

TM-CP2B Quick Start Guide



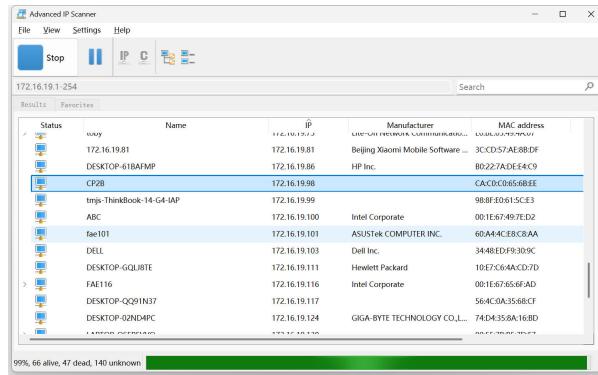
Login steps

- Connect the power and plug in the network cable
 - The power supply range is 9V-36V, with the negative terminal near the LED end.
Note: Incorrect polarity can cause damage to the machine.

- The default network connection method of the machine is DHCP to automatically obtain an IP address.
After the network is connected normally, the two indicator lights on the network port will light up simultaneously.

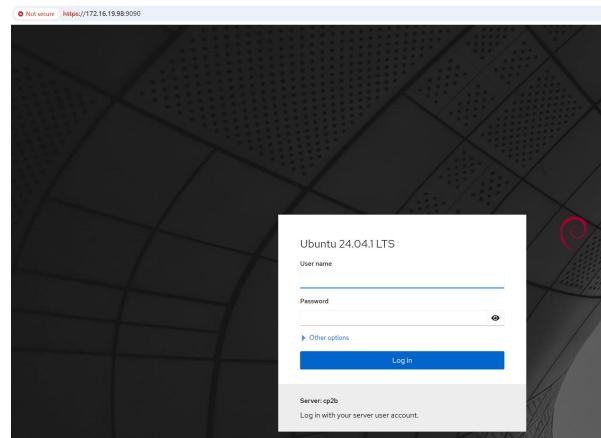
- Obtain the IP address of the machine
 - Use LAN IP address scanning software [Advanced_IP_Scanner](#) to obtain all scanned LAN IP addresses.
 - After downloading and installing the software, open the software and you will see the following interface. Click Scan to start scanning. The IP address corresponding to the "CP2B" device is the

actual IP address of the machine.

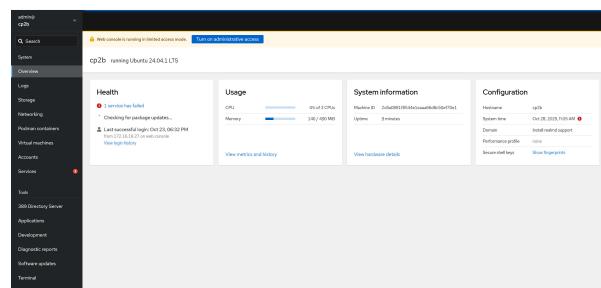


- The machine with the device name cp2b in the scanning results corresponds to the DHCP address of the machine.
- Login device
 - Enter the following link in the browser to enter the login interface:
https://your_ipaddress:9090/

Replace "your_ipaddress" with the actual IP address of the machine.



- Enter your username and password, then click login.
The default username for the machine is "admin", and the password is "admin".



- Click on the terminal to enter the shell interface, where you can update various devices of the system or operating system.

```
admin@cp2b: ~
[admin@cp2b ~]# sudo apt update
[sudo] password for admin:
Hit:1 http://ports.ubuntu.com/ubuntu-ports noble-updates InRelease [116 kB]
Get:2 http://ports.ubuntu.com/ubuntu-ports noble-updates InRelease [126 kB]
Get:3 http://ports.ubuntu.com/ubuntu-ports noble-updates InRelease [126 kB]
Get:4 http://ports.ubuntu.com/ubuntu-ports noble-security InRelease [126 kB]
Get:5 http://ports.ubuntu.com/ubuntu-ports noble-security InRelease [126 kB]
Get:6 http://ports.ubuntu.com/ubuntu-ports noble-updates/main Translation-en [664 kB]
Get:7 http://ports.ubuntu.com/ubuntu-ports noble-updates/main Translation-en [294 kB]
Get:8 http://ports.ubuntu.com/ubuntu-ports noble-updates/universe armtf Packages [1192 kB]
Get:9 http://ports.ubuntu.com/ubuntu-ports noble-updates/universe Translation-en [383 kB]
Get:10 http://ports.ubuntu.com/ubuntu-ports noble-updates/universe armtf Packages [1192 kB]
Get:11 http://ports.ubuntu.com/ubuntu-ports noble-updates/restricted armtf Packages [462 kB]
Get:12 http://ports.ubuntu.com/ubuntu-ports noble-updates/restricted Translation-en [868 kB]
Get:13 http://ports.ubuntu.com/ubuntu-ports noble-updates/multiverse armtf Packages [383 kB]
Get:14 http://ports.ubuntu.com/ubuntu-ports noble-updates/multiverse Translation-en [3840 kB]
Get:15 http://ports.ubuntu.com/ubuntu-ports noble-updates/multiverse armtf Packages [3840 kB]
Get:16 http://ports.ubuntu.com/ubuntu-ports noble-backports/main armtf Packages [484 kB]
Get:17 http://ports.ubuntu.com/ubuntu-ports noble-backports/main Translation-en [3280 kB]
Get:18 http://ports.ubuntu.com/ubuntu-ports noble-backports/main armtf Packages [484 kB]
Get:19 http://ports.ubuntu.com/ubuntu-ports noble-backports/universe armtf Packages [28.9 kB]
Get:20 http://ports.ubuntu.com/ubuntu-ports noble-backports/universe Translation-en [17.8 kB]
Get:21 http://ports.ubuntu.com/ubuntu-ports noble-backports/restricted armtf Packages [216 kB]
Get:22 http://ports.ubuntu.com/ubuntu-ports noble-backports/restricted Translation-en [216 kB]
Get:23 http://ports.ubuntu.com/ubuntu-ports noble-backports/restricted armtf Components [216 kB]
Get:24 http://ports.ubuntu.com/ubuntu-ports noble-backports/multiverse armtf Components [216 kB]
Get:25 http://ports.ubuntu.com/ubuntu-ports noble-security/main armtf Packages [211 kB]
Get:26 http://ports.ubuntu.com/ubuntu-ports noble-security/main Translation-en [211 kB]
Get:27 http://ports.ubuntu.com/ubuntu-ports noble-security/universe armtf Packages [172 kB]
Get:28 http://ports.ubuntu.com/ubuntu-ports noble-security/universe Translation-en [208 kB]
Get:29 http://ports.ubuntu.com/ubuntu-ports noble-security/universe armtf Components [52.3 kB]
```

- You can also log in to the machine using SSH or console interface. The machine does not have SSH service installed by default. Please refer to the following command for installation:

```
sudo apt install ssh
```

The machine integrates a USB to UART circuit internally, and uses a Type-C cable to connect the console interface to the computer. The computer will detect a serial device, and if it is used for the first time, a [USB driver](#) is required.

Interface operation

- Interface
 - The correspondence between ttySx device nodes and interfaces.

ttyS1 -- RS485-2(A2 B2)

ttyS2 -- RS485-1(A1 B1)

ttyS3 -- TTL(TX RX)

ttyS4 -- LORA

spidev0.0 -- LORA-WAN

ttyUSB0-ttyUSB3 -- 4G-LTE

- RS485&TTL

```
stty -F /dev/ttyS1 raw speed 115200 //Configure RS485 baud rate to
115200
echo "hello world" > /dev/ttyS1      //Send "hello world" to RS485 port
```

You can also operate the serial port through C or Python.

- 4G-LTE

After inserting the 4G module and SIM card, the machine will automatically complete the dialing operation after booting up.

After successful dialing, the system will display the following ppp0 network nodes.

The default 4G-LET module model currently used is EC20.

```

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

用户账户
服务
工具
软件更新
应用程序
终端

ppp0: flags=4305<UP,POINTOPOINT,RUNNING,NOARP,MULTICAST>  mtu 1500
      inet 10.30.99.86  netmask 255.255.255.255  destination 10.64.64.64
      PPP txqueuelen 3  (Point-to-Point Protocol)
      RX packets 4  bytes 52 (52.0 B)
      RX errors 0  dropped 0  overruns 0  frame 0
      TX packets 14  bytes 198 (198.0 B)
      TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0

usb0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
      ether aa:1f:47:b4:d3:0d  txqueuelen 1000  (Ethernet)
      RX packets 0  bytes 0 (0.0 B)
      RX errors 0  dropped 0  overruns 0  frame 0
      TX packets 0  bytes 0 (0.0 B)
      TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0

wlan0: flags=4099<UP,BROADCAST,MULTICAST>  mtu 1500
      txqueuelen 1000  (Ethernet)
      RX packets 0  bytes 0 (0.0 B)
      RX errors 0  dropped 0  overruns 0  frame 0
      TX packets 0  bytes 0 (0.0 B)
      TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0
  
```

- WIFI&BT

The default WIFI module model used by the machine is BL-M8800DU6-D80, which uses the AIC8800D80 chip. Support wifi 802.11a/b/g/n/ac/ax and bt5.4.

The system has already integrated drivers and firmware by default, and can be used by plugging in the module.



- LORA-WAN

- CP2B supports LORA WAN modules with SPI interfaces, such as the EBYTE E106 series, as shown in the following figure, which defaults to using the MINI-PCIE interface.

SX1302 LoRaWAN网关 SPI接口 433/470/868/915MHz

- PCI-e接口
- 高速率
- 多个路径通道
- 内置PA+LNA



- Test according to the following steps.

```

git clone https://github.com/coolpi-george/sx1302.git /*Clone
code to any path on the CP3B*/
cd sx1302
git checkout cp2b
make clean all
make -j8
cp tools/reset_lgw.sh util_chip_id/
cp tools/reset_lgw.sh packet_forwarder/
cp tools/reset_lgw.sh libloragw/
cd util_chip_id/
sudo ./chip_id                                         /*Obtain
module EUI*/
[sudo] password for admin:
CoreCell reset through GPIO55...
Opening SPI communication interface
Note: chip version is 0x10 (v1.0)
INFO: using legacy timestamp
ARB: dual demodulation disabled for all SF

INFO: concentrator EUI: 0x0016c001f11a1f85

Closing SPI communication interface
CoreCell reset through GPIO55...
cd libloragw/
sudo ./test_loragw_reg                                     /*Traverse
the registers of the module*/
CoreCell reset through GPIO55...
Opening SPI communication interface
Note: chip version is 0x10 (v1.0)
## TEST#1: read all registers and check default value for non-
read-only registers
-----
TEST#1 PASSED
-----

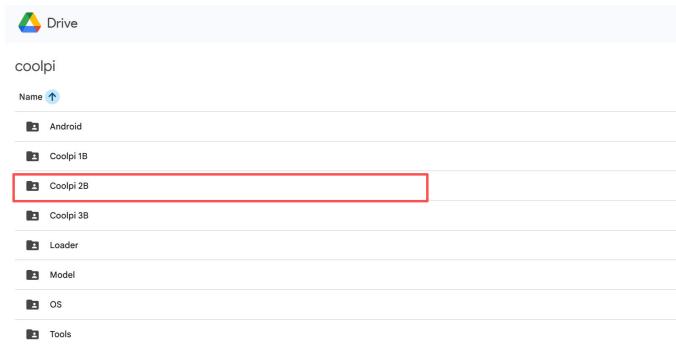
## TEST#2: read/write test on all non-read-only, non-pulse,
non-w0clr, non-w1clr registers
-----
TEST#2 PASSED                                         /*The
successful identification module is running normally*/
-----

Closing SPI communication interface
CoreCell reset through GPIO55...
```
- Configure as gateway and connect to TNN server according to
[Official Documents](https://semtech.my.salesforce.com/sfc/p/#E0000000Je1G/a/RQ0000043BUT/kDK2Unqnoazf9_UbC7um6mY7NnVzIWECoCudd3xuUnU).

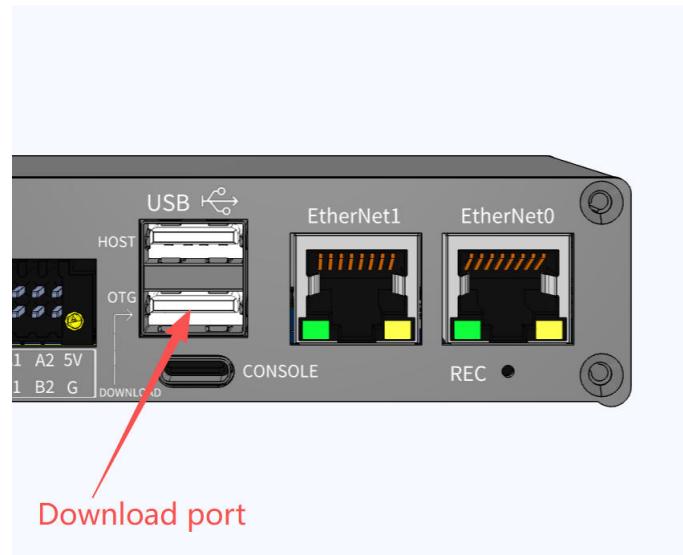
```

## Update the firmware

- Download firmware and upgrade tools from [Google Drive](#) or [Baidu Cloud](#).



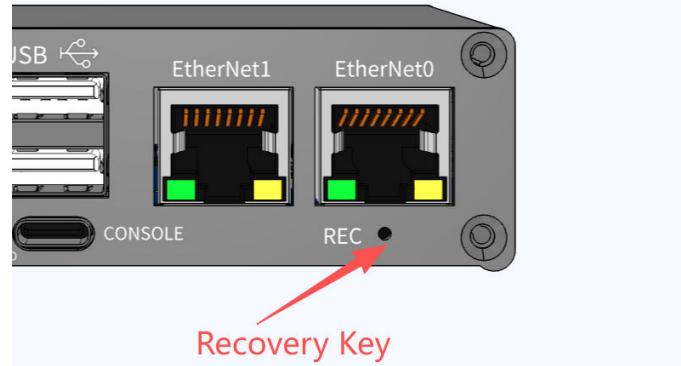
- Connect the USB port of CP2b to the computer.



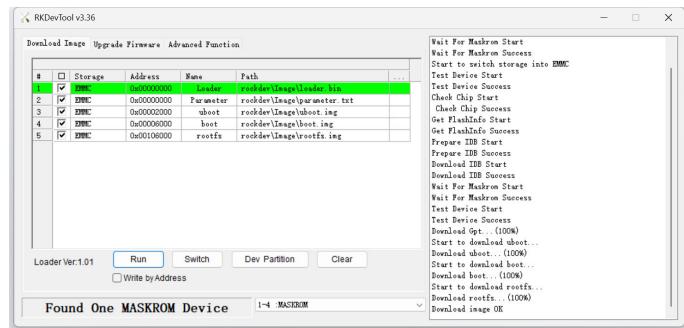
- Install USB driver using the DriveAssitant-v5.12 tool.



- Press and hold the REC button on the machine, then turn on the power and the machine will enter MASKROM mode.



- Open RKdevtool, by default, the firmware path has already been selected. Click on 'Run' directly.



## Compile and update the kernel

- Synchronize kernel code and compile

```
git clone https://github.com/coolpi-george/coolpi-kernel.git
git checkout linux-6.1-stan-rkr5.1
./build-kernel.sh arm
```

- update kernel

After compilation, the following deb file will be generated and copied to the machine for installation using the "dpkg -i linux-image-6.1.115\_6.1.115-21\_armhf.deb" command.

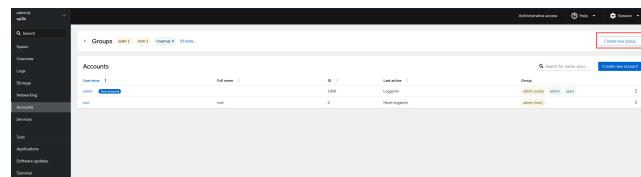
|                                              |                  |              |           |
|----------------------------------------------|------------------|--------------|-----------|
| linux-headers-6.1.115_6.1.115-21_armhf.deb   | 2025/10/28 16:09 | DEB 文件       | 8,502 KB  |
| linux-libc-dev_6.1.115-21_armhf.deb          | 2025/10/28 16:09 | DEB 文件       | 1,338 KB  |
| linux-image-6.1.115_6.1.115-21_armhf.deb     | 2025/10/28 16:09 | DEB 文件       | 6,214 KB  |
| linux-image-6.1.115-dbg_6.1.115-21_armhf.deb | 2025/10/28 16:10 | DEB 文件       | 68,897 KB |
| linux-upstream_6.1.115-21_armhf.buildinfo    | 2025/10/28 16:10 | BUILDINFO 文件 | 7 KB      |
| linux-upstream_6.1.115-21_armhf.changes      | 2025/10/28 16:10 | CHANGES 文件   | 3 KB      |

## Common problems and solutions

- How to change default password ?

```
sudo passwd admin
```

- How to add a new user? As shown in the following figure, new users can be added and permissions can be configured through the backend management software.



- How to connect to WiFi network ?

```
/*Find available WiFi networks*/
nmcli dev wifi list
/*To connect to a WiFi network, you need to replace<SSID>with the network
name you want to connect to, and<password>with the password for that
network:*/
nmcli --ask dev wifi connect <SSID> password <password>
```

- Unable to register for 4G network?

Pay attention to the insertion direction of the SIM card as shown in the figure below, with the notch facing outward.



- How to Backup File System?

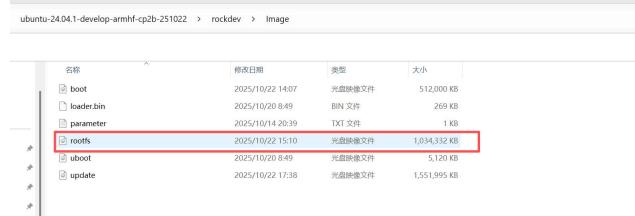
In the process of user development, after building their own application, it is usually necessary to back up the file system and then copy it to other machines. The following provides [backup scripts](#) and operation methods:

- It is best to use a file system with a capacity greater than twice that of a USB flash drive, for example, if the file system is 4GB, choose an 8GB capacity USB flash drive and format it in NTFS format.
- Copy the script file to a USB drive.
- Insert the USB drive into the USB port of the CP1B machine and turn it on.

- Use the following command to mount a USB drive to the/mnt directory.  
`sudo mount /dev/sda1 /mnt`
- Enter the/mnt directory and execute the script.

```
cd /mnt
sudo ./backup-cp2b.sh
```

- After the script is executed, the root directory of the USB drive will generate a \* \*. img file, which can be used to replace the rootfs. img file in the compressed image file.

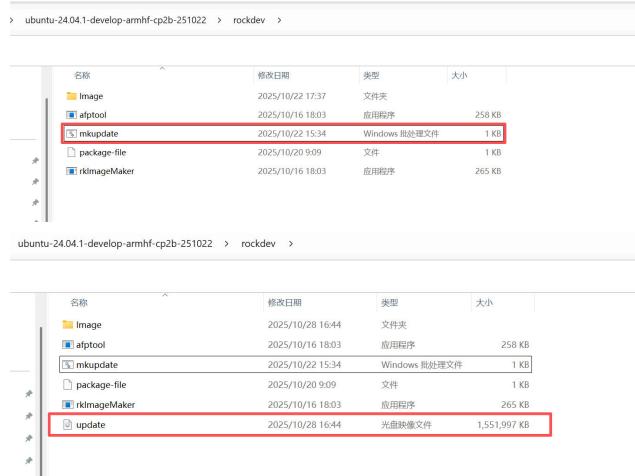


| 名称         | 修改日期             | 类型     | 大小           |
|------------|------------------|--------|--------------|
| boot       | 2025/10/22 14:07 | 光盘映像文件 | 512,000 KB   |
| loader.bin | 2025/10/20 8:49  | BIN 文件 | 269 KB       |
| parameter  | 2025/10/14 20:39 | TXT 文件 | 1 KB         |
| rootfs     | 2025/10/22 15:10 | 光盘映像文件 | 1,034,332 KB |
| uboot      | 2025/10/20 8:49  | 光盘映像文件 | 5,120 KB     |
| update     | 2025/10/22 17:38 | 光盘映像文件 | 1,551,995 KB |

- How to make mass production firmware?

The mass production process can be completed using production tools, which require loading production firmware and cannot use development firmware. The following steps for generating production firmware are introduced:

- Open the folder shown in the following figure, double-click to execute mkupdate.bat, and the script will generate update.img after running



| 名称           | 修改日期             | 类型            | 大小     |
|--------------|------------------|---------------|--------|
| Image        | 2025/10/22 17:37 | 文件夹           |        |
| apitool      | 2025/10/16 18:03 | 应用程序          | 258 KB |
| mkupdate     | 2025/10/22 15:34 | Windows 批处理文件 | 1 KB   |
| package-file | 2025/10/20 9:09  | 文件            | 1 KB   |
| rkimageMaker | 2025/10/16 18:03 | 应用程序          | 265 KB |

| 名称           | 修改日期             | 类型            | 大小           |
|--------------|------------------|---------------|--------------|
| Image        | 2025/10/28 16:44 | 文件夹           |              |
| apitool      | 2025/10/16 18:03 | 应用程序          | 258 KB       |
| mkupdate     | 2025/10/22 15:34 | Windows 批处理文件 | 1 KB         |
| package-file | 2025/10/20 9:09  | 文件            | 1 KB         |
| rkimageMaker | 2025/10/16 18:03 | 应用程序          | 265 KB       |
| update       | 2025/10/28 16:44 | 光盘映像文件        | 1,551,997 KB |

- Open the factory tool, load update.img, run, and the machine will automatically execute the burning process when it enters maskrom mode. It is recommended not to burn more than 16 machines simultaneously.

