BananaPi R2

- Hardware
  - Debug-UART
  - power-on (10 second button-press)
  - additional Hardware:
    - Power Supply: 12V/2A recommended, Powerplug 5.5 mm/2.1 mm Reichelt
    - microSD-Card (>=8GB): Sandisk 32GB Ultra
    - WLAN-antennas: 5dbi with rp-SMA To IPEX/u-FL cable
    - PCIe Wlan-Card: Forum ⇒ MT76
  - emmc: Forum
- Software
  - OS-Images:
    - official: GoogleDrive dev.banana-pi.org.cn
    - my own (Debian/Ubuntu): GoogleDrive
    - Debian | Ubuntu
    - LEDE / OpenWRT
    - Ubuntu forum
    - CentOS Forum
  - Kernel/Uboot (debian):
    - 4.4.70 on official GitHub
    - 4.14+ on my GitHub
    - official 4.14
    - precompiled Kernel on my GDrive
  - Networkconfiguration
  - GPIO
  - WLAN / Bluetooth
  - HW-NAT 4.9/4.14 / NF-Tables 5.12+
  - VLAN-Support
  - CryptoDev
  - HDMI
  - IR
  - irq handling
- help / documentation:
  - GitBook: BPI-R2 GitBook
  - Schematics: GoogleDrive
  - Datasheet: v1.0 v1.1 (GoogleDrive)
  - Forum EN
- links

known Issues / Limitation

- on-board-WLAN (AP-Mode working so far),4.14 + 4.19 + 5.x
- bluetooth working (beta) in 4.14 + 4.19 (except btle)
- BPi-R2 does not power-off: GitHub (working in 4.14+4.19+5.x)
- Switch 7530 Port-separation/vlan in 4.4.x not supported, in 4.14+ vlan+separation works
- 2nd GMAC works on 4.14+4.19, in 5.x this is only available on phylink-branches,5.10-gmac has dsa patches for second gmac,above gmac2 can be used with vlan aware bridge

From:
https://wiki.fw-web.de/ - **FW-WEB Wiki**

Permanent link:
https://wiki.fw-web.de/doku.php?id=en:bpi-r2:start

Last update: **2023/06/08 17:06**

Bluetooth

blutooth should work in 4.14 and 4.19 except BTLE

http://forum.banana-pi.org/t/bpi-r2-new-image-ubuntu-16-04-v1-2-1-bt-and-wifi-ap-mode-are-working-fine-2017-11-27/4291

http://forum.banana-pi.org/t/bpi-r2-kernel-bluetouh-module/4592/46

```
root@bpi-r2-ubuntu:~# apt-get install bluez
# #enable combo-chip via wmt-tools
root@bpi-r2-ubuntu:~# wmt_loader
root@bpi-r2-ubuntu:~# stp_uart_launcher -p /etc/firmware &
root@bpi-r2-ubuntu:~# hciconfig -a
hci0:   Type: Primary  Bus: SDIO
        BD Address: 00:00:00:00:00:00  ACL MTU: 0:0  SCO MTU: 0:0
        DOWN
        RX bytes:0 acl:0 sco:0 events:0 errors:0
        TX bytes:0 acl:0 sco:0 commands:0 errors:0
        Features: 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00
        Packet type: DM1 DH1 HV1
        Link policy:
        Link mode: SLAVE ACCEPT #<<<<<< wrong mode for connect to client-
devices
root@bpi-r2-ubuntu:~# hciconfig -a hci0 lm master
root@bpi-r2-ubuntu:~# hciconfig -a
hci0:   Type: Primary  Bus: SDIO
        BD Address: 00:00:00:00:00:00  ACL MTU: 0:0  SCO MTU: 0:0
        DOWN
        RX bytes:0 acl:0 sco:0 events:0 errors:0
        TX bytes:0 acl:0 sco:0 commands:0 errors:0
        Features: 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00
        Packet type: DM1 DH1 HV1
        Link policy:
        Link mode: MASTER #<<<<<<<<<<<<<< now we are in master-mode
root@bpi-r2-ubuntu:~# hciconfig hci0 up
root@bpi-r2-ubuntu:~# hciconfig -a
hci0:   Type: Primary  Bus: SDIO
        BD Address: 00:00:46:85:90:01  ACL MTU: 1021:4  SCO MTU: 184:1
        UP RUNNING
        RX bytes:554 acl:0 sco:0 events:27 errors:0
        TX bytes:386 acl:0 sco:0 commands:27 errors:0
        Features: 0xbf 0x3e 0x8d 0xfe 0xdb 0xff 0x5b 0x87
        Packet type: DM1 DM3 DM5 DH1 DH3 DH5 HV1 HV2 HV3
        Link policy: RSWITCH SNIFF
        Link mode: MASTER
        Name: 'MTK MT8590 #1'
        Class: 0x001f00
        Service Classes: Unspecified #<<<<<<<<<<<<<<<<<<<<<<<<
```

```
        Device Class: Invalid Device Class! #<<<<<<<<<<<<<<<<<<<<<<<<
        HCI Version: 4.0 (0x6)  Revision: 0x0
        LMP Version: 4.1 (0x7)  Subversion: 0x0
        Manufacturer: MediaTek, Inc. (70)
root@bpi-r2-ubuntu:~# hciconfig hci0 class ff0104 #try to fix class
 root@bpi-r2-ubuntu:~# hciconfig -a | grep Class
        Class: 0xff0104
        Service Classes: Positioning, Networking, Rendering, Capturing,
Object Transfer, Audio, Telephony, Information
        Device Class: Computer, Desktop workstation
root@bpi-r2-ubuntu:~# hciconfig hci0 features
hci0:   Type: Primary  Bus: SDIO
        BD Address: 00:00:46:85:90:01  ACL MTU: 1021:4  SCO MTU: 184:1
        Features page 0: 0xbf 0x3e 0x8d 0xfe 0xdb 0xff 0x5b 0x87
                <3-slot packets> <5-slot packets> <encryption> <slot offset>
                <timing accuracy> <role switch> <sniff mode> <RSSI>
                <channel quality> <SCO link> <HV2 packets> <HV3 packets>
                <CVSD> <power control> <transparent SCO> <broadcast encrypt>
                <EDR ACL 2 Mbps> <EDR ACL 3 Mbps> <enhanced iscan>
                <interlaced iscan> <interlaced pscan> <inquiry with RSSI>
                <extended SCO> <EV4 packets> <EV5 packets> <AFH cap. slave>
                <AFH class. slave> <LE support> <3-slot EDR ACL>
                <5-slot EDR ACL> <sniff subrating> <pause encryption>
                <AFH cap. master> <AFH class. master> <EDR eSCO 2 Mbps>
                <EDR eSCO 3 Mbps> <3-slot EDR eSCO> <extended inquiry>
                <LE and BR/EDR> <simple pairing> <encapsulated PDU>
                <non-flush flag> <LSTO> <inquiry TX power> <EPC>
                <extended features>
        Features page 1: 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00
root@bpi-r2-ubuntu:~# hcitool -i hci0 scan
Scanning ...
        B4:9D:0B:5F:2A:B9       Aquaris X5 Plus
root@bpi-r2-ubuntu:~# hciconfig hci0 piscan #make r2 visible to other
devices
root@bpi-r2-ubuntu:~# hcitool cc B4:9D:0B:5F:2A:B9
root@bpi-r2-ubuntu:~# hcitool auth B4:9D:0B:5F:2A:B9
HCI authentication request failed: Connection timed out
root@bpi-r2-ubuntu:~# l2ping B4:9D:0B:5F:2A:B9
Ping: B4:9D:0B:5F:2A:B9 from 00:00:46:85:90:01 (data size 44) ...
44 bytes from B4:9D:0B:5F:2A:B9 id 0 time 5.33ms
44 bytes from B4:9D:0B:5F:2A:B9 id 1 time 14.76ms
44 bytes from B4:9D:0B:5F:2A:B9 id 2 time 46.78ms
44 bytes from B4:9D:0B:5F:2A:B9 id 3 time 22.62ms
^C4 sent, 4 received, 0% loss
```

nothing happens on my mobile-phone during connect and auth. if i try to connect from mobile-phone to r2, i got a pin-request but i have no pin defined...hcid.conf does not exist with bluez4

issues

Journalcontrol shows error

```
$ journalctl | grep bluetooth
bluetoothd[462]: a2dp-source profile connect failed for 18:F0:E4:FE:F5:58:
Protocol not available
```

this is maybe caused by a bluez5-bug

```
$nano /etc/systemd/system/dbus-org.bluez.service
append: ExecStart=/usr/lib/bluetooth/bluetoothd --compat (or -C)
$systemctl daemon-reload
$systemctl restart bluetooth
```

thanks to user „yenseaman“ from bananapi-forum

From:
https://wiki.fw-web.de/ - **FW-WEB Wiki**

Permanent link:
https://wiki.fw-web.de/doku.php?id=en:bpi-r2:bluetooth

Last update: **2023/06/08 17:06**

centos

https://www.lucas-nussbaum.net/blog/?p=385

https://wiki.centos.org/SpecialInterestGroup/AltArch/armhfp

From:
https://wiki.fw-web.de/ - **FW-WEB Wiki**

Permanent link:
https://wiki.fw-web.de/doku.php?id=en:bpi-r2:centos

Last update: **2023/06/08 17:06**

Cryptodev

http://forum.banana-pi.org/t/is-it-possible-to-have-the-crypto-extensions-working/4034/11

integrated to Main-Branch of my 4.14-Kernel: https://github.com/frank-w/BPI-R2-4.14/tree/main

From:
https://wiki.fw-web.de/ - **FW-WEB Wiki**

Permanent link:
https://wiki.fw-web.de/doku.php?id=en:bpi-r2:cryptodev

Last update: **2023/06/08 17:06**

Debian

debootstrap

https://blog.night-shade.org.uk/2013/12/building-a-pure-debian-armhf-rootfs/

ubuntu

in jessie/Ubuntu 14.x (64bit) there are problems in second-stage: link, here debootstrap must be updated (first stage must be called again): download

```
sudo apt-get install qemu-user-static debootstrap binfmt-support
distro=buster
arch=armhf
#for bpi-r64 use arch=arm64
targetdir=$(pwd)/debian_${distro}_${arch}
mkdir $targetdir
sudo debootstrap --arch=$arch --foreign $distro $targetdir
sudo cp /usr/bin/qemu-arm-static $targetdir/usr/bin/
#for r64 use
# sudo cp /usr/bin/qemu-aarch64-static $targetdir/usr/bin/
sudo cp /etc/resolv.conf $targetdir/etc
sudo distro=$distro chroot $targetdir
export LANG=C
/debootstrap/debootstrap --second-stage
```

like it's done in source hostname,apt (local Server),...must be configured

- hostname (/etc/hostname)

```
bpi-r2
```

- /etc/apt/sources.list

```
deb http://ftp.us.debian.org/debian buster main contrib non-free
deb-src http://ftp.us.debian.org/debian buster main contrib non-free
deb http://ftp.us.debian.org/debian buster-updates main contrib non-free
deb-src http://ftp.us.debian.org/debian buster-updates main contrib non-free
deb http://security.debian.org/debian-security buster/updates main contrib
non-free
deb-src http://security.debian.org/debian-security buster/updates main
contrib non-free
```

- fstab (boot,root)

```
# <file system>       <dir>    <type>    <options>        <dump>    <pass>
/dev/mmcblk0p1      /boot    vfat    errors=remount-ro    0    1
```

```
/dev/mmcblk0p2      /    ext4    defaults        0    0
```

- Dns-server in /etc/resolv.conf

```
nameserver 192.168.0.5
```

- Network-Configuration

```
auto eth0
iface eth0 inet manual
  pre-up ip link set $IFACE up
  post-down ip link set $IFACE down

auto lan0
iface lan0 inet static
  hwaddress ether 08:00:00:00:00:00 # if you want to set MAC manually
  address 192.168.0.10
  netmask 255.255.255.0
  gateway 192.168.0.5
  pre-up ip link set $IFACE up
  post-down ip link set $IFACE down
...
```

in existing Jessie-Image clear root-partition (except lib/modules/<kernelversion>) and put contents of Bootstrap-folder (debian_stretch) to it

Uploaded work so far on my gdrive

Image (sd+emmc in separate img's): gdrive

(deb_stretch_...)

on EMMC-Image SSH-Login is not possible as root because of missing option in ssh-server

add the following line to /etc/ssh/sshd_config and reboot/restart sshd:

```
PermitRootLogin yes
```

change working Image

- clear logs (echo -n „“>logfile),
- delete backup-files,
- clear cache (e.g. APT),
- remove unused Kernel-Modules (/lib/modules/)
- delete own Users
- reset PW for root

fill free space in the Image with null-file (for better packing):

```
loopdev=$(losetup -f) #get first free loopdevice
losetup ${loopdev} /path/to/file
```

```
partprobe ${loopdev}
mount ${loopdev}p2 /mnt

#if you want to install/update things
cp /usr/bin/qemu-arm-static /mnt/usr/bin/
chroot /mnt

#overwrite free space
dd if=/dev/zero of=/mnt/null.dat
rm /mnt/null.dat

losetup -d ${loopdev}
```

https://softwarebakery.com/shrinking-images-on-linux

alternative https://wiki.debian.org/DebianInstaller/Arm/OtherPlatforms

my Image for BPI-R2

network

you have ports wan, lan0-lan4 with kernel 4.14+

current config can be displayed using ip a or by editing /etc/network/interfaces

```
wan:dhcp # maybe needs additional "auto wan" to go up on boot
lan0: 192.168.0.10/24
lan1+lan2 are in a bridge with address 192.168.40.1/24 #currently also auto
lanx+manual missing
lan3 unconfigured
```

to add default-gateway / DNS server do this

```
ip route add default via <your router ip>
nano /etc/resolv.conf #change ip of your gateway for DNS
```

keyboard

for configuring keyboard you have to install console-setup then you will be prompted for keyboard-layout

```
apt-get install console-setup
```

From:
https://wiki.fw-web.de/ - **FW-WEB Wiki**

Permanent link:
https://wiki.fw-web.de/doku.php?id=en:bpi-r2:debian

Last update: **2023/06/08 17:06**

BananaPi R2 - Debug-UART

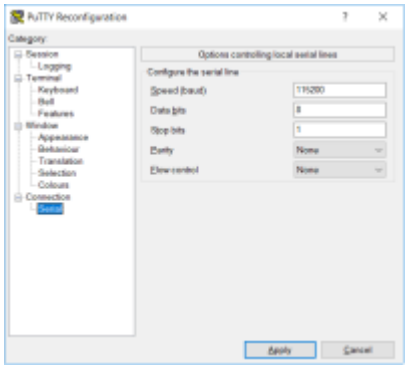
- USB2Serial-Adapter (e.g. CP2102 or FTDI, known problems with Profilic- and ch340g-Chipsets)
- using Uart-Pins (not 40-pin-connector)
- each TX ⇒ RX (r2 tx to rx of usb2serial, tx of usb2serial to r2 rx)
- application for PC:
    - Linux: minicom
    - Windows: putty
- settings: 115200 8N1 FlowControl: off

micro-usb-cp2102-adapter:
https://www.ebay.de/itm/CP2102-MICRO-USB-to-UART-TTL-Module-6-Pin-Serial-Converter-STC-Replace-FT232/401269171476

configurating Putty

Windows-driver for CP2102:
https://www.silabs.com/products/development-tools/software/usb-to-uart-bridge-vcp-drivers

cp210x_universal_windows_driver.zip



reg-file Putty-session (com-port must be changed):

bpi-r2-serial-putty-reg.zip

configurating minicom

first get name of device:

```
grep ttyUSB /var/log/syslog
Oct 15 12:44:59 Frank-Laptop kernel: [ 5113.456306] usb 3-1: cp210x
converter now attached to ttyUSB0
```

```
ls -l /dev/ttyUSB0
crw-rw---- 1 root dialout 188, 0 Mai 10 15:12 /dev/ttyUSB0
#add current user to group dialout (for access to tty-device)
sudo adduser $USER dialout
#login/logout needed to join the group
#show groups of the user
groups
frank adm dialout cdrom sudo dip plugdev lpadmin sambashare
```

Install minicom & setup

```
sudo apt-get install minicom
sudo minicom -s
```

```
          +-----[configuration]------+
          | Filenames and paths      |
          | File transfer protocols  |
          | Serial port setup        |  <<<<<<<<<<<<
          | Modem and dialing        |
          | Screen and keyboard      |
          | Save setup as dfl        |
          | Save setup as..          |
          | Exit                     |
          | Exit from Minicom        |
          +--------------------------+
```

```
+-----------------------------------------------------------------------+
| A -    Serial Device      : /dev/ttyUSB0                              |
| B - Lockfile Location     : /var/lock                                 |
| C -   Callin Program      :                                           |
| D -  Callout Program      :                                           |
| E -    Bps/Par/Bits       : 115200 8N1                                |
| F - Hardware Flow Control : No                                        |
| G - Software Flow Control : No                                        |
|                                                                       |
|    Change which setting?                                              |
+-----------------------------------------------------------------------+
```

- „save as .dfl“
- „Exit from Minicom“

now start minicom as user (without sudo)

exit with Ctrl+a,q or Ctrl+a,x

From:
https://wiki.fw-web.de/ - **FW-WEB Wiki**

Permanent link:
https://wiki.fw-web.de/doku.php?id=en:bpi-r2:debug-uart

Last update: **2023/06/08 17:06**

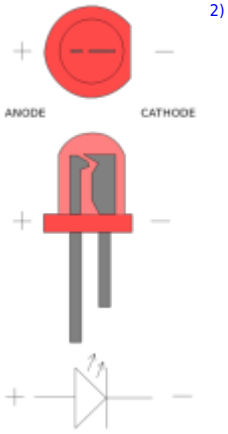
GPIO

Basics

https://wiki.openwrt.org/doc/hardware/port.gpio

1)

2)



Pin define

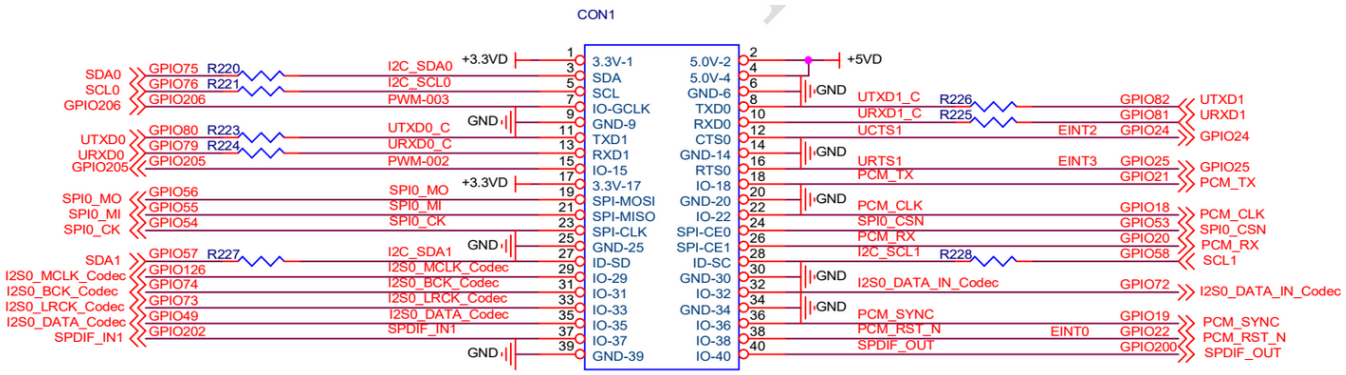


image taken from bpi-r2 schematics

Sparefunction	Mainfunction	pin#	xxxxxxxxxxxxxxxxxxxx	pin#	Mainfunction	Sparefunction
-	3V3	1	1 2	2	5V	-
I2C_SDA0	GPIO 75	3	3 4	4	5V	-
I2C SCL0	GPIO 76	5	5 6	6	GND	-
PWM3	GPIO 206	7	7 8	8	GPIO 82	UART1 TX
-	GND	9	9 10	10	GPIO 81	UART1 RX
UART0 TX	GPIO 80	11	11 12	12	UCTS1 / INT2	GPIO 24 (*)
UART0 RX	GPIO 79	13	13 14	14	GND	-
PWM2	GPIO 205	15	15 16	16	GPIO 25 / INT3	URTS1
-	3V3	17	17 18	18	GPIO 21	PCM_TX
SPI0_MO	GPIO 56	19	19 20	20	GND	-
SPI0_MI	GPIO 55	21	21 22	22	GPIO 18	PCM_CLK
SPI0_CK	GPIO 54	23	23 24	24	GPIO 53	SPI0_CSN
-	GND	25	25 26	26	GPIO 20	PCM_RX
I2C_SDA1	GPIO 57	27	27 28	28	GPIO 58	I2C_SCL1
	GPIO 126	29	29 30	30	GND	-
I2S0_BCK	GPIO 74	31	31 32	32	GPIO 72	I2S0_DATA_IN
I2S0_LRCK	GPIO 73 (?)	33	33 34	34	GND	-
I2S0_DATA	GPIO 49 (M)	35	35 36	36	GPIO 19	PCM_SYNC
SPDIF_IN1	GPIO 202	37	37 38	38	INT0	GPIO 22 (*) / PCM_RST_IN
-	GND	39	39 40	40	GPIO 200	SPDIF_OUT

	main	spare
Pin 1	3V3	-
Pin 2	5V	-
Pin 3	GPIO 75	I2C_SDA0
Pin 4	5V	-
Pin 5	GPIO 76	I2C SCL0
Pin 6	GND	-
Pin 7	GPIO 206	PWM3
Pin 8	GPIO 82	UART1 TX
Pin 9	GND	-
Pin 10	GPIO 81	UART1 RX
Pin 11	GPIO 80	UART0 TX
Pin 12	Int2	GPIO 24 (*) / UCTS1
Pin 13	GPIO 79	UART0 RX
Pin 14	GND	-
Pin 15	GPIO 205	PWM2
Pin 16	GPIO 25 / Int3	URTS1

	main	spare
Pin 17	3V3	-
Pin 18	GPIO 21	PCM_TX
Pin 19	GPIO 56	SPI0_MO
Pin 20	GND	-
Pin 21	GPIO 55	SPI0_MI
Pin 22	GPIO 18	PCM_CLK
Pin 23	GPIO 54	SPI0_CK
Pin 24	GPIO 53	SPI0_CSN
Pin 25	GND	-
Pin 26	GPIO 20	PCM_RX
Pin 27	GPIO 57	I2C_SDA1
Pin 28	GPIO 58	I2C_SCL1
Pin 29	GPIO 126	
Pin 30	GND	-
Pin 31	GPIO 74	I2S0_BCK
Pin 32	GPIO 72	I2S0_DATA_IN
Pin 33	GPIO 73 (?)	I2S0_LRCK
Pin 34	GND	-
Pin 35	GPIO 49 (M)	I2S0_DATA
Pin 36	GPIO 19	PCM_SYNC
Pin 37	GPIO 202	SPDIF_IN1
Pin 38	INT0	GPIO 22 (*) / PCM_RST_IN
Pin 39	GND	-
Pin 40	GPIO 200	SPDIF_OUT

```
(*) special GPIO need memory-patch and mode-set in 4.4.70
(?) currently not working
(M) mode-setting needed
```

Kernel 4.4.70

Access standard GPIO

```
root@bpi-r2:~# GPIO=/sys/devices/platform/1000b000.pinctrl/mt_gpio
root@bpi-r2:~# echo "mode 25 0" >$GPIO #not needed for every GPIO
root@bpi-r2:~# echo "dir 25 1" >$GPIO
root@bpi-r2:~# echo "out 25 1" >$GPIO
```

works with LED on Pin 14 (-) and Pin 16 (+), incl. resistor (220 Ohm)

Access special GPIO

for the GPIOs 22/(23??)/24 it is neccessary to set a register (siehe issue#17 comment #15)

mwrite

```
root@bpi-r2:~# ./mwrite /dev/mem 0x10005b10 0x00000038
./mwrite offset : 10005b10, val : 00000038
b6f03b10
root@bpi-r2:~# GPIO=/sys/devices/platform/1000b000.pinctrl/mt_gpio
root@bpi-r2:~# echo "dir 24 1" >$GPIO
root@bpi-r2:~# echo "out 24 1" >$GPIO
root@bpi-r2:~# echo "mode 24 0" >$GPIO
```

for GPIO24 (pin 12) i have to set mode to 0

Kernel 4.14

GPIO_SYSFS and CONFIG_DEBUG_GPIO must be set in Kernel-Config (.config)

Access standard GPIO

```
root@bpi-r2# mount -t debugfs none /sys/kernel/debug
root@bpi-r2# cat /sys/kernel/debug/pinctrl/1000b000.pinctrl/gpio-ranges
GPIO ranges handled:
0: 1000b000.pinctrl GPIOS [232 - 511] PINS [0 - 279] #base=232, first value
of GPIOS
root@bpi-r2# GPIO_NO=$((232+25))  #base + number of gpio
root@bpi-r2# echo $GPIO_NO
257
root@bpi-r2# echo $GPIO_NO > /sys/class/gpio/export
```

Pin 14=GND/16=GPIO25 (+)

GPIO as Output

```
root@bpi-r2# echo out > /sys/class/gpio/gpio${GPIO_NO}/direction
root@bpi-r2# echo 1 > /sys/class/gpio/gpio${GPIO_NO}/value
root@bpi-r2# echo 0 > /sys/class/gpio/gpio${GPIO_NO}/value
```

used for LED+resistor (220 Ohm) on Pin 14=GND/16=GPIO25 (+)

GPIO as Input

now try with high-active button-circuit on GPIO 200 (pin 40 between button and resistor, using pin 39 as GND [resistor] and pin 17 as 3v3-vcc)

```
[10:54] root@bpi-r2:~# echo in > /sys/class/gpio/gpio${GPIO_NO}/direction
[10:56] root@bpi-r2:~# cat /sys/class/gpio/gpio${GPIO_NO}/value
```

```
0 #button not pressed
[10:56] root@bpi-r2:~# cat /sys/class/gpio/gpio${GPIO_NO}/value
1 #button pressed
[10:56] root@bpi-r2:~# cat /sys/class/gpio/gpio${GPIO_NO}/value
0 #button not pressed
#check every 1/4s
watch -n 0.25 cat /sys/class/gpio/gpio${GPIO_NO}/value
```

Special GPIO

memory-hack (like in 4.4.70) not needed

here example for GPIO24 (pin12):

```
root@bpi-r2# GPIO_NO=$((232+24))
root@bpi-r2# echo $GPIO_NO > /sys/class/gpio/export
root@bpi-r2# echo out > /sys/class/gpio/gpio${GPIO_NO}/direction
root@bpi-r2# echo 1 > /sys/class/gpio/gpio${GPIO_NO}/value
```

LED becomes on :)

on-board LEDs

on-board leds used here are near the power-socket (not next to gpio)

http://forum.banana-pi.org/t/control-on-board-leds/4287/13

on ⇒

```
echo 1 > /sys/class/leds/bpi-r2:isink:green/brightness
```

off ⇒

```
echo 0 > /sys/class/leds/bpi-r2:isink:green/brightness
```

blink (creates delay_on/off-nodes for frequency) ⇒

```
echo timer > /sys/class/leds/bpi-r2:isink:green/trigger
```

change blink frequency (on and off time in ms) ⇒

```
echo 100 > /sys/class/leds/bpi-r2:isink:green/delay_on
echo 100 > /sys/class/leds/bpi-r2:isink:green/delay_off
```

in my tests, green is blinking by default (red+blue are simply on/off), i don't know how to disable blinking of green led

```
L=/sys/class/leds/bpi-r2\:isink
```

```
echo 0 > $L:red/brightness          #goes off
echo 1 > $L:red/brightness          #goes on
echo 0 > $L:green/brightness        #goes off
echo 1 > $L:green/brightness        #starts blinking
```

UART

change DTS(i)

with Kernel 4.4.x the DeviceTree-sections are missing, these can be copied from a newer Kernel (dtsi). in the *bpi*.dts or *bananapi*.dts set state to enabled

definitions in the mt7623.dtsi:
http://elixir.free-electrons.com/linux/v4.13-rc7/source/arch/arm/boot/dts/mt7623.dtsi

now in bananapi.dts set „status=okay“ for your desired uart

notice that in mt7623.dtsi first must come uart2 then the others, else after uboot-message „Starting Kernel“ no more output is shown

Uart3 can be routed to UCTS2/URTS2. These ports are next to Debug-UART-connector (here)

port preferences

```
#show preferences of serial port (replace ttyS2 with your port or ttyUSB0 if a USB2serial-adapter is used):
stty -F /dev/ttyS2 -a

#set speed to 9600 baud, 8 bits, 1 stop bit, no parity:
stty -F /dev/ttyS2 9600 cs8 -cstopb -parenb

#deactivate processing (character conversion,linebreaks,...)
stty -F /dev/ttyS2 -opost

#raw Modus
stty -F /dev/ttyS2 raw
```

Usage

```
pin 8/10 = uart1 (tx/rx) = 11003000
pin 11/13 = uart0 (tx/rx) = 11002000
```

```
#!/bin/bash
DEV=/dev/ttyS2
#stty -F ${DEV} sane
#stty -F ${DEV} 9600 cs8 -cstopb -parenb -crtscts -echo
```

```
stty -F ${DEV} 9600 cs8 -cstopb -parenb raw -echo

dmesg | grep "ttyS.*MMIO" | sed 's/^\[.*\] \(\d*.*\) at.*$/\1/'

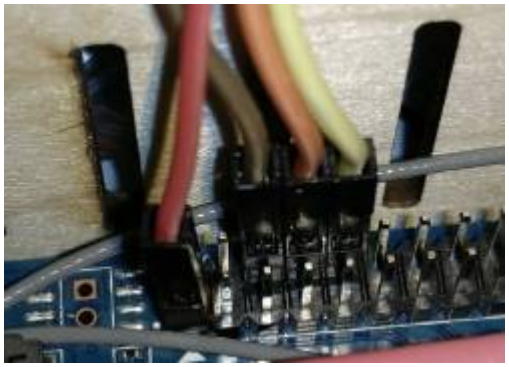
echo "11002000 = uart0 (tx/rx) = pin 11/13"
echo "11003000 = uart1 (tx/rx) = pin 8/10"

echo "using $DEV"
echo "send data using \"echo \"AT\" >$DEV\""

while read line; do
#  if [[ -n "$line" ]]; then
    echo "["$(date "+%Y-%m-%d %H:%M:%S")"] received: "$line
#  fi
done < <(cat $DEV)
```

```
echo "AT" >/dev/ttyS2
```

simple example for Arduino (Nano)



```
PI             Levelshifter       Arduino
1 (3V3)  -------  LV   HV -------- 5V
6 (GND)  ---------  GND  --------- GND
8 (TX)   ------- LV2  HV2 -------- RX
10 (RX)  ------- LV1  HV1 -------- TX
```

PWM

kernel-option PWM_MEDIATEK must be set (module possible), depends on PWM(=y)

using gpio 206 (pin 7) as pwm3

```
echo 3 >/sys/class/pwm/pwmchip0/export
echo 200000 >/sys/class/pwm/pwmchip0/pwm3/period
```

```
echo 100000 >/sys/class/pwm/pwmchip0/pwm3/duty_cycle
echo 1 >/sys/class/pwm/pwmchip0/pwm3/enable
```

https://www.kernel.org/doc/Documentation/pwm.txt

```
period The total period of the PWM signal (read/write). Value is in
nanoseconds and is the sum of the active and inactive time of the PWM.
duty_cycle The active time of the PWM signal (read/write). Value is in
nanoseconds and must be less than the period.
```

```
period=200000ns=200ms=5kHz
duty_cycle=100000ns=1/2 period=50% high + 50% low Signal
```

currently output has the wrong frequency (1kHz instead of 5kHz) see forum and issue

2018-03-02: frequency is now right: Commit in 4.14-main

SPI

http://forum.banana-pi.org/t/bpi-r2-spi-communication/4779/27

I2C

how to add a i2c RTC

```
apt-get install i2c-tools
```

in ubuntu 18.4 you need to add universe to /etc/apt/sources.list

```
[17:13] root@bpi-r2:~# modprobe i2c-dev
[17:14] root@bpi-r2:~# i2cdetect -y 0
```

added an rtc ds1307 (with removed pullups) to i2c0 (I2C_SDA0=pin3, I2C_SCL0=pin5, 5V=pin4, GND=pin6)

```
#!/bin/bash
modprobe i2c-dev
modprobe rtc-ds1307
echo ds1307 0x68 > /sys/class/i2c-adapter/i2c-0/new_device
#cat /sys/class/i2c-dev/i2c-0/device/0-0068/rtc/rtc0/time
#read rtc
hwclock -r
#set system-clock to rtc-value
#hwclock -s
#set rtc to system-time
#hwclock -w
```

1Wire

thanks to user phil from bpi-r2-forum

To enable w1-gpio on GPIO 19 add the following to the main node of mt7623n-bananapi-bpi-r2.dts

```
w1 {
   pinctrl-names = “default”;
   compatible = “w1-gpio”;
   gpios = <&pio 19 0>;
   status = “okay”;
};
```

dts for kernel 4.9 with 1wire-node

add settings to kernel config

```
CONFIG_W1=m
CONFIG_W1_MASTER_GPIO=m
CONFIG_W1_SLAVE_THERM=m
```

and recompile the kernel

Your devices should now show up at /sys/bus/w1/devices/

GPS-PPS

http://mtnstormdaq.com/blog/2012/10/gps-pps-use-as-a-time-reference/

phil on bpi-r2-forum got pps working with kernel 4.9

HW-module

adding this to main-section of mt7623n-bananapi-bpi-r2.dts:

```
pps {
   pinctrl-names = “default”;
   compatible = “pps-gpio”;
   gpios = <&pio 72 0>;
   status = “okay”;
};
```

modified 4.9 dts

and add this options in the kernel-config for GPIO PPS support:

```
CONFIG_PPS=m
```

```
CONFIG_PPS_CLIENT_LDISC=m
CONFIG_PPS_CLIENT_GPIO=m
```

thermal

not really gpio, but i wont spend an own page

```
cat /sys/class/thermal/thermal_zone0/temp
```

http://forum.banana-pi.org/t/lm-sensors-support/4145/30

1)

Quelle: wiki.openwrt.org

2)

Quelle: commons.wikimedia.org

From:
https://wiki.fw-web.de/ - **FW-WEB Wiki**

Permanent link:
https://wiki.fw-web.de/doku.php?id=en:bpi-r2:gpio

Last update: **2023/06/08 17:06**

Hardware

- Debug-UART
- power-on: hold power-switch 10 seconds, without power-switch: Forum
    - solder bridge next to power switch. Can cause problems since poweroff-feature. To fix this blacklist pmic-keys-module or uninstall acpid package see issue 35
    - input 5V to OTG
- poweroff (4.14.53+) needs acpid package for power-button (command works without it)
    - http://forum.banana-pi.org/t/bpi-r2-already-has-an-on-board-rtc/5347/31
- emmc: Forum
    - emmc on v1.2 is maybe from vendor SEC which needs different pinctrl setting (pu/pd) in uboot to be recognized: forum issue was introduced in u-boot 2022.04 and fixed with patch from daniel (for 2023.07)
- wifi-chip onBoard: MT6625 Forum (dev) Forum (usr)
- Switch-Chip onBoard: MT7530 Forum
- GPIO

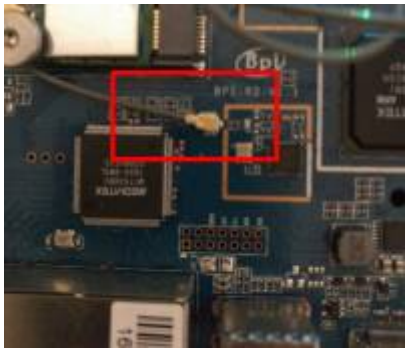
additional hardware

- Power-supply: 12V/2A recommended,plug coaxial: 2,1 x 5,5 mm Reichelt
- microSD-card (>=8GB): Sandisk Ultra 32GB
- WLAN-antennas: 5dBi mit rp-SMA auf IPEX/u-FL Kabel
- PCIe wifi-card: Forum ⇒ MT76
- case:
    - official: alu-case,acrylic case
    - inofficial files for 3D Printer
- heatsinks:
    - mt7623: 20x20mm
    - mt7530: 14x14mm (like rpi soc)

cables

wifi

- rp-SMA to IPEX/u-FL cable for wifi-antennas



SATA

- 2,54mm XH 2pin/4pin for SATA power or 2-Pin combined with SATA

it works with GND next to power (1+2,3+4). 12V is yellow, 5V is red

both jacks near sata-connector have voltage, also if R2 is „halted“, the fan-jack (near IR-receiver) had no direct-current output in my tests (4.14), this seems my gnd there is broken...with another gnd i have the 5V output

battery

- ~~1,25mm SH 6pin for battery~~ AliExpress~~, no circuit known~~ battery-connector does not work see here

mount

- wifi mPCIe-card bolted with M2x10 screws and knurled nut + 2 x washers of Polyamid (isolation+protection of the board)

From:
https://wiki.fw-web.de/ - **FW-WEB Wiki**

Permanent link:
https://wiki.fw-web.de/doku.php?id=en:bpi-r2:hardware

Last update: **2023/06/08 17:06**

HDMI

HDMI works with Kernel 4.16 in some cases (1080p,1280×1024). Driver with fbdev was backported to 4.14 and ported to 4.19.in 5.0 hdmi works but fbdev does not.

Discussion in Forum.

Resolution can be set in BPI-BOOT/bananapi/bpi-r2/linux/uEnv.txt:

```
bootopts=vmalloc=496M debug=7 initcall_debug=0 video=1280x1024-32
```

this resolution will be used for both virtual Console and X-Server.

more about this setting: https://nouveau.freedesktop.org/wiki/KernelModeSetting/

to resize only font on Console this can be done:

```
setfont Uni3-TerminusBold32x16.psf.gz
```

available fonts are in /usr/share/consolefonts/

Thanks to ForumUser Alex R. aka „DeadMeat“

Normally hdmi will be turned off if no monitor is connected (or switched off) while bootup. You can change this with a video-option:

```
video=HDMI-A-1:1280x1024-32@60D
```

This keeps hdmi enabled and allow hotplug,but in my case there is a resolution-issue.

http://forum.banana-pi.org/t/bpi-r2-hdmi-in-uboot-and-linux/4651/123

DRM Debugging

add „drm.debug=0x04“ to bootopts in uEnv.txt (maybe 0x07 optionally with log_buf_len=1M)

LIMA

in 5.4 lima is merged to mainline-kernel and do need only some small changes for r2

but it needs mesa compiled with some small changes too

mesa.patch

```
git clone --single-branch --branch mesa-19.3.2
https://gitlab.freedesktop.org/mesa/mesa.git mesa-source
```

```
cd mesa-source/
git checkout mesa-19.3.2
patch -p1 < mesa.patch
meson ./build --buildtype plain --libdir lib/arm-linux-gnueabihf --
localstatedir /var/lib --prefix /usr --sysconfdir /etc --wrap-mode
nodownload -Dplatforms=surfaceless,x11,drm -Dllvm=true -Dlmsensors=false -
Dlibunwind=false -Dgallium-nine=false -Dgallium-va=false -Dgallium-
vdpau=false -Dgallium-xa=false -Dgallium-xvmc=false -Dgallium-
opencl=disabled -Dosmesa=none -Dbuild-tests=false -Dglx=dri -Dshared-
glapi=true -Ddri3=true -Degl=true -Dgbm=true -Dgles1=false -Dgles2=true -
Dglvnd=false -Dselinux=false -Dvalgrind=false -Ddri-drivers= -Dgallium-
drivers=kmsro,lima
ninja -C build/
sudo ninja -C build/ install
```

important for debian is „–libdir lib/arm-linux-gnueabihf“ with only „lib“ x-server crashes while loading with segmentation fault.

maybe libdrm needs to be compiled too

```
git clone --single-branch --branch libdrm-2.4.100
https://gitlab.freedesktop.org/mesa/drm.git
cd drm
git checkout libdrm-2.4.100
meson build/  --prefix=/usr
#meson --reconfigure  build/ --prefix /usr --libdir lib/arm-linux-gnueabihf
-Damdgpu=false -Dradeon=false -Dnouveau=false
ninja -C build/
sudo ninja -C build/ install
#or install to a folder
mkdir install
DESTDIR=$(pwd)/install ninja -C build/ install
```

usage

while lima initialization hdmi display needs to connected (best at boottime). You can call initialization again by unloading+loading lima module (modprobe -r lima;modprobe lima)

Xorg needs a little adjustment too,else you get this error (/var/log/Xorg.0.log):

```
(EE) Cannot run in framebuffer mode. Please specify busIDs for all
framebuffer devices
```

just create /etc/X11/xorg.conf.d/40-serverflags.conf (there is no /etc/X11/xorg.conf by default) with the following content:

```
Section "ServerFlags"
      Option "AutoAddGPU" "off"
EndSection
```

disable glamore

in case dri crashes on x startup with Sementation fault (because mesa is compiled for another kernel) you can disable it.

first check /var/log/Xorg.0.log for something like this:

```
[ 26.398] (II) Loading sub module "glamoregl"
[ 26.398] (II) LoadModule: "glamoregl"
[ 26.399] (II) Loading /usr/lib/xorg/modules/libglamoregl.so
[ 26.424] (II) Module glamoregl: vendor="X.Org Foundation"
[ 26.424] compiled for 1.20.4, module version = 1.0.1
[ 26.424] ABI class: X.Org ANSI C Emulation, version 0.4
[ 26.829] (EE)
[ 26.830] (EE) Backtrace:
[ 26.830] (EE)
[ 26.830] (EE) Segmentation fault at address 0xbc
[ 26.830] (EE)
Fatal server error:
[ 26.830] (EE) Caught signal 11 (Segmentation fault). Server aborting
[ 26.830] (EE)
[ 26.831] (EE)
```

then create a file named /etc/X11/xorg.conf.d/00-noglamore.conf with following content:

```
Section "Device"
    Identifier "nogpu"
    Driver "modesetting"
    Option "Accelmethod" "none"
EndSection

Section "Module"
    Disable "glamoregl"
EndSection
```

and then restart your display-manager (e.g. service lightdm restart)

source: https://wiki.raptorcs.com/wiki/Troubleshooting/GPU

From:
https://wiki.fw-web.de/ - **FW-WEB Wiki**

Permanent link:
https://wiki.fw-web.de/doku.php?id=en:bpi-r2:hdmi

Last update: **2023/06/08 17:06**

Hardware-NAT

- available in LEDE forum
- seems to work with 4.14, Test: https://github.com/frank-w/BPI-R2-4.14/tree/4.14-hnat forum
- no support in 4.4.70 forum

Current implementation works only between lan and wan (not wifi or other interfaces)

at least you have to setup nat on wan:

```
ipt=/sbin/iptables
if_wan=wan
${ipt} -t nat -F
${ipt} -t nat -A POSTROUTING -o ${if_wan} -j MASQUERADE
```

and load hnat-module:

```
modprobe mtkhnat
```

Check if it is working with

```
#via debugfs (working shows BIND-state)
cat /sys/kernel/debug/hnat/all_entry
#via interrupts (counter does not increase after a while (download))
cat /proc/interrupts | grep 'ethernet'
```

original-Patch for 4.9 (lede)

From:
https://wiki.fw-web.de/ - **FW-WEB Wiki**

Permanent link:
https://wiki.fw-web.de/doku.php?id=en:bpi-r2:hwnat

Last update: **2023/06/08 17:06**

IR

(5.5) kernel has basic support for the IR-receiver

```
CONFIG_MEDIA_SUPPORT=y
CONFIG_MEDIA_RC_SUPPORT=y
CONFIG_RC_CORE=y
CONFIG_RC_DEVICES=y
CONFIG_LIRC=y
CONFIG_IR_MTK=y
CONFIG_RC_DECODERS=y
CONFIG_IR_NEC_DECODER=m
```

to use the IR-receiver you can try this commands

```
apt install evtest
apt install v4l-utils
apt install ir-keytable
```

```
root@bpi-r2:~# ir-ctl -r
root@bpi-r2:~# ir-keytable -s rc0 -p all -t
root@bpi-r2:~# evtest
```

From:
https://wiki.fw-web.de/ - **FW-WEB Wiki**

Permanent link:
https://wiki.fw-web.de/doku.php?id=en:bpi-r2:ir

Last update: **2023/06/08 17:06**

Kernel

Overview

using which Kernel?

Kernel-features-overview on github

compile official kernel

4.4.70 on GitHub

```
sudo apt-get install git make gcc-arm-linux-gnueabihf u-boot-tools
git clone https://github.com/BPI-SINOVOIP/BPI-R2-bsp.git bpi-r2
cd bpi-r2
./build.sh
```

see also Patch for build.sh

- Option 4 kernel-config (menuconfig),
- 1 to compile all,
- 3 compile only Kernel, after that option 6, to copy compiled data to SD-folder

if successful finished, Kernel is in folder SD/BPI-BOOT modules in SD/BPI-ROOT

kernel can be copied to Boot-Partition (path, previous Backup!) the modules must be copied as Root

```
cp SD/BPI-BOOT/bananapi/bpi-r2/linux/uImage /media/$USER/BPI-
BOOT/bananapi/bpi-r2/linux/
sudo cp -r SD/BPI-ROOT/lib/modules/4.4.70-BPI-R2-Kernel /media/$USER/BPI-
ROOT/lib/modules/
```

compile my 4.4 Kernel

i have forked the official kernel-repo and updated to actual kernel-version, added mt76x2 wireless driver and activated uart0+1:

4.4.x on GitHub

steps to build/install are same except the git clone :)

```
sudo apt-get install git make gcc-arm-linux-gnueabihf u-boot-tools
git clone https://github.com/frank-w/BPI-R2-4.4.git bpi-r2
cd bpi-r2
```

```
./build.sh
```

compiled kernel 4.4.x

v1.2.1 (Nov 23, 2017),including mt76x2/x3 wireless driver

Bootloader (u-boot):
https://drive.google.com/file/d/11XQ4n6WYSj7gGPtMwug4SITtYn0GZxOb/view?usp=sharing

https://drive.google.com/drive/folders/1kkFyxeHoskszI7CpSsL6Wi6ROjDrBLOB?usp=sharing

the 4.4-archive (Kernelversion-BPI-R2-Kernel_boot+root.tgz) contains 2 folders, which contents have to be copied to the partiton of SD/MMC with the same name.

- BPI-BOOT ⇒ boot-Partition uImage=Kernel-file, backup first your existing before overwriting it
- BPI-ROOT ⇒ partition with root-fs, here are the kernel-modules (/lib/modules/kernelname). With new kernel-version a new subfolder is created, so you need no backup else the name of kernel is the same.

using own (newer) Kernel

http://forum.banana-pi.org/t/what-s-the-best-practice-to-build-own-kernel/3937

using my github-repo

https://github.com/frank-w/BPI-R2-4.14.git

manual way using kernel.org

```
  sudo apt-get install git make gcc-arm-linux-gnueabihf u-boot-tools
  git clone
https://git.kernel.org/pub/scm/linux/kernel/git/torvalds/linux.git bpi-r2
  cd bpi-r2

  export CROSS_COMPILE=arm-linux-gnueabihf-
  export ARCH=arm
  export INSTALL_MOD_PATH=$(pwd)/mod/
  mkdir $INSTALL_MOD_PATH

  #git reset --hard v4.14
  cp pfad/zu/mt7623n_evb_bpi_defconfig arch/arm/configs/
  make mt7623n_evb_bpi_defconfig
  make

  ...
```

```
  #cat arch/arm/boot/zImage arch/arm/boot/dts/mt7623n-bananapi-bpi-r2.dtb
>zImage_dtb
  cat arch/arm/boot/zImage arch/arm/boot/dts/mt7623n-bananapi-bpi-r2.dtb >
arch/arm/boot/zImage-dtb
  #mkimage -A arm -O linux -C none -T kernel -a 0x80008000 -e 0x80008000 -d
zImage_dtb uImage
  mkimage -A arm -O linux -T kernel -C none -a 80008000 -e 80008000 -n
"Linux Kernel 4.14" -d arch/arm/boot/zImage-dtb ./uImage

  rm -r $INSTALL_MOD_PATH/lib
  make modules_install

  cp uImage /media/$USER/BPI-BOOT/bananapi/bpi-r2/linux/  #<< change
mountpoint
  sudo cp -r $INSTALL_MOD_PATH/lib /media/$USER/BPI-ROOT/ #<< change
mountpoint
```

config from user RyderLee
ramdisk

pcie.patch
for PCIe-Slot (CONFIG_PCIEPORTBUS,CONFIG_PCIEPORTBUS,CONFIG_PCIEPORTBUS have to be activated)

Script to compile/copy
changed defconfig

above 4.14.14 only pcie-patch may be needed..

with kernel 4.4.70 mmc0=sd-card and mmc1=emmc, so i have done this also for kernel 4.14 (else you have to change u-boots uenv.txt and Mount-Points in system)

to do this set mmc1-section above mmc0-section arch/arm/boot/dts/mt7623n-bananapi-bpi-r2.dts and in arch/arm/boot/dts/mt7623.dtsi

mmc-swap.diff

how to use the build.sh

with my github-Repo:

```
sudo apt-get install git make gcc-arm-linux-gnueabihf
git clone https://github.com/frank-w/BPI-R2-4.14.git bpi-r2-4.14
cd bpi-r2-4.14
./build.sh importconfig
./build.sh config #if you want to change the kernel-config
./build.sh
```

compiled 4.14.x

https://drive.google.com/drive/folders/1EGN1TvqCpDHdOAS-mjRg9ipi0kahnOUV?usp=sharing

the 4.14-archive (bpi-r2_kernelversion_gitbranch.tar.gz) contains 2 folders, which contents have to be copied to the partiton of SD/MMC with the same name.

- BPI-BOOT ⇒ boot-Partition, uImage=Kernel-file, backup first your existing uImage before overwriting it
- BPI-ROOT ⇒ partition with root-fs, here are the kernel-modules (/lib/modules/kernelname). With new kernel-version a new subfolder is created, so you need no backup else the name of kernel is the same. The files in the other folders are mainly to get internal wifi working (wmt-tools, firmware, config) and have not to be copied every time (currently not changing). These are only packed in for users who haven't these tools already

there are also binary-packages on github-releases (made by travis-ci) with same file-structure:
https://github.com/frank-w/BPI-R2-4.14/releases

you can use this

script

for installing the tar.gz-package

install kernel

The linux-kernel has 2 parts...the „main-kernel-image“ (uImage) and the modules.

The kernel-image is loaded by uboot from BPI-BOOT-Partition folder bananapi/bpi-r2/linux/. By default the file „uImage“ is loaded, but you can choose another name and set variable „kernel“ in uEnv.txt.

the modules have to be put to BPI-ROOT-partition (/lib/modules/kernelname).

change DTS (device tree)

using dts(i)-nodes (arch/arm/boot/dts/) and porting them to older kernel

- mt7623.dtsi
- mt7623n-bananapi-bpi-r2.dts ⇒ mt7623n-bpi-r2.dts

issues

- evbug will be loaded and displays messages on terminal if local keyboard is connected
  - you can disable via „modprobe -r evbug“ (temporary, will be loaded again after next reboot) or add file /etc/modprobe.d/evbug.conf with content „blacklist evbug“ (survives reboot)

From:
https://wiki.fw-web.de/ - **FW-WEB Wiki**

Permanent link:
https://wiki.fw-web.de/doku.php?id=en:bpi-r2:kernel

Last update: **2023/06/08 17:06**

LEDE (OpenWRT)

- Garys github
- Forum (img)
- Forum (compile)

```
git clone https://github.com/garywangcn/bpi-r2_lede.git
cd bpi-r2_lede/
git checkout bpi-r2-on-lede-v1
make menuconfig
```

here „Target System“ must be set to „MediaTek Ralink ARM“ and in „Boot Loaders“ the entry „u-boot-bpi_r2“ must be selected with a ∗

build with:

```
make -j1 V=s
```

nach dem kompilieren liegen 2 img-Dateien im Ordner build_dir/target-arm_cortex-a7+neon-vfpv4_musl_eabi/linux-mediatek_32/:

```
frank@Frank-Laptop:/media/data_ext/bpi-r2_lede$ ls -lh build_dir/target-arm_cortex-a7+neon-vfpv4_musl_eabi/linux-mediatek_32/*.img
-rw-r--r-- 1 frank frank 67M Jan 16 12:03 build_dir/target-arm_cortex-a7+neon-vfpv4_musl_eabi/linux-mediatek_32/mtk-bpi-r2-EMMC.img
-rw-r--r-- 1 frank frank 67M Jan 16 12:03 build_dir/target-arm_cortex-a7+neon-vfpv4_musl_eabi/linux-mediatek_32/mtk-bpi-r2-SD.img
```

img-files are on my gdrive

Please note that lede currently not support hdmi-output (you will see only a purple screen).

SD

```
  dd if=mtk-bpi-r2-SD.img of=/dev/sdx
```

EMMC

1. Copy EMMC image to a running system which is on SD card
2. start system and look which mmcblk is the emmc: cat /proc/partitions (search for boot0 and this mmcblk [0|1] is the emmc)
3. flash EMMC image to User Data Area of EMMC: dd if=mtk-bpi-r2-EMMC.img of=/dev/mmcblkX
4. Unlock EMMC boot0 block: echo 0 > /sys/block/mmcblkXboot0/force_ro
5. flash preloader to EMMC boot0 block: dd if=mtk-bpi-r2-EMMC.img of=/dev/mmcblkXboot0 bs=1M count=1
6. Change the Partition configurion of EMMC to 48h: reboot the system which is running from SD,

and enter the U-boot-console, run command „emmc pconf 48“
7. Power off, remove SD card, and then power on R2 board.

Network-configuration

By default lan-ports are bridged together with IP-address 192.168.1.1

Further configuration see here: https://wiki.openwrt.org/doc/uci/network

From:
https://wiki.fw-web.de/ - **FW-WEB Wiki**

Permanent link:
https://wiki.fw-web.de/doku.php?id=en:bpi-r2:lede

Last update: **2023/06/08 17:06**

Links

own repos

- https://github.com/frank-w/BPI-R2-4.4
- https://github.com/frank-w/BPI-R2-4.14

official repos

- https://github.com/BPI-SINOVOIP/BPI-R2-bsp
- https://github.com/BPI-SINOVOIP/BPI-R2-bsp-4.14
- https://github.com/garywangcn/bpi-r2_lede

other repos

- https://github.com/abbradar/mt6625l-wlan-gen2 (separated wifi-driver)
- https://github.com/abbradar/openwrt/tree/bpi-upstream/target/linux/mediatek/patches-4.14 (patches)

Mainline-Kernel

- https://www.kernel.org/
- https://git.kernel.org/pub/scm/linux/kernel/git/torvalds/linux.git
- https://git.kernel.org/pub/scm/linux/kernel/git/stable/linux-stable.git/?h=linux-5.15.y
- https://patchwork.kernel.org/project/linux-mediatek/list/

From:
https://wiki.fw-web.de/ - **FW-WEB Wiki**

Permanent link:
https://wiki.fw-web.de/doku.php?id=en:bpi-r2:links

Last update: **2023/06/08 17:06**

Network-boot

Some infos in debian-wiki: https://www.debian.org/releases/stretch/armhf/ch05s01.html.en

Dnsmasq as tftp-server

```
# Enable the TFTP server
enable-tftp
tftp-root=/var/lib/tftp

mkdir /var/lib/tftp


systemctl enable dnsmasq.service
systemctl start dnsmasq.service

or

service dnsmasq restart
```

tftp-boot

- https://rechtzeit.wordpress.com/2013/01/16/tftp-boot-using-u-boot/

```
setenv ipaddr 192.168.0.11
setenv netmask 255.255.255.0
setenv serverip 192.168.0.10
setenv bootfile uImage
setenv bootargs console=ttyS0,115200 root=/dev/mmcblk0p2 rw rootwait ip=dhcp
```

```
tftp 0x80200000 ${bootfile}
bootm 0x80200000
```

uenv.txt (using my uboot + reload bootmenu):

```
ipaddr=192.168.0.11
netmask=255.255.255.0
serverip=192.168.0.10

bootfile=uImage
netbootargs=console=ttyS0,115200 root=/dev/mmcblk0p2 rw rootwait ip=dhcp

bootmenu_1=2. Boot kernel from TFTP.=run bootnet
bootnet=printenv; setenv bootargs ${netbootargs};tftp 0x80200000
${bootfile};bootm
```

this replaces bootmenu-entry #2 with tftp-boot and loads bootfile from tftp-server (serverip)...this part works so far with dnsmasq-tftp

loading and display a file

Fix Me!

needs strings-command (currently untested)

```
tftp ${loadaddr} 192.168.0.10:files.lst
strings ${loadaddr} ${filesize}
```

InitRamFS

Fix Me! Currently untested

Initrd-steps on ibm: https://www.ibm.com/developerworks/library/l-initrd/

to use a initramfs add this option (change filename) to your .config (tested with bitmasters initramfs)

```
CONFIG_INITRAMFS_SOURCE="../initramfs.cpio"
```

IP-configration

add this to your netbootargs:

```
ip=192.168.0.11:192.168.0.10:192.168.0.10:255.255.255.0::wan
```

documentation says

```
ip=<client-ip>:<server-ip>:<gw-
ip>:<netmask>:<hostname>:<device>:<autoconf>:<dns0-ip>:<dns1-ip>:<ntp0-ip>
```

NFS-server

- https://lauri.võsandi.com/2014/08/cubietruck-over-nfs.html
- https://gmelikov.com/2018/02/25/debian-stretch-diskless-pxe-boot/

debootstrap needs exec and dev-options on target-partition, add these to fstab and run „mount -o remount mountpoint“

```
targetdir=bpi-r2-nfs-root
```

```
sudo apt-get install debootstrap qemu-user-static binfmt-support

sudo debootstrap --foreign --arch=armhf stretch $targetdir
sudo cp /usr/bin/qemu*-arm-static $targetdir/usr/bin/
sudo chroot $targetdir /debootstrap/debootstrap --second-stage

sudo mkdir -p $targetdir/dev/pts
sudo mount --bind /dev/pts $targetdir/dev/pts
sudo mount --bind /proc $targetdir/proc/
#sudo mount --bind /home $targetdir/home/

sudo chroot $targetdir

ln -s /proc/self/mounts /etc/mtab

nano /etc/fstab

proc /proc proc defaults 0 0
/dev/nfs / nfs nolock 0 0
none /tmp tmpfs defaults 0 0
none /var/tmp tmpfs defaults 0 0
none /var/lib/lightdm tmpfs defaults 0 0
none /media tmpfs defaults 0 0
none /var/log tmpfs defaults 0 0
#192.168.81.1:/home /home nfs nolock 1 2

nano /etc/apt/sources.list

deb http://deb.debian.org/debian stretch main
deb-src http://deb.debian.org/debian stretch main

deb http://security.debian.org stretch/updates main
deb-src http://security.debian.org stretch/updates main



nano /etc/resolv.conf
nameserver 192.168.0.10

#change root-passwort, without it you cannot login!
passwd

#change hostname
nano /etc/hostname
#add hostname also to loopback-device (127.0.0.1 + ::1)
nano /etc/hosts

apt-get update

#change network-config if you don't want to set it via uboot or dhcp
nano /etc/network/interfaces
```

NFS-Server-install:

```
sudo apt-get install nfs-kernel-server
#status
sudo service nfs-kernel-server status
#create root-folder
mkdir -p /var/nfs/bpi-r2
#add this folder to exports-file
echo "/var/nfs/bpi-r2 192.168.0.0/24(rw,sync,no_root_squash)">>/etc/exports
#unpack the debootstrapped rootfs to it
```

test on client

```
apt install nfs-common
mkdir /mnt/nfs
mount -t nfs 192.168.0.10:/var/nfs/bpi-r2 /mnt/nfs
```

with fstab

```
#entry in /etc/fstab:
192.168.0.10:/var/nfs/bpi-r2 /media/nfs nfs rw 0 0

#create target-dir and mount it
sudo mkdir /media/nfs
sudo mount /media/nfs
```

Uboot:

```
#netboot
ipaddr=192.168.0.11
netmask=255.255.255.0
serverip=192.168.0.10
bootfile=uImage_4.14.73
set_netargs=setenv netargs "console=ttyS0,115200 root=/dev/mmcblk0p2 rw
rootwait
ip=${ipaddr}:${serverip}:${serverip}:${netmask}::wan::${serverip}:8.8.8.8"
nfs_ip=192.168.0.10
nfs_rfs=/var/nfs/bpi-r2
#netargs=console=ttyS0,115200
ip=192.168.0.11:192.168.0.10:192.168.0.10:255.255.255.0::wan::192.168.0.10:8
.8.8.8 root=/dev/nfs rw nfsroot=192.168.0.10:/var/nfs/bpi-r2,nfsvers=3,tcp
nfsrootdebug
set_nfsargs=setenv nfsargs "root=/dev/nfs rw
nfsroot=${nfs_ip}:${nfs_rfs},nfsvers=3,tcp nfsrootdebug"
#new bootmenuentries (start at bootmenu_6) require reload bootmenu because
uenv.txt is not loaded before display menu
bootnet=run set_netargs;run set_nfsargs; setenv bootargs "${netargs}
${nfsargs}"; printenv; tftp 0x80200000 ${bootfile};bootm
bootmenu_1=2. Boot kernel from TFTP.=run bootnet
```

other links

- https://wiki.dave.eu/index.php/Setting_up_tftp_and_nfs
- https://elinux.org/TFTP_Boot_and_NFS_Root_Filesystems
- https://community.nxp.com/docs/DOC-95051
- https://systemausfall.org/wikis/howto/FAI (de)

From:
https://wiki.fw-web.de/ - **FW-WEB Wiki**

Permanent link:
https://wiki.fw-web.de/doku.php?id=en:bpi-r2:netboot

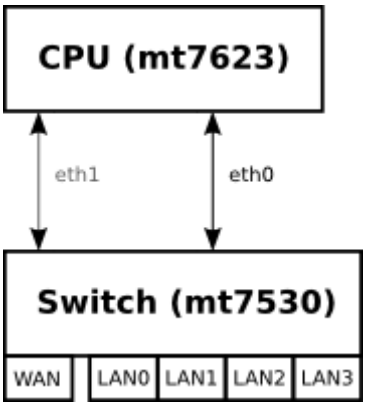
Last update: **2023/06/08 17:06**

Network-Configuration

ip-command needs package iproute2

Configuration on this page is based on debian stretch, should work the same way with Ubuntu needs Kernel 4.14 or above (DSA-driver for Port-separation)

in Kernel 4.14 eth0 is the connection between CPU and the Switch-Circuit (mt7530), on which the Ports wan and lan0-4 are connected. this connection have to be set to „up“ first.



bringing up then cpu-port(s)

```
ip link set eth0 up
ip link set eth1 up
```

or via /etc/network/interfaces

```
auto eth0
iface eth0 inet manual
  pre-up ip link set $IFACE up
  post-down ip link set $IFACE down

auto eth1
iface eth1 inet manual
  pre-up ip link set $IFACE up
  post-down ip link set $IFACE down
```

the mapping of ports to gmac is defined in dts-file and can be shown with „ip a“

With 4.14 >.52 on my repo gmac #2 (eth1) is added and wan is connected to this.

by default each lan-port is separated and needs an own ip-configuration in different subnets

most users like to use all lan-ports in 1 network-segment, so these can be bridged together to make only 1 ip-configuration for „LAN“

my 6.3-rc brings some patches for re-introduce second gmac. by default port 6 (trgmii) is used and can be changed in userspace:

```
ip link set wan type dsa master eth1
```

* requires iproute2 v6.1+

add backports in sources.list:

```
deb http://deb.debian.org/debian bullseye-backports main contrib non-free
```

install

```
apt update
apt -t bullseye-backports install iproute2
```

MAC-Address

The MAC-address can only be set for the GMAC (connection between Switch and CPU). In Kernel 4.14 only 1 GMAC is detected (eth0). There are 2 GMACs in Hardware.

UDEV

here

```
$ cat /etc/udev/rules.d/00-static-mac-address.rules
ACTION=="add", SUBSYSTEM=="net", KERNELS=="1b100000.ethernet",
RUN+="/sbin/ip link set dev %k address ae:fc:de:ad:be:ef"
```

interfaces-file

/etc/network/interfaces

```
iface lan0 inet static
  address 192.168.0.10
  netmask 255.255.255.0
  gateway 192.168.0.5
#  pre-up ip link set $IFACE up
  pre-up ip link set $IFACE address 02:01:02:03:04:08 up
```

using systemd

/etc/systemd/network/10-wan.link

```
[Match]
OriginalName=wan

[Link]
```

```
MACAddress=XX:XX:XX:XX:XX:XX
```

http://forum.banana-pi.org/t/set-mac-address-on-boot/7224/7

device-tree

```
local-mac-address = [00 0a 35 00 00 01];
mac-address = [00 0a 35 00 00 01];
```

http://forum.banana-pi.org/t/set-mac-address-on-boot/7224/4

this can also be used in devicetree-overlays

set via uboot

if devicetree (with mac-address property) is loaded separately (fdt), an alias for ethernet-node is defined and ethaddr-variable is set in uboot this is used in linux

http://forum.banana-pi.org/t/set-mac-address-on-boot/7224/6

interface-name

Ubuntu 18.4+ (and debiaan 10+) using new interface names. Wifi devices are no more named like wlanX more like wlpXsY

to avoid this, „net.ifnames=0“ can be added to Kernel-Cmdline (uEnv.txt for uboot) and/or rename via udev

/etc/udev/rules.d/70-persistent-net.rules

```
SUBSYSTEM=="net", ACTION=="add", ATTR{address}=="f8:62:aa:50:15:c8",
NAME="wlan1"
```

find attributes like this

```
udevadm info --attribute-walk /sys/class/net/<interface-name>
```

to apply the rule(s) (but does not rename back):

```
udevadm control --reload-rules && udevadm trigger
```

after reboot all works

if driver is compiled as module it can be reloaded (after activating the udev-rules)

```
modprobe -r mt76x2e
```

```
modprobe mt76x2e
```

IP

iproute2 cheatsheet

permanent

/etc/network/interfaces:

```
#first set the upstream-Port (NIC between CPU and MT7530-Switch) up
auto eth0
iface eth0 inet manual
  pre-up ip link set $IFACE up
  post-down ip link set $IFACE down

auto eth1
iface eth1 inet manual
  pre-up ip link set $IFACE up
  post-down ip link set $IFACE down

#then configure the lan-ports
auto lan0
iface lan0 inet static
  hwaddress ether 08:00:00:00:00:00 # if you want to set MAC manually
  address 192.168.0.10
  netmask 255.255.255.0
  gateway 192.168.0.5
  pre-up ip link set $IFACE up
  post-down ip link set $IFACE down
```

systemd

/etc/systemd/network/eth0.network:

```
[Match]
Name=eth0

[Network]
DHCP=no
LinkLocalAddressing=no
ConfigureWithoutCarrier=true
```

/etc/systemd/network/wan.network

```
[Match]
```

```
Name=wan

[Network]
BindCarrier=eth0
#ConfigureWithoutCarrier=true

#IPForward=yes
#IPMasquerade=yes
Address=192.168.0.18/24
DNS=192.168.0.10
Gateway=192.168.0.10
```

Is ConfigureWithoutCarrier set on wan-port, the default-route will not be set,because Address is invalid (Network is down at time of configuration). This should only be set if no default-route is needed.

systemd

second Ethernet lane (gmac)

Needs kernel-patch for eth1 + aux interfaces (currently only in 5.15)

```
- create a bridge for use for wan

    /etc/systemd/network/11-wanbr.netdev
    [NetDev]
    Name=wanbr
    Kind=bridge

    [Bridge]
    DefaultPVID=0 # should be different to other vlan-aware bridges (like
lanbr)
    VLANFiltering=1

- map aux and wan to vlan-aware bridge
- traffic will be tagged inside with vlan-id 99

    /etc/systemd/network/12-wanbr-bind.network
    [Match]
    Name=wan aux

    [Link]
    RequiredForOnline=no

    [Network]
    BindCarrier=eth0
    Bridge=wanbr

    [BridgeVLAN]
    VLAN=99
```

```
    PVID=99
    EgressUntagged=99

- put wanbr up by default

    /etc/systemd/network/13-wanbr.network
    [Match]
    Name=wanbr

    [Network]
    BindCarrier=eth0
    ConfigureWithoutCarrier=true

- configure eth1 as wan

    /etc/systemd/network/15-wan.network
    [Match]
    Name=eth1

    [Network]
    BindCarrier=eth0

    Address=192.168.0.18/24
    Gateway=192.168.0.10
    DNS=192.168.0.10

    IPForward=yes
```

temporary way

```
brdev=gmacbr
ip link add name $brdev type bridge
ip link set dev $brdev up
ip link set dev wan master $brdev
ip link set dev aux master $brdev
ip link set $brdev type bridge vlan_filtering 1
ip a del 192.168.0.18/24 dev wan #remove ip from original interface
ip a add 192.168.0.18/24 dev eth1
ip link set eth1 up
bridge vlan add vid 100 dev wan pvid untagged
bridge vlan add vid 100 dev aux pvid untagged
bridge vlan del vid 1 dev aux
bridge vlan del vid 1 dev wan
ip link set aux up
ip link set wan up
```

how to check

```
bridge vlan
```

```
ip a
ping 192.168.0.21
ethtool -S eth1 #traffic on this gmac?
iperf3 -c 192.168.0.21 #throughput in one direction
iperf3 -c 192.168.0.21 -R #throughput in the other direction
```

temporary

```
ifconfig lan0 192.168.0.10 netmask 255.255.255.0 broadcast 192.168.0.255
```

```
ip addr add 192.168.0.10/24 broadcast 192.168.0.255 dev lan0
```

make sure only 1 port is in the specific subnet.

```
ip a
#or
ip addr show lan0
```

DHCP

Client

/etc/network/interfaces:

```
auto lan3
allow-hotplug lan3
iface lan3 inet dhcp
```

Renew ip via

```
sudo dhclient -v -r lan3
```

Server

/etc/dnsmasq.conf (activate line by removing # on begin of line)

```
conf-dir=/etc/dnsmasq.d
```

/etc/dnsmasq.d/interfaces.conf

```
interface=wlan1
interface=ap0

# DHCP-Server not active for Interface
no-dhcp-interface=eth0
no-dhcp-interface=eth1
```

```
#dhcp-authoritative (interface+range+leasetime, default-gateway-ip as option 3)
dhcp-range=ap0,192.168.10.100,192.168.10.150,255.255.255.0,48h
dhcp-option=ap0,3,192.168.10.1
dhcp-range=wlan1,192.168.11.100,192.168.11.150,255.255.255.0,48h
dhcp-option=wlan1,3,192.168.11.1
```

/etc/dnsmasq.d/interfaces.conf

```
service dnsmasq start
```

more info here: dnsmasq

IPv6

disabling

https://www.thomas-krenn.com/de/wiki/IPv6_deaktivieren

temporary

```
echo 1 > /proc/sys/net/ipv6/conf/all/disable_ipv6
```

to make it permanent create file /etc/sysctl.d/01-disable-ipv6.conf with content:

```
net.ipv6.conf.all.disable_ipv6 = 1
```

testing:

```
ip addr show | grep inet6
```

NAT/Routing

NAT

to enable Network Adress Translation (net with private IPs behind one public IP)

```
ipt=/sbin/iptables
if_wan=wan
${ipt} -t nat -A POSTROUTING -o ${if_wan} -j MASQUERADE
```

HW-Nat

HW-Nat is currently only available in LEDE (Kernel 4.9)

i have merged the Lede-Patches to my 4.9-main and ported to 4.14 (4.14-hnat), see HW-NAT

Routing

enable routing for IPv4

```
echo 1 > /proc/sys/net/ipv4/ip_forward
```

alternative:

```
nano /etc/sysctl.conf
#activate net.ipv4.ip_forward=1 and net.ipv6.conf.all.forwarding=1 by
removing # at beginning of line
sysctl -p /etc/sysctl.conf
```

manipulating default route:

```
ip route del default
ip route add default via 192.168.50.2
```

show routing table

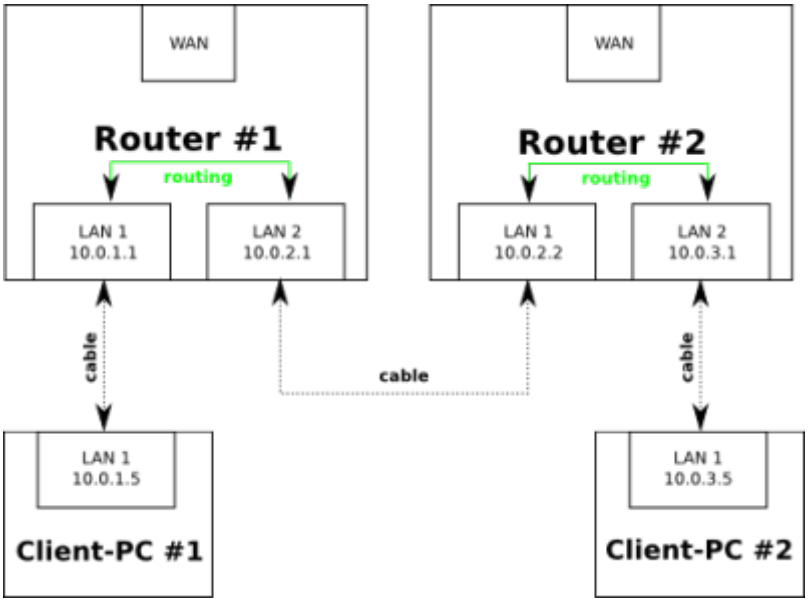
```
ip route show
```

remember you need DNS-resolving (/etc/resolv.conf) for translating domains to ip-addresses

adding static routes to other networks

Pakets are sent to the default-gateway, if the net is not known (directly connected or route available). In normal home-networks there is only 1 router and in this the default-gateway is the Internet-interface and on client-PCs the default-gateway is this router.

static routes are needed, if a net is not directly connected to a router and not accessable via its default-gateway



- in router #1 a static route must be added for net 10.0.3.0/24 with next-hop 10.0.2.2 (send pakets over lan#2)
  - `ip route add 10.0.3.0/24 via 10.0.2.2`
- in router #2 a static route must be added for net 10.0.1.0/24 with next-hop 10.0.2.1 (send pakets over lan#1)
  - `ip route add 10.0.1.0/24 via 10.0.2.1`

example for net 192.168.50.x behind router with ip 192.168.0.10

```
ip route add 192.168.50.0/24 via 192.168.0.10
```

DNS

/etc/resolv.conf

contains ip-adress to nameserver, e.g.

```
nameserver 192.168.0.10
```

on newer debian/ubuntu this file is a symlink to

/run/systemd/resolve/stub-resolv.conf

Netbridge

if 2 or more lan-ports should use same network-segment (configure only 1 IP-address for „LAN“), you can bridge ports together.

```
apt-get install bridge-utils
```

/etc/network/interfaces:

```
auto lan1
iface lan1 inet manual
auto lan2
iface lan2 inet manual

auto br0
iface br0 inet static
    address 192.168.40.1
    netmask 255.255.255.0
    bridge_ports lan1 lan2
    bridge_fd 5
    bridge_stp no
```

temporary bridge

```
brctl addbr br0
brctl addif br0 lan1
brctl addif br0 lan2
ip addr add 192.168.40.1/24 dev br0
ip link set br0 up

brctl show br0
```

Note:

brctl is deprecated please use ip/bridge

```
ip link add name br0 type bridge
ip link set dev br0 up
ip link set dev lan0 master br0
ip link set dev lan1 master br0

#remove interface from bridge
ip link set dev lan0 nomaster

#remove bridge
ip link del br0
```

https://unix.stackexchange.com/a/255489

VLAN

vlan on dsa-ports need

additional Patch

/etc/network/interfaces:

```
auto lan3.60
iface lan3.60 inet static
  address 192.168.60.10
  netmask 255.255.255.0
```

temporary

```
#!/bin/bash
netif=wan
ip link set $netif up
ip link add link $netif name vlan110 type vlan id 110
ip link set vlan110 up
ip addr add 192.168.110.1/24 dev vlan110
#tcpdump -i $netif -nn -e vlan &
```

vlan aware bridge

With 4.16 vlan aware bridge support was added.

vlan_filtering needs to be enabled before dsa-ports are added to the bridge, else all traffic (untagged too) is blocked after this setting.

```
#!/bin/bash
BRIDGE=lanbr0
netif=lan0
vid=500
vlanip=192.168.110.5/24

#ip link add name ${BRIDGE} type bridge
ip link add name ${BRIDGE} type bridge vlan_filtering 1 vlan_default_pvid 1
ip link set ${BRIDGE} up
ip link set $netif master ${BRIDGE}
ip link set $netif up
bridge vlan add vid $vid dev ${BRIDGE} self
bridge vlan add vid $vid dev $netif

#extract vlan from bridge to own netdev
ip link add link ${BRIDGE} name vlan$vid type vlan id $vid
ip a a $vlanip dev vlan$vid
ip link set vlan$vid up
```

testing

```
sudo tcpdump -ei lan1 arp or icmp
```

-e shows link-layer information like vlan

```
sudo tcpdump -XXi lan1 arp or icmp
```

shows arp and icmp-packets as hex-dump on the interface

offset 0x0c should show 8100 followed by hex-value of vlan-number (here vlan 500 = 0x01f4)

```
12:16:26.491644 IP 192.168.50.11 > frank-G5: ICMP echo reply, id 4294, seq
5, length 64
0x0000:  3c18 a003 c3a4 c63a 3897 5920 8100 01f4  <......:8.Y.....
```

Firewall (iptables)

iptables NFTables

Monitoring

```
sudo tcpdump -i eth0 port not 22 > tcpdump.log
sudo tcpdump -XXi lan1 arp or icmp
```

traceroute with tcp-port instead of icmp

```
sudo traceroute -n -T -p 443 domain
```

ethernet-stats:

```
ip -s link show dev lan0
ethtool -S lan0
```

PPPoE

example creates a pppoe-connection over vlan 140 like my ISP

Server

```
sudo apt install pppoe

sudo ip link add link enx00e04c680683 name wan.140 type vlan id 140
```

```
/etc/ppp/pap-secrets
"bpi-r2"    *   "1234578"   *

/etc/ppp/pppoe-server-options
# PPP options for the PPPoE server
# LIC: GPL
debug
#plugin /etc/ppp/plugins/rp-pppoe.so
require-pap
mtu 1492
mru 1492
ktune
proxyarp
lcp-echo-interval 10
lcp-echo-failure 2
nobsdcomp
noccp
novj
noipx
```

```
iptables -t nat -A POSTROUTING -s 192.168.1.0/24 -o enp3s0 -j MASQUERADE
echo 1 > /proc/sys/net/ipv4/ip_forward

pppoe-server -I wan.140 -L 192.168.1.1 -R 192.168.1.100 -F &
```

Client

```
apt install pppoeconf
ip link add link wan name wan.140 type vlan id 140
ip link set wan.140 up
pppoeconf wan.140
```

/etc/ppp/peers/dsl-provider (should be created by pppoeconf)

```
# Minimalistic default options file for DSL/PPPoE connections

noipdefault
defaultroute
replacedefaultroute
hide-password
#lcp-echo-interval 30
#lcp-echo-failure 4
noauth
persist
#mtu 1492
#persist
#maxfail 0
#holdoff 20
plugin rp-pppoe.so wan.140
```

```
user "bpi-r2"
usepeerdns
```

```
pon dsl-provider
```

i had to delete old route to my local lan-subnet which is used to connect my dns (then the default route through ppp is used).

```
root@bpi-r2:~# ping 192.168.0.10
PING 192.168.0.10 (192.168.0.10) 56(84) bytes of data.
^C
--- 192.168.0.10 ping statistics ---
3 packets transmitted, 0 received, 100% packet loss, time 12ms
root@bpi-r2:~# cat /etc/resolv.conf
nameserver 192.168.0.10
root@bpi-r2:~# ip route
default dev ppp0 scope link
192.168.0.0/24 dev wan proto kernel scope link src 192.168.0.12
192.168.1.1 dev ppp0 proto kernel scope link src 192.168.1.100
192.168.90.0/24 dev lan3 proto kernel scope
root@bpi-r2:~# ip route del 192.168.0.0/24
root@bpi-r2:~# ping 192.168.0.10
PING 192.168.0.10 (192.168.0.10) 56(84) bytes of data.
64 bytes from 192.168.0.10: icmp_seq=1 ttl=63 time=1.56 ms
64 bytes from 192.168.0.10: icmp_seq=2 ttl=63 time=2.22 ms
^C
--- 192.168.0.10 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 3ms
rtt min/avg/max/mdev = 1.563/1.893/2.224/0.333 ms
root@bpi-r2:~# ping www.google.de
PING www.google.de (142.250.185.99) 56(84) bytes of data.
64 bytes from fra16s49-in-f3.1e100.net (142.250.185.99): icmp_seq=1 ttl=120
time=7.57 ms
64 bytes from fra16s49-in-f3.1e100.net (142.250.185.99): icmp_seq=2 ttl=120
time=7.48 ms
^C
--- www.google.de ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 2ms
rtt min/avg/max/mdev = 7.483/7.526/7.570/0.097 ms
root@bpi-r2:~# ping www.google.de
PING www.google.de (142.250.185.99) 56(84) bytes of data.
64 bytes from fra16s49-in-f3.1e100.net (142.250.185.99): icmp_seq=1 ttl=120
time=7.57 ms
64 bytes from fra16s49-in-f3.1e100.net (142.250.185.99): icmp_seq=2 ttl=120
time=7.48 ms
^C
--- www.google.de ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 2ms
rtt min/avg/max/mdev = 7.483/7.526/7.570/0.097 ms

root@bpi-r2:~# traceroute 8.8.8.8
```

```
traceroute to 8.8.8.8 (8.8.8.8), 30 hops max, 60 byte packets
 1  192.168.1.1 (192.168.1.1)  0.594 ms  0.389 ms  0.278 ms
 2  bpi-r2-emmc (192.168.0.10)  1.408 ms  1.327 ms  1.109 ms
 3  me60.stadtnetz-bamberg.de (217.61.144.1)  4.962 ms  4.873 ms  4.789 ms
```

From:
https://wiki.fw-web.de/ - **FW-WEB Wiki**

Permanent link:
https://wiki.fw-web.de/doku.php?id=en:bpi-r2:network:start

Last update: **2023/08/23 19:39**

DNSMASQ

activate config-folder

before own config-files in /etc/dnsmasq.d get loaded, this folder have to be activated in /etc/dnsmasq.conf by removing the sharp (#) on beginning of the following line

```
conf-dir=/etc/dnsmasq.d
```

Interfaces

The Interface-config is separated in my setup to /etc/dnsmasq.d/interfaces.conf. In this file interfaces and their Settings are defined.

which Interfaces should respond to DHCP-Requests (binding)?

```
interface=lan0
interface=wlan1
#interface=tun0
interface=lxcbr0
interface=ap0
```

which interfaces send no DHCP-answer?

```
no-dhcp-interface=eth0
no-dhcp-interface=eth1
```

the settings for DHCP:

```
dhcp-range=lan0,192.168.0.100,192.168.0.150,255.255.255.0,48h
dhcp-option=lan0,3,192.168.0.10
dhcp-range=ap0,192.168.10.100,192.168.10.150,255.255.255.0,48h
dhcp-option=ap0,3,192.168.10.1
dhcp-range=wlan1,192.168.11.100,192.168.11.150,255.255.255.0,48h
dhcp-option=wlan1,3,192.168.11.1
dhcp-range=lxcbr0,10.0.3.100,10.0.3.150,255.255.255.0,48h
dhcp-option=lxcbr0,3,10.0.3.1
```

dhcp-range=interface,ip-range-start,ip-range-end,netmask,lease-time dhcp-option=interface,3,Default-Gateway

Per client dns setting:

Add set option to host between mac and ip (after name)

```
set:specialhosts
```

And set dns-servers for that name

```
dhcp-option=tag:specialhosts,option:dns-server,8.8.8.8,8.8.4.4
```

https://serverfault.com/questions/509388/per-client-dns-servers-with-dnsmasq

static IP for MAC

DNSMASQ can answer a DHCP-Request from a specific MAC-address always with same IP-address. For that i have also created a separate config-file (/etc/dnsmasq.d/mac.conf).

an Entry for that looks like this:

```
dhcp-host=b8:27:eb:90:e6:06,raspberry,192.168.0.2,12h
```

MAC-Address of the Client, a name, the desired IP-Address and last param is the Lease-time (how long lasts the DHCP-answer ⇒ renew of IP needed)

on wifi i have the problem, that i have 2 WIFI-APs (with different IPv4-Subnet) running on R2 and on both the same MAC-addresses can send a query. here the entry can contain the interface:

```
dhcp-host=ap0,00:25:d3:f5:32:4b,media,192.168.10.11,12h
dhcp-host=wlan1,00:25:d3:f5:32:4b,media,192.168.11.11,12h
```

dhcp-cache

```
# service dnsmasq stop
Stopping DNS forwarder and DHCP server: dnsmasq.
# nano /var/lib/misc/dnsmasq.leases
# service dnsmasq start
Starting DNS forwarder and DHCP server: dnsmasq.
```

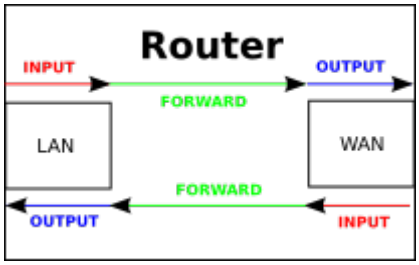
From:
https://wiki.fw-web.de/ - **FW-WEB Wiki**

Permanent link:
https://wiki.fw-web.de/doku.php?id=en:bpi-r2:network:dnsmasq

Last update: **2023/06/08 17:06**

IPTables

🔧 Fix Me!



IPv4

```
#delete previous rules
${ipt} -F
${ipt} -X
${ipt} -t nat -F
${ipt} -t nat -X
${ipt} -t mangle -F
${ipt} -t mangle -X

# Default-Rule for IPv4: drop all
${ipt} -P INPUT   DROP
${ipt} -P OUTPUT  DROP
${ipt} -P FORWARD DROP
```

```
# policy for TCP-Reset/UDP-Reject as alternative to "-j DROP"
${ipt} -N REJECTED
if [[ ! "${LOG}" = "" ]];
then
  echo "enable IPv4-Firewall-Logging (all)...";
  ${ipt} -A REJECTED -m limit --limit 10/min -j LOG --log-prefix
"NETFILTER4-REJECTED: " --log-level 4
fi
${ipt} -A REJECTED -p tcp -j REJECT --reject-with tcp-reset
${ipt} -A REJECTED -p udp -j REJECT --reject-with icmp-port-unreachable
${ipt} -A REJECTED -j DROP
```

```
# localhost
${ipt} -A INPUT  -i lo -j ACCEPT
${ipt} -A OUTPUT -o lo -j ACCEPT

${ipt} -A OUTPUT -j ACCEPT
${ipt} -A INPUT  -m state --state RELATED,ESTABLISHED -j ACCEPT # incoming
connetions which are requested
${ipt} -A INPUT  -p icmp -m limit --limit 5/s --icmp-type echo-request -j
```

```
ACCEPT # ICMP incoming, max 5/s
```

```
#Block Teredo-Stuff
#${ipt} -I FORWARD -p udp --dport 3544 -j REJECTED
#${ipt} -I FORWARD -p udp --sport 3544 -j REJECTED
#http://en.wikipedia.org/wiki/List_of_IP_protocol_numbers
${ipt} -A FORWARD -p 41 -j REJECTED #IPv6 Encapsulation
${ipt} -A FORWARD -p 43 -j REJECTED #Routing Header for IPv6
${ipt} -A FORWARD -p 44 -j REJECTED #Fragment Header for IPv6
${ipt} -A FORWARD -p 58 -j REJECTED #ICMP for IPv6
${ipt} -A FORWARD -p 59 -j REJECTED #No Next Header for IPv6
${ipt} -A FORWARD -p 60 -j REJECTED #Destination Options for IPv6
```

```
#ssh with rate-limit (replacing hosts.allow)
${ipt} -I INPUT -p tcp --dport 22 -i ${if_ext} -m state --state NEW -m
recent --set
${ipt} -I INPUT -p tcp --dport 22 -i ${if_ext} -m state --state NEW -m
recent --update --seconds 60 --hitcount 4 -j REJECTED #4 connections in 1
minute
${ipt} -A INPUT  -p tcp --dport  22 -j ACCEPT #SSH incoming
```

```
${ipt} -A FORWARD -i ${if_int} -o ${if_ext} -j ACCEPT #Forwarding Int->Ext
${ipt} -A FORWARD -i ${if_ext} -o ${if_int} -m state --state
ESTABLISHED,RELATED -j ACCEPT #Forwarding Ext->Int (only existing/requested
connections)

${ipt} -A INPUT -i ${if_int} -j ACCEPT #accept all request from internal
```

... (some other rules, e.g. port-forwardings)

```
# REJECT/RESET for everything else
${ipt} -A INPUT   -j REJECTED
${ipt} -A OUTPUT  -j REJECTED
${ipt} -A FORWARD -j REJECTED
```

additional options:

```
#Kernel-option for SYN-Cookies
echo 1 > /proc/sys/net/ipv4/tcp_syncookies  #enable syn cookies (prevent
against 'syn flood attack')

if [ -f /proc/sys/net/ipv4/conf/all/accept_redirects ]; then
    echo "   Kernel ignores all ICMP redirects"
    echo 0 > /proc/sys/net/ipv4/conf/all/accept_redirects
fi

if [ -f /proc/sys/net/ipv4/icmp_echo_ignore_broadcasts ]; then
   echo "   Kernel ignores ICMP Echo requests sent to broadcast/multicast
addresses"
   echo 1 > /proc/sys/net/ipv4/icmp_echo_ignore_broadcasts
```

```
fi
```

Port-Forwardings

setup

forward port 522 to Client 192.168.0.5 port 22

```
${ipt} -t nat -A PREROUTING -p tcp --dport 522 -j DNAT --to-destination
192.168.0.5:22
```

show

```
iptables -L -t nat

Chain PREROUTING (policy ACCEPT)
target     prot opt source               destination
DNAT       tcp  --  anywhere             anywhere             tcp dpt:522
to:192.168.0.5:22
```

active-ftp

to allow active-ftp from a client you need to load 2 modules and set 1 iptables-rule

```
modprobe ip_conntrack_ftp
modprobe ip_nat_ftp ports=21
```

```
${ipt} -A INPUT -p tcp --sport 20 -m state --state ESTABLISHED,RELATED -j
ACCEPT
```

IPv6

From:
https://wiki.fw-web.de/ - **FW-WEB Wiki**

Permanent link:
https://wiki.fw-web.de/doku.php?id=en:bpi-r2:network:iptables

Last update: **2023/06/08 17:06**

NFTables

Example from eric with combined ipv4+ipv6 (latter untested):
https://github.com/ericwoud/buildR64arch/blob/main/rootfs/etc/nftables.conf

forwarding: https://www.eukhost.com/kb/how-to-enable-ip-forwarding-on-linux-ipv4-ipv6/

```
apt install nftables
echo 1 > /proc/sys/net/ipv4/ip_forward

nft list ruleset
nft add table nat
nft add chain ip nat prerouting { type nat hook prerouting priority 100 \; }
nft add chain ip nat postrouting { type nat hook postrouting priority 100 \;
}
nft add rule nat postrouting masquerade

#portforwarding
nft add rule nat prerouting iif lan1 tcp dport 443 dnat 192.168.0.10:443 #
ip needs to be routed to other interface then in-interface (here lan1)
```

named priorities (>=0.9.1):
https://thermalcircle.de/doku.php?id=blog:linux:nftables_packet_flow_netfilter_hooks_detail

links

- https://developers.redhat.com/blog/2017/01/10/migrating-my-iptables-setup-to-nftables/
- https://wiki.gentoo.org/wiki/Nftables/Examples#Basic_routing_firewall
- https://wiki.nftables.org/wiki-nftables/index.php
- https://wiki.nftables.org/wiki-nftables/index.php/Quick_reference-nftables_in_10_minutes

hwnat

https://github.com/frank-w/BPI-R2-4.14/commits/5.12-hnat

ipv6 mangle does not support hnat (connection reset!)

to get hwnat working, a newer version of nftables is needed than available in debian buster

https://github.com/frank-w/nftables-bpi

compiled: https://drive.google.com/drive/folders/1hajKvqQa96WRrAy52fQX90i59I1s0h-i?usp=sharing

basic IPv4 Ruleset:

```
flush ruleset
table ip filter {
```

```
        flowtable f {
                hook ingress priority filter + 1
                devices = { lan3, lan0, wan }
                flags offload;
        }
        chain input {
                type filter hook input priority filter; policy accept;
        }

        chain output {
                type filter hook output priority filter; policy accept;
        }

        chain forward {
                type filter hook forward priority filter; policy accept;
                ip protocol { tcp, udp } flow add @f
        }
}
table ip nat {
        chain post {
                type nat hook postrouting priority filter; policy accept;
                oifname "wan" masquerade
        }

        chain pre {
                type nat hook prerouting priority filter; policy accept;
        }
}
```

basic v6 Ruleset (hw-nat for IPv6 not supported):

```
flush ruleset
table ip6 filter {
        flowtable f {
                hook ingress priority 1
                devices = { lan3, lan0, wan }
                flags offload;
        }
        chain input {
                type filter hook input priority 0; policy accept;
        }

        chain output {
                type filter hook output priority 0; policy accept;
        }

        chain forward {
                type filter hook forward priority 0; policy accept;
#                ip6 nexthdr { tcp, udp } flow add @f
        }
}
```

```
table ip6 nat {
        chain post {
                type nat hook postrouting priority 0; policy accept;
                #oifname "wan" masquerade
        }

        chain pre {
                type nat hook prerouting priority 0; policy accept;
        }
}
```

test it:

```
nft -f nft-nat-flowoffload.nft
#generate traffic from client e.g. iperf3
cat /sys/kernel/debug/mtk_ppe/entries
```

IPV6-Setup

```
#!/bin/bash
#on main-router:
#ip -6 route add fd00:a2::/64 via fd00:a::12
#ip -6 route add 2001:470:xxxx:a2::/64 via 2001:470:xxxx::12

ip -6 addr add fd00:a::12/64 dev wan
ip -6 addr add fd00:a2::12/64 dev lan3

ip -6 addr add 2001:470:xxxx::12/64 dev wan
ip -6 addr add 2001:470:xxxx:a2::12/64 dev lan3

sysctl -w net.ipv6.conf.all.forwarding=1
```

From:
https://wiki.fw-web.de/ - **FW-WEB Wiki**

Permanent link:
https://wiki.fw-web.de/doku.php?id=en:bpi-r2:network:nftables

Last update: **2023/09/10 16:40**

openwrt

cloning repo

```
git clone https://www.github.com/frank-w/openwrt
cd openwrt
```

installing depencies:

```
./scripts/feeds update -a
./scripts/feeds install -a
```

configuration:

```
make menuconfig
=> Target System (MediaTek Ralink ARM)
=> Subtarget (MT7623)
=> Target Profile (Bpi Banana Pi R2)
=> Target Images
   => tar.gz
```

edit Makefile to change Kernel from 4.19 to 5.4 (already done in my repo)

```
nano target/linux/mediatek/Makefile
=> KERNEL_PATCHVER:=5.4
make -j8 kernel_menuconfig
```

From:
https://wiki.fw-web.de/ - **FW-WEB Wiki**

Permanent link:
https://wiki.fw-web.de/doku.php?id=en:bpi-r2:openwrt

Last update: **2023/06/08 17:06**

OTG

OTG is not yet working, page only created to save debug-steps with musb-driver merged in 5.7

https://github.com/frank-w/BPI-R2-4.14/tree/5.9-otg

kernel-options:

```
CONFIG_CONFIGFS_FS=y
#OTG
CONFIG_USB_CONN_GPIO=y
CONFIG_USB_MUSB_HDRC=y
CONFIG_USB_MUSB_MEDIATEK=y
CONFIG_NOP_USB_XCEIV=y
CONFIG_USB_CONFIGFS=y
#CONFIG_USB_CONFIGFS_MASS_STORAGE=y
#CONFIG_PHY_MTK_TPHY=y
CONFIG_USB_GADGET=y
CONFIG_USB_MUSB_DUAL_ROLE=y
CONFIG_USB_INVENTRA_DMA=y
```

for dma-error i enabled

```
CONFIG_USB_INVENTRA_DMA=y
```

and additional for role-switch

```
CONFIG_USB_GADGET=y
CONFIG_USB_MUSB_DUAL_ROLE=y
```

```
mount -t configfs none /sys/kernel/config
cd /sys/kernel/config/usb_gadget
mkdir g1
cd g1
echo "0x0e8d" > idVendor
echo "0x2002" > idProduct
mkdir strings/0x409
echo "0123456789ABCDEF" > strings/0x409/serialnumber
echo "Mediatek Inc." > strings/0x409/manufacturer
echo "mass_storage gadget" > strings/0x409/product
mkdir functions/mass_storage.usb0
mkdir configs/c.1
mkdir configs/c.1/strings/0x409
ln -s functions/mass_storage.usb0/ configs/c.1
ls /sys/class/udc/
echo xxx > UDC
```

From:
https://wiki.fw-web.de/ - **FW-WEB Wiki**

Permanent link:
https://wiki.fw-web.de/doku.php?id=en:bpi-r2:otg

Last update: **2023/06/08 17:06**

Software

first steps

download Operation System: Google Drive or dev.banana-pi.org.cn

copy Image to SD-Card (Linux):

```
dd if=path/to/image.img of=/dev/sdx bs=1M
```

Debian

first boot (do this using Debug-UART)

login: root **password:** bananapi

System-Update & Clock-Settings

```
apt-get update && apt-get upgrade
echo "bpi-r2">/etc/hostname
dpkg-reconfigure tzdata
```

some changes to environment (to make persistent: echo „...“»~/.bashrc)

```
#fix for nano issues on debug-console
if [[ "$(tty)" =~ "ttyS" ]]; then export TERM=vt100;fi
#prompt with time
export PS1='[\A] \u@\h:\W# '
#easier access to GPIO by $GPIO
export GPIO=/sys/devices/platform/1000b000.pinctrl/mt_gpio
```

Network-Configuration

Network-configuration

temporary

```
#4.4.70:
ifconfig eth0 192.168.0.10/24
route add default gw 192.168.0.5
echo "nameserver 192.168.0.5" > /etc/resolv.conf
```

```
#4.14:
#ifconfig eth0 up
ip link set eth0 up
#ifconfig lan0 192.168.0.10/24
ip addr add 192.168.0.10/24 dev lan0
#ifconfig lan0 up
ip link set lan0 up
#route add default gw 192.168.0.5
ip route add 0.0.0.0/0 via 192.168.0.5
echo "nameserver 192.168.0.5" > /etc/resolv.conf
```

fixed (also after reboot)

4.4.70:

```
nano /etc/network/interfaces
```

```
auto eth0
  iface eth0 inet static
  hwaddress ether 08:00:00:00:00:01
  address 192.168.0.10
  netmask 255.255.255.0
  gateway 192.168.0.5

auto eth1
  iface eth1 inet static
```

4.14:

```
auto eth0
iface eth0 inet manual
  pre-up ip link set $IFACE up
  post-down ip link set $IFACE down

auto lan0
iface lan0 inet static
  hwaddress ether 08:00:00:00:00:00 # if you want to set MAC manually
  address 192.168.0.10
  netmask 255.255.255.0
  gateway 192.168.0.5
  pre-up ip link set $IFACE up
  post-down ip link set $IFACE down

auto lan1
iface lan1 inet static
  hwaddress ether 08:00:00:00:00:01 # if you want to set MAC manually
  address 192.168.1.10
  netmask 255.255.255.0
  pre-up ip link set $IFACE up
```

```
  post-down ip link set $IFACE down

auto lan2
iface lan2 inet static
  hwaddress ether 08:00:00:00:00:02 # if you want to set MAC manually
  #...

auto lan3
iface lan3 inet static
  hwaddress ether 08:00:00:00:00:03 # if you want to set MAC manually
  #...

auto wan
iface wan inet static
  hwaddress ether 09:00:00:00:00:01 # if you want to set MAC manually
  #...
```

in debian 9 hwaddress does not work, here you can set the MAC this way:

```
iface lan0 inet static
  address 192.168.0.10
  netmask 255.255.255.0
  gateway 192.168.0.5
#  pre-up ip link set $IFACE up
  pre-up ip link set $IFACE address 02:01:02:03:04:08 up
  post-down ip link set $IFACE down
```

UDEV-way from here

```
$ cat /etc/udev/rules.d/00-static-mac-address.rules
ACTION=="add", SUBSYSTEM=="net", KERNELS=="1b100000.ethernet",
RUN+="/sbin/ip link set dev %k address ae:fc:de:ad:be:ef"
```

DHCP

```
allow-hotplug lan3
iface lan3 inet dhcp
```

bridge (4.14)

Fix Me! untested !

```
apt-get install bridge-utils
```

```
brctl addbr br0
brctl addif br0 lan1 lan2 lan3 #bridging lan1-lan3 (keep lan0 separately)
```

```
root@bpi-r2:~# brctl show br0
bridge name     bridge id               STP enabled     interfaces
br0             8000.6acba7512bc1       no              lan1
                                                        lan2
                                                        lan3
```

/etc/metwork/interfaces:

```
iface br0 inet static
    address 192.168.40.1
    netmask 255.255.255.0
    bridge_ports lan1 lan2
    bridge_fd 5
    bridge_stp no
```

vlan

4.14:

/etc/metwork/interfaces:

```
auto lan3
iface lan3 inet manual

auto lan3.60
iface lan3.60 inet static
  address 192.168.60.10
  netmask 255.255.255.0
#  gateway 192.168.0.5
  pre-up ip link set $IFACE address 02:01:02:03:04:03 up #setting mac does
not work currently
```

From:
https://wiki.fw-web.de/ - **FW-WEB Wiki**

Permanent link:
https://wiki.fw-web.de/doku.php?id=en:bpi-r2:software

Last update: **2023/06/08 17:06**

Storage

Partition		SD card	eMMC	Update(BPI-Tool)	Update(RAW command)
BootLoader	Header	0-2KB	Boot0: 0-2KB	SD (On HOST PC or target board) : *bpi-bootsel BPI-R2-720P-2k.img.gz /dev/mmcblkX* (for updating both uboot and preloader) eMMC(Only on target board): *echo 0 > /sys/block/mmcblkYboot0/force_ro* *bpi-bootsel BPI-R2-EMMC-boot0-DDR1600-0k-0905.img.gz /dev/mmcblkyboot0* (for updating bootloader to EMMC boot0) *bpi-bootsel BPI-R2-720P-2k.img.gz /dev/mmcblkY* (for updating Uboot to eMMC user block)	We just have RAW command to update EMMC preloader under uboot: *emmc write 1 84000000 0 200*
	Preloader	2-320KB	Boot0: 0-320KB		
	Uboot	320KB - 1MB	User Data Area: 320KB - 1MB		Write Uboot to SD or EMMC under uboot: mmc dev Z mmc write 84000000 280 200 Where Z is the number of disk, 0-emmc, 1-SD
	Reserved	1MB - 100MB	User Data Area: 1MB - 100MB		
File System Partition1(FAT32)		100MB - 356MB	User Data Area:100MB - 356MB		
File System Partition2(EXT4)		356MB - 7456MB	User Data Area:356MB - 7456MB	cd SD;bpi-update -c bpi-r2.conf -d /dev/<device>	

image by garywang

it looks like /dev/mmcblk1, /dev/mmcblk1boot0 and /dev/mmcblk1boot1 are independend devices (bootx not partitions inside /dev/mmcblk1)

preloader

for SD

have to be written at 2k-offset (0x800)

```
sudo dd if=BPI-R2-EMMC-boot0-DDR1600-20190722-2k.img of=/dev/sdc bs=1k seek=2
```

(SD-Card needs additional MMC_BOOT & BRLYT header, see below)

for EMMC

have to be written at 0-offset (0x0) of boot0-Partition

```
sudo dd if=BPI-R2-EMMC-boot0-DDR1600-20190722-0k.img of=/dev/mmcblk1boot0
```

new preloaderfiles from here: https://github.com/BPI-SINOVOIP/BPI-files/tree/master/SD/100MB

SD-Card

sdcard-bootsektor reverse-engineering
http://forum.banana-pi.org/t/boot-fails-with-self-build-u-boot/5460/20

http://forum.banana-pi.org/t/how-to-build-an-ubuntu-debian-sd-image-from-scratch/6805/8

SD-Headers

bpi-r2-head440-0k.img
bpi-r2-head1-512b.img

- SDMMC_BOOT-Signatur + address of 2nd header (0x00000200) - first 440 byte (before

partitiontable):

```
gunzip -c BPI-R2-HEAD440-0k.img.gz | dd of=/dev/loop8 bs=1024 seek=0
```

- BRLYT-signature + preloader-Address (0x00000800):

```
gunzip -c BPI-R2-HEAD1-512b.img.gz | dd of=/dev/loop8 bs=512 seek=1
```

short install-guide

this guide is derived from user ul90 but using a fixed partition table (existing image/sdcard) and variables for loopdevice

```
dd if=/dev/zero of=../bpi-r2-buster.img bs=1M count=7168
loopdev=$(losetup -f)
sudo losetup ${loopdev} ../bpi-r2-buster.img
echo $loopdev
sudo dd if=~/Downloads/BPI-R2-preloader-DDR1600-20190722-2k.img
of=${loopdev} bs=1k seek=2
sudo dd if=~/Downloads/BPI-R2-HEAD440-0k.img of=${loopdev} bs=1024 seek=0
sudo dd if=~/Downloads/BPI-R2-HEAD1-512b.img  bs=512 seek=1
sudo dd if=/path/to/u-boot/u-boot.bin of=${loopdev} bs=1k seek=320
sudo sfdisk ${loopdev} < ~/Downloads/parttable.dat
sudo partprobe ${loopdev}
ls ${loopdev}*
sudo mkfs -t vfat ${loopdev}p1
sudo mkfs -t ext4 ${loopdev}p2
sudo fatlabel ${loopdev}p1 BPI-BOOT
sudo e2label ${loopdev}p2 BPI-ROOT
```

install debian (from bootstrapped rootfs)

```
sudo mount ${loopdev}p2 /mnt/
sudo cp -r debian_buster_armhf/. /mnt/
#install kernel-modules to same partition
kernelpack=/path/to/bpi-r2_<version>_main.tar.gz
sudo tar -xzf ${kernelpack} -C /mnt/ --strip-components=1 BPI-ROOT
#install kernel to boot-partition
sudo umount /mnt
sudo mount ${loopdev}p1 /mnt/
sudo tar -xzf ${kernelpack} -C /mnt/ --strip-components=1 BPI-BOOT
#maybe create a uEnv.txt
sudo umount /mnt
```

```
sudo losetup -d ${loopdev}
#now write the image to card (make sure /dev/sdc is your sdcard-device and
no partition is mounted)
sudo dd if=../bpi-r2-buster.img of=/dev/sdc
sync
```

Step-by-Step-Guide from "ul90"

1. download binary files:

SD-Headers (old not supporting full uboot-size)

2. Create image file (8GB):

```
dd if=/dev/zero bs=1M count=7296 | pv | dd of=bpir2.img
```

3. Load image as virtual drive:

```
losetup /dev/loop8 bpir2.img
```

4. Make partitions and format:

```
parted -s /dev/loop8 mklabel msdos
parted -s /dev/loop8 unit MiB mkpart primary fat32 -- 100MiB 356MiB
parted -s /dev/loop8 unit MiB mkpart primary ext2 -- 356MiB 7295MiB
partprobe /dev/loop8
mkfs.vfat /dev/loop8p1 -I -n BPI-BOOT
mkfs.ext4 -O ^has_journal -E stride=2,stripe-width=1024 -b 4096 /dev/loop8p2
-L BPI-ROOT
sync
parted -s /dev/loop8 print
```

5. extended boot-headers

- SDMMC_BOOT-signature + address of 2nd header 0x00000200 (before partition-table):

```
gunzip -c BPI-R2-HEAD440-0k.img.gz | dd of=/dev/loop8 bs=1024 seek=0
```

- BRLYT-signature + preloader-address (0x00000800):

```
gunzip -c BPI-R2-HEAD1-512b.img.gz | dd of=/dev/loop8 bs=512 seek=1
```

6. Write preloader and u-boot bootloader:

```
dd if=preloader_iotg7623Np1_sd_1600M.bin of=/dev/loop8 bs=1024 seek=2
dd if=u-boot.bin of=/dev/loop8 bs=1024 seek=320
sync
```

7. Copy rootfs + kernel

create bootstraped rootfs

- Ubuntu
- Debian

Install Kernel:

Kernel

8. Remove loop device:

```
losetup -d /dev/loop8
```

MMC-Utils

with the mmc-utils you can test out of a running system, if the EMMC-partitioning is correct (should be 48 see change_partition-configuration_of_emmc).

```
./mmc extcsd read /dev/mmcblk1
....
Boot configuration bytes [PARTITION_CONFIG: 0x48]
....
```

i have added the mmc-utils also to my Kernel-Repo (with changed Makefile for Cross-Compile)

a forum user give me the tip here that partition config can be wrote with mmc-utils too

```
./mmc bootpart enable 1 1 /dev/mmcblk1
```

```
[18:02] root@bpi-r2:~# ./mmc extcsd read /dev/mmcblk1 | grep
PARTITION_CONFIG
Boot configuration bytes [PARTITION_CONFIG: 0x00]
[18:02] root@bpi-r2:~# ./mmc bootpart enable 1 1 /dev/mmcblk1
[18:03] root@bpi-r2:~# ./mmc extcsd read /dev/mmcblk1 | grep
PARTITION_CONFIG
Boot configuration bytes [PARTITION_CONFIG: 0x48]
```

install OS on EMMC

http://forum.banana-pi.org/t/bpi-r2-new-image-ubuntu-16-04-v1-2-1-bt-and-wifi-ap-mode-are-working-fine-2017-11-27/4291

1. change_partition-configuration_of_emmc
2. activate write mode /dev/mmcblk1boot0:

```
echo 0 > /sys/block/mmcblk1boot0/force_ro
```

3. write preloader from here to the emmc boot-device:
   - ```
     gunzip -c BPI-R2-EMMC-boot0-DDR1600-0k-0905.img.gz | sudo dd
     of=/dev/mmcblk1boot0 bs=1024 seek=0
     ```
   - with bpi-tools:

     ```
     bpi-bootsel BPI-R2-EMMC-boot0-DDR1600-0k-0905.img.gz
     /dev/mmcblk1boot0
     ```
4. copy of OS-image to EMMC (device=/dev/mmcblk1):
   - ```
     gunzip deb_stretch_emmc.img.gz
     dd if=deb_stretch_emmc.img of=/dev/mmcblk1 bs=1M
     ```

   - ```
     unzip -p <XXX.img.zip> | pv | dd of=<device> bs=10M status=noxfer
     ```
   - alternate (with bpi-tools):

     ```
     bpi-copy  <XXX.img.zip> <device>
     ```
5. poweroff, remove SD card and boot again

if sd-card-image does not fit emmc: shrinking image

manual copy of OS

- if you want bootable sdcard,you need to copy first 2k from existing sd-image/card.
  - first 2kB without preloader/uboot
    first MB with preloader/uboot

  ```
  gunzip bpi-r2-sd-boot*.img.gz
  dd if=bpi-r2-sd-boot1m.img of=/dev/sdx
  sync
  #remove card after "sync" and reenter
  sudo mkfs -t vfat /dev/sdx1
  sudo mkfs -t ext4 /dev/sdx2
  #set filesystem-labels, partition must not be mounted!
  sudo mlabel -i /dev/sdx1 ::BPI-BOOT
  sudo tune2fs -L BPI-ROOT /dev/sdx2
  ```
- install uboot (not needed if bpi-r2-sd-boot1m.img is flashed):

  ```
  dd if=BPI-R2-720P-2k.img of=/dev/mmcblk1 bs=1k seek=2 count=1022
  ```
- export partitiontable from SD
  parttable.dat
  and write it to emmc (not needed if bpi-r2-sd-bootx.img is flashed):
  ```

```
root@bpi-r2:~# sfdisk -d /dev/mmcblk0 > parttable.dat
root@bpi-r2:~# sfdisk /dev/mmcblk1 < parttable.dat
```

- check/resize
- create filesystems (mkfs) for p1=vfat (apt-get install dosfstools) and p2=ext4 (here emmc)

```
mkfs -t vfat /dev/mmcblk1p1
mkfs -t ext4 /dev/mmcblk1p2
```

- create+configure mount-points in existing system

```
mkdir -p /mnt/emmc/boot
mkdir -p /mnt/emmc/root
nano /etc/fstab
# <file system>         <dir>           <type>  <options>
<dump>  <pass>
/dev/mmcblk0p2          /               ext4    errors=remount-ro
0       1
/dev/mmcblk0p1          /boot           vfat    defaults
0       0
/dev/mmcblk1p2          /mnt/emmc/root  ext4    errors=remount-
ro,noauto  0       1
/dev/mmcblk1p1          /mnt/emmc/boot  vfat    defaults,noauto
0       0
```

- mount it:

```
mount /mnt/emmc/root
mount /mnt/emmc/boot
```

- unpack bootstrapped debian or copy full rootfs from sdcard

```
rsync -aAXv --
exclude={"/dev/*","/proc/*","/sys/*","/tmp/*","/run/*","/mnt/*","/media
/*","/lost+found","/boot/*"} / /mnt/emmc/root/
```

- copy kernel (p1) and modules (p2)

```
mkdir -p /mnt/emmc/boot/bananapi/bpi-r2/linux
cp /boot/bananapi/bpi-r2/linux/uImage /mnt/emmc/boot/bananapi/bpi-
r2/linux
mkdir -p /mnt/emmc/root/lib/modules/
cp -r /lib/modules/$(uname -r) /mnt/emmc/root/lib/modules/
```

- configure uboot to load kernel from right partition

```
sed 's/mmcblk0/mmcblk1/' /boot/bananapi/bpi-r2/linux/uEnv.txt >
/mnt/emmc/boot/bananapi/bpi-r2/linux/uEnv.txt
```

From:
https://wiki.fw-web.de/ - **FW-WEB Wiki**

Permanent link:
https://wiki.fw-web.de/doku.php?id=en:bpi-r2:storage

Last update: **2023/06/08 17:06**

U-Boot

reachable via debug-uart

```
    *** U-Boot Boot Menu ***
       1. System Load Linux to SDRAM via TFTP.
       2. System Load Linux Kernel then write to Flash via TFTP.
       3. Boot Linux from SD.
       4. System Load Boot Loader then write to Flash via TFTP.
       5. System Load Linux Kernel then write to Flash via Serial.
       6. System Load Boot Loader then write to Flash via Serial.
       7. Boot system code via Flash.
       U-Boot console       <<<<<<<<
    Press UP/DOWN to move, ENTER to select
```

Distroboot support

update Uboot

The emmc-command is integrated in uboot since September 29th 2017 (version: „U-Boot 2014.04-rc1 (Oct 16 2017 - 19:33:23)“)

if you want to create new SD-Card (Image) you have to write partitiontable and some additional headers, see here for more

compile U-Boot from official GitHub my GitHub

official way to install is described on Github using „bpi-update -c bpi-r2.conf“

```
sudo dd if=/dev/sdx of=bpi-r2-first10M.img bs=1M count=10 #Backup of first
10MB
sudo dd if=BPI-R2-720P-2k.img of=/dev/sdx bs=1k seek=2 count=1022 #unzipped
img!

sudo dd of=/dev/sdx if=bpi-r2-first10M.img bs=1M count=10 #restore first
10MB (on problems)
```

actual compiled uboot can be found on my gDrive

alternative: flash only uboot (after option 2 in build.sh, don't forget backup):

```
sudo dd of=/dev/sdb if=u-boot-mt/u-boot.bin bs=1k seek=320
```

there are also patches for 2018-11 uboot-patchwork, which i have applied in a uboot-fork. more details here in r2-forum

uboot 2018-11

Mediatek have released patches for BPI-R2...these i have imported to a uboot-fork and configured (build.sh, config, default-Environment, ...): https://github.com/frank-w/u-boot

kernel from SD-Card can be loaded (emmc should also work), ethernet-driver is also built in

uboot-environment is saved (saveenv-command) to boot-device

if you want to load default environment instead of the saved one:

```
env default -a
printenv
```

List of commands

```
BPI-IoT> help
```

back to menu with command „bootmenu“

```
BPI-IoT> help
?       - alias for 'help'
backup_message- print backup message.
base    - print or set address offset
bdinfo  - print Board Info structure
boot    - boot default, i.e., run 'bootcmd'
bootd   - boot default, i.e., run 'bootcmd'
bootm   - boot application image from memory
bootmenu- ANSI terminal bootmenu
bootp   - boot image via network using BOOTP/TFTP protocol
cmp     - memory compare
coninfo - print console devices and information
cp      - memory copy
crc32   - checksum calculation
echo    - echo args to console
editenv - edit environment variable
emmc    - eMMC sub system
env     - environment handling commands
esw_read- esw_read   - Dump external switch/GMAC status !!

exit    - exit script
false   - do nothing, unsuccessfully
fatinfo - print information about filesystem
fatload - load binary file from a dos filesystem
fatls   - list files in a directory (default /)
go      - start application at address 'addr'
help    - print command description/usage
image_blks- read image size from img_size or image header if no specifying
```

```
img_.
image_check- check if image in load_addr is normal.
iminfo  - print header information for application image
imxtract- extract a part of a multi-image
itest   - return true/false on integer compare
loadb   - load binary file over serial line (kermit mode)
loads   - load S-Record file over serial line
loadx   - load binary file over serial line (xmodem mode)
loady   - load binary file over serial line (ymodem mode)
loop    - infinite loop on address range
md      - memory display
mdio    - mdio   - Ralink PHY register R/W command !!

mm      - memory modify (auto-incrementing address)
mmc     - MMC sub-system
mmc2    - MMC sub system
mmcinfo - display MMC info
mtk_image_blks- read image size from image header (MTK format) located at
load_.
mw      - memory write (fill)
nm      - memory modify (constant address)
nor     - nor   - nor flash command

ping    - send ICMP ECHO_REQUEST to network host
printenv- print environment variables
reco_message- print recovery message.
reg     - reg   - Ralink PHY register R/W command !!

reset   - Perform RESET of the CPU
run     - run commands in an environment variable
saveenv - save environment variables to persistent storage
serious_image_check- seriously check if image in load_addr is normal.
setenv  - set environment variables
showvar - print local hushshell variables
sleep   - delay execution for some time
snor    - snor   - spi-nor flash command

source  - run script from memory
test    - minimal test like /bin/sh
tftpboot- boot image via network using TFTP protocol
true    - do nothing, successfully
uboot_check- check if uboot in load_addr is normal.
version - print monitor, compiler and linker version
```

change partition-configuration of EMMC

```
BPI-IoT> emmc --help
emmc - eMMC sub system
Usage:
```

```
  emmc read part addr blk# cnt
  emmc write part addr blk# cnt
  emmc ecsd       - Dump ext csd
  emmc pconf val  - Set Part Config val
```

```
BPI-IoT> emmc ecsd
============================================================
[EXT_CSD] EXT_CSD rev.              : v1.7 (MMCv5.0)
[EXT_CSD] CSD struct rev.           : v1.2
[EXT_CSD] Supported command sets    : 1h
[EXT_CSD] HPI features              : 1h
[EXT_CSD] BG operations support     : 1h
[EXT_CSD] BG operations status      : 0h
[EXT_CSD] Correct prg. sectors      : 0h
[EXT_CSD] 1st init time after part. : 3000 ms
[EXT_CSD] Min. write perf.(DDR,52MH,8b): 0h
[EXT_CSD] Min. read perf. (DDR,52MH,8b): 0h
[EXT_CSD] TRIM timeout: 0 ms
[EXT_CSD] Secure feature support: 55h
[EXT_CSD] Secure erase timeout  : 8100 ms
[EXT_CSD] Secure trim timeout   : 5100 ms
[EXT_CSD] Access size           : 3072 bytes
[EXT_CSD] HC erase unit size    : 512 kbytes
[EXT_CSD] HC erase timeout      : 300 ms
[EXT_CSD] HC write prot grp size: 8192 kbytes
[EXT_CSD] HC erase grp def.     : 0h
[EXT_CSD] Reliable write sect count: 1h
[EXT_CSD] Sleep current (VCC) : 7h
[EXT_CSD] Sleep current (VCCQ): 7h
[EXT_CSD] Sleep/awake timeout : 26214400 ns
[EXT_CSD] Sector count : e90000h
[EXT_CSD] Min. WR Perf.  (52MH,8b): 0h
[EXT_CSD] Min. Read Perf.(52MH,8b): 0h
[EXT_CSD] Min. WR Perf.  (26MH,8b,52MH,4b): 0h
[EXT_CSD] Min. Read Perf.(26MH,8b,52MH,4b): 0h
[EXT_CSD] Min. WR Perf.  (26MH,4b): 0h
[EXT_CSD] Min. Read Perf.(26MH,4b): 0h
[EXT_CSD] Power class: 0
[EXT_CSD] Power class(DDR,52MH,3.6V): 0h
[EXT_CSD] Power class(DDR,52MH,1.9V): 0h
[EXT_CSD] Power class(26MH,3.6V)    : 0h
[EXT_CSD] Power class(52MH,3.6V)    : 0h
[EXT_CSD] Power class(26MH,1.9V)    : 0h
[EXT_CSD] Power class(52MH,1.9V)    : 0h
[EXT_CSD] Part. switch timing    : 1h
[EXT_CSD] Out-of-INTR busy timing: 5h
[EXT_CSD] Card type       : 57h
[EXT_CSD] Command set     : 0h
[EXT_CSD] Command set rev.: 0h
[EXT_CSD] HS timing       : 0h
```

```
[EXT_CSD] Bus width       : 0h
[EXT_CSD] Erase memory content : 0h
[EXT_CSD] Partition config      : 0h      <<<<<<<<<<<<<<<<<<<<<<<<<<<<< wrong
partition config
[EXT_CSD] Boot partition size   : 4096 kbytes
[EXT_CSD] Boot information      : 7h
[EXT_CSD] Boot config protection: 0h
[EXT_CSD] Boot bus width        : 0h
[EXT_CSD] Boot area write prot  : 0h
[EXT_CSD] User area write prot  : 0h
[EXT_CSD] FW configuration      : 0h
[EXT_CSD] RPMB size : 512 kbytes
[EXT_CSD] Write rel. setting  : 1fh
[EXT_CSD] Write rel. parameter: 4h
[EXT_CSD] Start background ops : 0h
[EXT_CSD] Enable background ops: 0h
[EXT_CSD] H/W reset function   : 0h
[EXT_CSD] HPI management       : 0h
[EXT_CSD] Max. enhanced area size : 136h (2539520 kbytes)
[EXT_CSD] Part. support  : 7h
[EXT_CSD] Part. attribute: 0h
[EXT_CSD] Part. setting  : 0h
[EXT_CSD] General purpose 1 size : 0h (0 kbytes)
[EXT_CSD] General purpose 2 size : 0h (0 kbytes)
[EXT_CSD] General purpose 3 size : 0h (0 kbytes)
[EXT_CSD] General purpose 4 size : 0h (0 kbytes)
[EXT_CSD] Enh. user area size : 0h (0 kbytes)
[EXT_CSD] Enh. user area start: 0h
[EXT_CSD] Bad block mgmt mode: 0h
```

```
============================================================
BPI-IoT> emmc pconf 0x48      <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<< change
partition config
BPI-IoT> emmc ecsd            <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<< verify
============================================================
```

```
[EXT_CSD] EXT_CSD rev.              : v1.7 (MMCv5.0)
[EXT_CSD] CSD struct rev.           : v1.2
[EXT_CSD] Supported command sets    : 1h
[EXT_CSD] HPI features              : 1h
[EXT_CSD] BG operations support     : 1h
[EXT_CSD] BG operations status      : 0h
[EXT_CSD] Correct prg. sectors      : 0h
[EXT_CSD] 1st init time after part. : 3000 ms
[EXT_CSD] Min. write perf.(DDR,52MH,8b): 0h
[EXT_CSD] Min. read perf. (DDR,52MH,8b): 0h
[EXT_CSD] TRIM timeout: 0 ms
[EXT_CSD] Secure feature support: 55h
[EXT_CSD] Secure erase timeout  : 8100 ms
```

```
[EXT_CSD] Secure trim timeout   : 5100 ms
[EXT_CSD] Access size           : 3072 bytes
[EXT_CSD] HC erase unit size    : 512 kbytes
[EXT_CSD] HC erase timeout      : 300 ms
[EXT_CSD] HC write prot grp size: 8192 kbytes
[EXT_CSD] HC erase grp def.     : 0h
[EXT_CSD] Reliable write sect count: 1h
[EXT_CSD] Sleep current (VCC) : 7h
[EXT_CSD] Sleep current (VCCQ): 7h
[EXT_CSD] Sleep/awake timeout : 26214400 ns
[EXT_CSD] Sector count : e90000h
[EXT_CSD] Min. WR Perf.  (52MH,8b): 0h
[EXT_CSD] Min. Read Perf.(52MH,8b): 0h
[EXT_CSD] Min. WR Perf.  (26MH,8b,52MH,4b): 0h
[EXT_CSD] Min. Read Perf.(26MH,8b,52MH,4b): 0h
[EXT_CSD] Min. WR Perf.  (26MH,4b): 0h
[EXT_CSD] Min. Read Perf.(26MH,4b): 0h
[EXT_CSD] Power class: 0
[EXT_CSD] Power class(DDR,52MH,3.6V): 0h
[EXT_CSD] Power class(DDR,52MH,1.9V): 0h
[EXT_CSD] Power class(26MH,3.6V)    : 0h
[EXT_CSD] Power class(52MH,3.6V)    : 0h
[EXT_CSD] Power class(26MH,1.9V)    : 0h
[EXT_CSD] Power class(52MH,1.9V)    : 0h
[EXT_CSD] Part. switch timing    : 1h
[EXT_CSD] Out-of-INTR busy timing: 5h
[EXT_CSD] Card type       : 57h
[EXT_CSD] Command set     : 0h
[EXT_CSD] Command set rev.: 0h
[EXT_CSD] HS timing       : 1h
[EXT_CSD] Bus width       : 0h
[EXT_CSD] Erase memory content : 0h
[EXT_CSD] Partition config      : 48h     <<<<<<<<<<<<<<<<<<<<<<<<<<<<
[EXT_CSD] Boot partition size   : 4096 kbytes
[EXT_CSD] Boot information      : 7h
[EXT_CSD] Boot config protection: 0h
[EXT_CSD] Boot bus width        : 0h
[EXT_CSD] Boot area write prot  : 0h
[EXT_CSD] User area write prot  : 0h
[EXT_CSD] FW configuration      : 0h
[EXT_CSD] RPMB size : 512 kbytes
[EXT_CSD] Write rel. setting  : 1fh
[EXT_CSD] Write rel. parameter: 4h
[EXT_CSD] Start background ops : 0h
[EXT_CSD] Enable background ops: 0h
[EXT_CSD] H/W reset function   : 0h
[EXT_CSD] HPI management       : 0h
[EXT_CSD] Max. enhanced area size : 136h (2539520 kbytes)
[EXT_CSD] Part. support  : 7h
[EXT_CSD] Part. attribute: 0h
[EXT_CSD] Part. setting  : 0h
```

```
[EXT_CSD] General purpose 1 size : 0h (0 kbytes)
[EXT_CSD] General purpose 2 size : 0h (0 kbytes)
[EXT_CSD] General purpose 3 size : 0h (0 kbytes)
[EXT_CSD] General purpose 4 size : 0h (0 kbytes)
[EXT_CSD] Enh. user area size : 0h (0 kbytes)
[EXT_CSD] Enh. user area start: 0h
[EXT_CSD] Bad block mgmt mode: 0h
```

in newer uboot (2018):

http://forum.banana-pi.org/t/add-latest-u-boot-support-for-bpi-r2-bpi-r64-not-yet/6938/26

```
mmc partconf 0 1 1 0
```

System-start out of Console

```
BPI-IoT> printenv
...
boot10=mmc init; run boot_normal; bootm
...
bootmenu_2=3. Boot Linux from SD.=run boot10
...
```

```
run boot10
```

set Kernel-filename

in BPI-BOOT/bananapi/bpi-r2/linux/uEnv.txt change the param kernel:

```
#kernel=uImage
#kernel=uImage_4.14.33
kernel=uImage_4.9.92
```

with this you can have multiple kernel-files on disk to fast switch back to older one (if you have different filenames). For multiboot code of uboot has to be changed... The uEnv.txt is not loaded before displaying the boot-menu, it is loaded after „Boot Linux from SD“ is selected.

manually load uEnv.txt

uEnv.txt will be loaded after „Boot from SD“ is selected. here the manual way from console

```
#Boot from emmc
enter to uboot-console
execute “mmc init 0”
```

```
execute “setenv partition 0:1”
execute “run loadbootenv”
execute “env import -t ${scriptaddr} ${filesize} ”

#Boot from SD:
enter to uboot-console
execute “mmc init 1”
execute “setenv partition 1:1”
execute “run loadbootenv”
execute “env import -t ${scriptaddr} ${filesize} ”
```

Source: http://forum.banana-pi.org/t/how-to-extend-the-uboot-menu/5415/7

because of „loadenv“ does not exist in upstream u-boot, these commands/variable-definitions are needed to get same result (load uEnv.txt + kernel):

```
setenv scriptaddr 0x83000000
setenv bpi bananapi
setenv board bpi-r2
setenv service linux
setenv device mmc
setenv partition 1:1
setenv bootenv uEnv.txt
setenv loadbootenv fatload ${device} ${partition} ${scriptaddr}
${bpi}/${board}/${service}/${bootenv}
run loadbootenv
#not define importenv before run loadbootenv (or at least with escaped $ on
filesize-var)!
setenv importenv env import -t ${scriptaddr} \${filesize}
run importenv

printenv

setenv newboot "fatload ${device} ${partition} ${loadaddr}
${bpi}/${board}/${service}/\${kernel}; bootm"
run newboot

#check for boot-device (emmc/sd)
setenv checksd fatinfo ${device} 1:1
setenv selectmmc "if run checksd; then echo Boot from SD ; setenv partition
1:1;else echo Boot from eMMC; setenv partition 0:1 ; fi;"

run selectmmc
```

helpful commands

MMC

```
U-Boot> mmc list
mmc@11230000: 0 (eMMC)
mmc@11240000: 1 (SD)

#set mmc-device
U-Boot> mmc dev 1

#read current device
U-Boot> mmc dev
switch to partitions #0, OK
mmc1 is current device

U-Boot> mmcinfo
Device: mmc@11240000
Manufacturer ID: 1b
OEM: 534d
Name: 00000
Bus Speed: 50000000
Mode : SD High Speed (50MHz)
Rd Block Len: 512
SD version 2.0
High Capacity: Yes
Capacity: 7.6 GiB
Bus Width: 4-bit
Erase Group Size: 512 Bytes

#partitionconfig
#mmc partconf dev [boot_ack boot_partition partition_access]
# - Show or change the bits of the PARTITION_CONFIG field of the specified
device
#example for mode 0x48 (needed for emmc-boot on bpi-r2)
U-Boot> mmc partconf 0
EXT_CSD[179], PARTITION_CONFIG:
BOOT_ACK: 0x1
BOOT_PARTITION_ENABLE: 0x1
PARTITION_ACCESS: 0x0

#set via
U-Boot> mmc partconf 0 1 1 0
```

directory-listing

```
ls mmc 1:1 bananapi/bpi-r2/linux
```

ask for kernel

```
lskernel=ls ${device} ${partition} ${bpi}/${board}/${service};
askkernel=askenv kernelinput "enter uImage-name:";
```

```
boot0=run lskernel;run askkernel;if printenv kernelinput ;then setenv kernel
${kernelinput}; run newboot; fi
bootmenu_0=1. Enter kernel-name to boot from SD/EMMC.=run boot0
```

netboot

netboot

PCIe

uboot before 2020-10 (my version) has bug which causes hang on pci enum if there is no card inserted into slot

```
BPI-R2> pci enum
BPI-R2> pci 0
Scanning PCI devices on bus 0
BusDevFun  VendorId   DeviceId   Device Class       Sub-Class
_____________________________________________________________
00.00.00   0x14c3     0x0801     Bridge device           0x04
00.01.00   0x14c3     0x0801     Bridge device           0x04
BPI-R2> pci 1
Scanning PCI devices on bus 1
BusDevFun  VendorId   DeviceId   Device Class       Sub-Class
_____________________________________________________________
01.00.00   0x14c3     0x7612     Network controller      0x80
BPI-R2> pci 2
Scanning PCI devices on bus 2
BusDevFun  VendorId   DeviceId   Device Class       Sub-Class
_____________________________________________________________
02.00.00   0x1b21     0x0611     Mass storage controller 0x01
BPI-R2> scsi scan
scanning bus for devices...
SATA link 0 timeout.
Target spinup took 0 ms.
AHCI 0001.0200 32 slots 2 ports 6 Gbps 0x3 impl SATA mode
flags: 64bit ncq stag led clo pmp pio slum part ccc sxs
  Device 0: (1:0) Vendor: ATA Prod.: ST750LM022 HN-M7 Rev: 2AR1
            Type: Hard Disk
            Capacity: 715404.8 MB = 698.6 GB (1465149168 x 512)
BPI-R2>
```

SATA

see pcie (pci enum + scsi scan) and then access hdd via

```
ls scsi 0:1
```

USB

```
BPI-R2> usb start
starting USB...
Bus usb@1a1c0000: hcd: 0x1a1c0000, ippc: 0x1a1c4700
u2p:1, u3p:1
Register 200010f NbrPorts 2
Starting the controller
USB XHCI 0.96
Bus usb@1a240000: hcd: 0x1a240000, ippc: 0x1a244700
u2p:1, u3p:1
Register 200010f NbrPorts 2
Starting the controller
USB XHCI 0.96
scanning bus usb@1a1c0000 for devices... 1 USB Device(s) found
scanning bus usb@1a240000 for devices... 2 USB Device(s) found
       scanning usb for storage devices... 1 Storage Device(s) found
BPI-R2> usb tree
USB device tree:
  1  Hub (5 Gb/s, 0mA)
     U-Boot XHCI Host Controller

  1  Hub (5 Gb/s, 0mA)
  |  U-Boot XHCI Host Controller
  |
  +-2  Mass Storage (480 Mb/s, 200mA)
       USB      Flash Disk       906B030002F4

BPI-R2> ls usb 0:1
            efi/
  4767728   kernel

1 file(s), 1 dir(s)
```

Links

patchwork archive

git

From:
https://wiki.fw-web.de/ - **FW-WEB Wiki**

Permanent link:
https://wiki.fw-web.de/doku.php?id=en:bpi-r2:uboot

Last update: **2023/06/08 17:06**

Ubuntu

debootstrap

Steps to get ubuntu-system (use armhf as arch):
https://help.ubuntu.com/lts/installation-guide/armhf/apds04.html

```
sudo apt-get install qemu-user-static debootstrap binfmt-support

distro=bionic
#distro=focal
arch=armhf
#for bpi-r64 use arch=arm64
targetdir=$(pwd)/debootstrap_ubuntu_${distro}_${arch}


sudo debootstrap --arch=$arch --foreign $distro $targetdir

#if you got this error: E: Cannot install into target '...' mounted with
noexec or nodev
sudo mount -i -o remount,exec,dev /mounted_dir

sudo cp /usr/bin/qemu-arm-static $targetdir/usr/bin/
sudo cp /etc/resolv.conf $targetdir/etc
sudo distro=$distro chroot $targetdir
export LANG=C
/debootstrap/debootstrap --second-stage

#echo "deb-src http://archive.ubuntu.com/ubuntu $distro
main">>/etc/apt/sources.list
#echo "deb http://security.ubuntu.com/ubuntu $distro-security
main">>/etc/apt/sources.list
#echo "deb-src http://security.ubuntu.com/ubuntu $distro-security
main">>/etc/apt/sources.list
echo "deb http://ports.ubuntu.com/ubuntu-ports/ $distro
main">>/etc/apt/sources.list
echo "deb-src http://ports.ubuntu.com/ubuntu-ports/ $distro
main">>/etc/apt/sources.list
echo "deb http://ports.ubuntu.com/ubuntu-ports/ $distro-updates
main">>/etc/apt/sources.list
echo "deb-src http://ports.ubuntu.com/ubuntu-ports/ $distro-updates
main">>/etc/apt/sources.list
echo "deb http://ports.ubuntu.com/ubuntu-ports/ $distro-security
main">>/etc/apt/sources.list
echo "deb-src http://ports.ubuntu.com/ubuntu-ports/ $distro-security
main">>/etc/apt/sources.list

echo "bpi-r2-ubuntu" >/etc/hostname
```

```
#set root-password! else you will not be able to login
passwd
```

configure fstab/... like it's done for debian ubuntu 18.4 uses netplan as default network-framework bionic releasenotes

```
#exit chroot
exit
```

pack:

```
cd $targetdir
sudo tar cvpzf ../debootstrap_${distro}_${arch}.tar.gz .
```

install System to SD-Card

prepare SD-Card

```
#unpacking in root-partiton with
sudo tar -xpzf /path/to/debootstrap_$distro.tar.gz
sudo mkdir lib/modules/
cd lib/modules/
#unpack kernel-modules here
```

temporary configure network:

```
ip a
ip link set eth0 up
ip addr add 192.168.0.11/24 dev lan0 #ip/prefix in your lan-segment not used
ip link set lan0 up
ip route add default via 192.168.0.10 #ip from your router
echo "nameserver 192.168.0.10" >>/etc/resolv.conf #ip from your router for
dns resolution
```

install “ifupdown” and add “netcfg/do_not_use_netplan=true” to bootopts in /boot/bananapi/bpi-r2/linux/uEnv.txt

after reboot the „old“ system with /etc/network/interfaces is used. currently /etc/resolv.conf is resetted every reboot

```
root@bpi-r2-ubuntu:~# ls -l /etc/resolv.conf
lrwxrwxrwx 1 root root 39 Jun 13 10:27 /etc/resolv.conf ->
../run/systemd/resolve/stub-resolv.conf
#delete symlink and replace it by a static file with your settings
rm /etc/resolv.conf
echo "nameserver 192.168.0.10" >>/etc/resolv.conf
```

in ubuntu 18.4 there is a own dns-service running, which have to be disabled (followed by a reboot or stop), in order to get e.g. DNSMasq running (like in my wifi.sh-script)

```
systemctl disable systemd-resolved
systemctl stop systemd-resolved
```

create the image

```
imgfile=/path/to/ubuntu-18.04-bpi-r2-preview.img

#create img file from sdcard (optional if no img file available)
sudo dd if=/dev/sdx of=$imgfile
#watch state of dd from another terminal with "sudo kill -SIGUSR1 $(pidof
dd)"

#truncate image to last partitions end (mbr only,do not use for gpt)...make
sure your path does not contain spaces!
IFS=$'\t' #just for security reasons (ignore spaces in path)
ENDOFDATA=$(fdisk -l "$imgfile" |tail -1|awk '{print $3}')
echo $ENDOFDATA
truncate --size=$[($ENDOFDATA+1)*512] $imgfile
#check size
ls -lh "$imgfile"

#maybe do some stuff with the img
loopdev=$(losetup -f)
sudo losetup $loopdev $imgfile
sudo partprobe $loopdev
sudo mount ${loopdev}p2 /mnt
ls /mnt
#overwrite free space with zeros for better packing and not leaving possible
sensitive data
sudo sh -c 'cat /dev/zero >/mnt/null.dat'
sudo rm /mnt/null.dat
sudo umount /mnt

#pack image
gzip $imgfile
md5sum $imgfile.gz > $imgfile.gz.md5
```

i have uploaded my image to my gdrive for testing (ubuntu-18.04-bpi-r2-preview.img.gz)

Lan0-IPv4 is 192.168.0.11 (second port,next to wan) you can change it in /etc/network/interfaces. More on network

installation

Sdcard:

```
dd if=path/to/ubuntu.img of=/dev/sdx
```

Emmc:

- dd if=path/to/ubuntu.img of=/dev/mmcblkx
- maybe you need reboot reading the partitiontable
- First follow steps here
- in /boot/bananapi/bpi-r2/linux/uEnv.txt: find the line “root=…” and replace /dev/mmcblk0p2 with /dev/mmcblk1p2
- in /etc/fstab: replace /dev/mmcblk0p2 with /dev/mmcblk1p2

SSH

ssh-server is installed on my image ubuntu-18.04-bpi-r2-preview.img.gz from my gdrive, but root-login have to be enabled

```
echo "PermitRootLogin yes" >> /etc/ssh/sshd_config
service sshd restart
```

you should create new host-keys for ssh…

```
#first delete your old keys
rm /etc/ssh/ssh_host_*
#reconfigure sshserver-package
dpkg-reconfigure openssh-server
```

From:
https://wiki.fw-web.de/ - **FW-WEB Wiki**

Permanent link:
https://wiki.fw-web.de/doku.php?id=en:bpi-r2:ubuntu

Last update: **2023/06/08 17:06**

VLAN-Support

- Port-separation (4x LAN) works with 4.14, vlan-suport can be tested on my github-repo
- no support in 4.4.70 forum

From:
https://wiki.fw-web.de/ - **FW-WEB Wiki**

Permanent link:
https://wiki.fw-web.de/doku.php?id=en:bpi-r2:vlan

Last update: **2023/06/08 17:06**

WLAN

Config-files don't have any security-params enabled (e.g. WLAN-security/authentification in HostAPd), these are only a start for quick testing

internal

in Kernel 4.4.70 code is available, but must be activated

GitHub Forum

Patch Patch #2

wpasupplicant must be removed and hostapd+dnsmasq installed:

```
apt-get remove wpasupplicant

apt-get install hostapd dnsmasq
```

copy cfg to /system/etc/firmware/

helper-applications for next steps (unpack to /usr/bin)
and
Firmware (unpack to /etc/firmware/)
from here

1. wmt_loader
2. stp_uart_launcher -p /etc/firmware &
3. load driver module (if compiled as module 5.4+): modprobe wlan_gen2
4. echo A >/dev/wmtWifi (activate AP-Mode)

in last step ap-device (Accesspoint) will be created, which can be used by hostapd

```
[14:14] root@bpi-r2:~# ifconfig -a|grep Link
ap0       Link encap:Ethernet  HWaddr 02:08:22:68:39:ff
bond0     Link encap:Ethernet  HWaddr e2:7c:e0:71:31:c1
eth0      Link encap:Ethernet  HWaddr 08:00:00:00:00:00
          inet6 addr: fe80::a00:ff:fe00:0/64 Scope:Link
eth1      Link encap:Ethernet  HWaddr 08:00:00:00:00:01
          inet6 addr: fe80::a00:ff:fe00:1/64 Scope:Link
lo        Link encap:Local Loopback
sit0      Link encap:IPv6-in-IPv4
tunl0     Link encap:IPIP Tunnel  HWaddr
wlan0     Link encap:Ethernet  HWaddr 00:08:22:68:39:ff
```

/etc/hostapd/hostapd.conf:

```
hw_mode=g
interface=ap0
driver=nl80211
channel=1
auth_algs=1
ssid=test
```

start hostapd

```
hostapd -dd /etc/hostapd/hostapd.conf
```

interface can now be further configured and DHCP-Server configured and started: ip-configuration

old Script to start-up wireless AP
wifi.sh

client-mode

Afair only working in vendor kernel versions (4.4,4.14)

http://forum.banana-pi.org/t/topic/5548

works with 4.14-mt6625 branch (4.14.32 + vendor-code + fixes)

```
apt install wpasupplicant
/usr/bin/wmt_loader
/usr/bin/stp_uart_launcher -p /etc/firmware &
echo 1 >/dev/wmtWifi
```

/etc/wpa_supplicant/wpa_supplicant.conf:

```
network={
        ssid="ssid"
        psk="12345678"
        priority=1
}
```

```
wpa_supplicant -i wlan0 -c /etc/wpa_supplicant/wpa_supplicant.conf
dhclient wlan0
```

Kernel 4.14

code from 4.4.70 has been merged to my github-repo

wifi-configuration is the same as in Kernel 4.4 internal

discussion here: forum

systemd init script here: http://forum.banana-pi.org/t/bpi-r2-internal-wifi-bt-mt6625l-kernel/4307/281

known issues

random numbers

after a (Re-)boot the random-number generator not filled enough, so that connection-attempts will be rejected.

in the hostapd-log there are entries like this:

```
random: Cannot read from /dev/random: Resource temporarily unavailable
random: Got 0/14 bytes from /dev/random
random: Only 6/20 bytes of strong random data available from /dev/random
random: Not enough entropy pool available for secure operations
WPA: Not enough entropy in random pool to proceed - reject first 4-way
handshake
...
WPA: Reject 4-way handshake to collect more entropy for random number
generation
random: Mark internal entropy pool to be ready (count=1/2)
...
random: Cannot read from /dev/random: Resource temporarily unavailable
random: Got 0/14 bytes from /dev/random
random: Only 6/20 bytes of strong random data available from /dev/random
random: Allow operation to proceed based on internal entropy
```

http://forum.banana-pi.org/t/bpi-r2-new-image-release-ubuntu-16-04-v1-3-2018-3-30/5293/25

```
apt-get install rng-tools
echo 'HRNGDEVICE=/dev/urandom' >> /etc/default/rng-tools
```

init failed

```
root@bpi-iot-ros-ai:~# wmt_loader init combo device
Opened combo device
get device id : 30243
set device id : 30243
do module init: 30243
failed to init module
```

This can be ignored because the combo-driver tries to init wifi,bt,gps and fm. Last 2 fail because driver does not support them.

stpnode

```
Can't open device node(/dev/stpwmt) error:-1
//in dmesg:
[WMT-DEV][W]WMT_open:wait_event_timeout (2000)ms,(200)jiffies,return -EIO
```

reboot fixed that...reason still unclear

Also this can be tried:

```
apt-get install network-manager
sudo nmcli radio wifi off
rfkill unblock 1
rfkill unblock 2
```

no traffic

Kernelmodules SCH_FQ / SCH_FQ_CODEL are known breaking ap0-functionality

maybe there are some kalHifAhbKalWakeLockTimeout messages in dmesg

maybe NETFILTER_XT_TARGET_NFQUEUE does have similar effects (see here)

rcu_stall

this is maybe caused by net.ifnames=0

```
[  437.885886] [MTK-WIFI] WIFI_open: WIFI_open: major 155 minor 0 (pid 2614)
[  437.892961] [MTK-WIFI] WIFI_write: WIFI_write A
[  437.892961]
[  437.938648] kalFirmwareImageMapping firmware size 225840
[  439.132189] [MTK-WIFI] register_set_p2p_mode_handler: (pid 2620) register
set p2p mode handler b6888aac
[  439.141960] [MTK-WIFI] WIFI_write: WMT turn on WIFI success!
[  460.170441] rcu: INFO: rcu_sched detected stalls on CPUs/tasks:
```

hang occours in kalioctl (drivers/misc/mediatek/connectivity/wlan/gen2/os/linux/gl_kal.c) while calling kernel-function down_interruptible (kernel/locking/semaphore.c), semaphore is created correctly in drivers/misc/mediatek/connectivity/wlan/gen2/os/linux/gl_init.c (sema_init(&prGlueInfo→ioctl_sem, 1);)

solution: do not use net.ifnames in cmdline and use udev-renaming

driver hang

check if there is a wlan0 device before starting wifi.sh/manual init of ap0 device. driver will hang after message

```
[MTK-WIFI] WIFI_write: WMT turn on WIFI success
```

if there is a wlan0 interface not mapped to the mt6625l driver (maybe pcie/usb-wifi-device).

rootcause is adding a handler to wrong netdev searched with hardcoded wlan0 in drivers/misc/mediatek/connectivity/common/conn_soc/linux/pub/wmt_chrdev_wifi.c

```
#define WLAN_IFACE_NAME "wlan0"
static INT8 *ifname = WLAN_IFACE_NAME;
netdev = dev_get_by_name(&init_net, ifname);
pf_set_p2p_mode(netdev, p2pmode);
```

you can rename existing wlan0 to anything else before starting ap0 init

```
ip link set wlan0 name wlan_2g
ip link set wlan1 name wlan_5g
```

did this with my mt7615 (dbdc mode) and wifi init finished with success in linux 5.10

client-mode does not work

client mode does not work in 4.9+

http://forum.banana-pi.org/t/bpi-r2-internal-wifi-bt-mt6625l-kernel/4307/293

i tested wifi-client in 5.10, here it is working with config posted here:

http://forum.banana-pi.org/t/bpi-r2-internal-wifi-bt-mt6625l-kernel/4307/329

external

MT7612

mt7612e on AliExpress

firmware neeeded: https://packages.debian.org/buster/firmware-misc-nonfree

4.4.70

forum

```
git clone https://github.com/BPI-SINOVOIP/BPI-R2-bsp.git bpi_r2_mt76
cd bpi_r2_mt76/
cd linux-mt/drivers/net/wireless/mediatek
git clone https://github.com/dfiloni/mt76.git
```

```
cd ../../../.. #bpi_r2_mt76/linux-mt/
patch -p1 < drivers/net/wireless/mediatek/mt76/kernel-patches/0001-add-basic-register-field-manipulation-macros.patch
nano drivers/net/wireless/mediatek/Makefile
#add: obj-$(CONFIG_MT76)     += mt76/
nano drivers/net/wireless/mediatek/Kconfig
#add before endif: before endif # WL_MEDIATEK: source "drivers/net/wireless/mediatek/mt76/Kconfig"
cd ..
./build.sh => 4
#networking support => wireless =>  <M>   Generic IEEE 802.11 Networking Stack (mac80211)
#Device Drivers  => Network device support => Wireless LAN => [*]   Mediatek Wireless LAN support => <M>   MediaTek MT76x2 802.11ac chips support
./build.sh => 1
cp SD/BPI-BOOT/bananapi/bpi-r2/linux/uImage /media/$USER/BPI-BOOT/bananapi/bpi-r2/linux/uImage
sudo cp -r SD/BPI-ROOT/lib/modules /media/$USER/BPI-ROOT/lib/
cp linux-mt/drivers/net/wireless/mediatek/mt76/firmware/* /media/$USER/BPI-ROOT/lib/firmware/
#scp linux-mt/drivers/net/wireless/mediatek/mt76/firmware/* root@192.168.0.10:/lib/firmware/
sync
```

4.14

Kernel 4.14 (in progress...):

import

PCIe-patch

, if not done yet

```
patch -p1 < pcie.patch
cd drivers/net/wireless/mediatek/
git clone https://github.com/openwrt/mt76.git
```

- in mt76/mt7603.h missing „#include <linux/interrupt.h>“
- in mt76/mac80211.c missing „#include <linux/of.h>“
- in Makefile missing „CFLAGS_trace.o := -I$(src)“ und „CFLAGS_mt76x2_trace.o := -I$(src)“
- and include code in drivers/net/wireless/mediatek/Makefile

```
obj-$(CONFIG_MT76)  += mt76/
```

and drivers/net/wireless/mediatek/Kconfig

```
source "drivers/net/wireless/mediatek/mt76/Kconfig"
```

unpack

full driver-code for mt76x2 + mt76x3
to drivers/net/wireless/mediatek/

activate following modules in kernel:

```
CONFIG_MAC80211=m
CONFIG_CFG80211=m
CONFIG_MT76=m

#pcie
CONFIG_PCIEPORTBUS=y
CONFIG_PCIE_MEDIATEK=y
CONFIG_PHY_MTK_TPHY=y
```

copy the firmware

```
sudo cp drivers/net/wireless/mediatek/mt76/firmware/* /media/$USER/BPI-ROOT/lib/firmware/
```

configure

```
[10:50] root@bpi-r2:~# ifconfig -a |grep wlan
wlan1     Link encap:Ethernet  HWaddr f8:62:aa:50:12:1d  <<<
```

if your wlan-device-number is above 1, you can edit that with

```
nano /etc/udev/rules.d/70-persistent-net.rules
```

/etc/hostapd/hostapd.conf (change interface)

```
interface=wlan1
#interface=ap0
driver=nl80211

ssid=r2_AP1

hw_mode=g
channel=1
#macaddr_acl=0
auth_algs=1
#ignore_broadcast_ssid=0
#wpa=2
#wmm_enabled=1
#wpa_passphrase=12345678
#wpa_key_mgmt=WPA-PSK
#wpa_pairwise=TKIP
#rsn_pairwise=CCMP
```

/etc/hostapd/hostapd_wlan1.conf

start hostapd (Debugmode):

```
hostapd -dd /etc/hostapd/hostapd.conf
```

set IP-address and start dnsmasq: ip-configuration

MT7615

firmware-files to /lib/firmware/mediatek/

http://forum.banana-pi.org/t/802-11ac-4x4-standard-size-mini-pcie-card-is-launched/11545/15?u=frank-w

I have added mt7615 driver in 5.4 on nov 19 2020 (should be 5.4.78).

Mt7615 is separate kernel-module which needs to be loaded

```
modprobe mt7615e
```

dual-ap

Dbdc mode was added in 5.7

```
echo 1 > /sys/kernel/debug/ieee80211/phy0/mt76/dbdc
```

There is a report that dbdc cannot be enabled after reboot,but on my tests this works without problems

You can only enable 2.4ghz on phy0, and 5ghz on phy1 simultaneously.

antenna connectors

```
WF0-1 are for phy0, and WF2-3 are for phy1.
```

http://forum.banana-pi.org/t/802-11ac-4x4-standard-size-mini-pcie-card-is-launched/11545/16

HostAPd

/etc/hostapd/hostapd.conf

/etc/hostapd/hostapd_wlan1.conf

5GHz

```
apt-get install iw
iw list | grep "Supported interface modes" -A 8
```

```
apt-get install iw wireless-regdb crda
```

country code

setting Country-Code (regulary domain) can be a bit tricky

```
iw reg set ISO_3166-1_alpha-2
iw reg set DE
iw reg get
```

wrong output:

```
global
country 00: DFS-UNSET
```

correct:

```
global
country US: DFS-ETSI
```

maybe try this:

```
modprobe cfg80211 ieee80211_regdom=US
COUNTRY=US crda
```

```
$ sudo nano /etc/modprobe.d/cfg80211.conf
options cfg80211 ieee80211_regdom=US
```

possible frequencies

```
iw list | grep MHz
```

Hostapd-Configuration

```
$ sudo nano /etc/hostapd/hostapd.conf
[...]
country_code=US
ieee80211n=1
ieee80211d=1
hw_mode=a
```

```
channel=48
[...]
```

Config-Examples taken from here (DE)

CountryCode-List

IP-Configuration

set IP-address:

```
#set IP and start
ip addr add 192.168.10.1/24 dev ap0
#ip link set dev ap0 up

ip addr add 192.168.11.1/24 dev wlan1
```

/etc/dnsmasq.conf (activate line by removing # on begin of line)

```
conf-dir=/etc/dnsmasq.d
```

/etc/dnsmasq.d/interfaces.conf

```
#interface=eth0
interface=wlan0
#interface=eth1
interface=ap0

# DHCP-Server not active for Interface
#no-dhcp-interface=ppp0
no-dhcp-interface=eth0
no-dhcp-interface=eth1

#dhcp-authoritative
dhcp-range=ap0,192.168.10.100,192.168.10.150,255.255.255.0,48h
dhcp-option=ap0,3,192.168.10.1
dhcp-range=wlan1,192.168.11.100,192.168.11.150,255.255.255.0,48h
dhcp-option=wlan1,3,192.168.11.1

#special handling of some hosts (here using another dns-server)
#dhcp-host=ap0,x:y:z:x:y:z,frank,set:specialhosts,192.168.10.27,12h
dhcp-option=tag:specialhosts,option:dns-server,192.168.0.11,8.8.4.4
```

/etc/dnsmasq.d/interfaces.conf

```
service dnsmasq start
```

Routing

```
nano /etc/sysctl.conf
#activate net.ipv4.ip_forward=1 and net.ipv6.conf.all.forwarding=1 by
removing # at beginning of line
sysctl -p /etc/sysctl.conf
```

if the BPI-R2 is not the main-router (access to internet), it is neccessary, that this main-router knows the networks (wlan) behind the R2.

The following commands must be entered on a (Debian-)Router, to append the networks to its routing-table (will be lost on reboot if not executed on boot):

```
route add -net 192.168.10.0 netmask 255.255.255.0 gw 192.168.0.10
route add -net 192.168.11.0 netmask 255.255.255.0 gw 192.168.0.10
```

here is 192.168.10.0 the 1st WLAN, 192.168.11.0 the 2nd WLAN and 192.168.0.10 is the LAN-IP of the BPI-R2 (same subnet as LAN-IP from main-router)

From:
https://wiki.fw-web.de/ - **FW-WEB Wiki**

Permanent link:
https://wiki.fw-web.de/doku.php?id=en:bpi-r2:wlan

Last update: **2023/06/08 17:06**

X-Server

```
sudo apt-get install xserver-xorg lxde xfonts-base (policykit-1) (xinit)
```

to install very basic xserver

log:

```
less /var/log/Xorg.0.log
```

list resolutions supported by fbdev:

```
cat /sys/class/graphics/fb0/modes
```

„no session for pid“ comes only up if xserver is started via startx. after install make a reboot the lightdm-login-form comes up and after login no error-message appears.

From:
https://wiki.fw-web.de/ - **FW-WEB Wiki**

Permanent link:
https://wiki.fw-web.de/doku.php?id=en:bpi-r2:x-server

Last update: **2023/06/08 17:06**