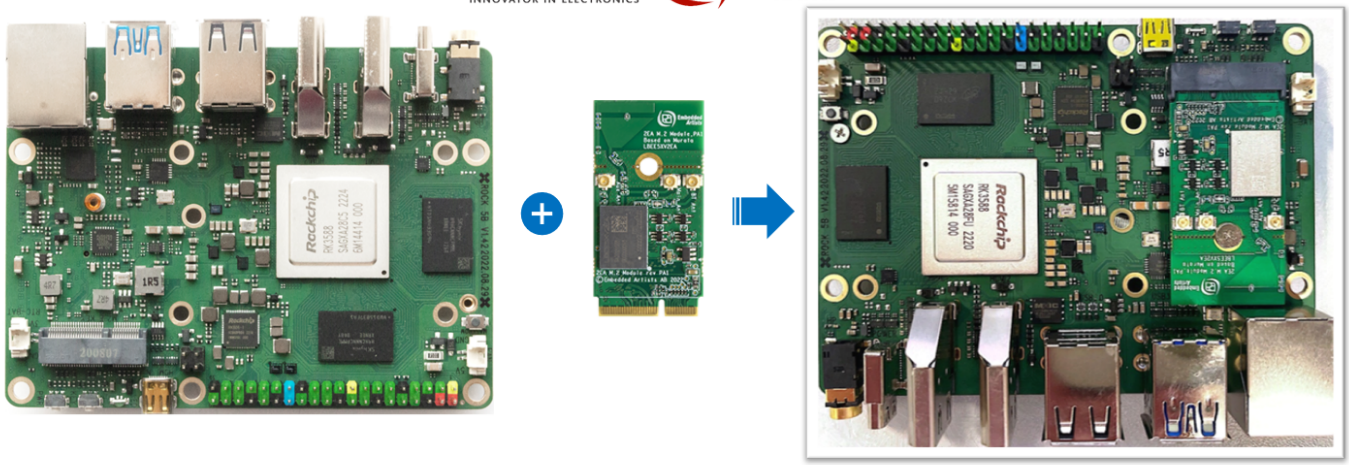
Infineon AIROC Wi-Fi/Bluetooth Chip Bring-up with RK3588 (ROCK 5B)

2 Infineon AIROC Wi-Fi Chip Bring up with OKDO ROCK 5B

HW Kit prepared for OKDO 5B (RK3588) with Infineon AIROC Wi-Fi 6/6E Combo

- **OKDO 5B (RK3588) 8GB + CYW55573 (MuRata Type-2EA) or CYW54591 (MuRata Type-1XA)**
- **OKDO 5B Order link -** https://us.rs-online.com/product/okdo/rock-5-b-8gb/73761293/?gclid=CjwKCAjwloynBhBbEiwAGY25dD4Z-QE4r__vpeY0suz4NVNWskQDbdFveGGlayw4xgVxs_ba4viJFxoC3FMQAvD_BwE&gclsrc=aw.ds
- **MuRata / EA 2EA Wi-Fi 6/6E Module (CYW55573) Order link -** https://www.embeddedartists.com/products/2ea-m-2-module/
- **MuRata / EA 1XA Wi-Fi 5 RSDB Module (CYW54591) Order link -** https://www.embeddedartists.com/products/1xa-m-2-module/

- **OKDO 5B (RK3588) 8GB + CYW55573 (MuRata Type-2EA)**



3 Rock 5B Introduction

Please refer to below sections for basic bring-up guide including how to flash the new Kernel image to SD card and boot.

Rock 5B is Rockchip RK3588 based SBC(Single Board Computer) by Radxa (https://wiki.radxa.com/Home) . It can run Linux, Android, BSD and other distributions.

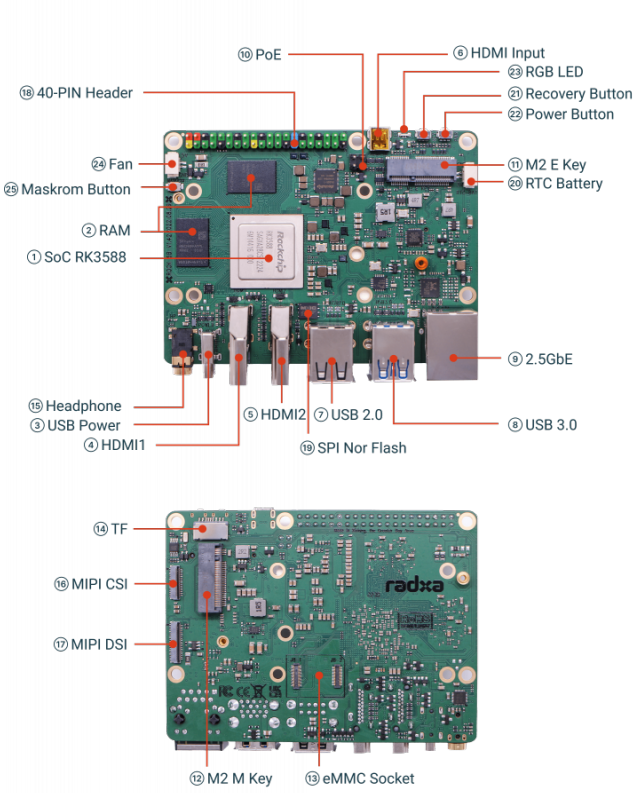
The SoC Rockchip RK3588 is Arm® DynamIQ™ Quad Cortex® A76, Quad Cortex® A55 CPU, and Arm Mali™ G610MC4 GPU , it has frequency up to **2.4 GHz**

HW schematic: radxa_rock_5b_v142_sch.pdf

Components position reference: radxa_rock_5b_v142_smd.pdf

It has a **M.2 E key** can support both **SDIO** and **PCIE**.

- Official website: https://wiki.radxa.com/Rock5/5b
- Where to buy: https://wiki.radxa.com/Buy



4 Bring-up Information

- Rock 5B (16 GB RAM+ 64 GB eMMC)
- **PCIE**: CYW955573PCM2FCIPA ((Hatchet-2 A2) ; **SDIO**: CYW955573SDM2FCIPA (Hatchet-2 A2), Quectel FC80A (CYW54591 SDIO)
- Ubuntu 20.04 (focal) server
- Kernel 5.10
- Infineon FMAC Godzilla ES100.1 - https://community.infineon.com/t5/Wi-Fi-Bluetooth-for-Linux/Cypress-Linux-WiFi-Driver-Release-FMAC-2023-08-01/td-p/459849

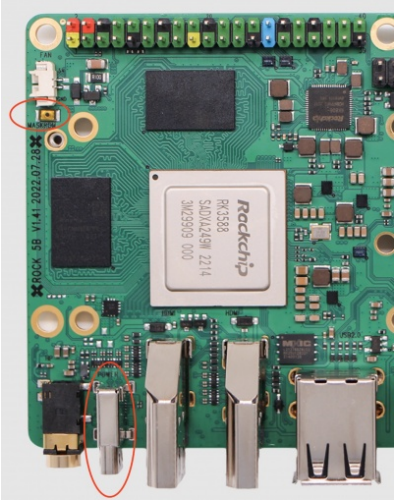
5 Install the image to eMMC from USB port

5.1

Please check this guide : https://wiki.radxa.com/Rock5/install/rockchip-flash-tools

5.2

- Power off the board.
- Press the golden button and hold it.
- Plug the USB-A to Type-C cable to ROCK 5B Type-C port, the other side to PC.
- Release the golded button.



5.3

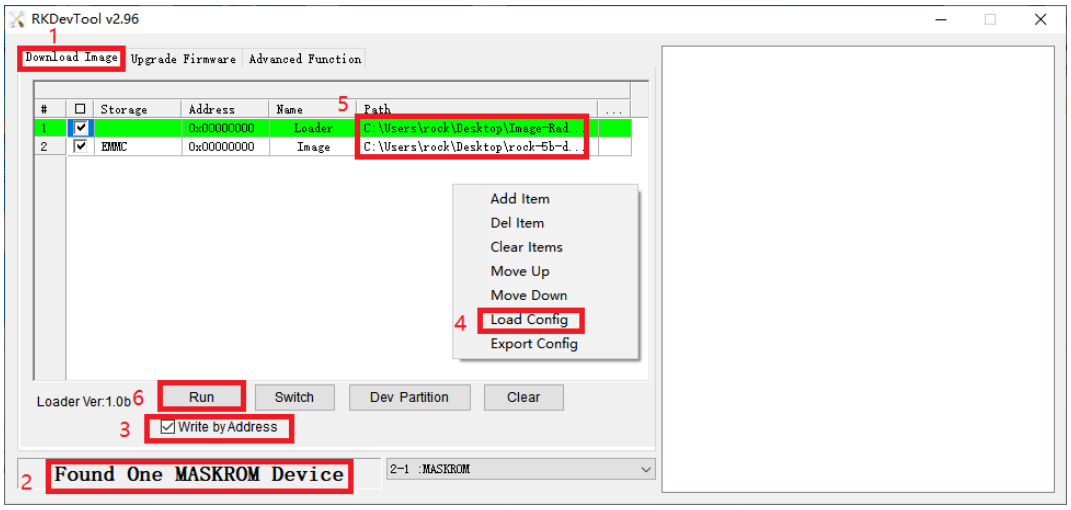
- RK3588 loader : [4] rk3588_spl_loader_v1.08.111.bin
- rock-5b-ubuntu-focal-server-arm64-20221031-1328-gpt.img : Download from this link : https://wiki.radxa.com/Rock5/downloads

5.4 Write image to eMMC module

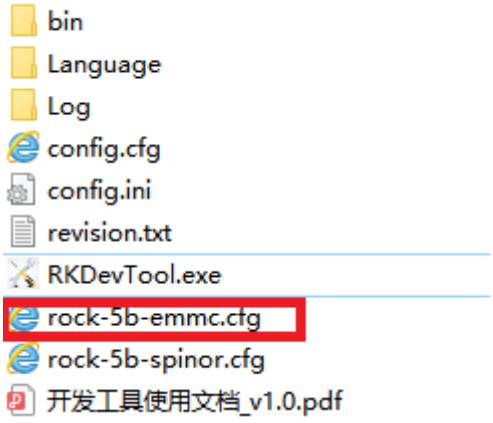
Windows only. For MacOS/Linux, please refer to : https://wiki.radxa.com/Rock5/install/usb-install-emmc

a. Run the RKDevTool.exe. If you do not have this, you can download it here. [5]RKDevTool_Release_v2.96.zip

b. Check whether it is in maskrom mode. The red part (2) in the figure below indicates that it has entered maskrom mode

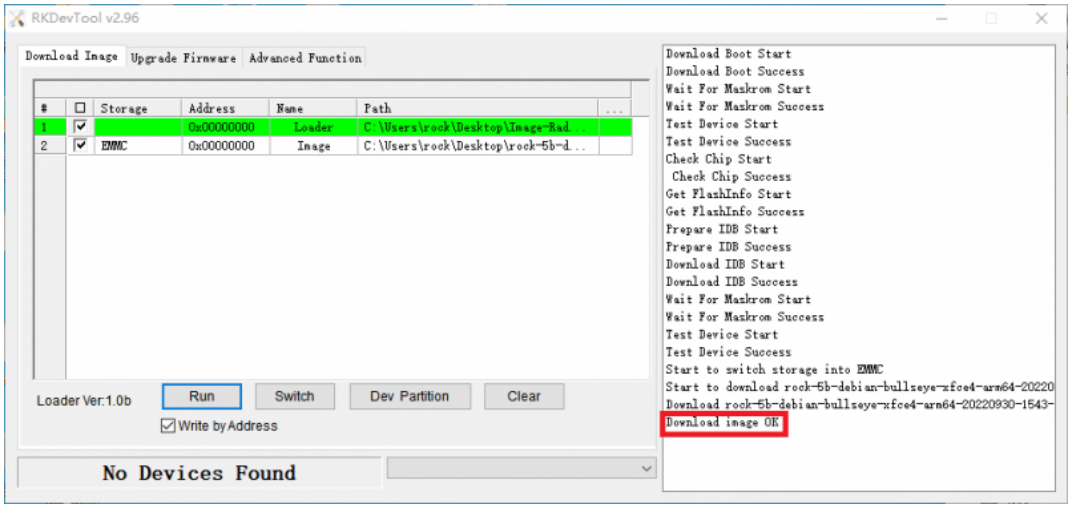


c. Check the "**Write by Address**" option.
d. Right-click in the blank area of the list and select Load configuration file



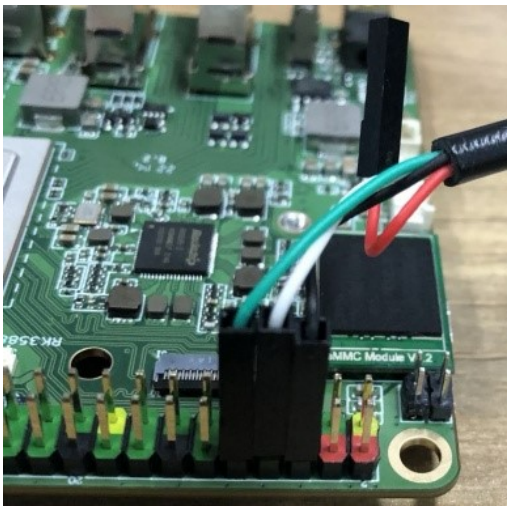
e. After loading the configuration file, you need to modify the path of the file to be downloaded.

f. Now, Click "**Run**", if successful, you will see the download successful message in the right box.



6 Access from the Host PC/Laptop

6.1 How to use serial console



Rock 5B	USB to TTL cable
Rx (Pin10)	Green wire
TX (Pin8)	White wire
GND (Pin6)	Black wire

The default serial setting for ROCK 5 u-boot and kernel console is:

```
baudrate: 1500000
data bit: 8
stop bit: 1
parity  : none
flow control: none
```

6.2 SSH

SSH server is enabled on port 22 of ROCK 5 default image.

```
ssh rock@ip address
```

Default account

```
User Name: rock
Password: rock
```

7 Kernel Build

Since the default Rockchip Ubuntu Kernel comes with certain configurations that are not suitable for FMAC build, you need to cross-compile it with correct kernel configurations.

PS: Cross-compiling the Kernel is preferred since crashes have been observed natively compiling the kernel on Rockpi 5B when using SD card boot.

7.1 Install essential packages

sudo apt-get update
sudo apt-get install -y git device-tree-compiler libncurses5 libncurses5-dev build-essential libssl-dev mtools bc python dosfstools bison flex rsync u-boot-tools make
Download cross-compiler toolchain
wget https://releases.linaro.org/components/toolchain/binaries/7.3-2018.05/aarch64-linux-gnu/gcc-linaro-7.3.1-2018.05-x86_64_aarch64-linux-gnu.tar.xz
sudo tar xvf gcc-linaro-7.3.1-2018.05-x86_64_aarch64-linux-gnu.tar.xz -C /usr/local/

7.2 Get the source code

mkdir ~/src/rockpi/rk3588-sdk && cd ~/src/rockpi/rk3588-sdk
git clone -b linux-5.10-gen-rkr3.4 https://github.com/radxa/kernel.git
git clone -b master https://github.com/radxa/rkbin.git
git clone -b debian https://github.com/radxa/build.git
And you will get
build kernel rkbin

7.3 Cross-compile Kernel package

Optionally, if you want to change the default kernel config
cd ~/src/rockpi/rk3588-sdk/kernel

```
export ARCH=arm64
export CROSS_COMPILE=/usr/local/gcc-linaro-7.3.1-2018.05-x86_64_aarch64-linux-gnu/bin/aarch64-linux-gnu-

make rockchip_linux_defconfig
#(change CONFIG_CFG80211 to m)
make menuconfig
make -j8

Build kernel deb package

export build_id="999"          # make sure it's big enough so that our kernel is the newest.
export lv="-$build_id-rockchip"
export kv=$(make kernelversion)
export debv="$kv$lv"

make bindeb-pkg -j8 LOCALVERSION=$lv KDEB_PKGVERSION=$debv
```

```
ls ../*.deb
../linux-firmware-image-4.4.167-999-rockchip-g8b7b311_4.4.167-999-rockchip_arm64.deb
../linux-headers-4.4.167-999-rockchip-g8b7b311_4.4.167-999-rockchip_arm64.deb
../linux-image-4.4.167-999-rockchip-g8b7b311_4.4.167-999-rockchip_arm64.deb
../linux-image-4.4.167-999-rockchip-g8b7b311-dbg_4.4.167-999-rockchip_arm64.deb
../linux-libc-dev_4.4.167-999-rockchip_arm64.deb
```

8 FMAC Driver Bring-up for CYW55573 and CYW54591

Cross-compile FMAC driver.

Host machine: Ubuntu 20.04 (should work on later versions as well)

export MY_KERNEL=~/src/rockpi/rk3588-sdk/kernel/
make KLIB=$MY_KERNEL KLIB_BUILD=$MY_KERNEL defconfig-brcmfmac
unset LDFLAGS
make ARCH=arm64 CROSS_COMPILE=/usr/local/gcc-linaro-7.3.1-2018.05-x86_64_aarch64-linux-gnu/bin/aarch64-linux-gnu- KLIB=$MY_KERNEL KLIB_BUILD=$MY_KERNEL modules

- Copy *.ko which are built above to the board.
  - ./compat/compat.ko
  - ./net/wireless/cfg80211.ko
  - ./drivers/net/wireless/broadcom/brcm80211/brcmutil/brcmutil.ko
  - ./drivers/net/wireless/broadcom/brcm80211/brcmfmac/brcmfmac.ko
- Unpack cypress-firmware-v5.15.58-2023_0801.tar.gz and copy firmware/* to /lib/firmware/cypress.
- Copy appropriate nvram files to /lib/firmware/cypress.
  - CYW54591-PCIE
    - Murata type.1XA
      - https://github.com/murata-wireless/cyw-fmac-nvram/blob/master/cyfmac54591-pcie.1XA.txt
    - AzureWave:
      - [6]cyw954591wlipa_rev1_3_2range_tssi_new_BPF_20220217.txt
    - Copy this as /lib/firmware/cypress/cyfmac54591-pcie.txt
  - CYW5557x-PCIE:
    - Murata type.2EA
      - Murata will upload nvram to the Github soon - https://github.com/murata-wireless/cyw-fmac-nvram
    - AzureWave AW-XH326
      - [7]0615_1323V04_cyfmac55572-pcie.txt
    - Copy it as /lib/firmware/cypress/cyfmac55572-pcie.txt

Check if device exist.

CYW54591-PCIE

```
Board $> lspci
(...)
0001:01:00.0 Network controller: Broadcom Inc. and subsidiaries Device 4417 (rev 0d)
```

CYW55573-PCIE

```
0001:01:00.0 Network controller: Anchor Chips Inc. Device bd31 (rev 01)
```

Load fmac driver.

```
Board $> sudo insmod ./compat.ko
Board $> sudo insmod ./cfg80211.ko
Board $> sudo insmod ./brcmutil.ko
Board $> sudo insmod ./brcmfmac.ko
```

dmesg log is as follows.

CYW54591-PCIE

```
[  552.801671] compat: loading out-of-tree module taints kernel.
[  552.807754] compat: module verification failed: signature and/or required key missing -
tainting kernel
[  552.818857] Loading modules backported from Linux version v5.15.58-2023_0222-0-
gd4d1a61163b5
[  552.818860] Backport generated by backports.git v5.15.58-1-0-g42a95ce7
[  552.990016] usbcore: registered new interface driver brcmfmac
[  552.990813] brcmfmac 0001:01:00.0: Adding to iommu group 9
[  552.991490] brcmfmac 0001:01:00.0: enabling device (0000 -> 0002)
[  553.102994] brcmfmac: brcmf_fw_alloc_request: using cypress/cyfmac54591-pcie for chip
BCM4355/13
[  553.125690] brcmfmac 0001:01:00.0: Direct firmware load for cypress/cyfmac54591-
pcie.nvidia,p3737-0000+p3701-0000.bin failed 2
[  553.138518] brcmfmac 0001:01:00.0: Falling back to sysfs fallback for: cypress/
cyfmac54591-pcie.nvidia,p3737-0000+p3701-0000.n
[  553.162006] brcmfmac: brcmf_fw_request_firmware: no board-specific nvram available
(ret=-2), device will use cypress/cyfmac54t
[  553.759557] brcmfmac: brcmf_fw_alloc_request: using cypress/cyfmac54591-pcie for chip
BCM4355/13
[  553.789000] brcmfmac: brcmf_c_preinit_dcmds: Firmware: BCM4355/13 wl0: Jan  3 2023 19:52:2
6 version 13.35.294 (38f3803 CY) FW4
```

CYW55573-PCIE

```
[  153.347313] compat: loading out-of-tree module taints kernel.
[  153.353408] compat: module verification failed: signature and/or required key missing -
tainting kernel
[  153.364267] Loading modules backported from Linux version v5.15.58-2023_0222-0-
gd4d1a61163b5
[  153.364269] Backport generated by backports.git v5.15.58-1-0-g42a95ce7
[  153.524430] usbcore: registered new interface driver brcmfmac
[  153.525234] brcmfmac 0001:01:00.0: Adding to iommu group 9
[  153.525857] brcmfmac 0001:01:00.0: enabling device (0000 -> 0002)
[  153.643314] brcmfmac: brcmf_fw_alloc_request: using cypress/cyfmac55572-pcie for chip
BCM55560/1
[  153.643482] brcmfmac 0001:01:00.0: Direct firmware load for cypress/cyfmac55572-
pcie.t.nvidia,p3737-0000+p3701-0000rxse failed with error -2
[  153.657568] brcmfmac 0001:01:00.0: Falling back to sysfs fallback for: cypress/
cyfmac55572-pcie.t.nvidia,p3737-0000+p3701-0000rxse
[  153.685558] brcmfmac: brcmf_fw_request_firmware: no board-specific nvram available
(ret=-2), device will use cypress/cyfmac55572-pcie.txt
[  154.490451] brcmfmac: brcmf_fw_alloc_request: using cypress/cyfmac55572-pcie for chip
BCM55560/1
```

```
[  154.510507] brcmfmac: brcmf_c_preinit_dcmds: Firmware: BCM55560/1 wl0: Feb 21 2023 07:58:34 version 18.53.180.7 (e1302c0) FWID 01-a35386f0
```

Trademarks
All referenced product or service names and trademarks are the property of their respective owners.

Edition yyyy-mm-dd

Published by

Infineon Technologies AG

81726 Munich, Germany

Do you have a question about this document?

Email: erratum@infineon.com

Document reference
AppNote Number

IMPORTANT NOTICE

The information contained in this application note is given as a hint for the implementation of the product only and shall in no event be regarded as a description or warranty of a certain functionality, condition or quality of the product. Before implementation of the product, the recipient of this application note must verify any function and other technical information given herein in the real application. Infineon Technologies hereby disclaims any and all warranties and liabilities of any kind (including without limitation warranties of non-infringement of intellectual property rights of any third party) with respect to any and all information given in this application note.

The data contained in this document is exclusively intended for technically trained staff. It is the responsibility of customer's technical departments to evaluate the suitability of the product for the intended application and the completeness of the product information given in this document with respect to such application.

For further information on the product, technology, delivery terms and conditions and prices please contact your nearest Infineon Technologies office (**www.infineon.com**).

WARNINGS

Due to technical requirements products may contain dangerous substances. For information on the types in question please contact your nearest Infineon Technologies office.

Except as otherwise explicitly approved by Infineon Technologies in a written document signed by authorized representatives of Infineon Technologies, Infineon Technologies' products may not be used in any applications where a failure of the product or any consequences of the use thereof can reasonably be expected to result in personal injury.