

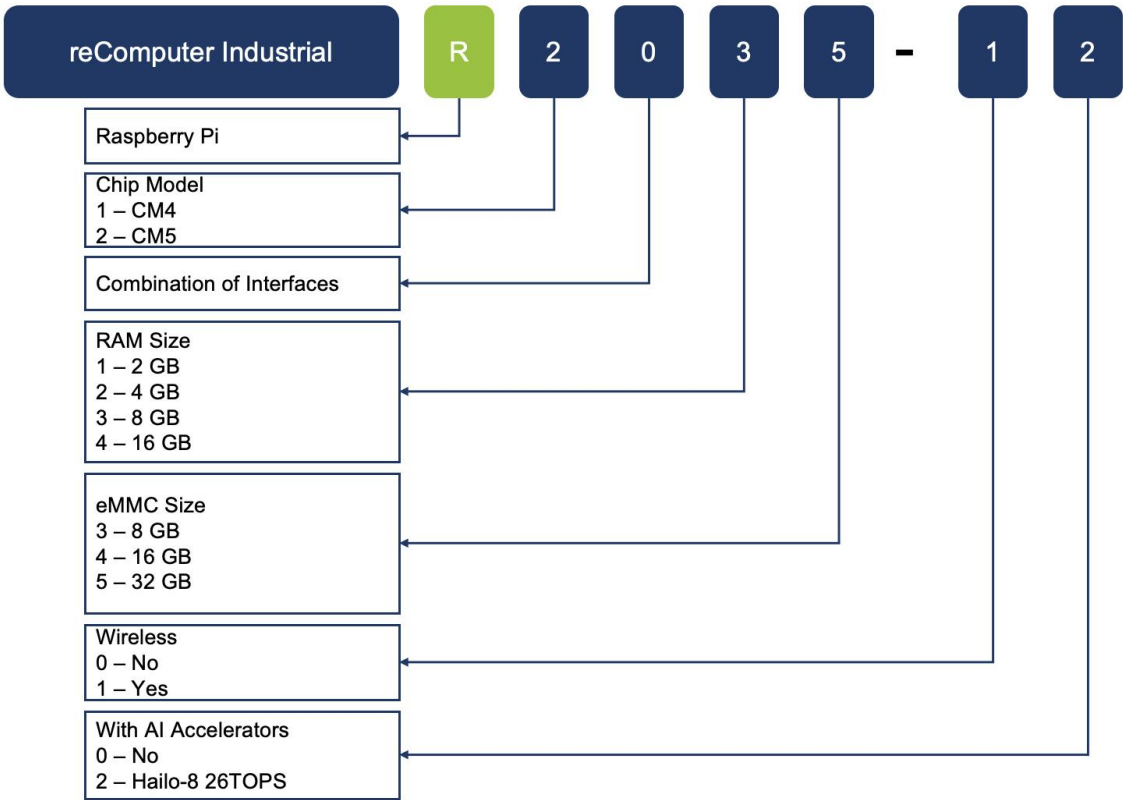


reComputer Industrial R2000

User Manual

Edge IoT Controller

Naming Conventions



For example, the naming of 8GB RAM and 32GB eMMC CM5 module with Hailo-8 26TOPS AI Accelerator function tailored for edge AI computer is reComputer Industrial R2035-12.

Revision History

Version	Date	Description
1.0	2025-07-14	

Contents

Naming Conventions.....	0
Revision History.....	0
C1. Introduction.....	1
1.1 Overview.....	1
1.2 Feature.....	1
1.3 Specification.....	2
1.4 Dimension.....	4
C2. Hardware Overview	5
2.1 System Overview.....	5
2.1.1 Interface Overview	5
2.1.2 Mainboard Overview	6
2.1.3 Power Diagram.....	7
2.1.4 Block Diagram	8
2.1.5 IIC Diagram	9
2.2 Interface Description	10
2.2.1 LED Indicator Status	10
2.2.2 Buzzer	12
2.2.3 RS485	12
2.2.4 RS232	13
2.2.5 DI	14
2.2.6 DO	15
2.2.7 Boot Switch	16
2.2.8 USB.....	16
2.2.9 SIM Slot(Internal)	16
2.2.10 M.2 M-KEY Slot.....	17
2.2.11 Mini-PCle Slot.....	18
2.2.12 M.2 B-KEY Slot.....	18
2.2.13 Reset Hole	19
2.2.14 Ethernet RJ45.....	19
2.2.15 HDMI	19
2.2.16 RTC	19
2.2.17 Watchdog	20
2.3 Optional Interfaces and Module	20
2.3.1 Wi-Fi/BLE	21
2.3.2 4G Module	21

2.3.3 5G Module	21
2.3.4 LoRa® Module	21
2.3.5 Zigbee Module	22
2.3.6 SSD	22
2.3.7 Encryption Chip TPM 2.0	22
2.3.8 UPS	23
C3. Configuring System	24
3.1 Flashing Image	24
3.2 Query GPIO Mappings	27
3.3 USER LED Testing	27
3.4 SPI Communication Testing	27
3.5 Wi-Fi Scanning	28
3.6 Bluetooth Scanning	28
3.7 LoRa® over Mini-PCle	28
3.7.1 LoRa® SPI	28
3.7.2 LoRa® USB	32
3.8 5G Cellular over M.2 B-KEY	33
3.9 4G Cellular over M.2 B-KEY	35
3.10 Zigbee over Mini-PCle	36
3.11 RS485 Testing	36
3.12 RS232 Testing	38
3.13 DI Testing	39
3.14 DO Testing	40
3.15 USB Hub Testing	41
3.16 RTC	41
3.17 Watchdog	41
3.18 Buzzer	43
3.19 TPM 2.0	43
3.20 ATECC608A	43
3.21 EEPROM	44
3.22 SSD	44
3.23 UPS for Safe Shut Down	44
3.24 AI Accelerator	48
3.25 Installing Ubuntu on reComputer Industrial R2000	49
3.26 Customized Linux: Yocto and Mender	49
3.27 Costomized Linux: Buildroot	50
C4. Assembly Guide	51
4.1 Disassembly Guide	51

4.2 Assemble Nano SIM Card 51

4.3 Assemble SSD 52

4.4 Assemble M.2 AI Accelerator 52

4.5 Assemble 5G/4G/LoRa®/Zigbee Module and Antenna 53

4.6 Assemble TPM 2.0 Module 54

4.7 Assemble UPS module 54

4.8 Mounting Guide 55

C5. Accessories List 56

C6. Warranty & Support 57

6.1 Warranty 57

6.2 Support 57

C1. Introduction

1.1 Overview

The reComputer Industrial R2000, powered by Raspberry Pi CM5, is an adaptable edge AI/IOT controller with AI capabilities. It features comprehensive industrial interfaces (2x Ethernet, 4xUSB, 3x RS485, 1x RS232, 8x DI and 8x DO) and flexible wireless connectivity options (4G/5G, LoRa®, Wi-Fi/BLE), making it ideal for diverse industrial applications.

Application

The reComputer Industrial R2000 series is ideal for use in industrial AI applications. It can be used in aspects such as data acquisition and process monitoring, automation and robot control, intelligent manufacturing, and industrial communication and networking. With its small size, flexibility, low cost, and programmability, it provides strong support for automation & IoT system and more.

1.2 Feature

Industrial-Grade Reliability

- Fanless compact PC with wide temperature support -20°C to 65°C
- Hardware watchdog
- UPS Supercapacitor(optional)
- High-quality metal case, compatible with DIN-rail and Wall installation
- Production Lifetime: reComputer Industrial R2000 will remain in production until at least December 2036

High-Efficiency AI Computing

- Powered by Raspberry Pi CM5
- Broadcom BCM2711 quad-core Cortex-A72 (ARM v8) 64-bit SoC @ 1.5GHz
- Up to 16GB RAM and 64GB eMMC
- Powered by Hailo-8 AI accelerator up to 26 TOPS(optional)

Rich Wireless Capabilities

- On-chip Wi-Fi
- On-chip BLE
- Mini-PCIe: LTE, USB LoRa®, USB Zigbee
- M.2 B-KEY: 4G/5G

Rich Interfaces

- 3x RS485 (isolated), 1x RS232 (isolated), 8x isolated DI ports, 8x isolated DO ports
- 1x 10M/100M/1000M Ethernet (Support PoE PD)
- 1x 10M/100M Ethernet
- 2x HDMI 2.0
- 3x Type-A USB3.0
- 1x Type-A USB2.0
- 1x Type-C USB2.0 (For flashing OS & Debug)
- 1x Nano SIM card slot

1.3 Specification

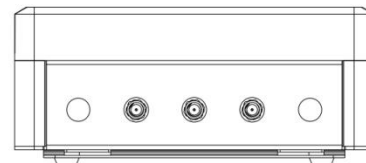
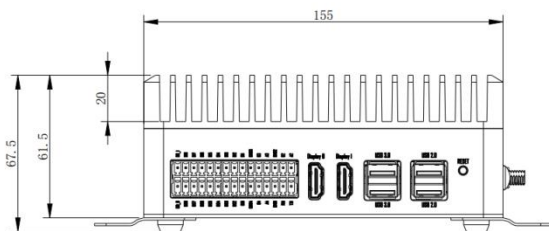
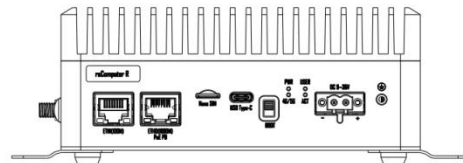
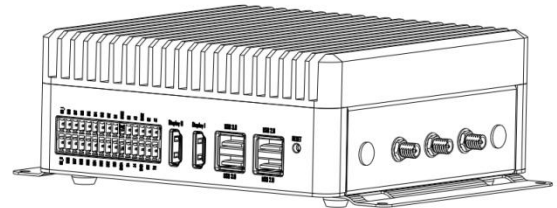
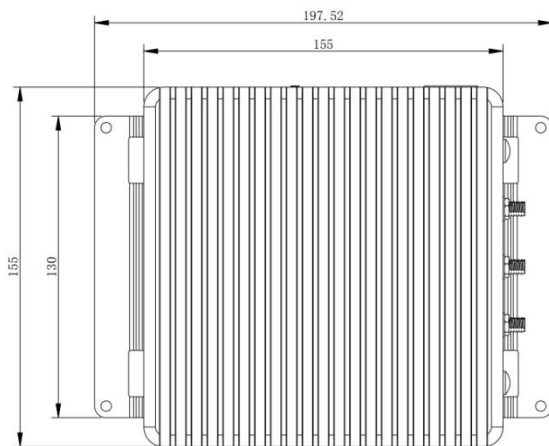
Parameter	Description	
Hardware Specification		
Product Series	R20xx-12	R20xx-10
CPU	Raspberry Pi Compute Module 5, 2.4GHz quad-core 64-bit Arm Cortex-A76	
AI Processor	Pre-installed 1x Hailo-8 M.2 AI Acceleration 26TOPS	*
Operating System	Raspbian, Debian	
RAM	2GB/4GB/8GB/16GB	
eMMC	16GB/32GB/64GB	
System Specification		
Power Input	DC 9V~36V, 2-pin Terminal Block	
PoE(as powered device)	IEEE 802.3at Standard 25.5W PoE	
Power Switch	No	
Reboot Switch	Yes	
Interface		
Ethernet	1 x 10/100/1000 Mbps(supports PoE PD)	
	1 x 10/100 Mbps IEEE 802.3/802.3u	
USB	3 x USB-A 3.0 Host; 1 x USB-A 2.0 Host	
	1 x USB-C 2.0 (For flashing OS & Debug)	
RS485	3x RS485(Isolated)	
RS232	1x RS232(Isolated)	
DI	8 x Isolated DI Ports	
	Input Voltage: 5~24V DC	
DO	8 x Isolated DO Ports	
	Output Voltage: <60V DC	
SIM Card	1x Nano SIM Card Slot	
M.2 Slot	1x M.2 M-KEY 2280 Slot for ;NVMe SSD ; 1x M.2 M-KEY 2280 Slot for AI Accelerator 1x M.2 NVMe Slot, t, M.2 M-KEY 2280 (Pre-installed 1x Hailo-8 M.2 AI Acceleration 26TOPS); 1x M.2 B-KEY 3042/3052 Slot for 5G/4G LTE;	1x M.2 M-KEY 2280 Slot for NVMe SSD ; 1x M.2 M-KEY 2280 Slot for AI Accelerator; 1x M.2 B-KEY 3042/3052 Slot for 5G/4G LTE 2x M.2 NVMe Slot, t, M.2 M-KEY 2280; 1x M.2 NVMe Slot, t, M.2 B-KEY
Mini-PCle	1x Mini-PCle for LoRa module	
LED	4 x LED indicators	
Buzzer	1	

Reset Button	1
HDMI	2 x HDMI 2.0
Wireless Communication	
Wi-Fi 2.4/5.0 GHz	On-chip Wi-Fi
BLE 5.0	On-chip BLE
LoRa®	USB LoRa®*/SPI LoRa®*
4G/5G Cellular	4G LTE/5G*
Zigbee	USB Zigbee*
Standards	
EMC	ESD: EN61000-4-2, Level 3
	EFT: EN61000-4-4, Level 2
	Surge: EN61000-4-5, Level 2
Certification	CE, FCC
	TELEC
	RoHS
	REACH
Ambient Conditions	
Ingress Protection	IP40
Operating Temperature	-20~65 °C
Operating Humidity	10~95% RH
Storage Temperature	-40~85 °C
Others	
Supercapacitor UPS	SuperCAP UPS LTC3350 Module*
Hardware Watchdog	1~255s
RTC	High Accuracy RTC
Security	Encryption Chip TPM 2.0*
	ATECC608A
Heat Dissipation	Fanless
Warranty	2 years
Production Lifetime	Until December 2036
Statement	Options marked with * require additional purchase according to the accessories list.

Component and Interface Status Statement	
Reserved	Designated for future use or expansion.
Optional	Non-essential components, users can choose to include or exclude.
Occupied	Currently in use and integral to product functionality.
Included	Essential components provided with standard package.

1.4 Dimension

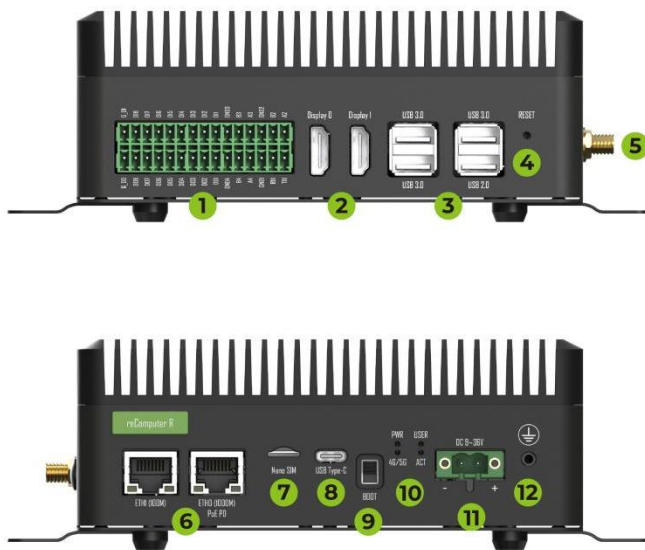
Mechanical	
Dimension(W x H x D)	197.5 mm x 155 mm x 67.5 mm
Enclosure	6063 aluminum alloy heat sink, 5052 aluminum alloy enclosure
Mounting	DIN-rail/Wall
Weight(Net)	1280g



C2. Hardware Overview

2.1 System Overview

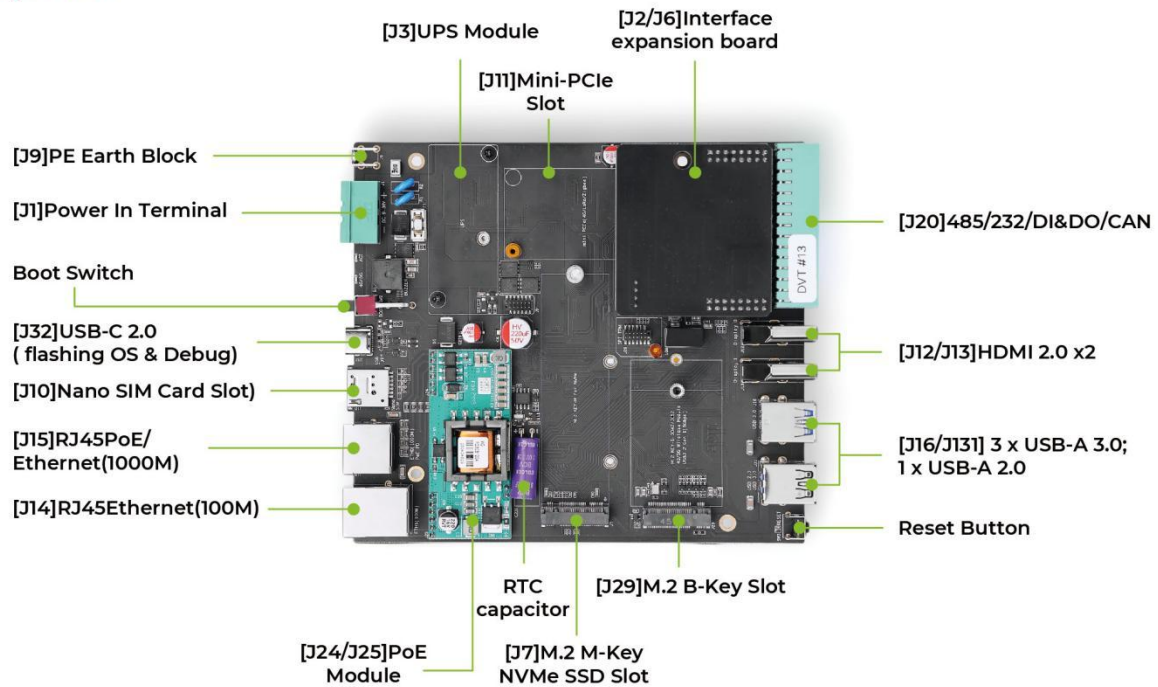
2.1.1 Interface Overview



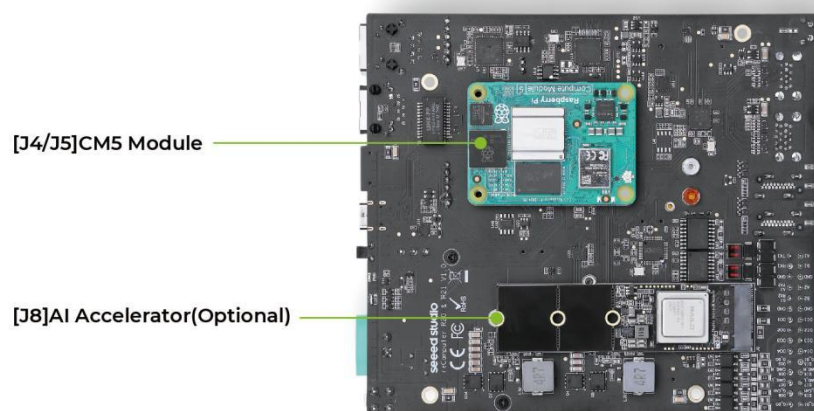
- 1 3x RS485(Isolated) 8 x Isolated DI Ports
1x RS232(Isolated) 8 x Isolated DO Ports
- 2 2 x HDMI 2.0
- 3 3 x USB-A 3.0 Host 1 x USB-A 2.0 Host
- 4 Reset Hole
- 5 5 x Reserved Antenna Ports for Wireless
- 6 1 x 10/100/1000 Mbps(supports PoE PD)
1 x 10/100 Mbps IEEE 802.3/802.3u
- 7 Nano SIM Card Slot
- 8 USB-C 2.0 (For flashing OS & Debug)
- 9 Reboot Switch
- 10 4 x LED indicators
- 11 DC 9V~36V, 2-pin Terminal Block
- 12 Grounding point

2.1.2 Mainboard Overview

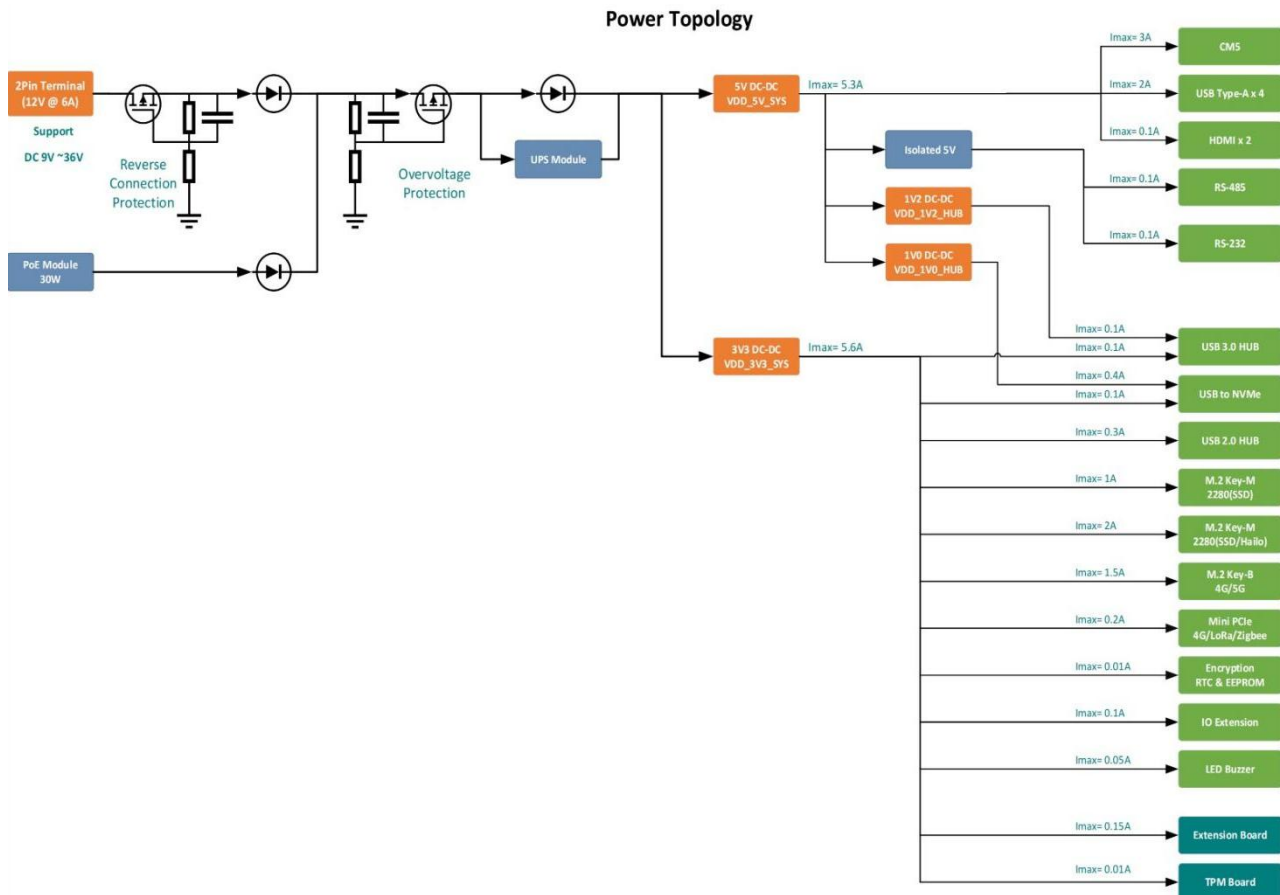
Top View



Bottom View

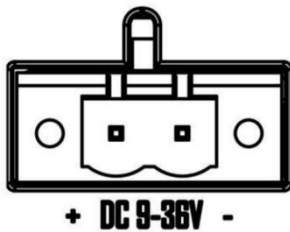


2.1.3 Power Diagram



The reComputer Industrial R2000 supports two power supply options: DC terminal and PoE port. This provides flexibility in power supply selection and allows for easy integration with various power sources.

2-Pin Power Terminal



The reComputer Industrial R2000 is supplied with a terminal DC voltage of 9~36V. The power supply is connected via the 2-pin power terminal block connector.



To ground the reComputer Industrial R2000, the ground wire can be secured to the screw located at the right of the power terminal.

PoE

The ETH0(PoE PD) port of reComputer Industrial R2000 can support PoE power supply, providing a convenient and efficient way to power the device over Ethernet. This option simplifies the installation process and reduces the amount of cabling required, making it an ideal solution for applications with limited power sources or where power outlets are not readily available.

- PoE PD: IEEE 802.3at, 25.5W Max

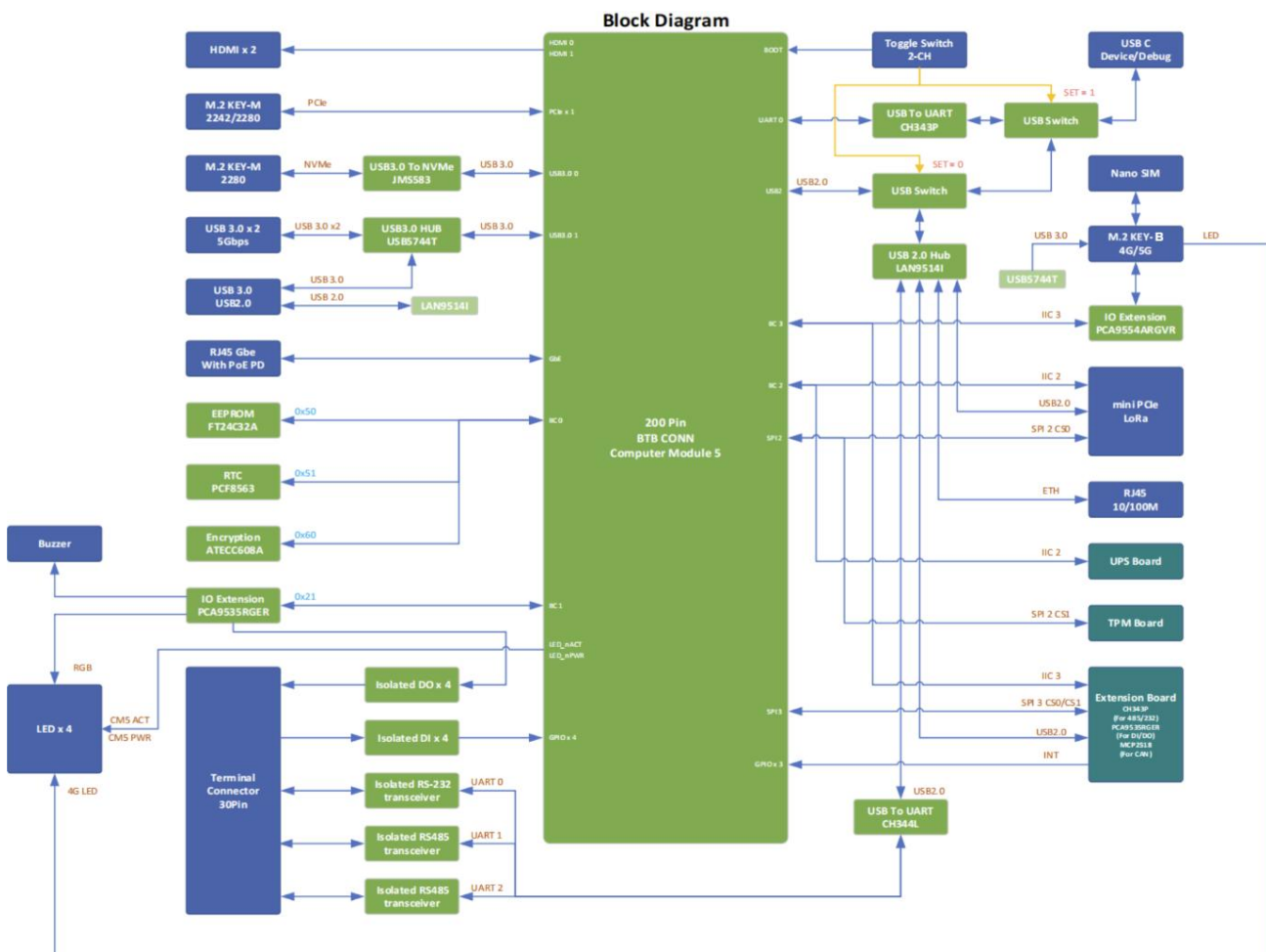
Note

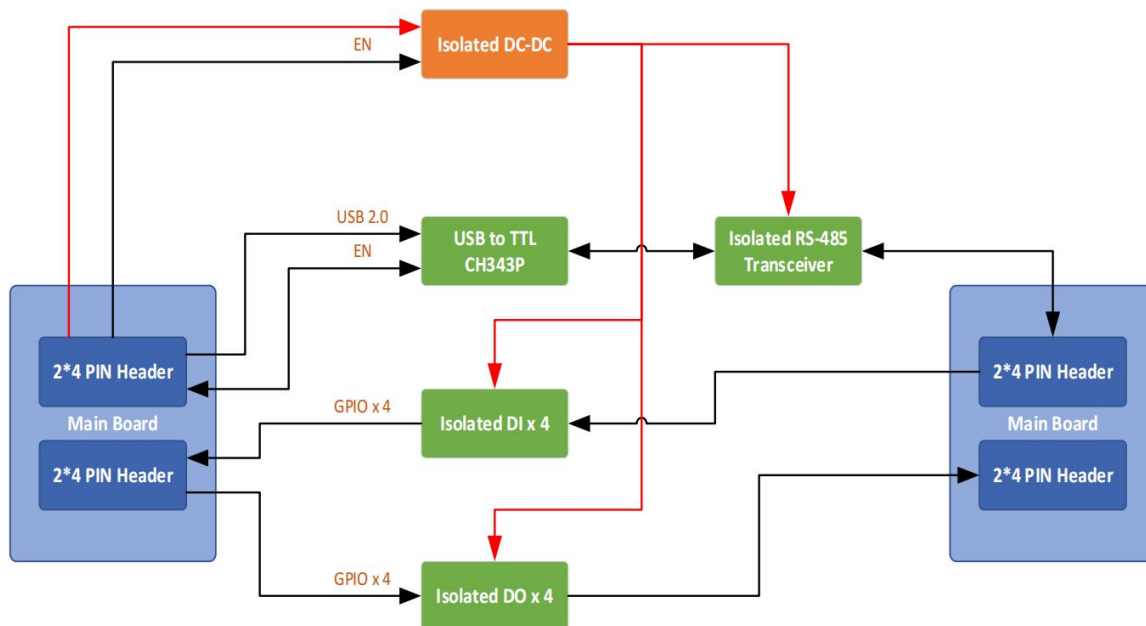
It's worth noting that the PoE module provided with the reComputer Industrial R2000 is compliant with the IEEE 802.3at standard and can provide a maximum power supply of 25W. Therefore, if there is a need to connect high-power peripherals such as 5G or 4G modules, the PoE power supply may not be sufficient. In this case, it's recommended to use the DC terminal for power supply instead to ensure stable and reliable operation of the device.

Power On and Power Off

The reComputer Industrial R2000 does not come with a power button by default, and the system will automatically start up once power is connected. When shutting down, please select the shutdown option in the operating system and wait for the system to fully shut down before cutting off power. To restart the system, simply reconnect to the power.

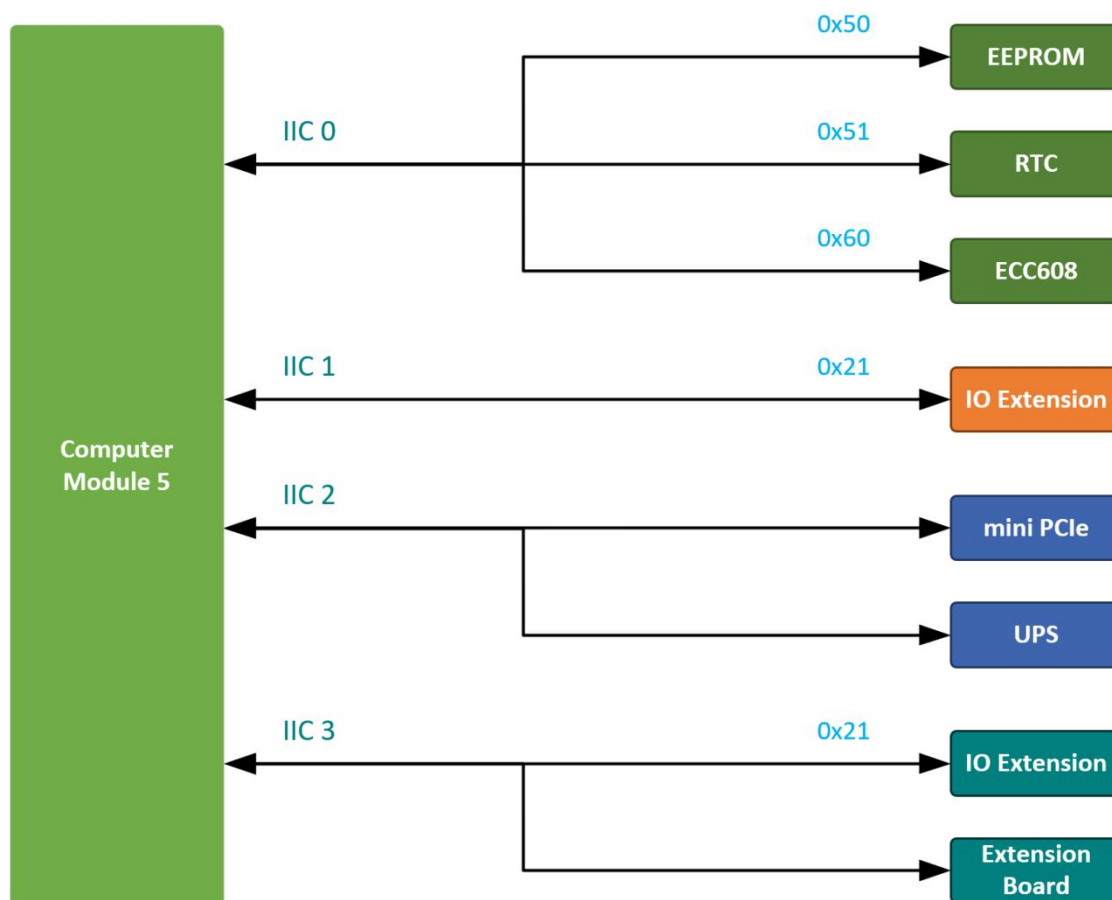
2.1.4 Block Diagram





2.1.5 IIC Diagram

IIC Tree



2.2 Interface Description

Interface	
Ethernet	1 x 10/100/1000 Mbps (supports POE PD)
	1 x 10/100 Mbps IEEE802.3/802.3u
USB	3 x USB-A 3.0 Host; 1 x USB-A 2.0 Host
	1 x USB-C 2.0 (for flashing OS and Debug)
RS485	3x RS485(Isolated)
RS232	1x RS232(Isolated)
DI	8 x Isolated DI Ports
	Input Voltage: 5~24V DC
DO	8 x Isolated DO Ports
	Output Voltage: <60V DC
HDMI	2 x HDMI 2.0
SIM Card Slot	supports Nano SIM Card
M.2 M-KEY Slot	1x 2280 for M.2 NVMe SSD
	1x 2280 for AI Accelerator
M.2 B-KEY Slot	1x for 5G/4G LTE
mini PCIe Slot	1x for 4G LTE/LoRa/Zigbee
LED	4 x LED indicators
Buzzer	1
Reset Button	1

2.2.1 LED Indicator Status

The reComputer Industrial R2000 features 4 LED indicators that serve to signal the machine's operational status. Please refer to the table below for the specific functions and status of each LED:

LED Indicator	Color	Status	Description
PWR	Green	On	The device has been connected to power.
		Off	The device is not connected to power.
ACT	Green		Under Linux this pin will flash to signify eMMC access. If any error occurs during booting, then this LED will flash an error pattern which can be decoded using the look up table(Raspberry Pi Documentation - Configuration) on the Raspberry Pi website.
USER	Green/Red/Blue		Need to be defined by user.
4G/5G	Green	On	The dial-up is successful and the connection is normal.
		Off	4G/5G signal is not connected or the device is not powered on.

ACT Status table

Long flashes	Short flashes	Status
0	3	Generic failure to boot
0	4	start*.elf not found
0	7	Kernel image not found
0	8	SDRAM failure
0	9	Insufficient SDRAM
0	10	In HALT state
2	1	Partition not FAT
2	2	Failed to read from partition
2	3	Extended partition not FAT
2	4	File signature/hash mismatch - Pi 4
4	4	Unsupported board type
4	5	Fatal firmware error
4	6	Power failure type A
4	7	Power failure type B

If the ACT LED blinks in a regular four blink pattern, it cannot find bootcode(start.elf).

If the ACT LED blinks in an irregular pattern then booting has started.

If the ACT LED doesn't blink, then the EEPROM code might be corrupted, try again without anything connected to make sure. For more detail please check the Raspberry Pi forum:

STICKY: Is your Pi not booting? (The Boot Problems Sticky) - Raspberry Pi Forums.

For more detail please check the Raspberry Pi forum: <https://forums.raspberrypi.com/viewtopic.php?f=28&t=58151>

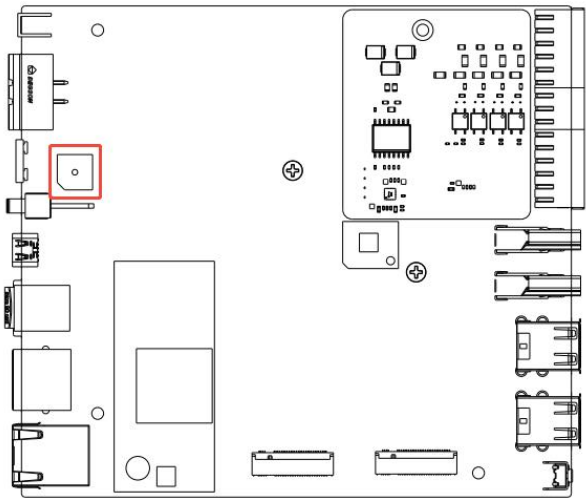
USER Indicator

The reComputer Industrial R2000 contains a USER indicator, and users can customize the status according to actual needs.

Note

For details of USER LED testing, please refer to section 3.3.

2.2.2 Buzzer



The reComputer Industrial R2000 contains an active buzzer, which can be used for various purposes such as alarm and event notifications. Enter in the terminal of reComputer Industrial R2000:

```
cat /sys/kernel/debug/gpio
```

This command will output the GPIO corresponding to the Buzzer_EN is gpio627.

Note

For details of buzzer testing, please refer to section 3.18.

2.2.3 RS485

The reComputer Industrial R2000 series equipment includes 3x RS485 ports, 6-Pin 3.5mm spacing phoenix terminals. The silkscreen of single RS485 is "A/B/GND".

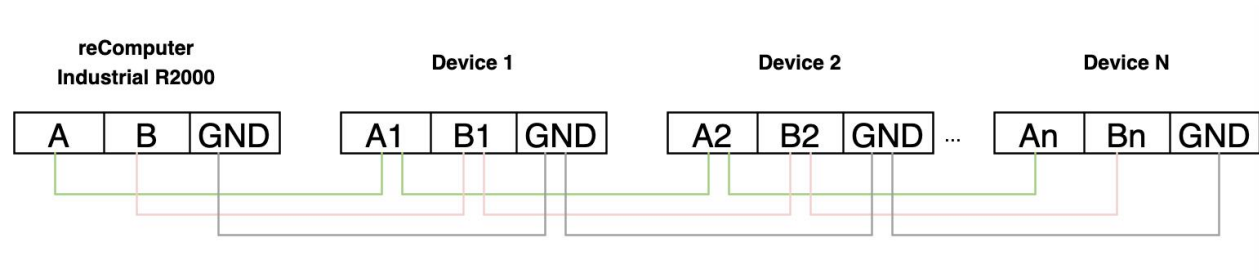
Pin Definition

Terminal pins are defined as follows:

		Pin ID	Pin Name
		1	R485-2_A
		3	R485-2_B
		5	R485-2_GND
		7	R485-3_A
		9	R485-3_B
		11	R485-3_GND
		8	R485-4_A
		10	R485-4_B
		12	R485-4_GND

Connecting Cables

Schematic diagram of RS485 wires is as follws:



Note

The product packaging contains some 120 Ohm terminal resistors. You can use them as needed when communicating via RS485.
For details of RS485 testing, please refer to section 3.11.

2.2.4 RS232

The reComputer Industrial R2000 series equipment includes 1x RS232 ports, 6-Pin 3.5mm spacing phoenix terminals. The silkscreen of single RS232 is "TX/RX/GND".

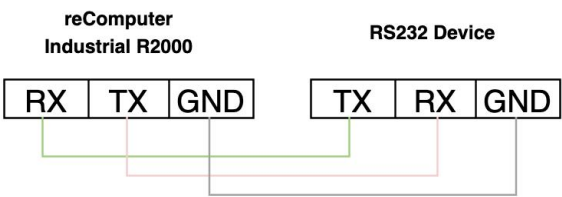
Pin Definition

Terminal pins are defined as follows:

	Pin ID	Pin Name
	2	R232-1_TX
	4	R232-1_RX
	6	R232-1_GND

Connecting Cables

Schematic diagram of RS232 wires is as follws:



Note

For details of RS232 testing, please refer to section 3.12.

2.2.5 DI

The reComputer Industrial R2000 series equipment includes 8x DI ports, 3-Pin 3.5mm spacing phoenix terminals. The silkscreen of single DI is "DI/G_DI".

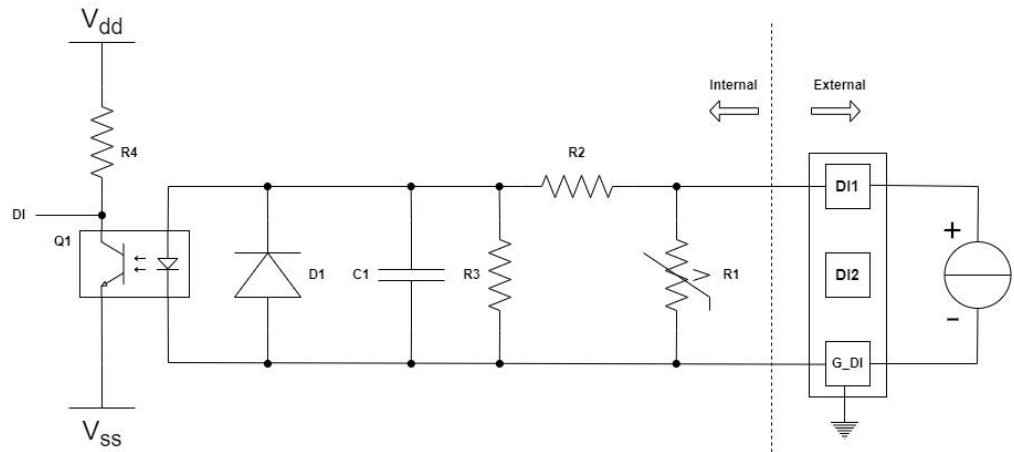
Pin Definition

Terminal pins are defined as follows:

	Pin ID	Pin Name
	13	DI1
	15	DI2
	17	DI3
	19	DI4
	21	DI5
	23	DI6
	25	DI7
	27	DI9
	29	G_DI

Connecting Cables

Schematic diagram of a single DI wires is as follows:



Parameter	Description
Input Type	PNP
Isolation Protection	5 kV
DI to G_DI	ON state: 5~30 VDC

Note

For details of DI testing, please refer to section 3.13.

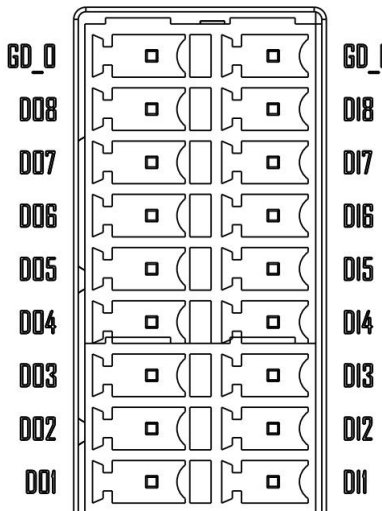
2.2.6 DO

The reComputer Industrial R2000 series equipment includes 8x DO ports, 3-Pin 3.5mm spacing phoenix terminals.

The silkscreen of single DO is "DO/G_DO".

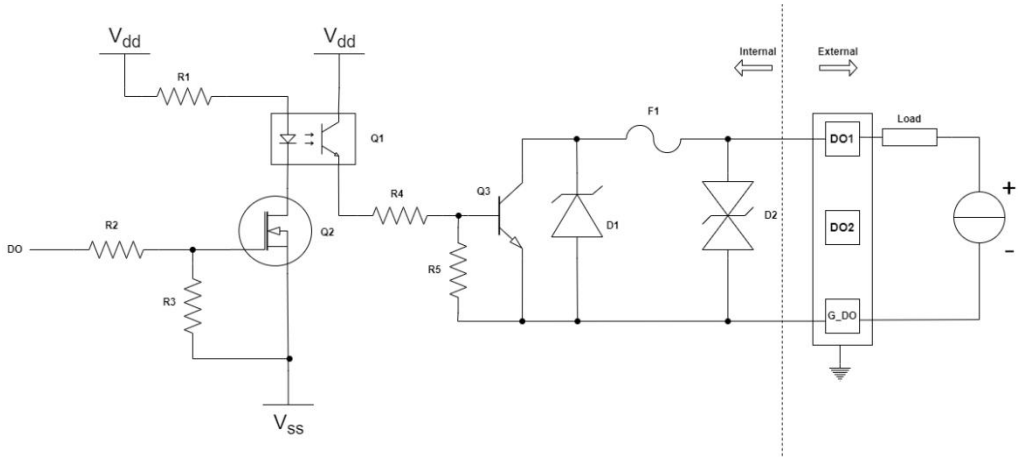
Pin Definition

Terminal pins are defined as follows:

	Pin ID	Pin Name
	14	DO1
	16	DO2
	18	DO3
	20	DO4
	22	DO5
	24	DO6
	26	DO7
	28	DO8
	30	G_DO

Connecting Cables

Schematic diagram of a single DO wires is as follows:



Parameter	Description
Onput Type	Transistor
Isolation Protection	5 kV
Output	< 60V DC



Note

For details of DO testing, please refer to section 3.14.

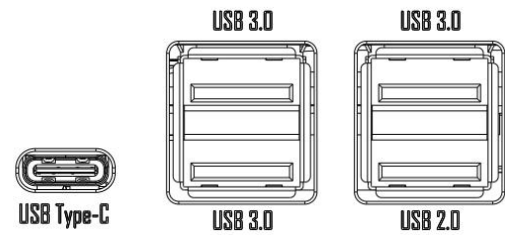
2.2.7 Boot Switch



The Boot Switch of the reComputer Industrial R2000 is connected to the nRPI_BOOT pin of CM5. This switch provides users with the option to select the boot source between eMMC and USB. In normal mode, the switch should be set away from the side with the "BOOT" label, enabling the system to boot from eMMC. Conversely, when users need to flash the system image, they should set the switch towards the "BOOT" label, allowing the system to boot from the Type-C USB interface.

Switch Position	Mode	Description	nRPI-BOOT
	Normal mode	Boot from eMMC	Low
	Flash mode	Boot from USB	High

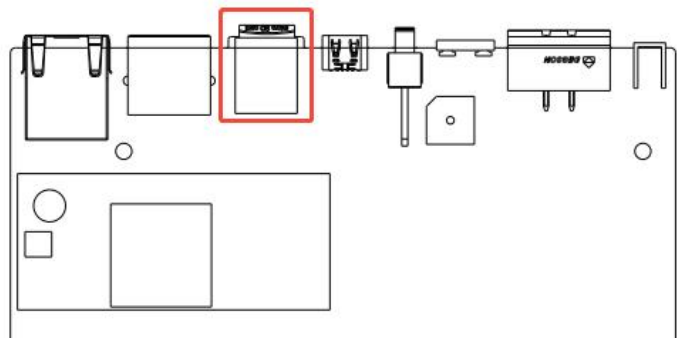
2.2.8 USB



The reComputer Industrial R2000 is equipped with 1x USB Type-C port and 4x USB Type-A ports. Please refer to the table below for their functions and descriptions.

Type	Quantity	Protocol	Function	Description
Type-C	*1	USB2.0	USB-Device	Used for serial port debugging, burning image, etc.
Type-A	*1	USB2.0	USB-Host	Connect different USB devices such as flash drives, USB keyboards or mice.
Type-A	*3	USB3.0	USB-Host	Connect different USB devices such as flash drives, USB keyboards or mice.

2.2.9 SIM Slot(Internal)



The reComputer Industrial R2000 series equipment includes an internal Nano SIM card slot, which is used to install Nano SIM card for obtaining 5G/4G signals.

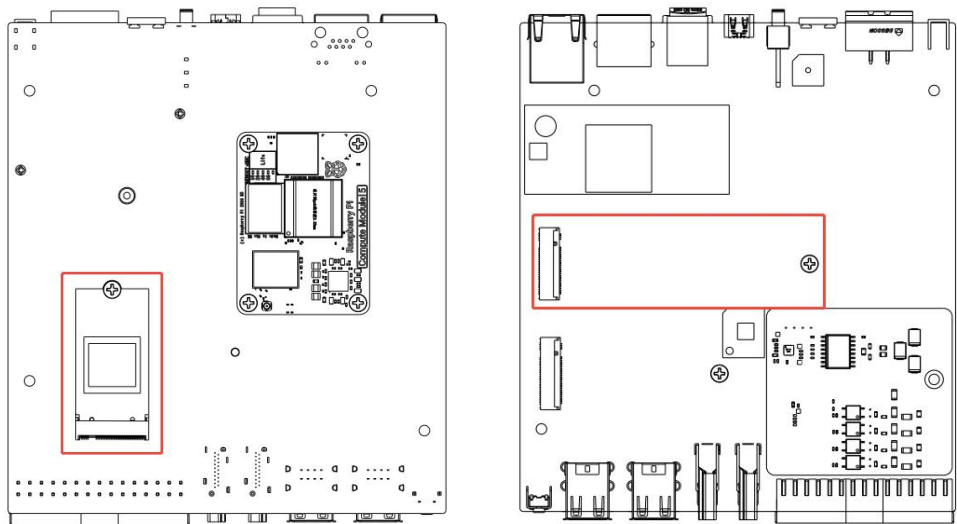
The size differences between standard SIM, Micro SIM and Nano SIM cards are as follows:



Note

Please note that the standard version of reComputer Industrial R2000 does not come with a 5G/4G module. If you require 5G/4G functionality, an additional 5G/4G module must be purchased separately. For more information, please refer to section "2.3.2 4G Module".

2.2.10 M.2 M-KEY Slot



Slot	Supported Protocol
M.2 M-KEY 2280	M.2 NVMe SSD
M.2 M-KEY 2280	M.2 AI Accelerator

The M.2 M-KEY 2280 slot on the reComputer Industrial R2000 is designed to accommodate NVMe M.2 2280 SSDs for 128GB, 256GB, 512GB, 1TB and 2TB in capacity. This slot allows for high-speed storage expansion, enabling users to enhance the performance and capacity of their system.

Note

There are two main uses for SSD cards:

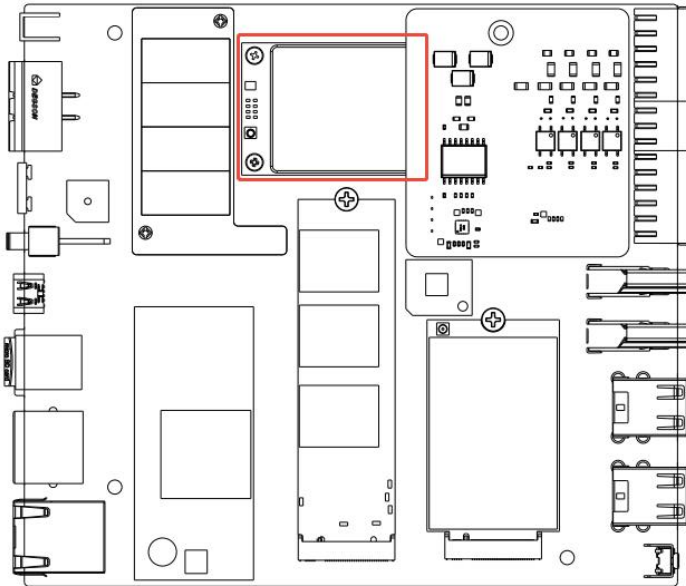
- High Capacity Storage:** SSD cards can be utilized for high-capacity storage needs.
- Boot Drive with Image:** Another usage involves using the SSD both as a high-

capacity storage and for storing system images, allowing booting directly from the SSD card.

It's important to note that not all SSD cards available in the market support the second usage. Therefore, if you intend to use it as a boot drive and are unsure about which model to purchase, we recommend opting for our recommended 2TB SSD(SKU 114993467). This model has been tested and verified for boot functionality, reducing the risk of compatibility issues and minimizing trial and error costs.

The M.2 M-KEY 2280 slot on the reComputer Industrial R2000 is designed to accommodate PCIe M.2 AI Accelerator. And the R20xx-12 series has been pre-installed with a Hailo-8 M.2 AI Acceleration up to 26TOPS.

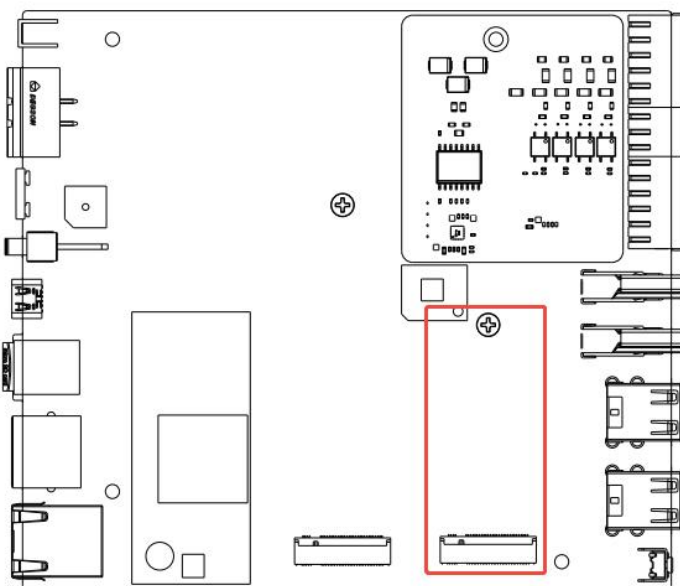
2.2.11 Mini-PCle Slot



The MiniPCle slot on the reComputer Industrial R2000 is designed to accommodate devices such as 4G LTE, USB LoRaWAN®, and USB Zigbee.

2.2.12 M.2 B-KEY Slot

The M.2 B-KEY slot on the reComputer Industrial R2000 is designed to accommodate 5G/4G LTE.

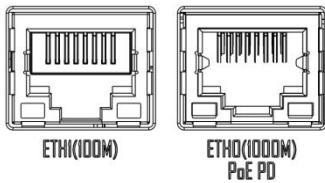


2.2.13 Reset Hole



There is a Mini Push Button Switch located in the reset hole of reComputer Industrial R2000. By pressing this button with a thin object, the CM5 can be reset. This pin when high signals that the CM5 has started. Driving this pin low resets the module

2.2.14 Ethernet RJ45



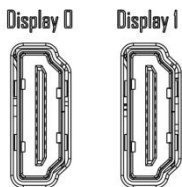
Name	Type	Speeds	PoE PD
ETH0	CM5 native Gigabit Ethernet	10/100/1000 Mbit/s	Supported
ETH1	Converted from USB	10/100 Mbit/s	Not Supported

The reComputer Industrial R2000 comes with two Ethernet RJ45 ports. ETH0 is a CM5 native Gigabit Ethernet interface that supports three different speeds: 10/100/1000 Mbit/s. It can enable power-over-Ethernet (PoE) delivery through this interface, providing power to the reComputer Industrial R2000. Another one ETH1 supports 10/100 Mbit/s which is converted from USB.

Note

For more detail about PoE, please check section "2.3.5 PoE".

2.2.15 HDMI



The reComputer Industrial R2000 features 2x native HDMI interface from CM5, supporting up to 4K @ 60 fps video output. It is ideal for applications that require multiple displays, allowing users to output their content to external large screens.

2.2.16 RTC



The reComputer Industrial R2000 features an RTC circuit that powered by the capacitor, enabling it to maintain timekeeping functionality even in the event of power loss.

Note

For details of RTC testing, please refer to section 3.16.

2.2.17 Watchdog

The reComputer Industrial R2000 comes equipped with an independent hardware watchdog circuit that ensures automatic system reboot in case of abnormal system crashes. The watchdog circuit is implemented through RTC and allows for flexible feeding times from 1 to 255 seconds.

Note

For details of watchdog testing, please refer to section 3.17.

2.3 Optional Interfaces and Module

The reComputer Industrial R2000 supports a rich selection of expansion modules and accessories, making it suitable for a wide range of scenarios and requirements. If you are interested in customizing the reComputer Industrial R2000, please contact contactodm@seeed.cc for more information.

Here is the accessories and optional modules list:

Remark	Item	Product Name	SKU
Must be used together for LoRaWAN® Function	LoRa® Module	Region optional LoRaWAN® Gateway Module(SPI)-US915	114992969
		Region optional LoRaWAN® Gateway Module(USB)-US915	114992629
		Region optional LoRaWAN® Gateway Module(USB)-US915	114992991
		Region optional LoRaWAN® Gateway Module(SPI)-EU868	114993268
		Region optional LoRaWAN® Gateway Module(SPI)-EU868	114992549
		Region optional LoRaWAN® Gateway Module(USB)-EU868	114992628
	LoRa® Antenna	LoRa® Antenna Kit - 868-915 MHz	110061501
	Zigbee Module	Mini-PCIe USB Zigbee Module	110992005
	Zigbee Antenna	Zigbee Antenna Kit for reComputer R	110061641
4G antenna with 4G module for 4G function, GPS antenna with 4G module for GPS function	4G module	LTE Cat 4 EC25-AFXGA-Mini-PCIe Module - for North American	113991134
		LTE Cat 4 EC25-EUXGR-Mini-PCIe Module - for EMEA and Thai	113991135
		LTE Cat 4 EC25-AUXGR-Mini-PCIe Module - for Australia	113991174
		LTE Cat 4 EC25-EFA-Mini-PCIe Module - for Thai	113991214
		LTE Cat 4 EC25-EMGA-Mini-PCIe Module - for Malaysia	113991234
		LTE Cat 4 EC25-JFA-mini-PCIe	113991296
	4G Antenna	4G Antenna Kit for 4G module	110061502
	GPS Antenna	GPS Antenna Kit for EC25 4G Module	110061521
	Encryption Chip	TPM 2.0 Module with infineon SLB9670	114993114

	TPM 2.0		
	SSD card	NVMe M.2 2280 SSD 2TB	114993467
		NVMe M.2 2280 SSD 1TB	112990267
		512GB NVMe M.2 PCIe Gen3x4 2280 Internal SSD	112990247
		256GB NVMe M.2 PCIe Gen3x4 2280 Internal SSD	112990246
		128GB NVMe M.2 PCIe Gen3x4 2280 Internal SSD	112990226
	UPS	SuperCAP UPS LTC3350 Module	110992004

The reComputer Industrial R2000 mainboard features two Mini-PCIe slots. Mini-PCIe slot 1 supports 4G module, LoRa® module using the USB protocol and Zigbee module using USB protocol; while Mini-PCIe slot 2 supports LoRa® module using the USB and SPI protocol and Zigbee module using USB protocol. Additionally, 4G module and LoRa® module shouldn't be used at the same time, can not plug in two LoRa® modules on board.

Note

Can not plug in 2 LoRa® modules on board.

2.3.1 Wi-Fi/BLE

The reComputer Industrial R2000 is powered by the CM5 with an onboard Wi-Fi/BLE version, providing the same Wi-Fi/BLE parameters as the CM5. For detailed parameter information, please refer to the Raspberry Pi official website.

2.3.2 4G Module

The reComputer Industrial R2000 mainboard features one M.2 B-KEY Slot and one Mini-PCIe Slot, Both M.2 B-KEY Slot and Mini-PCIe slot supporting a 4G module. The EC25 4G module from Quectel has been fully tested to be compatible with the reComputer Industrial R2000.

Note

Please note that if you require 4G functionality, it is necessary to purchase the corresponding 4G module and external antenna, and follow the instructions in section 4.5 "Assemble 4G/LoRa®/Zigbee Module and Antenna".

2.3.3 5G Module

The reComputer Industrial R2000 mainboard features one M.2 B-KEY Slot, supporting a 4G/5G module using the USB protocol. The EC25 4G module from Quectel has been fully tested to be compatible with the reComputer Industrial R2000.

Note

Please note that if you require 4G/5G functionality, it is necessary to purchase the corresponding 4G module and external antenna, and follow the instructions in section 4.5 "Assemble 4G/LoRa®/Zigbee Module and Antenna".

2.3.4 LoRa® Module

The Mini-PCIe slot supports LoRa® module using the USB and SPI protocol. The WM1302 module from Seeed Studio has been fully tested to be compatible with the reComputer Industrial R2000.

Note

Please note that if you require LoRa® functionality, it is necessary to purchase the corresponding LoRa® module and external antenna, and follow the instructions in section 4.5 "Assemble 4G/LoRa®/Zigbee Module and Antenna".

2.3.5 Zigbee Module

The Mini-PCIe slot offer support for Zigbee modules utilizing the USB protocol, allowing for seamless integration of Zigbee functionality into compatible devices. This feature enables efficient communication and control within Zigbee networks, enhancing the versatility and connectivity of the system. With the Mini-PCIe slot available for Zigbee modules, users have the flexibility to implement diverse applications for enhanced reliability.

Note

Please note that if you require Zigbee functionality, it is necessary to purchase the corresponding Zigbee module and external antenna, and follow the instructions in section 4.5 "Assemble 4G/LoRa®/Zigbee Module and Antenna".

2.3.6 SSD

The reComputer Industrial R2000 supports 2280 NVMe SSD through the use of a PCIe slot(J7). It is important to note that the CM5's PCIe is gen2.0 with a maximum theoretical speed of 5Gbps. If you are using a Gen3.0 or higher SSD, it may not be able to achieve the SSD's maximum speed. After testing, the reTerminal DM with installed SSD can achieve a maximum write speed of 230MB/s and a maximum read speed of 370MB/s. If you are unsure which SSDs are compatible, you can purchase following the accessories list below.

Note

Please note that:

1- The speed test results may vary depending on the SSD model, testing method, and testing environment. The values provided here are for reference purposes only and were obtained in Seeed's laboratory.

Note

There are two main uses for SSD cards:

1.High Capacity Storage: *SSD cards can be utilized for high-capacity storage needs.*

2.Boot Drive with Image: *Another usage involves using the SSD both as a highcapacity storage and for storing system images, allowing booting directly from the SSD card.*

It's important to note that not all SSD cards available in the market support the second usage. Therefore, if you intend to use it as a boot drive and are unsure about which model to purchase, we recommend opting for our recommended 1TB SSD(SKU 112990267). This model has been tested and verified for boot functionality, reducing the risk of compatibility issues and minimizing trial and error costs.

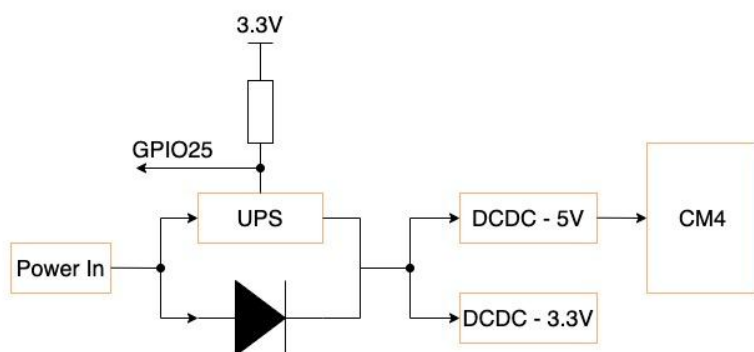
2.3.7 Encryption Chip TPM 2.0

The TPM features Infineon's OPTIGA™ TPM SLB9670 which is compliant to the Trusted Computing Group (TCG) TPM 2.0 specification is recommended as encryption chip to the reComputer Industrial R2000. The chip features an SPI interface applied for port J26 on board, to enable a root of trust for platform integrity, remote attestation, and cryptographic services.

Note

Please refer to section 4.6 "Assemble TPM 2.0 Module" for instruction.

2.3.8 UPS



The UPS is 7F, which operates in series. The UPS module is positioned between the DC5V and CM5 components, with a GPIO signal utilized to alert the CPU in the event of a power loss from the 5V supply. Upon receiving this signal, the CPU executes an urgent script before the super capacitor's energy is depleted, initiating a "\$ shutdown" command. The backup duration provided by the UPS heavily relies on the system load. Below are some typical scenarios tested with a CM5 module featuring 4GB RAM, 32GB eMMC storage, and a Wi-Fi module.

Mode of Operation	Time(s)	Remark
Idle	15	Testing under idle conditions with official driver program loaded
Full load of CPU	6	stress -c 4 -t 10m -v &

Note

For UPS function please contact us for more information, and the alarm signal is active LOW.

C3. Configuring System

3.1 Flashing Image

Note

The device comes pre-installed and pre-burned with the system image.

After powering on for the first time, you can log in using the following default account:

- Username: **recomputer**
- Password: **12345678**

It is recommended that you change your password immediately after logging in for security reasons.

Here are the steps to re-burn the image:

1. Image Download

Download address: [GitHub - Seeed-Studio/pi-gen-expand: This project maintains Raspberry OS Image for Seeed's Raspberry-related products](#)

Select the last one and click the date link to download:

Name	username & password	enable-ssh	stage-list	date
raspberrypi-arm64	pi & raspberry	1	stage0 stage1 stage2 stage3 stage4	2025-02-24
reTerminal-arm64	pi & raspberry	1	stage0 stage1 stage2 stage3 stage4 stage4a	2025-02-24
reComputer-R100x-arm64	recomputer & 12345678	1	stage0 stage1 stage2 stage3 stage4 stage4a	2025-02-24
reComputer-R110x-arm64	recomputer & 12345678	1	stage0 stage1 stage2 stage3 stage4 stage4a	2025-02-24
reComputer-AI-box-arm64	recomputer & 12345678	1	stage0 stage1 stage2 stage3 stage4 stage4a	2025-02-24
reTerminal-DM-arm64	pi & raspberry	1	stage0 stage1 stage2 stage3 stage4 stage4a	2025-02-24
reComputer-AI-box-cm5-arm64	recomputer & 12345678	1	stage0 stage1 stage2 stage3 stage4 stage4a	2025-03-25
reComputer-R2x-arm64	recomputer & 12345678	1	stage0 stage1 stage2 stage3 stage4 stage4a	2025-07-27

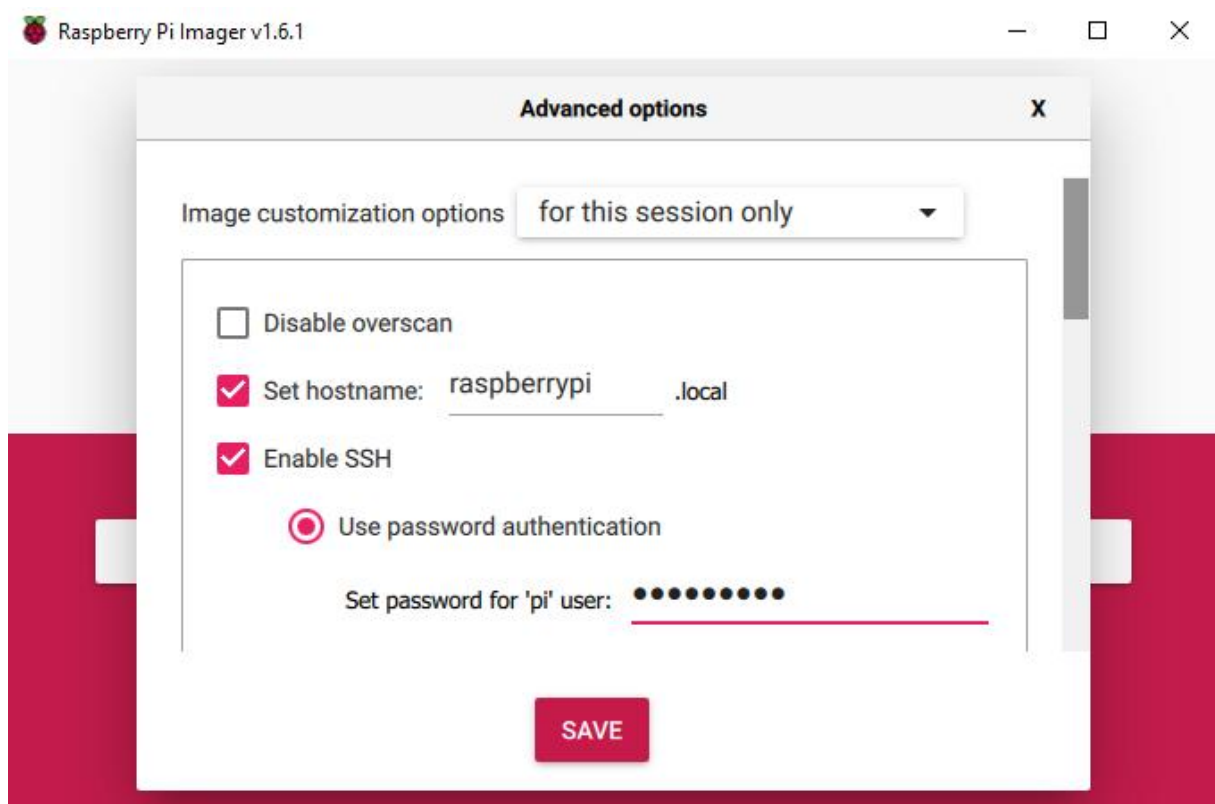
If you encounter network issues, you can use the wget command to download.

2. Image Burn

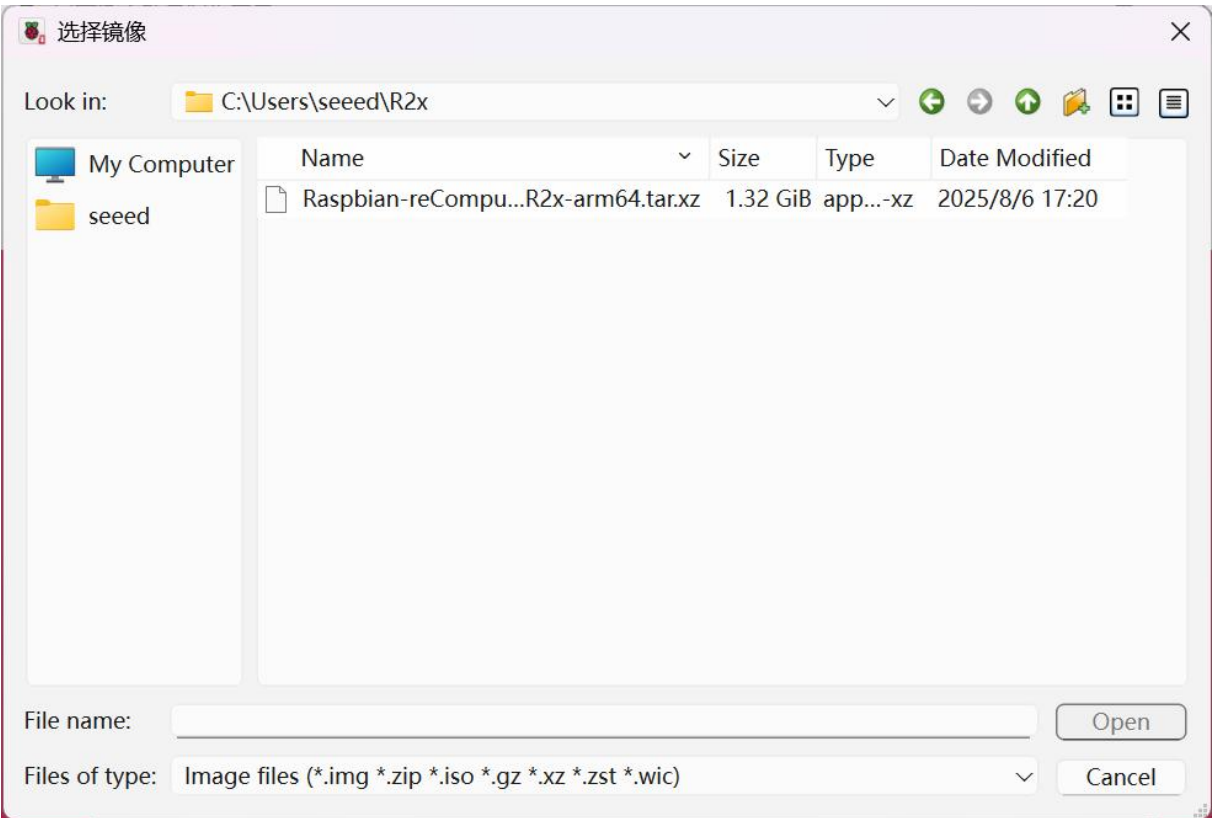
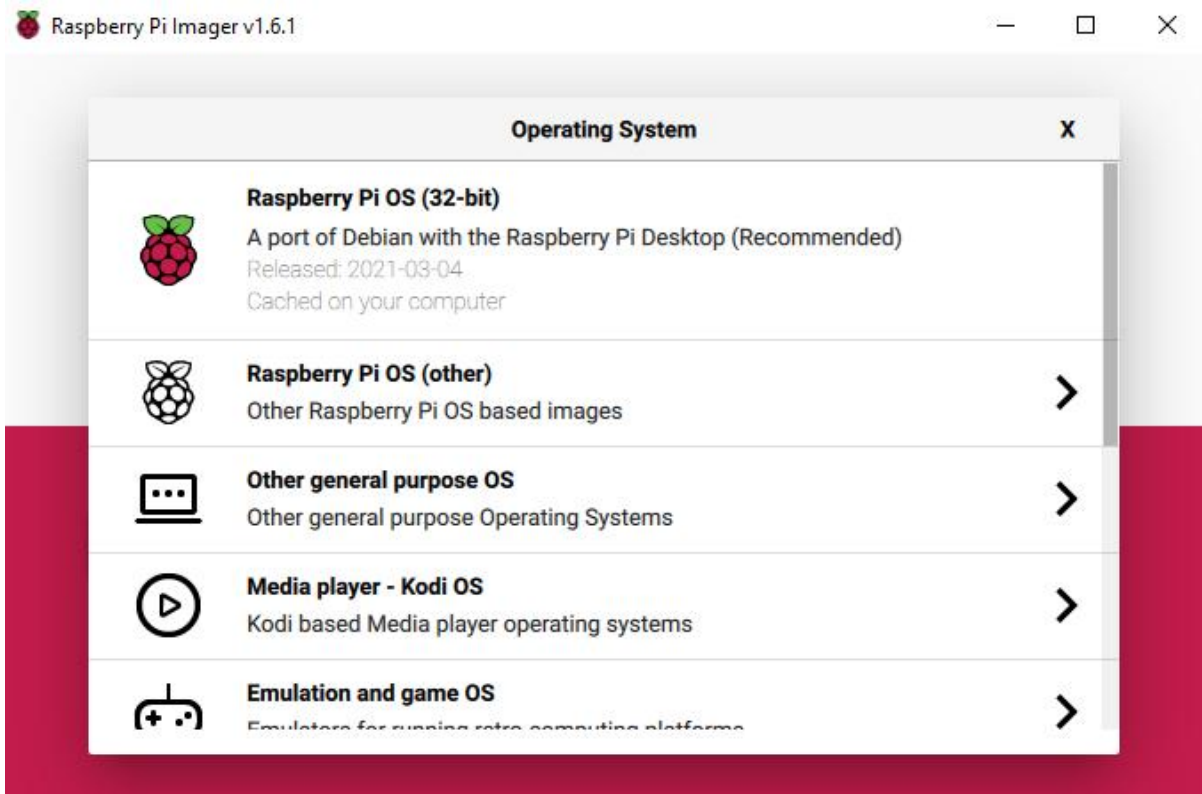
- Download the rpiboot setup installer by click [here](#) to install the necessary drivers and the boot tool.
- Connect reComputer R2000 the PC via USB Type-C cable.
- Windows will now find the hardware and install the necessary drivers.
- Search for rpiboot tool that we installed before and open it.
- Open file explorer and you will see the eMMC of the Computer Module 5 shown as a USB mass storage device.
- Download Raspberry Pi Imager software from [Raspberry Pi software – Raspberry Pi](#).
- Open Raspberry Pi Imager software.



h. Press CTRL + SHIFT + X on the keyboard to open Advanced options window.



i. Select the downloaded image file.



Continue burning the image until it succeeds.

Unplug the USB-C port, press the boot button again, and power on/off. If you can log in to the system normally, the image has been burned successfully and can be used normally.

3.2 Query GPIO Mappings

To query GPIO mappings and offsets, follow these steps:

1. Copy and paste the following command to query GPIO mappings:

```
cat /sys/kernel/debug/gpio
```

This command will provide you with the necessary information regarding GPIO mappings and offsets whenever needed throughout the process.

3.3 USER LED Testing

We provide LEDs in three colors of red, blue and green for users to use. You can enter the `/sys/class/leds/` directory to view :

```
cd /sys/class/leds/  
ls
```

Use the following command to light up the LED of the corresponding color.

```
sudo su  
echo 1 > /sys/class/leds/led-red//brightness  
echo 1 > /sys/class/leds/led-blue/brightness  
echo 1 > /sys/class/leds/led-green/brightness
```

The third led-user LED will light up.

3.4 SPI Communication Testing

To test SPI communication by shorting the TPM module's MISO and MOSI pins, follow these steps:

1. Clone the spidev-test repository:

```
# Don't forget to connect to network before running command  
git clone https://github.com/rm-hull/spidev-test.git
```

2. Navigate into the spidev-test directory:

```
cd spidev-test
```

3. Compile the spidev_test.c file:

```
gcc spidev_test.c -o spidev_test
```

4. Run the spidev_test program with the following command:

```
./spidev_test -D /dev/spidev10.0 -v -p hello
```

This command tests SPI communication on the specified SPI device (`/dev/spidev10.0`) with verbose output (`-v`) and sends the message "hello" (`-p hello`).

By shorting the TPM module's MISO and MOSI pins, you're effectively creating a loopback scenario, where data sent on MOSI is received on MISO. This setup allows you to test SPI communication without an actual device connected.

3.5 Wi-Fi Scanning

To scan for Wi-Fi networks:

```
sudo iwlist wlan0 scan
```

This command will list available Wi-Fi networks along with their details.

3.6 Bluetooth Scanning

To scan for Bluetooth devices:

```
sudo bluetoothctl
```

This command will open the Bluetooth control interface. From there, you can run additional commands to scan for nearby Bluetooth devices:

```
scan on
```

This command will start scanning for nearby Bluetooth devices. You can then use other commands within the **bluetoothctl** interface to interact with Bluetooth devices, such as pairing or connecting to them.

3.7 LoRa® over Mini-PCle

3.7.1 LoRa® SPI

After install the LoRa® SPI to Mini-PCle slot 2, can configure LoRa® SPI, follow these steps:

1. Clone the **SX1302_HAL** repository:

```
cd ~/
git clone https://github.com/Lora-net/sx1302_hal
```

2. Navigate into the cloned directory:

```
cd sx1302_hal
```

3. Modify the configuration file:

```
sudo nano ./libloragw/inc/loragw_i2c.h
```

Change **#define I2C_DEVICE "/dev/i2c-1"** to **#define I2C_DEVICE "/dev/i2c-2"**.

Press **ctrl+x** to exit, press **y** to save changes, and then press **Enter** to return to the command line page.

4. Add the packet_forwarder/reset_lgw.sh file:

```
sudo nano packet_forwarder/reset_lgw.sh
```

Add the execution code:

```
#!/bin/sh
```

```
# This script is intended to be used on SX1302 CoreCell platform, it performs
# the following actions:
#
#   - export/unexport GPIO23 and GPIO18 used to reset the SX1302 chip and to enable the LDOs
#   - export/unexport GPIO22 used to reset the optional SX1261 radio used for LBT/Spectral Scan
#
# Usage examples:
#   ./reset_lgw.sh stop
#   ./reset_lgw.sh start

# GPIO mapping has to be adapted with HW
#

SX1302_RESET_PIN=632  # SX1302 reset
SX1302_POWER_EN_PIN=633 # SX1302 power enable
SX1261_RESET_PIN=634  # SX1261 reset (LBT / Spectral Scan)
AD5338R_RESET_PIN=623  # AD5338R reset (full-duplex CN490 reference design)

WAIT_GPIO() {
    sleep 0.1
}

init() {
    # setup GPIOs
    echo "$SX1302_RESET_PIN" > /sys/class/gpio/export; WAIT_GPIO
    echo "$SX1261_RESET_PIN" > /sys/class/gpio/export; WAIT_GPIO
    echo "$SX1302_POWER_EN_PIN" > /sys/class/gpio/export; WAIT_GPIO
    echo "$AD5338R_RESET_PIN" > /sys/class/gpio/export; WAIT_GPIO

    # set GPIOs as output
    echo "out" > /sys/class/gpio/gpio$SX1302_RESET_PIN/direction; WAIT_GPIO
    echo "out" > /sys/class/gpio/gpio$SX1261_RESET_PIN/direction; WAIT_GPIO
    echo "out" > /sys/class/gpio/gpio$SX1302_POWER_EN_PIN/direction; WAIT_GPIO
    echo "out" > /sys/class/gpio/gpio$AD5338R_RESET_PIN/direction; WAIT_GPIO
}

reset() {
    echo "CoreCell reset through GPIO$SX1302_RESET_PIN..."
    echo "SX1261 reset through GPIO$SX1302_RESET_PIN..."
    echo "CoreCell power enable through GPIO$SX1302_POWER_EN_PIN..."
}
```

```

echo "CoreCell ADC reset through GPIO$AD5338R_RESET_PIN..."

# write output for SX1302 CoreCell power_enable and reset
echo "1" > /sys/class/gpio/gpio$SX1302_POWER_EN_PIN/value; WAIT_GPIO

echo "1" > /sys/class/gpio/gpio$SX1302_RESET_PIN/value; WAIT_GPIO
echo "0" > /sys/class/gpio/gpio$SX1302_RESET_PIN/value; WAIT_GPIO

echo "0" > /sys/class/gpio/gpio$SX1261_RESET_PIN/value; WAIT_GPIO
echo "1" > /sys/class/gpio/gpio$SX1261_RESET_PIN/value; WAIT_GPIO

echo "0" > /sys/class/gpio/gpio$AD5338R_RESET_PIN/value; WAIT_GPIO
echo "1" > /sys/class/gpio/gpio$AD5338R_RESET_PIN/value; WAIT_GPIO
}

term() {
    # cleanup all GPIOs
    if [ -d /sys/class/gpio/gpio$SX1302_RESET_PIN ]
    then
        echo "$SX1302_RESET_PIN" > /sys/class/gpio/unexport; WAIT_GPIO
    fi
    if [ -d /sys/class/gpio/gpio$SX1261_RESET_PIN ]
    then
        echo "$SX1261_RESET_PIN" > /sys/class/gpio/unexport; WAIT_GPIO
    fi
    if [ -d /sys/class/gpio/gpio$SX1302_POWER_EN_PIN ]
    then
        echo "$SX1302_POWER_EN_PIN" > /sys/class/gpio/unexport; WAIT_GPIO
    fi
    if [ -d /sys/class/gpio/gpio$AD5338R_RESET_PIN ]
    then
        echo "$AD5338R_RESET_PIN" > /sys/class/gpio/unexport; WAIT_GPIO
    fi
}

case "$1" in
    start)
        term # just in case
    init

```

```

    reset
    ;;
    stop)
    reset
    term
    ;;
    *)
    echo "Usage: $0 {start|stop}"
    exit 1
    ;;
esac

exit 0

```

Press **ctrl+x** to exit, press **y** to save changes, and then press **Enter** to return to the command line page.

5. Modify the configuration code:

```
sudo vim ./tools/reset_lgw.sh
```

Update the pin configurations:

```

SX1302_RESET_PIN=632  # SX1302 reset
SX1302_POWER_EN_PIN=633 # SX1302 power enable
SX1261_RESET_PIN=634  # SX1261 reset (LBT / Spectral Scan)
# AD5338R_RESET_PIN=13  # AD5338R reset (full-duplex CN490 reference design)

```

Comment out lines 18, 29, 35, 42, 53, and 54 respectively:

```

.....
# echo "$AD5338R_RESET_PIN" > /sys/class/gpio/export; WAIT_GPIO
.....
# echo "out" > /sys/class/gpio/gpio$AD5338R_RESET_PIN/direction; WAIT_GPIO
.....
# echo "CoreCell ADC reset through GPIO$AD5338R_RESET_PIN..."
.....
# echo "0" > /sys/class/gpio/gpio$AD5338R_RESET_PIN/value; WAIT_GPIO
# echo "1" > /sys/class/gpio/gpio$AD5338R_RESET_PIN/value; WAIT_GPIO

```

Press **ctrl+x** to exit, press **y** to save changes, and then press **Enter** to return to the command line page.

6. replace the default **SPI** port of the LoraWAN@Module in the **global_conf.json.sx1250.US915** config file(Configuration files are selected based on the module you are using):

```
sudo nano packet_forwarder/global_conf.json.sx1250.US915
```

Modify the com_path parameter, change **"com_path": "/dev/spidev0.0"** to **"com_path": "/dev/spidev2.0"**.

7. Compile the code:

```
sudo make
```

These steps will configure LoRa® SPI and run the packet forwarder with the specified configuration file.

```
seeed@seeed: ~/sx1302_hal/ x + v
# PUSH_DATA datagrams sent: 0 (0 bytes)
# PUSH_DATA acknowledged: 0.00%
### [DOWNSTREAM] ###
# PULL_DATA sent: 3 (0.00% acknowledged)
# PULL_RESP(onse) datagrams received: 0 (0 bytes)
# RF packets sent to concentrator: 0 (0 bytes)
# TX errors: 0
### SX1302 Status ###
# SX1302 counter (INST): 30732477
# SX1302 counter (PPS): 0
# BEACON queued: 0
# BEACON sent so far: 0
# BEACON rejected: 0
### [JIT] ###
src/jitqueue.c:440:jit_print_queue(): INFO: [jit] queue is empty
#-----
src/jitqueue.c:440:jit_print_queue(): INFO: [jit] queue is empty
### [GPS] ###
# GPS sync is disabled
### Concentrator temperature: 31 C ###
##### END #####

JSON up: {"stat":{"time":"2025-07-16 10:16:32 GMT","rxnb":0,"rxok":0,"rxfw":0,"ackr":0.0,"dwnb":0,"txnb":0,"temp":31.0}}

INFO: Received pkt from mote: 00000009 (fcnt=0)

JSON up: {"rxpk":[{"jver":1,"tmst":31753493,"chan":1,"rfch":0,"freq":904.100000,"mid": 8,"stat":1,"modu":"LORA","datr":"SF10BW125","codr":"4/5","rssi":-139,"lsnr":-18.5,"foff":903,"rssi":-121,"size":23,"data":"AAkAAAAAACAYRgwccDx9yz2/C6xBUE="}]}
```

3.7.2 LoRa® USB

For LoRa® USB, the previous commands remain the same as for LoRa® SPI. However, the final command needs to be changed to:

```
cho 632 > /sys/class/gpio/export
echo "out" > /sys/class/gpio/gpio632/direction
echo "1" > /sys/class/gpio/gpio632/value

sudo ./lora_pkt_fwd -c global_conf.json.sx1250.EU868.USB
```

```

seeed@seeed: ~/sx1302_hal/
JSON up: {"rxpk":[{"jver":1,"tmst":173349739,"chan":1,"rfch":1,"freq":868.300000,"mid": 8,"stat":1,"modu":"LORA","datr":"SF7BW125","codr":"4/5","rssi":-108,"lsnr":8.2,"foff":729,"rssi":-107,"size":22,"data":"gFT3ACeA6wkCDDLKU3p+I0f04KBQWg="}]}

INFO: Received pkt from mote: 27007CF5 (fcnt=13121)

JSON up: {"rxpk":[{"jver":1,"tmst":179162774,"chan":0,"rfch":1,"freq":868.100000,"mid": 8,"stat":1,"modu":"LORA","datr":"SF7BW125","codr":"4/5","rssi":-118,"lsnr":0.5,"foff":-91,"rssi":-116,"size":20,"data":"gPV8ACeAQTMd4Ha0c56XPDVN/s4="}]}

##### 2025-08-13 02:20:54 GMT #####
### [UPSTREAM] ###
# RF packets received by concentrator: 4
# CRC_OK: 100.00%, CRC_FAIL: 0.00%, NO_CRC: 0.00%
# RF packets forwarded: 4 (103 bytes)
# PUSH_DATA datagrams sent: 5 (1130 bytes)
# PUSH_DATA acknowledged: 0.00%
### [DOWNSTREAM] ###
# PULL_DATA sent: 3 (0.00% acknowledged)
# PULL_RESP(onse) datagrams received: 0 (0 bytes)
# RF packets sent to concentrator: 0 (0 bytes)
# TX errors: 0
### SX1302 Status ###
# SX1302 counter (INST): 181200641
# SX1302 counter (PPS): 0
# BEACON queued: 0
# BEACON sent so far: 0
# BEACON rejected: 0
### [JIT] ###
src/jitqueue.c:440:jit_print_queue(): INFO: [jit] queue is empty
#-----
src/jitqueue.c:440:jit_print_queue(): INFO: [jit] queue is empty
### [GPS] ###
# GPS sync is disabled
### Concentrator temperature: 37 C ###
##### END #####

JSON up: {"stat":{"time":"2025-08-13 02:20:54 GMT","rxnb":4,"rxok":4,"rxfw":4,"ackr":0.0,"dwnb":0,"txnb":0,"temp":36.6}}

INFO: Received pkt from mote: 2700F754 (fcnt=2540)

JSON up: {"rxpk":[{"jver":1,"tmst":184672991,"chan":1,"rfch":1,"freq":868.300000,"mid": 8,"stat":1,"modu":"LORA","datr":"SF7BW125","codr":"4/5","rssi":-108,"lsnr":8.2,"foff":729,"rssi":-108,"size":22,"data":"QFT3ACeA7AkC3Ux1AKNaV0Jx/MbXKQ="}]}

```

This command specifies the configuration file to be used for LoRa® USB.

3.8 5G Cellular over M.2 B-KEY

To interact with a 5G/4G module using AT commands via minicom, follow these steps:

1. Create a new power_5g.sh file :

```
nano power_5g.sh
```

Open with `sudo nano` and enter the following command, then press **ctrl+x** to save and exit.

```

#!/bin/bash

RESET_PIN=645
POWER_PIN=639

if [ ! -d "/sys/class/gpio/gpio$RESET_PIN" ]; then
    echo $RESET_PIN > /sys/class/gpio/export
fi

if [ ! -d "/sys/class/gpio/gpio$POWER_PIN" ]; then

```

```
echo $POWER_PIN > /sys/class/gpio/export
fi

echo "out" > /sys/class/gpio/gpio$RESET_PIN/direction
echo "out" > /sys/class/gpio/gpio$POWER_PIN/direction

echo 1 > /sys/class/gpio/gpio$RESET_PIN/value
echo 1 > /sys/class/gpio/gpio$POWER_PIN/value

echo "Start to reboot 5g module"

echo 0 > /sys/class/gpio/gpio$RESET_PIN/value
sleep 0.05
echo 0 > /sys/class/gpio/gpio$POWER_PIN/value

echo "5g module reboot completed"
```

2. Execute the file:

```
sudo ./power_5g.sh
```

After 10-15 seconds (it takes a while for the module to power on and enumerate USB), check whether the device node appears:

```
ls /dev/ttyUSB*
```

Output /dev/ttyUSB0, etc.:

```
seeed@seeed:~$ ls /dev/ttyUSB*
/dev/ttyUSB0 /dev/ttyUSB1 /dev/ttyUSB2 /dev/ttyUSB3
```

Open minicom with the appropriate serial port and baud rate:

```
sudo apt update
sudo apt install minicom
sudo minicom -D /dev/ttyUSB2 -b 115200
```

This command opens minicom with the specified serial port (**/dev/ttyUSB2**) at a baud rate of 115200.

1. Once minicom is open, you can start sending AT commands to the 4G module. For example:

```
AT
```

This command checks if the module is responsive. You should receive an **"OK"** response if the module is working properly.

2. To dial a phone number using the 4G module, you can use the ATD command followed by the phone number:

```
ATD<phone_number>;
```

Replace **<phone_number>** with the desired phone number you want to dial.

Make sure to include a semicolon ; at the end of the command to indicate the end of the phone number.

3.9 4G Cellular over M.2 B-KEY

Create a new power_4g.sh file:

```
sudo nano power_4g.sh
```

Open with sudo nano and enter the following command, then press ctrl+x to save and exit.

```
# SIM_MUX_SEL
echo 643 > export
echo out > gpio643/direction
echo 0 > gpio643/value
```

Execute the file:

```
sudo ./power_4g.sh
```

After 10-15 seconds (it takes a while for the module to power on and enumerate USB), check whether the device node appears:

```
ls /dev/ttyUSB*
```

Output /dev/ttyUSB0.

Confirm the actual action of GPIO:

```
cat /sys/class/gpio/gpio645/value # should be 0
cat /sys/class/gpio/gpio639/value # should be 0
```

Both values are 0 → the script has been correctly pulled down and the module is in working state.

Enter minicom to send commands:

```
sudo minicom -D /dev/ttyUSB2 -b 115200
```

Press **Ctrl+A, Z, E** in sequence. First send AT to test whether it is connected. If OK appears, the connection is successful.

After executing the following command, the module will automatically restart. If you do not exit minicom, you can see the corresponding configuration information.

ECM Dial-up Internet Access:

```
AT+QCFG="usbnet",1
```

Until the last line shows OK, it will be successful.

Note

The device needs to wait for a while, and then you can view the ip address of usb0 in ifconfig.

Test network status and communication :

```
# Check network status
ifconfig

# Test communication
ping www.baidu.com -I usb0
```


3.10 Zigbee over Mini-PCle

To test Zigbee communication with two Zigbee modules, follow these steps:

1. Check Serial Ports:

Use the following command to check available serial ports:

```
cat /dev/ttyUSB*
```

2. Install Serial Communication Tool:

```
sudo apt-get install cutecom
```

3. Open Serial Port for Coordinator (First Zigbee Module):

Open the **cutecom** tool and configure it for the first serial port:

Baud rate: **115200**

Check the "**Hex output**" option at the bottom of the interface.

Follow these steps to configure the first Zigbee module:

Set as coordinator: Send command '**55 04 00 05 00 05**', expect response '**55 04 00 05 00 05**'.

Reset device: Press reset button or send command '**55 07 00 04 00 FF FF 00 04**'.

Network formation: Send command '**55 03 00 02 02**'.

4. Open Serial Port for Router (Second Zigbee Module):

Open another instance of **cutecom** and configure it for the second serial port with the same settings as before.

Follow these steps to configure the second Zigbee module:

Set as router: Send command '**55 04 00 05 01 04**', expect response '**55 04 00 05 00 05**'.

Reset device: Press reset button or send command '**55 07 00 04 00 FF FF 00 04**'.

Network formation: Send command '**55 03 00 02 02**'.

5. Check Device Status:

Send command '**55 03 00 00 00**' to check the device status. Expect a response similar to '**55 2a 00 00 00 01 XX XX XX XX**', where 'XX' represents device information.

6. Enter Transparent Mode:

If network formation is successful, enter transparent mode by sending command **55 07 00 11 00 03 00 01 13**. Both modules should be in transparent mode for direct communication. To exit transparent mode, send **"+++"**.

7. Additional Notes:

- If router configuration fails, the device may already be a coordinator. Leave the network using command '**55 07 00 04 02 xx xx xx**'.
- Test transmission power using commands '55 04 0D 00 00 0D' (query) and '55 04 0D 01 XX XX' (set).

Ensure you replace **/dev/ttyUSB*** with the correct serial port for each Zigbee module. Follow these steps carefully to test Zigbee communication between the two modules successfully.

3.11 RS485 Testing

reComputer Industrial R2000 includes 3x RS485 ports, and the corresponding COM ports and device files are as follows:

Number of RS485 Ports	COM Port	Corresponding Silk Screen	Corresponding Device File
RS485-2	COM2	A2/B2/GND2	/dev/ttyACM1
RS485-3	COM3	A3/B3/GND3	/dev/ttyACM2
RS485-4	COM4	A4/B4/GND4	/dev/ttyACM3

To test the RS485 function, you can follow the steps below (take RS485_1 and RS485_2 as examples):

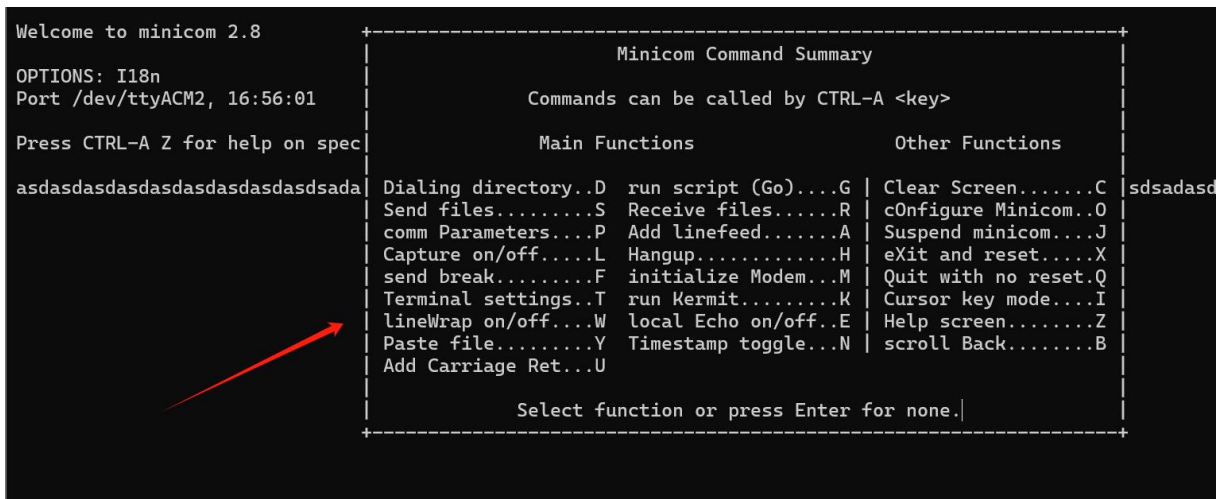
1. Please connect RS485_1 and RS485_2's A and B.
2. Open minicom in two terminal windows respectively:

```
sudo minicom -D /dev/ttyACM1
sudo minicom -D /dev/ttyACM2
```

Note

If there is an expansion board, the number needs to be moved back one place, for example `/dev/ttyAcM2` , `/dev/ttyAcM3`

3. The following operations need to be performed on both opened ACMs:
 - a. Press ctrl+a, then press Z, and the Minicom Command Summary interface will appear:



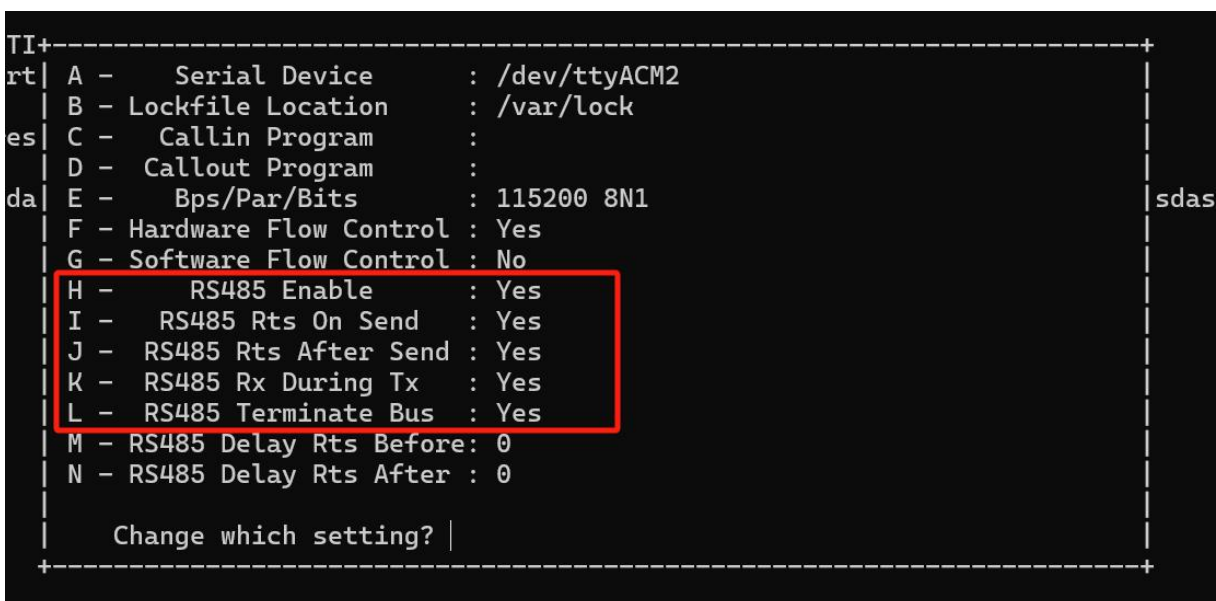
```
Welcome to minicom 2.8
OPTIONS: I18n
Port /dev/ttyACM2, 16:56:01
Press CTRL-A Z for help on spec
asdasdasdasdasdasdasdasdsada
```

Minicom Command Summary		
Commands can be called by CTRL-A <key>		
Main Functions	Other Functions	
Dialing directory..D	run script (Go)....G	Clear Screen.....C
Send files.....S	Receive files.....R	cOnfigure Minicom..O
comm Parameters...P	Add linefeed.....A	Suspend minicom...J
Capture on/off....L	Hangup.....H	eXit and reset....X
send break.....F	initialize Modem...M	Quit with no reset.Q
Terminal settings..T	run Kermit.....K	Cursor key mode....I
lineWrap on/off...W	local Echo on/off..E	Help screen.....Z
Paste file.....Y	Timestamp toggle...N	scroll Back.....B
Add Carriage Ret...U		

Select function or press Enter for none.

- b. Press O again to open configuration, select Serial port setup, and press Enter;

Open all RS485 related interfaces, press H/I/J/K/L in sequence to open;



```
TI+
rt| A - Serial Device      : /dev/ttyACM2
  | B - Lockfile Location  : /var/lock
es| C - Callin Program     :
  | D - Callout Program   :
da| E - Bps/Par/Bits       : 115200 8N1
  | F - Hardware Flow Control : Yes
  | G - Software Flow Control : No
  | H - RS485 Enable       : Yes
  | I - RS485 Rts On Send  : Yes
  | J - RS485 Rts After Send : Yes
  | K - RS485 Rx During Tx  : Yes
  | L - RS485 Terminate Bus : Yes
  | M - RS485 Delay Rts Before: 0
  | N - RS485 Delay Rts After : 0
  |
  | Change which setting? |
```

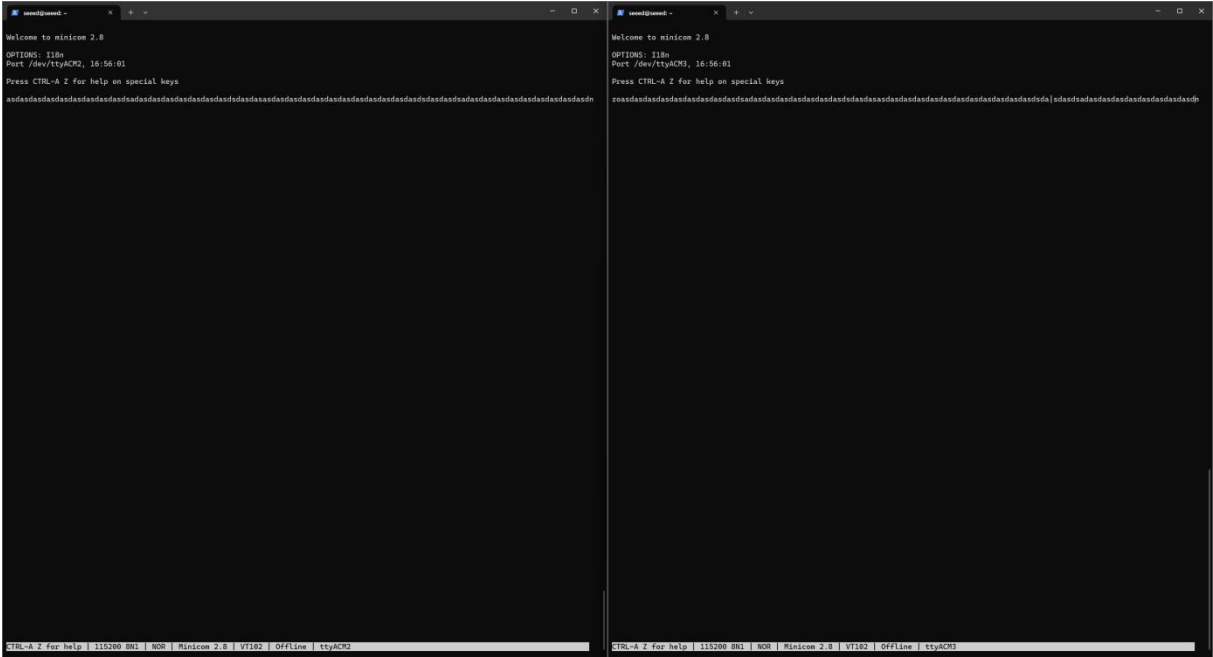
- c. After all "YES" are displayed, press Enter to return, and then select Exit to exit.

Note

Take ACM2 and ACM3 as an example:

If you want to send from ACM2 to ACM3, ACM2 needs to be set up again: **ctrl+A**, then press **Z** and then **E** , and then start the serial port write command. At this time, you can print strings in ACM2 at will, and you can see the contents of ACM2 in ACM3 at the same time;

Conversely, if you want to send from ACM3 to ACM2, ACM3 needs to be set up again: **ctrl+A**, then press **Z** and then **E** , and then start the serial port write command. At this time, you can print strings in ACM3 at will, and you can see the contents of ACM3 in ACM2 at the same time. As shown in the figure:

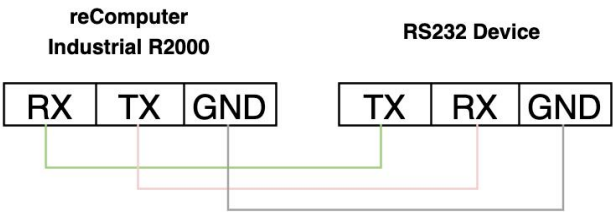


3.12 RS232 Testing

reComputer Industrial R2000 includes 1x RS232 ports, and the corresponding COM ports and device files are as follows:

Number of RS485 Ports	COM Port	Corresponding Silk Screen	Corresponding Device File
RS232-1	COM1	RX1/TX1/GND1	/dev/ttyACM0

Because RS232 is full-duplex communication, short-circuit the TX and RX of RS232 directly to perform a loopback test.



You need to open two terminals, ACM1 if the expansion board is connected, and ACM2 if the expansion board is not connected:

Terminal 1:

```
sudo minicom -D /dev/ttyACM1 -b 9600
```

If the expansion board is not connected, you need to change `/dev/ttyACM1` to `/dev/ttyACM0`.

Terminal 2:

```
printf "hello seeed\r\n" > /dev/ttyACM1
```

Terminal 1 will display the content requested by Terminal 2 to be printed.

3.13 DI Testing

reComputer Industrial R2000 contains 8x DI ports, user can configure these ports according to actual needs.

Number of ports	DI ports	Corresponding extended GPIO
8	DI1	GPIO588
	DI2	GPIO589
	DI3	GPIO590
	DI4	GPIO595
	DI5	GPIO573
	DI6	GPIO574
	DI7	GPIO575
	DI8	GPIO576

The input type of the DI ports is PNP. It supports input voltage is 5VDC~24VDC,current - 1000mA.

To test the functionality of DI, you can follow these steps to test it:

- 1、 The connection between the DI port of reComputer Industrial R2000 and the external load has been completed.
- 2、 Enter the following command to get the status of GPIO :

```
echo 588 > /sys/class/gpio/export
echo in > /sys/class/gpio/gpio588/direction
cat /sys/class/gpio/gpio588/value
```

3. 3、 When the external level is high, the value of `/sys/class/gpio/gpio588/value` is 0; when the external level is low, `/sys/class/gpio/gpio588/value` is 1.

3.14 DO Testing

reComputer Industrial R2000 contains 8x DO ports, user can configure these ports according to actual needs.

Number of ports	DI ports	Corresponding extended GPIO
8	DO1	GPIO638
	DO2	GPIO637
	DO3	GPIO636
	DO4	GPIO635
	DO5	GPIO577
	DO6	GPIO578
	DO7	GPIO594
	DO8	GPIO596

The output type of the DO ports is transistor. It supports output voltage - under 60 VDC, current capacity - 500 mA.

To test the functionality of DO, you can follow these steps to test it:

1. The connection between the DO port of reComputer Industrial R2000 and the external load has been completed.
2. Enter the following command to set the output to high level or low level :

```
echo 638 > /sys/class/gpio/export
echo out > /sys/class/gpio/gpio638/direction
echo 1 > /sys/class/gpio/gpio638/value
echo 0 > /sys/class/gpio/gpio638/value
```

3. When the external level is high, the value of `/sys/class/gpio/gpio638/value` is 0; when the external level is low, `/sys/class/gpio/gpio638/value` is 1.

3.15 USB Hub Testing

To test the USB hub, you can use the following steps:

1. Check if the USB hub is detected by running the **lsusb** command. This command lists all connected USB devices, including hubs.

```
lsusb
```

Running this command should display information about the USB devices connected to your system, including any USB hubs that are present.

If the USB hub is functioning properly, you should see its details listed in the output of the **lsusb** command. If it's not listed, there may be an issue with the hub or its connection to the system. In such cases, you may need to troubleshoot the USB hub or its connections.

3.16 RTC

To test the Real-Time Clock (RTC) functionality, follow these steps:

1. Disable automatic time synchronization:

```
sudo systemctl stop systemd-timesyncd  
sudo systemctl disable systemd-timesyncd
```

2. Set the time :

```
sudo hwclock --set --date "2025-7-17 12:00:00"
```

3. Synchronize the RTC time to the system:

```
sudo hwclock --hctosys
```

4. Check the RTC time:

```
sudo hwclock -r
```

This command will read and display the time stored in the RTC.

5. Disconnect the power source from the RTC, wait a few minutes, then reconnect it and check the RTC time again to see if it retained the correct time.

```
seeed@seeed:~$ sudo hwclock --set --date "2025-7-17 12:00:00"  
seeed@seeed:~$ sudo hwclock --hctosys  
^[[Aseeed@seeed:~$ shwclock -r  
2025-07-17 12:00:14.068559+01:00  
seeed@seeed:~$ |
```

3.17 Watchdog

To perform a watchdog test, follow these steps:

1. Install the watchdog software:

```
sudo apt install watchdog
```

2. Edit the watchdog configuration file:

```
# make sure you install vim already, if haven't, can install by the command below  
sudo apt-get install vim
```

```
sudo vim /etc/watchdog.conf
```

Modify the configuration as follows:

```
watchdog-device = /dev/watchdog

# Uncomment and edit this line for hardware timeout values that differ
# from the default of one minute.

watchdog-timeout = 120

# If your watchdog trips by itself when the first timeout interval
# elapses then try uncommenting the line below and changing the
# value to 'yes'.
#watchdog-refresh-use-settimeout = auto

# If you have a buggy watchdog device (e.g. some IPMI implementations)
# try uncommenting this line and setting it to 'yes'.
#watchdog-refresh-ignore-errors = no

# ===== Other system settings =====
#
# Interval between tests. Should be a couple of seconds shorter than
# the hardware time-out value.

interval = 15

max-load-1 = 24

#max-load-5 = 18

#max-load-15 = 12

realtime = yes

priority = 1
```

You can adjust other settings as needed.

3. Ensure the watchdog service is running:

```
sudo systemctl start watchdog
```

4. To test the watchdog functionality, execute the following command to simulate a system hang:

```
sudo su
echo 1 > /proc/sys/kernel/sysrq
echo "c" > /proc/sysrq-trigger
```

This command triggers a kernel crash and should cause the watchdog to reboot the system.

5. Monitor the system to confirm that it reboots after the specified timeout period.

These steps will help you test and ensure the functionality of the watchdog timer on your system.

```
seeed@seeed:~$ sudo nano /etc/watchdog.conf
seeed@seeed:~$ sudo systemctl start watchdog
seeed@seeed:~$ sudo sh -c "echo 1 > /proc/sys/kernel/sysrq"
sudo sh -c "echo 'c' > /proc/sysrq-trigger"
client_loop: send disconnect: Connection reset
PS C:\Users\seeed> |
```

3.18 Buzzer

The GPIO corresponding to the buzzer is gpio627. Enter the following script to turn the buzzer on/off :

1. Turn on the buzzer :

```
echo 627 > /sys/class/gpio/export
echo out > /sys/class/gpio/gpio627/direction
echo 1 > /sys/class/gpio/gpio627/value
```

2. Turn off the buzzer :

```
echo 627 > /sys/class/gpio/export
echo out > /sys/class/gpio/gpio627/direction
echo 0 > /sys/class/gpio/gpio627/value
```

3.19 TPM 2.0

If you connect TPM 2.0 module to device, the following code can help check TPM connection.

```
ls /dev | grep tpm
```

If you see **tpm0** and **tpmrm0** in the output, it means that TPM (Trusted Platform Module) devices are detected and available on your system. This indicates that the TPM hardware is recognized and accessible, which is a good sign. You can proceed with using TPM-related functionalities or applications knowing that the devices are present and accessible.

3.20 ATECC608A

To interact with the ATECC608A device and generate a random serial number, follow these steps:

1. Clone the atecc-util Repository:

```
curl -LJO https://github.com/wireboard/atecc-util/releases/download/v0.4.12/atecc-util_0.4.12_arm64.deb
```

2. Extract the contents of the .deb package to the current directory:

```
dpkg -x ./atecc-util_0.4.12_arm64.deb .
```

3. Navigate to the atecc Directory:

```
cd usr/bin
```

4. Generate a Random Serial Number:

```
./atecc -b 10 -s 192 -c 'serial'
```

-b 1 → Uses slot 1.

-s 192 → Sets the serial number size to **192 bits**.

-c 'serial' → Generates a random serial number.

This command instructs the ATECC utility to use slot 1 (-b 1), set the serial number size to 192 bits (**-s 192**), and generate a random serial number (**c 'serial'**). The output will be the generated serial number, such as **"01235595d3d621f0ee"**.

This process allows you to interact with the ATECC608A device and perform various operations, such as generating random serial numbers.

3.21 EEPROM

Here are the commands to interact with an EEPROM (Electrically Erasable Programmable Read-Only Memory):

1. Grant full permissions (read, write, and execute) to the EEPROM device file:

```
sudo chmod 777 /sys/bus/i2c/devices/10-0050/eeprom
```

2. Write the string "This is a test string" to the EEPROM device:

```
echo "This is a test string" > /sys/bus/i2c/devices/10-0050/eeprom
```

3. Read the contents of the EEPROM device and displays it in hexadecimal format using the **hexdump** utility:

```
sudo cat /sys/bus/i2c/devices/10-0050/eeprom | hexdump -C
```

3.22 SSD

To list the disks, including the SSD, you can use the **fdisk -l** command. Here's how:

```
sudo fdisk -l
```

This command will display a list of all disks connected to your system, including the SSD if it's properly detected. Look for entries that represent your SSD. They typically start with **/dev/sd** followed by a letter (e.g., **/dev/sda**, **/dev/sdb**, etc.).

Once you identify the entry corresponding to your SSD, you can proceed with partitioning or formatting it as needed.

3.23 UPS for Safe Shut Down

A GPIO6 between CPU and DC power in is used to alarm CPU when the power supply is down. Then the CPU should do something urgent in a script before energy exhaustion of super capacitor and run a "\$ shutdown".

Another way to use this function is Initiate a shutdown when GPIO pin changes. The given GPIO pin is configured as an input key that generates KEY_POWER events. This event is handled by systemd-logind by initiating a shutdown.

1. Hardware connection.

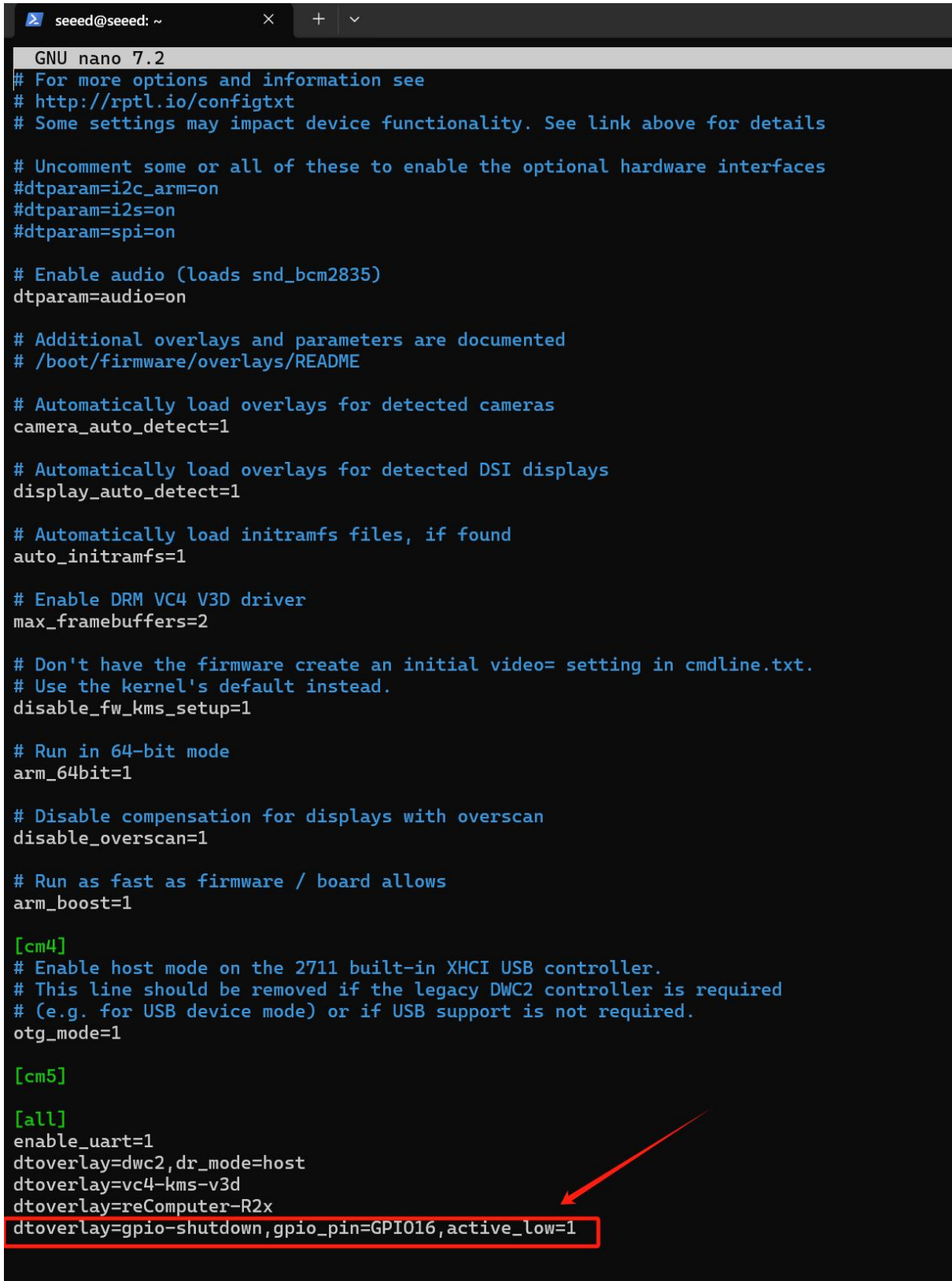
Please make sure that the `CM5_UPS_DET` pin of the UPS device is connected to the GPIO16 pin of the R2000 device.

2. Modify the configuration file.
 - a. Open the terminal.
 - b. Execute the following command to edit the configuration file:

```
sudo nano /boot/firmware/config.txt
```

3. Add the following content at the end of the file:

```
dtoverlay=gpio-shutdown,gpio_pin=GPIO16,active_low=1
```



```
GNU nano 7.2
# For more options and information see
# http://rptl.io/configtxt
# Some settings may impact device functionality. See link above for details

# Uncomment some or all of these to enable the optional hardware interfaces
#dtparam=i2c_arm=on
#dtparam=i2s=on
#dtparam=spi=on

# Enable audio (loads snd_bcm2835)
dtparam=audio=on

# Additional overlays and parameters are documented
# /boot/firmware/overlays/README

# Automatically load overlays for detected cameras
camera_auto_detect=1

# Automatically load overlays for detected DSI displays
display_auto_detect=1

# Automatically load initramfs files, if found
auto_initramfs=1

# Enable DRM VC4 V3D driver
max_framebuffers=2

# Don't have the firmware create an initial video= setting in cmdline.txt.
# Use the kernel's default instead.
disable_fw_kms_setup=1

# Run in 64-bit mode
arm_64bit=1

# Disable compensation for displays with overscan
disable_overscan=1

# Run as fast as firmware / board allows
arm_boost=1

[cm4]
# Enable host mode on the 2711 built-in XHCI USB controller.
# This line should be removed if the legacy DWC2 controller is required
# (e.g. for USB device mode) or if USB support is not required.
otg_mode=1

[cm5]

[all]
enable_uart=1
dtoverlay=dwc2,dr_mode=host
dtoverlay=vc4-kms-v3d
dtoverlay=reComputer-R2x
dtoverlay=gpio-shutdown,gpio_pin=GPIO16,active_low=1
```

Save and exit the editor (press `Ctrl+O` to save, `Enter` to confirm, and `Ctrl+X` to exit).

4. Prepare Python script

- Create a new Python script file:

```
cd ~  
sudo nano ups_shutdown.py
```

- b. Copy and paste the following code into the file:

```
import RPi.GPIO as GPIO  
import time, os  
  
num = 0  
  
GPIO.setmode(GPIO.BCM)  
  
# Set GPIO16 to input mode  
# Add 500ms anti-shake time to stabilize the software  
GPIO.setup(16, GPIO.IN, pull_up_down=GPIO.PUD_UP)  
GPIO.add_event_detect(16, GPIO.FALLING, bouncetime=500)  
  
while True:  
    if GPIO.event_detected(16):  
        print("...External power off...")  
        print("")  
  
        # Sync data to disk  
        os.system('sync')  
        print("...Data saving...")  
        print("")  
  
        # Sleep for 3 seconds  
        time.sleep(3)  
  
        # Synchronize data again  
        os.system('sync')  
  
        # Countdown 5 seconds  
        while num < 5:  
            print('-----')  
            s = 5 - num  
            print('---' + str(s) + '---')  
            num = num + 1
```

```
time.sleep(1)

print('-----')

# Execute shutdown command

os.system('sudo shutdown -h now')
```

Save and exit the editor (press **Ctrl+O** to save, **Enter** to confirm, and **Ctrl+X** to exit).

5. Run the script.
 - c. Open the terminal.
 - d. Execute the following command to run the script:

```
sudo python3 ups_shutdown.py
```

Note

Use `sudo` to ensure that the script has sufficient permissions to execute the shutdown command.

6. Simulate power failure test
 - a. Cut off the external power supply.
 - b. Observe whether the system automatically saves data and shuts down.

```
seeed@seeed:~$ sudo nano /boot/firmware/config.txt
seeed@seeed:~$ sudo nano ups_shutdown.py
seeed@seeed:~$ sudo python3 /home/seeed/ups_shutdown.py
...External power off...

...Data saving...

-----
---5---
-----
-----
---4---
-----
-----
---3---
-----
-----
---2---
-----
-----
---1---
-----

Broadcast message from root@seeed on pts/2 (Thu 2025-07-17 23:28:48 BST):
The system will power off now!

Broadcast message from root@seeed on pts/2 (Thu 2025-07-17 23:28:48 BST):
The system will power off now!

Connection to 10.0.0.116 closed by remote host.
Connection to 10.0.0.116 closed.
PS C:\Users\seeed> |
```

7. Verify the result
 - a. Reconnect the power supply.
 - b. Check whether the system data is complete and starts normally.

1. For UPS function please contact us for more information.
2. The alarm signal is active LOW.

The M.2 M-KEY 2280 slot on the reComputer Industrial R2000 is designed to accommodate PCIe M.2 AI Accelerator. And the R2000-12 series has been pre-installed with a Hailo-8 M.2 AI Acceleration up to 26TOPS.

1. Navigate to the test case directory

2. Start the virtual environment

3. Run the simple detection example

This is lightweight version of the detection example, mainly focusing on demonstrating Hailo performance while minimizing CPU load. The internal GStreamer video processing pipeline is simplified by minimizing video processing tasks, and the YOLOv6 Nano model is used.

If the reComputer you purchased does not include Hailo-8 and you are considering purchasing a Hailo device for integration, please refer to the official Hailo documentation (<https://github.com/hailo-ai>) to configure the firmware and environment, and run the examples to verify that the device can be used normally.

3.25 Installing Ubuntu on reComputer Industrial R2000

1. Burn Image
 - Follow the instructions provided on the <https://ubuntu.com/download/raspberry-pi> to burn the downloaded image.
2. Initial Setup:
 - Connect a monitor to your Raspberry Pi.
 - Power on the Raspberry Pi.
3. Initial Configuration:
 - Follow the onscreen instructions for the initial setup. This may include setting up the user account, language, and other preferences.
4. Install Sseed provided firmware:
 - Once logged into the system, open a terminal window.
 - Clone the Sseed-Studio GitHub repository by running the following command:

```
git clone --depth 1 https://github.com/Sseed-Studio/seeed-linux-dtoverlays.git
```

Navigate to the cloned repository directory:

```
cd seeed-linux-dtoverlays
```

Run the provided script to configure the display device. For example, to configure for reComputer-R2x, run:

```
sudo ./scripts/reTerminal.sh --device reComputer-R2x
```

5. Reboot:

After running the script, reboot your reComputer Industrial R2000 by running:

```
sudo reboot
```

3.26 Customized Linux: Yocto and Mender

The reComputer Industrial R2000 is an edge IoT controller device that utilizes the Raspberry Pi CM5 as its processor. When you require a customized Linux distribution to run on your device, as well as a convenient solution for managing software updates, Yocto Project and Mender come into play.

Yocto Project is a powerful tool tailored for creating custom Linux distributions specifically designed for embedded devices, ensuring that your device's requirements are met. On the other hand, Mender serves as an open-source over-the-air (OTA) software update manager for embedded Linux devices, simplifying the process of managing software updates. It enables remote management of software updates via the internet, eliminating the need for physical access to the device and ensuring its security and stability.

Therefore, we provide the official reComputer R110X script along with Yocto Project-generated images, as well as scripts for Mender updates, for your reference and convenience.

Note

More details and tutorial please refer to https://wiki.seeedstudio.com/recomputer_industrial_r2000_yocto_mender/

3.27 Costomized Linux: Buildroot

Buildroot is a powerful tool for developers who want to create a customized and lightweight Linux environment tailored to their embedded device's hardware and software requirements. Buildroot is another tool similar to Yocto Project that can be used to create custom Linux distributions for reComputer Industrial R2000. It is less resource-intensive than Yocto Project, making it a suitable choice for devices with limited processing power or memory. It is recommended to consult the Buildroot documentation and community resources to determine its compatibility with the device.

Note

More details and tutorial please refer to https://wiki.seeedstudio.com/recomputer_Industrial_R2000_buildroot/

- To download and compile the Sseed Studio Linux Buildroot code, you may need to install the following libraries if they are not already installed:

```
sudo -E apt-get install sed make binutils build-essential gcc g++ bash patch gzip bzip2 perl tar cpio unzip rsync  
file bc wget python cvs git  
  
mercurial subversion
```

- clone the code from github

```
git clone --depth 1 https://github.com/Seeed-Studio/seeed-linux-buildroot.git
```

- For 32-bit or 64-bit systems, configure and compile using predefined configuration files:

```
cd seeed-linux-buildroot  
  
# For 64-bit:  
make reComputer_R2x_64_defconfig  
  
# For 32-bit:  
make reComputer_R2x_32_defconfig  
  
make
```

The final firmware files will be located in the following path: ***seeed-linux-buildroot/output/images/***.

C4. Assembly Guide

4.1 Disassembly Guide

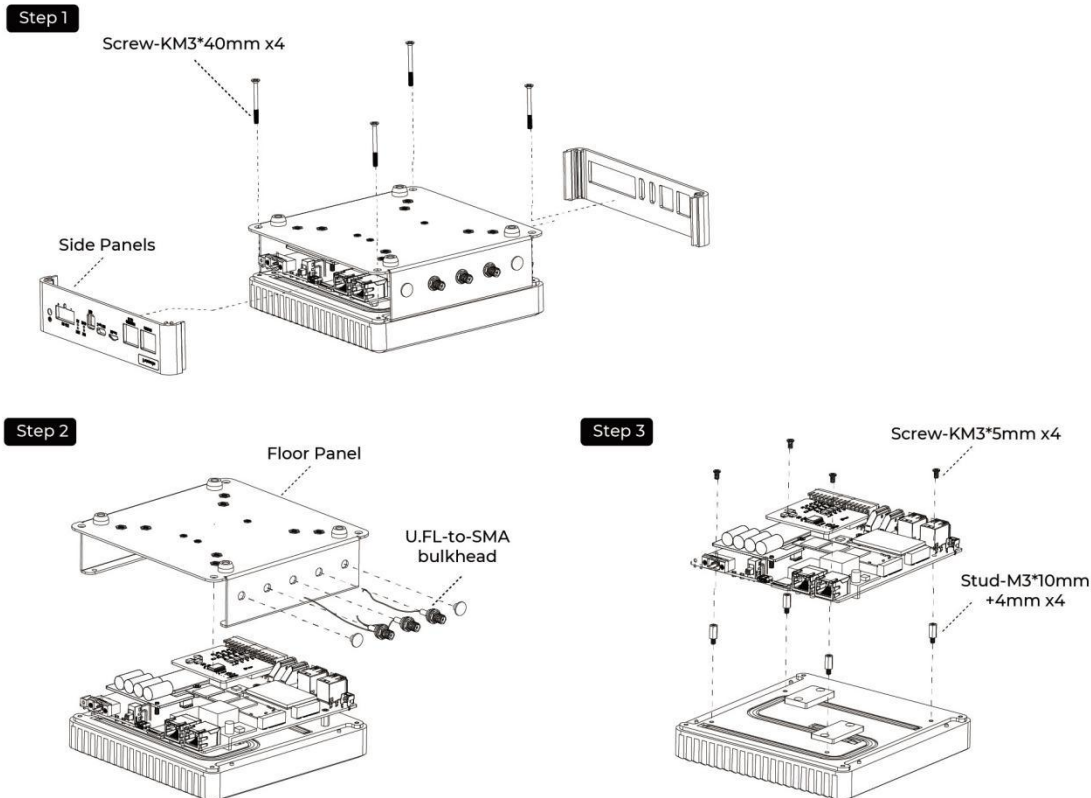
Following these steps should help you disassemble the device without any issues.

Step 1: Remove the Four Screws at the Bottom and remove the front and rear panels:

- Locate and unscrew the four screws located at the bottom of the device using an appropriate screwdriver.
- Once the screws are removed, carefully lift off the front and rear panels from the device

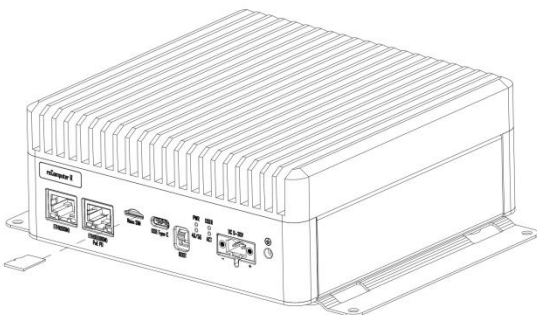
Step 2: Unscrew the side antenna connector nuts and remove the Floor Panel.

Step 3: Remove the four screws that secure the PCB in place



4.2 Assemble Nano SIM Card

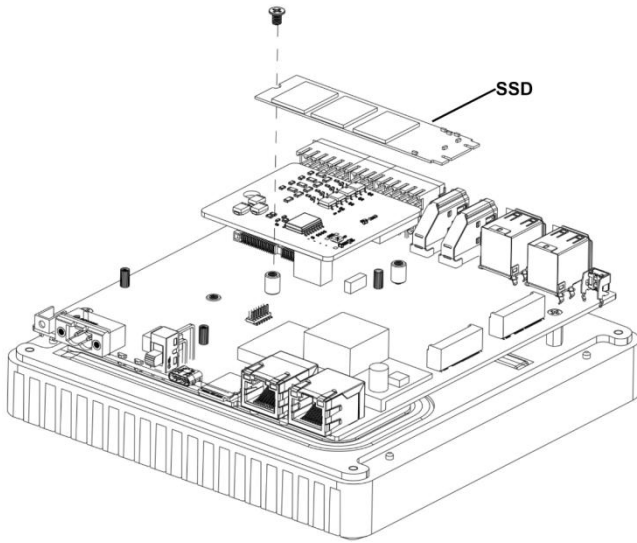
Step 1: Load the Nano SIM Card into the SIM slot.



4.3 Assemble SSD

Step 1: Remove the back cover following the disassembly guide.

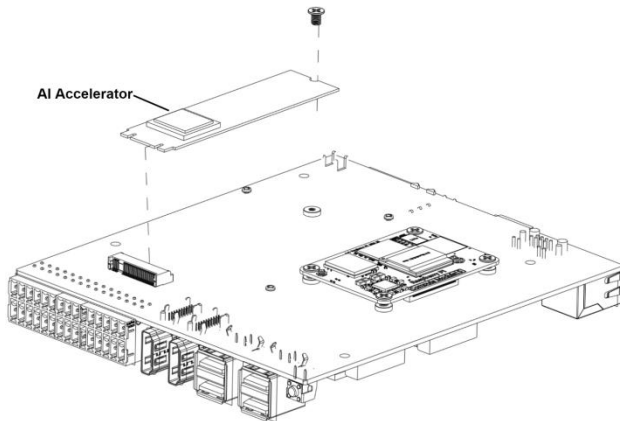
Step 2: Load the SSD into the M.2 socket and lock the screws.



4.4 Assemble M.2 AI Accelerator

Step 1: Disassemble the entire device following section 4.1 "Disassembly Guide".

Step 2: Load the AI Accelerator into the M.2 socket and lock the screws.



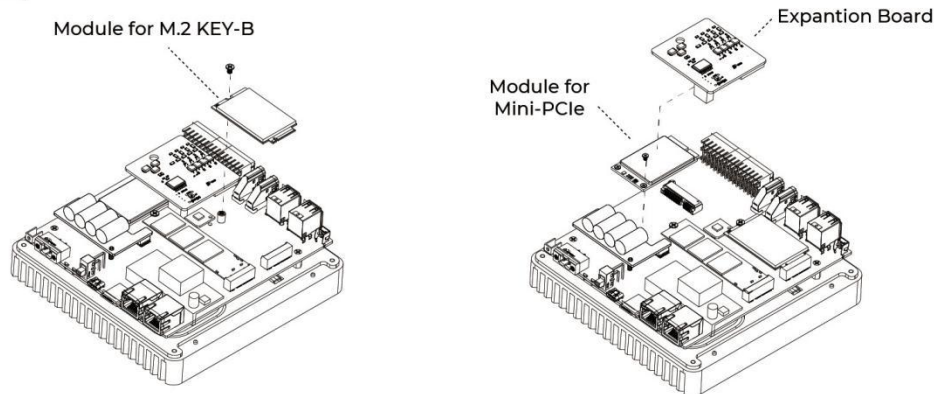
4.5 Assemble 5G/4G/LoRa®/Zigbee Module and Antenna

Step 1: Remove the Expansion Board and load the 4G module/LoRa® Module/Zigbee Module into the Mini-PCle/M.2 B-KEY slot and lock the screws.

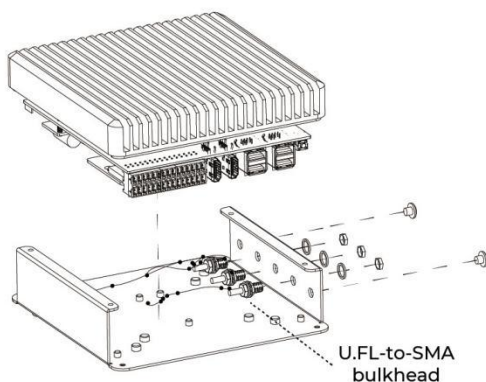
Step 2: Connect the feeder to the antenna hole on the housing as shown in the following diagram.

Step 3: Install the feeder into the antenna base of the corresponding module

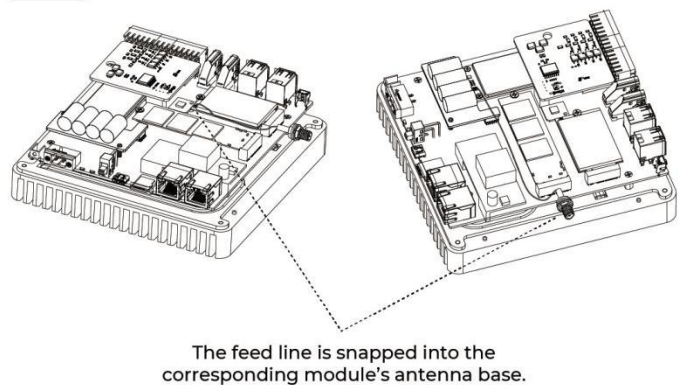
Step 1



Step 2



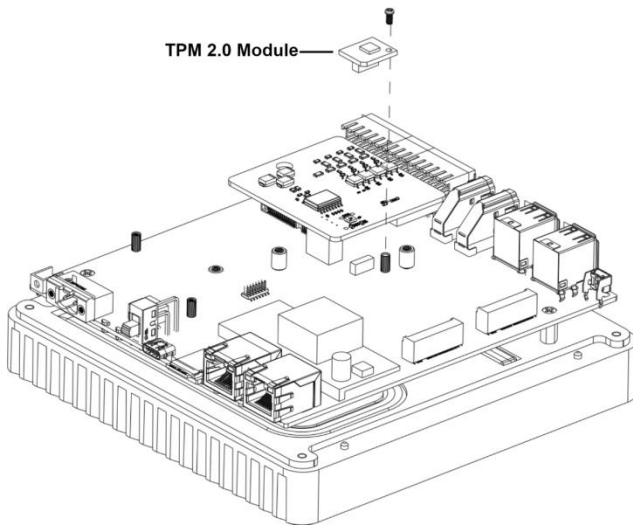
Step 3



4.6 Assemble TPM 2.0 Module

Step 1: Remove the back cover following the disassembly guide.

Step 2: Load the TPM 2.0 module into the J26 socket.



4.7 Assemble UPS module

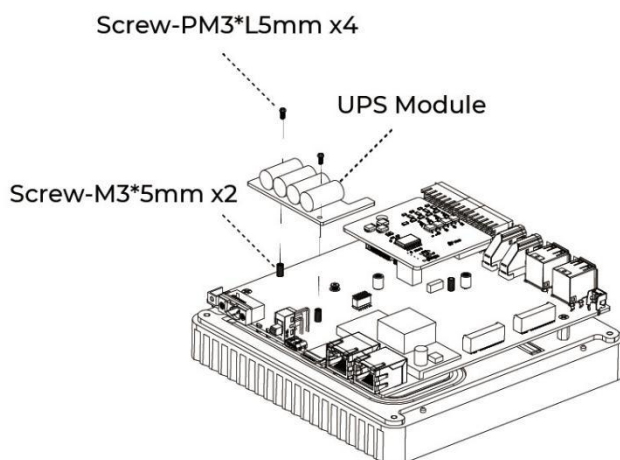
Step 1: Before installing the UPS and PoE module on the CM5 module side of board, disassemble the entire device following the disassembly guide provided.

Step 2:

- Using two PM2.0xL5.0 screws and M2.0x5.0 standoffs, secure the UPS module onto two holes without metal contact pads.
- Make sure the UPS module is aligned properly and firmly attached using the provided screws and standoffs.

Step 3: Install the PoE Module

- Align the PoE module with the designated aperture on the board.
- Carefully solder the PoE module onto the board. Due to the compact nature of the board, exercise caution while soldering to avoid damaging nearby components.



4.8 Mounting Guide

4.8.1 DIN-rail Mounting Guide

reComputer Industrial R2000 offers various installation methods. The DIN-rail clip and installation screws are included in the packaging. Follow the diagram to correctly attach the DIN-rail clip to the mounting holes on the side of the device. Once the screws are securely fastened, you can then install the device onto the mounting rail.

Step 1: Place the device and rail clip on the upper edge of the standard profile rail at the position shown and push the device down.

Step 2: Swing the rail clip of the device from below through the standard profile rail.

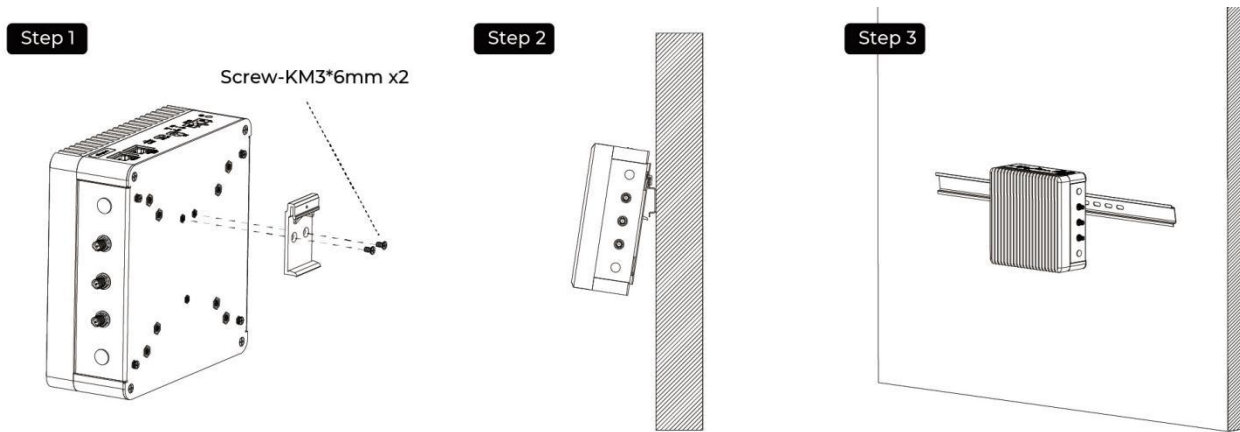
Step 3: Push the device in the direction of the standard profile rail. You will hear the device click into place.

Removing

Step 1: Push down the device until it is released by the rail clip.

Step 2: Swing the device out of the standard profile rail.

Step 3: Lift the device up and of



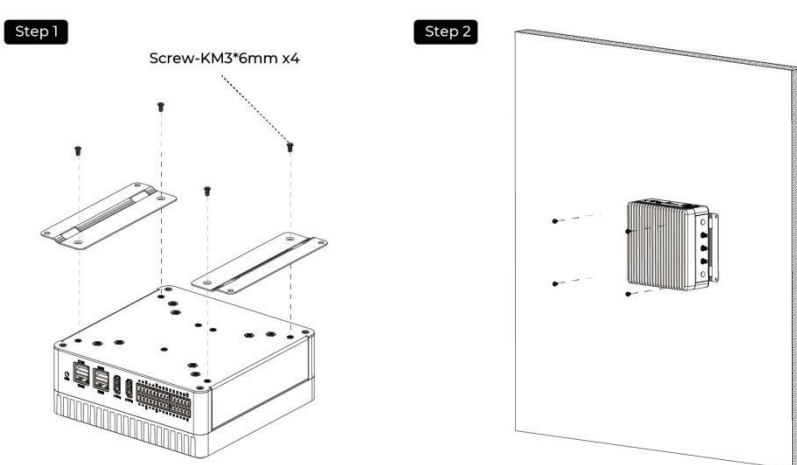
4.8.2 Wall Mounting Guide

Vertical mounting method is also suitable for reComputer Industrial R2000, however the mounting brackets are not included in box, that need additional purchase.

Step 1: Lay the mounting brackets on the rear of the device.

Step 2: Fasten the brackets with supplied screws.

Step 3: Mark the bore holes, drill the required holes in the wall and fasten the device to the wall using two screws.



C5. Accessories List

Item	Product	Product Name	SKU
LoRa® module		Region optional LoRaWAN Gateway Module(SPI)-US915	114992969
		Region optional LoRaWAN Gateway Module(SPI)-EU868	114993268
		Region optional LoRaWAN Gateway Module(USB)-US915	114992991
		Region optional LoRaWAN Gateway Module(USB)-EU868	114992628
LoRa® Antenna		LoRa Antenna Kit - 868-915 MHz	110061501
Wi-Fi/BLE Antenna		Raspberry Pi Compute Module 4 Antenna Kit	114992364
4G Module		LTE Cat 4 EC25-AFXGA-Mini-PCle Module - for North American	113991134
		LTE Cat 4 EC25-EUXGR-Mini-PCle Module - for EMEA and Thai	113991135
		LTE Cat 4 EC25-AUXGR-Mini-PCle Module - for Australia	113991174
		LTE Cat 4 EC25-EFA-Mini-PCle Module - for Thai	113991214
		LTE Cat 4 EC25-EMGA-Mini-PCle Module - for Malaysia	113991234
		LTE Cat 4 EC25-JFA-mini-PCle	113991296
4G Antenna		4G Antenna Kit for 4G module	110061502
GPS Antenna		GPS Antenna Kit for EC25 4G Module	110061521
UPS Supercapacitor		SuperCAP UPS LTC3350 Module	110992004
Encryption chip TPM 2.0		TPM 2.0 Module with infineon SLB9670	114993114
SSD Card		NVMe M.2 2280 SSD 2TB	114993467
		NVMe M.2 2280 SSD 1TB	112990267
		512GB NVMe M.2 PCIe Gen3x4 2280 Internal SSD	112990247
		256GB NVMe M.2 PCIe Gen3x4 2280 Internal SSD	112990246
		128GB NVMe M.2 PCIe Gen3x4 2280 Internal SSD	112990226

C6. Warranty & Support

6.1 Warranty

1. From the date of sale, the company provides 24 months free warranty for the products.
2. Warranty coverage is limited to products purchased from the official Seeed Studio website or authorized distributors. Customers need to keep receipts and purchase vouchers.
3. The products to be repaired shall be properly packaged and transported, and the customer shall be responsible for any loss or damage during transportation.
4. During the warranty period, the freight and maintenance costs arising from product quality failures shall be borne by Seeed Studio. If the warranty period exceeds 24 months, Seeed Studio will charge the fee for replacing parts according to the product failure, and the freight is borne by the user.
5. During the free warranty period, in case of any of the following events, Seeed Studio has the right to refuse service or charge materials and service fees at its discretion.

Product failure or damage caused by improper use by users.

The product label is damaged and the product information cannot be identified.

Even within the warranty period, if the product has functional issues or is difficult to repair due to improper customer use, unauthorized disassembly or modification, poor operating environment, improper maintenance, accidents, or other reasons. Seeed Studio reserves the right to make judgments on the above situations and collect maintenance fees.

Other unavoidable external factors cause product failure and damage.

The above warranty regulations are only applicable to the above Seeed Studio reComputer Industrial R2000 series, other products are not applicable!

6.2 Support

Quick start guide:

https://wiki.seeedstudio.com/recomputer_r/

Tech support email:

If you encounter any issues while deploying or testing, please don't hesitate to contact our technical support team at techsupport@seeed.io, or refer to our online knowledge base, <https://wiki.seeedstudio.com>.

Customized service email:

For further information about customizations, welcome you to directly reach out at edge@seeed.cc, we will provide prompt reply.

Discord:

Discord community:

Welcome to join our official community, where you can exchange product-related questions and get relevant support.

<https://discord.seeed.cc>

