To communicate with a laptop to a BananaPi-R2 the laptop needs an Ethernet port and a cable needs to be connected between the BananaPi-R2 and thelaptop’s Ethernet port.

1. The laptop will need to have the IP address 192.168.2.10 and the BananaPi address is 192.168.2.2 or 192.168.2.1

Graphical user interface

Description automatically generated

1. Connect the Ethernet cable to the First LAN port on the left (not the Wan port). The blue circle.

A picture containing text, electronics

Description automatically generated

1. Use PuTTY to connect to the BananaPi, using SSH 192.168.2.2 port 22.

Graphical user interface, application

Description automatically generated

This is the SD card slot:

A picture containing text, file

Description automatically generated

# Creating a Linux image of the BananaPi-R2 eMMC in a SDcard

1. Get a 32Gbytes micro SDcard (I think 16Gbytes should also do, as any SDcard with higher capacity than 32Gbytes) and place the SDcard in the slot of the BananaPi-R2.

There is a small slot on the right side of the box, the bright pins in the SDcard should face up. It's a spring, insert until it clicks and then just let go. To remove press until clicks and it just comes out.

1. Execute the command "**lsblk**".

In my system it gives:

NAME         MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT

mmcblk0      179:0    0   30G  0 disk

mmcblk1      179:8    0  7.3G  0 disk

├─mmcblk1p1  179:9    0  256M  0 part /boot

└─mmcblk1p2  179:10   0    7G  0 part /

mmcblk1boot0 179:16   0    4M  1 disk

mmcblk1boot1 179:24   0    4M  1 disk

mmcblk1rpmb  179:32   0  512K  0 disk

Identify the SDcard, in this case because it's a 32Gbytes SDcard it's **mmcblk0** since the SIZE is 30G. **ATTENTION**: selecting the wrong device may destroy the Linux installation in the BananaPi.

1. Format the Sdcard by executing "**mkfs -V /dev/mmcblk0**". This should take some minutes.
2. Mount the SDcard by executing "**mount /dev/mmcblk0 /mnt/slot**".
3. Copy the eMMC to the SDcard by executing "**dd if=/dev/mmcblk1 of=/mnt/slot/eMMC.img bs=4096**". This should take over 30 minutes.
4. Verify that a file called eMMC.img is now present in the SDcard by executing "**ls -al /mnt/slot**".

The output should be something like this:

total 7642436

drwxr-xr-x 3 root root       4096 Nov 22 16:28 .

drwxr-xr-x 3 root root       4096 Nov 22 15:55 ..

**-rw-r--r-- 1 root root 7818182656 Nov 22 16:56 eMMC.img     <<----**

drwx------ 2 root root      16384 Nov 22 16:02 lost+found

1. Unmount the SDcard by executing "**umount /mnt/slot**".
2. You can now remove the SDcard from the slot.

I put here just the commands BUT be very sure that the SDcard is **mmcblk0** otherwise the eMMC may be destroyed!

**lsblk**

**------------------------------ Check that the SDcard is mmcblk0**

**mkfs -V /dev/mmcblk0**

**mount /dev/mmcblk0 /mnt/slot**

**dd if=/dev/mmcblk1 of=/mnt/slot/eMMC.img bs=4096**

**ls -al /mnt/slot**

**------------------------------ Check that the eMMC.img file is present!**

**umount /mnt/slot**

# Response from BananaPi.com

The operation process:

1. Find one 128GB USB Disk, to store image that dump from the EMMC device.

2. Boot R2 board from SD Card with SD card boot image.

3. Mount the USB disk to one fold, enter the mounted fold.

4. Dump the emmc device’s data into mounted fold with linux’s dd command. If your EMMC is very big, you need wait for one long time.

5. After dump complete, Umount the mounted folder, power off the R2 board, unplug the USB disk.

6. Then you may insert the USB disk into the New R2 board, The new R2 board still boot from SD card.

7. Then you flash the new board with USB Disk’s image, the flash action still implement with linux’s dd commnader.