



AIB-2541M Series User Manual

V1.0.0, July 2026

The technical support and operating information provided in this manual are applicable to the following products



Authorized: Weikai Wang

Edited: Anna Hunag

Warranty

All products manufactured by ICP DAS Co., Ltd. are warranted to the original purchaser against defective materials for a period of one year from the date of delivery.

Disclaimer

ICP DAS Co., Ltd. assumes no legal liability for any damages resulting from the use of this product. The company reserves the right to change or modify the contents of this document at any time without prior notice. The information contained herein is subject to change without notice. The company endeavors to provide accurate and reliable information; however, it does not guarantee the use of this information or any use by third parties that may infringe upon patents or rights. ICP DAS assumes no legal responsibility for any technical or editorial errors or omissions contained herein.

Copyright

Copyright © 2026 ICP DAS Co., Ltd. All Rights Reserved.

Trademark Acknowledgements

All company names, trademarks, and product names mentioned in this document are the property of their respective owners.

Technical Support

If you have any questions, please contact our customer service department. We will provide assistance as soon as possible.

E-mail: service@icpdas.com

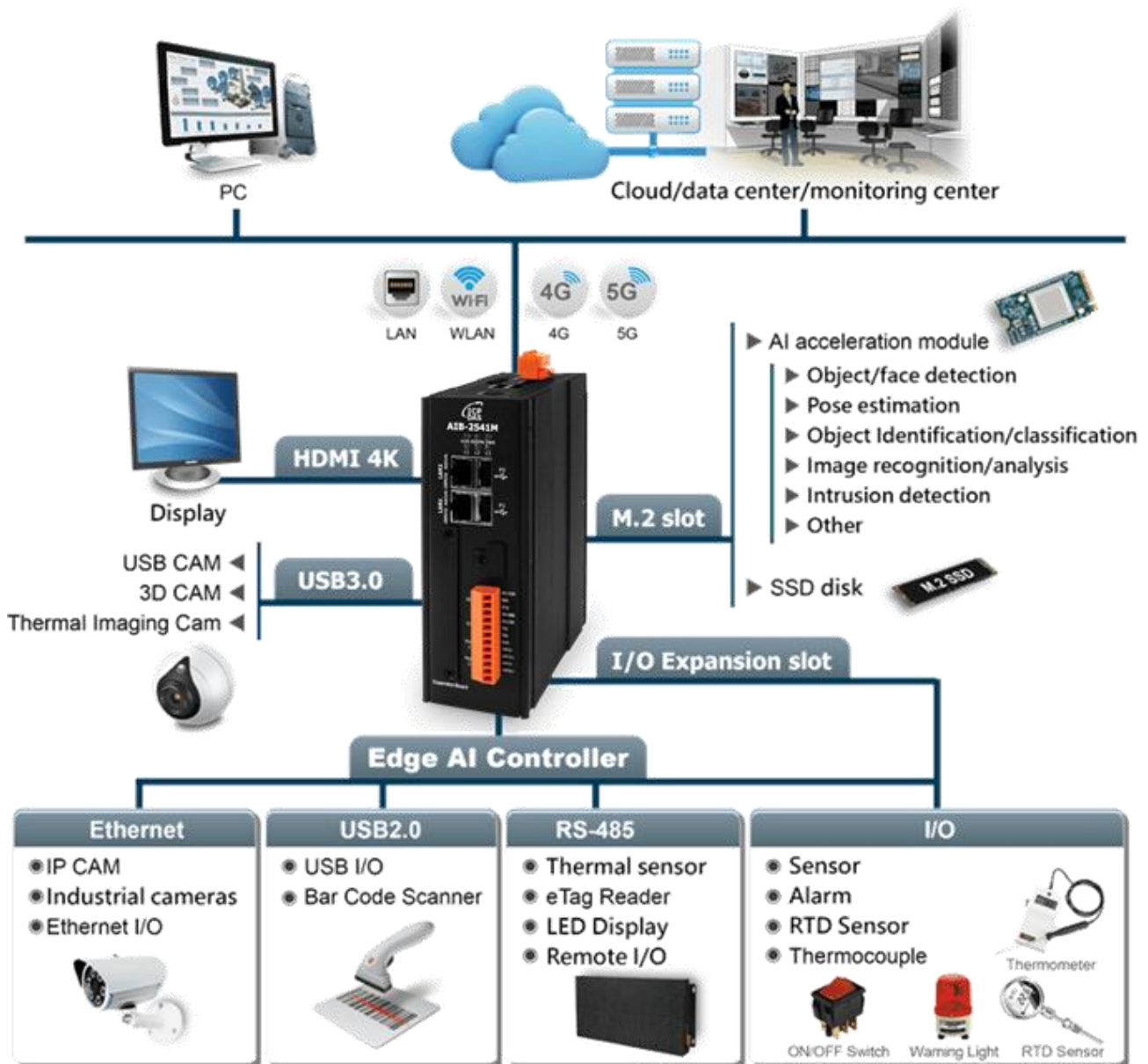
Contents

Contents.....	3
1. Introduction	5
1.1. Features.....	7
1.2. Specifications.....	12
1.3. Overview	15
1.4. Dimensions	22
2. Quick Start.....	23
2.1. Installation and Startup	24
2.2. Expansion Accessory Installation	26
2.2.1. Installing an AI Module/SSD	26
2.2.2. Installing an XV-Board/XB-Board	27
2.2.3. Inserting a SIM Card	30
2.2.4. Installing the Antenna.....	31
2.3. Changing the User Interface Language	32
2.4. IP Configuration	35
3. Tools and Utilities	37
3.1. System tools	37
3.1.1. Changing the User Interface Language	38
3.1.2. IP Configuration	40
3.1.3. Display Resolution Settings.....	42
3.1.4. User Account Management	45
3.2. Software Tools	49
3.2.1. Utility – Viewing System Information	49
3.2.2. Text Editor.....	52
3.2.3. Performance Monitor	53
3.2.4. Automatic Time Synchronization	54
3.2.5. CuteCom	56
3.3. Other Support Tools.....	57
3.3.1. Remote Connection Software	57
3.3.2. SSH File Transfer	60
4. Application Development.....	62
4.1. Development Environment Preparation	63
4.2. Creating Your First Python Application	64
4.2.1. Creating a New Project	64
4.2.2. Adding Control Widgets to the Window	69
4.2.3. Adding Events to the Control Widgets	71

4.2.4. Running the Application	73
4.2.5. Python Examples	74
5. Selecting I/O Expansion Modules and SDKs	75
6. System Update and Upgrade	78
Appendix A. Practical Usage Notes.....	86
A.1. How to Check the Kernel Version	86
A.2. How to Check the HailoRT Version	87
A.3. How to Configure Automatic Network Time Synchronization	88
A.4. How to Replace the RTC Battery.....	90
Appendix B. Hailo-8L Development and Application	91
B.1. Environment Setup.....	93
B.1.1. Runtime Environment Setup: HailoRT and Hailo PCIe Driver Installation	93
B.1.2. PC Environment Setup: CUDA and Deep Learning Framework Installation	95
B.2. AI Model Training and Performance Evaluation	100
B.2.1. Model Training.....	100
B.2.2. Model Performance Evaluation	102
B.3. Model Conversion	104
B.3.1. Installing the Model Conversion Tool and Converting Models for Hailo-8L	104
B.3.2. Installing the Hailo Dataflow Compiler	105
B.3.3. Converting Models for Hailo-8L.....	106
B.4. AI Model Application Development and Usage	107
Appendix C. XV-Board and XB-Board Series I/O Expansion Cards	109
Appendix D. 4G Communication	111
Appendix E. Revision	114

1. Introduction

This chapter provides an overview of the AIB-2541M series modules and their components, introducing the basic concepts to help users become familiar with the AIB-2541M.



The AIB-2541M series is a fanless, secure, compact, and low-power Edge AI Controller equipped with an ARM Cortex-A76 processor. It provides an M.2 expansion slot for various AI accelerator modules, enabling scalable, high-performance, and energy-efficient AI computing at the edge for industrial applications. For I/O expansion, optional high-protection XV-Board/XB-Board modules are available. In addition, the system provides a microSD card slot and an M.2 Key M slot (for NVMe SSD expansion), allowing users to store applications, images, and data while conveniently expanding storage capacity.

The AIB-2541M Edge AI Controller can be equipped with peripheral interfaces including Gigabit Ethernet, HDMI, USB 3.0, digital I/O, and 4G wireless communications. Its rugged metal enclosure and thermal design make it suitable for operation in environments ranging from -20°C to +70°C.

1.1. Features

The AIB-2541M series provides comprehensive system configuration capabilities and convenient remote system update services. It is suitable for a wide range of applications and can meet diverse user requirements. The following sections describe the major features of the AIB-2541M.

Ubuntu 24.04 (Default OS)

Ubuntu is an easy-to-use operating system suitable for both beginners and experienced users. It provides an intuitive user interface and a rich set of features for a wide range of applications, making it one of the most widely used Linux distributions today. When deployed on the AIB-2541M, Ubuntu fully utilizes the platform's capabilities and performance. Ubuntu 24.04 is the default operating system for the AIB-2541M.



- Excellent user experience
- Ease of use, versatility, and security
- Extensive community support
- Supports embedded services such as web, FTP, and SSH servers
- Rich software support and development tools, including dedicated SDKs, Perl, Java, GUI software, and more

Raspberry Pi OS



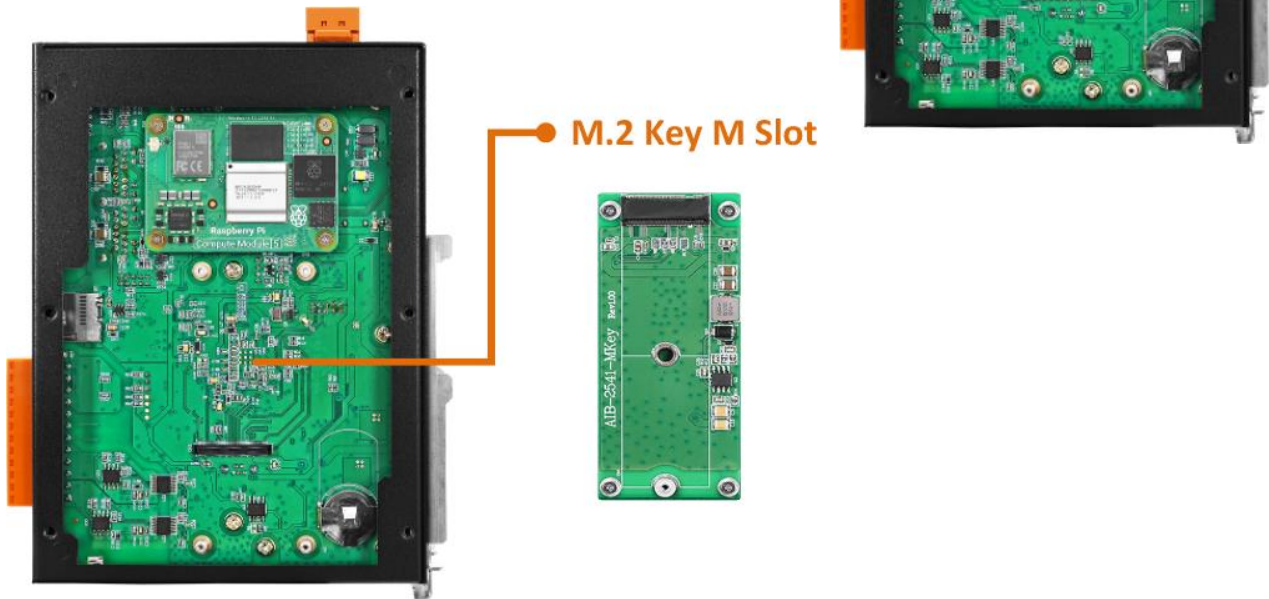
The AIB-2541M is pre-installed with Ubuntu 24.04. If required, users may also install Raspberry Pi OS. The Edge AI Controller supports Raspberry Pi OS (64-bit). Raspberry Pi OS is based on Debian Linux and supports command-line operation, making it suitable for users familiar with Linux environments. It is also user-friendly for Windows users by providing a simplified desktop interface with wallpapers, taskbars, shortcuts, and application menus similar to Windows systems.

- Extensive Raspberry Pi software ecosystem
- More than 35,000 software packages available for easy installation
- Pre-installed applications including Python 3 and a Python IDE
- Numerous ready-to-use applications available online

AI Accelerator Modules

- M.2 Key M slot supporting AI accelerator modules or NVMe SSDs

Flexible support for various AI accelerator modules through the M.2 Key M interface



Hardware Security Key Protection

The AIB-2541M provides a 64-bit hardware security key for software copy protection.



Multiple Storage Expansion Interfaces

The AIB-2541M supports multiple reliable storage devices, allowing users to select a suitable storage solution based on their application requirements.

- One microSD card slot
- One M.2 Key M slot for M.2 NVMe SSD expansion (**optional**)

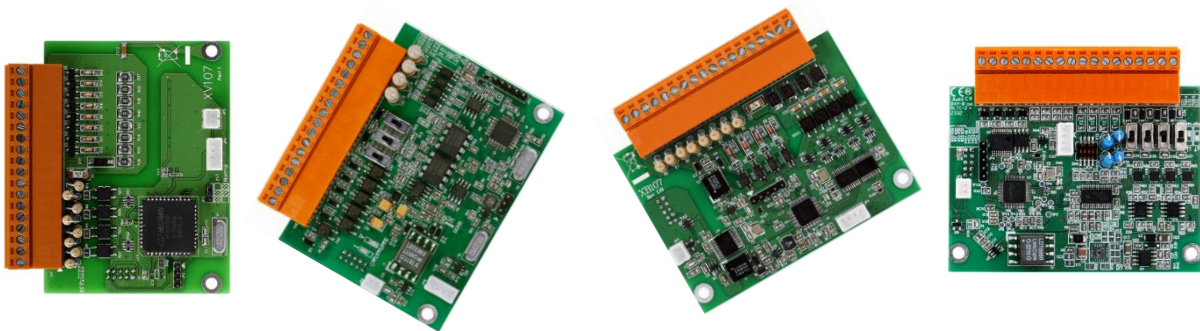


Recommended SSD Models

- Innodisk M.2 (P80) 3TE4
<https://www.innodisk.com/tw/products/flash-storage/m2-pcie/m2-p80-3te4-b-m-key>
- APacer PV220-M280
https://www.apacer.com/en/product/industrial-product/detail/industrial_ssd/pv220-m280

Since SSD compatibility may vary across different brands, the two industrial-grade SSD models listed above have been verified for stable operation. Other SSD models have not been validated and compatibility cannot be guaranteed. Users may use other SSDs, but ICP DAS only validates the models listed above.

Support for Multiple I/O Expansion Modules



The AIB-2541M is equipped with an I/O expansion bus that supports optional XV-Board or XB-Board expansion modules. These modules can be used to implement various I/O functions such as DI, DO, A/D, D/A, Timer/Counter functions, and communication interfaces including RS-232/422/485.

- For more information about XV-Board I/O modules:
https://www.icpdas.com/tw/product/guide+Remote_I_O_Module_and_Unit+PAC_I_O_Modules+XV-board
- For more information about XB-Board I/O modules:
https://www.icpdas.com/tw/product/guide+Remote_I_O_Module_and_Unit+PAC_I_O_Modules+XB-board

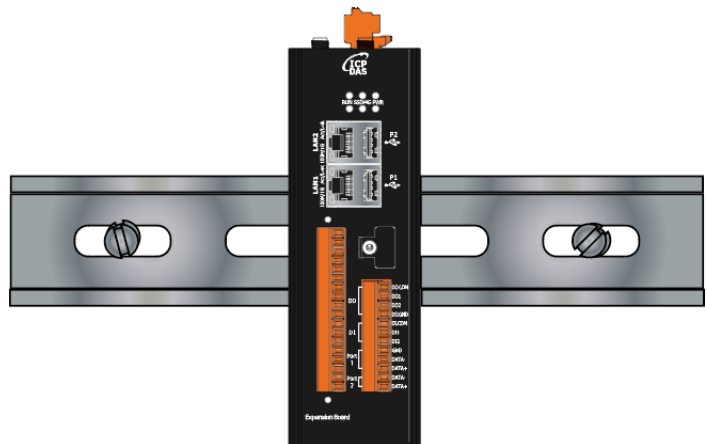
The AIB-2541M is equipped with multiple RS-485 serial communication ports, dual Ethernet ports, and USB interfaces for I/O expansion and external device connectivity.



- For more information about Ethernet I/O modules:
<https://www.icpdas.com/en/product/p02.php?root=537&kind=539>
- For more information about USB I/O modules:
<https://www.icpdas.com/en/product/p02.php?root=537&kind=541>
- For more information about RS-485 I/O modules:
<https://www.icpdas.com/en/product/p02.php?root=537&kind=538>

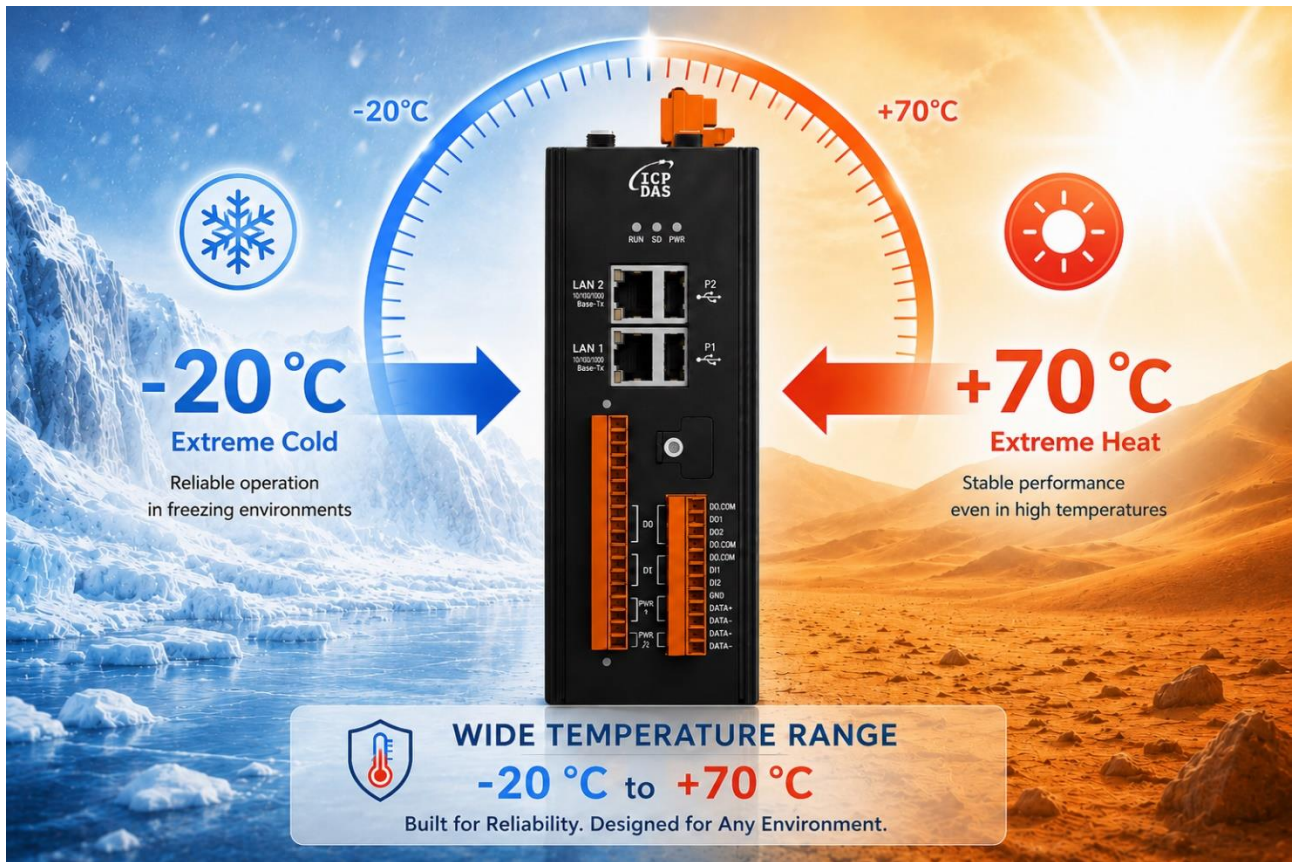
The AIB-2541M is housed in a rugged metal enclosure.

The AIB-2541M features a compact and lightweight design and supports DIN-Rail mounting, making it suitable for installation in limited-space environments.



Reliable Operation in Harsh Environments

The AIB-2541M is designed to operate reliably under extreme temperature and humidity conditions.



- Operating temperature: -20 °C to +70 °C
- Storage temperature: -40 °C to +80 °C
- Relative humidity: 10% to 90% RH, non-condensing

1.2. Specifications

The following table lists the specifications of the AIB-2541M series modules.

System Software	
OS	Ubuntu 24.04 (Default) / Raspberry Pi OS (64-bit)
SDK	C / Python libraries
Language Support	English, Simplified Chinese, Traditional Chinese
CPU Module	
CPU	Quad-core ARM Cortex-A76 (ARMv8) 64-bit SoC @ 2.4GHz
GPU	VideoCore VII (OpenGL ES 3.0, H.265 HEVC, 4K 60fps decode)
SDRAM	4GB LPDDR4 SDRAM
Storage	Built-in 64GB eMMC
Other Peripherals	
Additional Storage	1 x M.2 Key M connector (PCIe 2.0 interface) (optional), supports PCIe SSD 2230/2242/2260/2280 installation. M.2 SSD card is not included.
FRAM	128KB
RTC	Provides second, minute, hour, day, week, month, and year information
Watchdog Timer	Supported
LED Indicators	1 x Power LED indicator, 1 x System RUN LED indicator, and 3 x programmable LED indicators
AI Accelerator Module	
M.2 Key M connector	1 x M.2 Key M connector (PCIe 2.0 interface). (optional) Supports optional Hailo-8L M.2 AI accelerator module, 2230/2242/2260/2280. Frameworks: TensorFlow/TensorFlow Lite, ONNX, Keras, and PyTorch.
M.2 Key B connector	1 x M.2 Key B connector (USB 3.0 interface). (optional) Supports optional Kneron M.2 AI accelerator module, 2230/2242/2260/2280. Frameworks: TensorFlow/TensorFlow Lite, Caffe, Keras, and PyTorch.

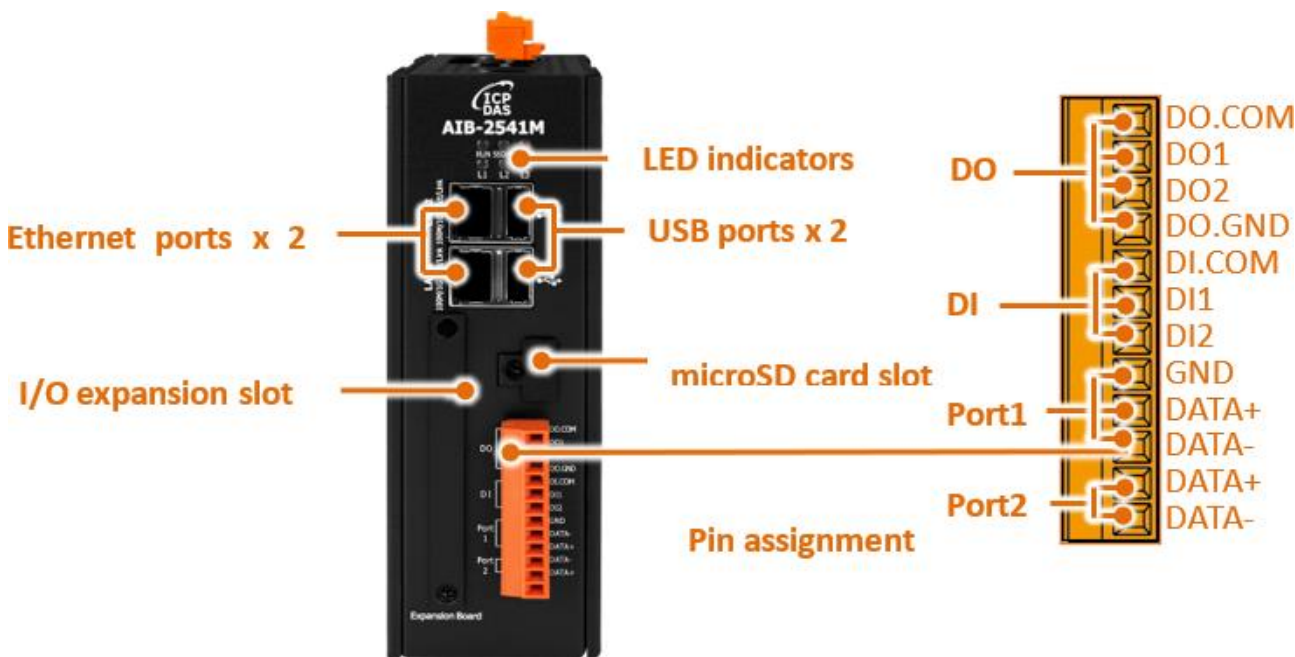
Display and Communication Interfaces	
Display	1 x HDMI port, supporting 4K 60fps with audio
Ethernet	2 × RJ-45, 10/100/1000 Base-TX, with auto-negotiation LED indicators and auto MDI/MDI-X LED indicators
USB 2.0 (host)	2
USB 3.0 (host)	2
Serial Port 1	RS-485 (Data+, Data-) with 2500 VDC isolation
Serial Port 2	RS-485 (Data+, Data-) with 2500 VDC isolation
Wireless Communication	1 x expansion connector for an optional 4G/LoRa/NB-IoT module (USB 2.0 interface) and 1 optional Nano SIM card slot
Digital Input	
Channels	2
Type	Wet contact
Sink/Source (NPN/PNP)	Sink/Source
ON Voltage Level	Wet contact: +19 VDC to +24 VDC
OFF Voltage Level	Wet contact: +11 VDC maximum
Digital Output	
Channels	2
Type	Isolated open collector
Load Voltage	+24 VDC
Load Current	100 mA/channel
Overvoltage Protection	60 VDC
Overload Protection	1.1 A
Short-circuit Protection	Yes
I/O Expansion	
I/O Type	XV-Board/XB-Board (optional)
Power	
Power Input	+12 ~ +48 VDC
Power Consumption	Typical 4.8 W (0.2 A @ 24 VDC)

Mechanical	
Enclosure	Metal
Dimensions (mm)	56 mm x 177 mm x 128 mm
Installation	Standard DIN-Rail mounting / wall mounting (optional)
Environment	
Operating Temperature	-20 ~ +70 °C
Storage Temperature	-40 ~ +80 °C
Humidity	10 to 90% RH, non-condensing

1.3. Overview

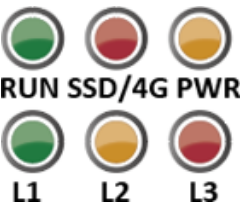
The AIB-2541M series modules are equipped with multiple communication interfaces. External devices can be connected and integrated through Ethernet, USB, RS-232/RS-485, and other interfaces. The following sections provide an overview and brief description of the external components and communication interfaces of the AIB-2541M.

Front View



LED Indicators

The AIB-2541M provides 6 LED indicators:



LED Indicator	Color	Description
RUN	Green	System running
PWR	Red	Power supply is normal
4G/SSD	Orange	4G communication activity / SSD activity
L1	Green	User-programmable control
L2	Orange	
L3	Red	

Ethernet Ports



The AIB-2541M supports 10/100/1000 Base-TX Ethernet with auto-negotiation LED indicators and auto MDI/MDI-X LED indicators. The current network speed can be identified by the LED status of the Ethernet port.

LED Indicator	Status (Color)	Description
1G	ON (orange)	1 Gbps connection
	OFF	10/100 Mbps connection
Link/Act	ON (green)	Network communication link established
	OFF	No network communication link established
	Blinking (green)	Data packet transmission in progress

USB Ports



The AIB-2541M provides two USB 2.0 ports for connecting USB devices, such as a mouse, keyboard, or other USB peripherals.

microSD Card Slot

The microSD card slot allows users to use a microSD memory card for system recovery and storage expansion. microSD card expansion supports up to 32 GB.

XV-Board / XB-Board Slot

After an XV-Board or XB-Board I/O expansion module is installed, this is the location of the connection terminal.

For installation procedures, refer to Section 2.2.2, “Installing the XV-Board/XB-Board.”

Pin Assignment

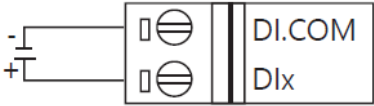
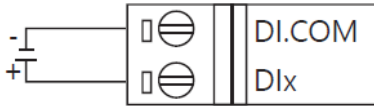
Pin	Signal	Description
1	DO.PWR	Digital Output x 2
2	DO0	
3	DO1	
4	DO.GND	
5	DI.COM	Digital Input x 2
6	DI0	
7	DI1	
8	RS-485.GND	Serial Port 2 (RS-485)
9	D2-	
10	D2+	
11	D1-	Serial Port 1 (RS-485)
12	D1+	

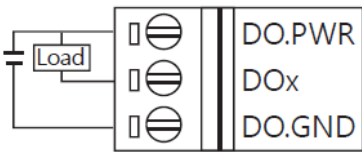
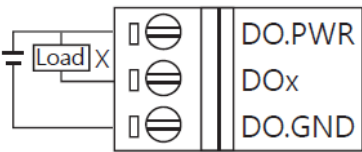
Note



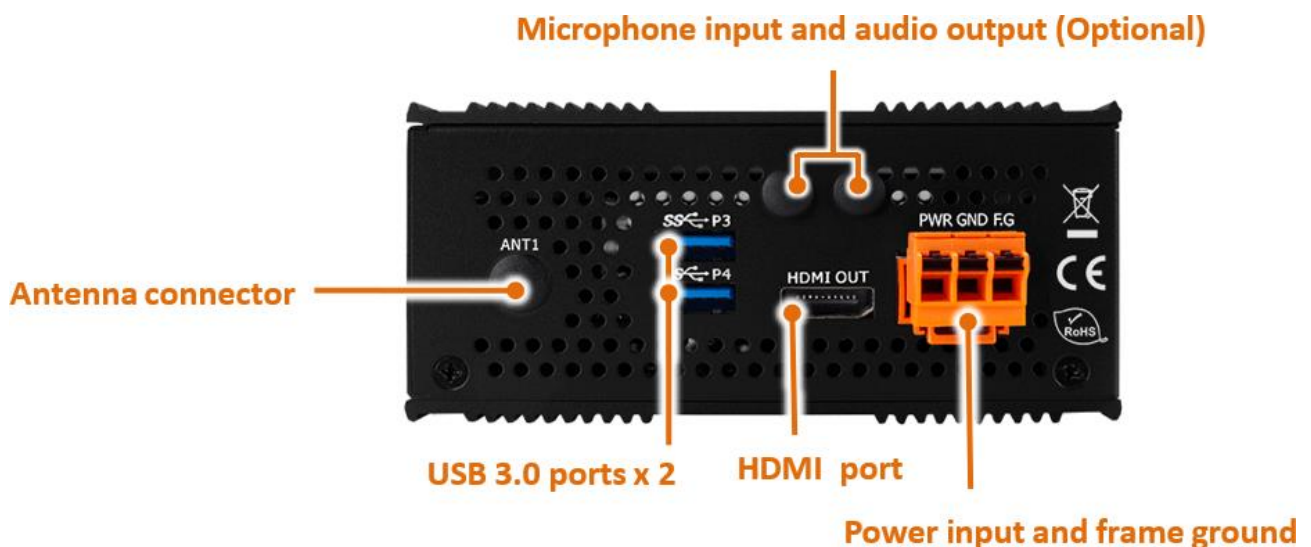
Serial Port 1 is identified as /dev/ttyAMA2 in Ubuntu.
Serial Port 2 is identified as /dev/ttyAMA3 in Ubuntu.

Wire Connections

Digital Input	ON State Readback as 1	OFF State Readback as 0
Sink	+19 ~ 24 VDC	+11 VDC
		
Source	+19 ~ 24 VDC	+11 VDC
		

Digital Output	ON State Readback as 1	OFF State Readback as 0
Sink	24 VDC	24 VDC
		

Top View



HDMI Port

HDMI output port x 1 (Audio Supported)

Maximum Resolution: 4K (3840 × 2160) @ 60 Hz

USB 3.0 Ports

The AIB-2541M series modules provide two USB 3.0 ports for connecting high-speed USB devices.

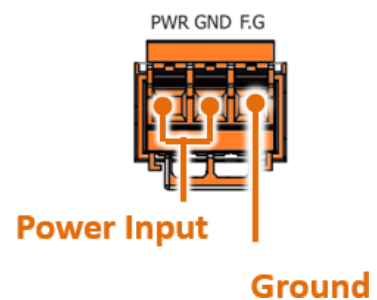
Antenna

One antenna connector is provided for use with the 4G module. (Optional)

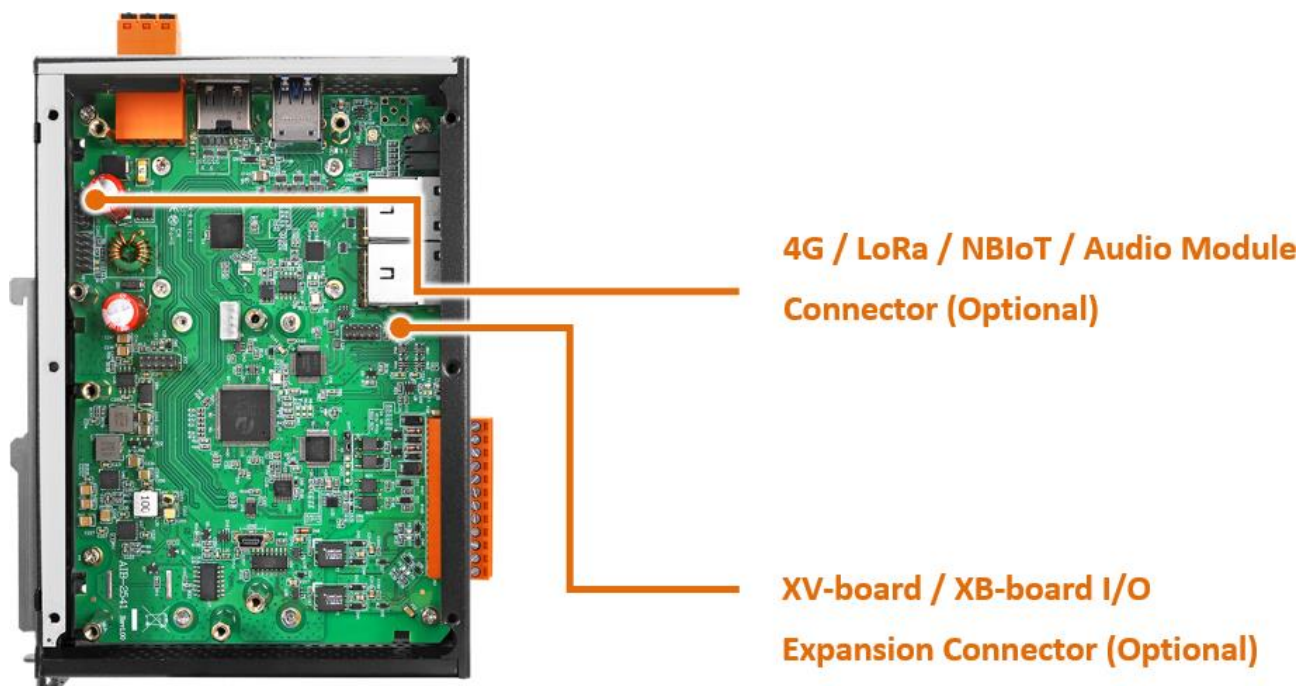
Microphone Input and Audio Output (Optional)

Power Input and Ground

The AIB-2541M provides a 3-pin connector. Two pins are used for power input, and one pin is used for frame ground (F.G.).



Left View



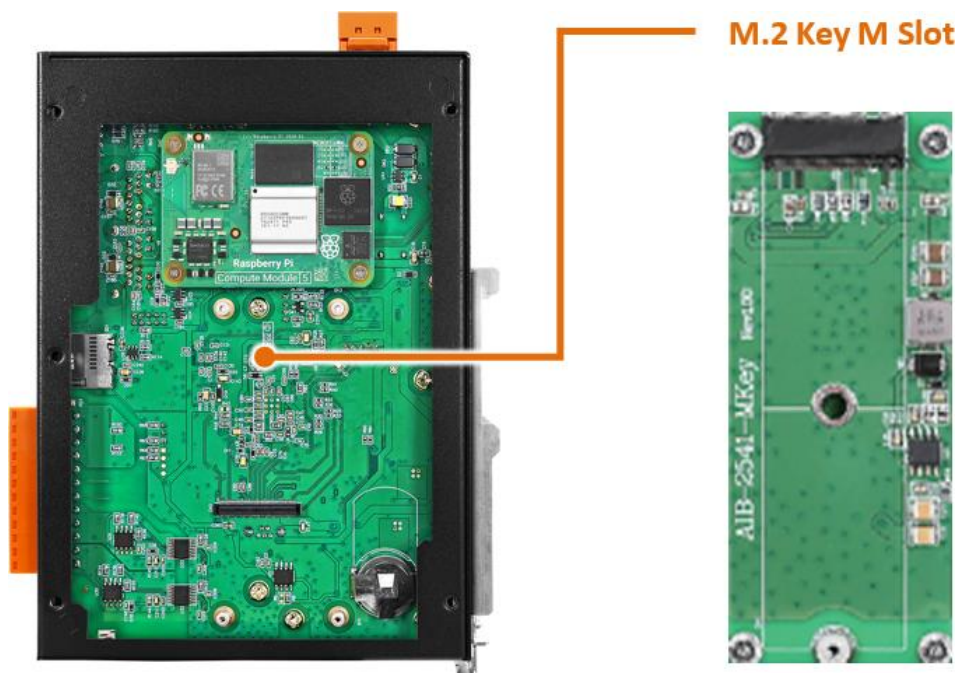
4G / LoRa / NB-IoT / Audio Module Connector (Optional)

This connector is used to install an optional 4G/LoRa/NB-IoT module via a USB 2.0 interface and includes one optional nano-SIM card slot.

XV-Board / XB-Board I/O Expansion Connector

The I/O expansion slot is used for XV-Board/XB-Board installation. For installation procedures, refer to Section 2.2.2, “Installing the XV-Board/XB-Board.”

Right View



M.2 Key M Slot (Optional)

The AIB-2541M provides one M.2 Key M connector with PCIe x1 2.0 interface. This connector is used to install M.2 Key M AI accelerator modules or M.2 NVMe SSDs in 2230/2242/2260/2280 form factors. M.2 SSD cards are not included.



Recommended SSD Models

Because there are many SSD brands available, the following two industrial-grade SSD models have been tested and verified for normal operation:

Innodisk M.2 (P80) 3TE4

<https://www.innodisk.com/tw/products/flash-storage/m2-pcie/m2-p80-3te4-b-m-key>

APacer PV220-M280

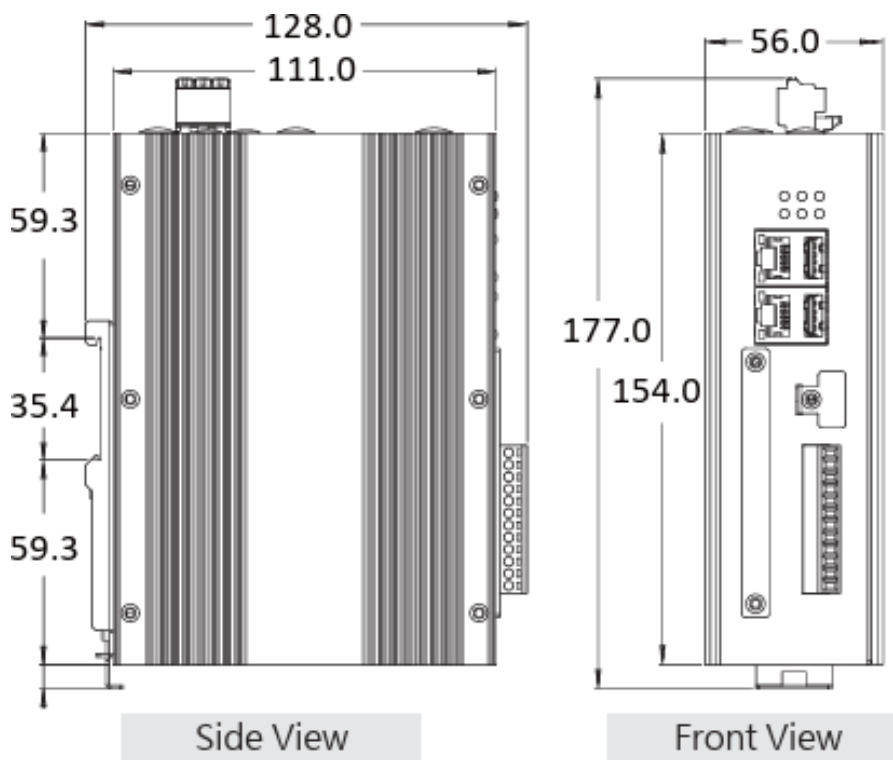
https://www.apacer.com/en/product/industrial-product/detail/industrial_ssd/pv220-m280

Other SSD brands and models have not been validated and may have compatibility issues.

1.4. Dimensions

The following figure shows the dimensions of the AIB-2541M series modules. When installing the module inside an enclosure, reserve sufficient space for heat dissipation.

All dimensions are in millimeters.



2. Quick Start

This chapter describes the installation and configuration procedures for the AIB-2541M. It provides step-by-step guidance for first-time users to understand how to install, configure, and operate the AIB-2541M.

Before using the AIB-2541M, please check whether the package includes the following items. If any item is missing or damaged during shipment, please contact your distributor immediately.



AIB-2541M



European terminal blocks * 16



Screwdriver

Optional cellular communication expansion module accessories



**GSM/GPRS antenna
(ANT-421-04)**



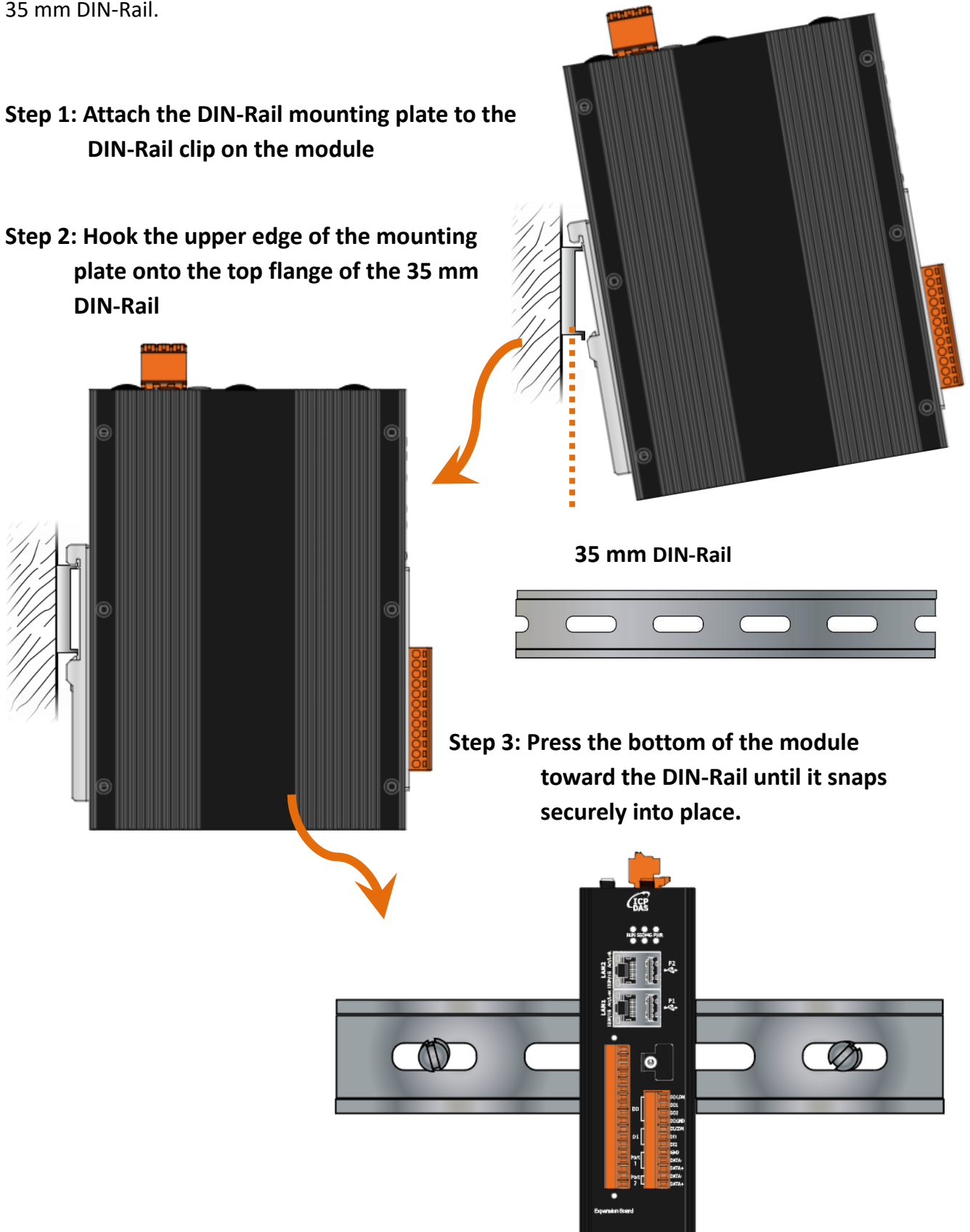
**GPS active antenna
(ANT-115-03)**

2.1. Installation and Startup

The AIB-2541M series module is equipped with a DIN-Rail clip for secure installation on a standard 35 mm DIN-Rail.

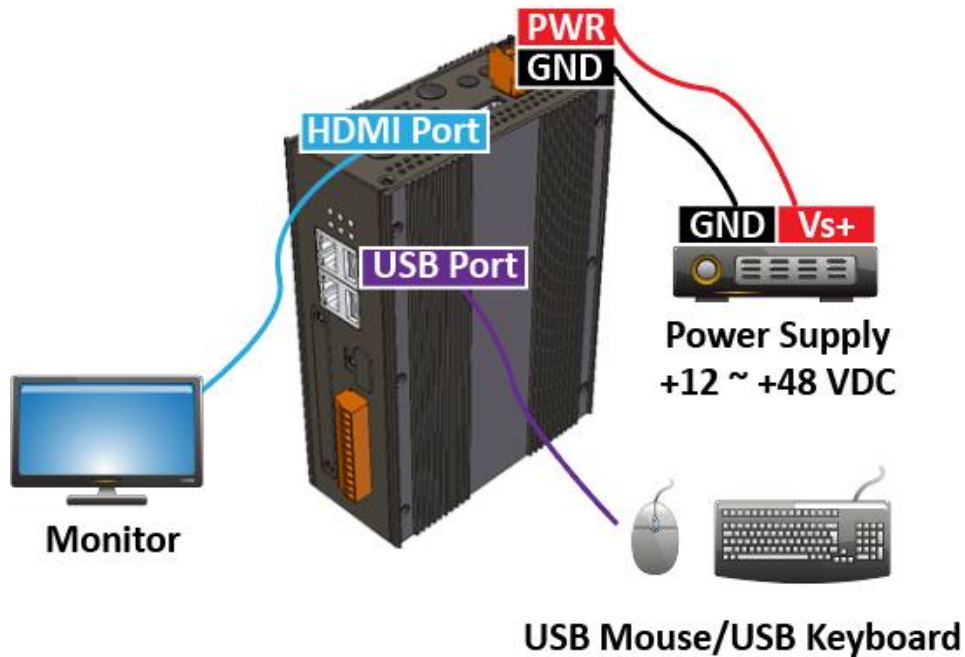
Step 1: Attach the DIN-Rail mounting plate to the DIN-Rail clip on the module

Step 2: Hook the upper edge of the mounting plate onto the top flange of the 35 mm DIN-Rail



Step 4: Connect Power and Peripheral Devices

- i. Connect a USB mouse or USB keyboard to the USB ports.
- ii. Connect a monitor to the HDMI port.
- iii. Connect a +12 VDC to +48 VDC power supply to the PWR and GND terminals

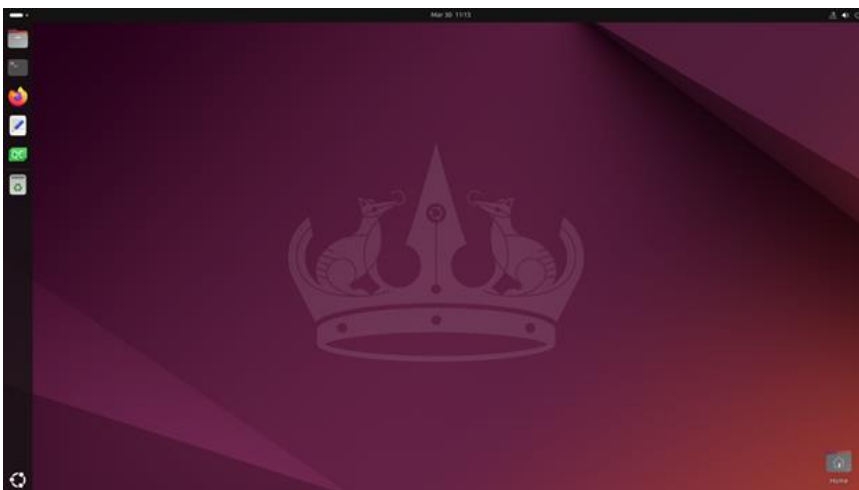


Step 5: After connecting the power, turn on the system. The user interface will appear after startup.

Default user account: pi

Default password: icpdas

If sudo privileges are required, enter the default password.



Note & Warning



For security reasons, please change the default password immediately after first use.

2.2. Expansion Accessory Installation

2.2.1. Installing an AI Module/SSD

Step 1: Use a screwdriver to remove the screws from the right-side cover

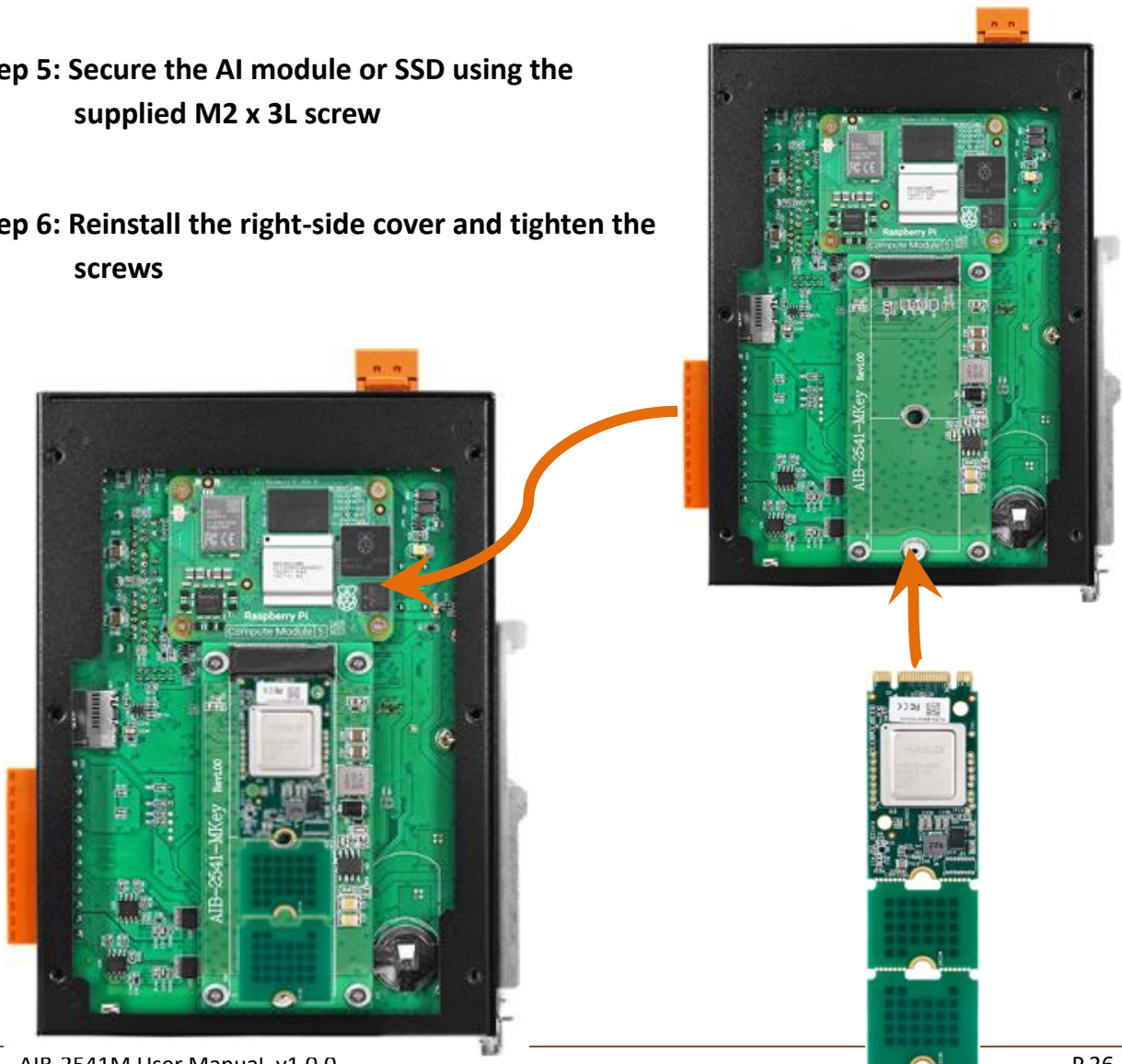
Step 2: Remove the right-side cover

Step 3: Locate the available M.2 slot for the AI module or SSD

Step 4: Insert the AI module or SSD into the M.2 slot

Step 5: Secure the AI module or SSD using the supplied M2 x 3L screw

Step 6: Reinstall the right-side cover and tighten the screws



2.2.2. Installing an XV-Board/XB-Board

The AIB-2541M series modules include an I/O expansion slot that supports optional XV-Board/XB-Board expansion modules.

For XV-Board I/O module specifications (Optional):

<https://www.icpdas.com/tw/product/guide+Remote+I+O+Module+and+Unit+PAC+I+O+Modules+XV-board>

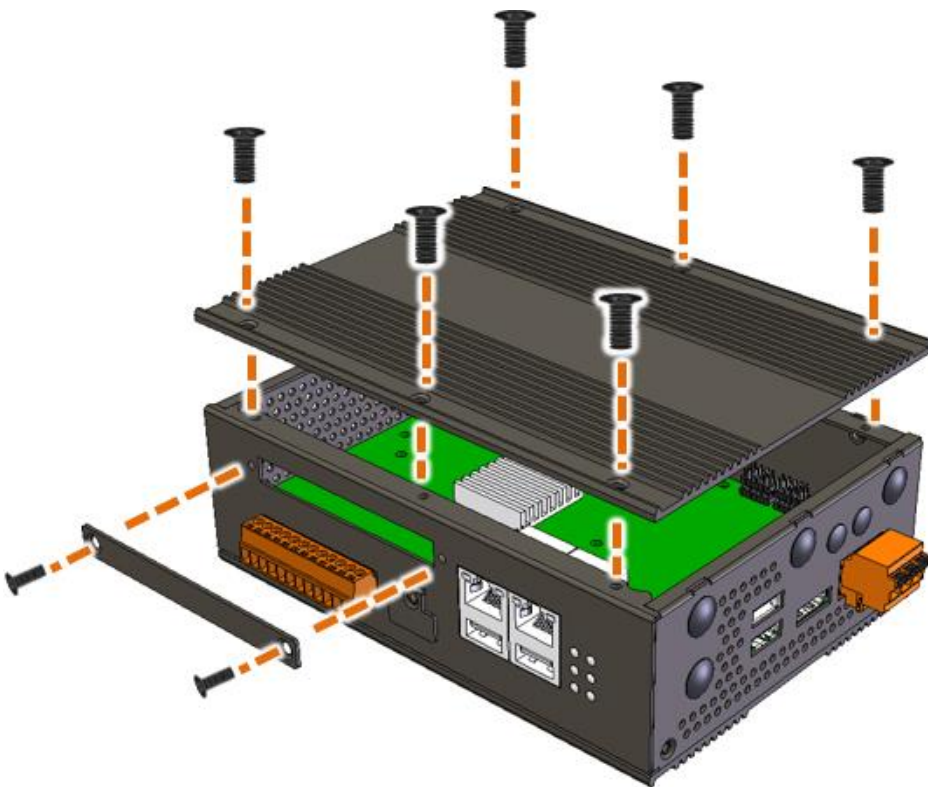
For XB-Board I/O module specifications (Optional):

<https://www.icpdas.com/tw/product/guide+Remote+I+O+Module+and+Unit+PAC+I+O+Modules+XB-board>

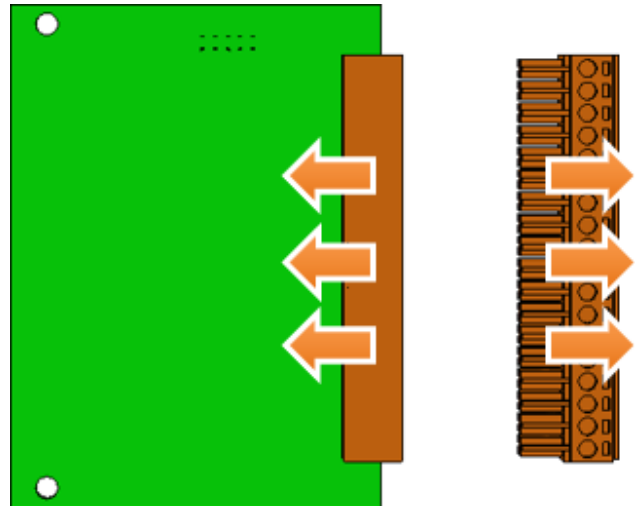
Step 1: Remove the screws from the left-side cover

Step 2: Remove the left-side cover

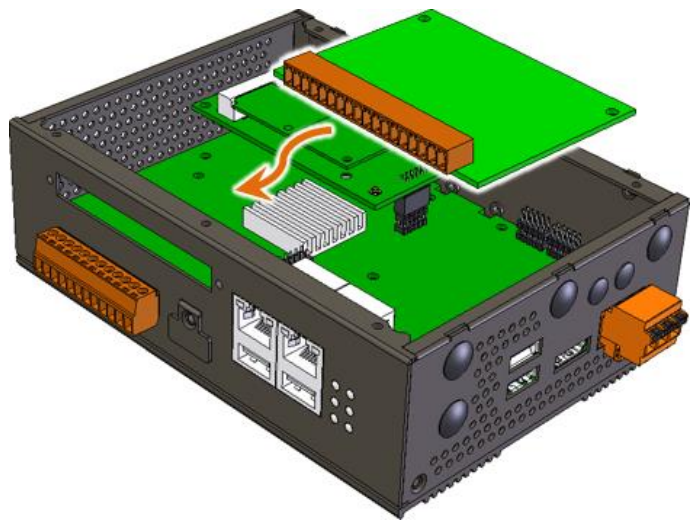
Step 3: Remove the screws from the I/O expansion slot cover on the front panel



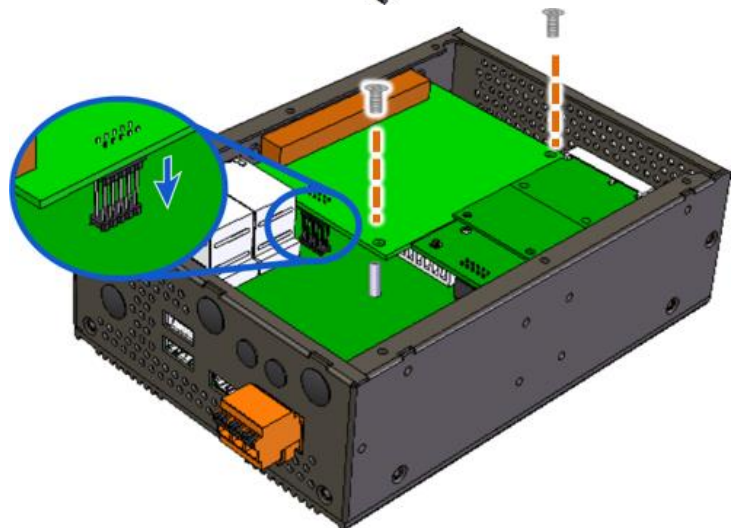
Step 4: Separate the XV-Board/XB-Board from the I/O terminal block



Step 5: Hold the XV-Board/XB-Board vertically and align it carefully with the I/O slot. Gently press the XV-Board/XB-Board into the I/O expansion slot



Step 6: Carefully connect the XV-Board/XB-Board to the I/O expansion connector and secure it using the supplied screws



Notice & Warning



When installing the XV-Board/XB-Board expansion module, ensure that the pin headers on the XV-Board/XB-Board are correctly aligned with the pin sockets on the AIB-2541M PCB before inserting the module.

Step 7: Reinstall the left-side cover and tighten the screw



Step 8: Insert the I/O terminal block

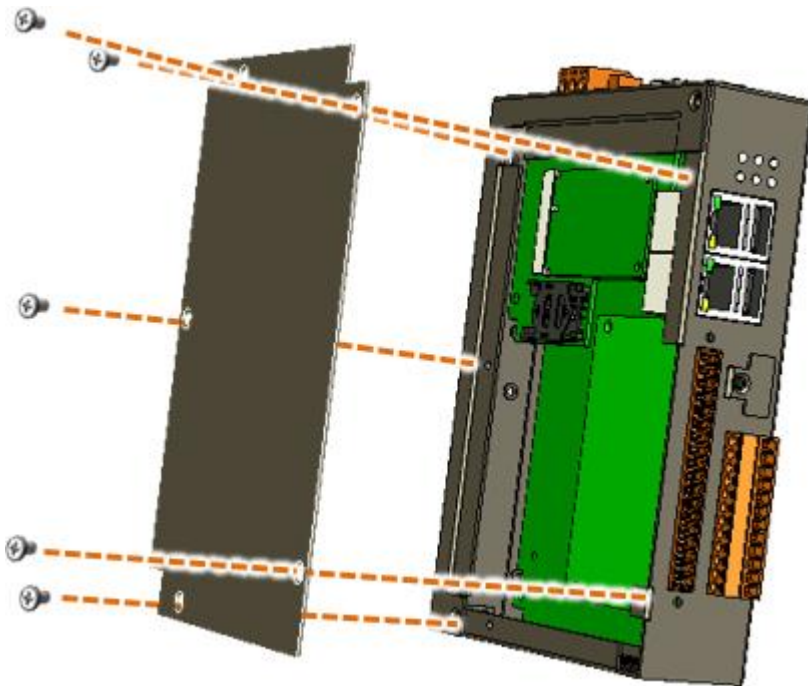


2.2.3. Inserting a SIM Card

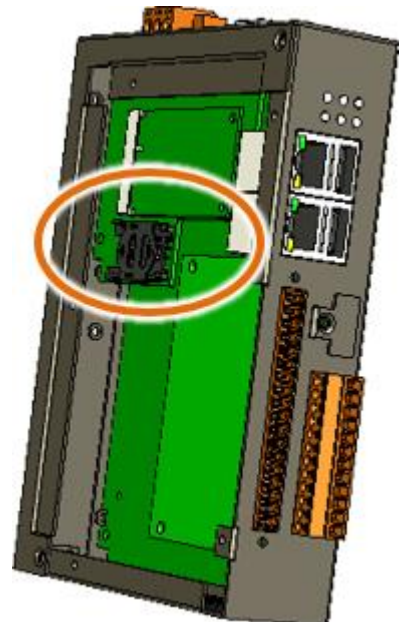
(Only applicable when a cellular communication expansion card is installed.)

The SIM card slot is located on the side of the module.

Step 1: Use a screwdriver to remove the screws on the side cover, then remove the side cover



Step 2: Insert the SIM card into the SIM card slot



Guidance & Notes



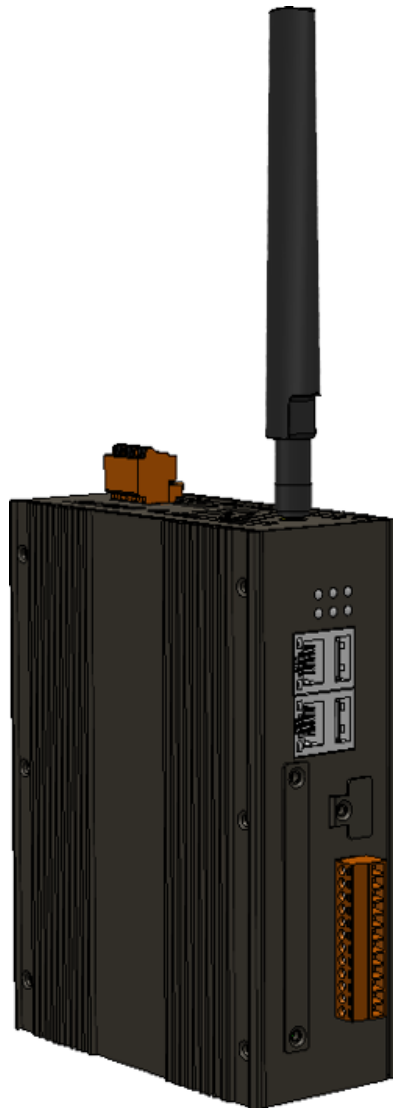
Before installing or removing the SIM card, ensure that the AIB-2541M is powered off.
Do not bend or scratch the SIM card.

2.2.4. Installing the Antenna

(Only applicable when a cellular communication expansion card is installed.)

The cellular communication module provides one antenna connector for connecting a 3G/4G antenna.

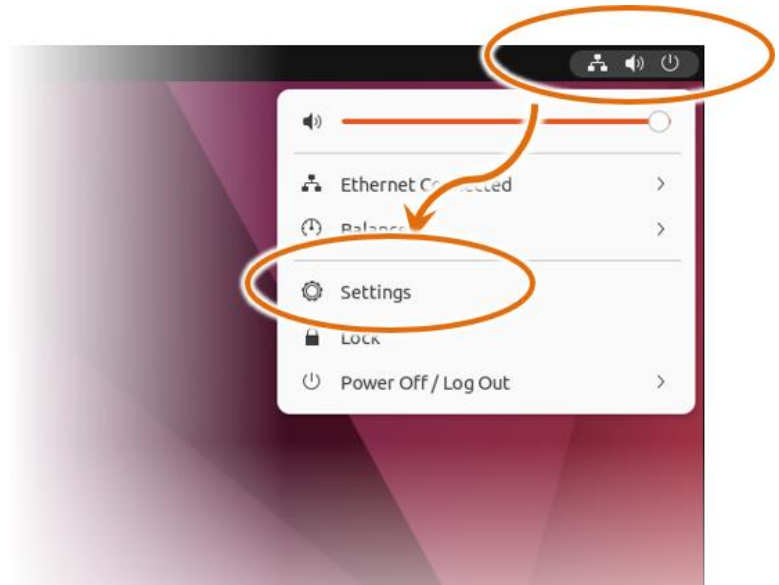
To install the antenna, securely fasten the antenna to the connector and place the antenna in an appropriate location.



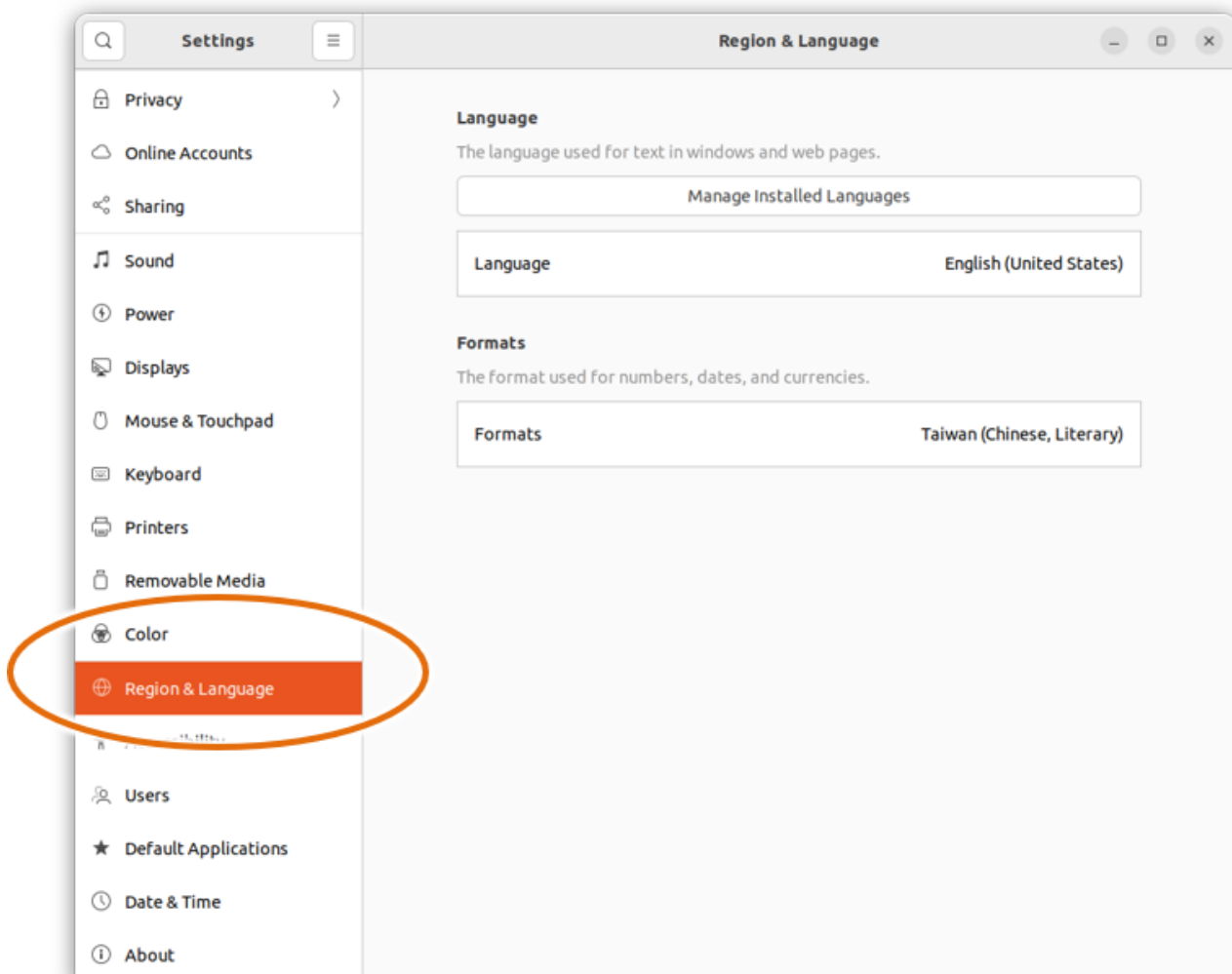
2.3. Changing the User Interface Language

Regional Settings is a built-in Ubuntu function that allows users to easily change the user interface language of the AIB-2541M series modules. Please follow the steps below to change the user interface language.

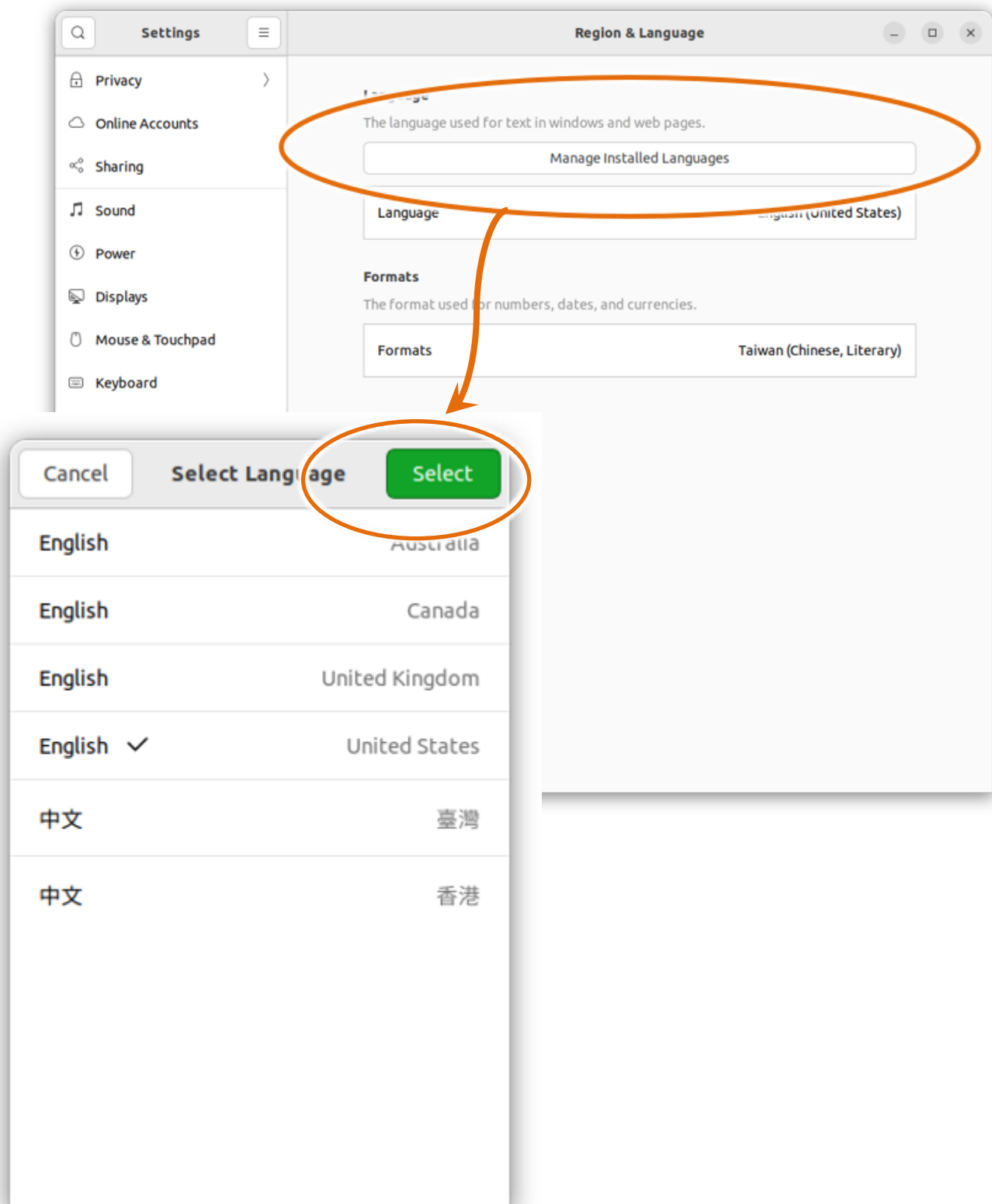
Step 1: Click the icon at the upper-right corner of the desktop, then select Settings from the pop-up menu



Step 2: Click Region & Language, and then select the Language option



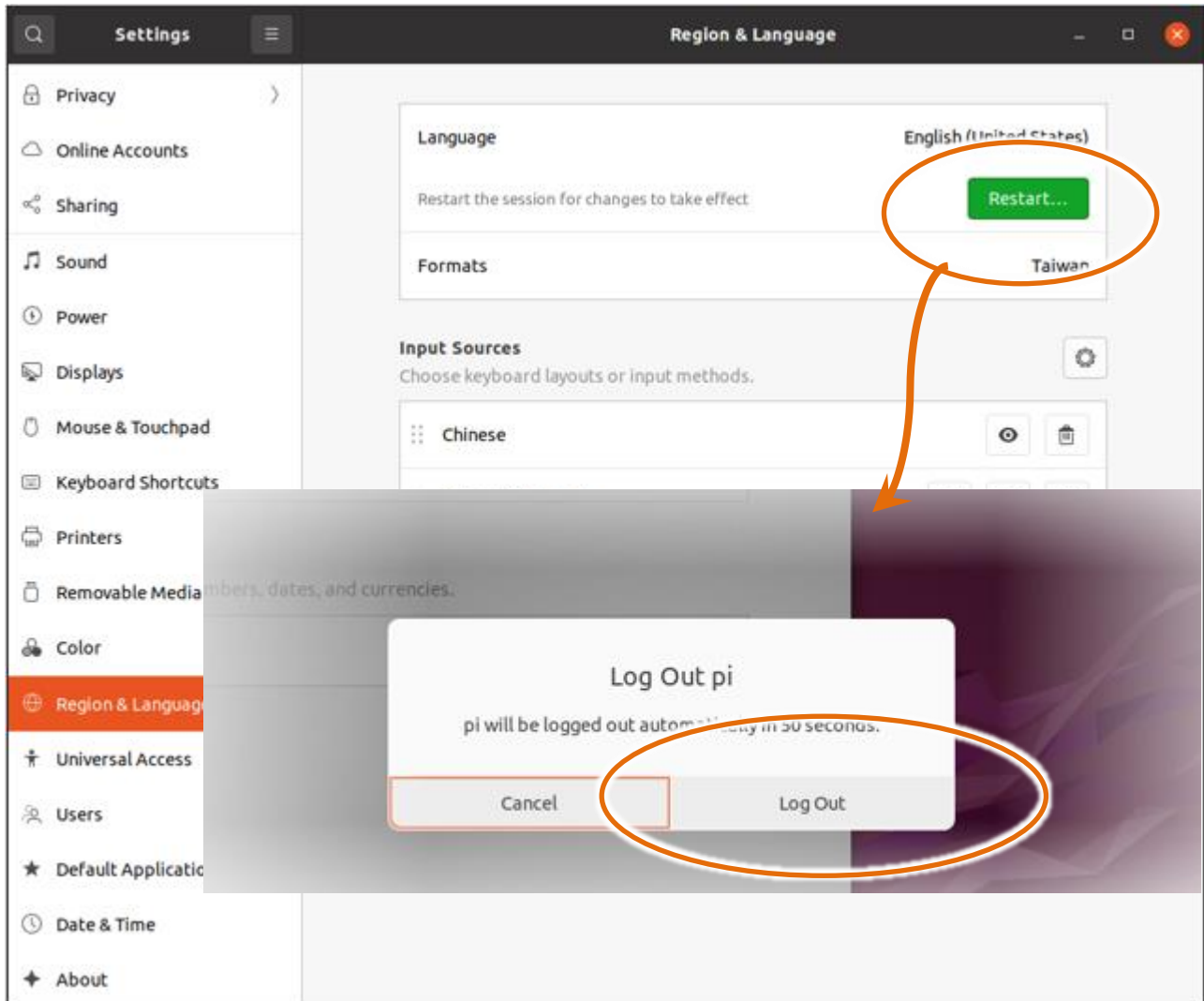
Step 3: Select the desired language from the pop-up list, then click the Select button



Step 4: Click the Restart button to apply the language settings

When a different language is selected, the Restart button will appear in the Region & Language dialog.

Step 5: Click Log Out to confirm the restart process

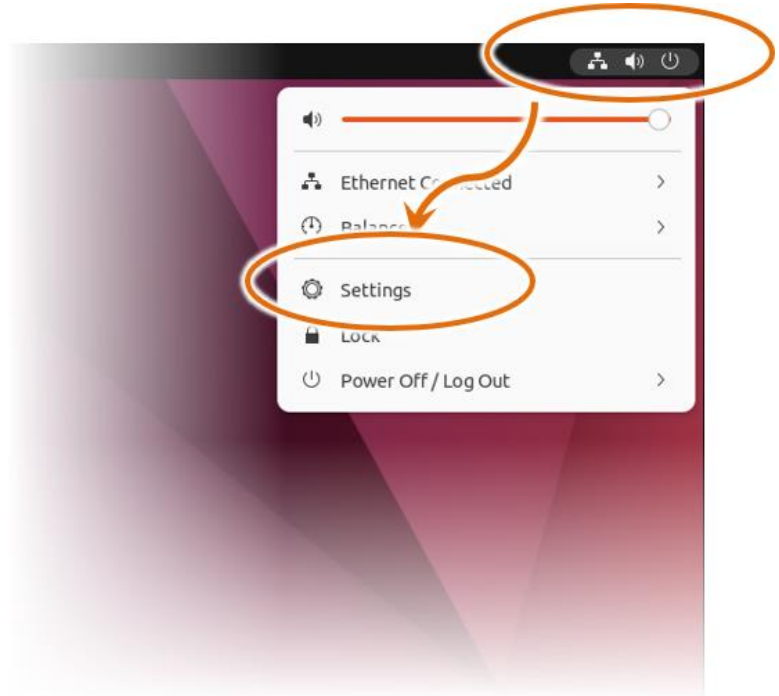


Step 6: At the login screen, enter the password and log in again to complete the system language change procedure

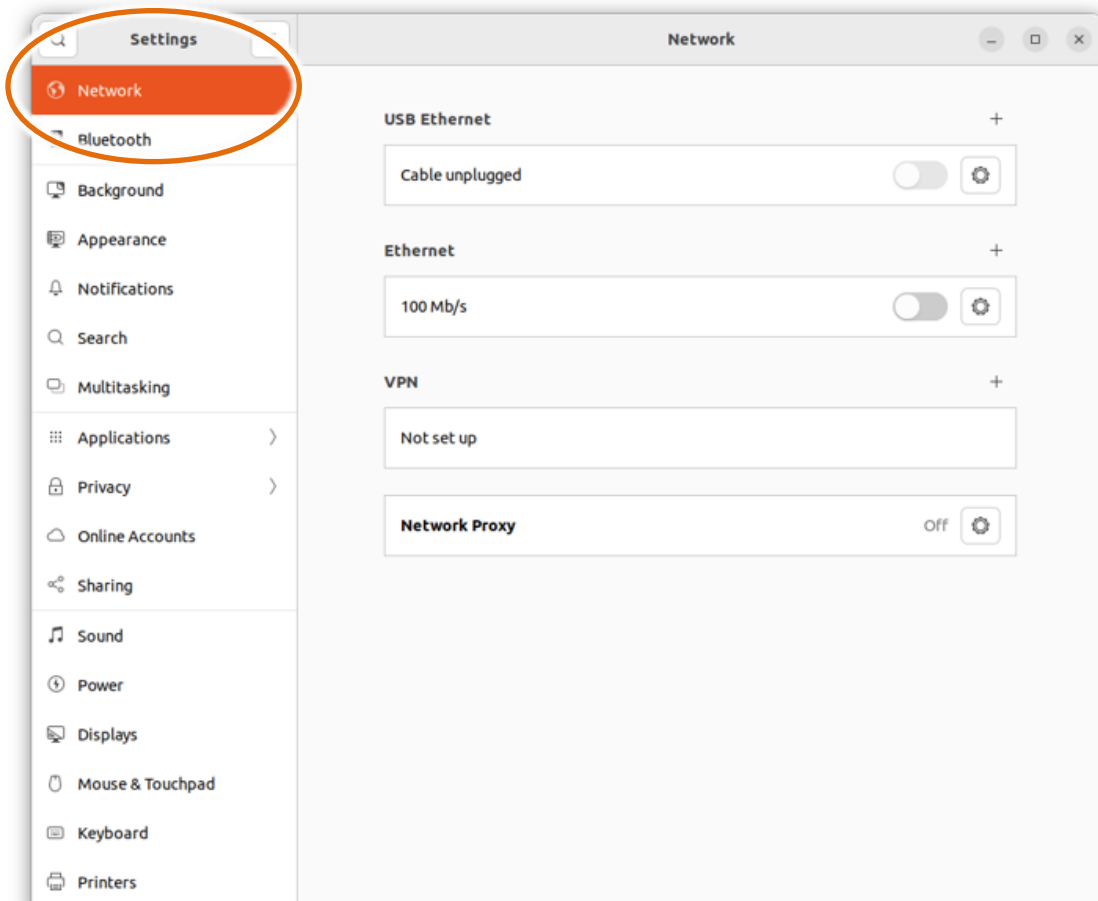
2.4. IP Configuration

Ubuntu provides a built-in IP configuration utility. Please follow the steps below to configure the IP address.

Step 1: Click the icon at the upper-right corner of the desktop, then select Settings from the pop-up menu



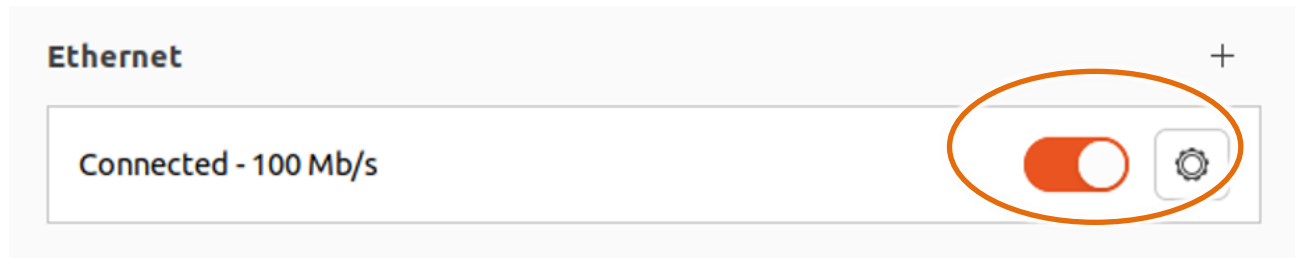
Step 2: Select Network



Step 3: Configure the IP Address

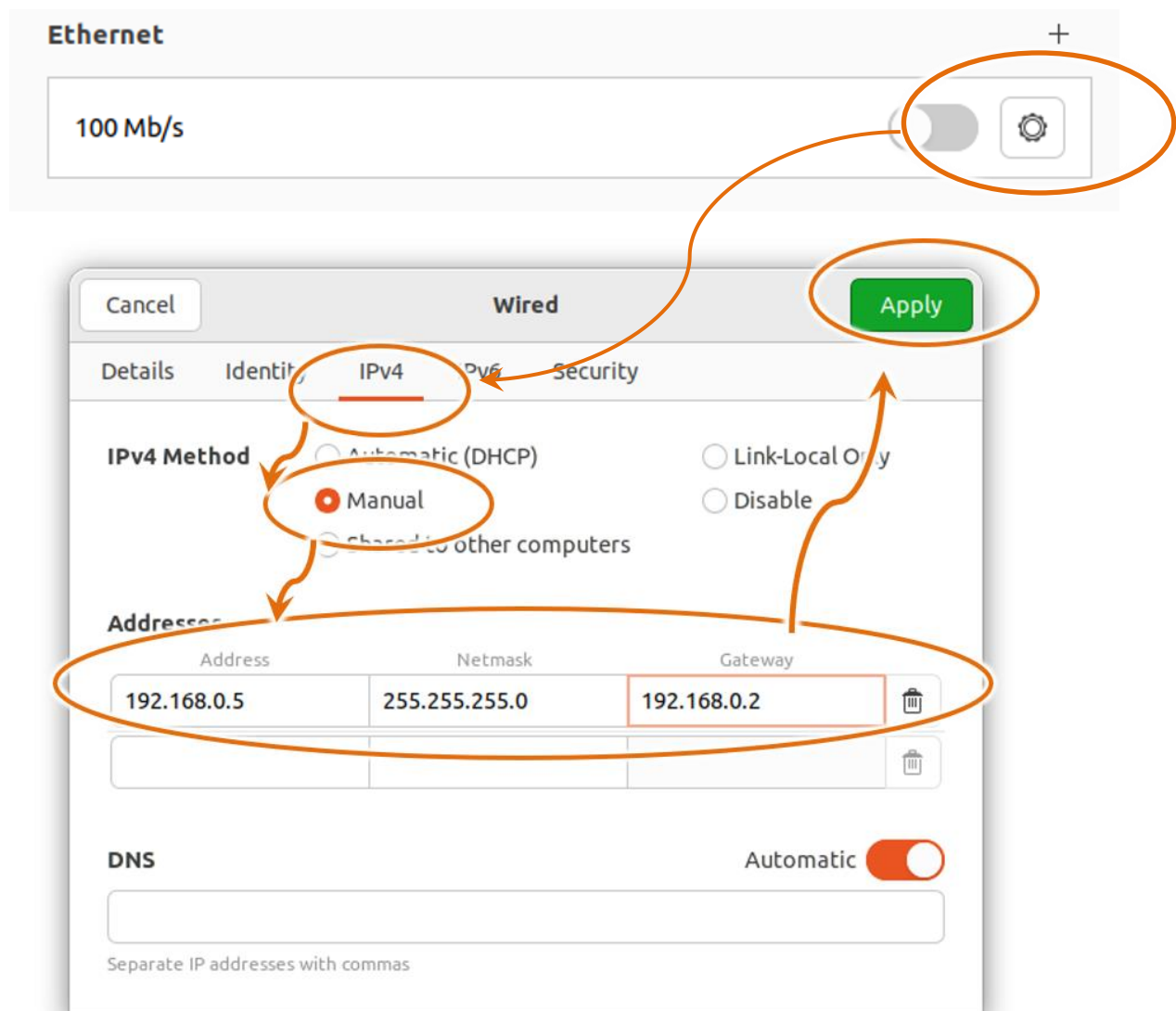
Assign IP Address Using DHCP

Enable the network connection directly using the switch button.



Manually Assign an IP Address

Click the Settings button. In the pop-up dialog, select the IPv4 tab, choose the Manual option, configure the IP address, and then click the Apply button to save the settings.



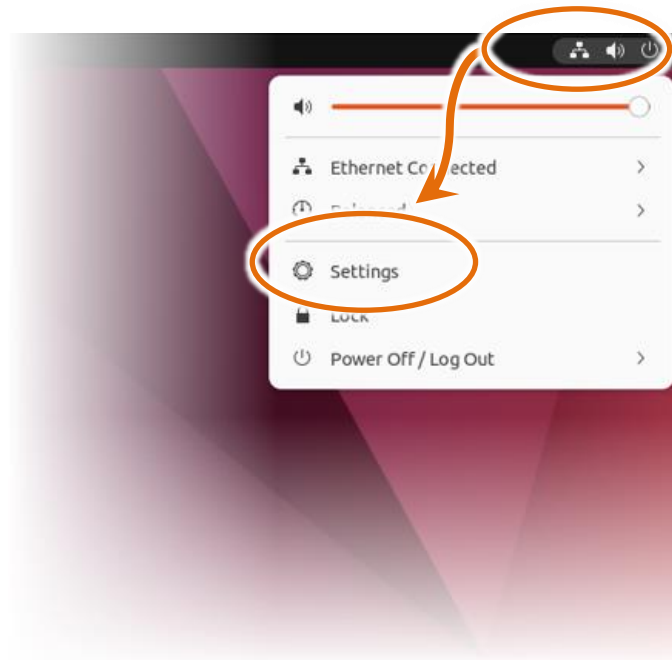
3. Tools and Utilities

This chapter briefly introduces the commonly used system and software tools of the AIB-2541M series modules and their advantages.

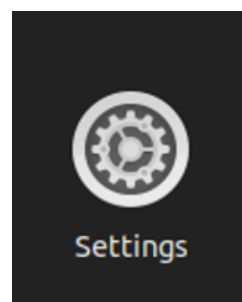
In addition to the built-in Ubuntu software tools, the AIB-2541M also includes several software tools specifically designed for the AIB-2541M.

3.1. System tools

System tools can be accessed by clicking the icon at the upper-right corner of the desktop and selecting **Settings**.



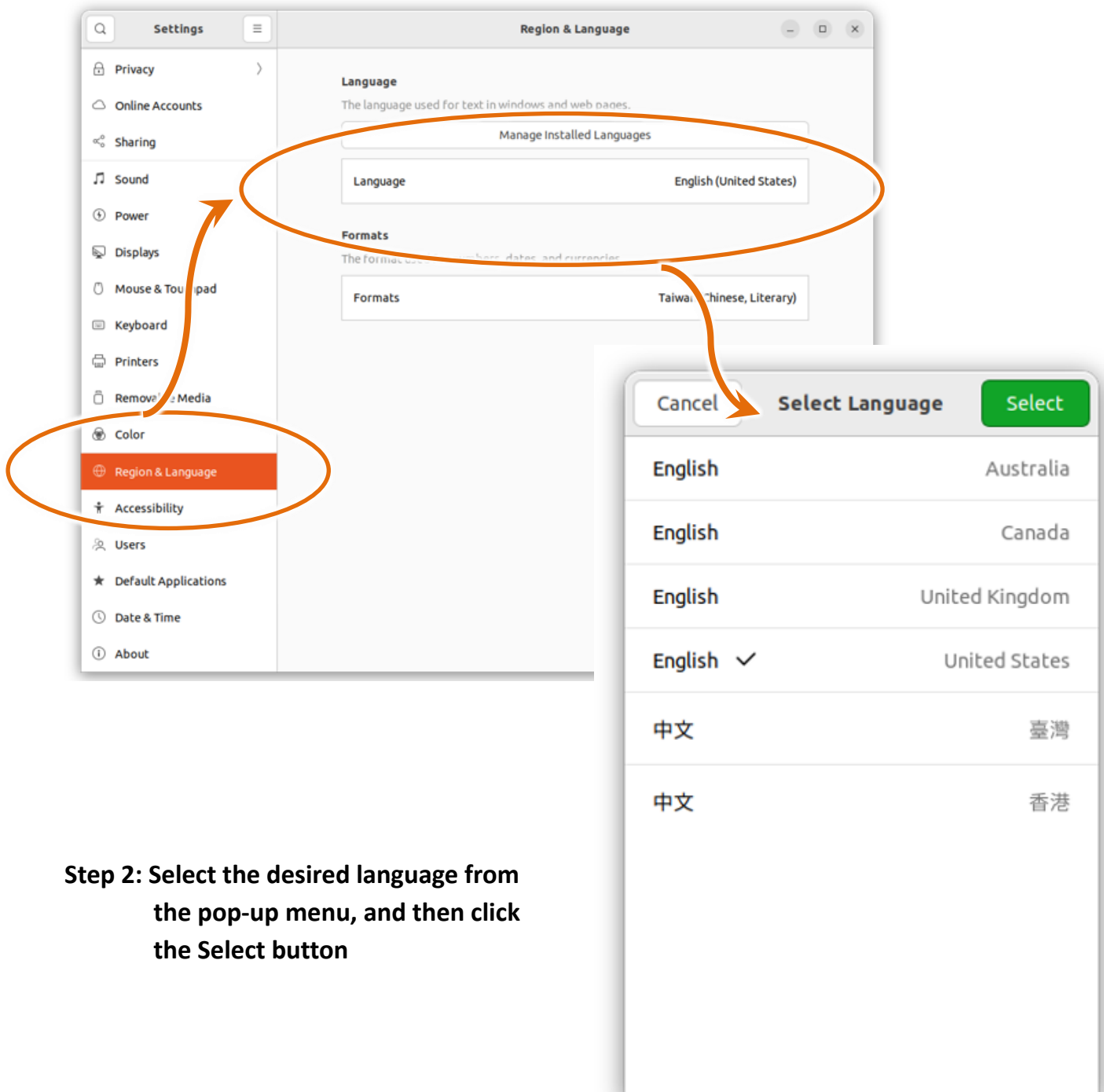
Alternatively, click the icon  at the lower-left corner and then select Settings.



3.1.1. Changing the User Interface Language

Regional Settings is a built-in Ubuntu feature that allows users to easily change the user interface language of the AIB-2541M series modules. Please follow the steps below to change the user interface language.

Step 1: Click Region & Language, then select the Language option.

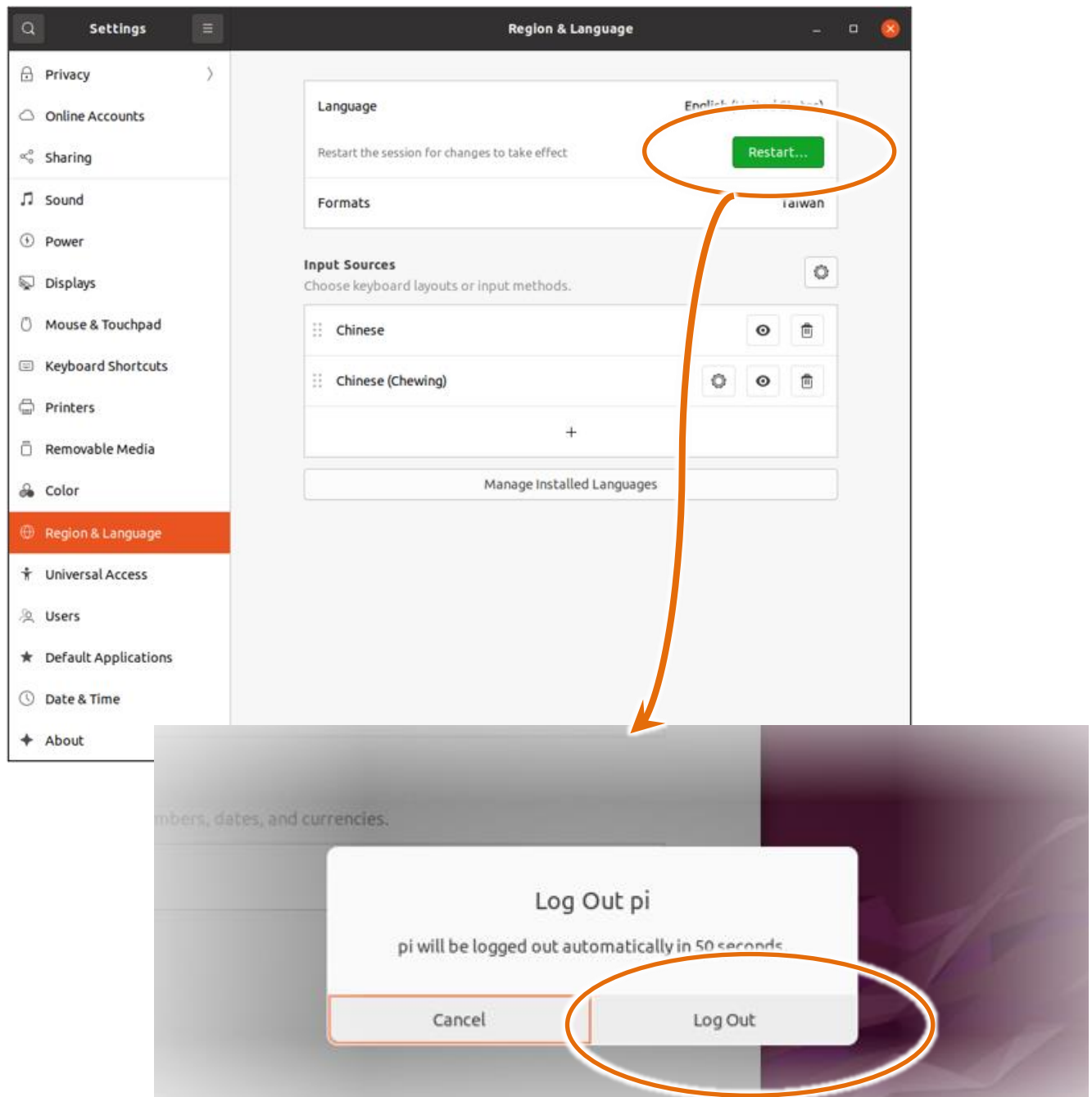


Step 2: Select the desired language from the pop-up menu, and then click the Select button

Step 3: Click the Restart button to restart the system and apply the settings

Return to the Region & Language dialog window. When a different language is selected, a Restart button will appear under the Language item.

Step 4: Click Log Out to confirm the restart process

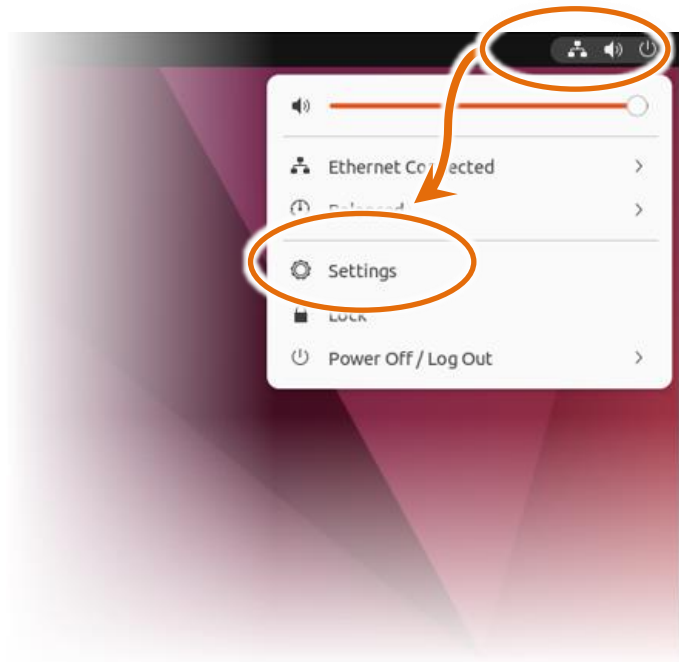


Step 5: At the user login screen, enter the password and log in again to complete the system language change process.

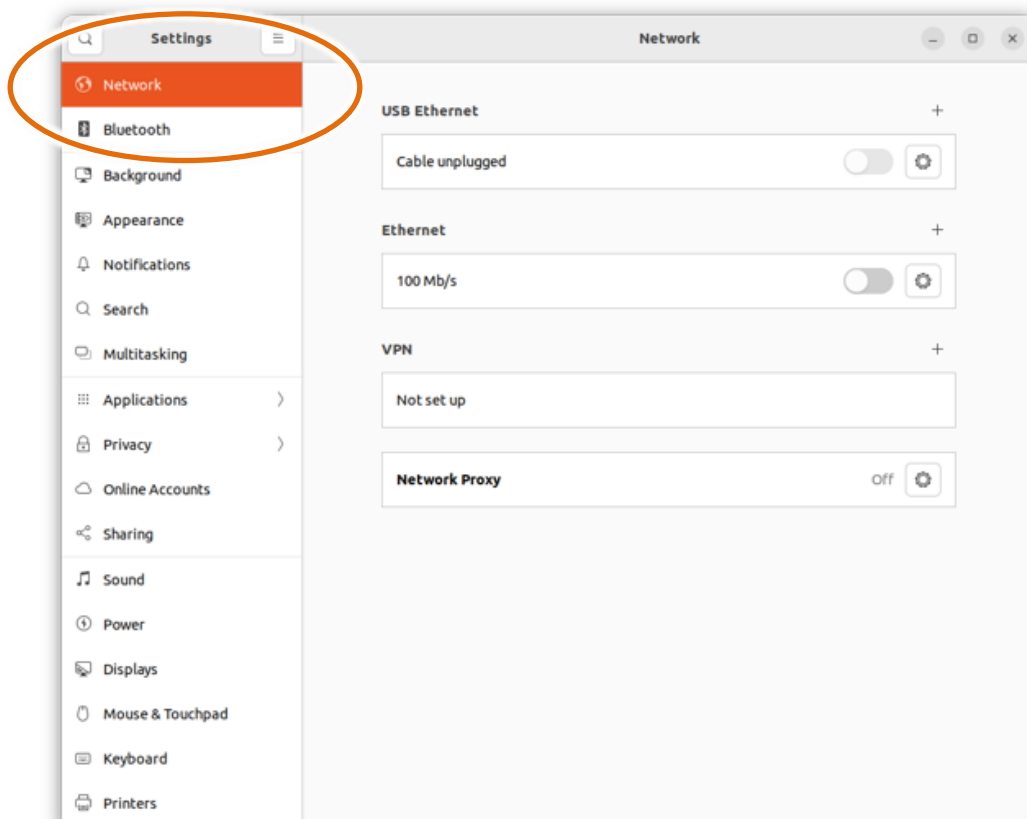
3.1.2. IP Configuration

Ubuntu includes a built-in IP configuration utility. Please follow the steps below to configure the IP address.

Step 1: Click the icon at the upper-right corner of the desktop, then select Settings from the pop-up menu



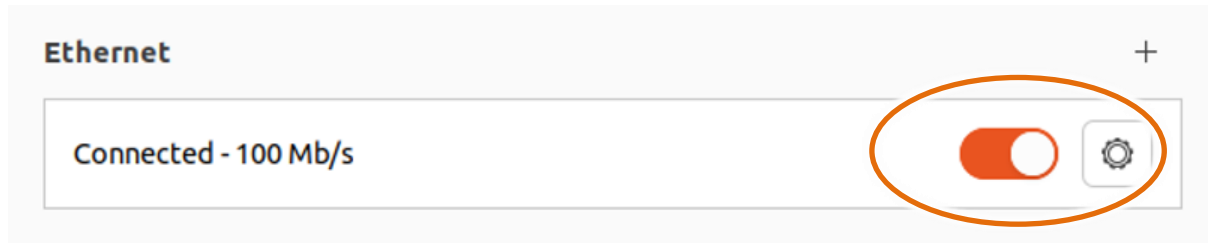
Step 2: Click Network



Step 3: Configure the IP Address

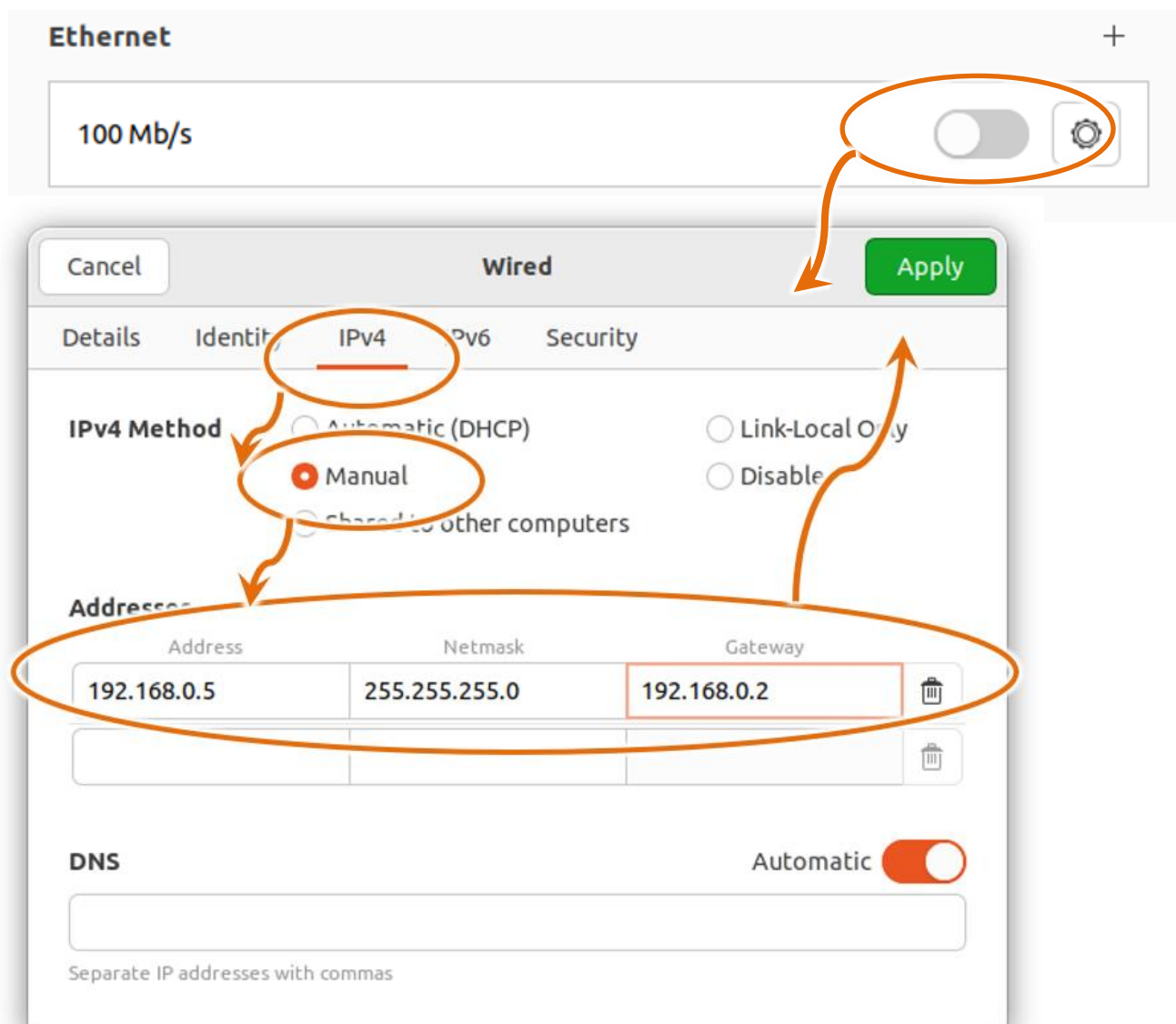
Assign IP Address via DHCP

Enable the network connection directly using the switch button.



Manually Assign an IP Address

Click the Settings button. In the pop-up dialog, select the IPv4 tab, choose the Manual option, configure the IP address, and then click the Apply button after completion.

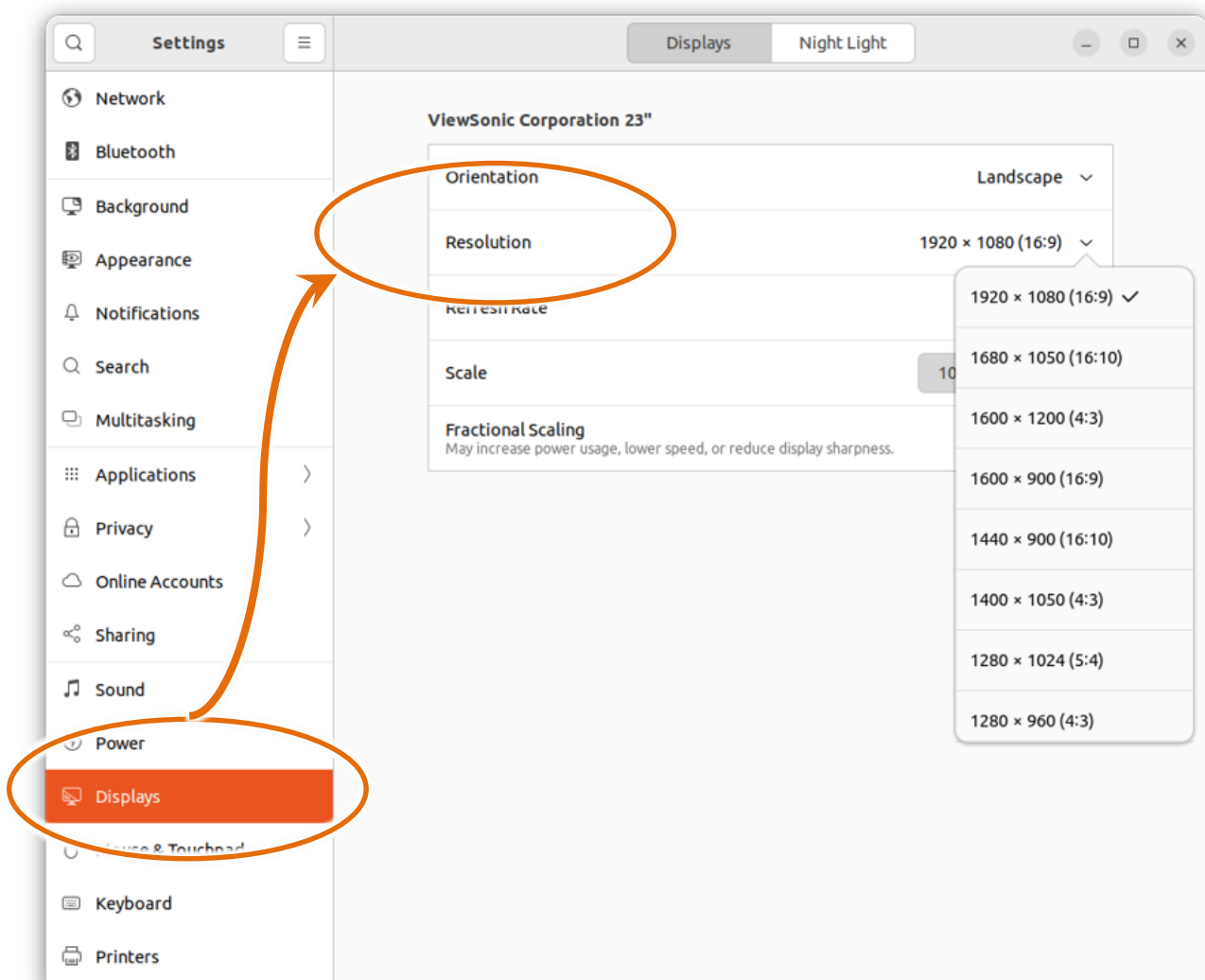


3.1.3. Display Resolution Settings

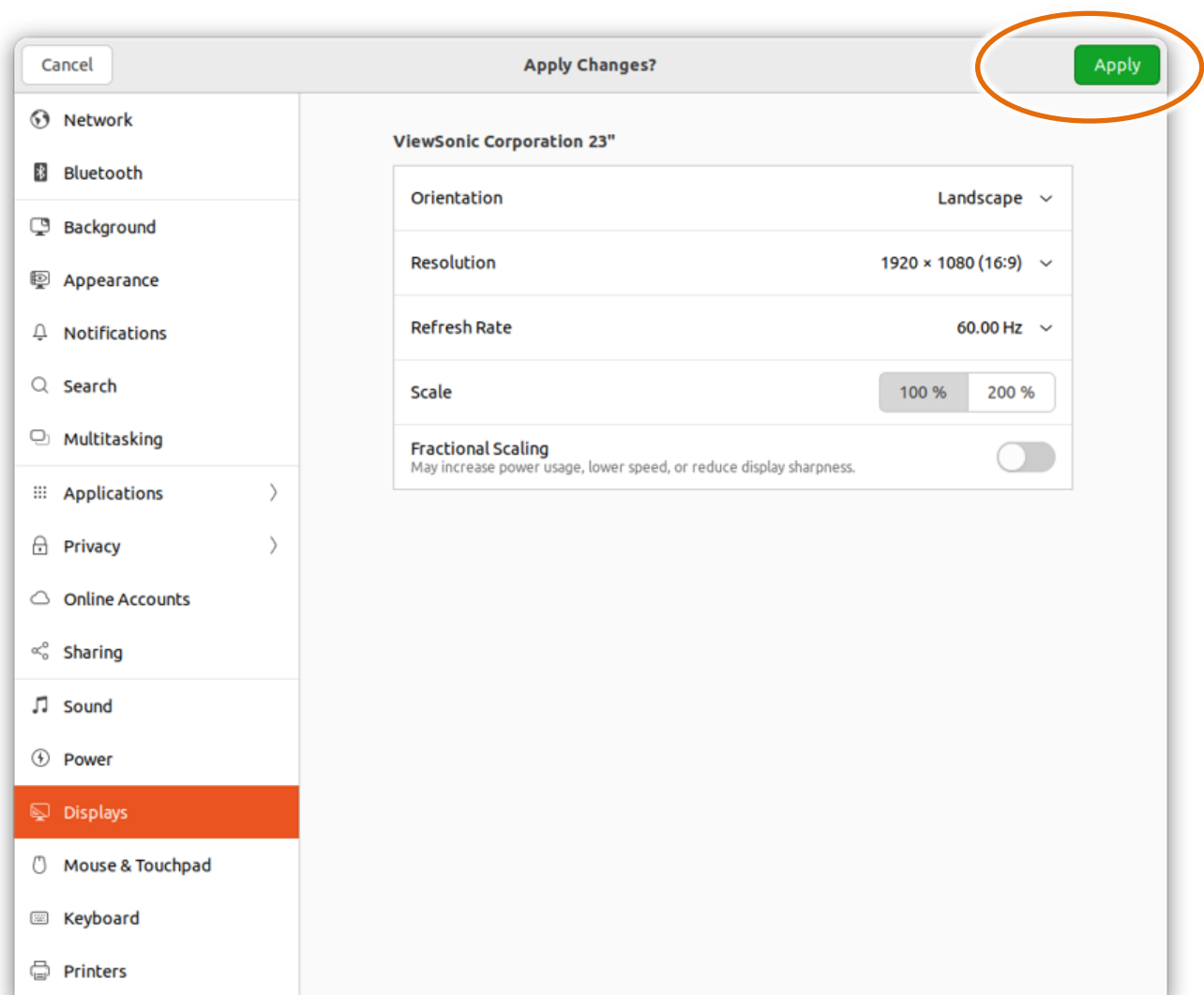
Ubuntu includes built-in display resolution configuration functionality. The supported resolution range is from 800 × 600 to 3840 × 2160 (4K). Please follow the steps below to configure the display resolution.

Step 1: Select the Display tab

Step 2: Click the Resolution item and configure the desired resolution

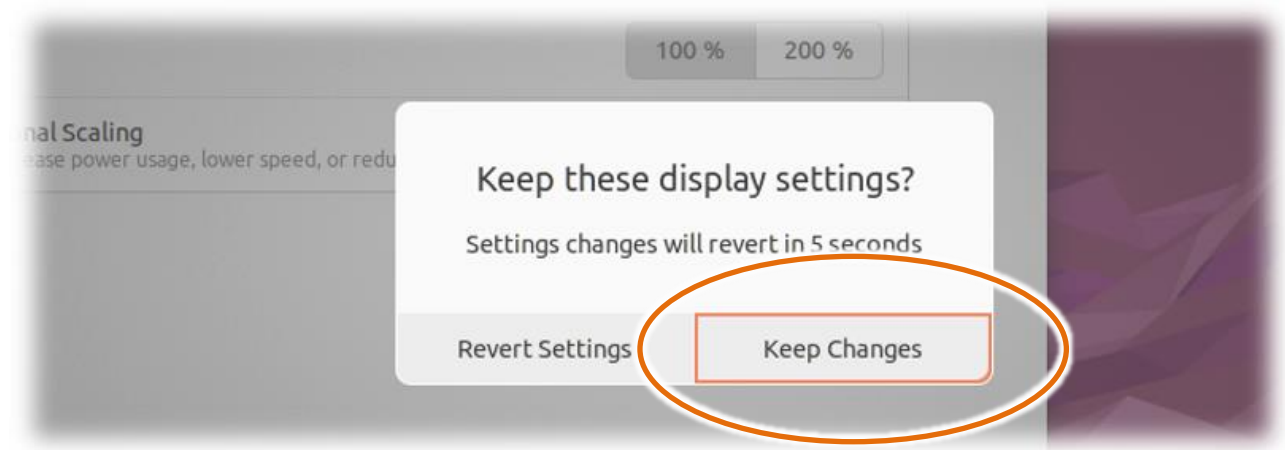


Step 3: After configuration is complete, click the Apply button



Step 4: If the selected resolution displays correctly, click Keep Changes in the confirmation dialog

If the display is abnormal, click Revert Settings or wait 20 seconds to restore the previous settings automatically.

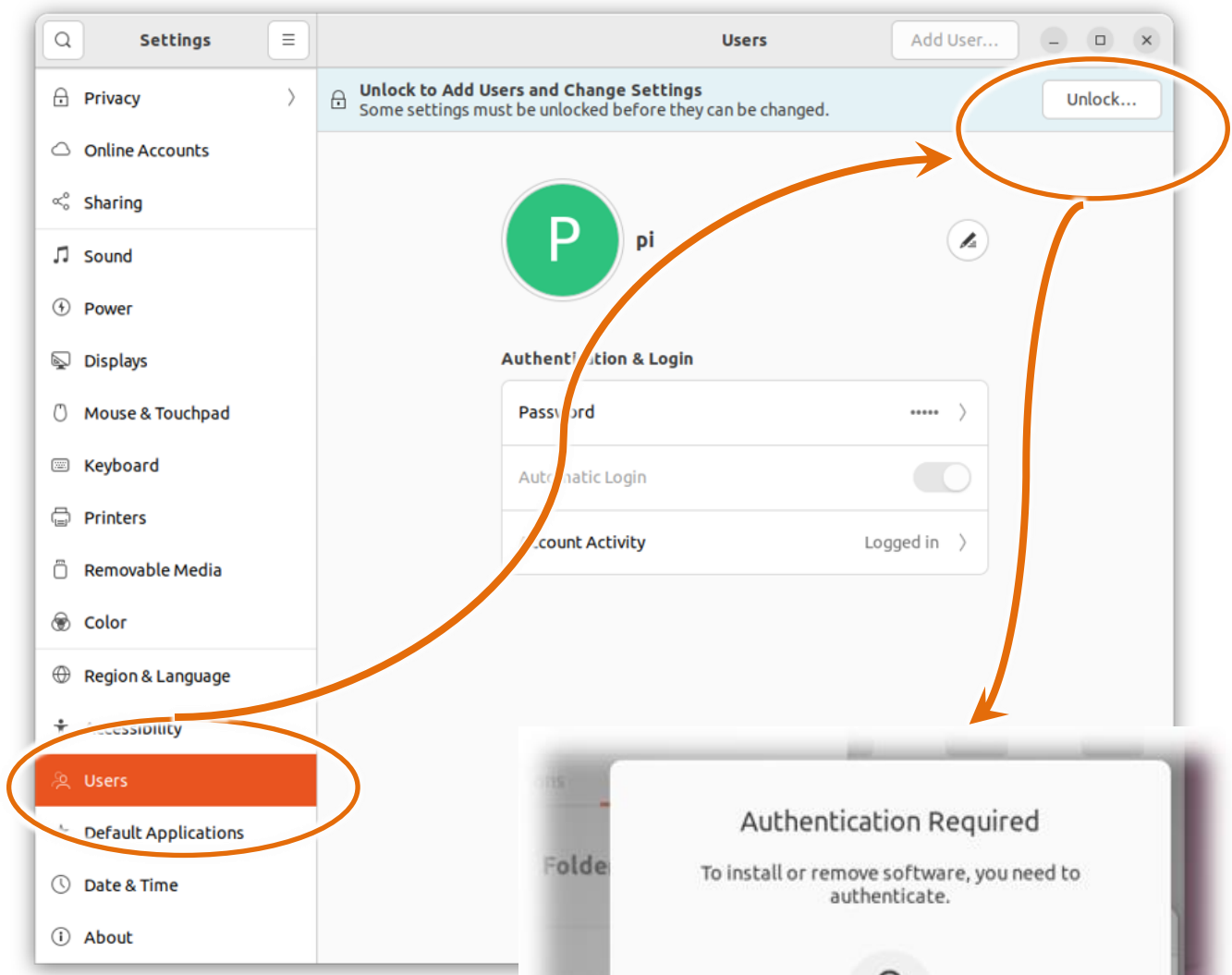


3.1.4. User Account Management

User account management is a security feature that helps prevent unauthorized system modifications to the AIB-2541M series modules.

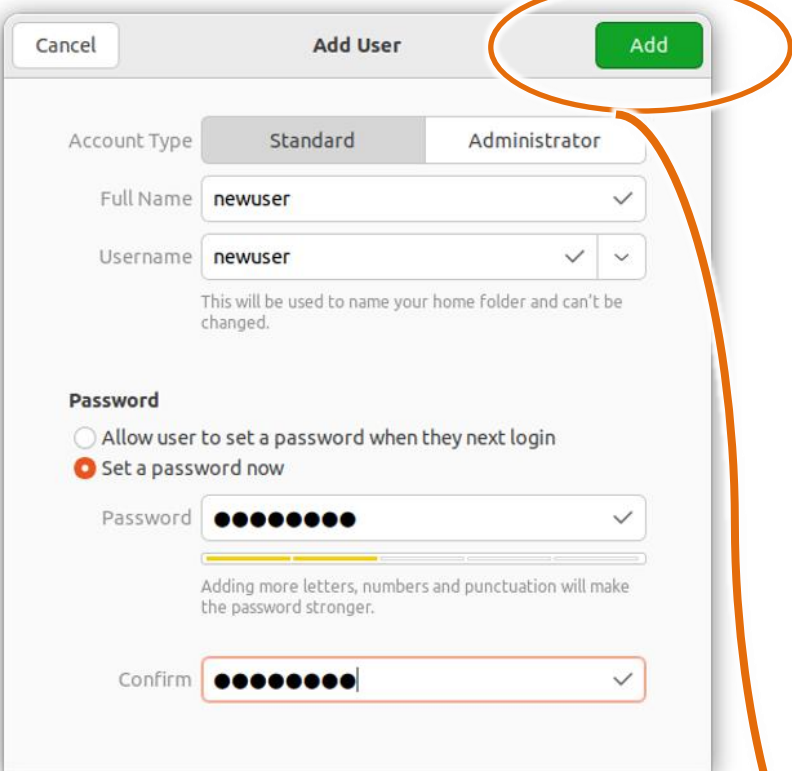
Adding a User Account

Step 1: Click “Users”, and then click the “Unlock...” button



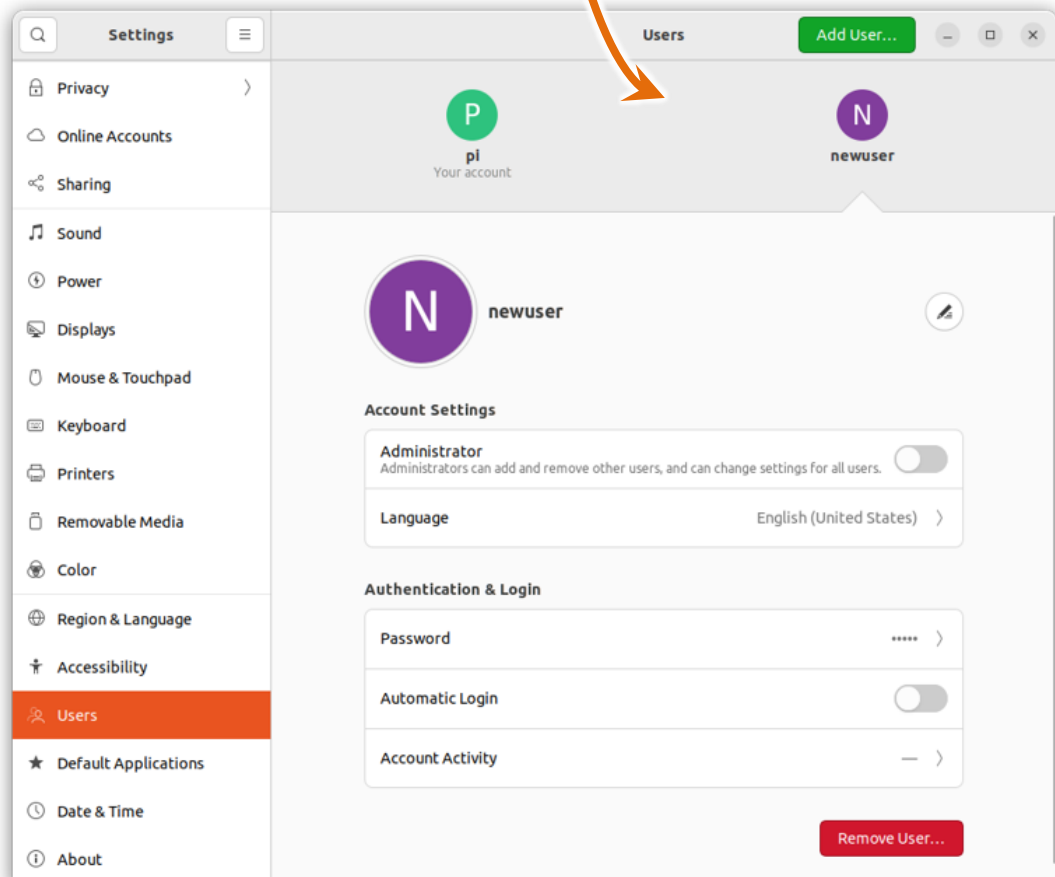
Step 2: In the pop-up dialog window, enter the password to unlock user management permissions

Step 3: In the Add User dialog window, enter the user account information, and then click the Add button to complete user creation



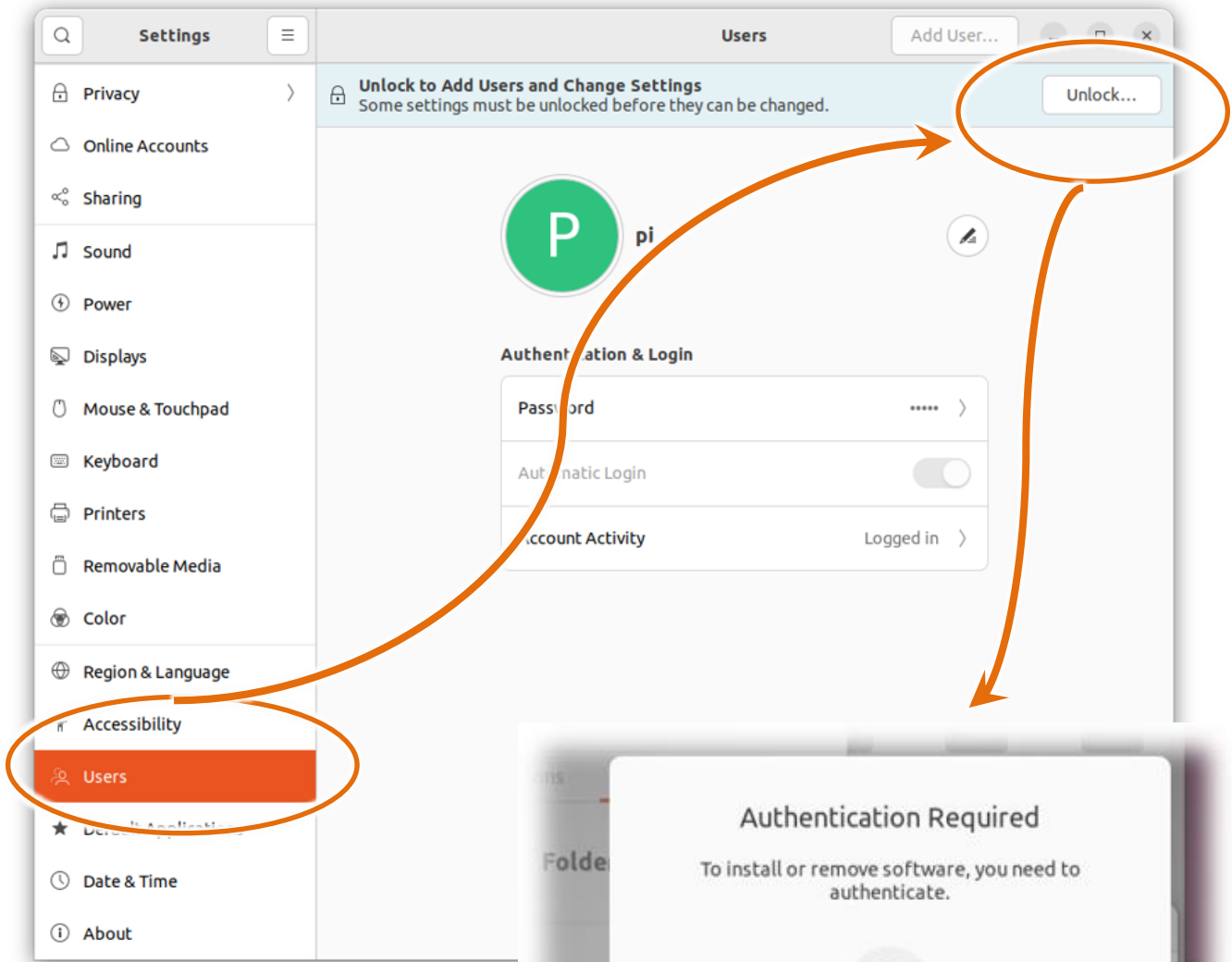
The 'Add User' dialog window is shown. It has a title bar with 'Cancel', 'Add User', and 'Add' buttons. The 'Add' button is circled in orange. The dialog contains the following fields and options:

- Account Type:** Two tabs, 'Standard' (selected) and 'Administrator'.
- Full Name:** A text field containing 'newuser' with a checkmark icon.
- Username:** A text field containing 'newuser' with a checkmark icon and a dropdown arrow. Below it, a note says: 'This will be used to name your home folder and can't be changed.'
- Password:** A section with two radio buttons: 'Allow user to set a password when they next login' (unselected) and 'Set a password now' (selected). Below the radio buttons is a password field with 10 dots and a checkmark icon. A strength indicator bar is shown below the field with the text: 'Adding more letters, numbers and punctuation will make the password stronger.'
- Confirm:** A text field with 10 dots and a checkmark icon.



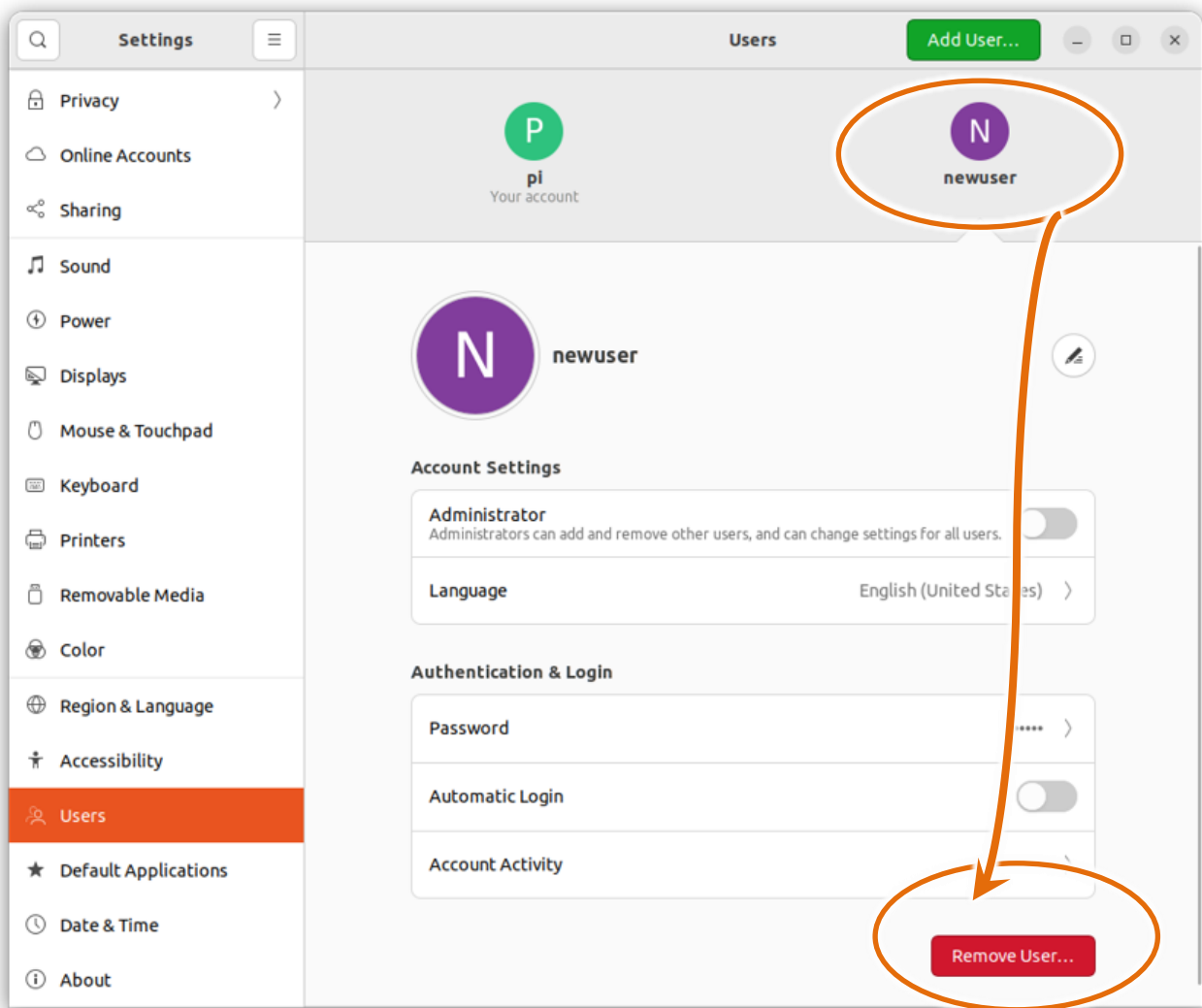
Deleting a User Account

Step 1: Click “Users”, and then click the “Unlock...” button

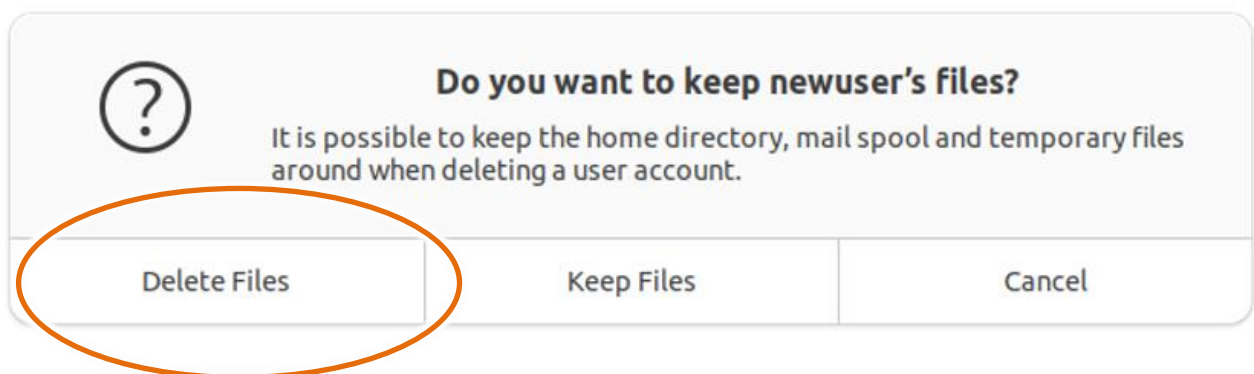


Step 2: In the pop-up dialog window, enter the password to unlock user management permissions

Step 3: Select the user account to be deleted, and then click “Remove User...”



Step 4: In the warning dialog window, click the Delete Files button to confirm deletion of the user account



3.2. Software Tools

Software tools can be accessed by clicking the icon  at the lower-left corner of the desktop. This section introduces several commonly used software tools.



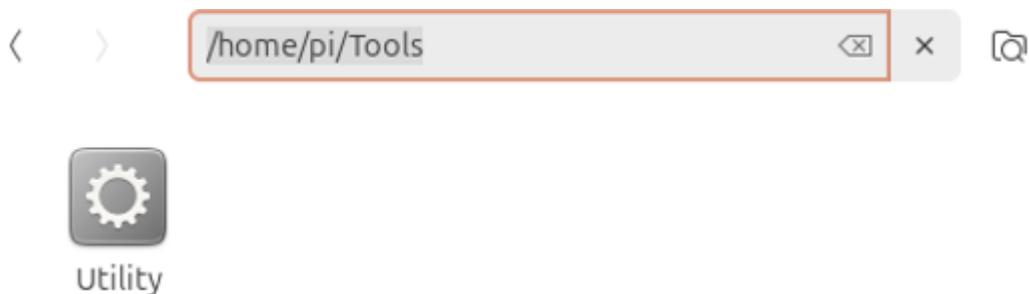
3.2.1. Utility – Viewing System Information

The Utility application provides a unified interface for viewing system information and configuring AIB-2541M functions.

Launching Utility

Program path: `/home/pi/Tools/`

Double-click the Utility file using the left mouse button to launch the Utility application.



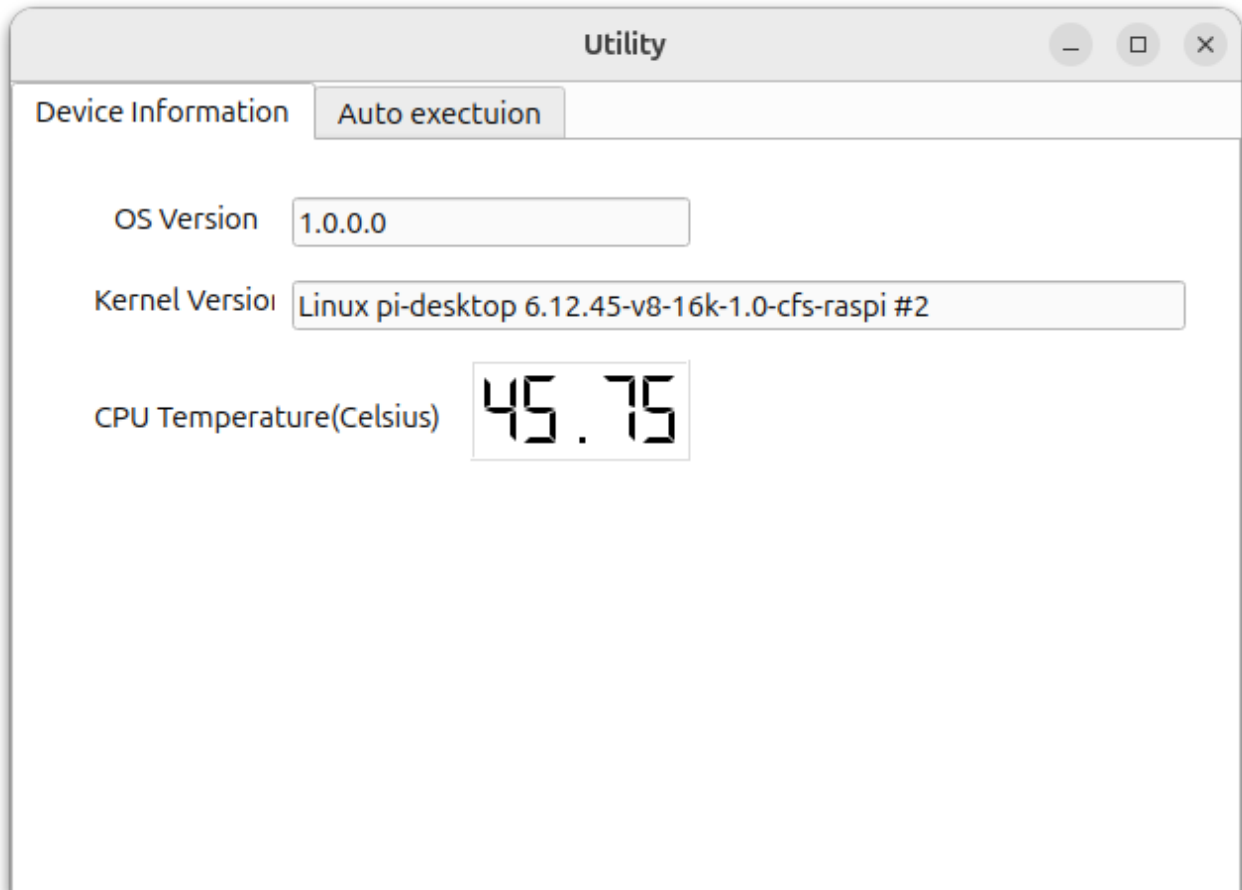
The following are the Utility tabs and their functions:

Tab	Description
Device Information	View OS version, kernel version, and CPU temperature
Auto Execution	Configure startup applications

Viewing System Version and CPU Temperature

The Device Information tab provides system information such as OS version and CPU temperature for the AIB-2541M series modules.

The following figure shows the system information provided by the Utility.



Managing Startup Applications

The Auto Execution tab provides the ability to manage applications that automatically execute after boot. Up to 7 startup applications are supported.

Guidance & Notes



Supported auto-execution file types are .py files and executable files.
Execution order is from Program 1 to Program 7.

After specifying the application paths, click the Apply button to complete the configuration.

The screenshot shows a window titled "Utility" with standard window controls (minimize, maximize, close). It has two tabs: "Device Information" and "Auto execuion". The "Auto execuion" tab is active and contains the following elements:

Program	Path	Action
Program 1		Browse
Program 2		Browse
Program 3		Browse
Program 4		Browse
Program 5		Browse
Program 6		Browse
Program 7		Browse

At the bottom right of the "Auto execuion" tab is an button.

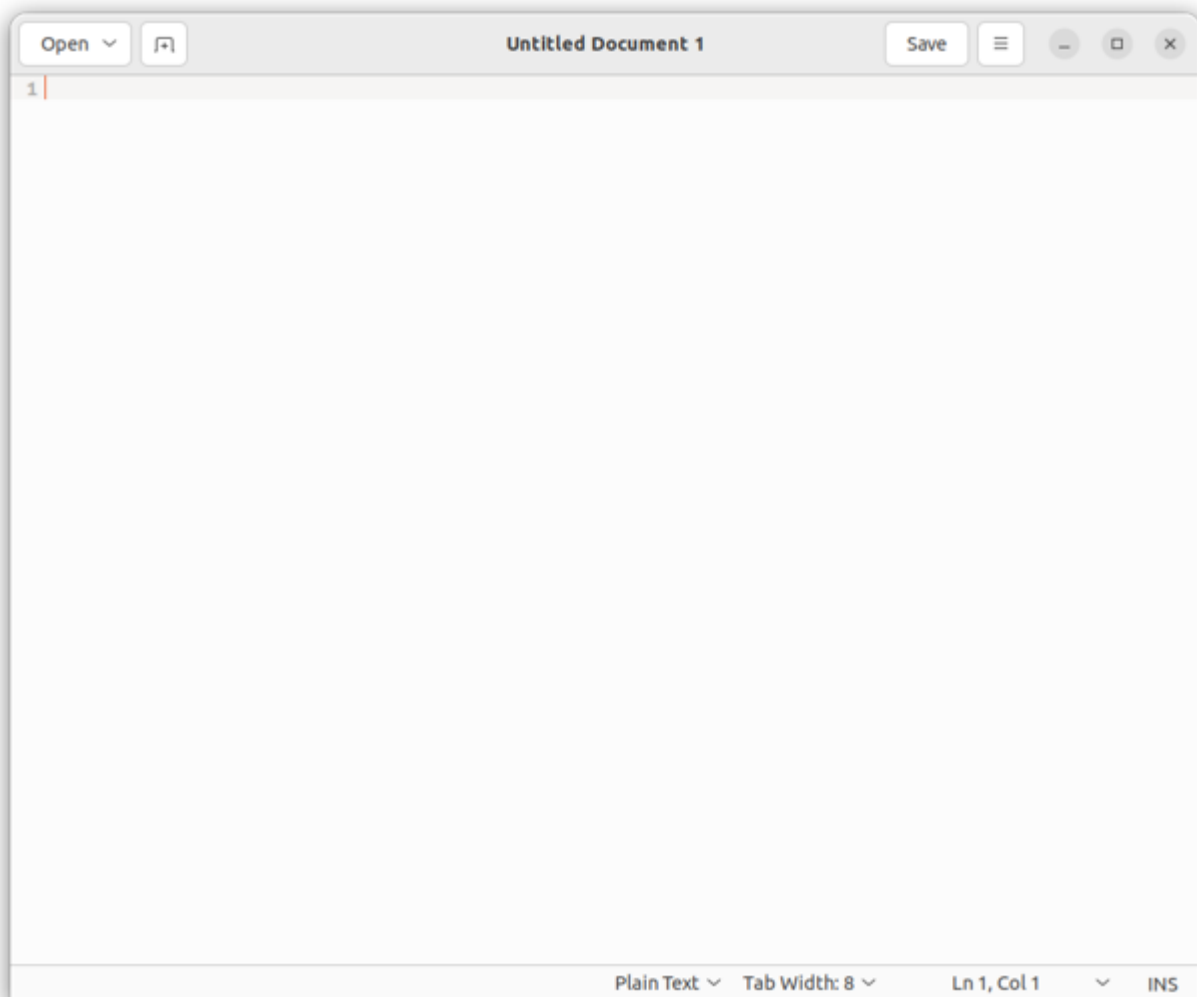
3.2.2. Text Editor

Text Editor is a simple text-editing application for creating and editing plain-text files without formatting or styles.

Select Text Editor from the application launcher to launch it.



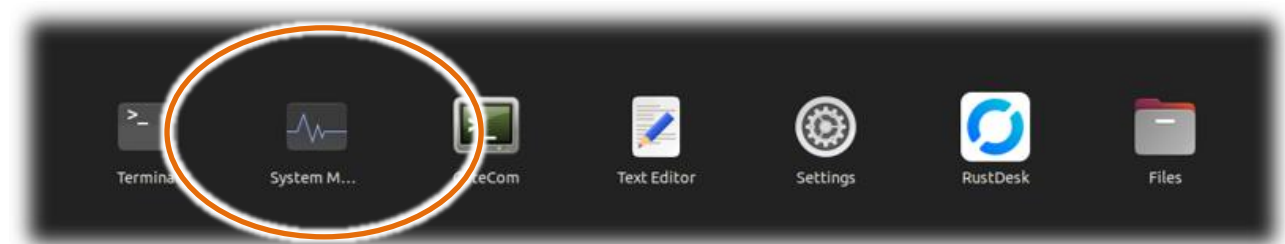
After launching Text Editor, the application window appears as shown below.



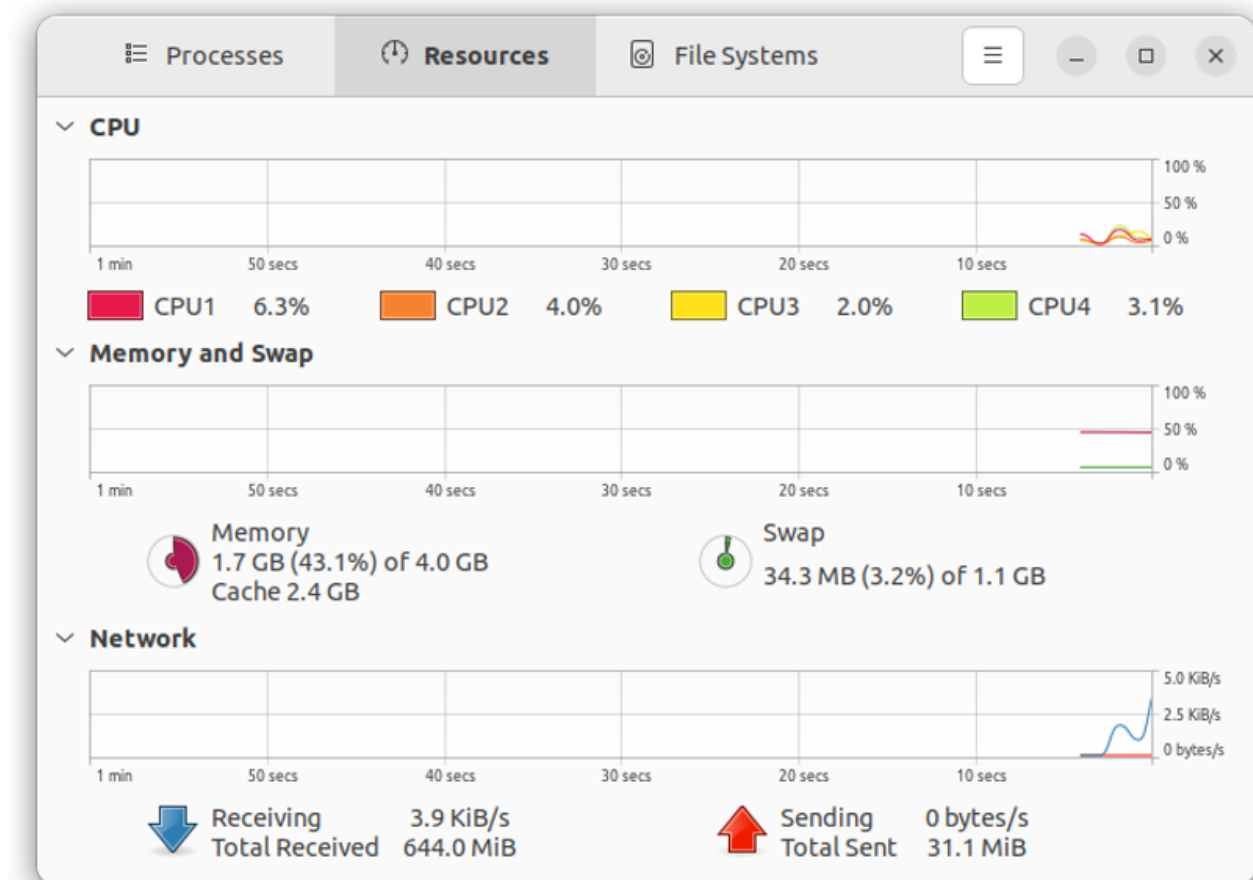
3.2.3. Performance Monitor

System Monitor provides real-time information for all processes and system threads, similar to the Windows Task Manager. Ubuntu already includes this software. Follow the steps below to launch it.

Click System Monitor.



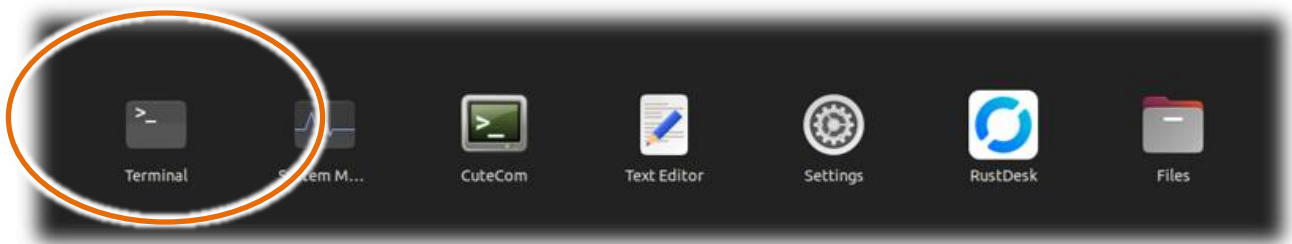
After launching System Monitor, the application window appears as shown below.



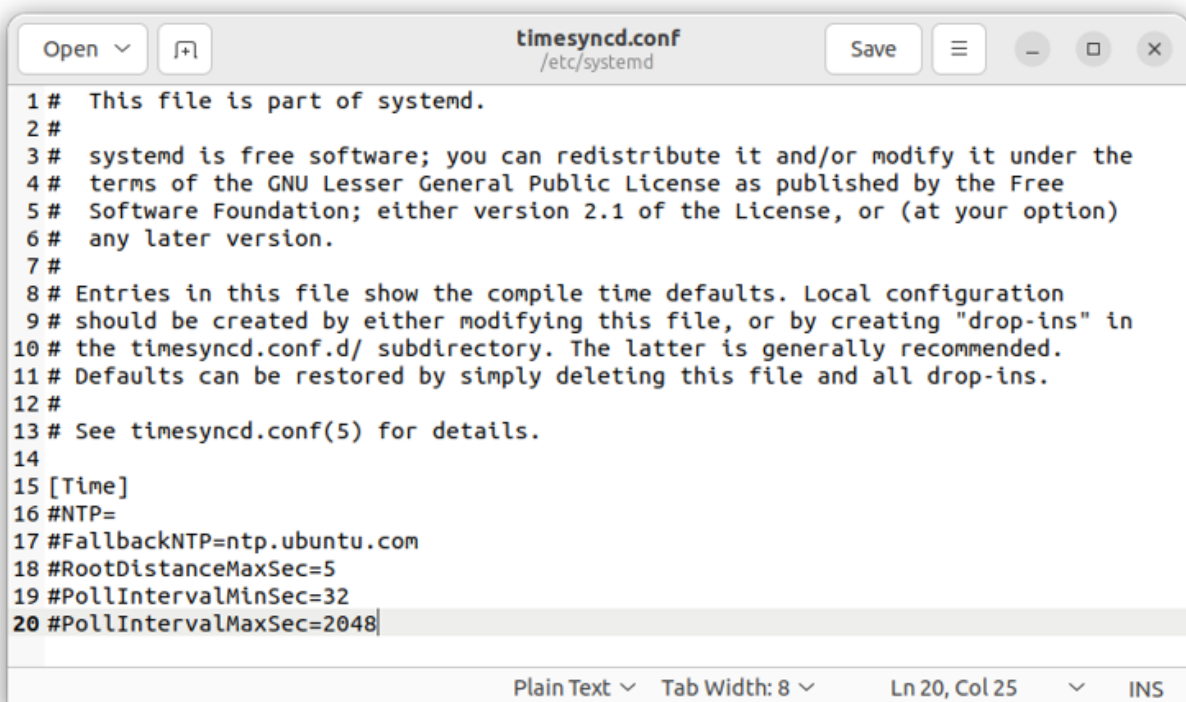
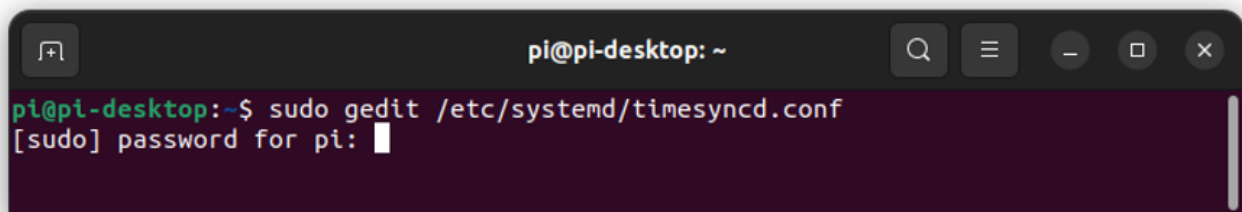
3.2.4. Automatic Time Synchronization

Ubuntu includes a built-in time synchronization feature. Please follow the steps below for configuration.

Step 1: Launch Terminal

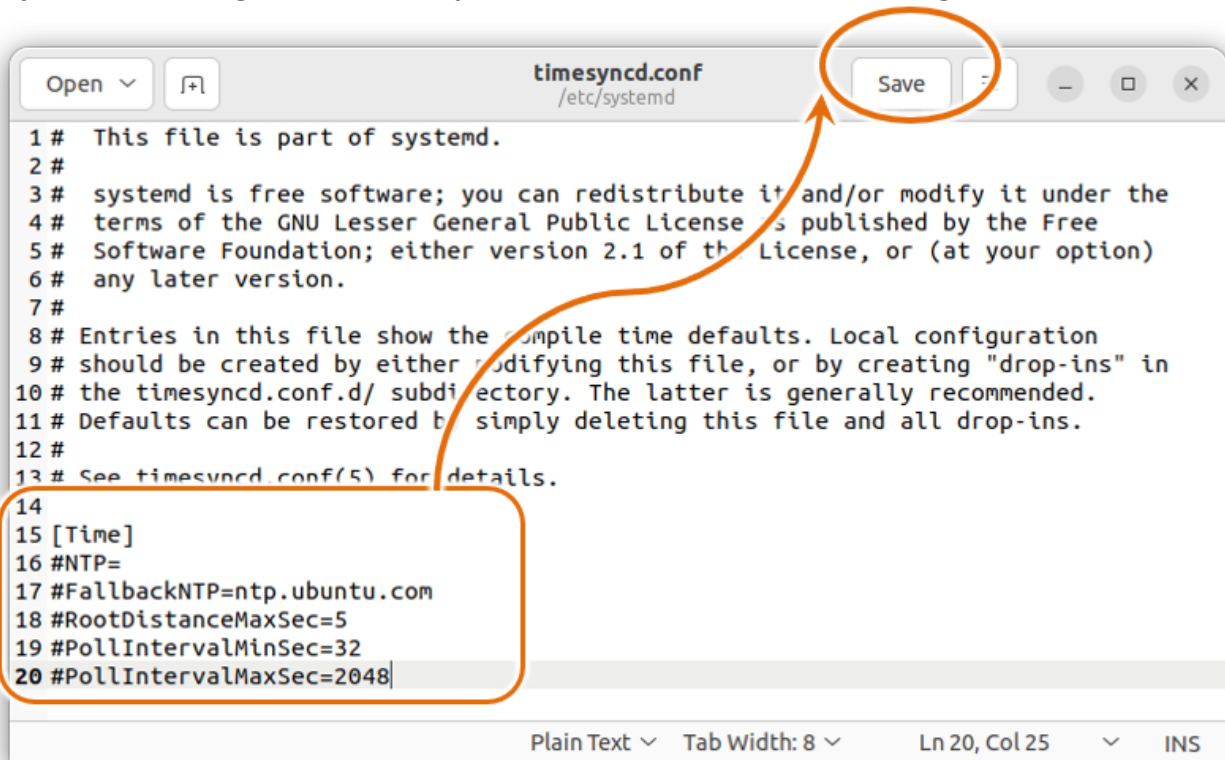


Step 2: Enter the following command “`sudo gedit /etc/systemd/timesyncd.conf`”, and then enter the password (the default password is “icpdas”)



Step 3: Configure the [Time] parameters in the timesyncd.conf file

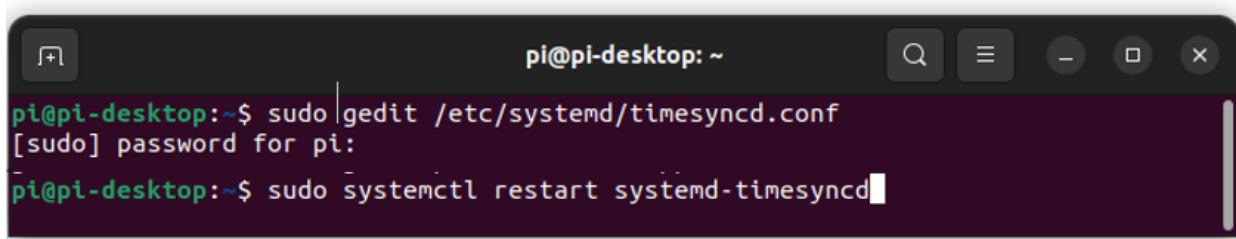
Step 4: After configuration is complete, click Save to save the settings



The parameters are described in the following table.

Field	Description
NTP	Primary time synchronization server
FallbackNTP	Backup time synchronization server
RootDistanceMaxSec	Maximum allowable time difference with the server (seconds)
PollIntervalMinSec	Minimum synchronization interval
PollIntervalMaxSec	Maximum synchronization interval
Field	Description

Step 5: In Terminal, enter the following command to restart the time synchronization service and load the new settings

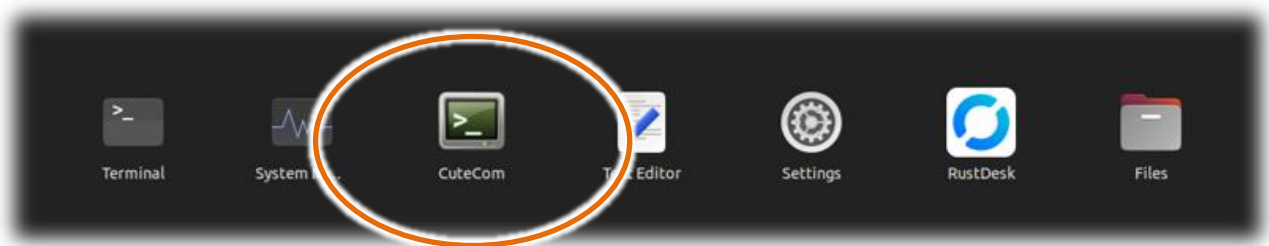


3.2.5. CuteCom

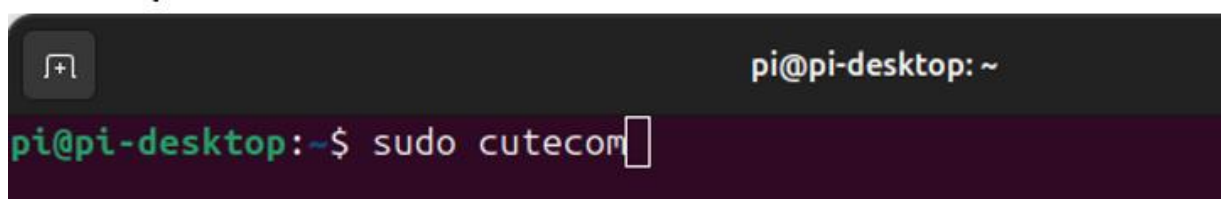
CuteCom is a graphical serial communication utility for Ubuntu. It can be used to send commands to external modules and receive data from external modules through the serial port.

Due to serial port access permission restrictions, please launch CuteCom with root privileges by executing "sudo cutecom" in Terminal.

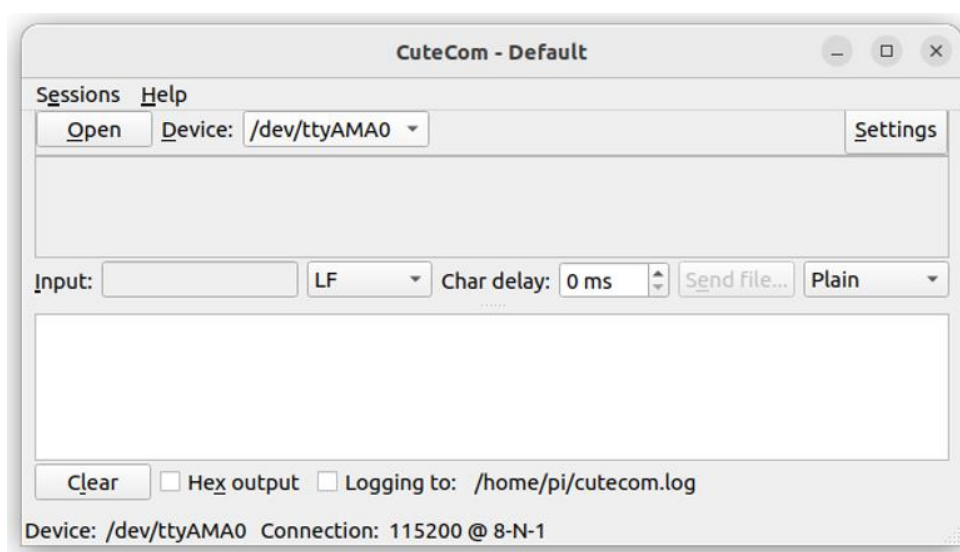
Step 1: Click Terminal



Step 2: Enter the command sudo cutecom, press Enter, and then enter the sudo password



The CuteCom main window is shown below after startup.



For more communication commands supported by external modules, please refer to:

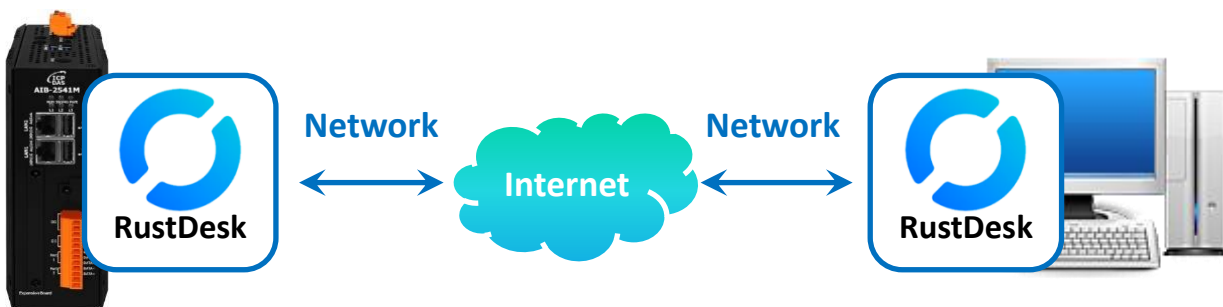
http://www.icpdas.com/root/product/solutions/remote_io/rs-485/i-8k_i-87k/i-8k_i-87k_selection.html#b

3.3. Other Support Tools

3.3.1. Remote Connection Software

Remote desktop software may introduce security concerns; therefore, the AIB-2541M series modules do not come with such software preinstalled. If remote desktop functionality is required, RustDesk has been tested and verified to operate correctly.

RustDesk and AIB-2541M Architecture



Step 1: On the PC side, visit the RustDesk official website

<https://rustdesk.com/>

Step 2: On the download page, download the Ubuntu ARM64 version

1.3.0 Latest Compare

rustdesk released this Aug 19 · 85 commits to master since this release · 1.3.0 · 2a0fd55

WARNING: YOU MAY BE BEING SCAMMED!

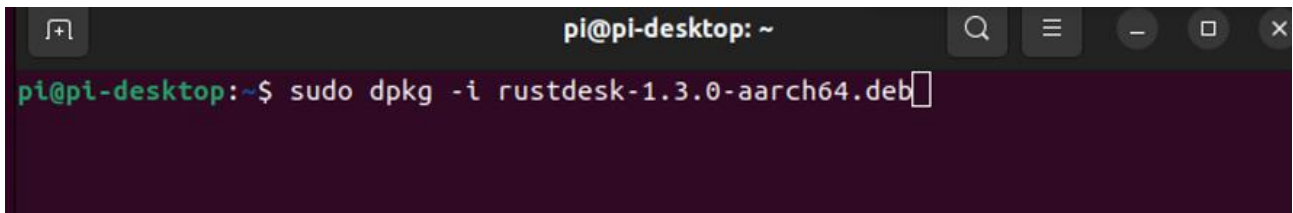
If you are on the phone with someone you DON'T know AND TRUST who has asked you to install RustDesk, do not install and hang up immediately.

They are likely a scammer trying to steal your money or other private information.

Architecture	Windows	Ubuntu	Mac	Android	Flatpak	AppImage	iOS
x86-64 (64-bit)	EXE MSI Universal	Download	Download	Universal	Download	Download	
AArch64 (ARM64)		Download	Download	Download	Download	Download	TestFlight
ARMv7 (32-bit)		Download		Download			
x86-32 (32-bit)	Download						

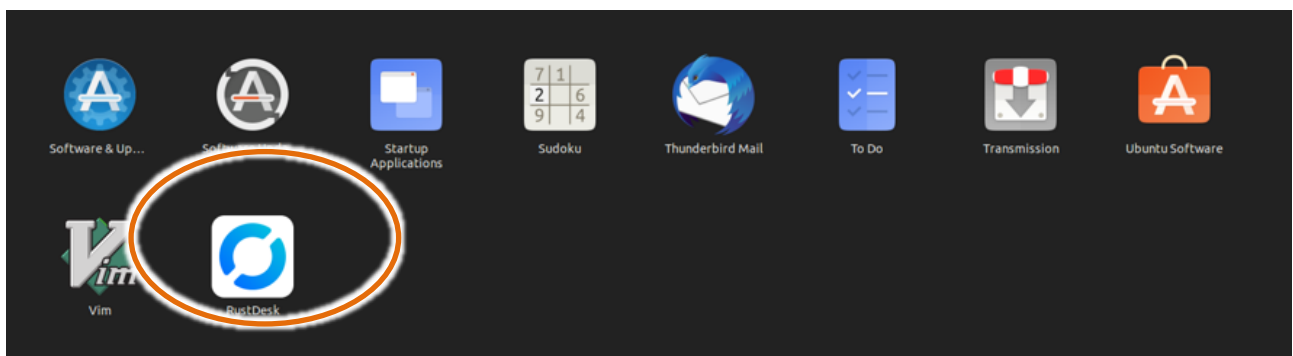
Step 3: After downloading, upload the rustdesk-1.3.0-aarch64.deb file (version 1.3.0 in this example) to the AIB-2541M

Step 4: Launch Terminal on the AIB-2541M, switch to the directory where the uploaded file is located, and enter “sudo dpkg -i rustdesk-1.3.0-aarch64.deb” to install RustDesk (version 1.3.0 in this example)

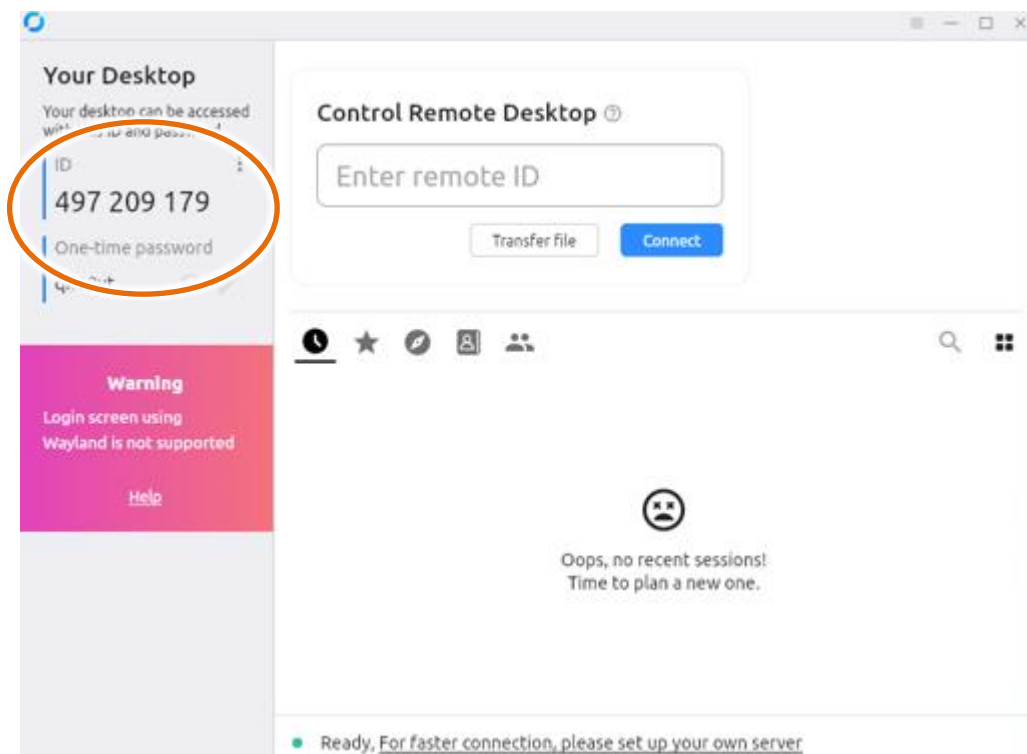


```
pi@pi-desktop: ~  
pi@pi-desktop:~$ sudo dpkg -i rustdesk-1.3.0-aarch64.deb
```

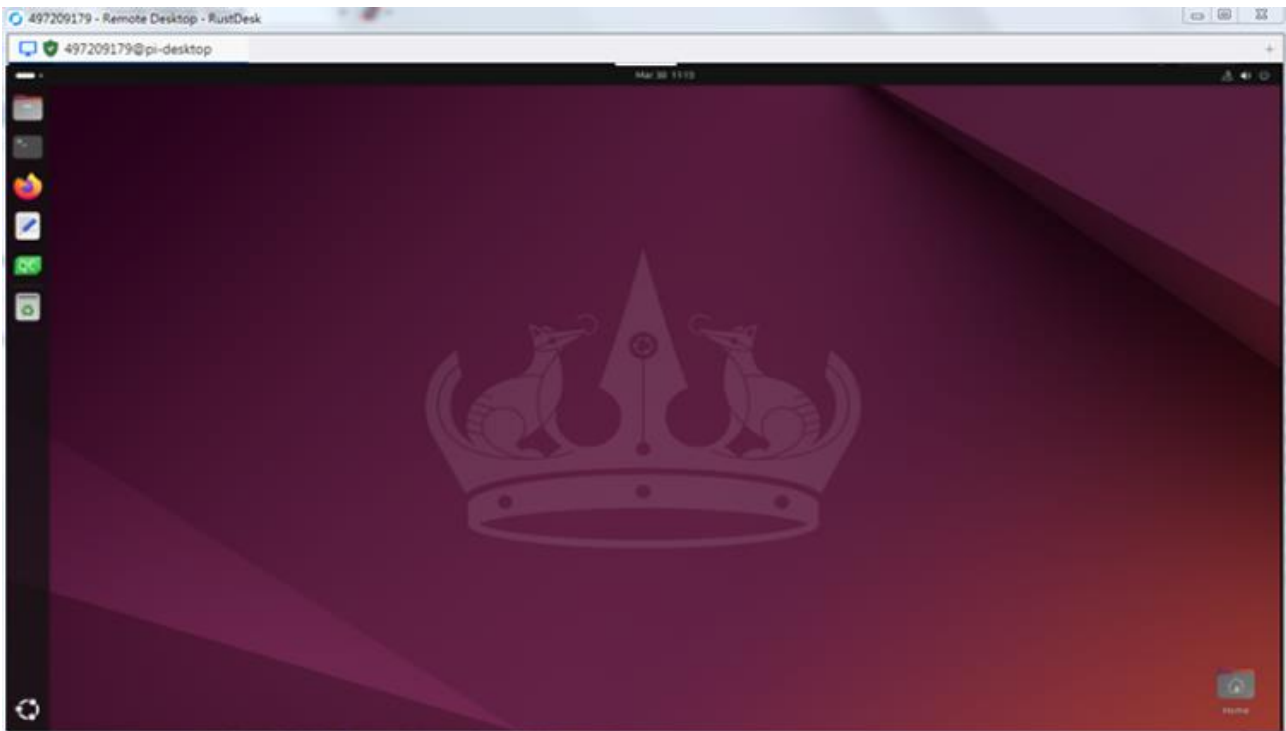
Step 5: After installation is complete, RustDesk will appear in the application launcher



Step 6: RustDesk operates similarly to TeamViewer. Launch RustDesk on the AIB-2541M, and then enter the displayed ID and one-time password into the RustDesk client on the PC side



Step 7: After the RustDesk client on the PC successfully connects, the AIB-2541M desktop will be displayed as shown below



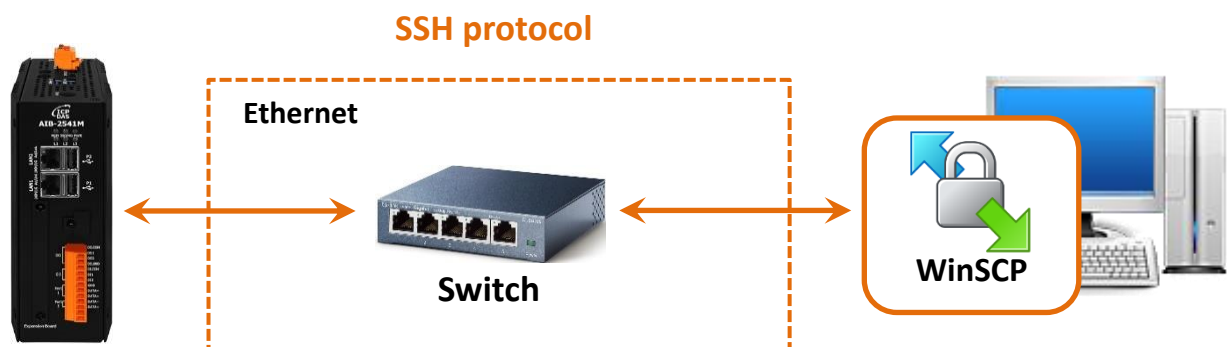
3.3.2. SSH File Transfer

SSH is enabled by default on the AIB-2541M series modules. To transfer files to or from the AIB-2541M over SSH, an SCP/SFTP client is required. This example uses WinSCP to demonstrate how to remotely connect and download files from the AIB-2541M.

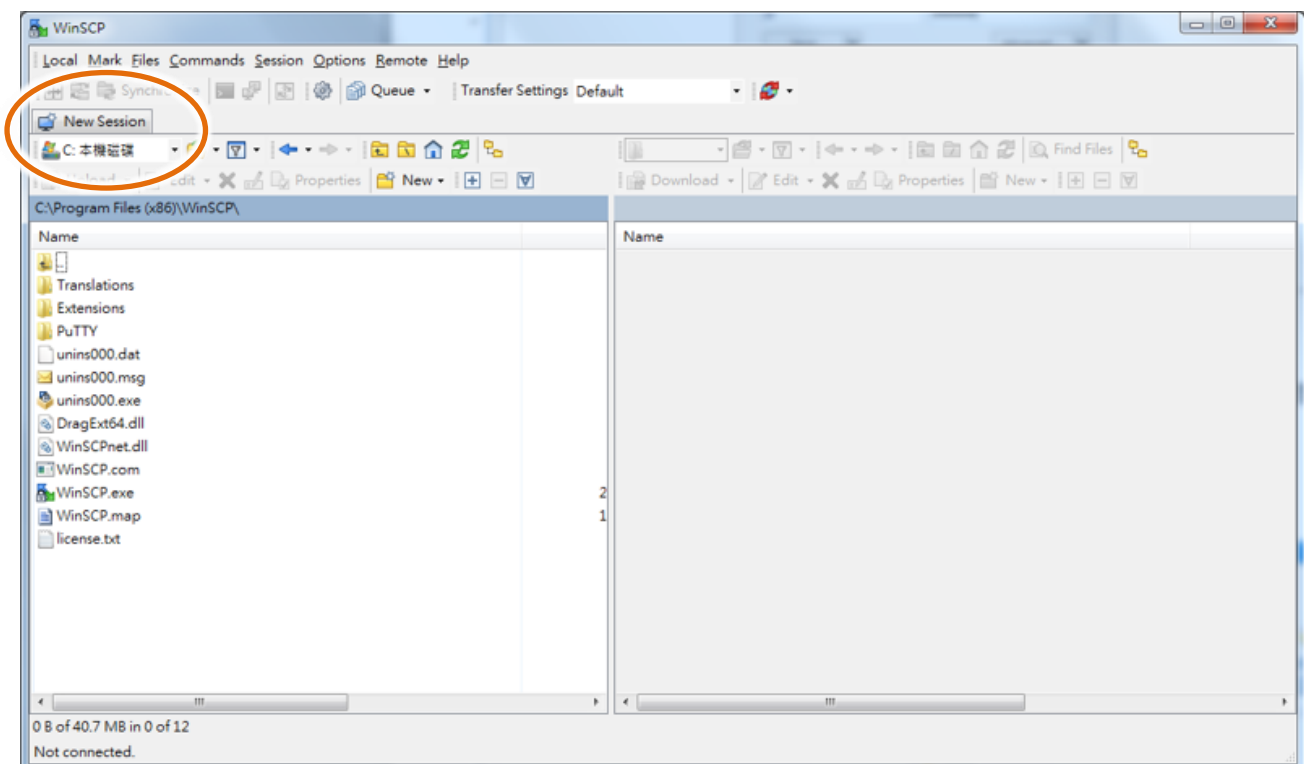
The latest version of WinSCP can be downloaded from:

<https://winscp.net/eng/download.php>

WinSCP and AIB-2541M Architecture

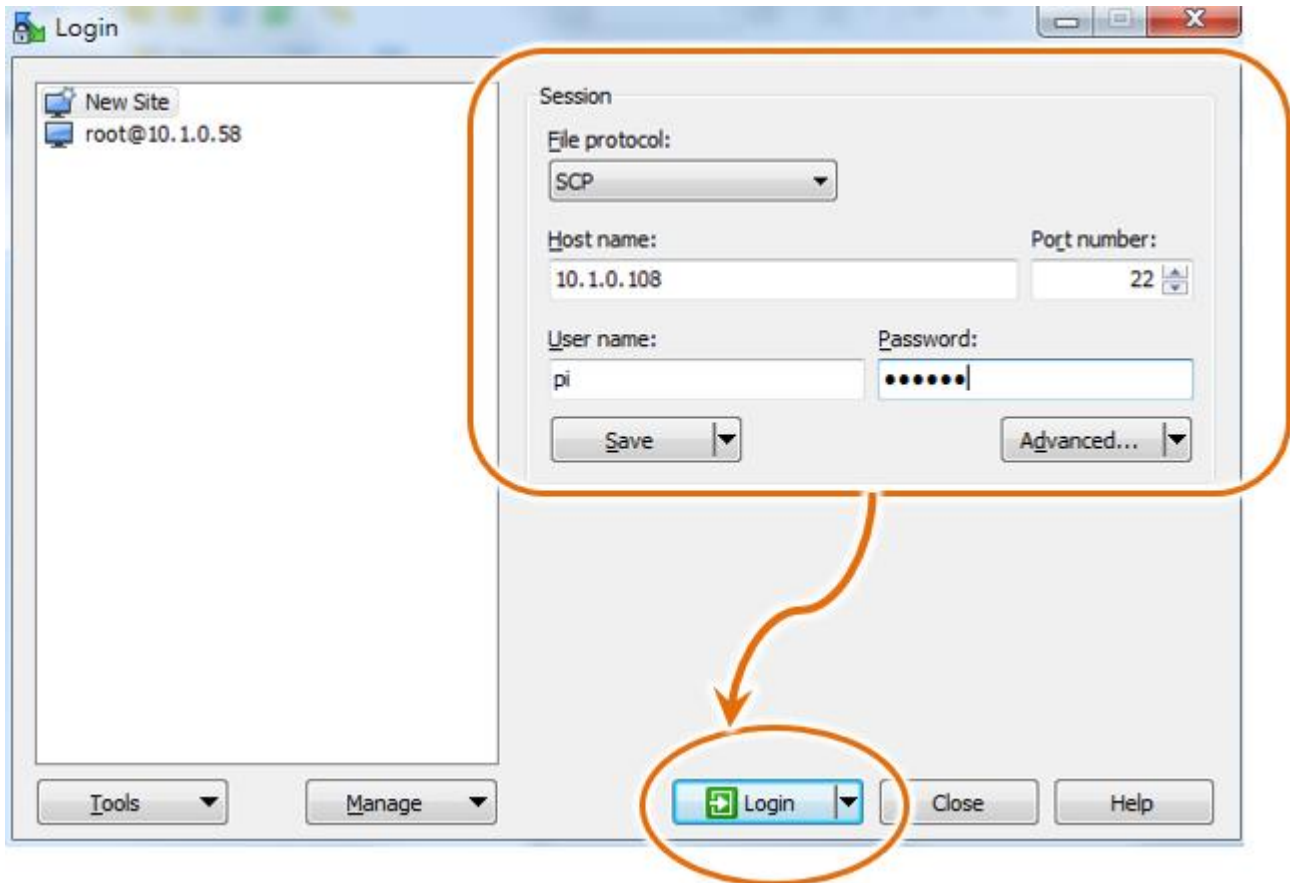


Step 1: Launch WinSCP and click “New Session”

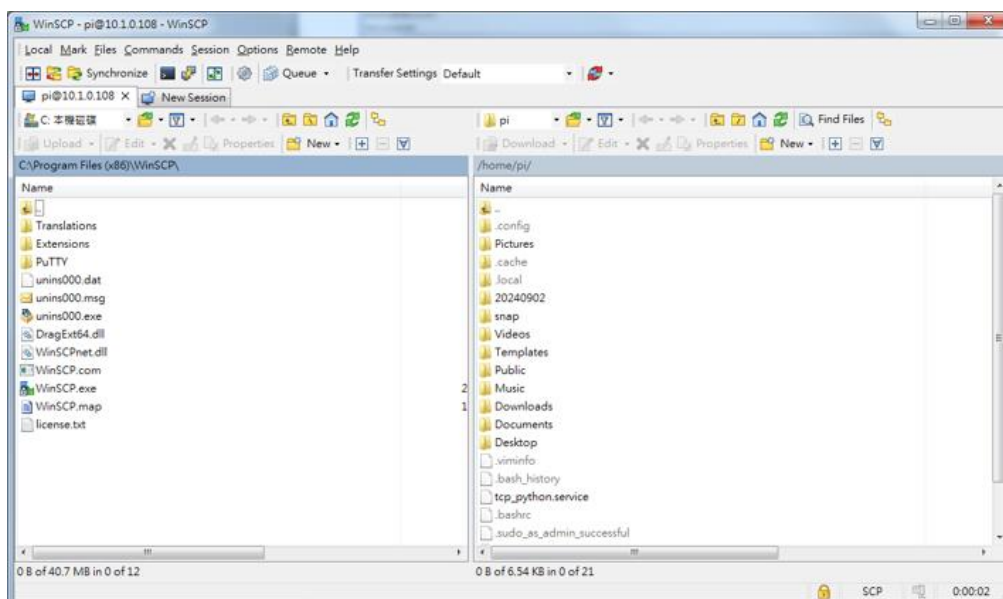


Step 2: In the Login dialog window, enter the following information, then click the Login button

- File protocol: SCP
- Host name: the IP address of the AIB-2541M
- Port number: 22
- User name: pi
- Password: icpdas



Step 3: After the connection is successfully established, file transfer operations can be performed



4. Application Development

This chapter provides guidance on how to create a development environment, and explains the procedures for downloading, installing, and configuring user applications on the AIB-2541M series modules.

Development Tools

The AIB-2541M is an Ubuntu-based product. Ubuntu is a mature Linux operating system that supports rapid software development.

The following table lists the supported development environments and programming languages for AIB-2541M application development.

Development Tool \ Development language	Native SDK	Python Example
Qt6	√	√

4.1. Development Environment Preparation

The AIB-2541M is an Ubuntu-based product. Ubuntu is a mature Linux operating system capable of rapid software development. The standard development environment is a highly integrated toolchain that fully supports application development for the AIB-2541M under Ubuntu.



The AIB-2541M comes pre-installed with Python 3 and Qt. The installed Python version is 3.12, and the Qt version is 6.4.2.

Qt is a cross-platform application development framework commonly used to create graphical user interfaces. When used with PySide6, Qt allows developers to create cross-platform graphical applications in Python.

1. Qt Creator

Qt Creator is pre-installed, as shown in the figure below. Qt Creator is the primary development environment for Qt applications.



2. PySide6

PySide6 is a Python binding for Qt that enables developers to create Qt-based graphical applications in Python. It is pre-installed on the system.

4.2. Creating Your First Python Application

The best way to learn application development on the AIB-2541M series modules is through hands-on practice by creating an actual AIB-2541M application.

The following example demonstrates how to create a demonstration application on the AIB-2541M using Python.

The main steps required to create a Python application are:

1. Create a new project
2. Add control widgets to the window
3. Add event handling for the control widgets
4. Run the application

Each of these steps is described in detail in the following sections.

In this tutorial, it is assumed that Qt 6 and PySide6 are already installed on the AIB-2541M.

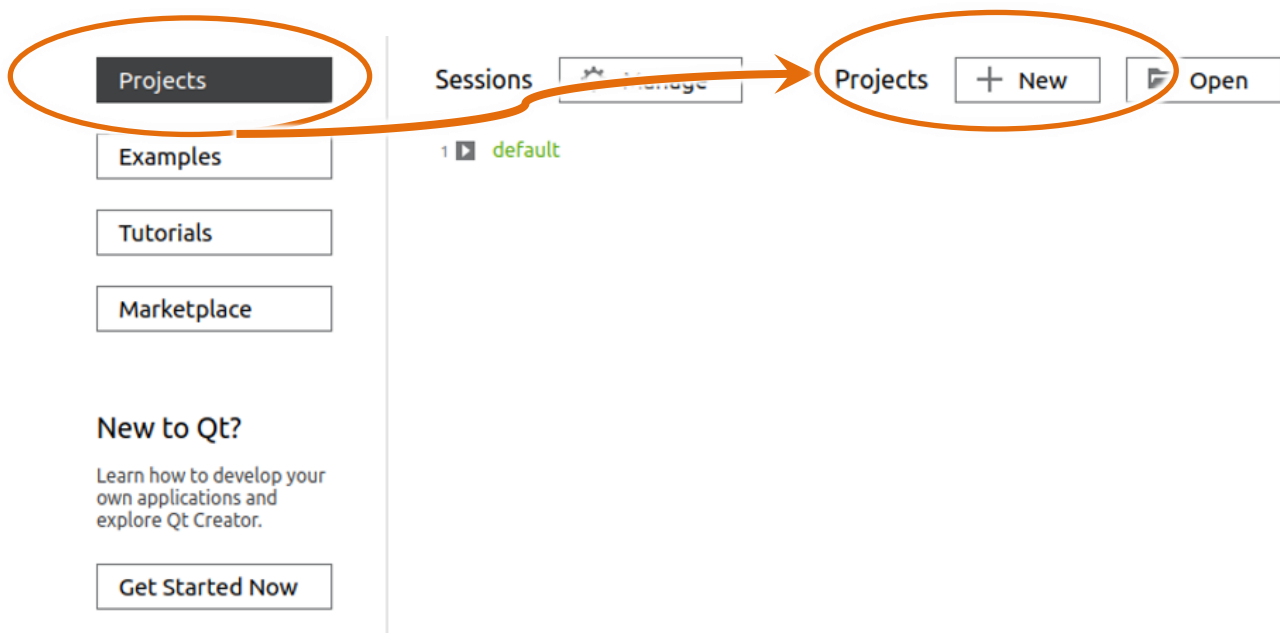
4.2.1. Creating a New Project

This example demonstrates how to create a new custom widget project and a sample user control application.

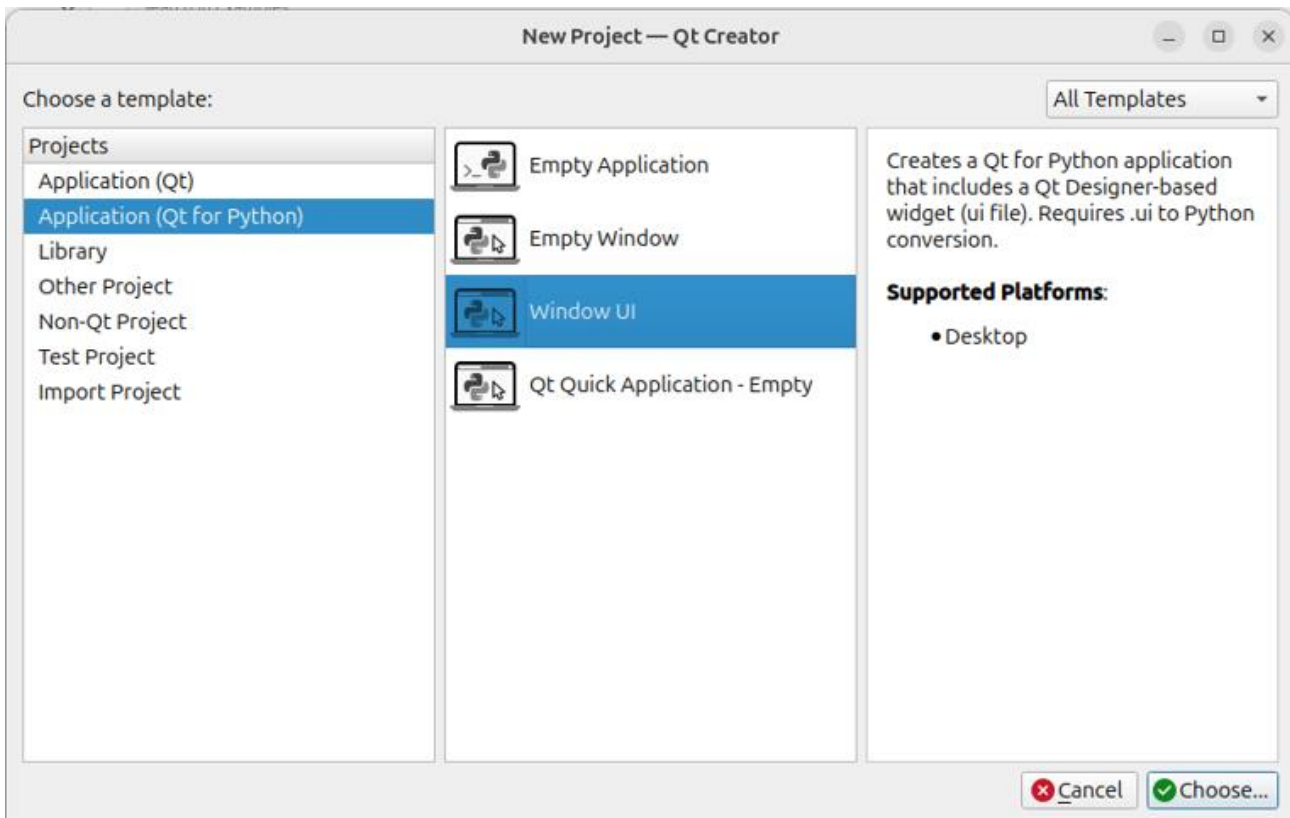
Step 1: Launch Qt Creator



Step 2: In the Project section, click NEW

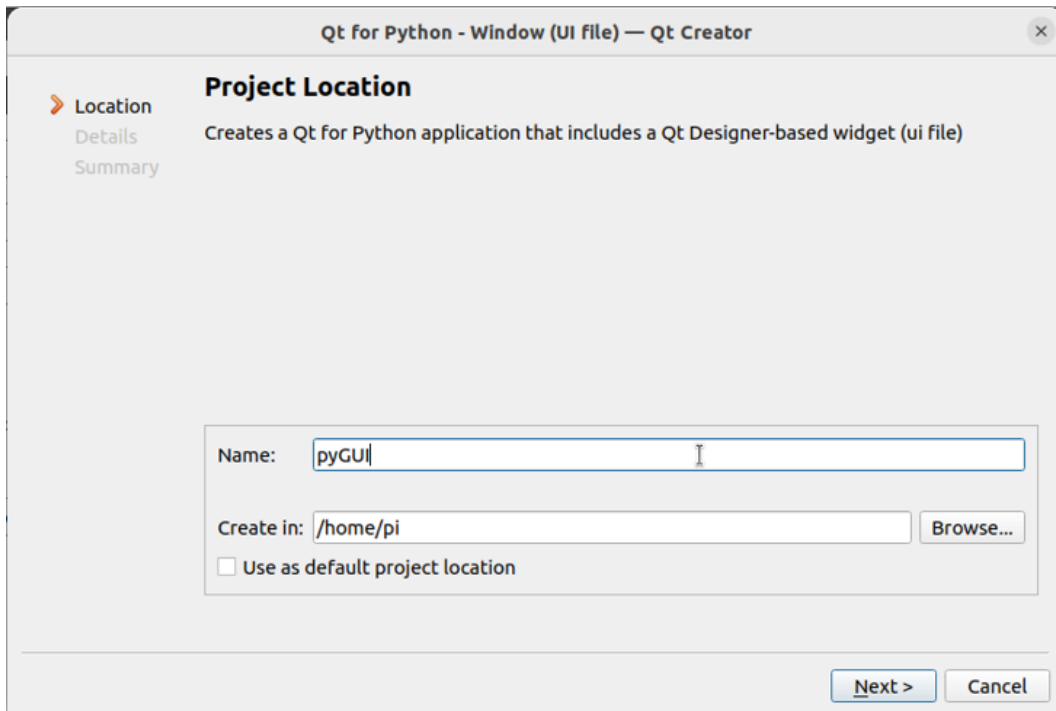


Step 3: Under Application (Qt for Python), select Window UI, and then click Choose



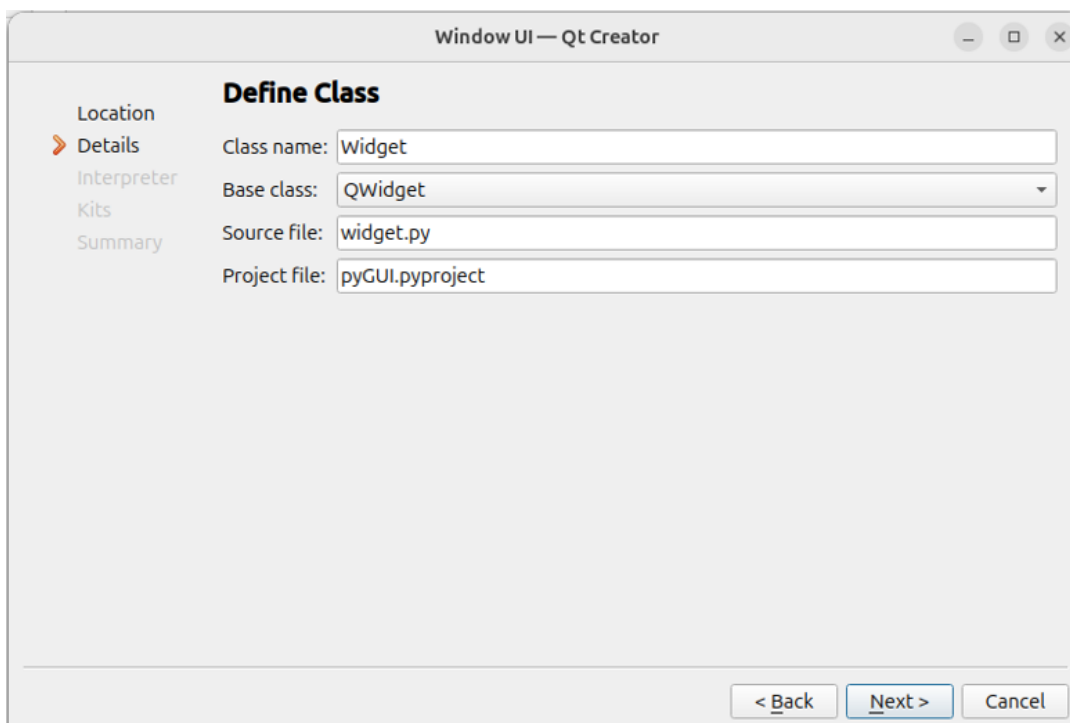
Step 4: Enter the project name in the Name field, and then click the OK button

In this example, the project name pyGUI is used. You may also click the Browse button to select a different project location

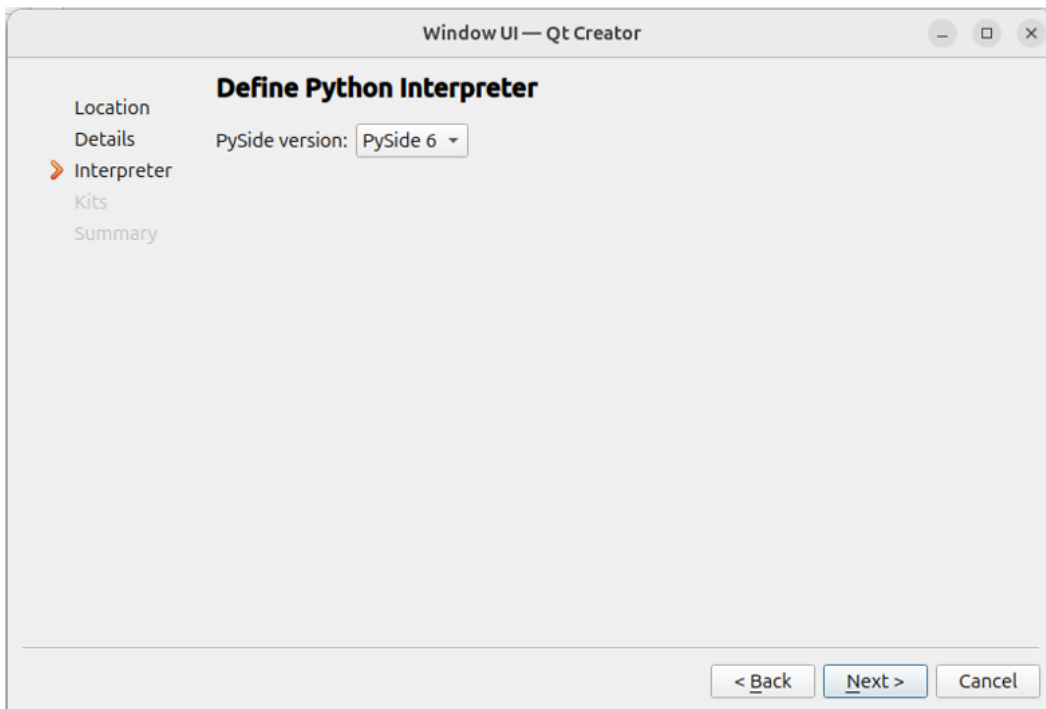


Step 5: Confirm that the project file name matches the previously entered project name

In this example, the name is pyGUI. After confirmation, click “Next”

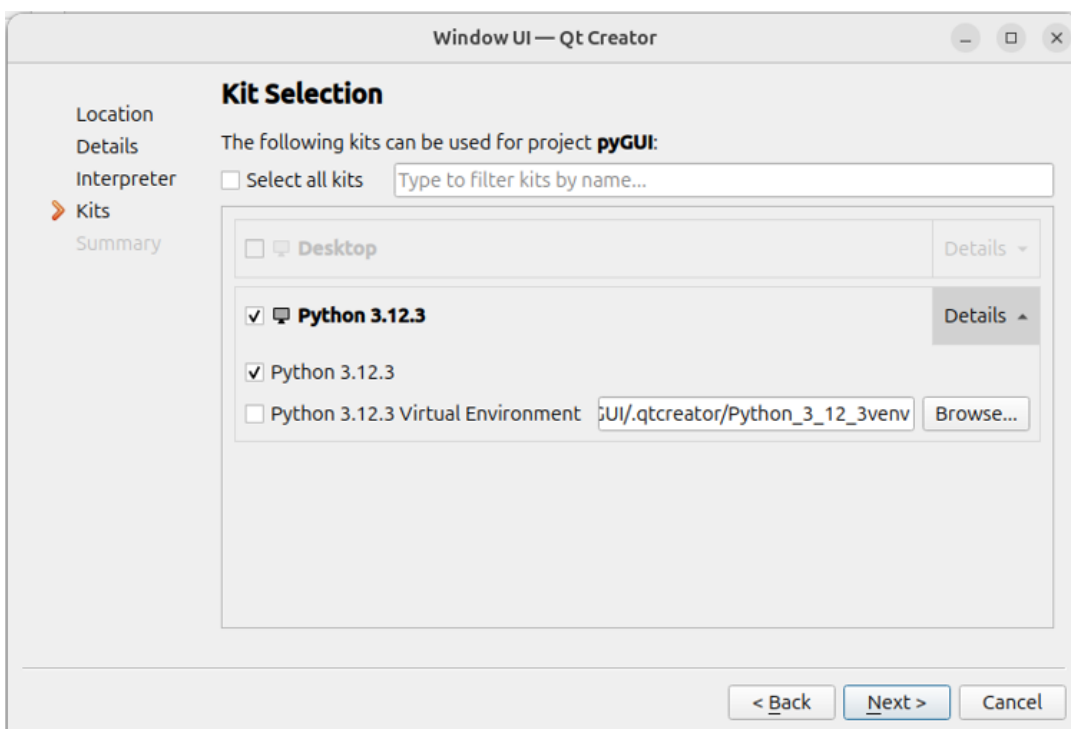


Step 6: In the PySide version field, select PySide 6, and then click “Next”



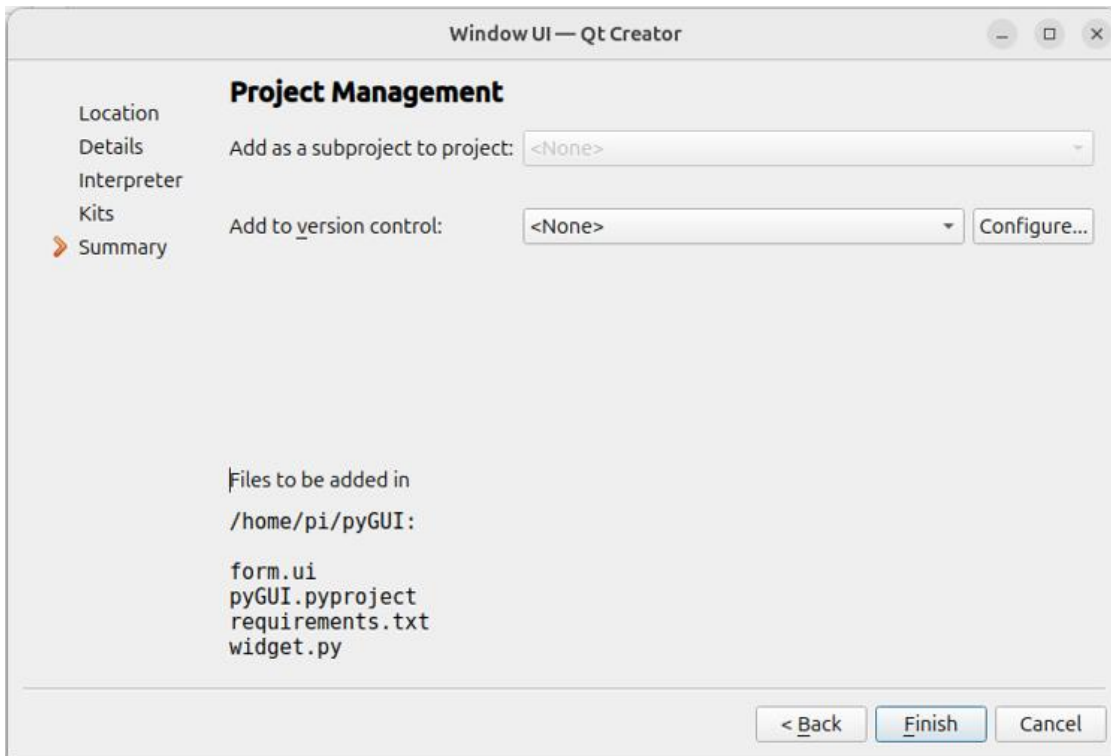
Step 7: By default, the option Python 3.12.3 virtual environment is selected

If you do not wish to test the application in a virtual environment, select Python 3.12.3 instead. In this example, the application will not run in a virtual environment, so Python 3.12.3 is selected. Then click the Next button.

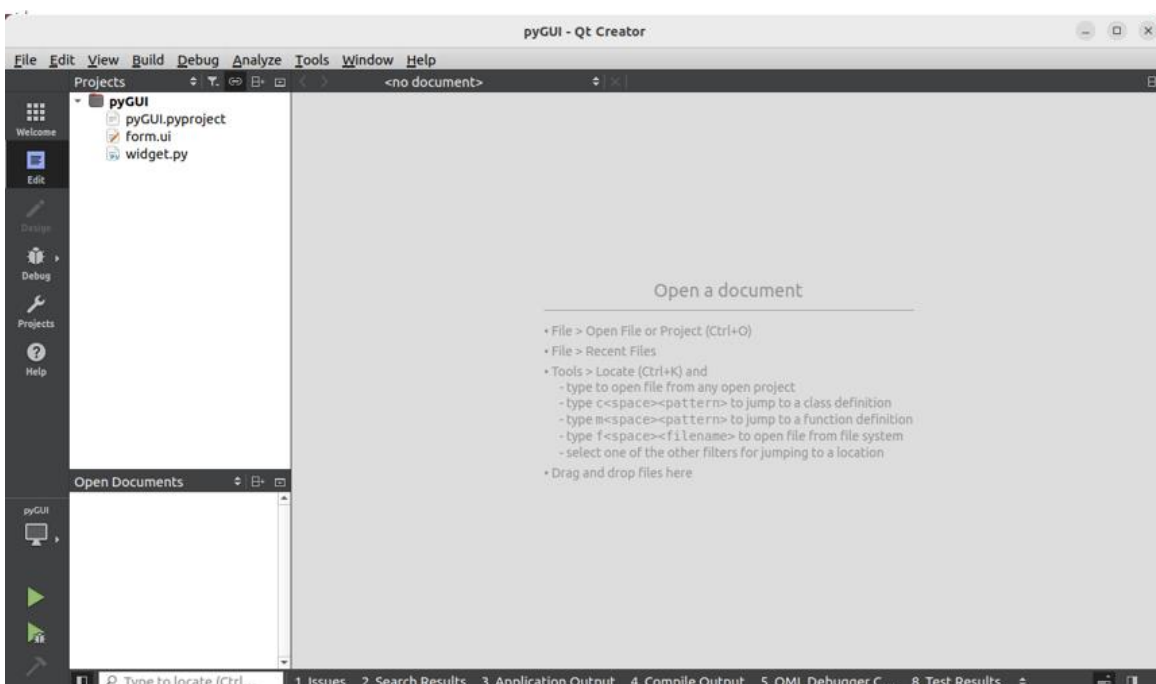


Step 8: In the Add to version control field, select a version control system if required. Otherwise, select None.

In this example, “None” is selected. Click Finish to complete project creation.



Step 9: The created project contains three files: form.ui, widget.py, and <project name>.pyproject. The project structure is shown in the figure below



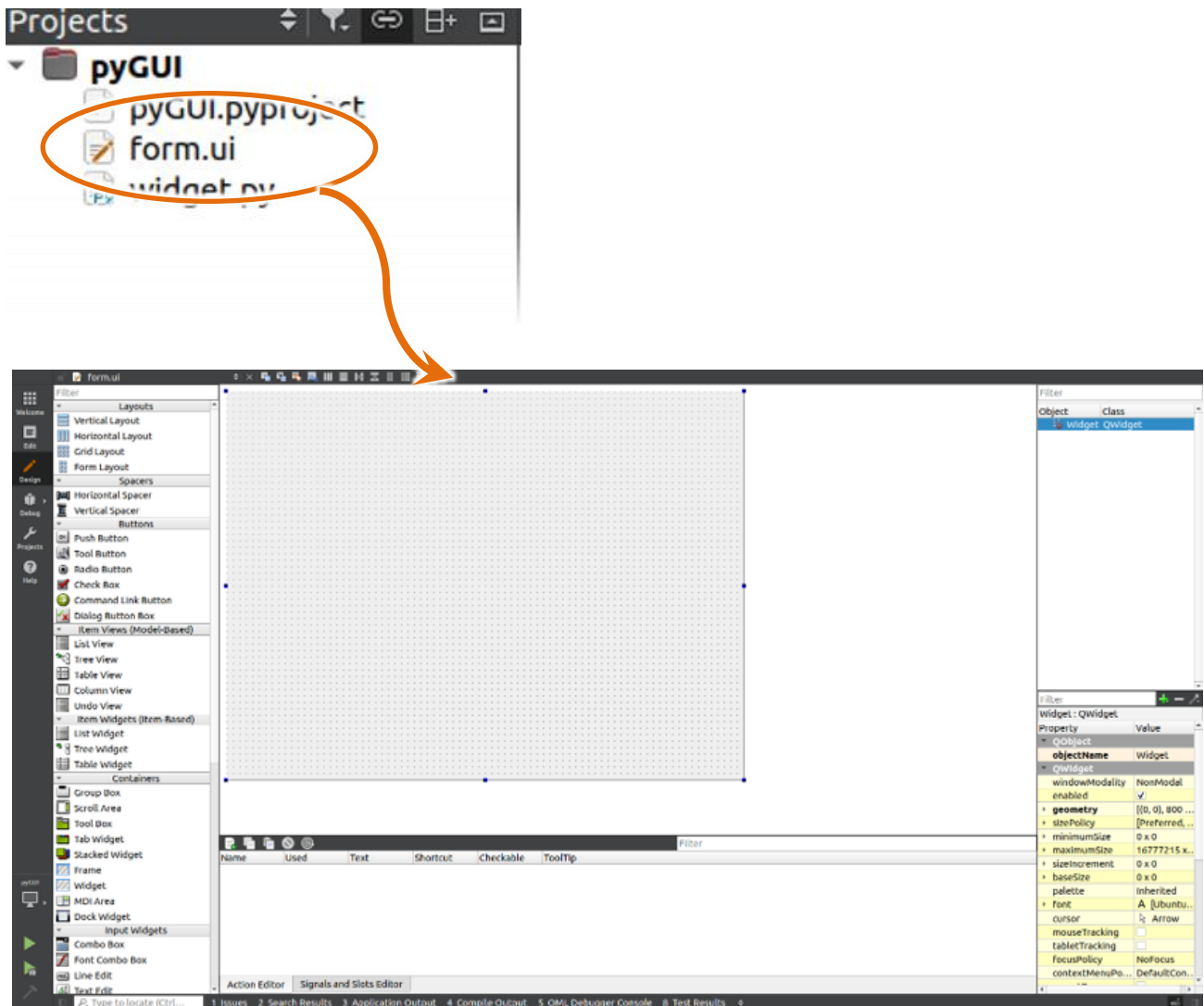
4.2.2. Adding Control Widgets to the Window

You may freely add control widgets from the Toolbox to the form and adjust their positions by drag-and-drop operations.

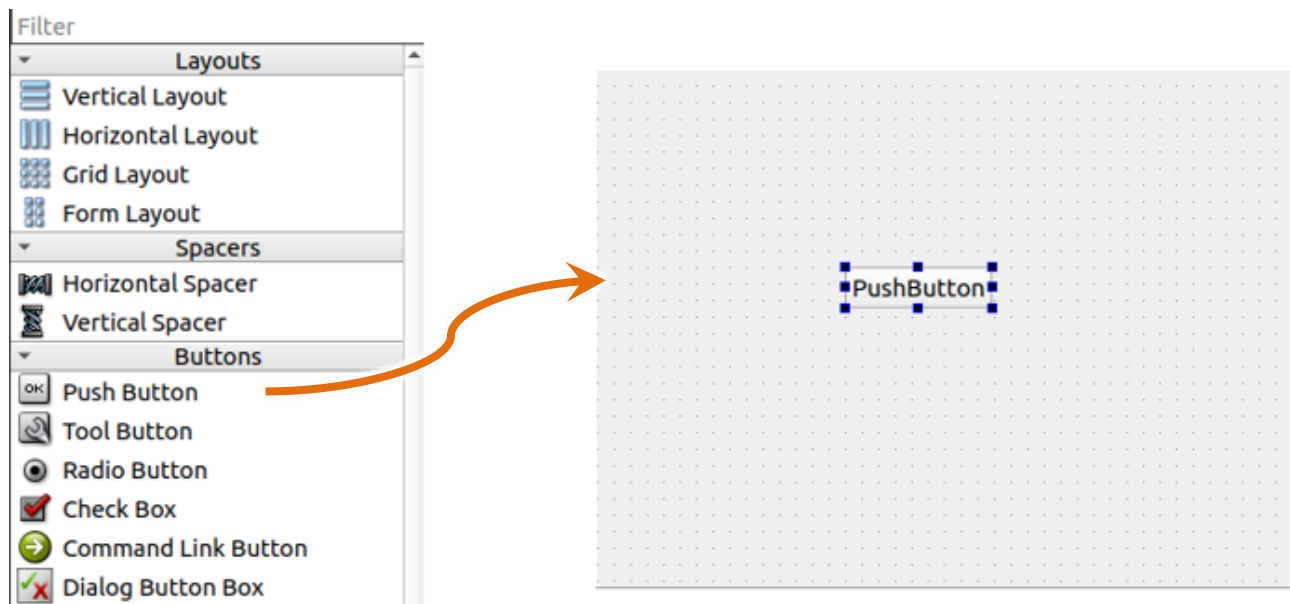
After adding widgets to the window, the corresponding properties can be modified in the Properties window, such as background color or default display text.

The property values configured in the Properties window will be initialized automatically when the application runs.

Step 1: Open the project form page and open the “form.ui” file



Step 2: In the Toolbox, drag the Push Button widget onto the window



Step 3: In the Properties window, set the objectName property to “btnClick” and then set the text property to “Show Message”

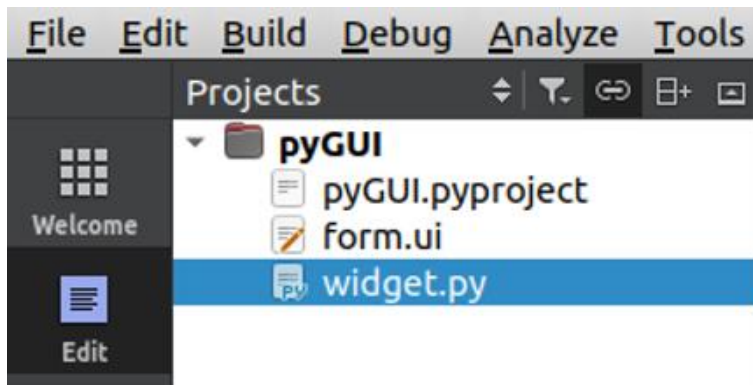
Property	Value
QObject	
objectName	btnClick
QWidget	
enabled	☒
geometry	[(150, 120), ...]
sizePolicy	[Minimum, ...]
minimumSize	0 x 0
maximumSize	16777215 x ..
sizeIncrement	0 x 0
baseSize	0 x 0
palette	Inherited
font	A [Ubuntu..
cursor	Arrow
mouseTracking	☐
tabletTracking	☐
focusPolicy	StrongFocus
contextMenuPo...	DefaultCon..
accentDrops	☐

QAbstractButton	
text	Show Message
icon	
iconSize	16 x 16
shortcut	
checkable	☐
checked	☐
autoRepeat	☐
autoExclusive	☐
autoRepeatDelay	300
autoRepeatInte...	100

4.2.3. Adding Events to the Control Widgets

After completing the application window design, you may begin writing the application functionality code.

Step 1: Select Edit mode and open “widget.py”



The source code shown in the following figure will appear.

```
< > widget.py Python 3.12.3 REPL
1  # This Python file uses the following encoding: utf-8
2  import sys
3
4  from PySide6.QtWidgets import QApplication, QWidget
5
6  # Important:
7  # You need to run the following command to generate the ui_form.py file
8  #     pyside6-uic form.ui -o ui_form.py, or
9  #     pyside2-uic form.ui -o ui_form.py
10 from ui_form import Ui_Widget
11
12 class Widget(QWidget):
13     def __init__(self, parent=None):
14         super().__init__(parent)
15         self.ui = Ui_Widget()
16         self.ui.setupUi(self)
17
18
19 if __name__ == "__main__":
20     app = QApplication(sys.argv)
21     widget = Widget()
22     widget.show()
23     sys.exit(app.exec())
24
```

Step 2: Modify “widget.py” as follows:

```
from PySide6.QtWidgets import QApplication, QWidget, QMessageBox
from ui_form import Ui_Widget

class Widget(QWidget):
    def __init__(self, parent=None):
        super().__init__(parent)
        self.ui = Ui_Widget()
        self.ui.setupUi(self)
        self.ui.btnClick.clicked.connect(self.show_message)

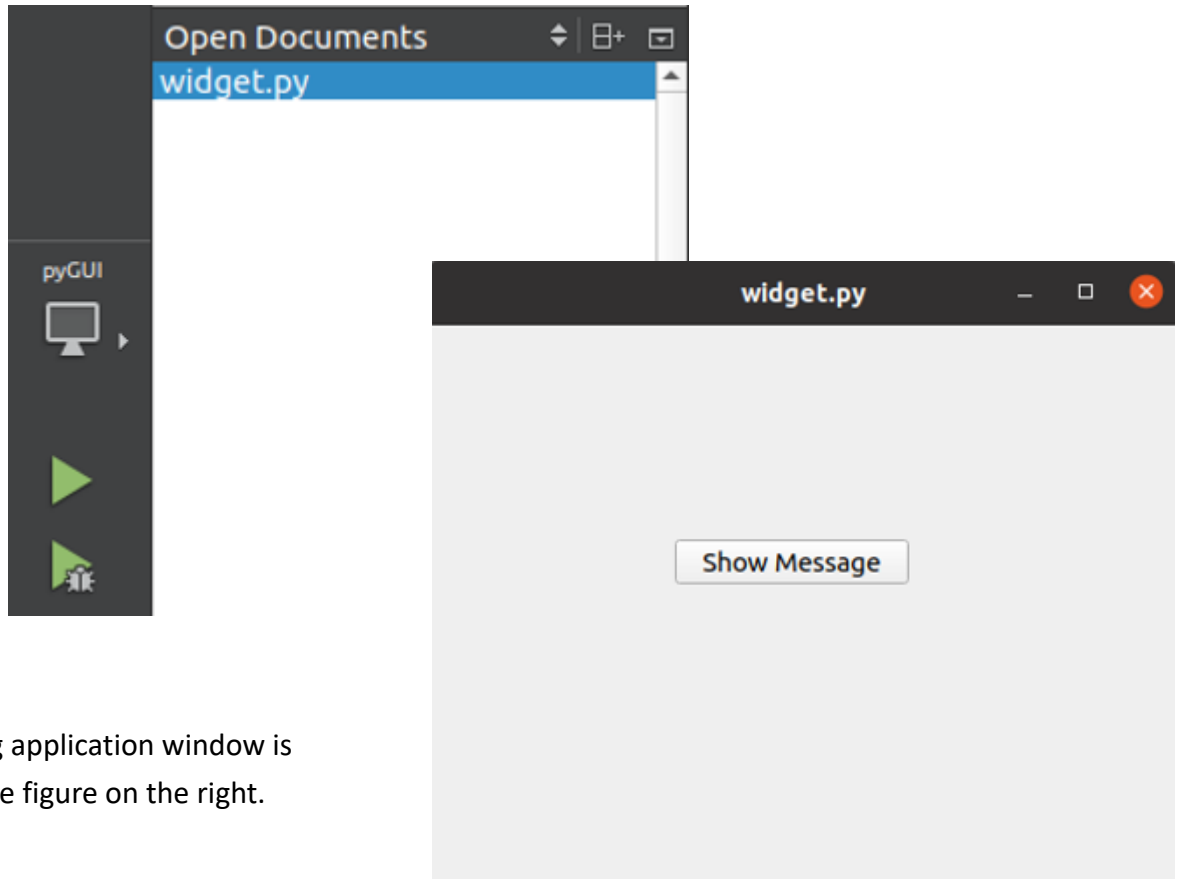
    def show_message(self):
        QMessageBox.information(self, "info", "first program")
```

The completed source code is shown in the figure below.

```
1  # This Python file uses the following encoding: utf-8
2  import sys
3
4  from PySide6.QtWidgets import QApplication, QWidget, QMessageBox
5
6  # Important:
7  # You need to run the following command to generate the ui_form.py file
8  #     pyside6-uic form.ui -o ui_form.py, or
9  #     pyside2-uic form.ui -o ui_form.py
10 from ui_form import Ui_Widget
11
12 class Widget(QWidget):
13     def __init__(self, parent=None):
14         super().__init__(parent)
15         self.ui = Ui_Widget()
16         self.ui.setupUi(self)
17         self.ui.btnClick.clicked.connect(self.show_message)
18
19     def show_message(self):
20         QMessageBox.information(self, "info", "first program")
21
22
23 if __name__ == "__main__":
24     app = QApplication(sys.argv)
25     widget = Widget()
26     widget.show()
27     sys.exit(app.exec())
28
```

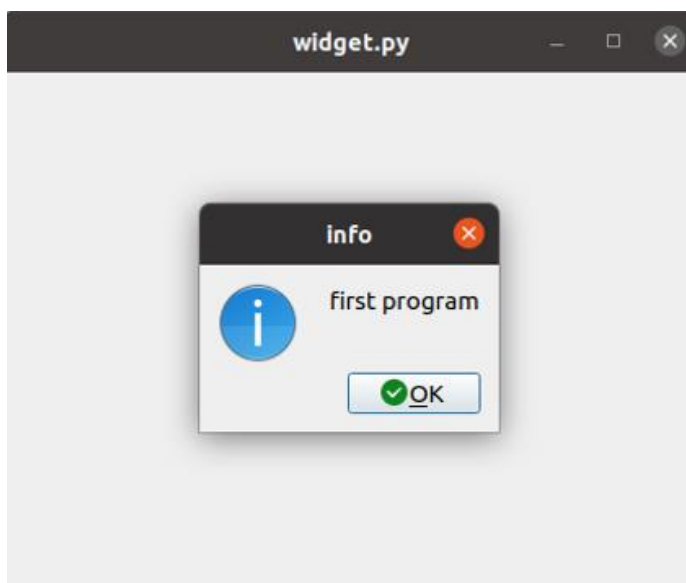
4.2.4. Running the Application

In the lower-right corner of Qt Creator, click `widget.py` to specify that `widget.py` will be executed next. Then click the Run icon  to start the application.



The running application window is shown in the figure on the right.

After pressing the “Show Message” button, the following MessageBox dialog will appear.



4.2.5. Python Examples

These examples are intended for use in the Python programming environment. The latest version can be downloaded from the ICP DAS website.

Python example programs are available at:

<https://www.icpdas.com/tw/download/index.php?model=AIB-2000%20Series>

The following example programs are included:

di.py: Reads the DI port status on the AIB-2541M.

do.py: Turns the DO ports on or off.

led.py: Turns LEDs L1–L3 on or off.

ser.py: Performs read/write operations on Port 1 and Port 2.

serial_number.py: Reads the serial number of the AIB-2541M.

watchdog_enable_refresh.py: Starts and updates the watchdog.

watchdog_enable_disable.py: Starts and stops the watchdog.

5. Selecting I/O Expansion Modules and SDKs

This chapter introduces how to select suitable I/O expansion modules and corresponding SDKs, allowing users to more flexibly utilize the AIB-2541M series modules and develop AIB-2541M applications more comprehensively.

The AIB-2541M supports the following I/O expansion configurations:



USB (USB-2000 Series and USB-87Pn Series)

The AIB-2541M can connect to USB-2000 and USB-87Pn modules through USB interfaces together with ICP DAS I-87k series I/O modules.

Module	Native SDK	Python Example
USB-2000 Series	Python Example	Python Example
USB-87Pn Series	Python Example	Python Example

Ethernet (ET-7000 Series and I-8KE4/8-MTCP)

The AIB-2541M can connect to ET-7000 and I-8KE4/8-MTCP modules through Ethernet interfaces together with ICP DAS I-87k series I/O modules, and supports the Modbus/TCP communication protocol.

Module	Native SDK	Python Example
ET-7000	Modbus Example	Modbus Example
I-8KE4/8-MTCP	Modbus Example	Modbus Example

RS-485 (I-7000 Series and M-7000 Series)

The AIB-2541M can connect to I-7000, M-7000, and RU-87Pn modules through RS-485 interfaces together with ICP DAS I-87k series I/O modules. The RS-485 network uses twisted-pair cables as the transmission medium and supports data rates of up to 115.2 kbps.

I-7000 Series I/O Modules

Module	Native SDK	Python Example
I-7000 Series	libAIB.so	libAIB.py
I-7088 (D)	libAIB.so	libAIB.py

For more information regarding I-7000 series I/O modules and example programs, please refer to: [https://www.icpdas.com/tw/product/guide+Remote I O Module and Unit+RS-485 I O Modules+I-7000](https://www.icpdas.com/tw/product/guide+Remote+I+O+Module+and+Unit+RS-485+I+O+Modules+I-7000)

M-7000 Series I/O Modules

Module	Native SDK	Python Example
M-7000 Series	Modbus Example	Modbus Example

For more information regarding M-7000 series I/O modules and example programs, please refer to: [https://www.icpdas.com/tw/product/guide+Remote I O Module and Unit+RS-485 I O Modules+I-7000](https://www.icpdas.com/tw/product/guide+Remote+I+O+Module+and+Unit+RS-485+I+O+Modules+I-7000)

RU-87Pn + I-87K Series I/O Modules

Module	Native SDK	Python Example
RU-87Pn + I-87K	libAIB.so	libAIB.py

Other Specified I/O

Module	Native SDK	Python Example
I-7088W	libAIB.so	libAIB.py

Local I/O Expansion Modules (XV-Board)

The AIB-2541M supports one XV-Board/XB-Board module.

XV-Board

Module	Native SDK	Python Example
XV-Board	Modbus Demo	Modbus Demo

For more supported module information, please refer to:

[https://www.icpdas.com/tw/product/guide+Remote I O Module and Unit+PAC I O Modules+XV-board](https://www.icpdas.com/tw/product/guide+Remote+I+O+Module+and+Unit+PAC+I+O+Modules+XV-board)

XB-Board

Module	Native SDK	Python Example
XB-Board	Modbus Demo	Modbus Demo

For more supported module information, please refer to:

[https://www.icpdas.com/tw/product/guide+Remote I O Module and Unit+PAC I O Modules+XB-board](https://www.icpdas.com/tw/product/guide+Remote+I+O+Module+and+Unit+PAC+I+O+Modules+XB-board)

6. System Update and Upgrade

This chapter provides the procedures required to update the AIB-2541M operating system (OS).

ICP DAS will continue adding additional features to the AIB-2541M OS in the future. Therefore, users are encouraged to regularly check the ICP DAS website for the latest updates.

The AIB-2541M OS is updated by flashing the system image using Raspberry Pi Imager. Raspberry Pi Imager must first be installed on a PC before updating the OS.

The latest OS installation image can be downloaded from the ICP DAS website:

<https://www.icpdas.com/tw/download/index.php?model=AIB-2000%20Series>

Step 1: Install Raspberry Pi Imager and rpiboot

The latest version of Raspberry Pi Imager can be downloaded from:

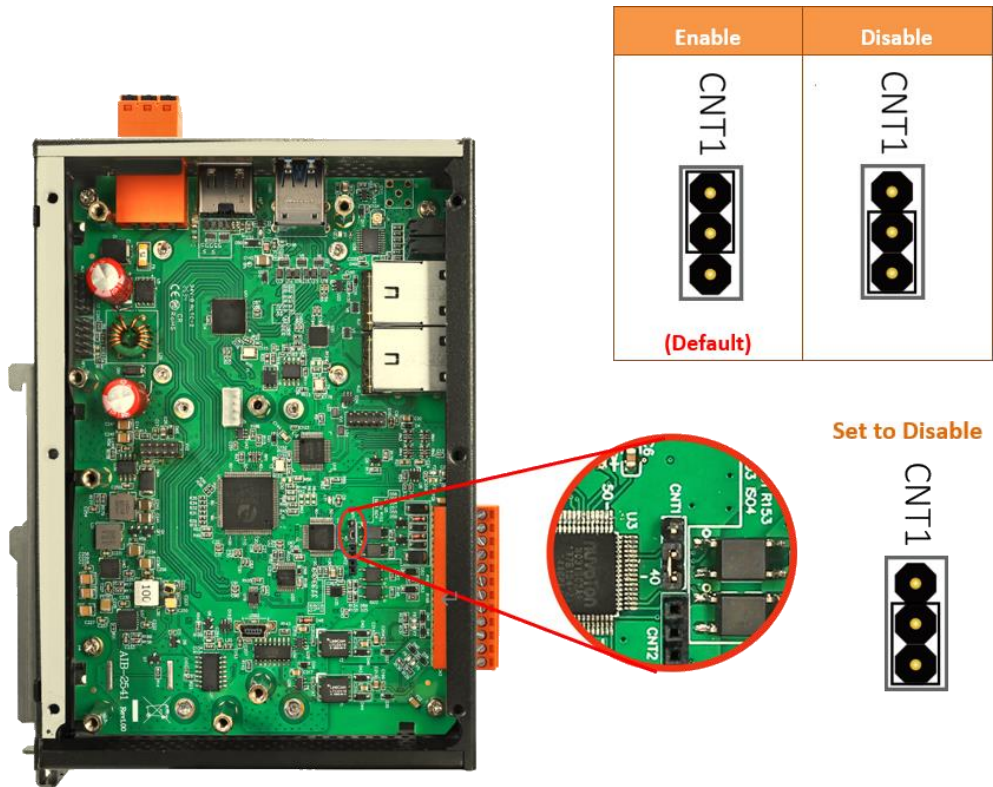
<https://www.raspberrypi.com/software/>

The latest version of rpiboot can be downloaded from:

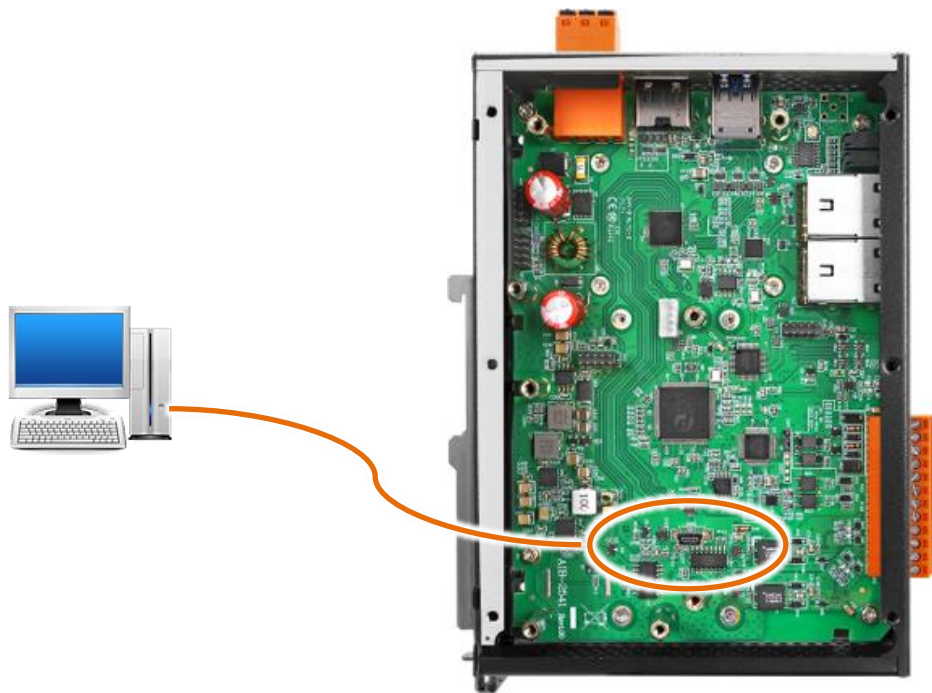
<https://github.com/raspberrypi/usbboot/releases>

If Raspberry Pi Imager and rpiboot are already installed, this step may be skipped.

Step 2: Disable the AIB-2541M hardware watchdog. Configure the hardware jumper as shown in the figure below.

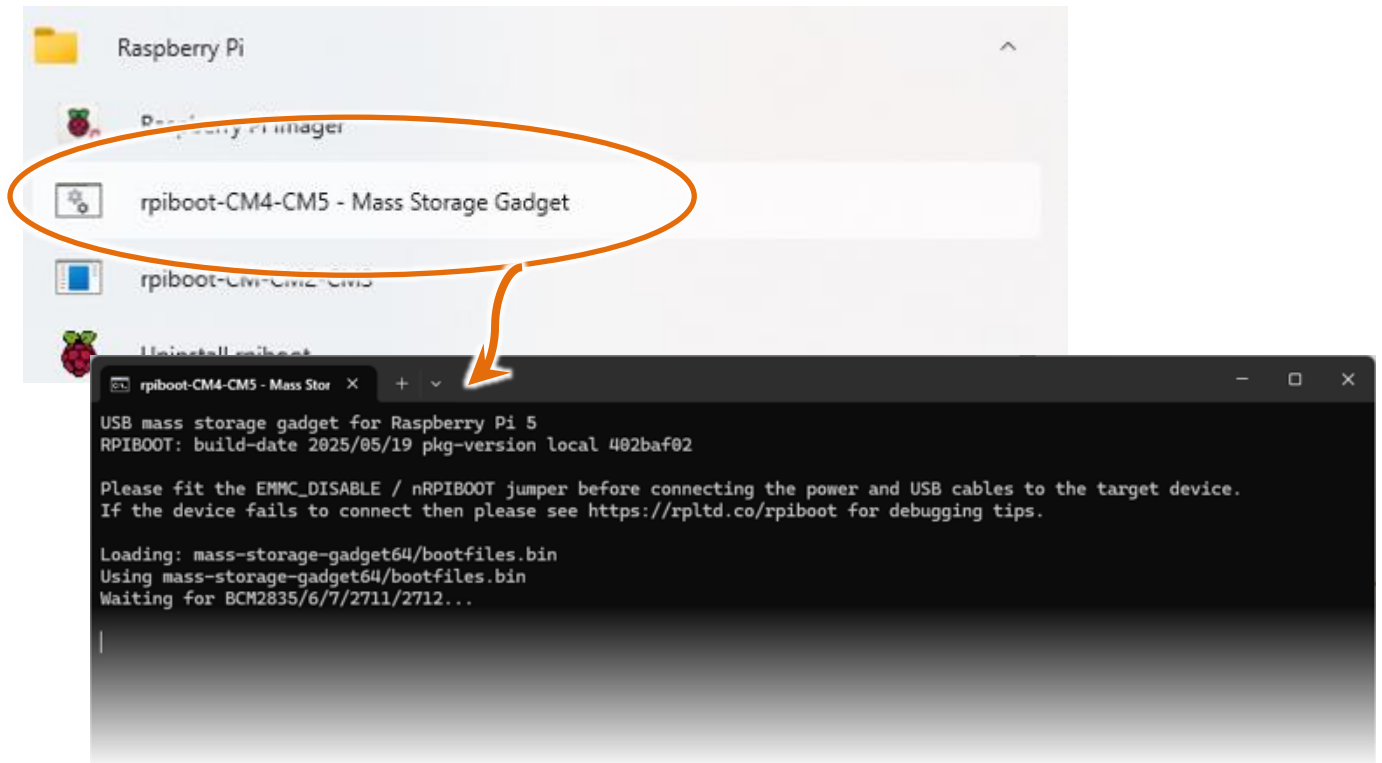


Step 3: Connect the mini USB port of the AIB-2541M to the PC



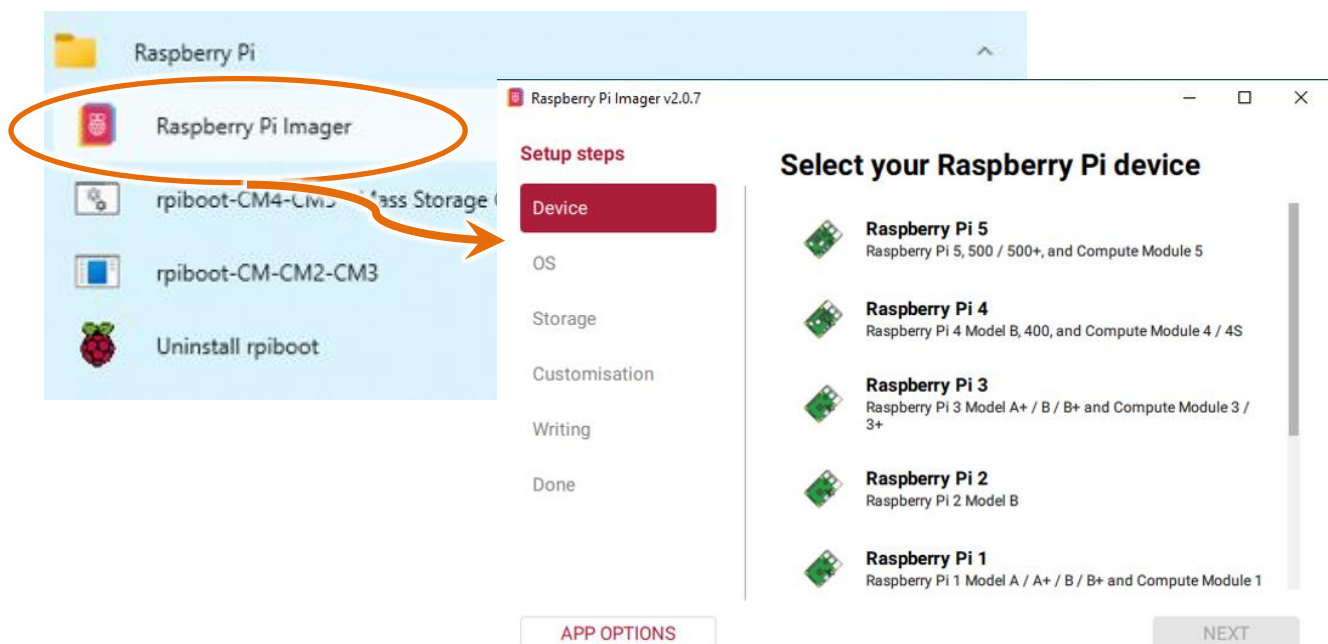
Step 4: Launch rpiboot on the PC

Wait for rpiboot to mount the eMMC storage. After mounting is completed, launch Raspberry Pi Imager to flash the OS.

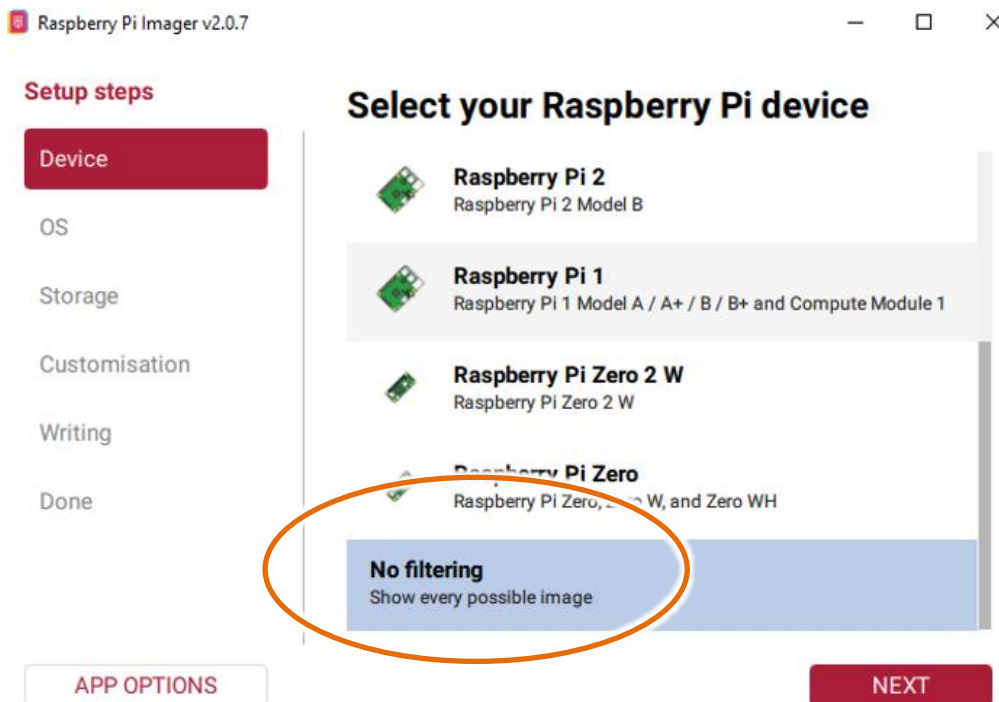


Step 5: Launch Raspberry Pi Imager

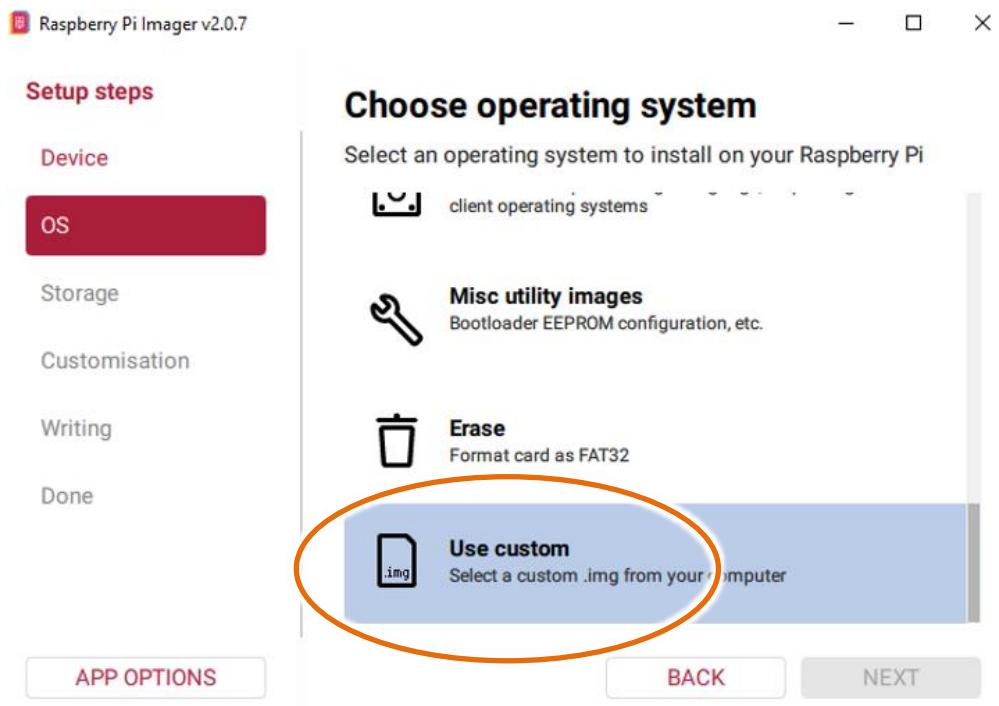
Raspberry Pi Imager requires configuration of three items. Please follow the steps below.



Step 6: In the Raspberry Pi Device field, select No Filtering



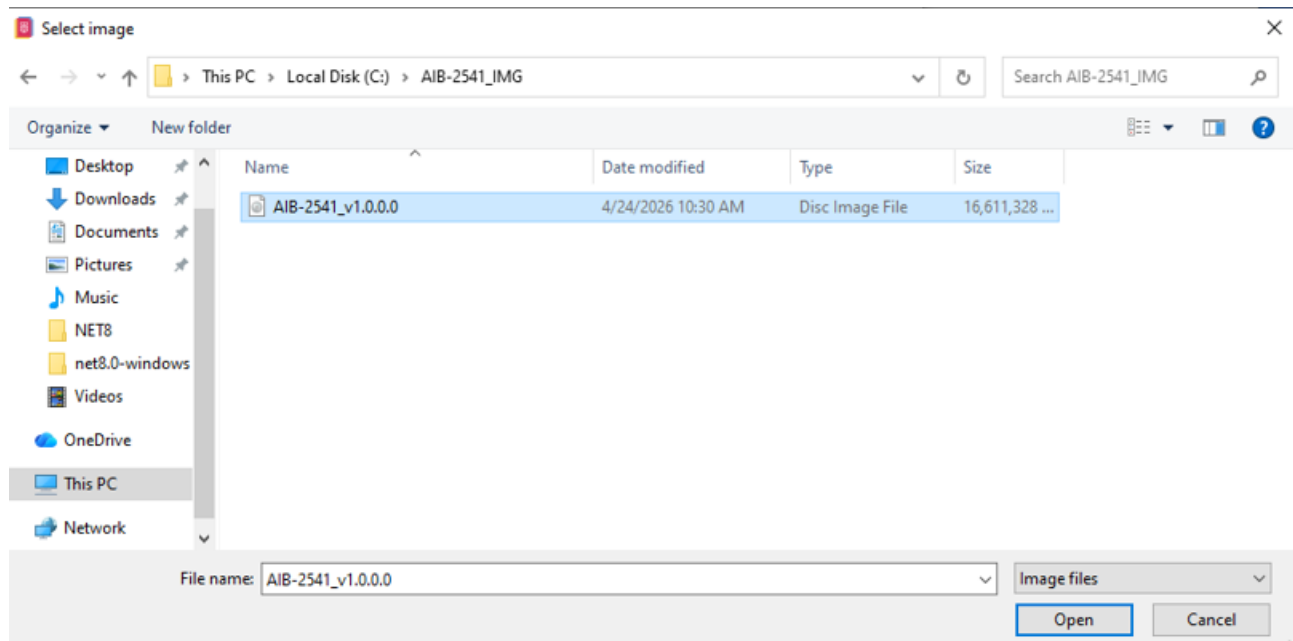
Step 7: In the Operating system field, select “Use custom”



Step 8: Select the OS image downloaded from

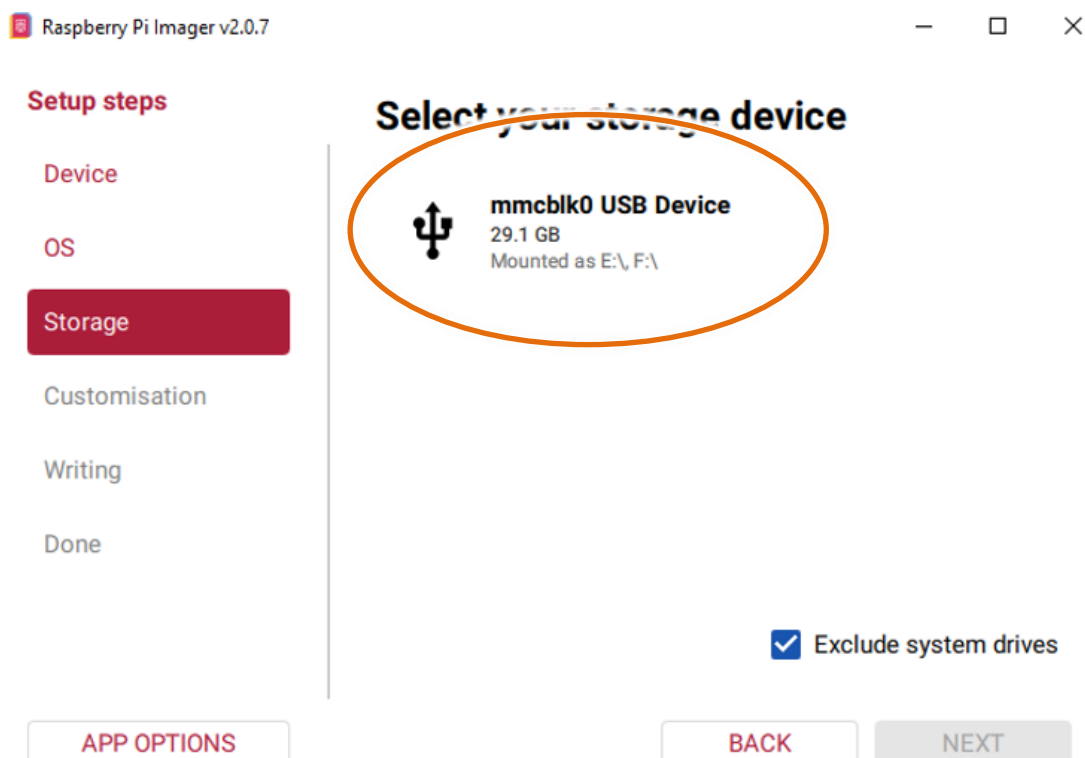
<https://www.icpdas.com/tw/download/index.php?model=AIB-2000%20Series>

In this example, the initial OS image AIB-2541_v1.0.0.0.img is used.

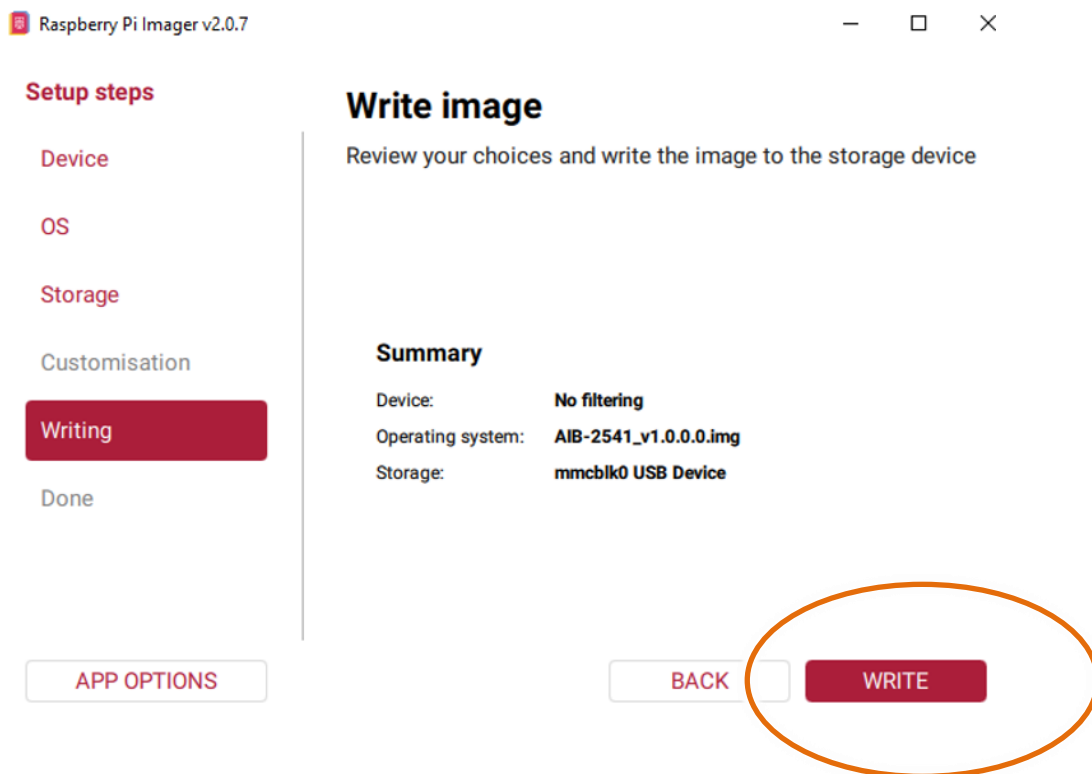


Step 9: In the Storage field, select mmcblk0, and then click NEXT

“mmcblk0” is the device name of the eMMC mounted by rpiboot.

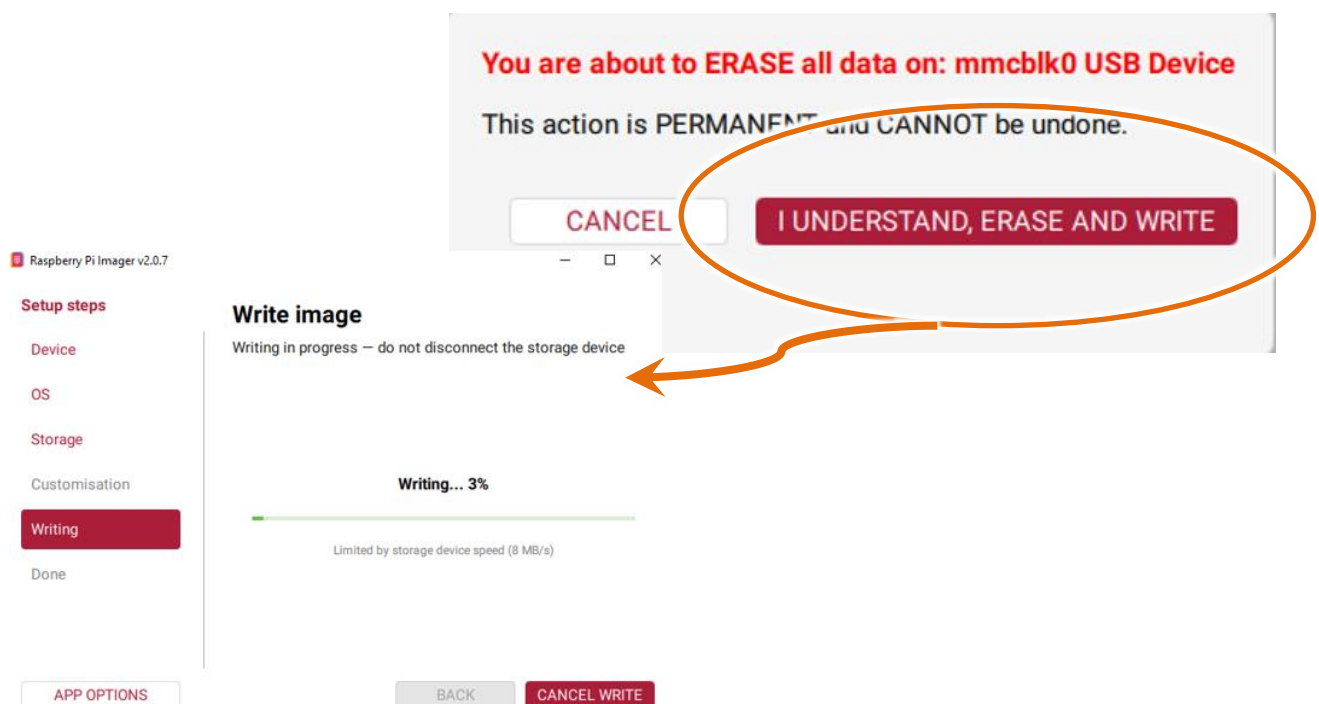


Step 10: Click WRITE to proceed to the next step



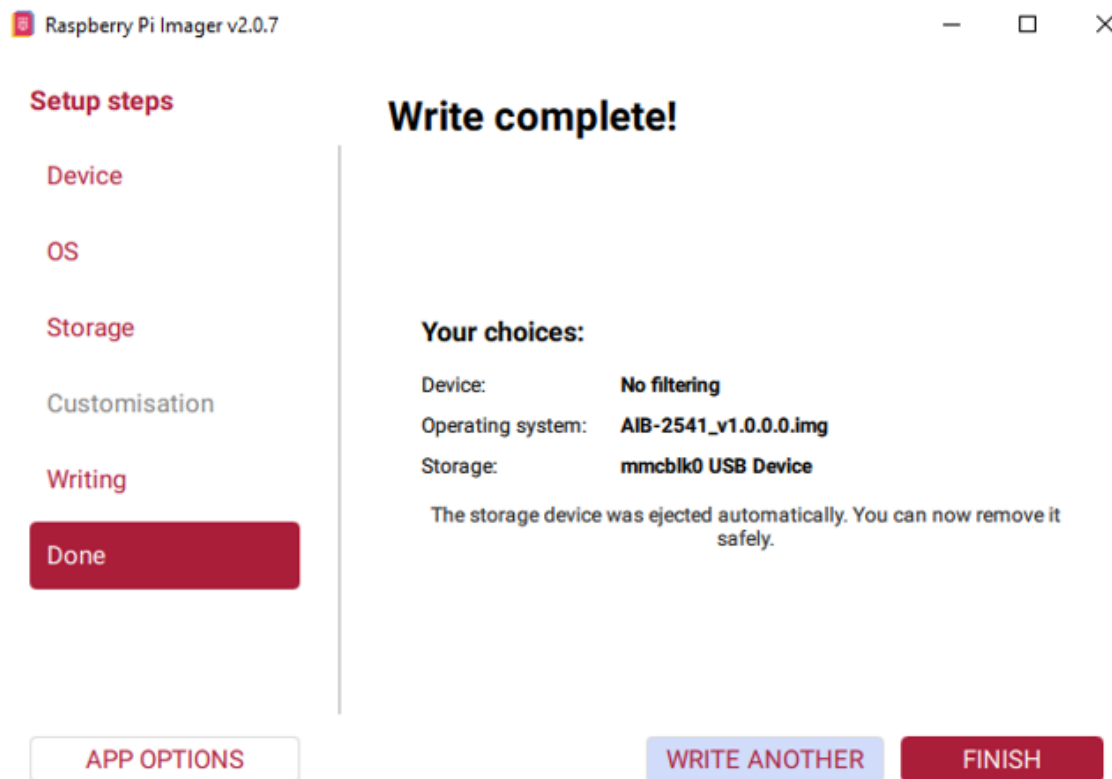
Step 11: A warning message will indicate that the disk will be erased. If the data has already been backed up, select "I UNDERSTAND, ERASE AND WRITE". The OS flashing process will then begin

The flashing process takes approximately 30 to 40 minutes to complete.



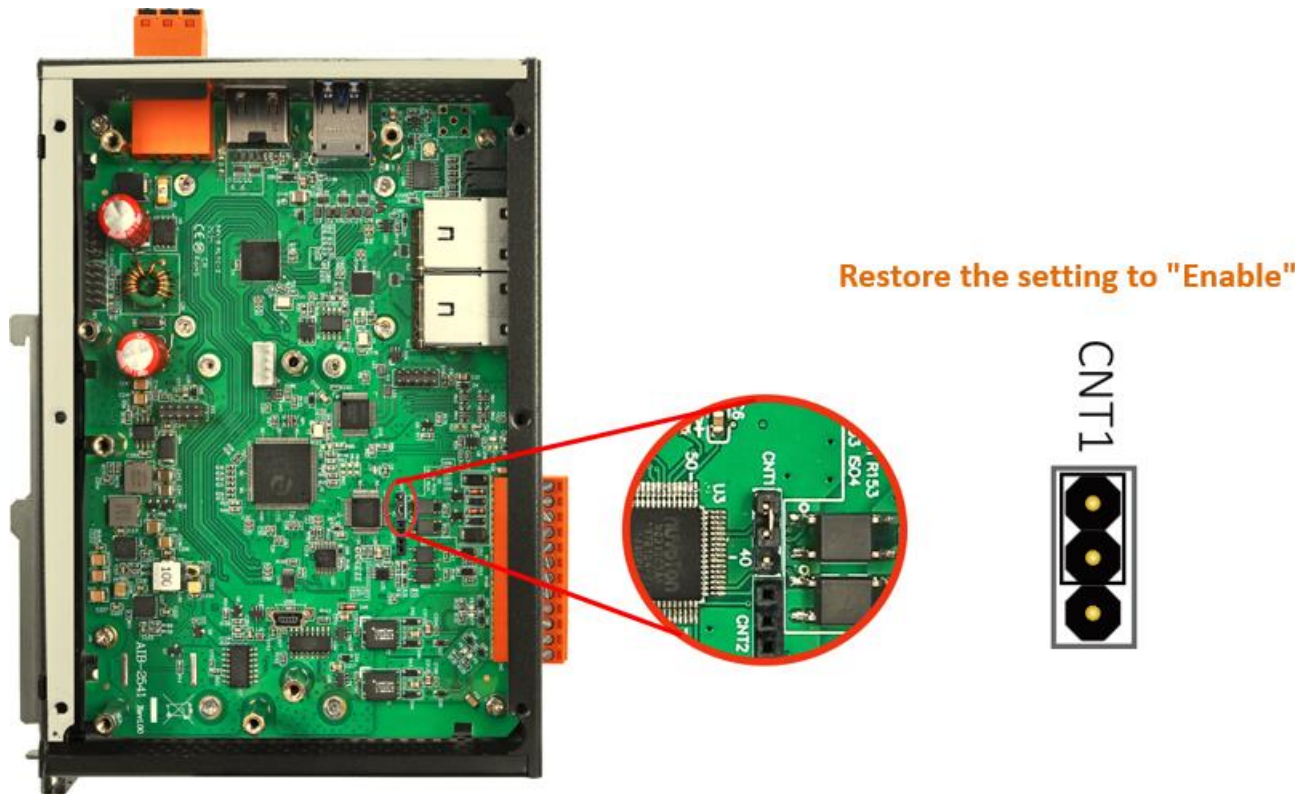
Step 12: The OS flashing process is complete.

The following figure shows the completed flashing screen.



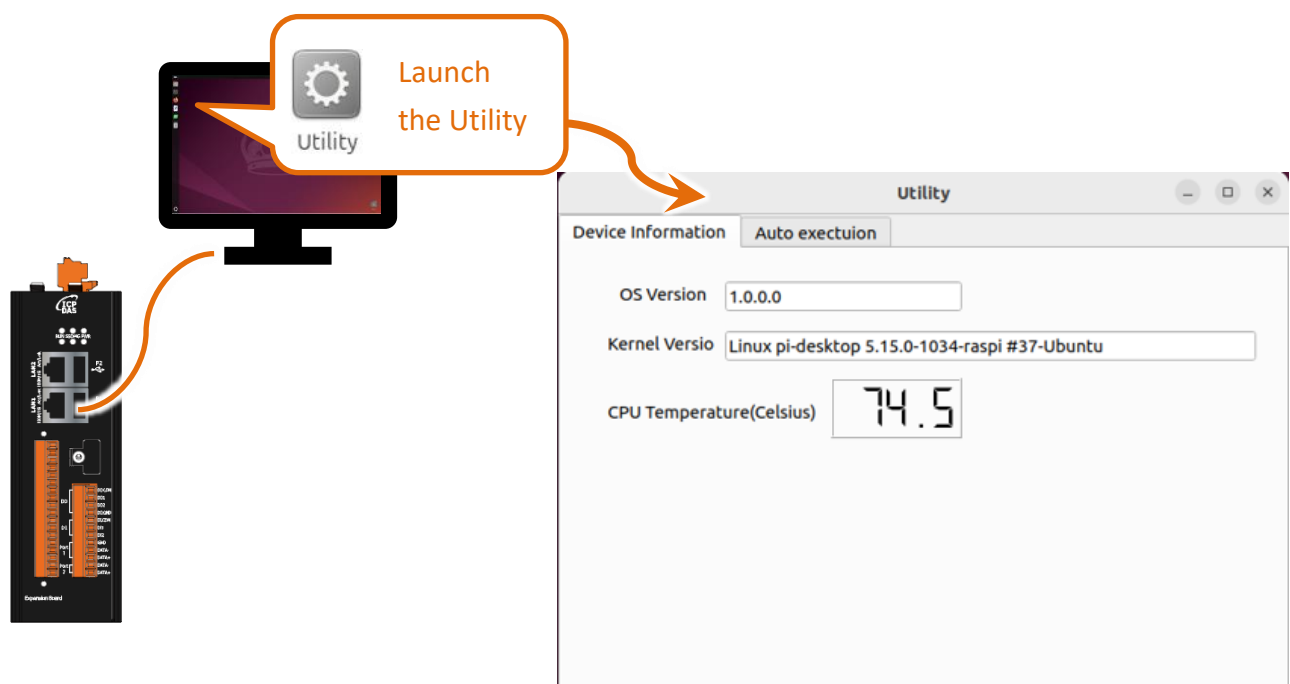
Step 13: Power off the AIB-2541M and disconnect the mini USB cable connected to the AIB-2541M

Step 14: Re-enable the AIB-2541M hardware watchdog and configure the jumper as shown below.



Step 15: Verify the OS Version

After flashing is completed, power on the AIB-2541M again. Please refer to Section 3.2.1 to launch the Utility application, then select the Device Information page to verify the OS version.



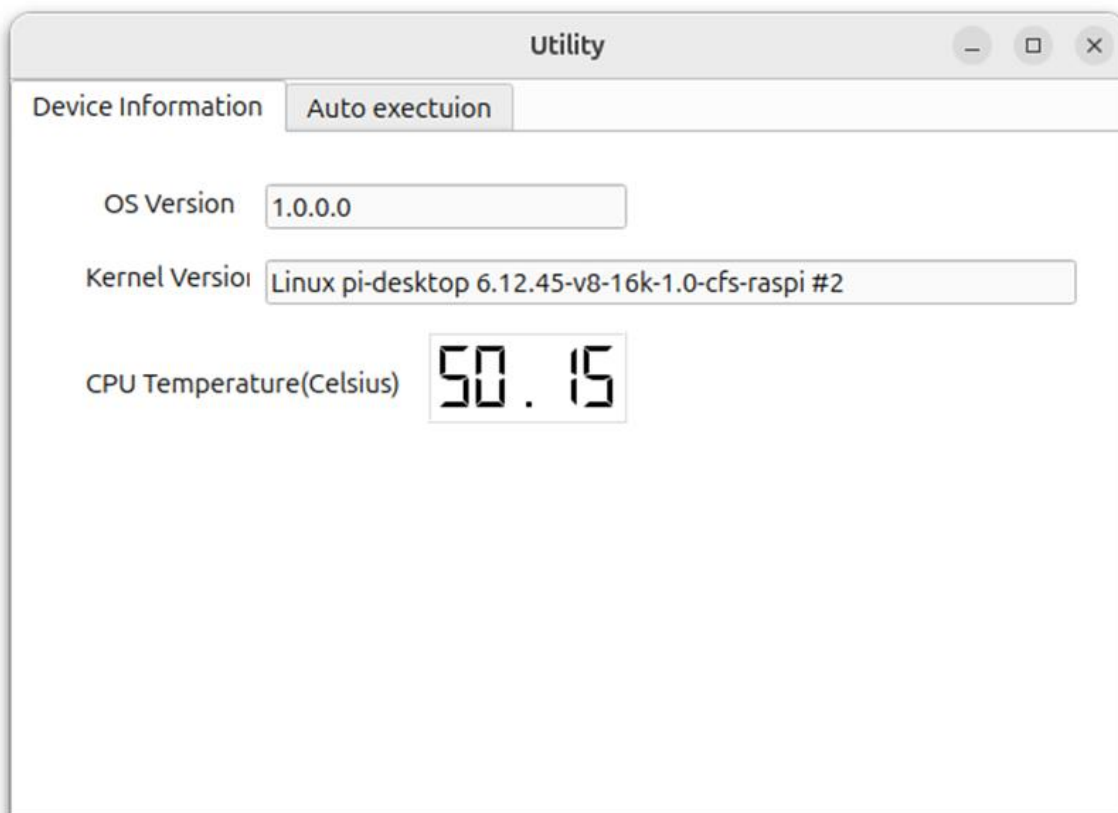
Appendix A. Practical Usage Notes

This chapter provides several useful tips for operating and maintaining the AIB-2541M series modules.

A.1. How to Check the Kernel Version

1. Using the Utility

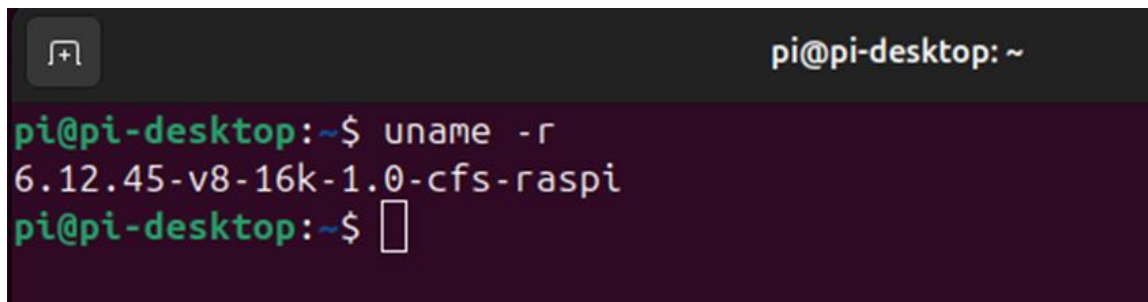
Refer to Section 3.2.1 Utility – View System Information on page 49 and launch the Utility.



The Kernel Version field displays the currently installed kernel version.

2. Using the uname -r Command

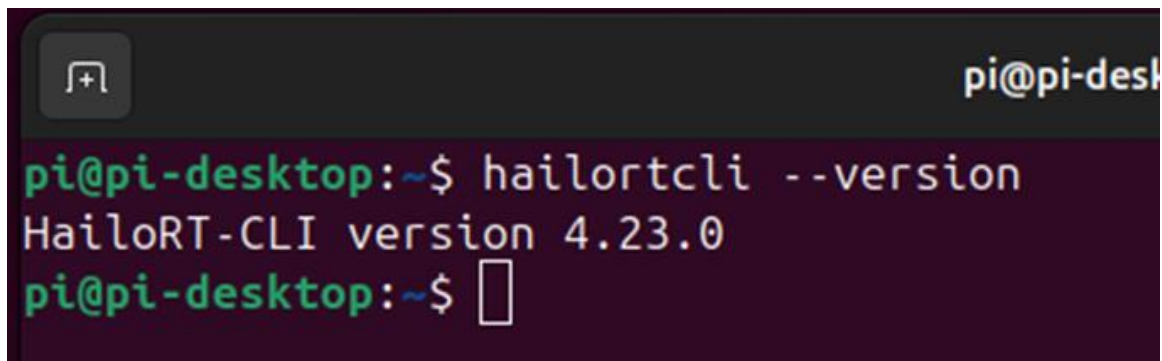
Enter the `uname -r` command in Terminal to display the current kernel version.

A terminal window with a dark background. The title bar shows a window icon and the text "pi@pi-desktop: ~". The prompt is "pi@pi-desktop:~\$". The command "uname -r" has been entered and executed, resulting in the output "6.12.45-v8-16k-1.0-cfs-raspi". The prompt is now "pi@pi-desktop:~\$" followed by a cursor.

```
pi@pi-desktop:~$ uname -r
6.12.45-v8-16k-1.0-cfs-raspi
pi@pi-desktop:~$
```

A.2. How to Check the HailoRT Version

To check the installed HailoRT version, open Terminal and enter the `hailortcli --version` command. The version number will be displayed as shown below. In the example below, the HailoRT version is 4.23.0.

A terminal window with a dark background. The title bar shows a window icon and the text "pi@pi-desk". The prompt is "pi@pi-desktop:~\$". The command "hailortcli --version" has been entered and executed, resulting in the output "HailoRT-CLI version 4.23.0". The prompt is now "pi@pi-desktop:~\$" followed by a cursor.

```
pi@pi-desktop:~$ hailortcli --version
HailoRT-CLI version 4.23.0
pi@pi-desktop:~$
```

A.3. How to Configure Automatic Network Time Synchronization

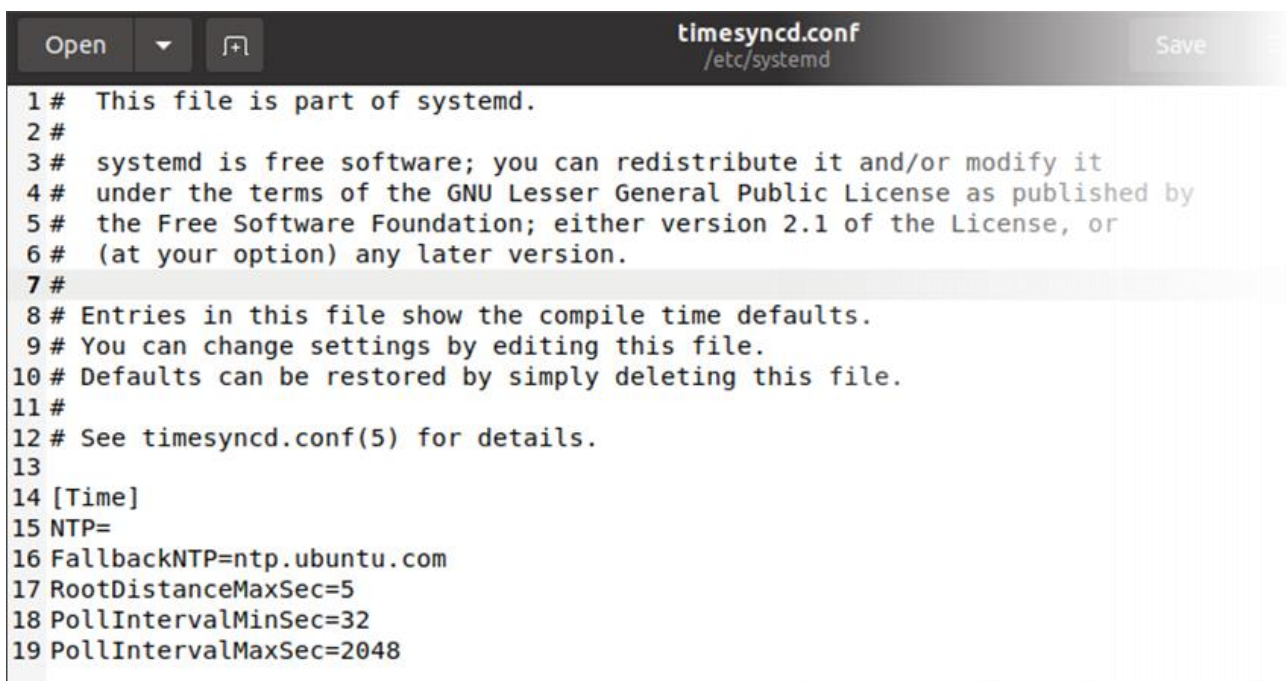
You can synchronize the system time of the AIB-2541M through a network time server.

When synchronization is enabled, the AIB-2541M system time will automatically synchronize with the network time server.

Step 1: Launch Terminal and enter the following command

sudo gedit /etc/systemd/timesyncd.conf

The configuration screen is shown in the figure below.

A screenshot of a text editor window titled 'timesyncd.conf /etc/systemd'. The window shows the contents of the file, which includes comments and configuration parameters. The parameters are: NTP=, FallbackNTP=ntp.ubuntu.com, RootDistanceMaxSec=5, PollIntervalMinSec=32, and PollIntervalMaxSec=2048.

```
1 # This file is part of systemd.
2 #
3 # systemd is free software; you can redistribute it and/or modify it
4 # under the terms of the GNU Lesser General Public License as published by
5 # the Free Software Foundation; either version 2.1 of the License, or
6 # (at your option) any later version.
7 #
8 # Entries in this file show the compile time defaults.
9 # You can change settings by editing this file.
10 # Defaults can be restored by simply deleting this file.
11 #
12 # See timesyncd.conf(5) for details.
13
14 [Time]
15 NTP=
16 FallbackNTP=ntp.ubuntu.com
17 RootDistanceMaxSec=5
18 PollIntervalMinSec=32
19 PollIntervalMaxSec=2048
```

The parameter descriptions are listed in the following table.

Field	Description
NTP	Primary time synchronization server
FallbackNTP	Backup time synchronization server
RootDistanceMaxSec	Maximum allowable time difference with the server (seconds)
PollIntervalMinSec	Minimum synchronization interval
PollIntervalMaxSec	Maximum synchronization interval

Step 2: After configuration is completed, click SAVE to save the settings.

Step 3: In Terminal, enter the following command to restart the time synchronization service and load the new settings

```
sudo systemctl restart systemd-timesyncd
```

A.4. How to Replace the RTC Battery

The RTC chip uses a lithium battery that can provide continuous power for up to 10 years.

Step 1: Power off the AIB-2541M

Step 2: Use a screwdriver to remove the screws on the left-side cover

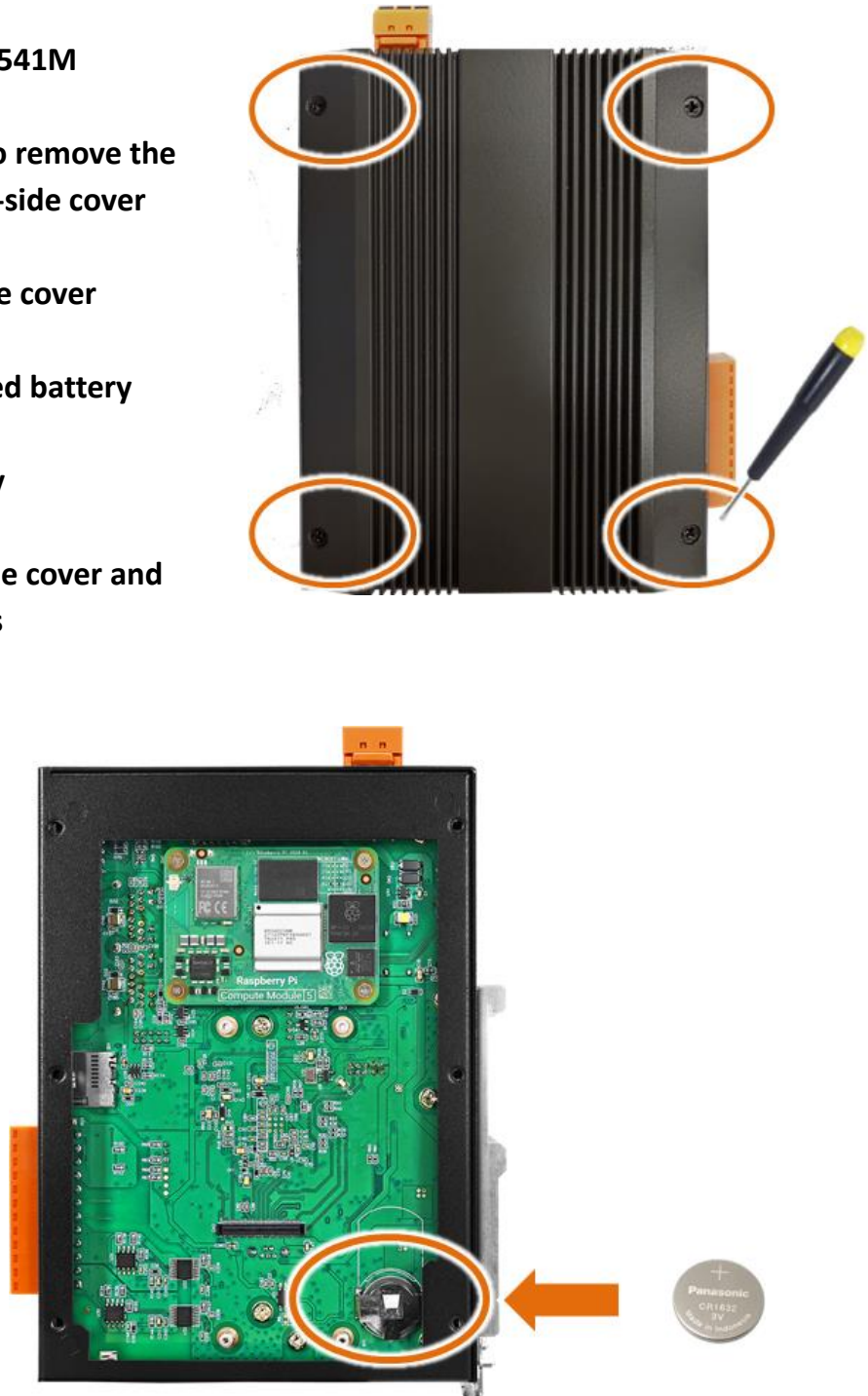
Step 3: Remove the left-side cover

Step 4: Remove the depleted battery

Step 5: Insert a new battery

Step 6: Reinstall the left-side cover and tighten the screws

Step 7: Set the RTC time



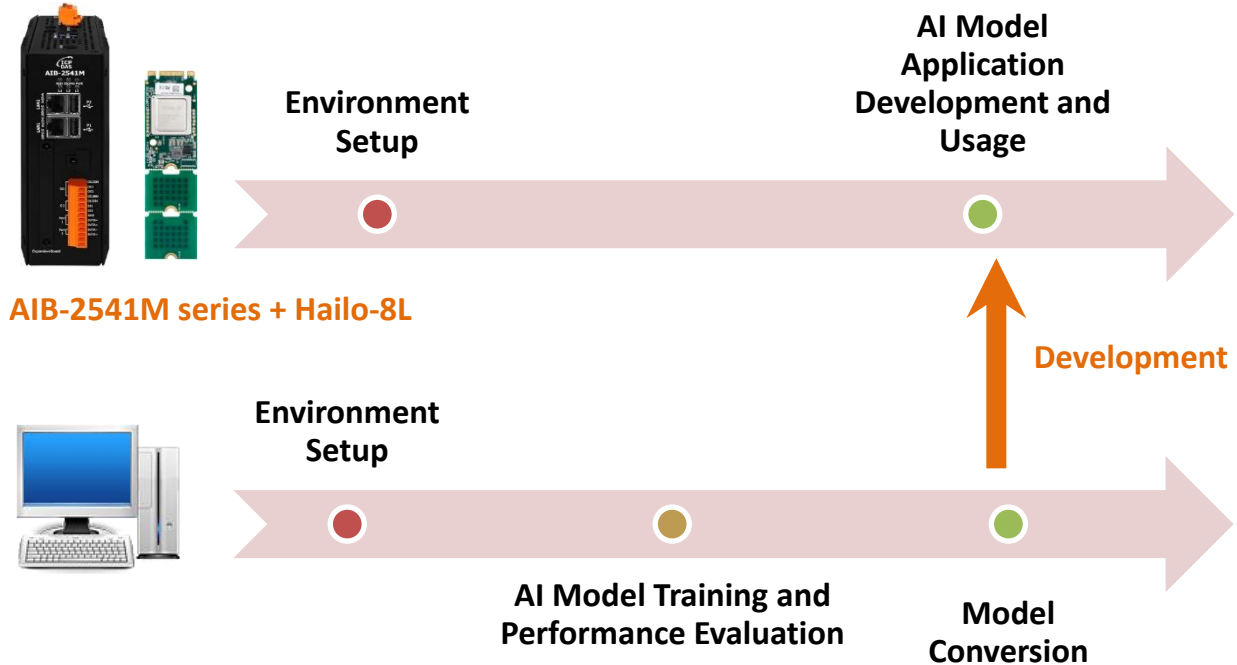
Ordering Information

Battery type: BR1632 (ICP DAS part number: 2LB010)

For additional information, please contact your local sales office or distributor.

Appendix B. Hailo-8L Development and Application

The development workflow for Hailo-8L AI learning models is illustrated in the figure below.



Environment Setup

- Runtime environment setup for the AIB-2541M series: Install HailoRT and the Hailo PCIe driver. (Refer to Section B.1.1)
- PC environment setup for AI model training: Install CUDA and the required deep learning frameworks. (Refer to Section B.1.2)



HailoRT and the Hailo PCIe Driver

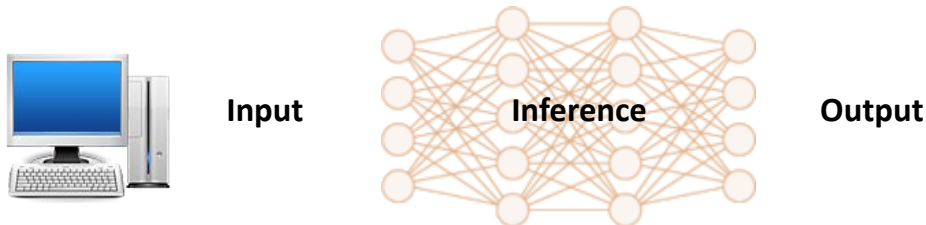


CUDA Toolkit and Deep Learning Frameworks (PyTorch, TensorFlow, etc.)

AI Model Training and Performance Evaluation

Modern AI models require users to provide large amounts of training data so that the AI system can extract target object features for recognition or prediction. After training is completed, the AI model must also be tested and evaluated to verify inference performance.

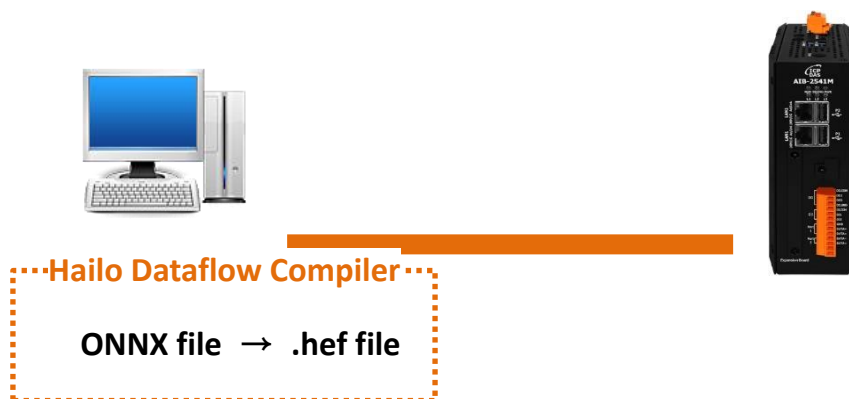
For more information regarding AI model training and performance evaluation, please refer to Section B.2.



Model Conversion

The trained AI model must be converted into a file format supported by the Hailo-8L runtime using Hailo conversion tools.

For installation procedures and model conversion workflows, please refer to Section B.3



AI Model Application Development and Deployment

After the AI model conversion process is completed, the converted AI model is deployed on the AIB-2541M. An application program is required to load the AI model and input data into the Hailo-8L accelerator, while also receiving the inference results generated by the AI model running on Hailo-8L.

For more information regarding AI model application development and deployment, please refer to Section B.4.

The following chapters describe the installation and operation procedures step by step. In this example, the PC operating system used is Ubuntu 20.04, and the GPU used is an NVIDIA RTX 2060.

B.1. Environment Setup

B.1.1. Runtime Environment Setup: HailoRT and Hailo PCIe Driver

Installation

To operate a Hailo AI accelerator card on the AIB-2541M, HailoRT and the Hailo PCIe driver must be installed. The following section describes the installation procedure for HailoRT and the Hailo PCIe driver. First, download HailoRT and the Hailo PCIe driver from the Hailo website listed below.
<https://hailo.ai/developer-zone/software-downloads/>

The following example demonstrates the installation of HailoRT version 4.15 and the corresponding PCIe driver.

Guidance & Notes



The versions of HailoRT and the PCIe driver must match. If the versions are inconsistent, problems may occur during HailoRT operation.

Step 1: Open Terminal and enter the following command, and then enter the password “sudo apt-get install build-essential”

Step 2: Open Terminal, switch to the HailoRT installation package directory, and enter “sudo dpkg -i hailort_4.15.0_arm64.deb”

The following screen indicates that the installation has completed successfully. (This example uses HailoRT version 4.15.)

```
pi@pi-desktop:~$ sudo dpkg -i hailort_4.15.0_arm64.deb
(Reading database ... 168833 files and directories currently installed.)
Preparing to unpack hailort_4.15.0_arm64.deb ...
Unpacking hailort (4.15.0) over (4.15.0) ...
Setting up hailort (4.15.0) ...
Do you wish to activate hailort service? (required for most pyHailoRT use cases) [y/N]:
Starting hailort.service
Created symlink /etc/systemd/system/multi-user.target.wants/hailort.service → /lib/systemd/system/hailort.service.
```

Step 3: Open Terminal, switch to the Hailo PCIe Driver installation package directory, and enter “sudo dpkg -i hailort-pcie-driver_4.15.0_all.deb”

The following screen indicates that the installation has completed successfully. (This example uses HailoRT version 4.15.)

```
pi@pi-desktop:~$ sudo dpkg -i hailort-pcie-driver_4.15.0_all.deb
Selecting previously unselected package hailort-pcie-driver.
(Reading database ... 168833 files and directories currently installed.)
Preparing to unpack hailort-pcie-driver_4.15.0_all.deb ...
Could not test for SecureBoot, assuming SecureBoot is disabled on this machine.
Unpacking hailort-pcie-driver (4.15.0) ...
Setting up hailort-pcie-driver (4.15.0) ...

WARNING: apt does not have a stable CLI interface. Use with caution in scripts.

build-essential/jammy,now 12.9ubuntu3 arm64 [installed]
Do you wish to use DKMS? [Y/n]:
Please reboot your computer for the installation to take effect.
```

Step 4: Open Terminal and enter “hailortcli scan” to verify whether the Hailo hardware is detected.

If the Hailo hardware is detected, a device ID will be displayed, indicating that HailoRT and the Hailo PCIe driver have been installed correctly.

```
pi@pi-desktop:~$ hailortcli scan
Hailo Devices:
[-] Device: 0000:04:00.0
```

B.1.2. PC Environment Setup: CUDA and Deep Learning Framework

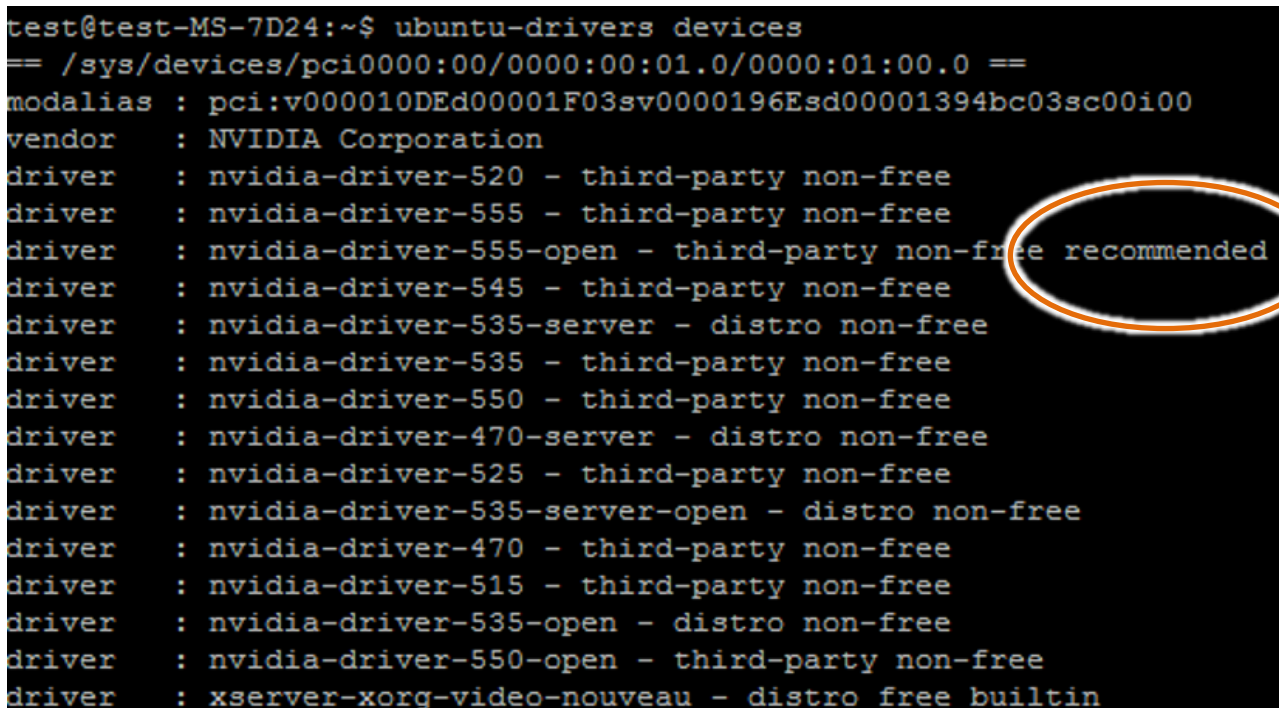
Installation

This example uses an NVIDIA GPU and therefore requires an NVIDIA driver and a compatible CUDA Toolkit. Many deep learning frameworks can use CUDA for GPU acceleration. The following section describes how to install CUDA and the required deep learning framework in an Ubuntu environment.

CUDA Installation

Step 1: In the Ubuntu environment, open Terminal and enter “ubuntu-drivers devices”

The available NVIDIA GPU drivers supported by the OS will be listed. It is recommended to install the driver marked as “recommended”. This driver version has been validated by NVIDIA and is less likely to encounter issues.



```
test@test-MS-7D24:~$ ubuntu-drivers devices
== /sys/devices/pci0000:00/0000:00:01.0/0000:01:00.0 ==
modalias : pci:v000010DEd00001F03sv0000196Esd00001394bc03sc00i00
vendor   : NVIDIA Corporation
driver   : nvidia-driver-520 - third-party non-free
driver   : nvidia-driver-555 - third-party non-free
driver   : nvidia-driver-555-open - third-party non-free recommended
driver   : nvidia-driver-545 - third-party non-free
driver   : nvidia-driver-535-server - distro non-free
driver   : nvidia-driver-535 - third-party non-free
driver   : nvidia-driver-550 - third-party non-free
driver   : nvidia-driver-470-server - distro non-free
driver   : nvidia-driver-525 - third-party non-free
driver   : nvidia-driver-535-server-open - distro non-free
driver   : nvidia-driver-470 - third-party non-free
driver   : nvidia-driver-515 - third-party non-free
driver   : nvidia-driver-535-open - distro non-free
driver   : nvidia-driver-550-open - third-party non-free
driver   : xserver-xorg-video-nouveau - distro free builtin
```

Step 2: After installing the GPU driver, please enter "nvidia-smi" in Terminal

The CUDA Version field indicates the maximum CUDA version supported by the installed NVIDIA driver. Select a compatible CUDA Toolkit version for the framework being used.

```
test@test-MS-7D24:~$ nvidia-smi
Wed Nov 13 09:10:54 2024

+-----+
| NVIDIA-SMI 535.183.01                  Driver Version: 535.183.01    CUDA Version: 12.0     |
+-----+-----+-----+-----+-----+-----+
| GPU  Name                Persistence-M | Bus-Id        Disp.A    Volatile Uncorr. ECC |
| Fan  Temp   Perf          Pwr:Usage/Cap |      Memory-Usage | GPU-Util  Compute M. |
|=====+=====+=====+=====+=====+=====+
| 0  NVIDIA GeForce RTX 2060        Off | 00000000:01:00.0  On          N/A           |
| 45%   32C    P8              16W / 184W | 182MiB / 12288MiB |      0%      Default |
+-----+-----+-----+-----+-----+-----+
| Processes:                                                       GPU Memory |
|  GPU   GI    CI          PID    Type   Process name                  Usage      |
|=====+=====+=====+=====+=====+=====+
| 0      N/A  N/A         1176     G   /usr/lib/xorg/Xorg              23MiB     |
| 0      N/A  N/A         1685     G   /usr/lib/xorg/Xorg              38MiB     |
| 0      N/A  N/A         1814     G   /usr/bin/gnome-shell            111MiB    |
+-----+-----+-----+-----+-----+-----+

```

Step 3: Please download the corresponding CUDA Toolkit installation package from the following NVIDIA website

<https://developer.nvidia.com/cuda-toolkit-archive>

The website is shown in the figure below. In this example, CUDA version 12.0 is used.

CUDA Toolkit Archive

Previous releases of the CUDA Toolkit, GPU Computing SDK, documentation and developer drivers can be found using the links below. Please select the release you want from the list below, and be sure to check www.nvidia.com/drivers for more recent production drivers appropriate for your hardware configuration.

[Download Latest CUDA Toolkit](#)

[Learn More about CUDA Toolkit](#)

Latest Release

[CUDA Toolkit 12.6.2 \(October 2024\)](#), [Versioned Online Documentation](#)

Archived Releases

- [CUDA Toolkit 12.6.1 \(August 2024\)](#), [Versioned Online Documentation](#)
- [CUDA Toolkit 12.6.0 \(August 2024\)](#), [Versioned Online Documentation](#)
- [CUDA Toolkit 12.5.1 \(July 2024\)](#), [Versioned Online Documentation](#)
- [CUDA Toolkit 12.5.0 \(May 2024\)](#), [Versioned Online Documentation](#)
- [CUDA Toolkit 12.4.1 \(April 2024\)](#), [Versioned Online Documentation](#)
- [CUDA Toolkit 12.4.0 \(March 2024\)](#), [Versioned Online Documentation](#)
- [CUDA Toolkit 12.3.2 \(January 2024\)](#), [Versioned Online Documentation](#)
- [CUDA Toolkit 12.3.1 \(November 2023\)](#), [Versioned Online Documentation](#)
- [CUDA Toolkit 12.3.0 \(October 2023\)](#), [Versioned Online Documentation](#)
- [CUDA Toolkit 12.2.2 \(August 2023\)](#), [Versioned Online Documentation](#)
- [CUDA Toolkit 12.2.1 \(July 2023\)](#), [Versioned Online Documentation](#)
- [CUDA Toolkit 12.2.0 \(June 2023\)](#), [Versioned Online Documentation](#)
- [CUDA Toolkit 12.1.1 \(April 2023\)](#), [Versioned Online Documentation](#)
- [CUDA Toolkit 12.1.0 \(February 2023\)](#), [Versioned Online Documentation](#)
- [CUDA Toolkit 12.0.1 \(January 2023\)](#), [Versioned Online Documentation](#)
- [CUDA Toolkit 12.0.0 \(December 2022\)](#), [Versioned Online Documentation](#)
- [CUDA Toolkit 11.8.0 \(October 2022\)](#), [Versioned Online Documentation](#)

Step 4: Please select the installation options according to the software environment used

by the PC

Select the OS, hardware architecture, and other required options in sequence. In this example, the hardware platform is an x86 PC, the OS is Ubuntu 20.04, and the installation method uses .deb packages. Therefore, select the options shown in the figure below.

Operating System	Linux	Windows							
Architecture	x86_64	ppc64le	arm64-sbsa	aarch64-jetson					
Distribution	CentOS	Debian	Fedora	KylinOS	OpenSUSE	RHEL	Rocky	SLES	Ubuntu
Version	20.04	22.04							
Installer Type	deb (local)	deb (network)	runfile (local)						

Step 5: Follow the installation instructions provided on the website and execute the commands step by step in Terminal to complete the installation

Download Installer for Linux Ubuntu 20.04 x86_64

The base installer is available for download below.

> Base Installer

Installation Instructions:

```
$ wget https://developer.download.nvidia.com/compute/cuda/repos/ubuntu2004/x86_64/cuda-ubuntu2004.pin
$ sudo mv cuda-ubuntu2004.pin /etc/apt/preferences.d/cuda-repository-pin-600
$ wget https://developer.download.nvidia.com/compute/cuda/12.2.0/local_installers/cuda-repo-ubuntu2004-12-2-local_12.2.0-535.54.03-1_amd64.deb
$ sudo dpkg -i cuda-repo-ubuntu2004-12-2-local_12.2.0-535.54.03-1_amd64.deb
$ sudo cp /var/cuda-repo-ubuntu2004-12-2-local/cuda-*-keyring.gpg /usr/share/keyrings/
$ sudo apt-get update
$ sudo apt-get -y install cuda
```

Step 6: After CUDA installation is completed, please enter "nvcc --version" in Terminal to verify whether the installed CUDA version is correct

In this example, CUDA version 12.0 is installed, and the output is shown below.

```
test@test-MS-7D24:~$ nvcc --version
nvcc: NVIDIA (R) Cuda compiler driver
Copyright (c) 2005-2022 NVIDIA Corporation
Built on Mon_Oct_24_19:12:58_PDT_2022
Cuda compilation tools, release 12.0, V12.0.76
Build cuda_12.0.r12.0/compiler.31968024_0
```

Deep Learning Framework Installation

There are many deep learning frameworks available, such as PyTorch and TensorFlow. The required framework depends on the model implementation and training workflow. Therefore, first determine which framework is required by the selected model.

In this example, the AI model used is YOLOv5, and the corresponding deep learning framework is PyTorch. Therefore, this section focuses on PyTorch installation.

Step 1: Please visit the PyTorch website <https://pytorch.org/> to begin the installation process

Select the appropriate options according to your PC environment, as shown in the figure below. In this example, installation is performed on Ubuntu using pip. Select the PyTorch build that is compatible with your NVIDIA driver and supported by the selected PyTorch release. For example, the CUDA version installed in the previous section is 12.0, so version 11.8 is selected.

Because installation is performed using pip, enter the installation command shown under “Run this Command” in Terminal.

PyTorch Build	Stable (2.5.1)		Preview (Nightly)	
Your OS	Linux	Mac	Windows	
Package	Conda	Pip	LibTorch	Source
Language	Python		C++ / Java	
Compute Platform	CUDA 11.8	CUDA 12.1	CUDA 12.4	ROCm 6.2
Run this Command:	pip3 install torch torchvision torchaudio --index-url https://download.pytorch.org/whl/cu118			

Step 2: After installation is completed, the installed packages can be verified using “pip list”. The result shown below indicates that PyTorch has been successfully installed.

```
test@test-MS-7D24:~$ pip list | grep torch
torch                2.0.1+cu118
torchaudio           2.0.2+cu118
torchvision          0.15.2+cu118
```

B.2. AI Model Training and Performance Evaluation

There are many types of AI models available. Users should select an AI model according to their application requirements. In this example, YOLOv5 is used. YOLOv5 is an object detection model implementation developed by Ultralytics. The following sections describe the AI model training workflow and performance evaluation process for YOLOv5.

B.2.1. Model Training

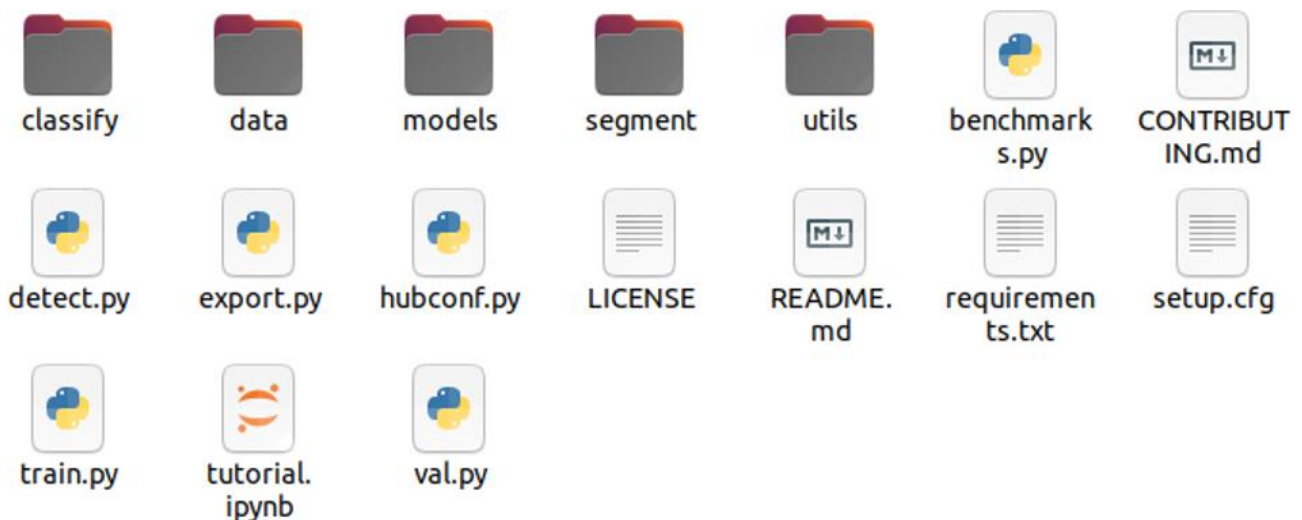
Step 1: Download YOLOv5 from the Ultralytics GitHub Repository

The latest version of YOLOv5 can be downloaded from the following link:

<https://github.com/ultralytics/yolov5>

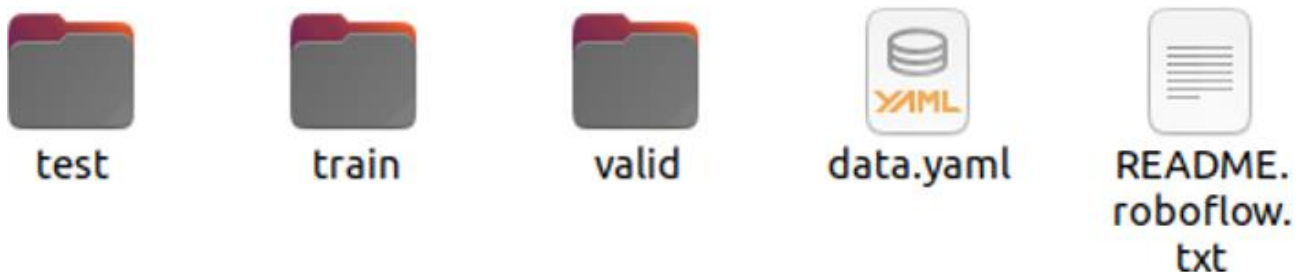
This example uses YOLOv5 v7.0.

Step 2: After extracting the downloaded YOLOv5 package, the directory structure is shown below.



Step 3: Add the Training Dataset

Currently, many websites provide free training image datasets, such as COCO and Roboflow. In this example, the training dataset is downloaded from Roboflow. The training target in this example is vehicle and license plate detection. The figure below shows the dataset used for this training example.



Step 4: Open Terminal, switch to the YOLOv5 directory, and enter the following training command:

```
python3 train.py --img-size 640 --batch 16 --epochs 300 --data ./car_License/data.yaml  
--weights yolov5s.pt
```

The main parameters are described below:

--data: Path to the dataset YAML file

--epochs: Number of training epochs. If the dataset contains 100 images, processing all 100 images once is considered one epoch.

--batch: Number of data samples input to the GPU for each training iteration. Larger values require more GPU memory.

For additional parameters, enter “python3 train.py --help” for detailed descriptions. After entering the above command in Terminal, training will begin as shown in the figure below.

```
AutoAnchor: 3.98 anchors/target, 1.000 Best Possible Recall (BPR). Current anchors are a good fit to dataset ✓  
Plotting labels to runs/train/exp/labels.jpg...  
Image sizes 640 train, 640 val  
Using 8 dataloader workers  
Logging results to runs/train/exp  
Starting training for 300 epochs...  
  
Epoch   GPU_mem  box_loss  obj_loss  cls_loss  Instances  Size  
0/299    3.46G    0.1238    0.03906   0.02909    90         640: 50% | 8/16 00:02
```

Step 5: After training is completed, the following information will be displayed

The trained AI models will be stored in “/runs/train/exp<number>/weights”; the number indicates how many training sessions have been executed previously. Since this example is the first training run, the directory will be exp.

Two model files will be generated in the weights folder:

best.pt: The best-performing model across all epochs

last.pt: The model generated during the final epoch

```
Validating runs/train/exp/weights/best.pt...
Fusing layers...
Model summary: 157 layers, 7015519 parameters, 0 gradients, 15.8 GFLOPs
   Class      Images  Instances    P      R    mAP50  mAP50-95: 100%| 3/3 00:00
   all         70       218    0.839  0.834  0.873    0.676
 license-plate'  70        84    0.888  0.893  0.941    0.705
   vehicle     70       134    0.79   0.776  0.805    0.646
Results saved to runs/train/exp
```

Guidance & Notes



The dataset used in this example contains approximately 300 images. The training GPU is an RTX 2060, the batch size is set to 16, and the training time is approximately 2 to 3 hours.

B.2.2. Model Performance Evaluation

After training is completed, the following performance information is displayed.

```
Validating runs/train/exp/weights/best.pt...
Fusing layers...
Model summary: 157 layers, 7015519 parameters, 0 gradients, 15.8 GFLOPs
   Class      Images  Instances    P      R    mAP50  mAP50-95: 100%| 3/3 00:00
   all         70       218    0.839  0.834  0.873    0.676
 license-plate'  70        84    0.888  0.893  0.941    0.705
   vehicle     70       134    0.79   0.776  0.805    0.646
Results saved to runs/train/exp
```

The mAP@0.5:0.95 metric evaluates object detection performance across multiple intersection over union (IoU) thresholds. Higher values indicate stronger object recognition capability.

For the meanings of each parameter, please refer to the Ultralytics documentation:

<https://docs.ultralytics.com/guides/yolo-performance-metrics/#introduction>

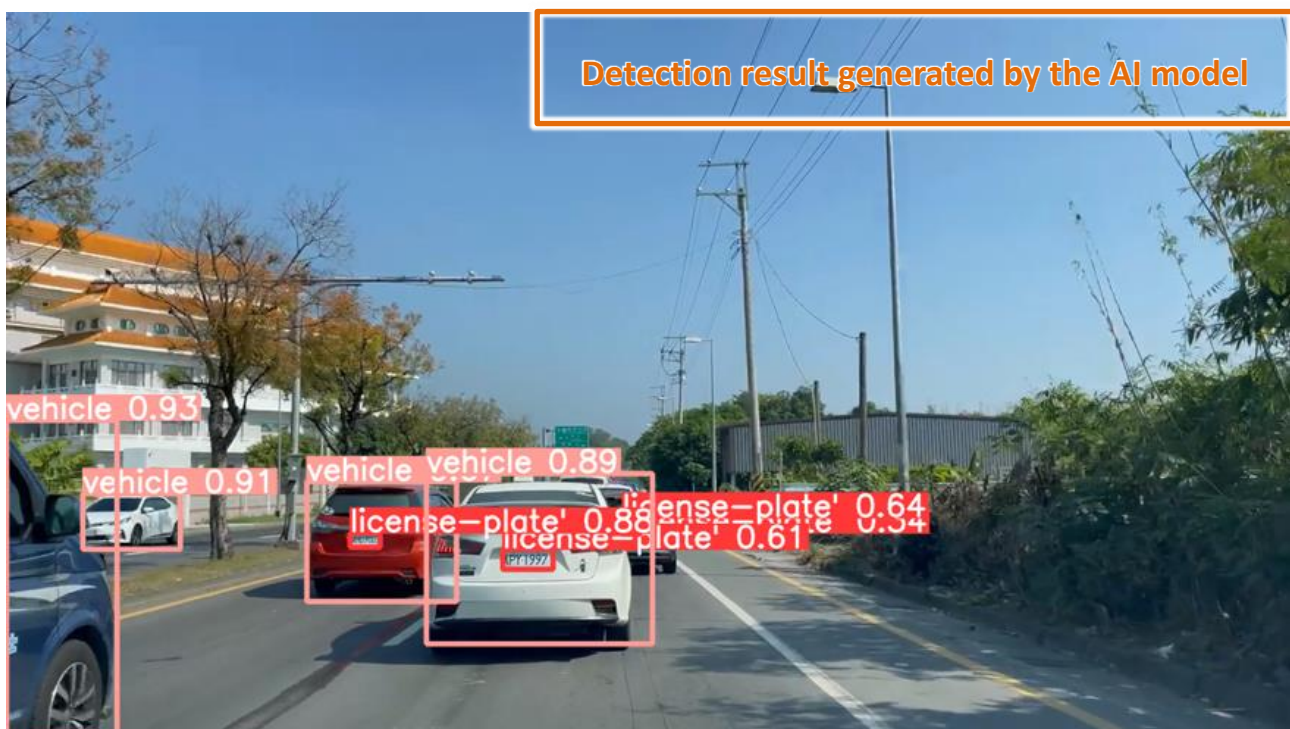
The trained AI model can be verified using detect.py to determine whether images outside the training dataset can be correctly recognized. Enter the following command:

```
python3 detect.py --source test.jpg --weights best.pt
```

The figure below shows the result of the trained AI model detecting vehicles and license plates.

--source specifies the image or video to be detected

--weights specifies the AI model used for detection



B.3. Model Conversion

B.3.1. Installing the Model Conversion Tool and Converting Models for Hailo-8L

Section B.2.1 has already explained how to train an AI model. However, the trained AI model cannot be directly used on the AIB-2541M because HailoRT only supports AI models in Hailo format. Therefore, additional AI model conversion is required.

In addition to installing the Hailo Dataflow Compiler, AI model conversion also requires additional conversion scripts. Since the conversion scripts must be modified according to the parameters used during the training process, and the converted results may require further adjustment, there is no universal conversion script available.

Support from Hailo is required for this process. You may submit your AI model conversion issues to us, and we will contact Hailo for technical assistance. This chapter only describes the installation of the Hailo Dataflow Compiler and the preparation procedures required before AI model conversion.

B.3.2. Installing the Hailo Dataflow Compiler

Step 1: Please first download the Hailo Dataflow Compiler from the following Hailo website

<https://hailo.ai/developer-zone/software-downloads/>

Guidance & Notes



HailoRT and the Hailo Dataflow Compiler have compatible version combinations, as shown below.

HailoRT	Hailo Dataflow Compiler
4.13	3.23
4.14	3.23-3.24
4.15	3.23-3.25

Future HailoRT versions released after version 4.15 follow the same compatibility rules.

1. The HailoRT version may be newer than the Hailo Dataflow Compiler version, but the Hailo Dataflow Compiler version cannot be newer than HailoRT. For example, if HailoRT is version 4.15, Hailo Dataflow Compiler versions 3.23 to 3.25 may be installed. If HailoRT is version 4.13, only Hailo Dataflow Compiler version 3.23 or earlier is supported.
2. If incompatible versions are installed, the compiled AI model may fail to run correctly on the AIB-2541M.
3. HailoRT and the Hailo PCIe Driver are pre-installed in the OS. Select the appropriate version of the Hailo Dataflow Compiler based on the HailoRT version installed in the operating system.

The examples in this section assume that HailoRT version 4.15 has already been installed as described in Section B.1.1. Therefore, Hailo Dataflow Compiler version 3.25 is used in the following example.

Step 2: Enter `pip install hailo_dataflow_compiler-3.25.0-py3-none-linux_x86_64.whl`

Because the Hailo Dataflow Compiler also requires downloading and installing multiple additional packages, the installation process may take a considerable amount of time. The following information will be displayed after installation is completed.

```
Successfully installed hailo-dataflow-compiler-3.25.0 keras-2.12.0 onnx-1.14.0 onnxsim-0.4.17 protobuf-3.20.3 pydantic-1.10.8 setuptools-68.0.0 tensorboard-2.12.3 tensorflow-2.12.0 tensorflow-estimator-2.12.0 tensorflow-probability-0.20.1 tf-lite-2.10.0
```

The installed Hailo Dataflow Compiler version can be verified using `pip list`

```
test@test-VirtualBox:~$ pip list | grep dataflow
hailo-dataflow-compiler 3.25.0
```

B.3.3. Converting Models for Hailo-8L

Before starting the conversion process, the `best.pt` model trained in the previous section “B.2.1. AI Model Training” must first be converted to ONNX format, because the Hailo Dataflow Compiler only supports ONNX and CKPT file formats. Therefore, `best.pt` must be converted.

The `export.py` script included in YOLOv5 supports exporting ONNX format. Please open Terminal, switch to the YOLOv5 directory, and enter `python3 export.py --weights best.pt --include onnx`.

The following figure shows a successful ONNX conversion result.

```
ONNX: export success 0.4s, saved as best.onnx (27.2 MB)

Export complete (1.3s)
Results saved to /home/test/yolov5-7.0
Detect:      python detect.py --weights best.onnx
Validate:    python val.py --weights best.onnx
PyTorch Hub: model = torch.hub.load('ultralytics/yolov5', 'custom', 'best.onnx')
Visualize:   https://netron.app
```

The three required steps for converting a model into the HEF format supported by Hailo are Parsing, Quantization, and Compilation. Each step requires separate conversion scripts. We are unable to provide assistance for these conversion steps, and professional support from Hailo is required. You may contact us to forward your issues to Hailo for further assistance.

B.4. AI Model Application Development and Usage

In the section “B.3 Converting the Trained AI Model into a File Format Supported by Hailo-8L,” your AI model has already been converted into the HEF format. The next step is to develop an application program to run on the AIB-2541M.

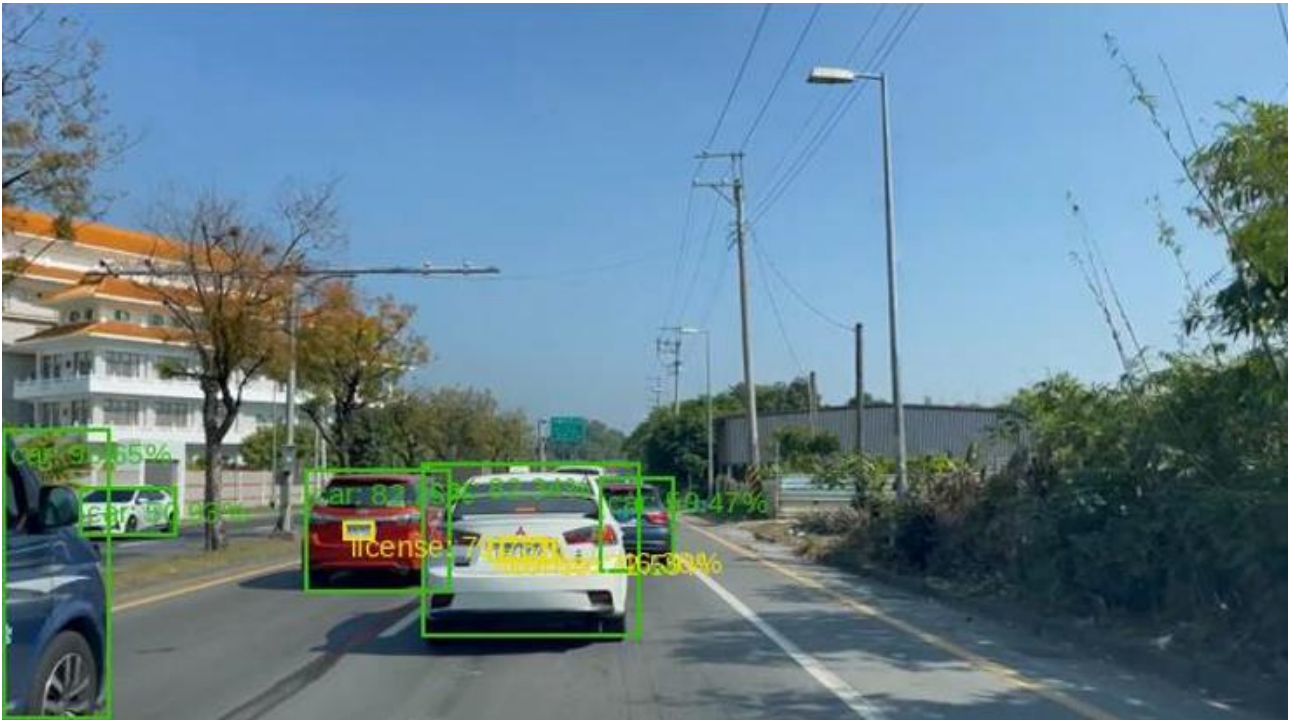
Hailo has already provided application examples on GitHub. Please download them from the following GitHub repository:

<https://github.com/hailo-ai/Hailo-Application-Code-Examples>

Hailo provides example applications in Python, C++, C#, and other languages for users to reference. Taking Python as an example, Hailo provides the following demo applications:

APP	Description
depth_estimation	Depth estimation with StereoNet
detection_with_tracker	Object detection with tracking using ByteTracker and Supervision
hailo_onnxruntime	Inference with a Hailo device and postprocessing with ONNX Runtime
instance_segmentation	Instance segmentation with yolov5_seg/yolov8_seg
lane_detection	Lane detection with yolo, ssd or centernet
object_detection	Object detection with yolo, ssd or centernet
pose_estimation	Pose estimation with yolov8
streaming	Object detection on a streaming input from a camera using OpenCV

The vehicle and license plate recognition AI model trained in the previous section can be applied to object_detection. The figure below shows an example running on the AIB-2541M after modifying the demo application.



Appendix C. XV-Board and XB-Board Series I/O Expansion Cards

The XV-Board and XB-Board series I/O expansion cards are supported by the AIB-2541M. Each controller can be expanded using one XV-Board or XB-Board series I/O expansion card.

The following tables describe the specifications of the XV-Board expansion modules.

DIO Expansion Modules

Module	DI			DO	
	Channels	Type	Sink/Source	Channels	Sink/Source
XV107	8	Wet	Source	8	Sink
XV107A			Sink		Source
XV110	16	Dry/Wet	Sink/Source	-	-
XV111	-	-	-	16	Sink
XV111A					Source

Relay Output Expansion Modules

Module	DI			Relay Output	
	Channels	Type	Sink/Source	Channels	Type
XV116	5	Wet	Sink/Source	2	Signal Relay
				4	Power Relay

Hybrid I/O Expansion Modules

Module	AI	AO	DI			DO	
	Channels		Type	Sink/Source	Channels	Sink/Source	
XV308	8	-	DI+DO=8	Dry/Wet	Source	DI+DO=8	Sink
XV310	4	5	4		Sink		Source

For more information regarding XV-Board expansion modules, please refer to:

http://www.icpdas.com/root/product/solutions/hmi_touch_monitor/touchpad/xv-board_selection.html

The following table describes the specifications of the XB-Board expansion modules.

Hybrid I/O Expansion Modules

Module	AI	AO	DI			DO	
	Channels			Type	Sink/Source	Channels	Sink/Source
XB310	4	4	4	Dry/Contact	Source	4	Sink

For more information regarding XB-Board expansion modules, please refer to:

[https://www.icpdas.com/en/product/guide+Remote I O Module and Unit+PAC I O Modules+XB-board](https://www.icpdas.com/en/product/guide+Remote+I+O+Module+and+Unit+PAC+I+O+Modules+XB-board)

Appendix D. 4G Communication

The AIB-2541M supports 4G connectivity. Each AIB-2541M can be equipped with an optional mobile communication expansion card and a 4G module to enable 4G communication.

The following link provides information about the 4G modules available from ICP DAS and their supported frequency bands:

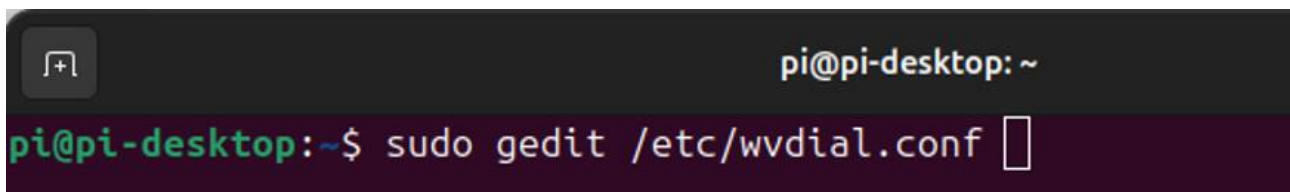
<https://www.icpdas.com/tw/product/guide+Wireless+Communication+3G+4G+5G+Products+Module>

If there are telecommunications certification requirements, users may also purchase a 4G module that complies with the certification requirements of their local region.

Dial-up Configuration

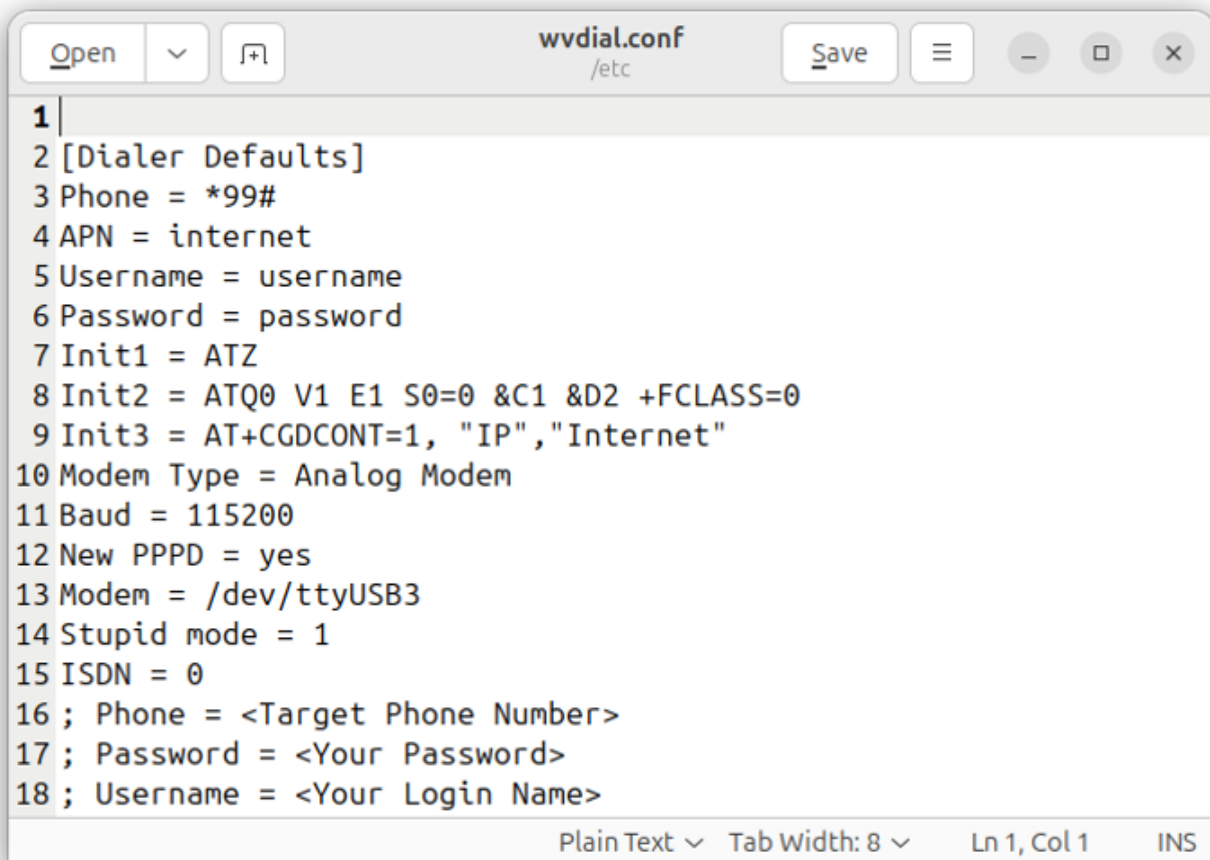
Ubuntu includes the built-in 4G dial-up software wvdial. The following steps describe the configuration and startup procedure for wvdial. The 4G module used in this example is the EC25-E, and the SIM card provider is Chunghwa Telecom.

Step 1: Open Terminal and enter “sudo gedit /etc/wvdial.conf”, then press Enter

A screenshot of a terminal window on a Raspberry Pi. The terminal title bar shows a window icon and the text 'pi@pi-desktop: ~'. The prompt is 'pi@pi-desktop:~\$'. The command 'sudo gedit /etc/wvdial.conf' has been entered, and a cursor is visible at the end of the line.

```
pi@pi-desktop: ~  
pi@pi-desktop:~$ sudo gedit /etc/wvdial.conf
```

Step 2: Modify the dial-up parameters in the gedit window as shown below

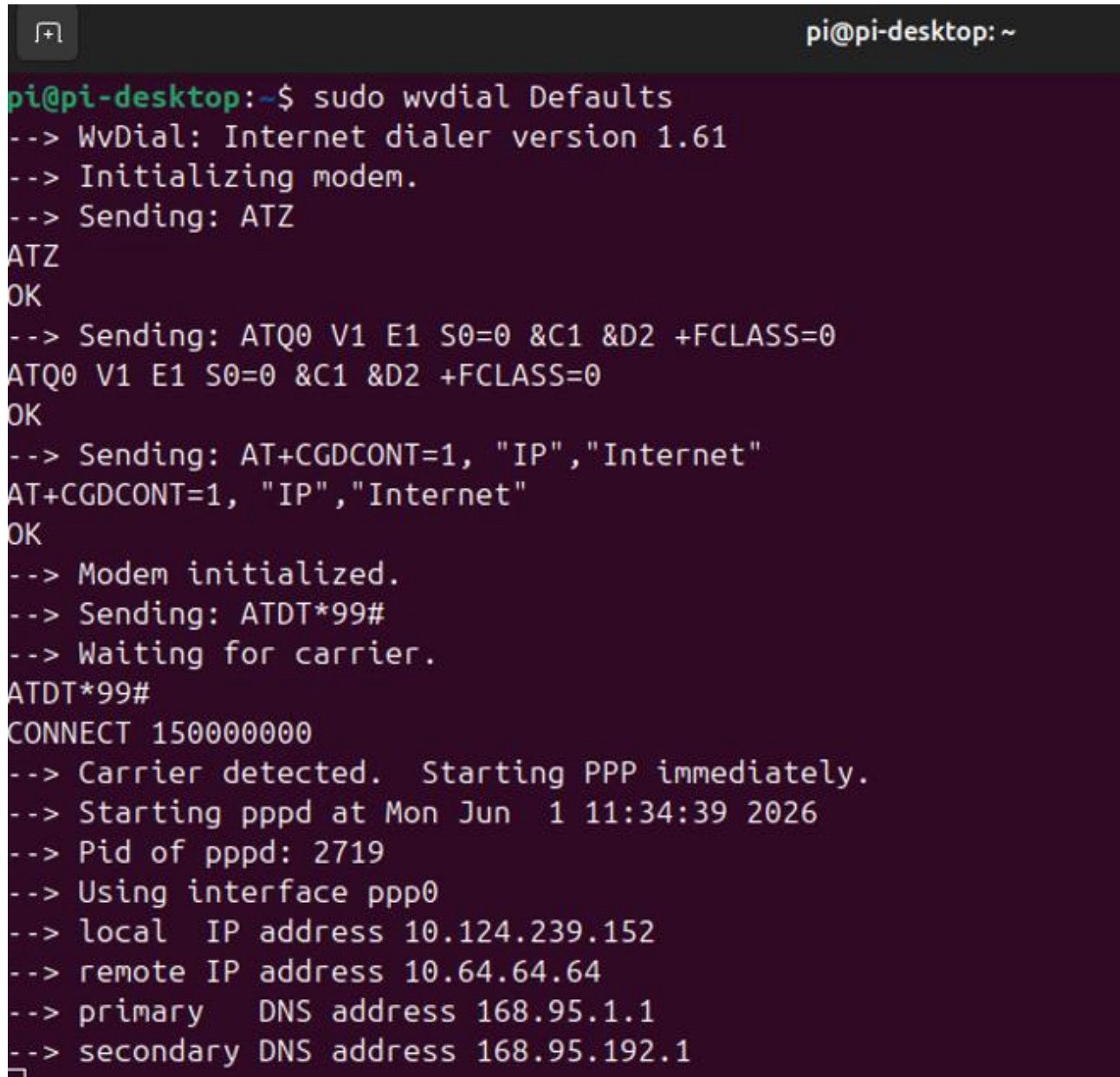


```
1 |
2 [Dialer Defaults]
3 Phone = *99#
4 APN = internet
5 Username = username
6 Password = password
7 Init1 = ATZ
8 Init2 = ATQ0 V1 E1 S0=0 &C1 &D2 +FCLASS=0
9 Init3 = AT+CGDCONT=1, "IP", "Internet"
10 Modem Type = Analog Modem
11 Baud = 115200
12 New PPPD = yes
13 Modem = /dev/ttyUSB3
14 Stupid mode = 1
15 ISDN = 0
16 ; Phone = <Target Phone Number>
17 ; Password = <Your Password>
18 ; Username = <Your Login Name>
```

1. [Dialer Defaults] indicates that the dial-up profile name used in this example is "Defaults". This profile name will be used when initiating the dial-up connection.
2. The Phone parameter is typically set to *99#. However, some mobile network operators may use a different dial-up number. If the dial-up connection fails, please contact your SIM card service provider for the correct setting.
3. The Init3 parameter may vary depending on the mobile network operator. In most regions, the parameter is the same as the one used in this example. If the dial-up connection fails, please contact your SIM card service provider for the correct setting.
4. The Modem parameter specifies the modem communication port. Please refer to the documentation of the 4G module being used. In this example, the EC25-E modem is used, and its modem port is the fourth USB port, displayed in Ubuntu as /dev/ttyUSB3.
5. After modifying the parameters, click the Save button to save the configuration, then close gedit.

Step 3: Open Terminal and enter “sudo wvdial Defaults”, then press Enter to start the dial-up connection.

A successful dial-up connection is shown below. The assigned IP parameters will be displayed at the end of the output.



```
pi@pi-desktop: ~  
pi@pi-desktop:~$ sudo wvdial Defaults  
--> WvDial: Internet dialer version 1.61  
--> Initializing modem.  
--> Sending: ATZ  
ATZ  
OK  
--> Sending: ATQ0 V1 E1 S0=0 &C1 &D2 +FCLASS=0  
ATQ0 V1 E1 S0=0 &C1 &D2 +FCLASS=0  
OK  
--> Sending: AT+CGDCONT=1, "IP","Internet"  
AT+CGDCONT=1, "IP","Internet"  
OK  
--> Modem initialized.  
--> Sending: ATDT*99#  
--> Waiting for carrier.  
ATDT*99#  
CONNECT 1500000000  
--> Carrier detected. Starting PPP immediately.  
--> Starting pppd at Mon Jun  1 11:34:39 2026  
--> Pid of pppd: 2719  
--> Using interface ppp0  
--> local  IP address 10.124.239.152  
--> remote IP address 10.64.64.64  
--> primary  DNS address 168.95.1.1  
--> secondary DNS address 168.95.192.1
```

Appendix E. Revision

This chapter provides the revision history of this User Manual.

The following table lists the release date and description for each revision of this document.

Version	Release Date	Description
1.0.0	July 2026	Initial release