

Cool Pi4 Model B User Guide

Cool Pi4 Model B User Guide

Welcome to Cool Pi

Start your Coolpi

Prepared

Connecting

Supply Power

Install Operating System

Upgrade Tools

Raspberry Pi tools

Rockchip Tools (Only on eMMC)

System image

Debian11

Ubuntu Desktop 22.04

System Code

Get Source Code

update bootloader

How to Compile

Update

Kernel update

Driver update

DTS update

Hardware design

Test

unixbench

benchmark

Extension interface

40PIN

UART

I2C

GPIO

PWM

Third party library

ROS2 && TEST

openCV (Python)

openCV (C++)

python

openCL

OpenGL ES

Pangolin

docker

Other

NPU

BT

Hardware Decoding

Hardware Encoding

QT5

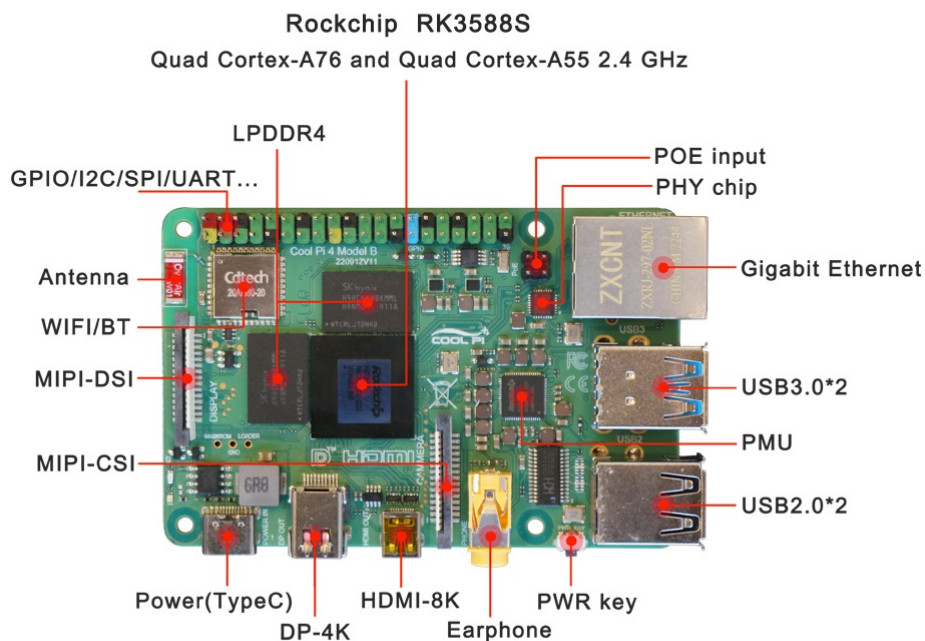
Welcome to Cool Pi

Cool Pi4 Model B is one of COOL PI SBCs-Single Board Computers. It has a powerful Rockchip RK3588s SoC, 2.4GHz Quad core ARM Cortex-A73 and 1.8GHz Quad core Cortex-A55 CPU, with 8K@60fps display and 6Tops Neural-network Processing Unit.

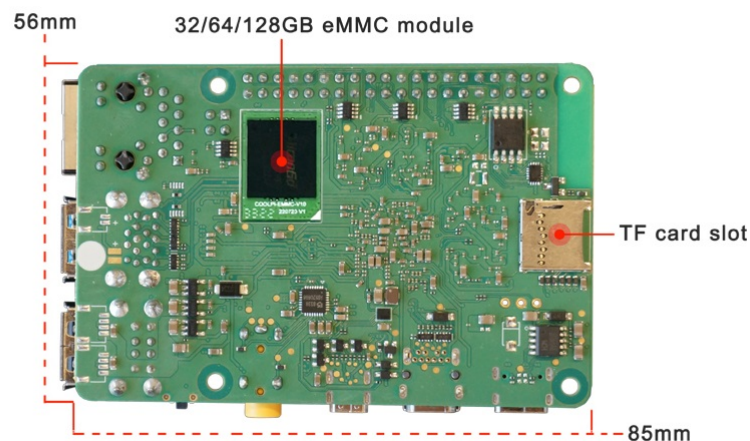
What's CoolPi 4 Model B?

It's an open-source single board computer. It can run Android12, Ubuntu22, Debian11. It uses the Rockchip RK3588S Soc, and has Onboard 2/4/8/16GB LPDDR4 memory and 32/64/128GB eMMC (Option)

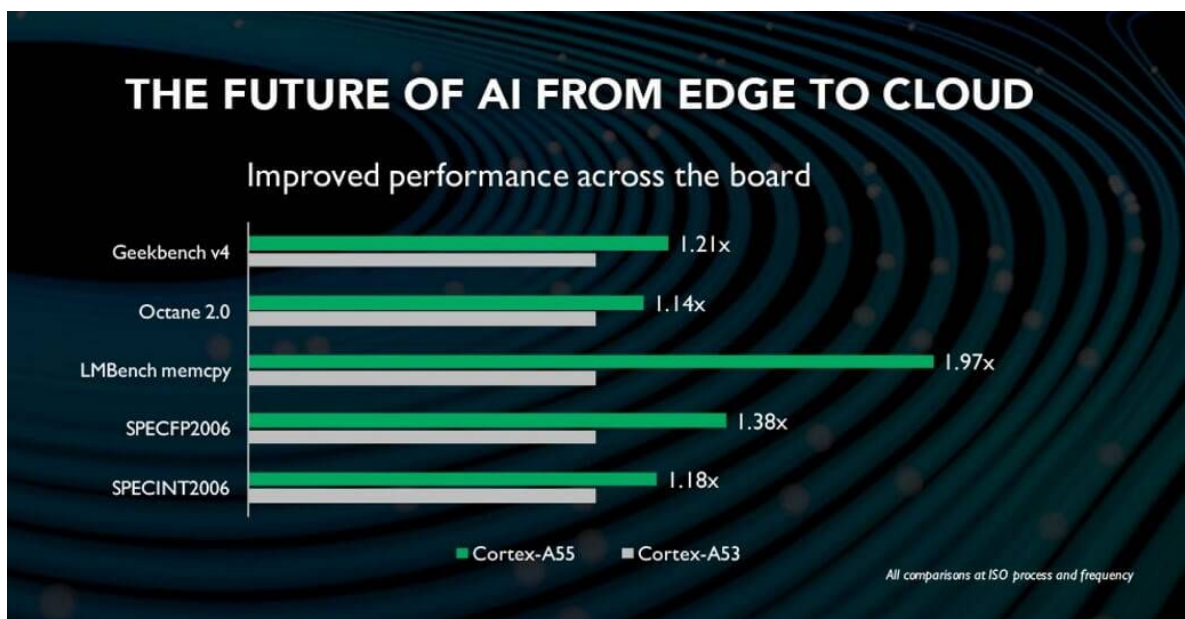
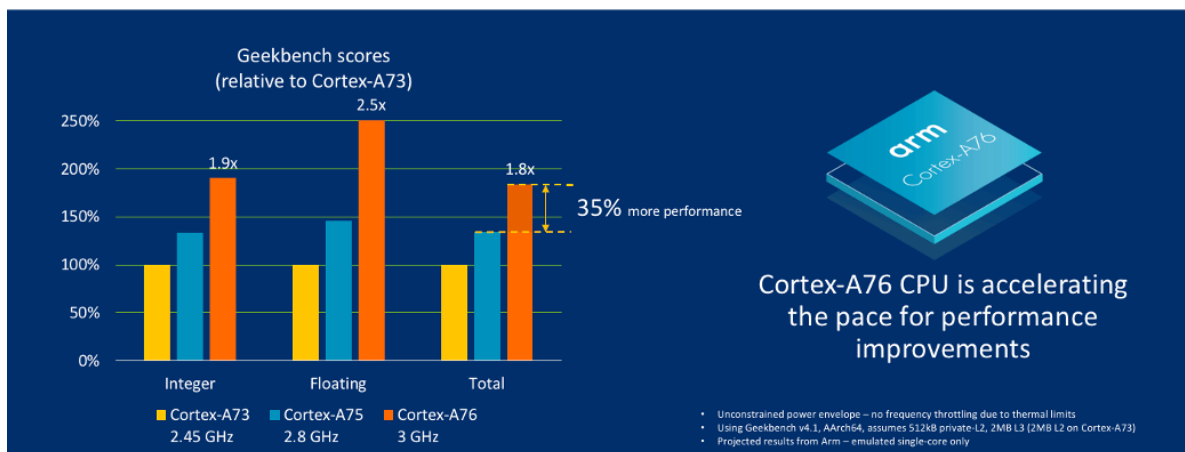
Top view



Bottom view



Accelerating performance for premium solutions



Start your Coolpi

Prepared

- 5V/9V Adapter with Type-C cable
- 4K HDMI 2.0/HDMI 2.1 cable
- HDMI-compatible 1080P/4K/8K monitor
- USB keyboard and mouse
- TF Card / eMMC module / Udisk / MHD (above 16GB)
- One card reader

Connecting

- Connect Coolpi to monitor with HDMI cable directly
- Connect Coolpi to keyboard and mouse
- Install the eMMC module into the card reader (eg. eMMC)

Supply Power

Cool Pi4 model B Cool supports various type-C interface adapters on the market, which can be powered by mainstream Raspberry Pi adapters which is 5V/3A.

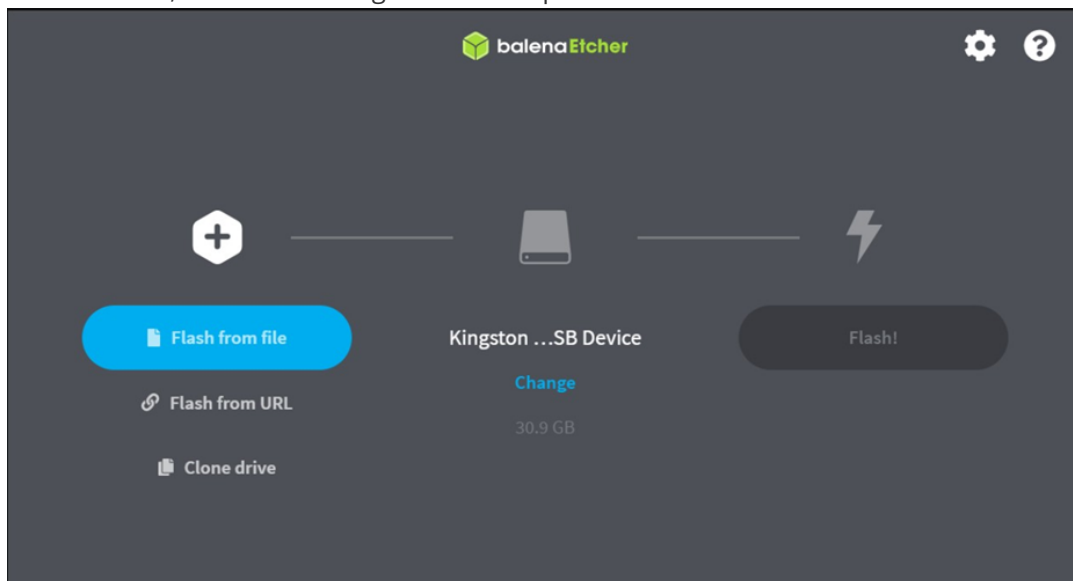
Install Operating System

Upgrade Tools

Coolpi series development boards support a variety of system upgrade methods. One is compatible with the offline method of raspberry pi tools, and the other is using RK tool to upgrade.

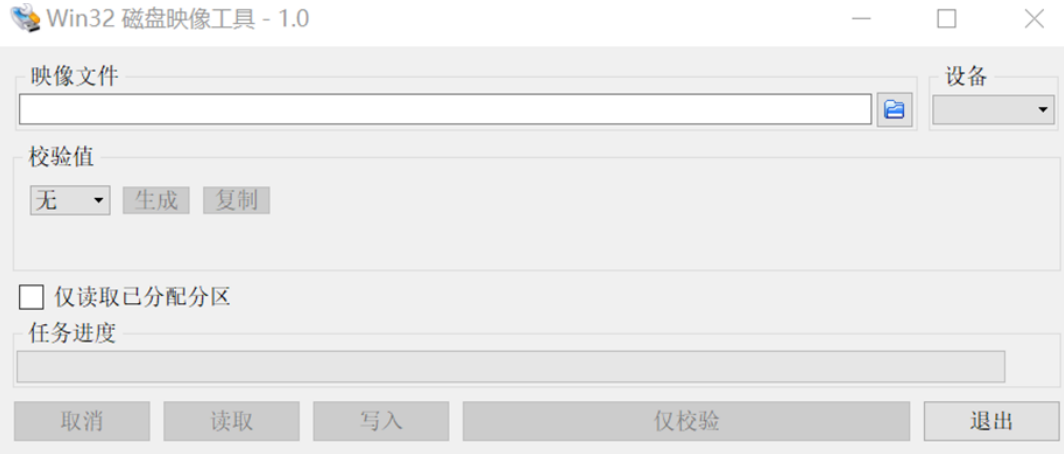
Raspberry Pi tools

- [balenaEtcher](#)
 - Open it with administrator privileges
 - Select firmware.img fiile
 - Insert the storage your Windows computer (eg. eMMC)
 - Select the storage device
 - Click Flash button, waiting
 - After finished, Install the storage on the coolpi.



- [Win32DiskImager](#)
 - Open & Select firmware.img file
 - Insert the storage your Windows computer (eg. eMMC)
 - Click Write button, waiting

- After finished, Install the storage on the coolpi.



- **Linux flash offline**

- Debian/Ubuntu system (eg. device node = /dev/sdx)
 - dd if=system.img of=/dev/sdx bs=1M status=progress; sync
 - tips: TF Card=/dev/mmcblk2; eMMC=/dev/mmcblk0

Rockchip Tools (Only on eMMC)

- **For Linux(Ubuntu)**

- Linux host for x86
- [upgrade tool.zip](#)

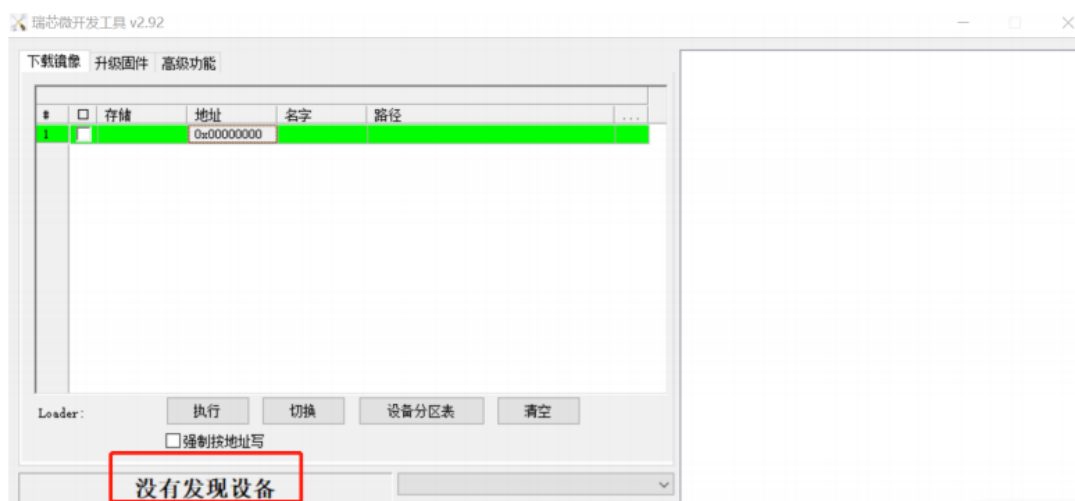
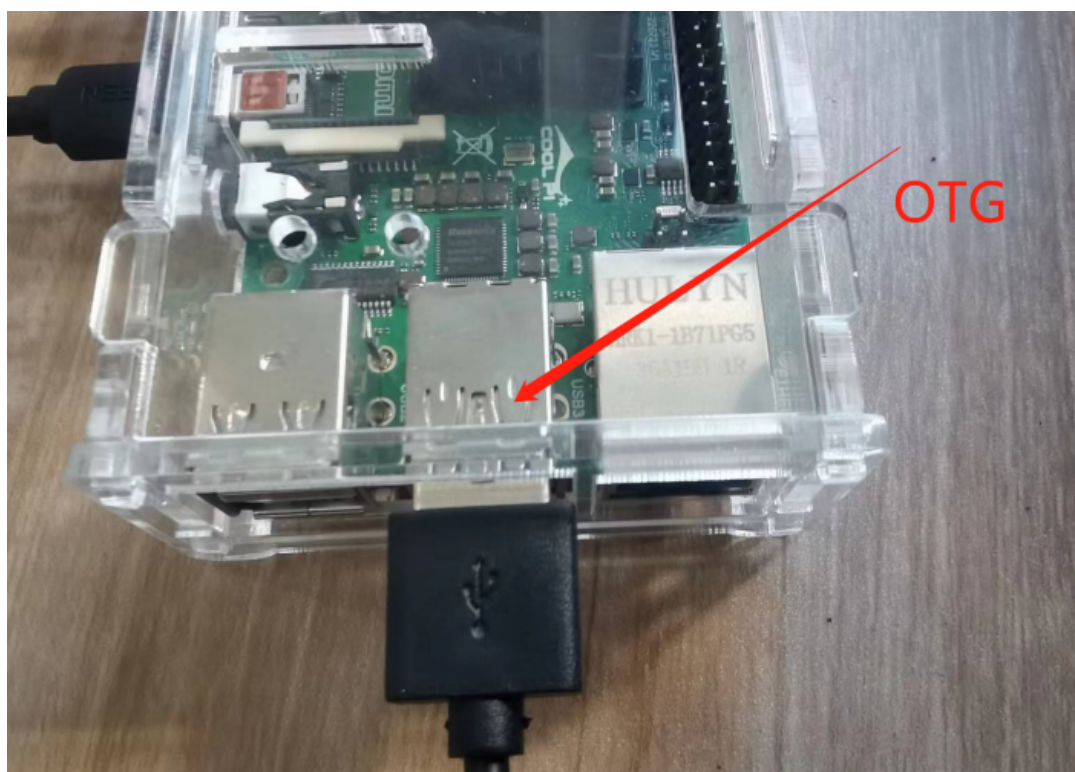
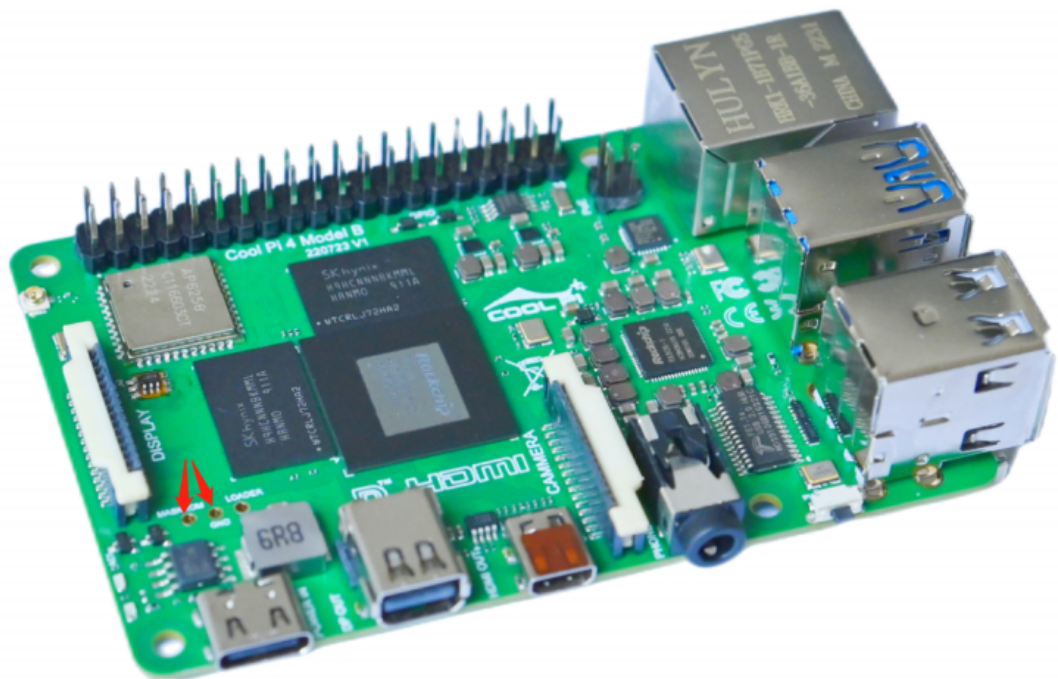
```
# unzip upgrade_tool.zip# cp upgrade_tool /usr/bin# upgrade_tool
Program Log will save in the /root/upgrade_tool/log/
List of rockusb connected
DevNo=1 Vid=0x2207,Pid=0x350b,LocationID=21 Mode=Maskrom
root@jack-server:/home/jack/Documents/rktool-img# upgrade_tool uf
20221017-ubuntu-22.04-preinstalled-desktop-arm64-coolpi-emmc-only.img
Program Log will save in the /root/upgrade_tool/log/
Loading firmware...
Support Type:RK3588 FW Ver:1.0.00 FW Time:2022-10-17 17:00:51
Loader ver:1.0b Loader Time:2022-09-03 21:39:46
Upgrade firmware ok.
root@jack-server:/home/jack/Documents/rktool-img#
```

- **For MAC(X86)**

- Mac host
- [upgrade tool v2.3 mac.zip](#)
- following linux method

- **For Windows(X86)**

- Connect Type-A USB to coolpi (Only upper USB3)
- Change to Maskrom Mode
- Power ON or Reset
- Install USB Driver
- Select image file && upgrade



下载镜像

升级固件

高级功能

固件

升级

切换

固件版本:

1.0.00

Loader版本:

1.0b

芯片信息:

RK3588

固件:

'ubuntu-22.04-preinstalled-desktop-arm64-coolpi-emmc-only.img'

System image

Debian11

20221106-debian-11-preinstall-desktop-coolpi.zip
[Offline mode, compatible with raspberry pi flash mode]
Optimize kernel driver, default English language, support rknn development environment, etc
Download: <https://pan.baidu.com/s/1dZRMJzGfCQ8TkWH4ZJ5BzQ>
Code: cicv

20221106-debian-11-preinstalled-desktop-arm64-coolpi-emmc-only.zip
[Online by RKDevTool.exe or upgrade_tool]
Download: https://pan.baidu.com/s/1-ei7F-FysYm6h9Fg_-xLRw
Code: i2h3

Ubuntu Desktop 22.04

20221106-ubuntu-22.04-preinstalled-desktop-arm64-coolpi.zip
System optimization, new driver support, desktop X11 version, supporting graphics video decoding
[Offline mode, compatible with raspberry pi flash mode]
Download: <https://pan.baidu.com/s/1Rv-TpKcmdV6DWjVMRWSoyQ>
Code: u6ta

20221106-ubuntu-22.04-preinstalled-desktop-arm64-coolpi-emmc-only.zip
[Online by RKDevTool.exe or upgrade_tool]
Download: <https://pan.baidu.com/s/1Gyx-6Kof1tCKPFcQUJ7tug>
Code: t0n3

System Code

Get Source Code

- linux kernel

<https://gitee.com/yanyitech/coolpi-kernel.git>

- bootloader

<https://gitee.com/yanyitech/coolpi-loader.git>

update bootloader

For the Cool Pi 4B, update the u-boot as follows:

Update to version 0904 if version is 0823

- download [uboot.zip](#) file
- Unzip uboot.zip and copy bin file to root directory of U disk or Tf card
- Insert U disk or Tf card into Cool PI
- Connect Debug Uart to PC, open putty
- Power on and input uppercase 'CP' to stop Startup

Note: only support USB3.0 lower port

- Output as follow

```
COOLPI  LOADER  BY  YANYI(Aug  23  2022)
```

```
DRAM:  3.7 GiB
```

```
CR: M/C/I
```

```
Using default environment
```

```
A) U disk
```

```
usb reset
```

```
load usb 0:1 0x800000 uboot.img
```

```
mtm erase nor0 0x200000 0x500000
```

```
mtm write nor0 0x800000 0x200000 0x500000
```

```
# Never power off or restart during the upgrade process
```

```
B) TF Card
```

```
mmc dev 1
```

```
load mmc 1:1 0x800000 uboot.img
```

```
mtm erase nor0 0x200000 0x500000
```

```
mtm write nor0 0x800000 0x200000 0x500000
```

```
# Never power off or restart during the upgrade process
```

- Wating finished, reset, Output:

COOLPI LOADER BY YANYI(Sep 03 2022)

DRAM: 3.7 GiB

Using default environment

How to Compile

connect the coolpi board to the network (via wifi or network cable)

- **Ubuntu 22.04 Desktop**

```
# Install environment
sudo ln -sf /usr/bin/python2 /usr/bin/python
sudo apt install flex bison

# Option
sudo dpkg-reconfigure dash # choose "no"

# download source code
mkdir test
cd test/
git clone https://gitee.com/yanyitech/coolpi-kernel.git

# Compile by root
sudo -i
cd /home/coolpi/test/coolpi-4b-kernel/
./build-kernel.sh

# you can change kernel config by: make ARCH=arm64 menuconfig
```

- **Debian 11**

```
# Install environment
sudo apt-get install git flex bison libssl-dev
ln /usr/bin/python3 /usr/bin/python

# download source code
git clone https://gitee.com/yanyitech/coolpi-4b-kernel.git

# Compile by root
sudo -i
cd /home/coolpi/git/coolpi-4b-kernel/
./build-kernel.sh
```

Update

SSH, Uart debug or Self terminal

```
coolpi login: coolpi
Password:
welcome to Ubuntu 22.04 LTS (GNU/Linux 5.10.66 aarch64)

* Documentation:  https://help.ubuntu.com
* Management:    https://landscape.canonical.com
* Support:       https://ubuntu.com/advantage

11 updates can be applied immediately.
11 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

coolpi@coolpi:~$
coolpi@coolpi:~$ cd /boot/firmware/
coolpi@coolpi:/boot/firmware$ ls
cmdline.txt  config.txt  initrd.img  modules.tar.gz  rk3588s-cp4.dtb  vmlinuz
coolpi@coolpi:/boot/firmware$
```

Kernel update

```
sudo cp vmlinuz /boot/firmware/
# From out/vmlinuz
```

Driver update

```
coolpi@coolpi:/boot/firmware$ sudo rm -rf /lib/modules/5.10.66/
[sudo] password for coolpi:
coolpi@coolpi:/boot/firmware$ sudo tar -xf modules.tar.gz -C /lib
coolpi@coolpi:/boot/firmware$
# From out/modules.tar.gz
```

DTS update

```
sudo cp rk3588s-cp4.dtb /boot/firmware/
# From out/rk3588s-cp4.dtb
```

Reset after operation

```
sudo reboot
```

Hardware design

Version	V10	V11
WIFI Module	AP6256	AIC8800
On-board antenna	Null	2.4G+5G
EMMC Connector	GB042	BM20B

V10 schematic: [COOLPI-4B-Schematic-V10.pdf](#)

V10 label: [COOLPI-4B-Part-V10.pdf](#)

V10 dxf: [coolpi-4b-dxf-v10.dxf](#)

V11 schematic: [COOLPI-4B-Schematic-V11.pdf](#)

V11 label: [COOLPI-4B-Part-V11.pdf](#)

V11 dxf: [COOLPI-4B-DXF-V11.dxf](#)

Test

unixbench

```
sudo apt install gcc make git
git clone https://gitee.com/yanyitech/byte-unixbench.git -b v5.1.3

cd byte-unixbench/UnixBench
./Run
```

benchmark

```
sudo apt install p7zip-full
7z b
```

Extension interface

40PIN

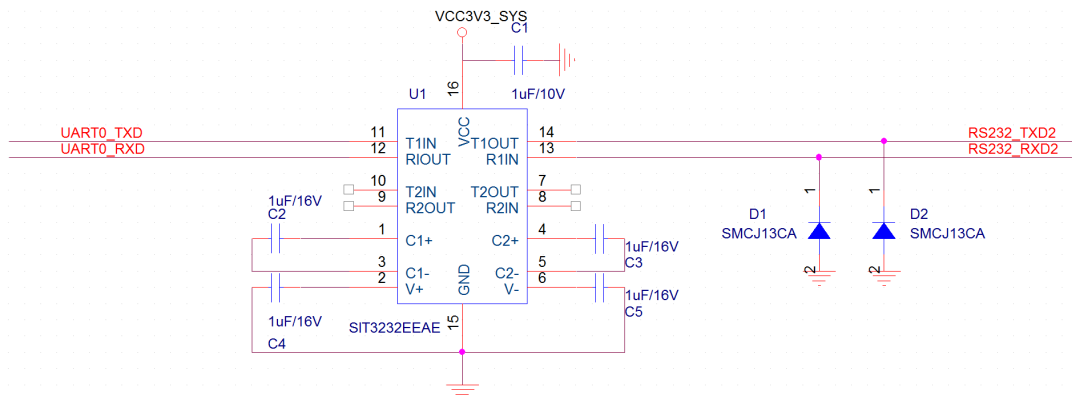
ROCKCHIP RK3588S GPIO can reuse multiple functions. The principle part in the following figure is just one of them. You can modify it according to product requirements. All GPIOs of RK can be used as external interrupts. The default pull-up/pull-down status and output drive strength can be configured.

[GO TO DETIALS](#)

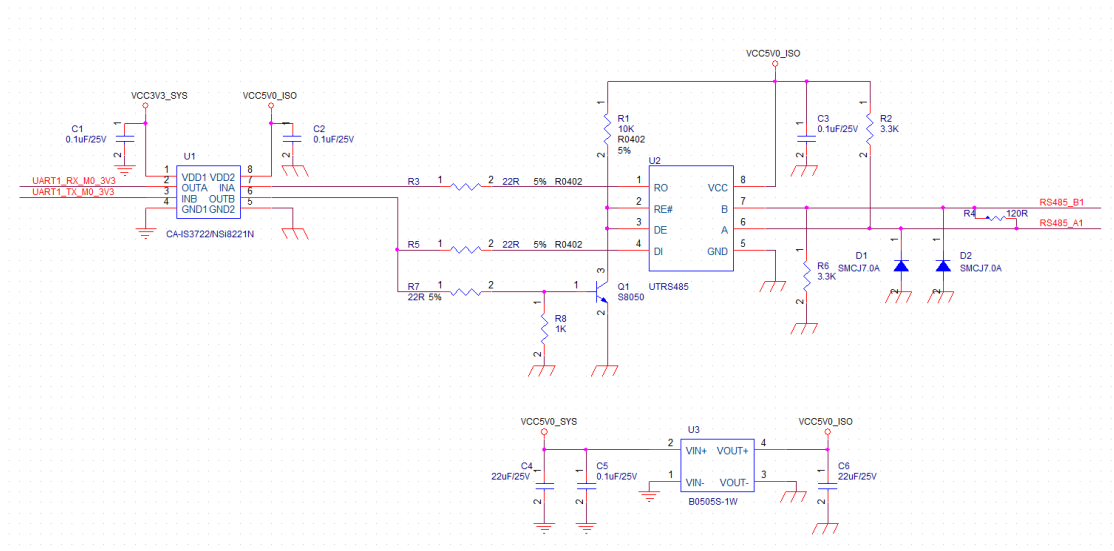
UART

The 40PIN can extend to 5 independent UART except the Debug one. The maximum baud rate is 1.5M.

- RS232



- RS485



[GO TO DETIALS](#)

I2C

The 40PIN can lead out 4 groups of I2C, I2C1 I2C3 I2C5 is an independent I2C interface. I2C6 board is internally connected to the RTC(HYM8563) with an address of 51H. Don't conflict the addresses.

[GO TO DETIALS](#)

GPIO

The 40PIN can be used as general IO except Debug UART and I2C6. All GPIOs support interrupt, pull-down and drive strength configurations.

[GO TO DETAIL](#)

PWM

The 40pin contains 7 GPIOs that can be used to output PWM signals.

[GO TO DETAIL](#)

Third party library

ROS2 && TEST

```
wget http://fishros.com/install -O fishros && bash fishros
```

[GO TO DETAIL](#)

openCV (Python)

```
pip install opencv-python
pip install opencv-contrib-python
# Check version
pip list | grep opencv
```

openCV (C++)

```
wget https://github.com/opencv/opencv/archive/4.6.0.zip
unzip 4.6.0.zip
cd opencv-4.6.0/
mkdir build
cd build/
cmake -D CMAKE_BUILD_TYPE=Release -D CMAKE_INSTALL_PREFIX=/usr/local ..
make -j8
sudo make install
```

python

```
# Install python3-dev
sudo apt-mark unhold python3* python3-* libpython3* libexpat1
sudo apt install python3-dev
# Install pip
sudo apt install python3-pip
```

openCL

```
sudo apt-get install ocl-icd-* opencl-headers
sudo apt-get install clinfo
# Check version
clinfo
```

openGL ES

```
sudo apt-get install libgles2-mesa-dev
# Check version
glxinfo |grep "OpenGL ES"
```

Pangolin

```
# Install GLEW
sudo apt-get install libglew-dev
# Install dependents
sudo apt-get install make cmake clang lld lz4 coreutils qemu qemu-user-static
bison flex libssl-dev bc git gcc g++ libeigen3-dev libpng-dev
# Install boost
sudo apt-get install libboost-dev libboost-thread-dev
# Install Pangolin
git clone http://github.com/stevenlovegrove/Pangolin.git
cd Pangolin
mkdir build
cd build
cmake -DCPP11_NO_BOOST=1 ..
make -j8
sudo make install
```

docker

```
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o
/etc/apt/keyrings/docker.gpg

echo "deb [arch=$(dpkg --print-architecture) signed-
by=/etc/apt/keyrings/docker.gpg] https://download.docker.com/linux/ubuntu \
$(lsb_release -cs) stable" | sudo tee /etc/apt/sources.list.d/docker.list >
/dev/null

apt update

apt-get install docker-ce docker-ce-cli containerd.io docker-compose-plugin
```

Other

[NPU](#)

[BT](#)

Hardware Decoding

Hardware Encoding

QT5

Support: www.cool-pi.com

Email: coolpi@cool-pi.com