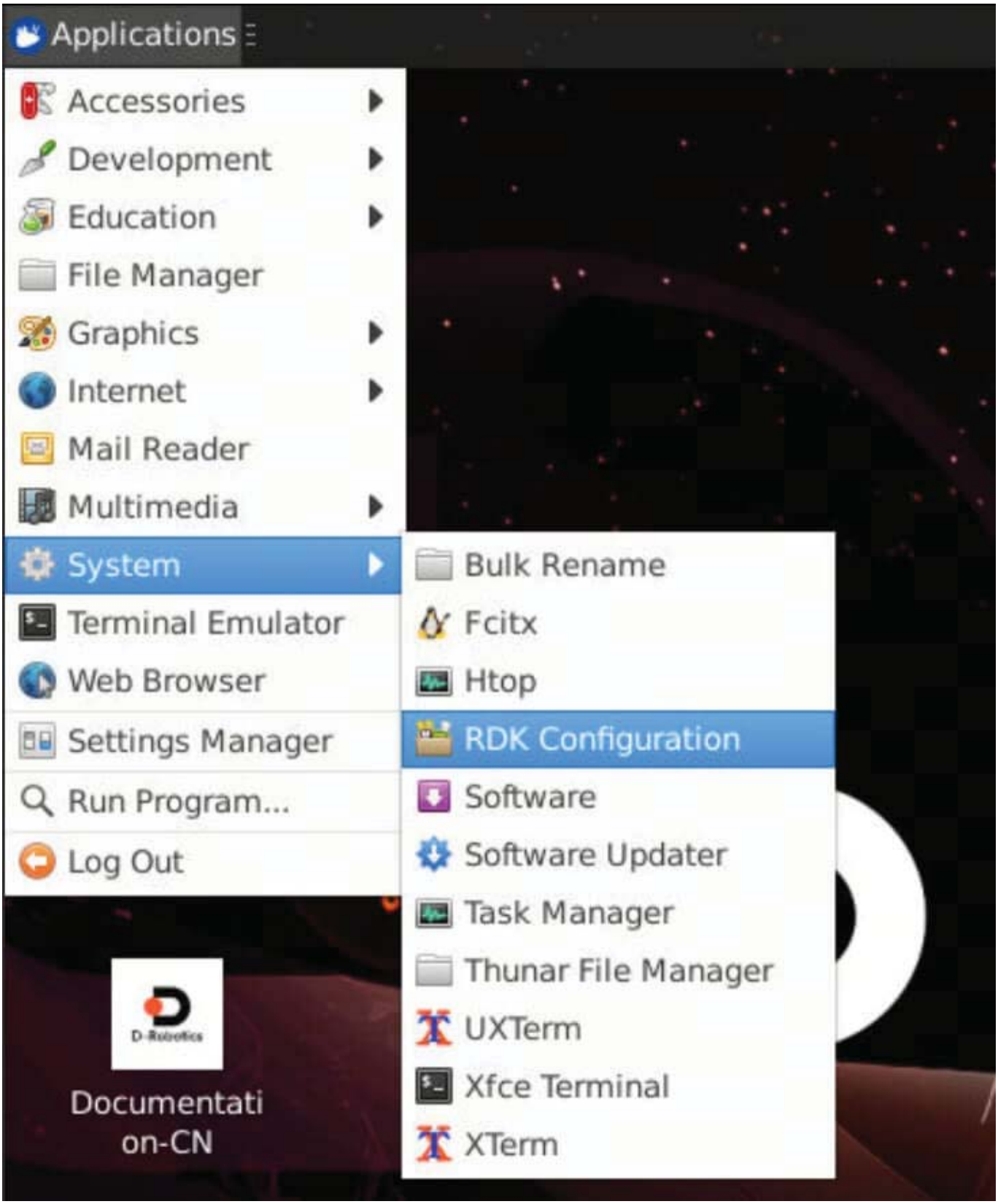
Execute the `sudo srpi-config` command to enter the configuration menu. Select Interface Options -> SSH item, and enable or disable the SSH service according to the prompts.



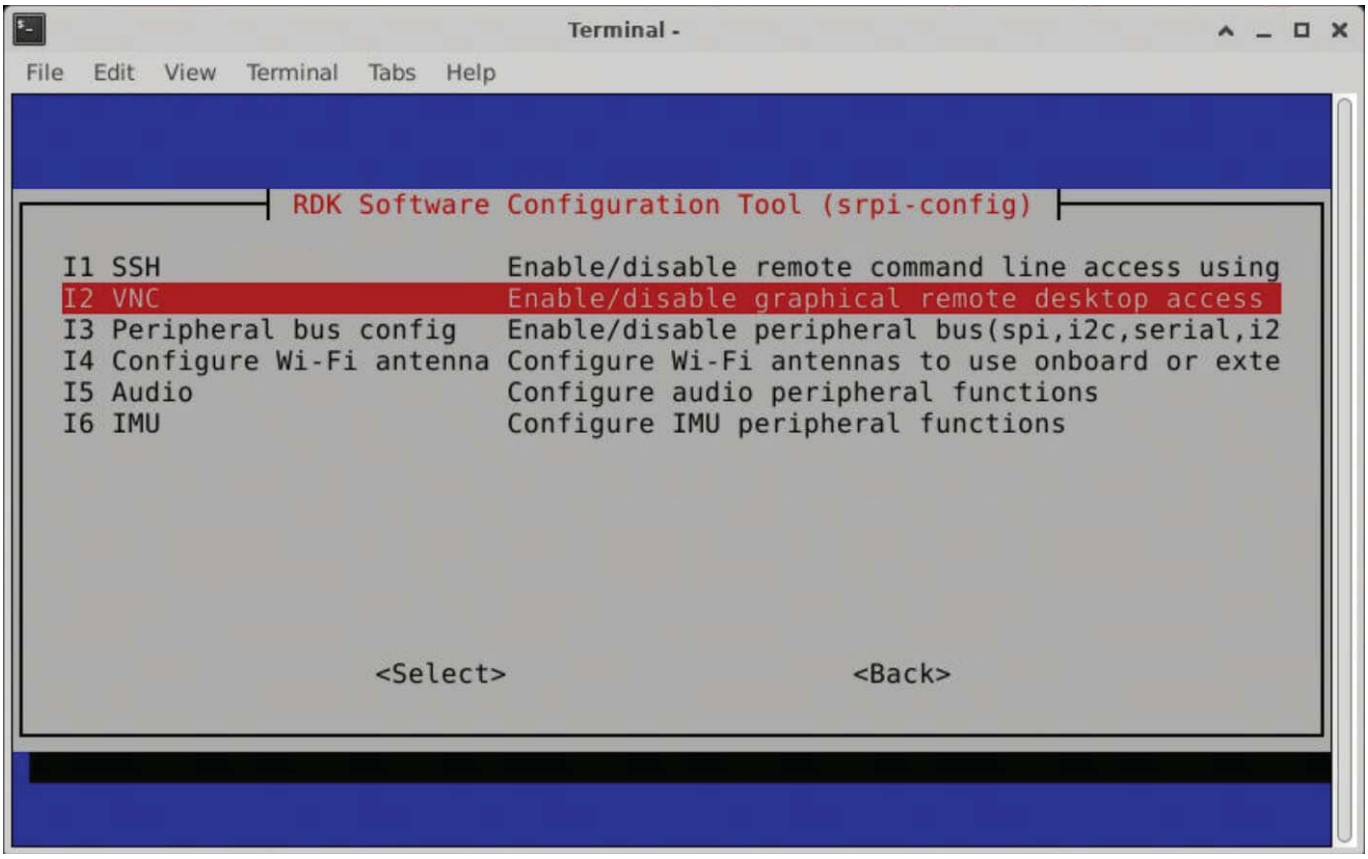
For the usage of SSH, please check Remote Login - SSH Login.

Enable VNC Service

Find the `RDK Configuration` item through the menu bar and click to open.



Select Interface Options -> VNC item, and enable or disable the VNC service according to the prompts. When enabling VNC, a login password needs to be set, which must be an 8-character string composed of characters.



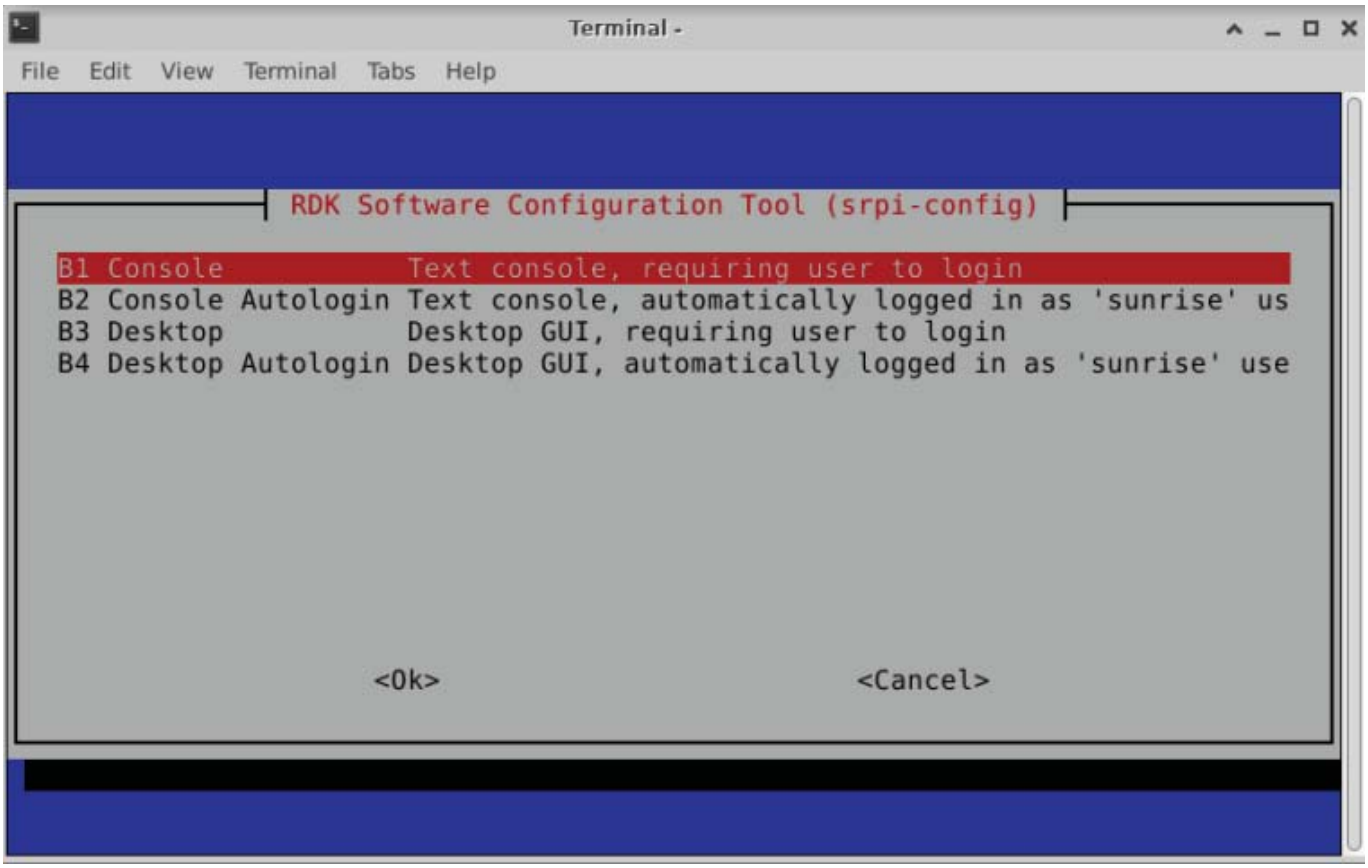
For the usage of VNC, please check Remote Login - VNC Login.

Set Login Mode

The desktop graphical system supports four login modes:

1. Enable graphical interface, automatic login
2. Enable graphical interface, user manual login
3. Character terminal, automatic login
4. Character terminal, user manual login

Find the `RDK Configuration` item through the menu bar and click to open. Select System Options -> Boot / Auto Login item to enter the following configuration items. Select the corresponding item according to your needs.



It takes effect after reboot.

Character terminal, supports two login modes:

1. Character terminal, automatic login
2. Character terminal, user manual login

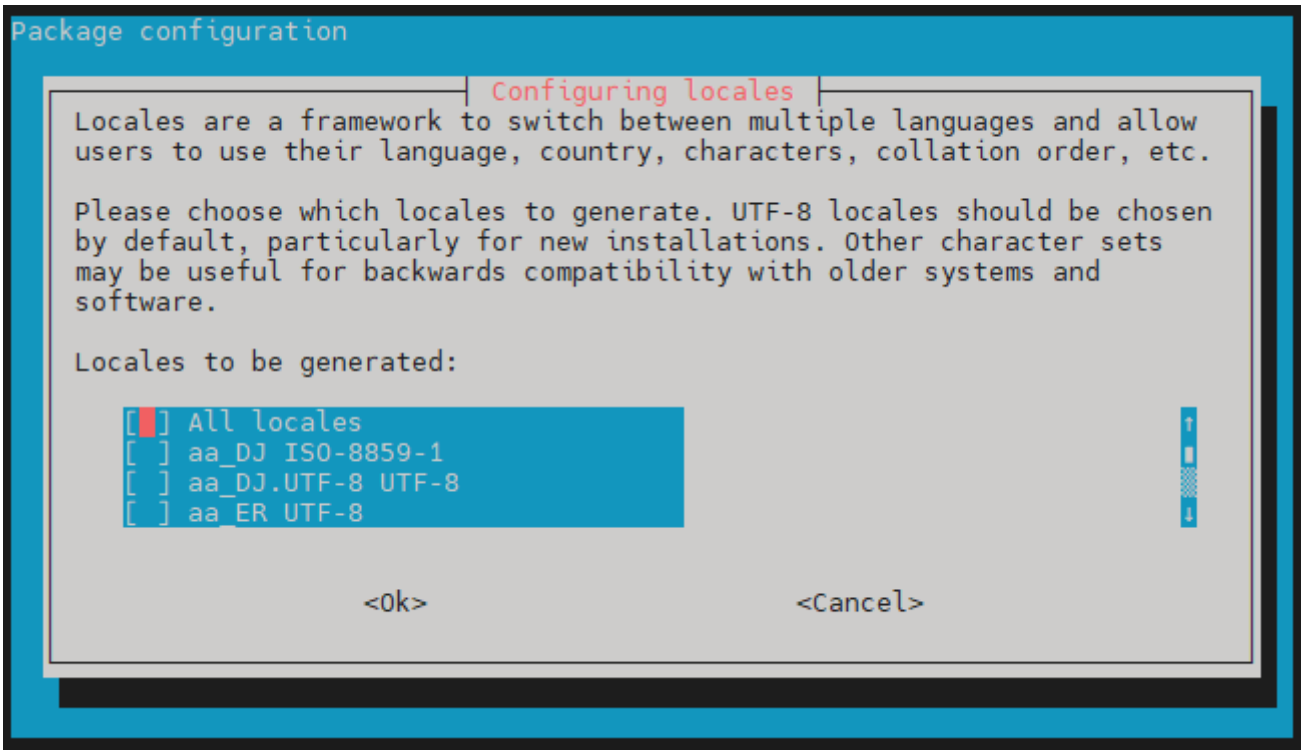
Execute the `sudo srpi-config` command to enter the configuration menu. Select System Options -> Boot / Auto Login item to enter the following configuration items. Select the corresponding item according to your needs.

It takes effect after reboot.

Set Chinese Environment

Find the `RDK Configuration` item through the menu bar and click to open. Select Localisation Options -> Locale item to enter the following configuration.

Step 1: Select the language environment that will be used (multiple choices), generally select `en_US.UTF-8 UTF-8` and `zh_CN.UTF-8 UTF-8`. Press Enter to confirm and proceed to the next step.



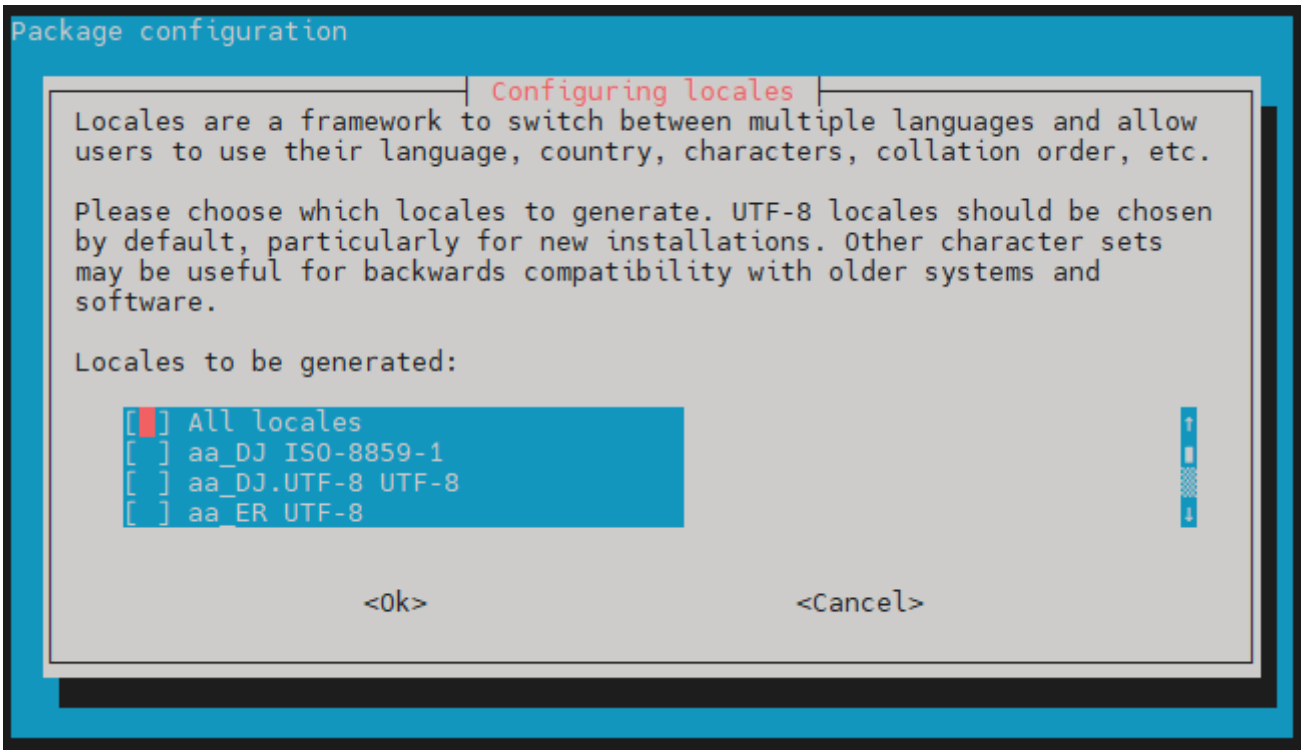
Step 2: Select the default language environment, choose `zh_CN.UTF-8 UTF-8` for the Chinese environment. After pressing Enter to confirm, wait for a while to complete the configuration.

Step 3: Restart the development board to make the latest configuration take effect. `sudo reboot`

Upon booting, it will prompt whether to update the names of several commonly used folders under the home directory. It is recommended to choose `Don't ask me again` `Keep Old Name`, so that the directory names such as `Desktop` `Documents` `Downloads` under the user's working directory do not change with the language environment.

Execute the `sudo srpi-config` command to enter the configuration menu. Select Localisation Options -> Locale item to enter the following configuration.

Step 1: Select the language environment that will be used (multiple choices), generally select `en_US.UTF-8 UTF-8` and `zh_CN.UTF-8 UTF-8`. Press Enter to confirm and proceed to the next step.



Step 2: Select the default language environment, choose `zh_CN.UTF-8 UTF-8` for the Chinese environment. After pressing Enter to confirm, wait for a while to complete the configuration.

Step 3: Restart the development board to make the latest configuration take effect. `sudo reboot`

Set Chinese Input Method

Step 1: Find the EN input method icon at the desktop end, right-click and select Preferences

Step 2: Click on input method -> add on the right side -> select Chinese

Step 3: Select smart pinyin, finally the upper right corner EN can right-click to select smart pinyin

Set RDK Studio

RDK Studio provides rich features and convenience for RDK users, including device management, quick start of Demo, quick access to community forums, etc. Next, we will introduce how to manage and use your RDX uniformly.

Step 1: Download RDK Studio (link: Download Link), after clicking download, the page will scroll to the bottom download position, downloading User Installer and ZIP are both fine, follow the installation steps offline.

Step 2: After opening Studio, the interface contains four left-side menus as follows:

(1) `Device Management`: Here, you can add devices for management via the `+RDK Device` button in the upper right corner

(Demonstrated here using a local area network IP connection), for wired connection methods, see Bilibili (Video Link), for flash connection methods, see the Tip section later in this chapter.

(2) `Example Applications`: Here, you can directly install some simple Demos onto your development board.

```(3) `Community`: This provides direct access to the Digua Robot Community, no need to open a webpage for checking.

(4) `NodeHub`: This provides direct access to NodeHub, offering a wealth of encapsulated example nodes.

(5) `Burning`: Please refer to Chapter 1.2 System Burning.
```

Step Three: Usage of Studio Integrated Tools

(1) `Terminal Usage`: Clicking the terminal button will automatically pop up the Windows terminal, and entering the password will automatically connect.

(2) `Vscode Usage`: Clicking the Vscode icon will automatically invoke the local Vscode Remote plugin for connection (PS: You must have Vscode and its plugin installed locally).

(3) `Other Functions`: Other functions like Jupyter that require installation can be installed by users as needed.

The above operations are universally applicable to various systems. For quick connection operation, note that it is only available for RDX X5H 's Type C interface.

Specific usage method is as follows:

Step One: Confirmation of Development Board Network

Taking X5H's 3.0 version image as an example (do not use Beta version images), the IP segment corresponding to the Type C network card is `192.168.128.10`. (PS: For other versions, you can check using `ifconfig`).

Step Two: Setting Up Personal PC Network

Open the control panel on your Windows computer, find Network and Internet -> Network and Sharing Center -> Change adapter settings on the left.

Find the Ethernet of the board (PS: Unplug and replug the connection line between the board and the computer multiple times to identify which one is the development board’s Ethernet) -> Right-click and select properties, fill in according to the figure below.

Step Three: Quick Connection Operation

Open the device management section of RDK Studio, click Add RDK Device at the upper right corner -> Select the quick connection option -> Select the network (PS: Choose the board’s network from the previous step) -> Select user -> Connect to the WIFI you want to configure for the board -> Finally add remarks information.

Note: Since connecting to WIFI takes time, it may show that WIFI is not found when the device addition is completed, wait a moment and refresh the card.

Currently, RDK Studio for Windows system has been officially released. Users of Linux and Mac should wait a little while our developers are typing furiously.

Currently, RDK Studio for Windows system has been officially released. Users of Linux and Mac should wait a little while our developers are typing furiously.

User Management

Modify Username

Taking the new username "usertest" as an example:

```
# Close all processes of sunrise user
sudo pkill -u sunrise
# Rename sunrise user to usertest
sudo usermod -l usertest sunrise
# Change the user's home directory to /home/usertest
sudo usermod -d /home/usertest -m sunrise
# Modify user password
sudo passwd usertest
```

Finally, change `autologin-user=sunrise` to `autologin-user=usertest` in the `/etc/lightdm/lightdm.conf.d/22-hobot-autologin.conf` file to update the automatic login username.

Add New User

Taking adding a new user "usertest" as an example:

```
sudo useradd -U -m -d /home/usertest -k /etc/skel/ -s /bin/bash -G
disk,kmem,dialout,sudo,audio,video,render,i2c,lightdm,vpu,gdm,weston-
launch,graphics,jpu,ipu,vps,misc,gpio usertest
sudo passwd usertest
sudo cp -aRf /etc/skel/. /home/usertest
sudo chown -R usertest:usertest /home/usertest
```

You can also refer to modifying the username, setting the new user as the automatic login user.

---

sidebar_position: 4

1.4 Remote Login

This chapter aims to introduce users who need to remotely access the development board through their personal computers (PC) on how to log in remotely via serial port and network (VNC, SSH).

Before logging in remotely via the network, the development board needs to be connected to the network through a wired Ethernet or wireless WiFi, and the IP address of the development board should be configured. For IP address information under the two connection methods, please refer to the following description:

- Wired Ethernet: The development board uses static IP mode by default, with the IP address `192.168.127.10`, subnet mask `255.255.255.0`, gateway `192.168.127.1`
- Wireless WiFi: The development board's IP address is usually assigned by the router, and can be checked using the `ifconfig` command in the device command line to view the wlan0 network's IP address.

Serial Port Login {#login_uart}

Video: https://www.bilibili.com/video/BV1rm4y1E73q/?p=2

Before using the serial port login, ensure that the development board's serial cable is correctly connected to the computer. Refer to the corresponding development board's debugging serial port chapter for the connection method:

- rdk_ultra Debugging Serial Port Chapter
- rdk_x3 Debugging Serial Port Chapter
- rdk_x5 Debugging Serial Port Chapter

Serial port login requires the assistance of a PC terminal tool. Commonly used tools include `Putty`, `MobaXterm`, etc. Users can choose according to their own usage habits. The configuration process for different tools' ports is similar. Below, we take `MobaXterm` as an example to introduce the process of creating a new serial port connection:

- When the USB-to-serial adapter board is inserted into the computer for the first time, the serial port driver needs to be installed. The driver program can be obtained from the Tools sub-section of the Resource Center. After the driver is installed, the device manager can normally recognize the serial port board port, as shown in the figure below:

- Open the `MobaXterm` tool, click `Session`, then select `Serial`.
- Configure the port number, such as `COM3`, the actual port number used is subject to the port number recognized by the PC.
- Set the serial port configuration parameters as follows:

Configuration Item	Parameter Value
Baud Rate	RDK X3 (921600), RDK X5H(115200)
Data Bits	8
Parity	None
Stop Bits	1
Flow Control	None

Click `OK`, input username: `root`, password: `root` to log in to the device

At this point, you can use the `ifconfig` command to query the development board's IP address, where eth0 and wlan0 represent wired and wireless networks respectively:

```
root@ubuntu:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.127.10  netmask 255.255.255.0  broadcast 192.168.1.255
        inet6 fe80::211:22ff:feaa:7637  prefixlen 64  scopeid 0x20<link>
        ether 00:11:22:aa:76:37  txqueuelen 1000  (Ethernet)
        RX packets 767  bytes 54006 (54.0 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 5766  bytes 246466 (246.4 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
        device interrupt 43  base 0xa000

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 3847  bytes 339115 (339.1 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 3847  bytes 339115 (339.1 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

wlan0: flags=4099<UP,BROADCAST,MULTICAST>  mtu 1500
        ether 08:e9:f6:ae:f8:8a  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
```

Network Status Confirmation {#network_config}

Video: https://www.bilibili.com/video/BV1rm4y1E73q/?p=3

Before using remote login, ensure that the communication between the computer and the development board network is normal. If it cannot `ping` through, follow these steps to confirm:

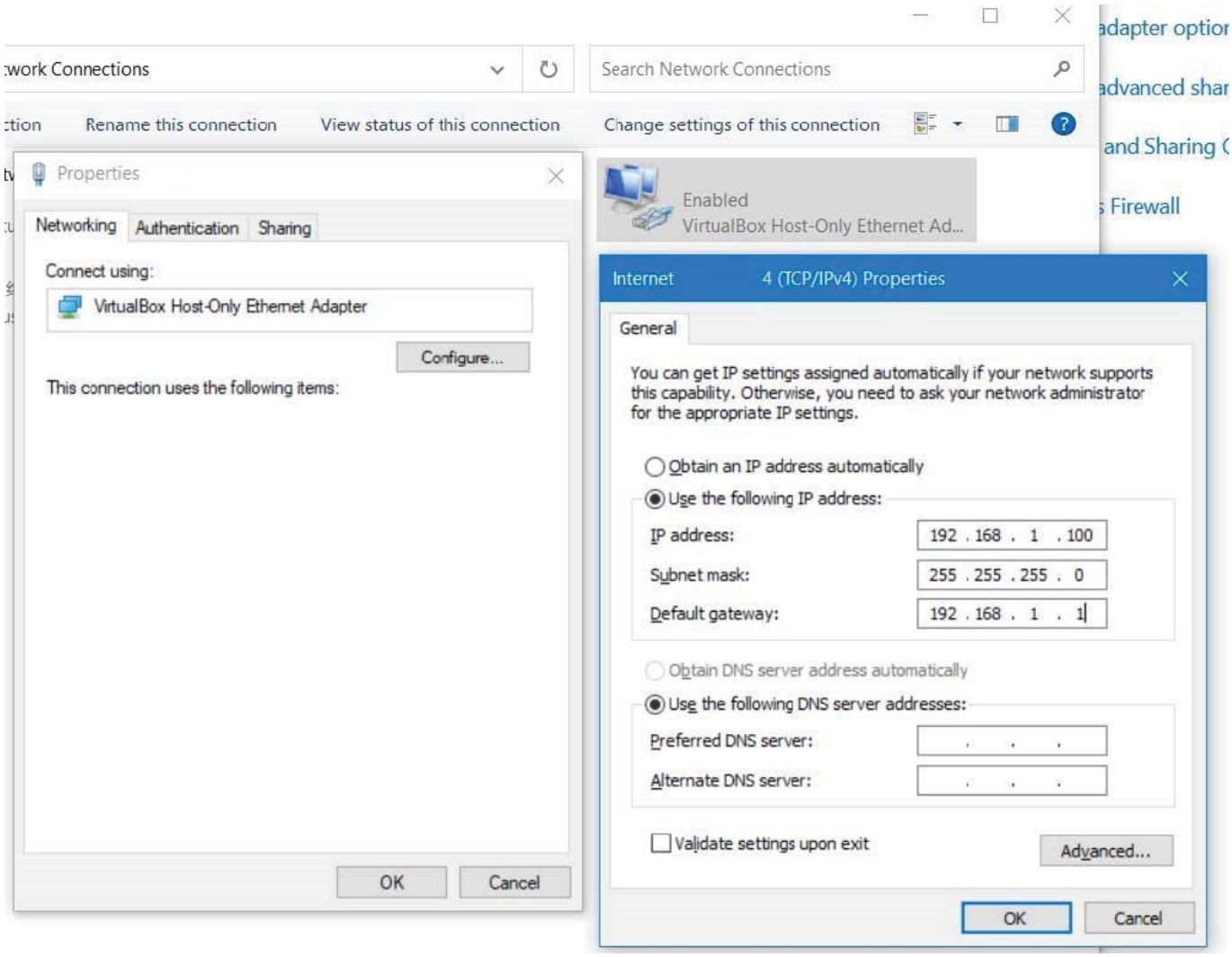
The IP addresses corresponding to each version of the image are:

| Board Series | Image Version | Ethernet IP Address | | ---------- | ---------- | ---------- || X3 | Less than or equal to 2.0.0 | 192.168.1.10/24 | | X3 | Greater than or equal to 2.1.0 | 192.168.127.10/24 | | X5 | 3.0.0 | 192.168.127.10/24 | | **Board Series** | **Image Version** | **Flash Connection Port (USB Device) IP Address** | | X5 | 3.0.0 | 192.168.128.10/24 |

> Confirm the IP address configuration of the development board and computer. Generally, the first three segments need to be the same, for example, development board: `192.168.127.10` computer: `192.168.127.100`
> Confirm whether the subnet mask and gateway configuration of the development board and computer are consistent
> Confirm whether the computer network firewall is turned off

The wired Ethernet of the development board uses static IP mode by default, with the IP address being `192.168.127.10`. In the case of direct connection between the development board and the computer, you only need to configure the computer with a static IP to ensure it is in the same network segment as the development board. Taking the WIN10 system as an example, the method to modify the static IP of the computer is as follows:

> Find the corresponding Ethernet device in the network connection and double-click to open it
> Find and double-click to open the Internet Protocol Version 4 option
>
> Enter the corresponding network parameters in the red box position in the figure below, and click OK



If you need to configure the wired network of the development board to dynamic DHCP mode, you can refer to the Wired Network chapter for configuration.

VNC Login

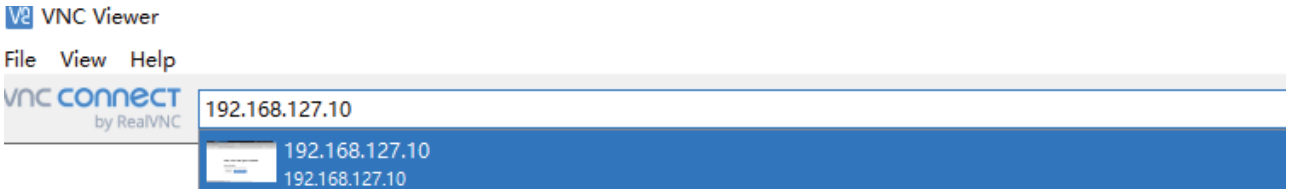
Video: https://www.bilibili.com/video/BV1rm4y1E73q/?p=4

This chapter is for users using the Ubuntu Desktop system version, introducing how to achieve remote desktop login through `VNC Viewer`. `VNC Viewer` is a graphical desktop sharing software that allows remote login and desktop control on a computer. This software can preview the development board system desktop on the computer display and use the computer's mouse and keyboard for remote operation. Users operating through VNC Viewer can achieve the same effect as local operations on the development board. Download link VNC Viewer.

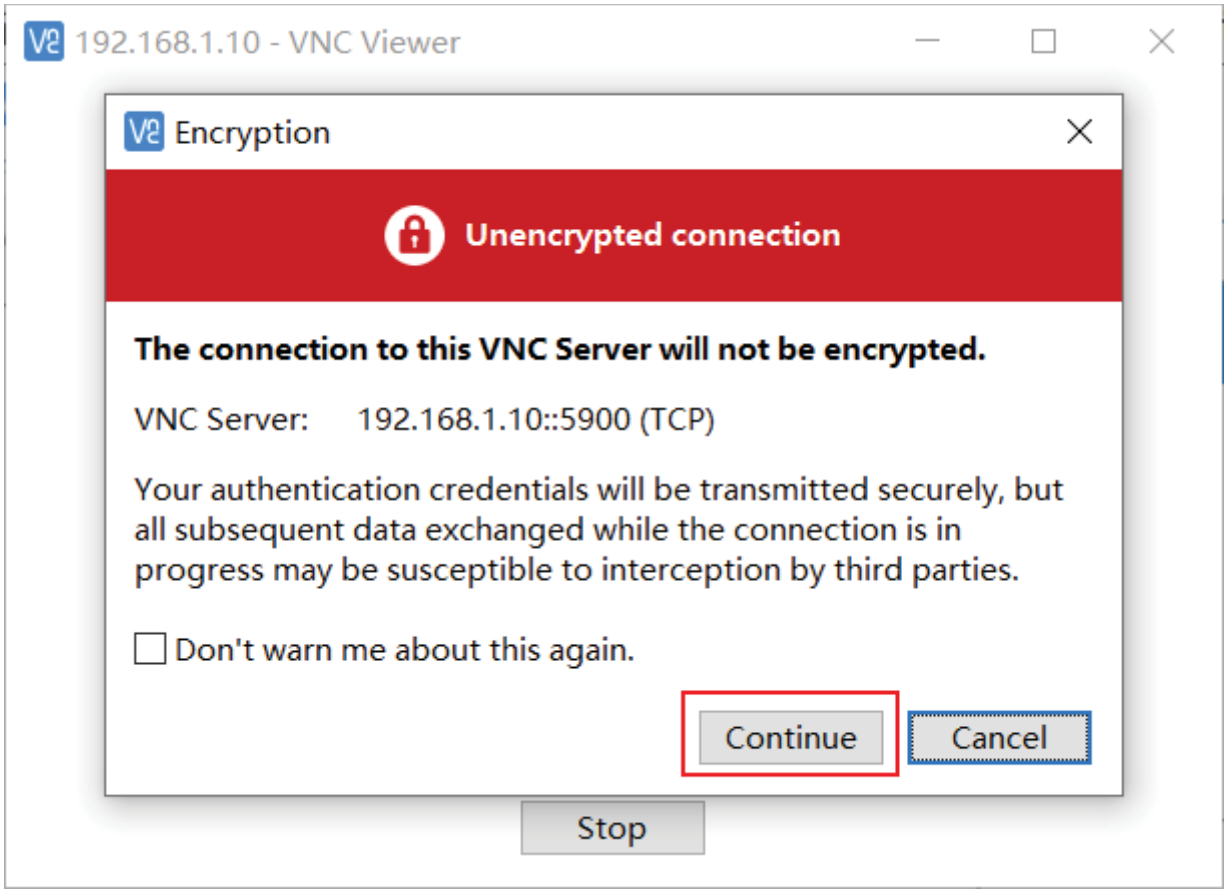
Connect to the Development Board

Currently, VNC supports two connection methods: direct and cloud. Users can choose according to their own situation. This article recommends using the direct connection method, and the connection steps are as follows:

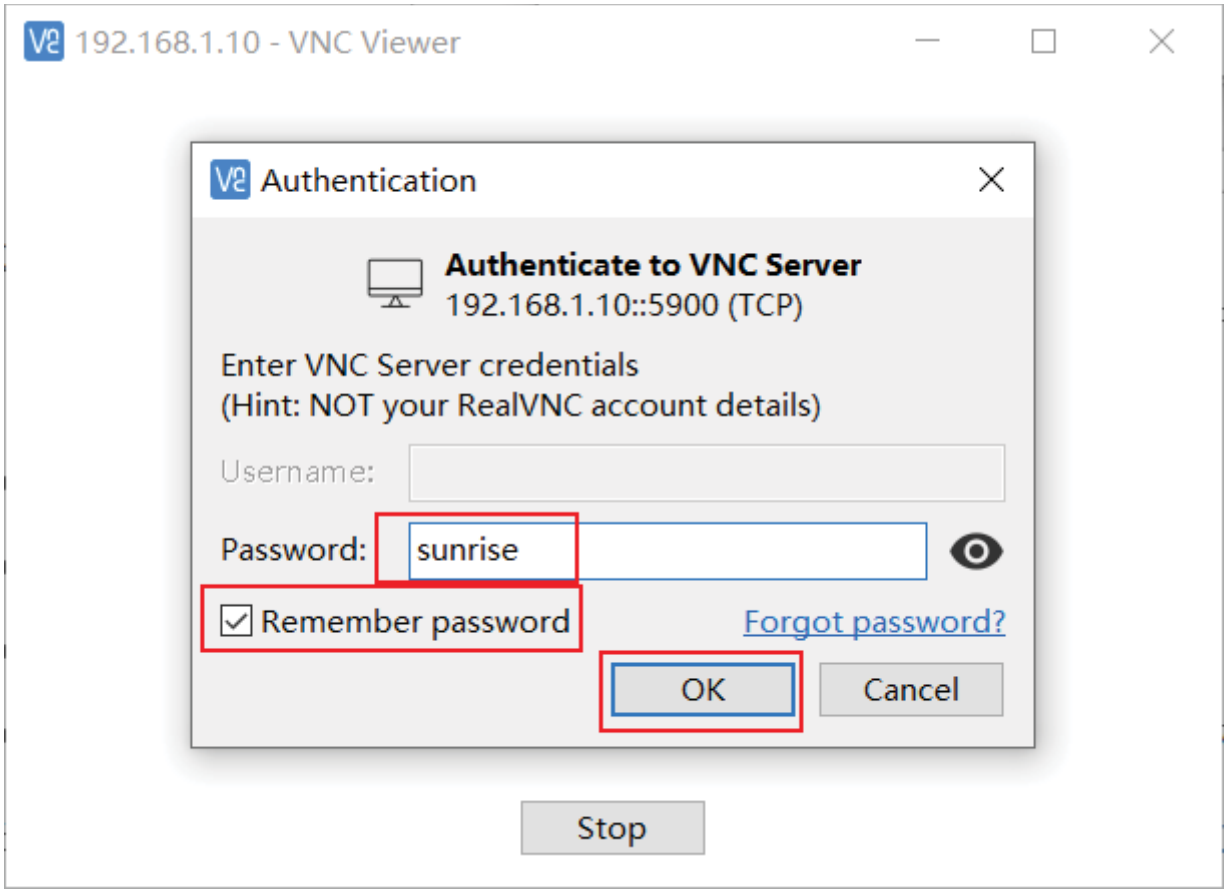
- Enter the device IP address, for example: 192.168.127.10



- After entering the IP address and pressing Enter, a prompt indicating that the link is not encrypted will pop up, click `Continue`



- Enter the password `sunrise`, check `Remember password`, and click `OK` to connect



SSH Login{#ssh}

In addition to VNC remote desktop login, you can also log in to the development board through SSH connection. The creation steps using terminal software and terminal command line are introduced below.

Terminal Software

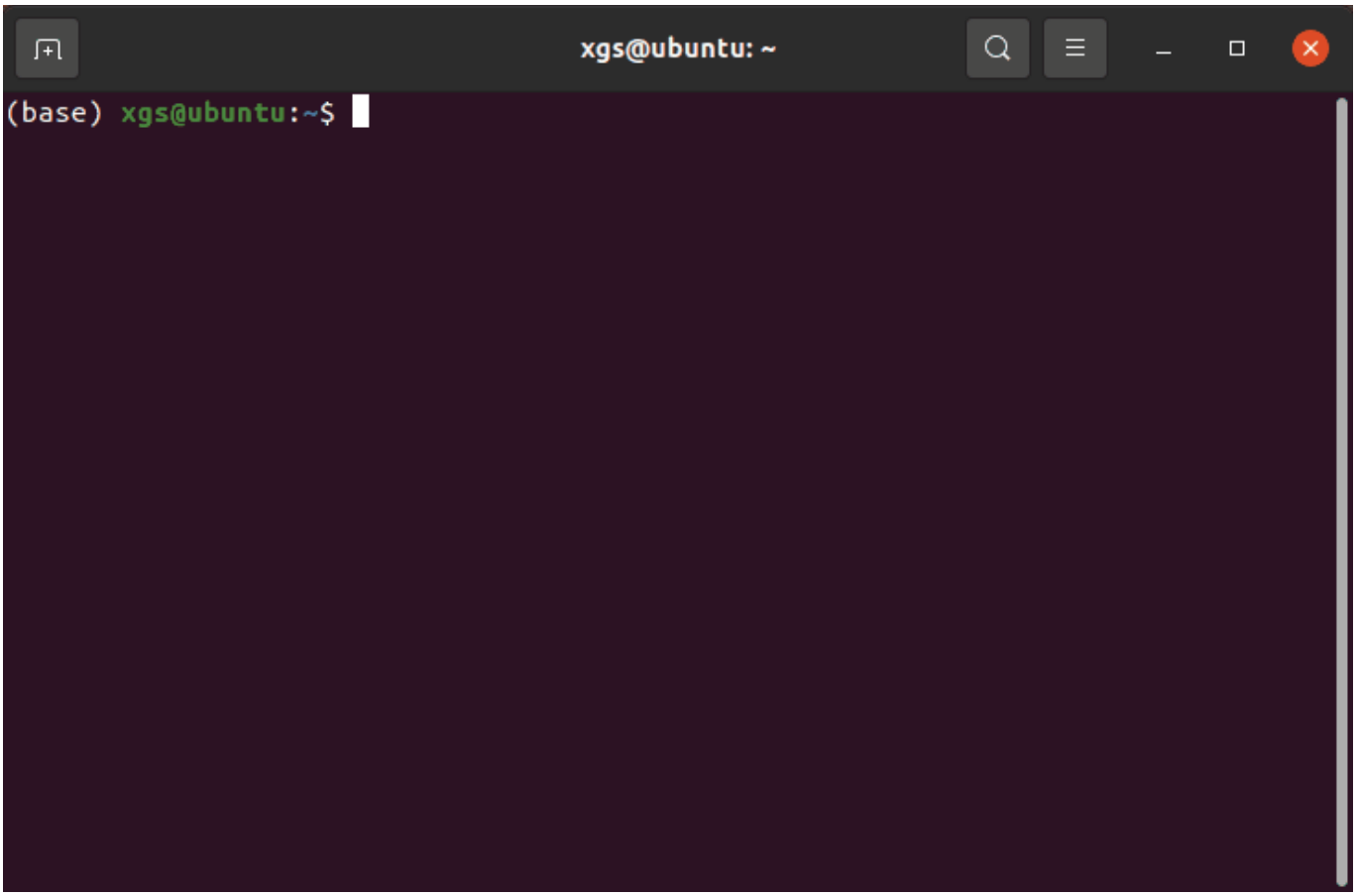
Commonly used terminal tools currently include `Putty`, `MobaXterm`, etc. Users can choose according to their own usage habits. The port configuration process for different tools is basically similar. Below, taking `MobaXterm` as an example, the process of creating a new SSH connection is introduced:

1. Open the `MobaXterm` tool, click `Session`, then select `SSH`
2. Enter the development board IP address, for example, `192.168.127.10`
3. Check `specify username` and enter `sunrise`
4. After clicking OK, enter the username (sunrise) and password (sunrise) to complete the login

Computer Command Line

Users can also perform SSH login through the command line. The steps are as follows:

1. Open the terminal window and enter the SSH login command, for example, `ssh sunrise@192.168.127.10`
2. A connection confirmation prompt will pop up, enter YES
3. Enter the password (sunrise) to complete the login



Local Area Network Construction

For the login of the above steps, the serial port line connection always needs to be maintained. Use the following commands to achieve local area network access

```
sudo nmcli device wifi rescan # Scan wifi network
sudo nmcli device wifi list # List the found wifi
sudo wifi_connect "SSID" "PASSWD" # Connect to the specified wifi
```

After the above commands are successful, `successfully xxx` will appear.

Finally, use `ifconfig` on the board end to obtain the board IP address, then you can unplug the serial port line and use the previous SSH login for remote connection.

---

sidebar_position: 1

2.1 Network and Bluetooth Configuration

This chapter mainly introduces the modification methods for the development board's wired and wireless network configurations.

Wired Network{#config_ethnet}

Video: https://www.bilibili.com/video/BV1rm4y1E73q/?p=11

The development board's wired network uses static IP configuration by default, with the initial IP address being `192.168.127.10`. Users can switch between static and DHCP modes using the following methods.

Modify Static IP Configuration

The development board's static network configuration is saved in the `/etc/network/interfaces` file. By modifying fields such as `address`, `netmask`, and `gateway`, the static IP configuration can be modified. `metric` is the network priority configuration, set to `700` to make the wired network's priority lower. When both wired and wireless networks are enabled, the wireless network will be prioritized, for example:

```
sudo vim /etc/network/interfaces
```

```
# interfaces(5) file used by ifup(8) and ifdown(8)
# Include files from /etc/network/interfaces.d:
source-directory /etc/network/interfaces.d
auto eth0
iface eth0 inet static
    pre-up /etc/set_mac_address.sh
    address 192.168.127.10
    netmask 255.255.255.0
    gateway 192.168.127.1
    metric 700
```

After modification, enter the `sudo restart_network` command in the command line to make the configuration take effect.

Modify DHCP Configuration

DHCP (Dynamic Host Configuration Protocol) is usually applied in local network environments. Its main function is centralized management and allocation of IP addresses, allowing hosts in the network environment to dynamically obtain IP addresses, Gateway addresses, DNS server addresses, and other information, and improving the utilization rate of addresses.

The development board's DHCP network configuration is saved in the `/etc/network/interfaces` file. By modifying the eth0 related configuration, the DHCP mode can be modified, for example:

```
sudo vim /etc/network/interfaces
```

```
source-directory /etc/network/interfaces.d
auto lo
iface lo inet loopback
auto eth0
iface eth0 inet dhcp
    metric 700
```

After modification, enter the `sudo restart_network` command in the command line to make the configuration take effect.

Modify MAC Address Configuration

To modify the development board's default MAC address, you can add `pre-up` configuration information in the `/etc/network/interfaces` file to specify the required MAC address, for example:

```
sudo vim /etc/network/interfaces
```

```
# interfaces(5) file used by ifup(8) and ifdown(8)
# Include files from /etc/network/interfaces.d:
source-directory /etc/network/interfaces.d
auto eth0
iface eth0 inet static
    pre-up /etc/set_mac_address.sh
    address 192.168.127.10
    netmask 255.255.255.0
    gateway 192.168.127.1
    metric 700
    pre-up ifconfig eth0 hw ether 00:11:22:9f:51:27
```

After modification, `reboot` to restart and make the configuration take effect.

Wireless Network

Video: https://www.bilibili.com/video/BV1rm4y1E73q/?p=12

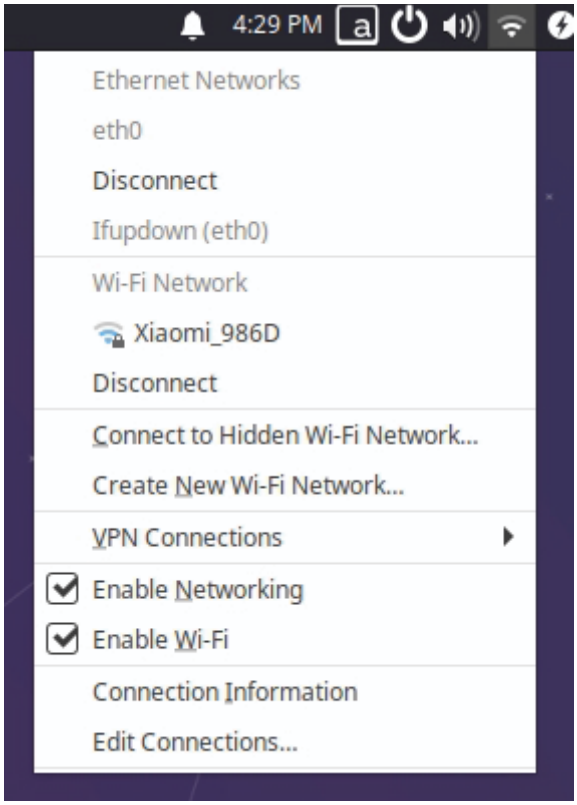
The development board integrates a 2.4GHz wireless WiFi module, supporting Soft AP and Station modes, and runs in Station mode by default. The usage methods of the two modes are introduced below.

Station Mode

In Station mode, the development board acts as a client, connecting to the router's wireless hotspot for networking.

- For users using the Ubuntu Desktop version system, click the Wi-Fi icon in the upper right corner of the desktop, select the corresponding hotspot and enter the password to complete the network

configuration, as shown in the figure below:



- For users using the Ubuntu Server version system, wireless network configuration can be completed through the command line. The steps are as follows:

1. Use the `sudo nmcli device wifi rescan` command to scan hotspots. If the following information is returned, it means the scanning is too frequent and you need to try again later

```
root@ubuntu:~# sudo nmcli device wifi rescan
Error: Scanning not allowed immediately following previous scan.
```

2. Use the `sudo nmcli device wifi list` command to list the scanned hotspots
3. Use the `sudo wifi_connect "SSID" "PASSWD"` command to connect to the hotspot. If the following information is returned, it means the network connection is successful

```
root@ubuntu:~# sudo wifi_connect "WiFi-Test" "12345678"
Device 'wlan0' successfully activated with 'd7468833-4195-45aa-aa33-
3d43da86e1a7'.
```

If after connecting to the hotspot, the following information is returned, it means the hotspot was not found, and you can execute the `sudo nmcli device wifi rescan` command to scan again and reconnect

```
root@ubuntu:~# sudo wifi_connect "WiFi-Test" "12345678"
Error: No network with SSID 'WiFi-Test' found.
```

Soft AP Mode

The development board's wireless network runs in Station mode by default. If you need to use Soft AP mode, please follow the steps below for configuration.

1. Install `hostapd` and `isc-dhcp-server`

```
sudo apt update
sudo apt install hostapd
```

```
    ```

2. Run the `sudo vim /etc/hostapd.conf` command to configure `hostapd.conf`, mainly focusing on the following fields:

    ```shell
    interface=wlan0 # Network card as AP hotspot
    ssid=Sunrise # WiFi name
    wpa=2 # 0 for WPA, 2 for WPA2, generally 2
    wpa_key_mgmt=WPA-PSK # Encryption algorithm, generally WPA-PSK
    wpa_passphrase=12345678 # Password
    wpa_pairwise=CCMP # Encryption protocol, generally CCMP
    ```

      - For password-free hotspot configuration, add the following content to the `hostapd.conf` file:

    ```shell
    interface=wlan0
    driver=nl80211
    ctrl_interface=/var/run/hostapd
    ssid=Sunrise
    channel=6
    ieee80211n=1
    hw_mode=g
    ignore_broadcast_ssid=0
    ```

      - For hotspot configuration with a password, add the following content to the `hostapd.conf` file:

    ```shell
    interface=wlan0
    driver=nl80211
    ctrl_interface=/var/run/hostapd
    ssid=Sunrise
    channel=6
    ieee80211n=1
    hw_mode=g
    ignore_broadcast_ssid=0
```

```
    wpa=2
    wpa_key_mgmt=WPA-PSK
    wpa_pairwise=CCMP
    wpa_passphrase=12345678
    ```

3. Configure the `isc-dhcp-server` file, follow these steps:

    - Execute `sudo vim /etc/default/isc-dhcp-server` to modify the `isc-dhcp-server` file, add the defined network interface:
    ```shell
    INTERFACESv4="wlan0"
    ```
    - Execute `sudo vim /etc/dhcp/dhcpd.conf` to modify the `dhcpd.conf` file, uncomment the following fields:
    ```shell
      authoritative;
    ```
    - Then add the following configuration at the end of the `/etc/dhcp/dhcpd.conf` file:
    ```shell
      subnet 10.5.5.0 netmask 255.255.255.0 { # Network segment and subnet mask
      range 10.5.5.100 10.5.5.254; # Obtainable IP range
      option subnet-mask 255.255.255.0; # Subnet mask
      option routers 10.5.5.1; # Default gateway
      option broadcast-address 10.5.5.31; # Broadcast address
      default-lease-time 600; # Default lease time in seconds
      max-lease-time 7200; # Maximum lease time in seconds
    }
    ```

4. Stop the `wpa_supplicant` service, and restart `wlan0`

    ```bash
    systemctl mask wpa_supplicant
    systemctl stop wpa_supplicant

    ip addr flush dev wlan0
    sleep 0.5
    ifconfig wlan0 down
    sleep 1
    ifconfig wlan0 up
    ```

5. Start the `hostapd` service according to the following steps:
   - Execute the `sudo hostapd -B /etc/hostapd.conf` command
   ```bash
    root@ubuntu:~# sudo hostapd -B /etc/hostapd.conf

    Configuration file: /etc/hostapd.conf
    Using interface wlan0 with hwaddr 08:e9:f6:af:18:26 and ssid "sunrise"
    wlan0: interface state UNINITIALIZED->ENABLED
    wlan0: AP-ENABLED
```

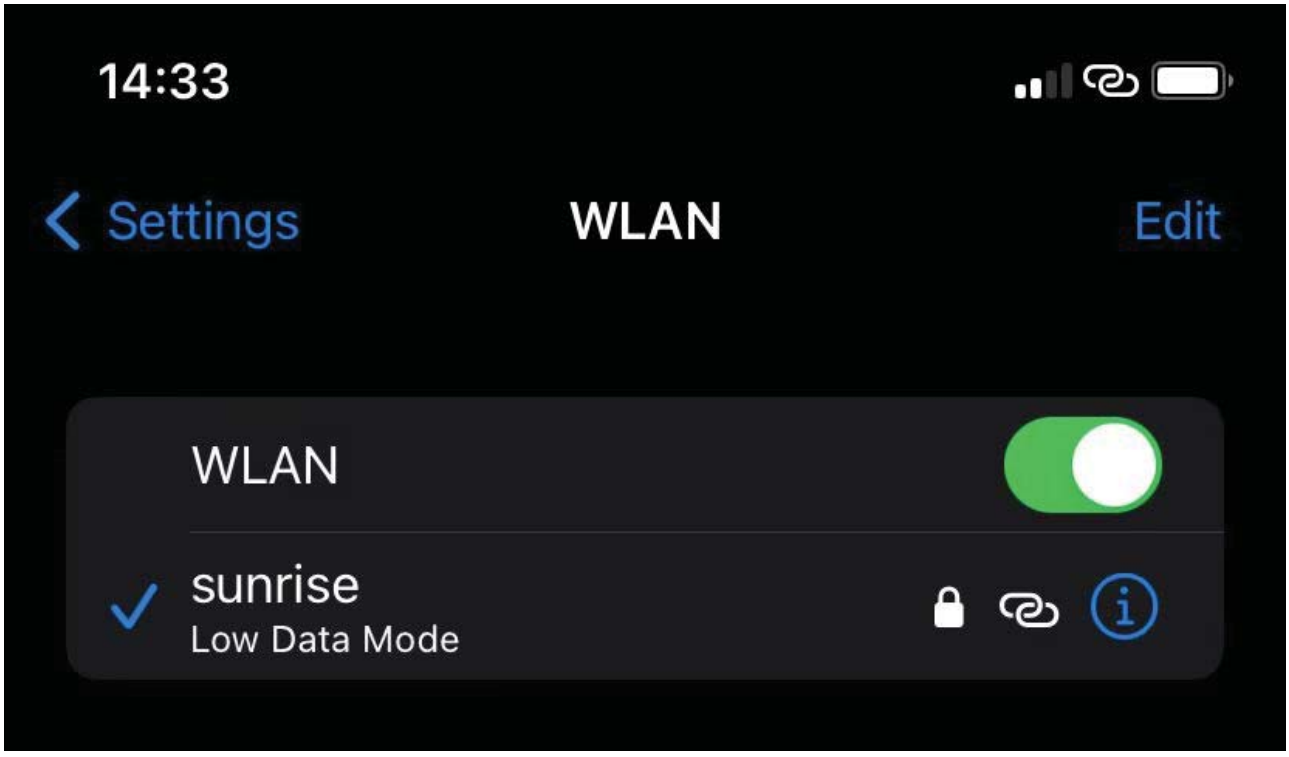
- Use the `ifconfig` command to configure the IP and network segment of the wireless interface `wlan0`. Note that it should be consistent with the configuration in step three.

```
```bash
sudo ifconfig wlan0 10.5.5.1 netmask 255.255.255.0
```
```

- Finally, start the `dhcp` server, connect to the hotspot and an IP address between `10.5.5.100` and `10.5.5.255` will be assigned to the client.

```
```bash
sudo ifconfig wlan0 10.5.5.1 netmask 255.255.255.0
sudo systemctl start isc-dhcp-server
sudo systemctl enable isc-dhcp-server
```
```

6. Connect to the development board hotspot, for example `sunrise`



7. If you need to switch back to `Station` mode, you can do so as follows:

   [RDK X5]

```
# Stop hostapd
killall -9 hostapd

# Clear the address of wlan0
```

```
ip addr flush dev wlan0
sleep 0.5
ifconfig wlan0 down
sleep 1
ifconfig wlan0 up

# Restart wpa_supplicant
systemctl unmask wpa_supplicant
systemctl restart wpa_supplicant

# Reinstall wifi driver
rmmod aic8800_fdrv
modprobe aic8800_fdrv

# Connect to hotspot, for specific operations, refer to the previous
chapter "Wireless Network"
wifi_connect "WiFi-Test" "12345678"
```

[Other]

```
# Stop hostapd
killall -9 hostapd

# Clear the address of wlan0
ip addr flush dev wlan0
sleep 0.5
ifconfig wlan0 down
sleep 1
ifconfig wlan0 up

# Restart wpa_supplicant
systemctl unmask wpa_supplicant
systemctl restart wpa_supplicant

# Connect to hotspot, for specific operations, refer to the previous
chapter "Wireless Network"
wifi_connect "WiFi-Test" "12345678"
```

DNS Service

Video: https://www.bilibili.com/video/BV1rm4y1E73q/?p=13

DNS(Domain Name Server) is a server that converts domain names to corresponding IP addresses.

The DNS configuration of the development board is managed through the `/etc/systemd/resolved.conf` file. Users can complete the DNS-related configuration by modifying this file. The steps are as follows:

1. Modify the `resolved.conf` file, add DNS server addresses, for example:

```
DNS=8.8.8.8 114.114.114.114
```

2. Enable DNS configuration with the following commands:

```
sudo systemctl restart systemd-resolved
sudo systemctl enable systemd-resolved
sudo mv /etc/resolv.conf  /etc/resolv.conf.bak
sudo ln -s /run/systemd/resolve/resolv.conf /etc/
```

System Update

For system security and stability considerations, it is recommended that users update the system via the `apt` command after installing the system.

In the `/etc/apt/source.list` file, the software source list of the `apt` command is saved. Before installing the software, you need to update the package list first via the `apt` command.

First, open the terminal command line and enter the following command:

```
sudo apt update
```

Next, upgrade all installed packages to the latest version, the command is as follows:

```
sudo apt full-upgrade
```

It is recommended to use `full-upgrade` instead of `upgrade` option, so that when related dependencies change, dependency packages will also be updated synchronously.

When running the `sudo apt full-upgrade` command, the system will prompt data download and disk occupation size, but `apt` does not check whether there is enough disk space. It is recommended that users manually check with the `df -h` command. In addition, the downloaded deb files during the upgrade process will be saved in the `/var/cache/apt/archives` directory. Users can delete cache files to release disk space with the `sudo apt clean` command.

After executing the `apt full-upgrade` command, drivers, kernel files, and some system software may be reinstalled. It is recommended that users manually restart the device to make the updates effective. The command is as follows:

```
sudo reboot
```

Bluetooth ConfigurationVideo: https://www.bilibili.com/video/BV1rm4y1E73q/?p=9

Initialization

The development board's Bluetooth function is not enabled by default and requires executing the `/usr/bin/startbt6212.sh` script for initialization. This script completes the following tasks:

- Reset Bluetooth
- Create the `messagebus` user and user group, which the `dbus-daemon` program needs to use when running
- Run `brcm_patchram_plus` to complete Bluetooth driver and firmware loading
- Loop to check if the `/sys/class/bluetooth/hci0` directory exists to confirm that the Bluetooth driver is running normally
- The appearance of **Done setting line discipline** indicates successful Bluetooth activation
- Execute `hciconfig hci0 up` to complete Bluetooth Link Up
- Execute `hciconfig hci0 piscan` for Bluetooth scanning; this step can be omitted depending on the situation

The log after successful script execution is as follows:

```
root@ubuntu:~# sudo startbt6212.sh
Waiting for bluetooth initialize..............Done setting line discpline
.Done
Check Bluetooth State...unblocked
Set Bluetooth Up...
Set Bluetooth piscan...
hci0: Type: Primary  Bus: UART
 BD Address: 08:E9:F6:AE:F8:8B  ACL MTU: 1021:8  SCO MTU: 64:1
 UP RUNNING PSCAN ISCAN
 RX bytes:759 acl:0 sco:0 events:48 errors:0
 TX bytes:2522 acl:0 sco:0 commands:48 errors:0

root@ubuntu:~#
```

In addition, users can use commands to check if the Bluetooth process is normal, as follows:

```
ps ax | grep "/usr/bin/dbus-daemon\|/usr/lib/bluetooth/bluetoothd"
/usr/bin/dbus-daemon

/usr/lib/bluetooth/bluetoothd
```

Network Connection

Execute `sudo bluetoothctl` to enter the Bluetooth configuration interface in interactive mode. The appearance of device information similar to the figure below indicates that Bluetooth has been recognized. Then use `show` to view Bluetooth information and pay attention to the `powered` and `discoverable` status of Bluetooth.

```
root@ubuntu:~# bluetoothctl
Agent registered
[CHG] Controller 08:E9:F6:AF:18:27 Pairable: yes
[bluetooth]# show
Controller 08:E9:F6:AF:18:27 (public)
        Name: ubuntu
        Alias: ubuntu
        Class: 0x00000000
        Powered: yes
        Discoverable: yes
        DiscoverableTimeout: 0x000000b4
        Pairable: yes
        UUID: Generic Attribute Profile (00001801-0000-1000-8000-00805f9b34fb)
        UUID: Generic Access Profile    (00001800-0000-1000-8000-00805f9b34fb)
        UUID: PnP Information           (00001200-0000-1000-8000-00805f9b34fb)
        UUID: A/V Remote Control Target (0000110c-0000-1000-8000-00805f9b34fb)
        UUID: A/V Remote Control        (0000110e-0000-1000-8000-00805f9b34fb)
        Modalias: usb:v1D6Bp0246d0535
        Discovering: no
Advertising Features:
        ActiveInstances: 0x00
        SupportedInstances: 0x05
        SupportedIncludes: tx-power
        SupportedIncludes: appearance
        SupportedIncludes: local-name
```

Execute `power on` to enable Bluetooth, as shown in the figure below:

```
[bluetooth]# power on
Changing power on succeeded
[bluetooth]#
```

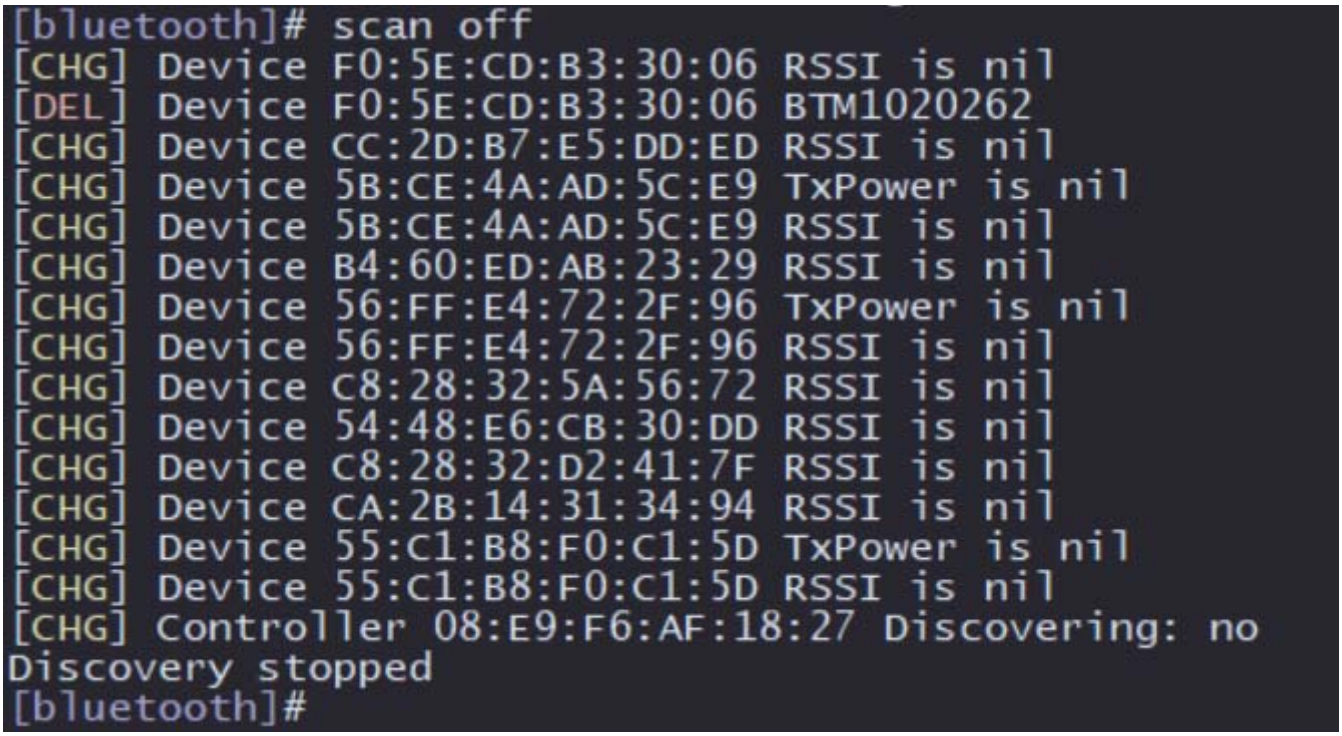
To make Bluetooth discoverable by nearby devices, execute `discoverable on` to enable Bluetooth and turn on the Bluetooth discoverable attribute, as shown in the figure below:

```
[bluetooth]# discoverable on
Changing discoverable on succeeded
[CHG] Controller 08:E9:F6:AF:18:27 Discoverable: yes
[bluetooth]#
```

At this point, using a mobile phone or computer to scan for Bluetooth will discover the Bluetooth device named `ubuntu`:

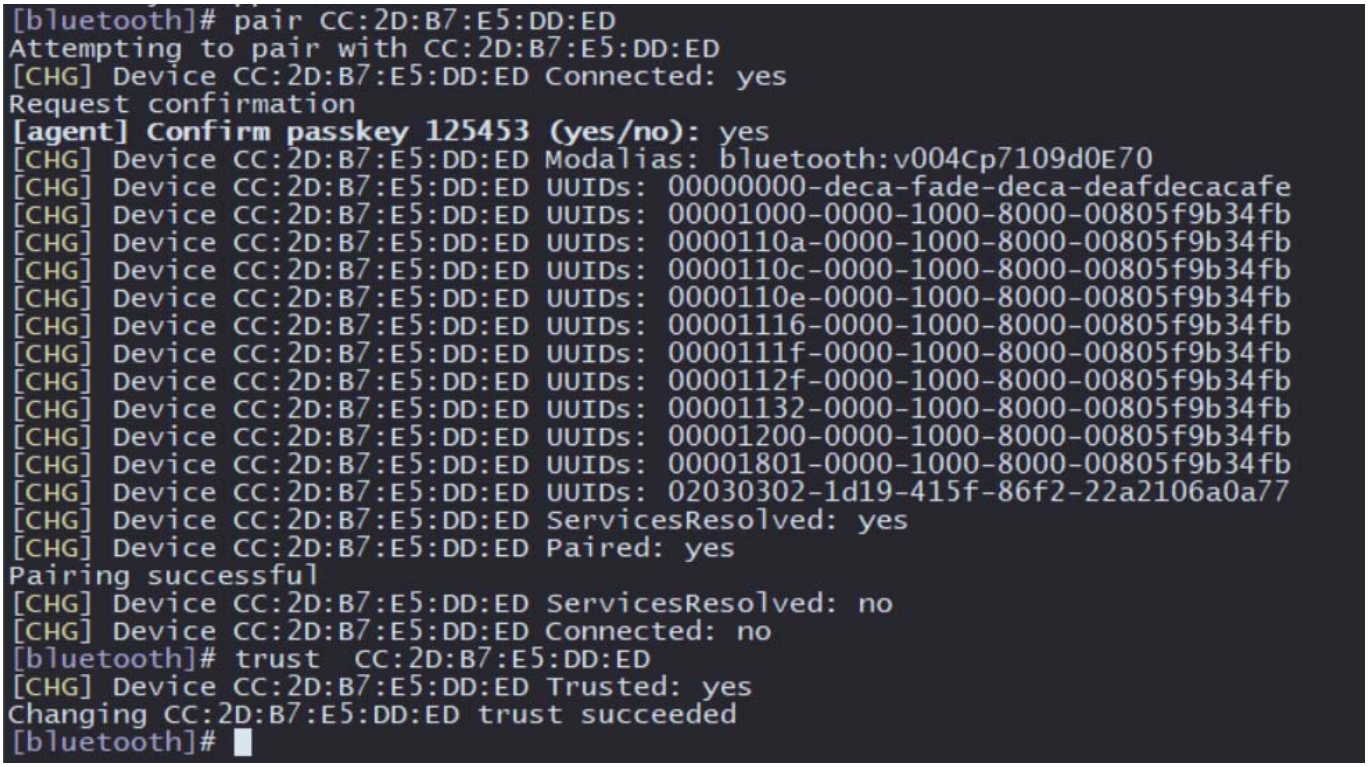
Next, test the active scanning function of Bluetooth. In the `bluetoothctl` interactive interface, input `scan on` to turn on active scanning, which will periodically print nearby devices. It can be seen that my mobile phone device has been discovered. Use `scan off` to turn off the scanning function and summarize the printed scanned Bluetooth devices:

```
[bluetooth]# scan on
Discovery started
[CHG] Controller 08:E9:F6:AF:18:27 Discovering: yes
[NEW] Device 55:C1:B8:F0:C1:5D 55-C1-B8-F0-C1-5D
[NEW] Device CA:2B:14:31:34:94 CA-2B-14-31-34-94
[NEW] Device C8:28:32:D2:41:7F C8-28-32-D2-41-7F
[NEW] Device 54:48:E6:CB:30:DD 54-48-E6-CB-30-DD
[CHG] Device C8:28:32:D2:41:7F RSSI: -69
[CHG] Device C8:28:32:D2:41:7F ServiceData Key: 0000fdaa-0000-1000-8000-00805f9b34fb
[CHG] Device C8:28:32:D2:41:7F ServiceData Value:
  30 35 31 21 26 22 a7 01 08 02 06 01 2d 6a 34 14  051!&".....-j4.
  67 83 e4 b8 bb e5 8d a7 e7 ff                    g.........
[NEW] Device F0:5E:CD:B3:30:06 BTM1020262
[NEW] Device C8:28:32:5A:56:72 C8-28-32-5A-56-72
[bluetooth]#
```



Then pair with other Bluetooth devices:

- Pairing command: `pair [targetMAC]`, after entering this command, input `yes` according to the prompt, and the paired Bluetooth device selects the `pair` option to complete pairing.
- After successful pairing, you can use `trust [targetMAC]` to allow automatic connection next time.



After the above operations, the basic functions of Bluetooth scanning and pairing are completed. If you need to use more functions, you can refer to the official help documentation of `BlueZ`.

---

sidebar_position: 2

2.2 srpi-config Tool Configuration

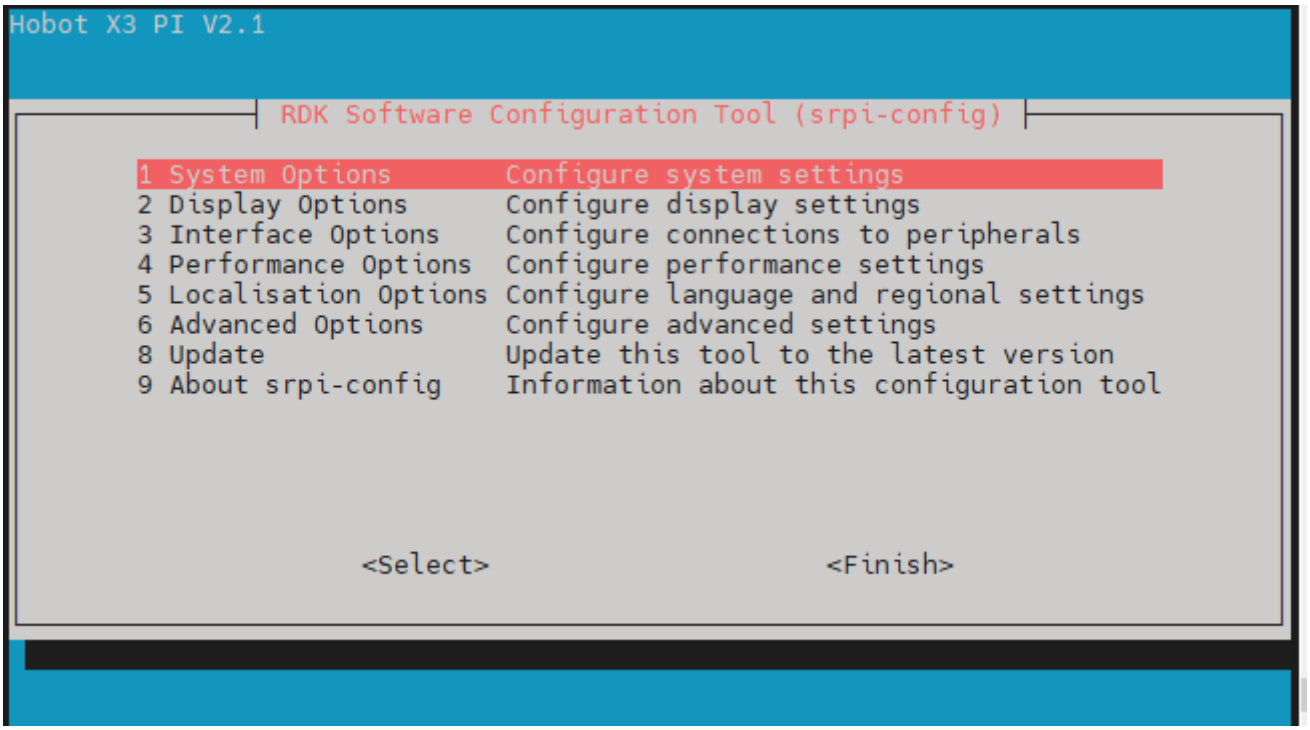
The `srpi-config` configuration tool is only applicable to `RDK X3`, `RDK X5` and `RDK X3 Module` development boards, not to `RDK Ultra` development boards.

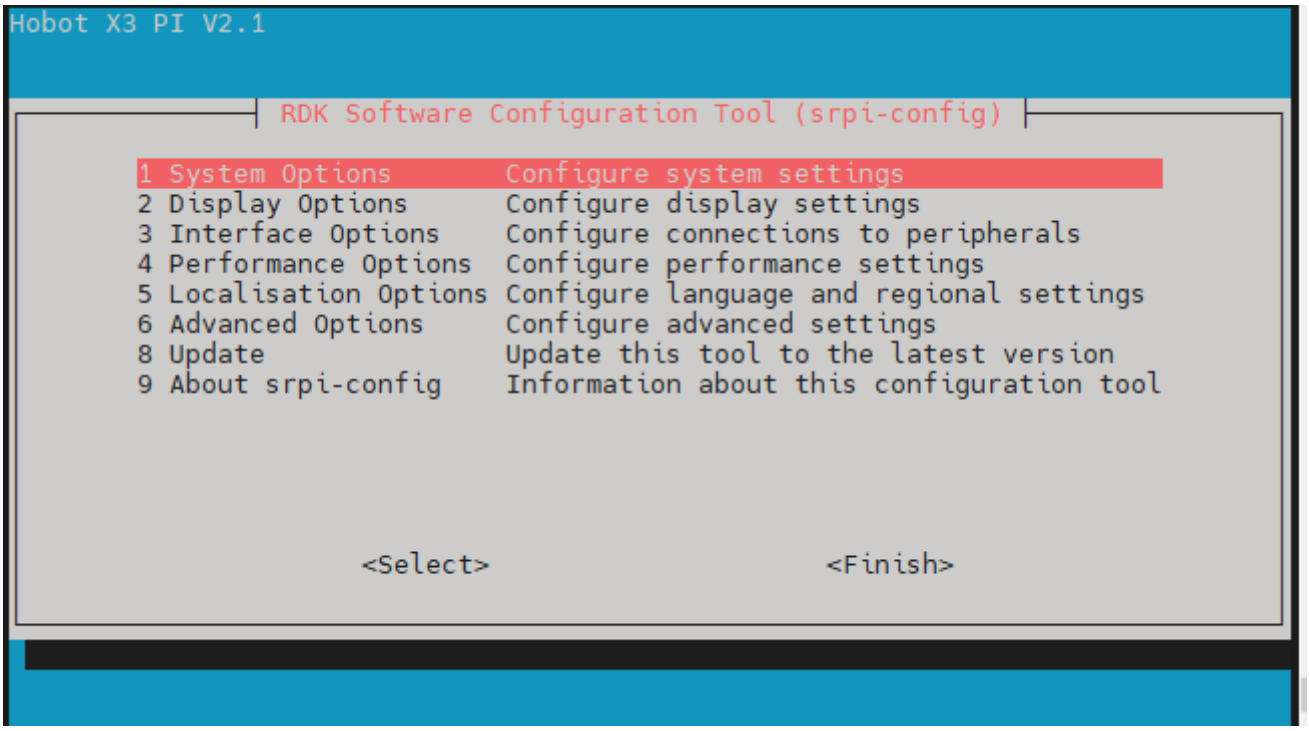
Introduction

`srpi-config` is a system configuration tool. To open the configuration tool, type the following command in the command line:

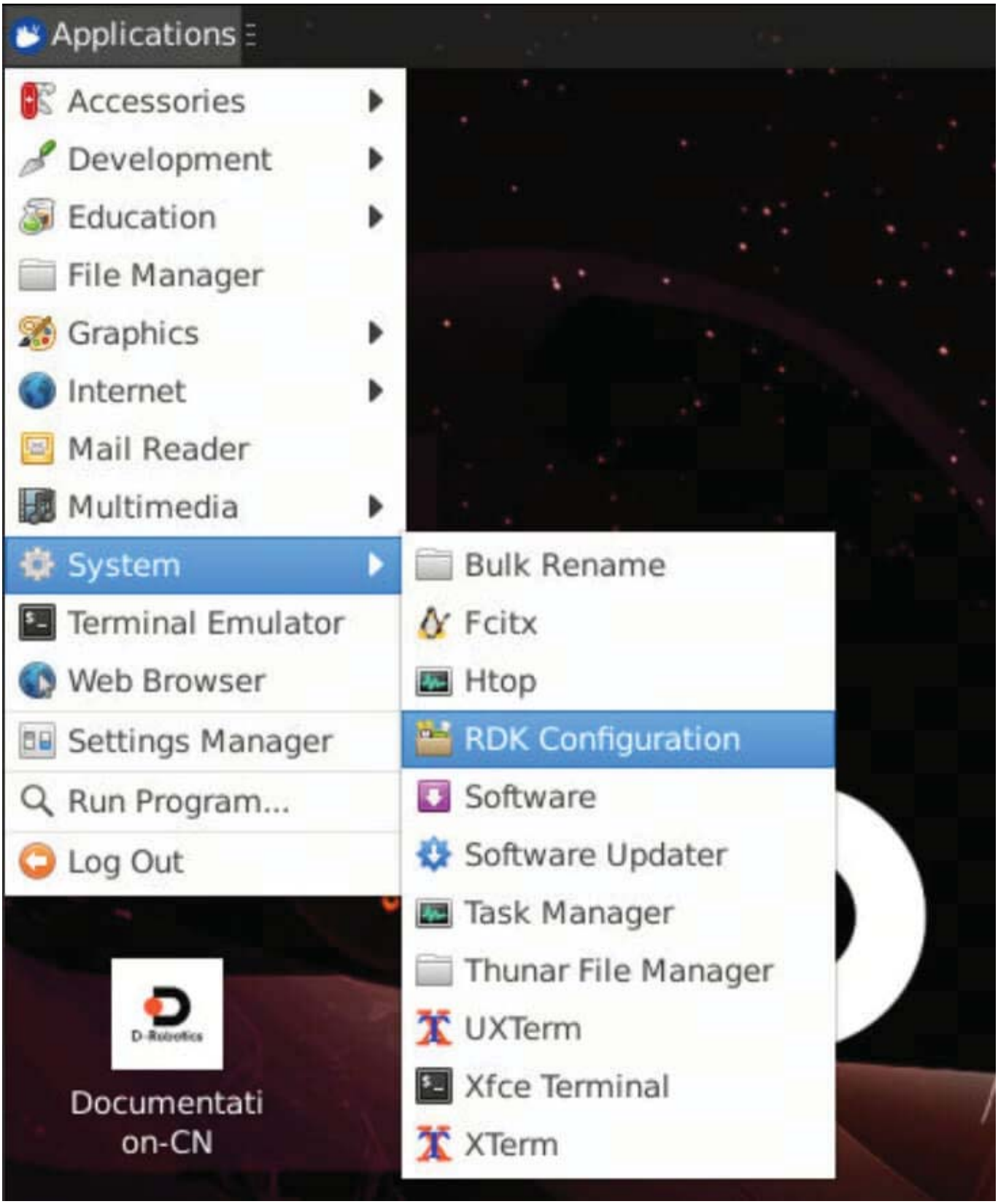
```
sudo srpi-config
```

> sudo is a privilege escalation management command, which must be entered here so that configuration management can be performed with root privileges. The default sunrise account does not have permission to modify system files.



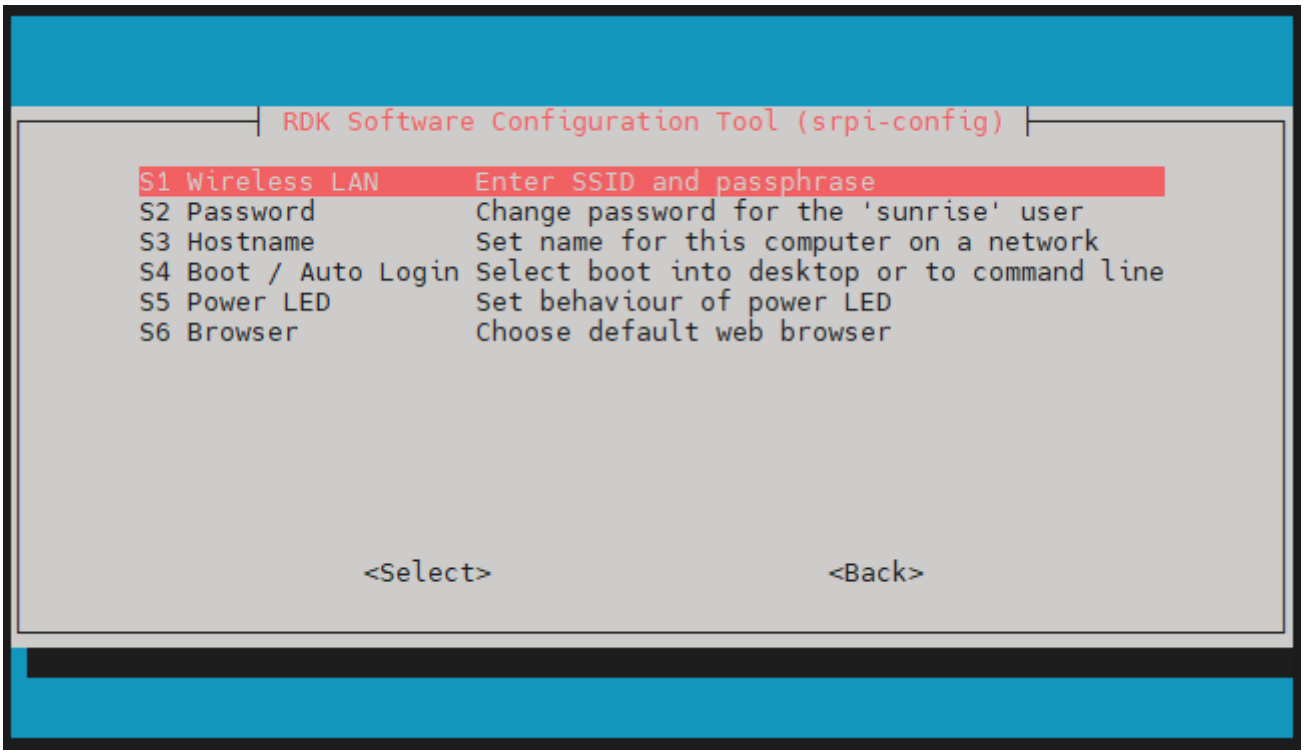


If you are using a desktop system, you can find the `RDK Configuration` application in the menu to perform configuration. It will also open a configuration terminal like the one shown in the figure above.



System Options

System options menu, allowing configuration changes to various parts such as Wi-Fi networks, user passwords, hostnames, system login modes, browser selection, and some system-level changes.



- **Wireless LAN**

  Set the `SSID` and `password` for the wireless network.

- **Password**

  Change the password of the "default" user, the default account is `sunrise`.

- **Hostname**

  Set the visible name of the current device on the network.

- **Boot / Auto login**

  Choose whether to boot into the console or desktop, and whether auto-login is required. If auto-login is selected, it will log in with the identity of the system's default account `sunrise`.

- **Power LED**

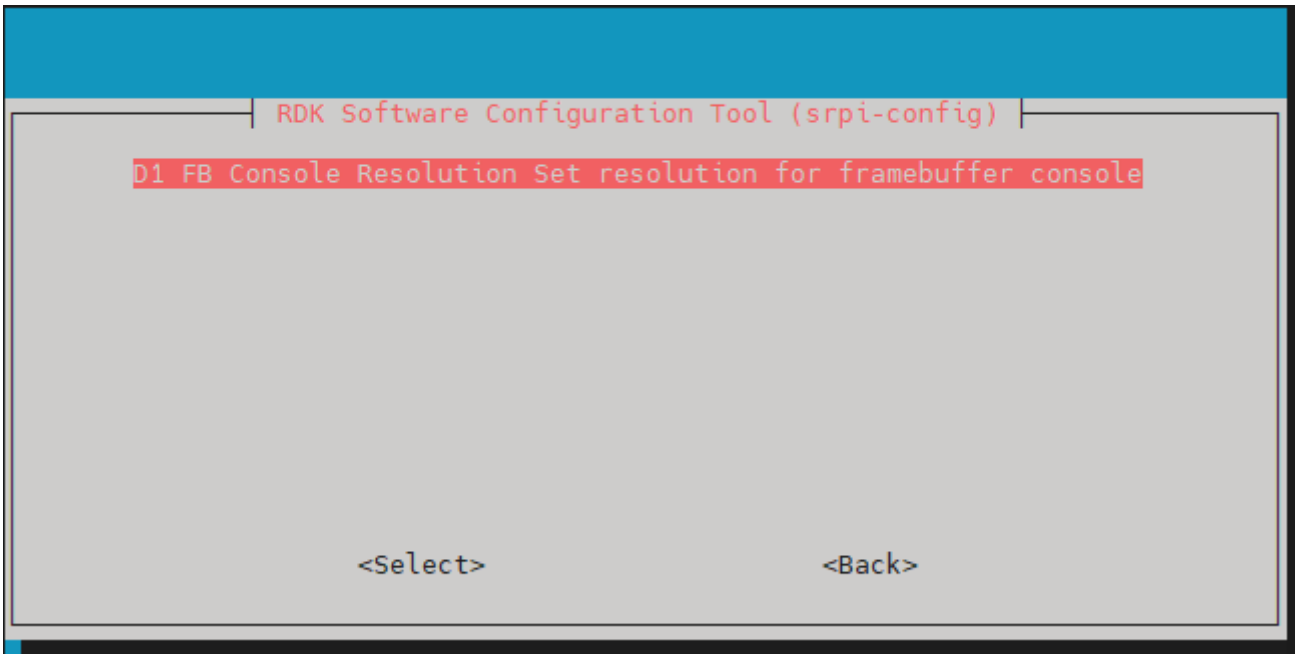
  If the model of RDK allows, the behavior of the power LED can be changed in this option. The default is off or blinking.

- **Browser**

  If using a desktop system, the default browser can be set. By default, `firefox` is used if not configured. Users can install the `chromium` browser via the command `sudo apt install chromium`.

Display Options

Display options menu.



- **FB Console Resolution**

  Supports setting HDMI display resolution in `Server` system and `console` mode.

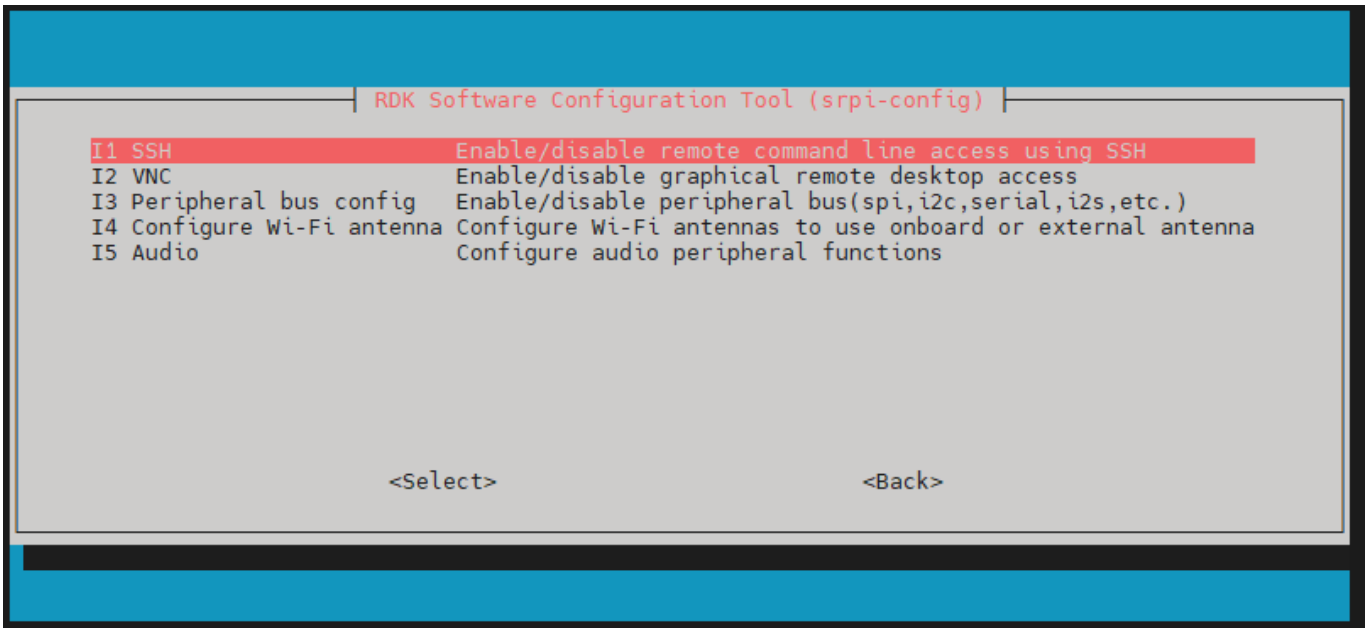
- **Display Chose DSI or HDMI**

  Supports switching between `DSI` and `HDMI` displays.

  Only RDK X5 supports display switching

Interface Options

Interface options menu, with the following options that can be enabled/disabled: SSH, VNC, SPI, I2C, I2S, Serial, etc.



- **SSH**

  Enable/disable remote login to RDK using SSH. By default, the system enables the SSH option.

- **VNC**

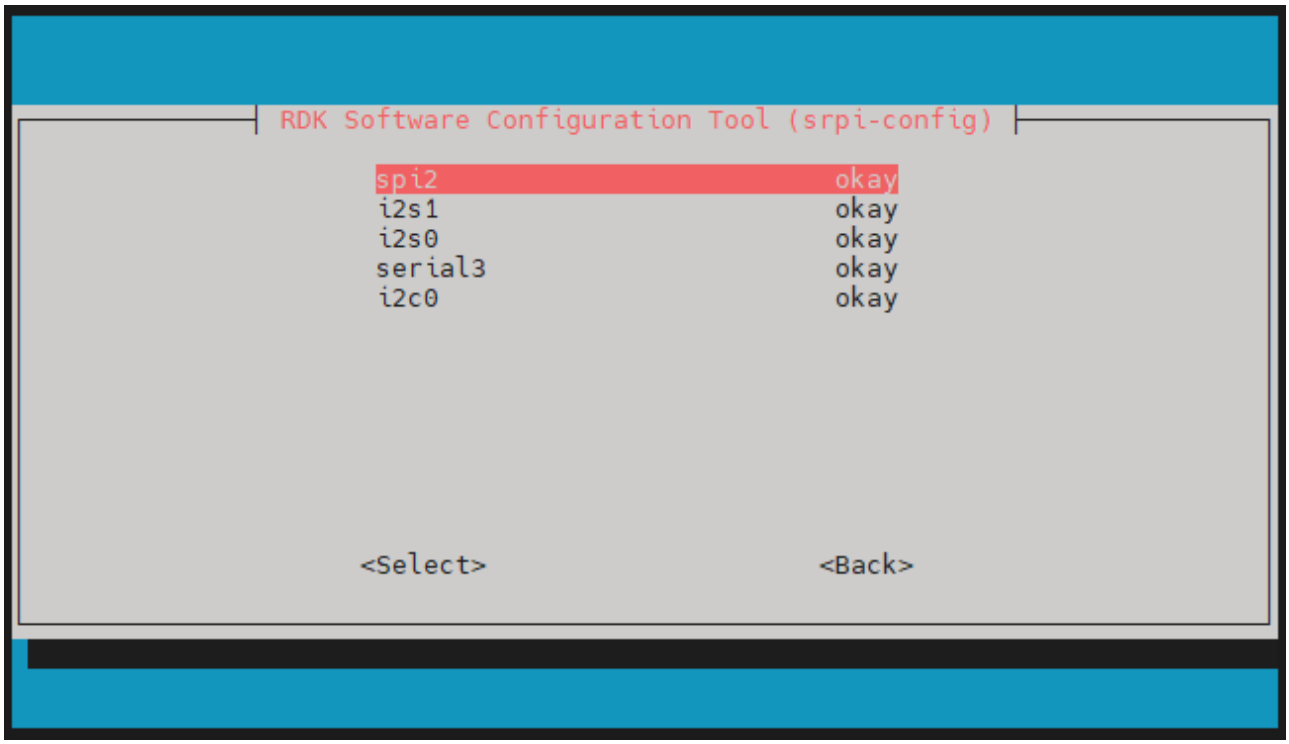
  Enable/disable the X11vnc Virtual Network Computing server.

> VNC (Virtual Network Console) is an abbreviation for Virtual Network Console. It is a remote control software. When remotely accessing the desktop system, this option can be enabled.

- **Peripheral bus config**

  Enable/disable interfaces such as SPI, I2C, Serial Port, I2S, etc., on the 40-pin interface. This configuration method directly modifies the `status` of the corresponding bus in the used device tree file, taking effect after reboot. [RDK X5] X5 adds PWM interface; In the table below, the interfaces in each row use the same pins, and only one function can be effective at the same time. When all interfaces in a row are disabled, the pin functions as a GPIO pin.

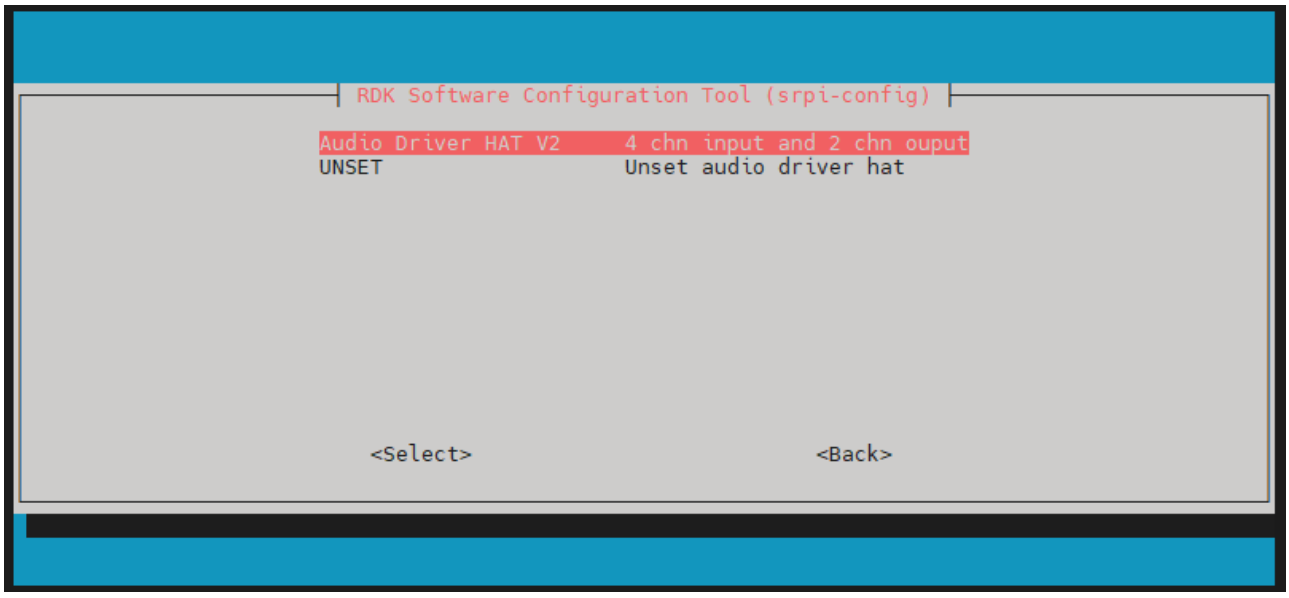
Function1	Function2
serial3	i2c5
i2c0	pwm2
spi2	pwm0
spi2	pwm1
i2c1	pwm3



- **Configure Wi-Fi antenna** Switch Wi-Fi antenna mode, support setting to use dipole antenna

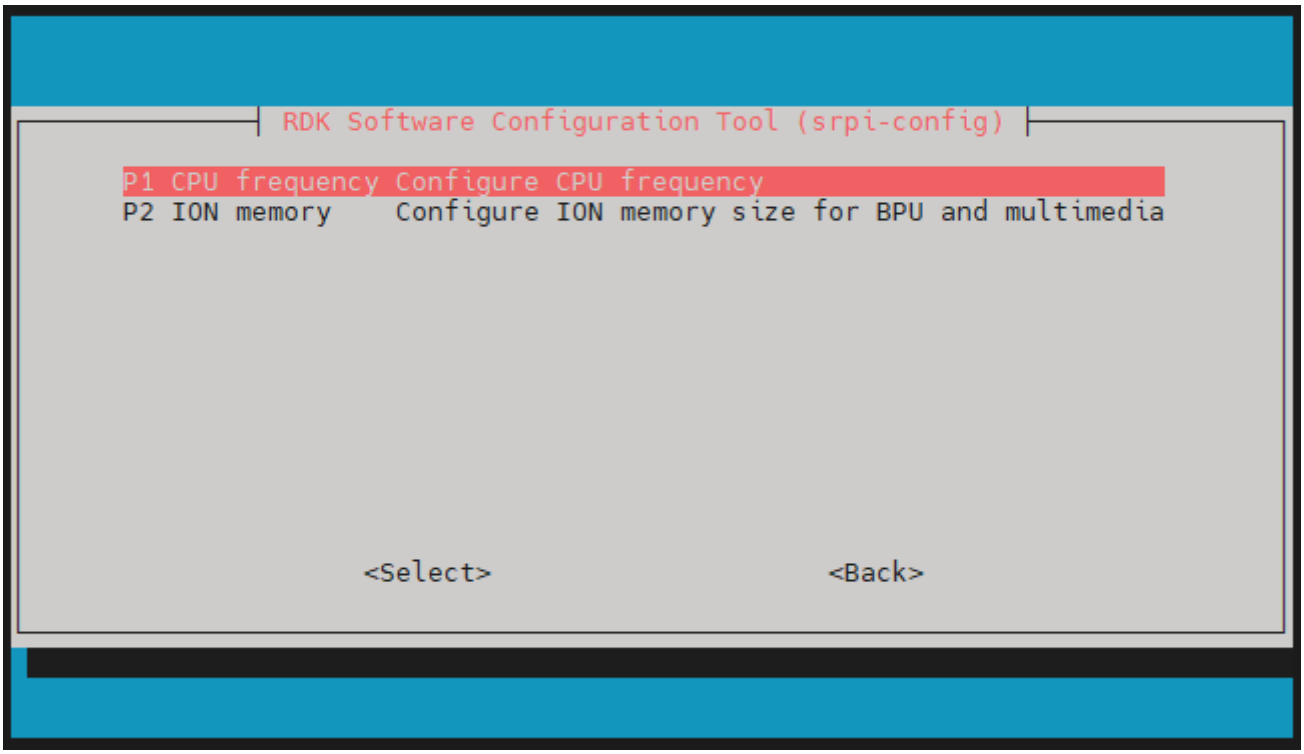
> Currently, only RDK X3 V2.1 and RDK X5 versions support.

- **Audio** Install and uninstall audio adapter boards Possible supported audio adapter boards include Audio Driver HAT V1, Audio Driver HAT V2, WM8960 Audio HAT. Different hardware supports different audio adapter boards too.



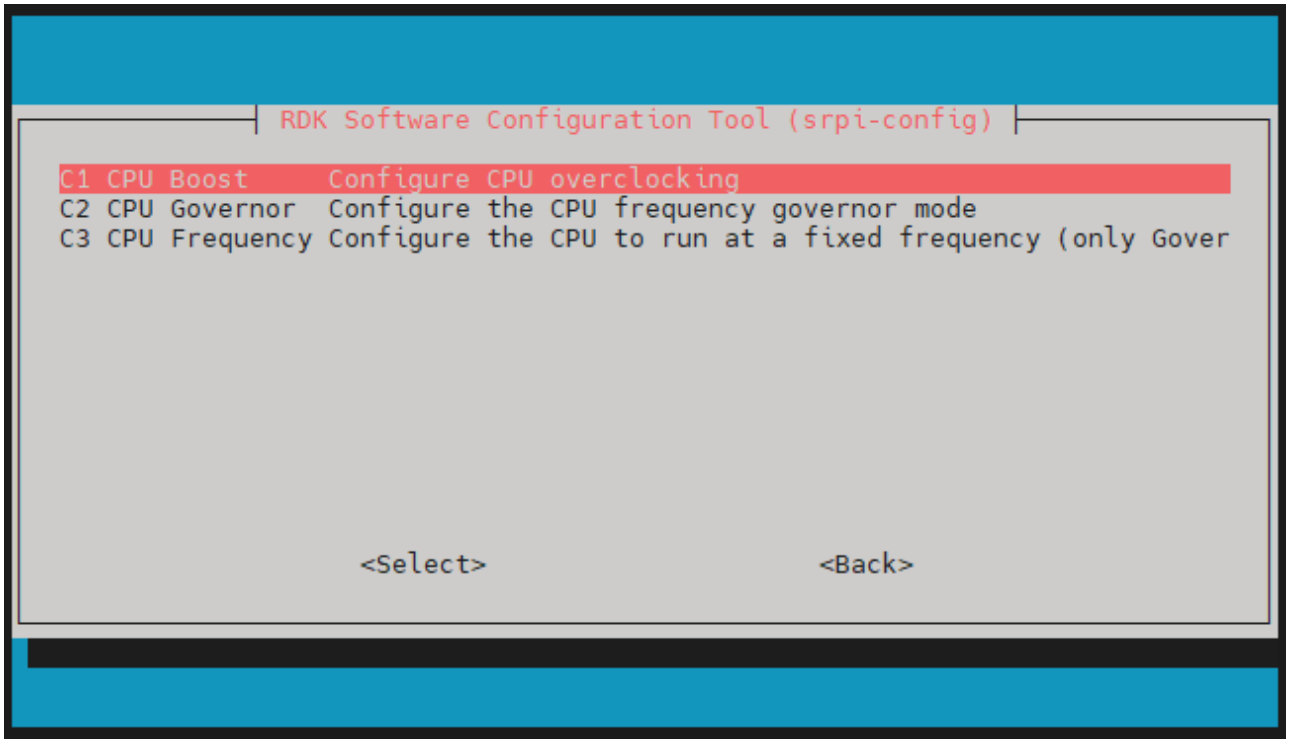
Performance Options

Performance options, including overclocking, CPU operation mode and fixed frequency settings, adjusting ION memory size, etc.



- **CPU frequency**

  This tool can be used to overclock the CPU of RDK. Generally, it is not recommended to enable it. If sufficient heat dissipation work is done, you can try modifying this option.

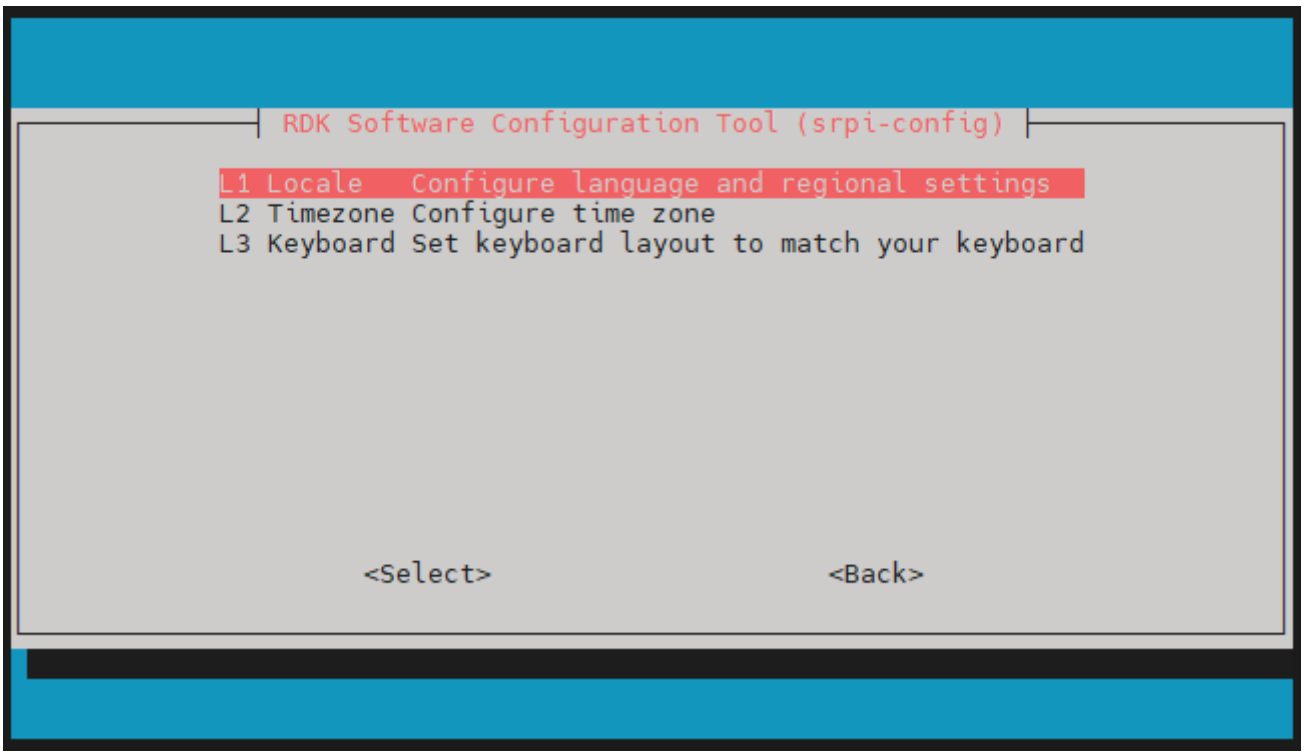


- **ION memory**

  The commonly used size of ION memory can be configured through this option.> ION memory is the physical memory space reserved for BPU and image, video multimedia functions. The default configuration size is 672MB. If you need to run relatively large algorithm models and code multiple video streams at the same time, please adjust the memory size according to your specific needs.

Localisation Options

Localisation options provide you with the following options to choose from: local language, time zone, keyboard layout.



- **Locale**

  Select a locale, such as configuring Chinese environment `zh_CN.UTF-8`, reboot to take effect.

- **Time Zone**

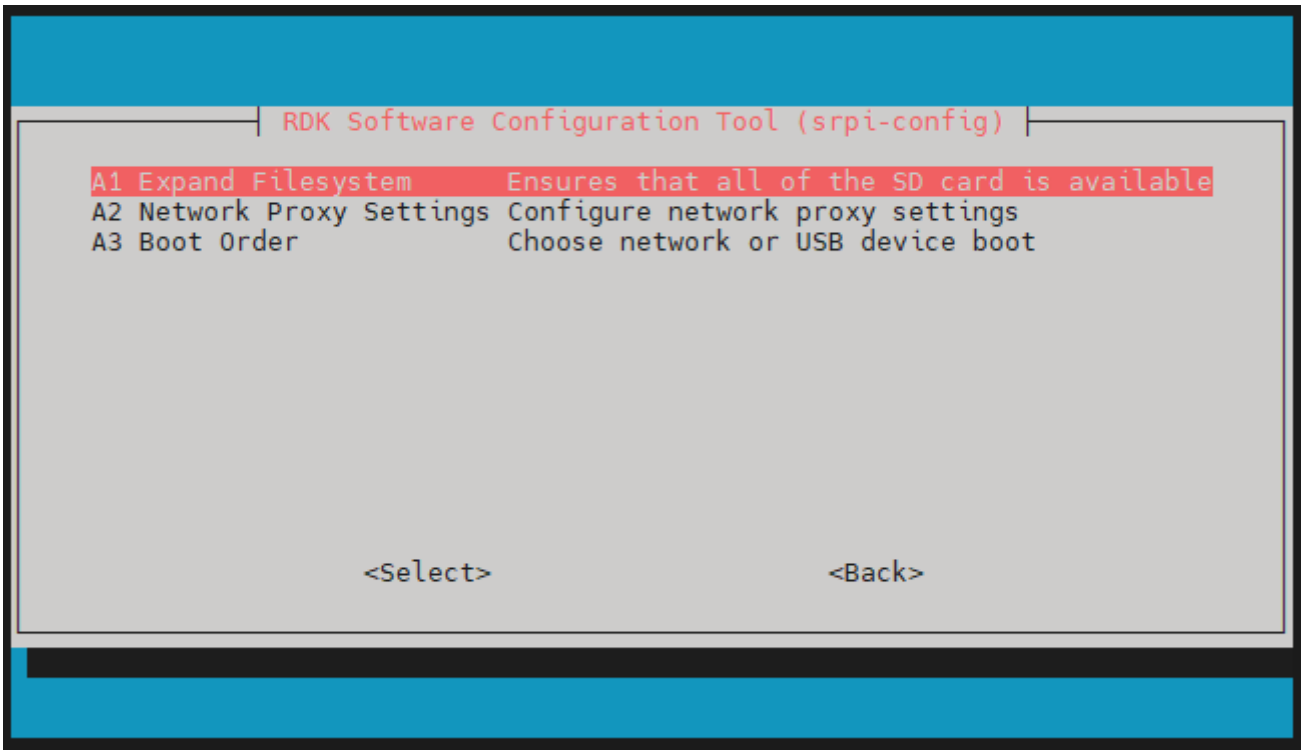
  Select your local time zone, starting from the region, such as Asia, then select a city, such as Shanghai. Type a letter to jump down the list to that point in the alphabet.

- **Keyboard**

  It takes a long time to display reading all keyboard types. Changes usually take effect immediately, but may require a restart.

Advanced Options

Advanced options can be used to set options such as disk expansion and network proxy.



- **Expand Filesystem**

  This option will expand the installation to fill the entire TF card, providing more space for the file system.

> If the system is copied from another system that has already completed the initialization configuration, the capacity of the system's TF card may not be automatically adjusted to the capacity of the current TF card. Using this feature can conveniently complete the expansion.

- **Network Proxy Settings**

  Configure network proxy settings.

- **Boot Order**

  RDK X3 Module supports booting the system from two modes: eMMC and SD card. This option is used to switch the boot mode.

Update

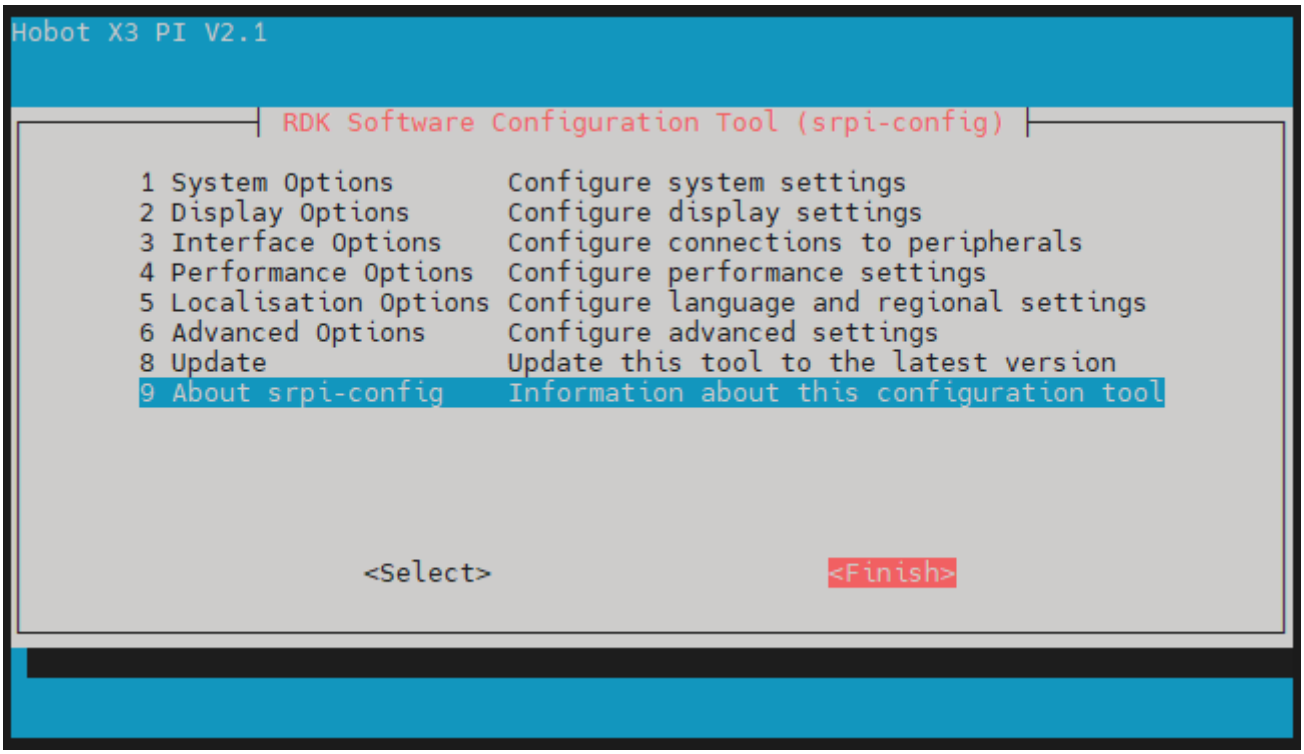
Update the `srpi-config` tool to the latest version.

About srpi-config

About `srpi-config` information

Selecting this option will display the following information:

Finish Button



After completing the changes, select the `Finish` button. The system will ask if you want to restart. When using it for the first time, it is best to restart.

For RDK X5H Module, FCC ID: 2BGUG-RDKX5HM, 5GHz Band（W52): Indoor use only

Integration instructions for host product manufacturers according to KDB 996369 D03 OEM Manual v01r01

2.2 List of applicable FCC ruler
CFR 47 FCC PART 15 SUBPART C&E has been investigated. It is applicable to the modular.

2.3 Specific operational use conditions
This module is stand-alone modular. If the end product will involve the Multiple simultaneously transmitting condition or different operational conditions for a stand-alone modular transmitter in a host, host manufacturer have to consult with module manufacturer for the installation method in end system.

2.4 Limited module procedures
Not applicable

2.5 Trace antenna designs
Not applicable

2.6 RF exposure considerations
To maintain compliance with FCC's RF Exposure guidelines, this equipment should be installed and operated with minimum distance of 20cm from your body.

2.7 Antennas
This radio transmitter FCC ID: 2BGUG-RDKX5HM has been approved by Federal Communications Commission to operate with the antenna types listed below, with the maximum permissible gain indicated. Antenna types not included in this list that have a gain greater than the maximum gain indicated for any type listed are strictly prohibited for use with this device.

maximum Gain / Antenna type	2.4G	5.2G	5.8G
Dipole antenna	2.62dBi	2.72dBi	2.60dBi

2.8 Label and compliance information
The final end product must be labeled in a visible area with the following " Contains FCC ID: 2BGUG-RDKX5HM" .

2.9 Information on test modes and additional testing requirements
Host manufacturer is strongly recommended to confirm compliance with FCC requirements for the transmitter when the module is installed in the host.

2.10 Additional testing, Part 15 Subpart B disclaimer
Host manufacturer is responsible for compliance of the host system with module installed with all other applicable requirements for the system such as Part 15 B.

FCC statement

This device complies with part 15 of the FCC Rules. Operation is subject to the following two conditions: (1) This device may not cause harmful interference, and (2) this device must accept any interference received, including interference that may cause undesired operation.
Any changes or modifications not expressly approved by the party responsible for compliance could void the user's authority to operate the equipment.

This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

– Reorient or relocate the receiving antenna.
– Increase the separation between the equipment and receiver.
– Connect the equipment to an outlet on a circuit different from that to which the receiver is connected.
– Consult the dealer or an experienced radio/TV technician for help.