How to Integrate the 60 Series Wi-Fi Module with Open WRT

60 Series Radios

Application Note *v2.0*

1 Introduction

This guide provides an example on how to integrate a Ezurio ST60-2230C-P Wi-Fi Module with a Raspberry Pi 4 Compute Module on OpenWRT.

Table 1: ST60 variants

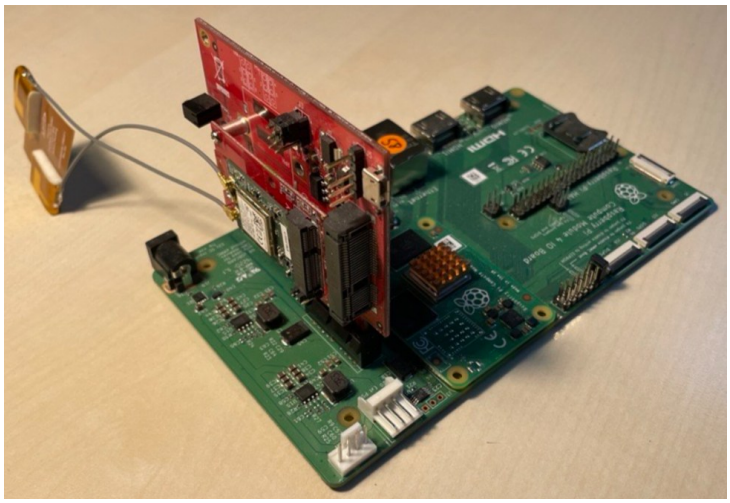
Module Name	Form Factor	Wi-Fi	BT	Part Number
ST60-2230C-P	M.2 2230	*PCIE*	*UART*	ST60-2230C-P

Note: This guide deals with the integration of the Wi-Fi portion of the ST60 module only.

2 Requirements

- ST60-2230C-P Wi-Fi module
- Suitable Wifi antenna – e.g. EFD2455A3S-10MH4L
- Raspberry Pi Compute Module 4
- Raspberry Pi Compute Module 4 IO Board
- PCIe x1 to mPCIe adapter – e.g. http://www.lycom.com.tw/DT-117.htm
- mPCIe to M2 adapter – e.g. https://www.getgoods.com/at-en/products/550535/Renkforce-GBIC-1x-M.2-NGFF-1x-Mini-PCI-Express.html
- Monitor, keyboard, mouse to access Raspeberry Pi
- Development PC with Yocto build environment

Below is a fully equipped RPi4CM board:



3 Build Environment Preparation

1. On page https://openwrt.org/toh/start identify the latest supported OpenWRT release for the Raspberry Pi 4 Compute Module.

Brand	Model	Version	Supported Current...	Device Page
Brand	Raspberry Pi 4	Compute Module	Supported Current Re	
Raspberry Pi Foundation	Raspberry Pi 4	Compute Module 4	23.05.0	raspberry_pi

Note: observe the filters in the "Model" and "Version" fields.

2. Follow the instructions on page https://openwrt.org/docs/guide-developer/toolchain/use-buildsystem

```
# Download and update the sources
git clone https://git.openwrt.org/openwrt/openwrt.git
```

```
alex@ubuntu:~/Projects/buildroot$ git clone https://git.openwrt.org/openwrt/openwrt.git
Cloning into 'openwrt'...
remote: Enumerating objects: 9390, done.
remote: Counting objects: 100% (9390/9390), done.
remote: Compressing objects: 100% (3770/3770), done.
remote: Total 671524 (delta 6104), reused 7906 (delta 5212), pack-reused 662134
Receiving objects: 100% (671524/671524), 192.34 MiB | 6.42 MiB/s, done.
Resolving deltas: 100% (470283/470283), done.
Updating files: 100% (9737/9737), done.
```

```
cd openwrt
...
```

```
git pull
```

```
alex@ubuntu:~/Projects/buildroot/openwrt$ git pull
Already up to date.
```

```
# Select a specific code revision
git branch -a
```

```
alex@ubuntu:~/Projects/buildroot/openwrt$ git branch -a
* master
  remotes/origin/HEAD -> origin/master
  remotes/origin/lede-17.01
  remotes/origin/main
  remotes/origin/master
  remotes/origin/openwrt-18.06
  remotes/origin/openwrt-19.07
  remotes/origin/openwrt-21.02
  remotes/origin/openwrt-22.03
  remotes/origin/openwrt-23.05
```

```
git tag
...
v23.05.0
v23.05.0-rc1
v23.05.0-rc2
v23.05.0-rc3
v23.05.0-rc4
v23.05.1
v23.05.2
```

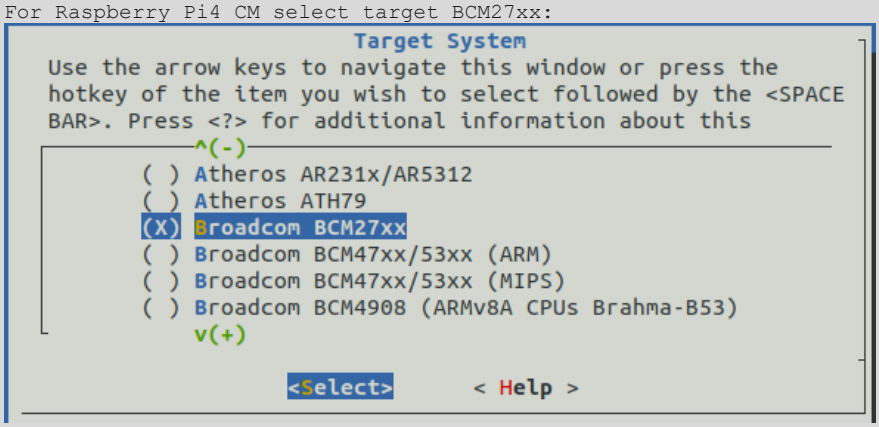
```
git checkout v23.05.0
alex@ubuntu:~/Projects/buildroot/openwrt$ git checkout v23.05.0
Note: switching to 'v23.05.0'.

You are in 'detached HEAD' state. You can look around, make experimental
changes and commit them, and you can discard any commits you make in this
wser e without impacting any branches by switching back to a branch.

If you want to create a new branch to retain commits you create, you may
do so (now or later) by using -c with the switch command. Example:

  git switch -c <new-branch-name>

Or undo this operation with:

  git switch -

Turn off this advice by setting config variable advice.detachedHead to false

HEAD is now at bd4f415efa OpenWrt v23.05.0: adjust config defaults
```

```
# Update the feeds
./scripts/feeds update -a
...
Create index file './feeds/routing.index'
Collecting package info: done
Collecting target info: done
Create index file './feeds/telephony.index'
Collecting package info: done
Collecting target info: done
```

```
./scripts/feeds install -a
...
Installing package 'sipgrep' from telephony
Installing package 'sipp' from telephony
Installing package 'siproxd' from telephony
Installing package 'sngrep' from telephony
Installing package 'yate' from telephony
```

```
# Configure the firmware image, mainly to select target
make menuconfig
alex@ubuntu:~/Projects/buildroot/openwrt$ make menuconfig
Collecting package info: done
tmp/.config-package.in:33662:error: recursive dependency detected!
tmp/.config-package.in:33662:   symbol PACKAGE_python3-pymysql depends on PYTHON3_PYMYSQL_SHA_PASSWORD_SUPPORT
feeds/packages/lang/python/pymysql/Config.in:4: symbol PYTHON3_PYMYSQL_SHA_PASSWORD_SUPPORT depends on PACKAGE_python3-pymysql
For a resolution refer to Documentation/kbuild/kconfig-language.rst
subsection "Kconfig recursive dependency limitations"
```

For Raspberry Pi4 CM select target BCM27xx:



For Raspberry Pi4 CM select subtarget BCM2711:

```
                        Subtarget
  Use the arrow keys to navigate this window or press the
  hotkey of the item you wish to select followed by the <SPACE
  BAR>. Press <?> for additional information about this

        ( ) BCM2708 boards (32 bit)
        ( ) BCM2709/BCM2710/BCM2711 boards (32 bit)
        ( ) BCM2710 boards (64 bit)
        (X) BCM2711 boards (64 bit)


                  <Select>      < Help >
```

For Raspberry Pi4 CM select target profile:

```
                      Target Profile
  Use the arrow keys to navigate this window or press the
  hotkey of the item you wish to select followed by the <SPACE
  BAR>. Press <?> for additional information about this

            ( ) Multiple devices
            (X) Raspberry Pi 4B/400/CM4 (64bit)


                  <Select>      < Help >
```

Add tar.gz target image format:

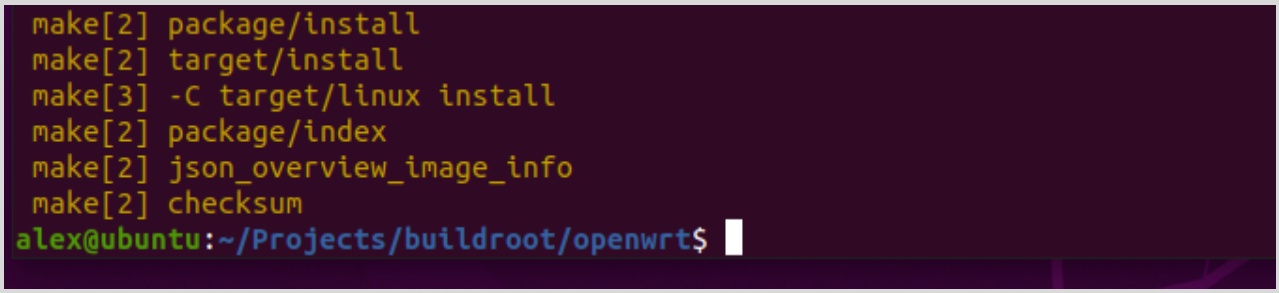
```
                                Target Images
e menu.  <Enter> selects submenus ---> (or empty subm
 includes, <N> excludes, <M> modularizes features.  P
*] built-in  [ ] excluded  <M> module  < > module cap

  [ ] ramdisk  ----
      *** Root filesystem archives ***
  [ ] cpio.gz
  [*] tar.gz
      *** Root filesystem images ***
  [*] ext4  --->
  [*] squashfs  --->
  [*] GZip images
      *** Image Options ***
  (64) Kernel partition size (in MiB)
  (104) Root filesystem partition size (in MiB)
  [ ] Make /var persistent
```

```
*** End of the configuration.
*** Execute 'make' to start the build or try 'make help'.
```

```
# Optional: configure the kernel (usually not required)
# Don't, unless have a strong reason to
make -j$(nproc) kernel_menuconfig
```

Note: As not necessary for the first build, this optional step from above OpenWRT guide has been skipped.

```
# Build the firmware image
make -j$(nproc) defconfig download clean world
...
```




```
Success.
```

Source: https://openwrt.org/docs/guide-developer/toolchain/use-buildsystem

Note: If the build does not complete due to missing packages (e.g. such as flex, bison, ncurses) on your Ubuntu development PC, please install them using "sudo apt-get install <package name>".

4 Locating the OS Image and Writing it to the RPi4 CM

1. Find the newly generated images at:

```
alex@ubuntu:~/Projects/buildroot/openwrt$ cd bin/targets/bcm27xx/bcm2711/

alex@ubuntu:~/Projects/buildroot/openwrt/bin/targets/bcm27xx/bcm2711$ ls -al
total 61580
drwxr-xr-x 3 alex alex     4096 Dez  6 14:53 .
drwxr-xr-x 3 alex alex     4096 Dez  6 11:59 ..
-rw-r--r-- 1 alex alex      130 Dez  6 14:25 config.buildinfo
-rw-r--r-- 1 alex alex      268 Dez  6 14:25 feeds.buildinfo
-rw-r--r-- 1 alex alex 14917525 Dez  6 14:53 openwrt-bcm27xx-bcm2711-rpi-4-ext4-factory.img.gz
-rw-r--r-- 1 alex alex 14918371 Dez  6 14:53 openwrt-bcm27xx-bcm2711-rpi-4-ext4-sysupgrade.img.gz
-rw-r--r-- 1 alex alex     3894 Dez  6 14:53 openwrt-bcm27xx-bcm2711-rpi-4.manifest
-rw-r--r-- 1 alex alex  5992927 Dez  6 14:52 openwrt-bcm27xx-bcm2711-rpi-4-rootfs.tar.gz
-rw-r--r-- 1 alex alex 13589896 Dez  6 14:53 openwrt-bcm27xx-bcm2711-rpi-4-squashfs-factory.img.gz
-rw-r--r-- 1 alex alex 13590742 Dez  6 14:53 openwrt-bcm27xx-bcm2711-rpi-4-squashfs-sysupgrade.img.gz
drwxr-xr-x 2 alex alex     4096 Dez  6 14:53 packages
-rw-r--r-- 1 alex alex     2140 Dez  6 14:53 profiles.json
-rw-r--r-- 1 alex alex     1022 Dez  6 14:53 sha256sums
-rw-r--r-- 1 alex alex       18 Dez  6 14:25 version.buildinfo
```

2. Extract the openwrt-bcm27xx-bcm2711-rpi-4-ext4-factory.img.gz archive:

```
alex@ubuntu:~/Projects/buildroot/openwrt/bin/targets/bcm27xx/bcm2711$ gzip -d openwrt-bcm27xx-bcm2711-rpi-
4-ext4-factory.img.gz
alex@ubuntu:~/Projects/buildroot/openwrt/bin/targets/bcm27xx/bcm2711$ ls -al
total 227236
drwxr-xr-x 3 alex alex      4096 Dez  7 10:01 .
drwxr-xr-x 3 alex alex      4096 Dez  6 11:59 ..
-rw-r--r-- 1 alex alex       130 Dez  6 14:25 config.buildinfo
-rw-r--r-- 1 alex alex       268 Dez  6 14:25 feeds.buildinfo
-rw-r--r-- 1 alex alex 184549376 Dez  6 14:53 openwrt-bcm27xx-bcm2711-rpi-4-ext4-factory.img
-rw-r--r-- 1 alex alex  14918371 Dez  6 14:53 openwrt-bcm27xx-bcm2711-rpi-4-ext4-sysupgrade.img.gz
-rw-r--r-- 1 alex alex      3894 Dez  6 14:53 openwrt-bcm27xx-bcm2711-rpi-4.manifest
-rw-r--r-- 1 alex alex   5992927 Dez  6 14:52 openwrt-bcm27xx-bcm2711-rpi-4-rootfs.tar.gz
-rw-r--r-- 1 alex alex  13589896 Dez  6 14:53 openwrt-bcm27xx-bcm2711-rpi-4-squashfs-factory.img.gz
-rw-r--r-- 1 alex alex  13590742 Dez  6 14:53 openwrt-bcm27xx-bcm2711-rpi-4-squashfs-sysupgrade.img.gz
drwxr-xr-x 2 alex alex      4096 Dez  6 14:53 packages
-rw-r--r-- 1 alex alex      2140 Dez  6 14:53 profiles.json
-rw-r--r-- 1 alex alex      1022 Dez  6 14:53 sha256sums
-rw-r--r-- 1 alex alex        18 Dez  6 14:25 version.buildinfo
```

4.1 Connecting the eMMC storage as USB drive to flash the image

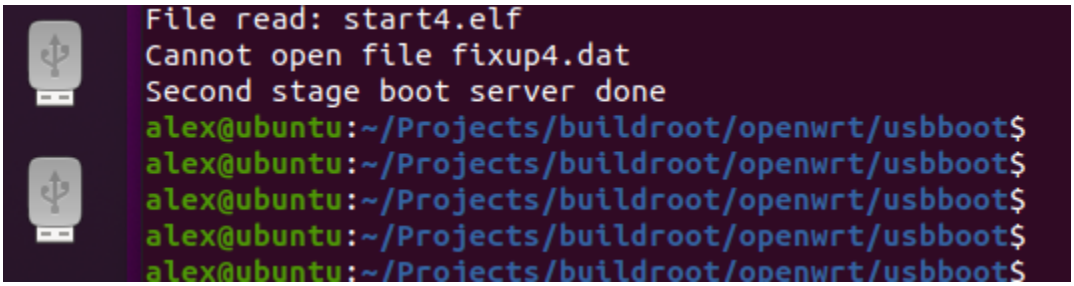
You should follow the instructions on this page:

https://www.jeffgeerling.com/blog/2020/how-flash-raspberry-pi-os-compute-module-4-emmc-usbboot

to connect the eMMC storage of the RPi4CM as a USB drive to flash the OpenWRT image to it.

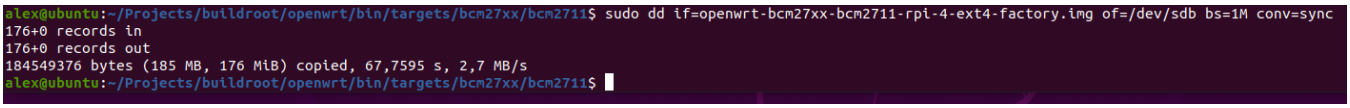
```
alex@ubuntu:~/Projects/buildroot/openwrt$ git clone --depth=1 https://github.com/raspberrypi/usbboot
Cloning into 'usbboot'...
remote: Enumerating objects: 94, done.
remote: Counting objects: 100% (94/94), done.
remote: Compressing objects: 100% (77/77), done.
remote: Total 94 (delta 7), reused 56 (delta 2), pack-reused 0
Unpacking objects: 100% (94/94), 88.62 MiB | 4.65 MiB/s, done.

alex@ubuntu:~/Projects/buildroot/openwrt/usbboot$ make
cc -Wall -Wextra -g -o bin2c bin2c.c
./bin2c msd/bootcode.bin msd/bootcode.h
./bin2c msd/start.elf msd/start.h
./bin2c msd/bootcode4.bin msd/bootcode4.h
./bin2c msd/start4.elf msd/start4.h
cc -Wall -Wextra -g -o rpiboot main.c `pkg-config --cflags --libs libusb-1.0` -DGIT_VER="\"922f3863\"" -
DPKG_VER="\"20221215~105525\""

alex@ubuntu:~/Projects/buildroot/openwrt/usbboot$ ls
bin2c    bootcode4.bin  docs          fmemopen.c  main.c    mass-storage-gadget  Readme.md  recovery.bin  rpi-
imager-embedded  secure-boot-msd       tools
bin2c.c  debian         eeprom-erase  LICENSE     Makefile  msd                  recovery   rpiboot
secure-boot-example  secure-boot-recovery  win32

alex@ubuntu:~/Projects/buildroot/openwrt/usbboot$ sudo ./rpiboot
RPIBOOT: build-date Dec 12 2023 version 20221215~105525 922f3863
Waiting for BCM2835/6/7/2711...
Loading embedded: bootcode4.bin
Sending bootcode.bin
Successful read 4 bytes
Waiting for BCM2835/6/7/2711...
Loading embedded: bootcode4.bin
Second stage boot server
Cannot open file config.txt
Cannot open file pieeprom.sig
Loading embedded: start4.elf
File read: start4.elf
Cannot open file fixup4.dat
Second stage boot server done
```

Now you should see the icons for the internal RPi4CM eMMC memory appearing:



Writing the image to the eMMC:

```
alex@ubuntu:~/Projects/buildroot/openwrt/bin/targets/bcm27xx/bcm2711$ sudo dd if=openwrt-bcm27xx-bcm2711-rpi-4-
ext4-factory.img of=/dev/sdb bs=1M conv=sync
176+0 records in
176+0 records out
184549376 bytes (185 MB, 176 MiB) copied, 67,7595 s, 2,7 MB/s
alex@ubuntu:~/Projects/buildroot/openwrt/bin/targets/bcm27xx/bcm2711$
```

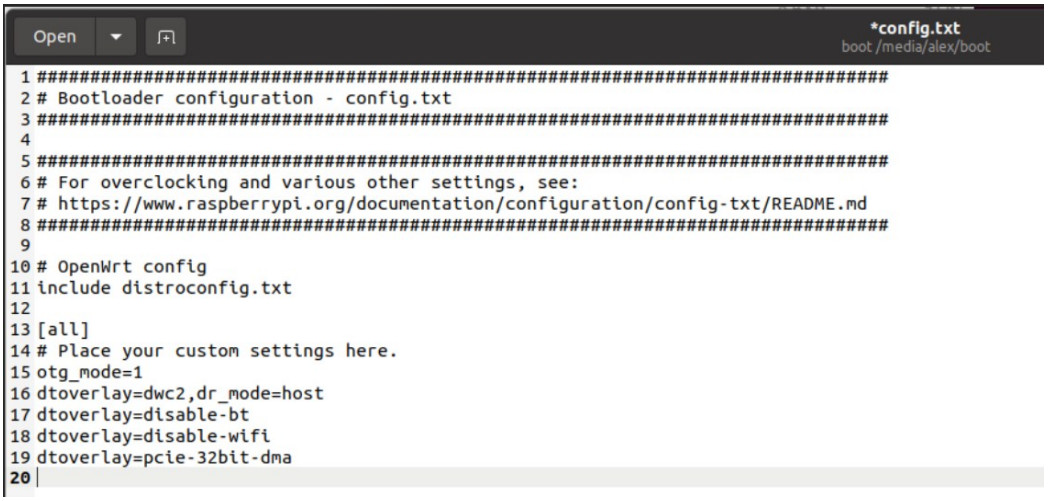


5 Preparing for Boot and First Boot

5.1 Customizing Device Tree settings

In the Boot partition of the eMMC memory of the RPi4CM make the following changes to the config.txt file:

```
################################################################################
# Bootloader configuration - config.txt
################################################################################

################################################################################
# For overclocking and various other settings, see:
# https://www.raspberrypi.org/documentation/configuration/config-txt/README.md
################################################################################

# OpenWrt config
include distroconfig.txt

[all]
# Place your custom settings here.
otg_mode=1
dtoverlay=dwc2,dr_mode=host
dtoverlay=disable-bt
dtoverlay=disable-wifi
dtoverlay=pcie-32bit-dma
```



"dtoverlay=dwc2,dr_mode=host" and "otg_mode=1" are necessary to enable the USB host ports to be able to operate the RPi4CM with a USB keyboard.

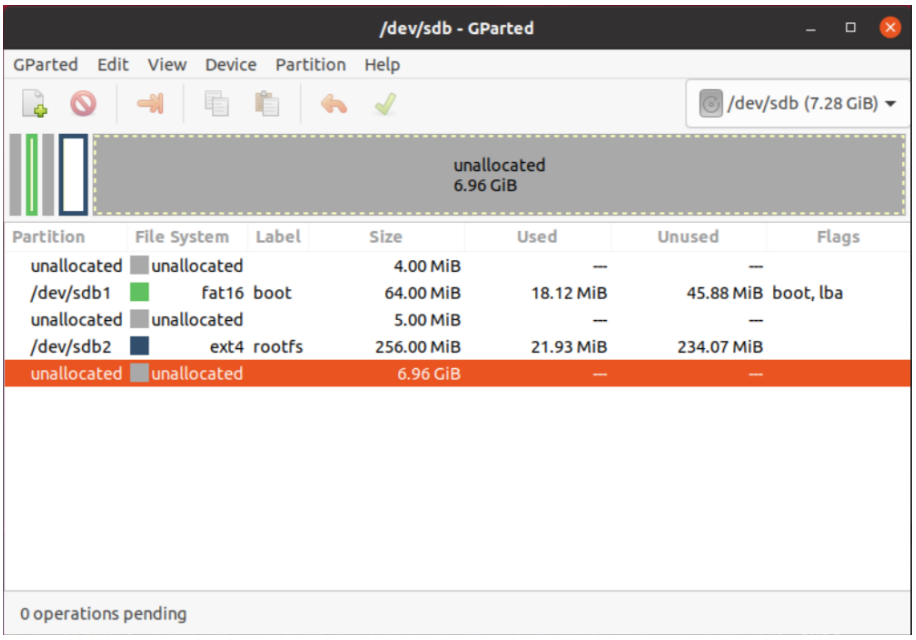
"dtoverlay=pcie-32bit-dma" is required to enable 32bit DMA to operate the ST60-2230-P Wifi module.

"dtoverlay=disable-wifi" and "dtoverlay=disable-bt" is to disable possible on-board radios.

Note: Make sure to save the changes and reboot.

5.2 Increasing rootfs Partition

Use "sudo gparted" to increase the size of the rootfs partition as the original size of around 100Mbyte is not enough to host the additional software that needs to be installed. In this example the size of the partition has been increased to 256Mbyte.



Note: Be *very careful* to perform this action with the correct drive! Failure to do so might result in a bricked development PC.

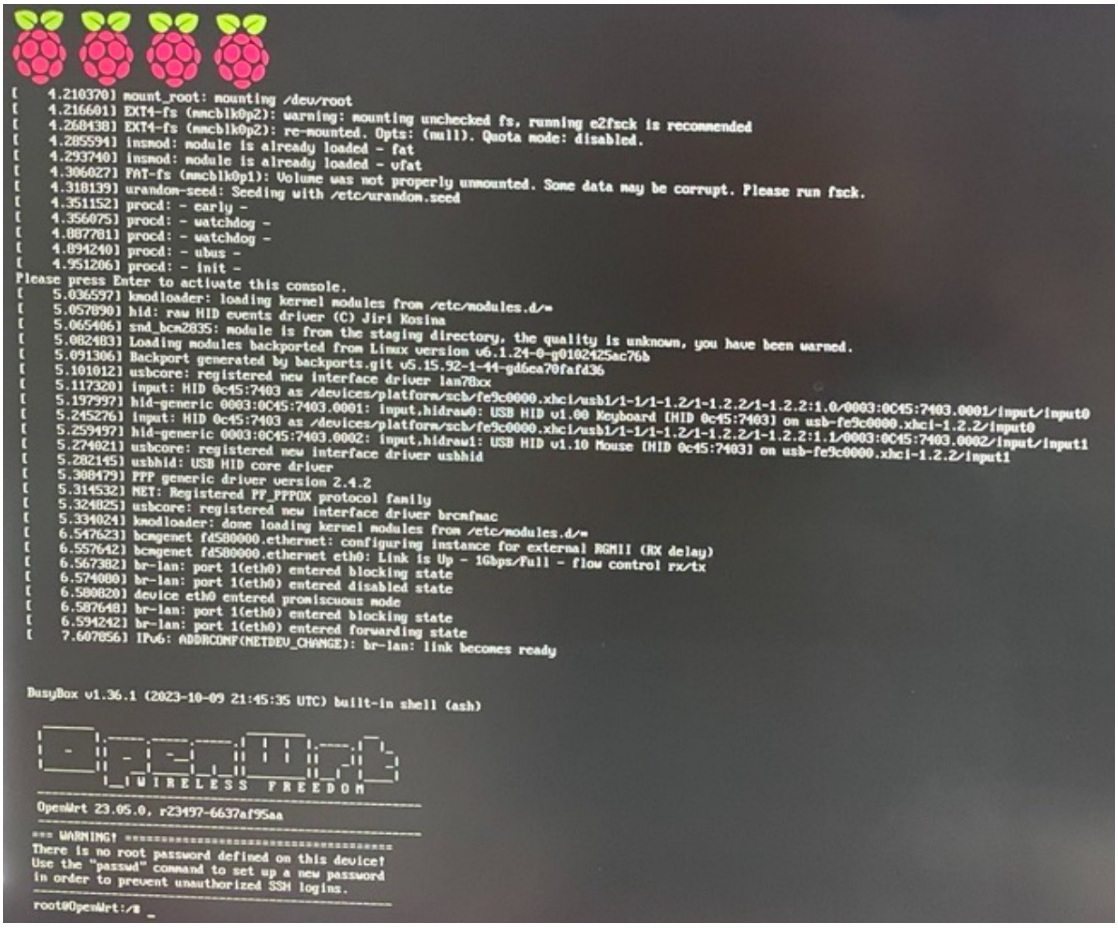
Increasing the partition size to 512MByte or larger was unsuccessful. But resolving this is beyond the scope of this guide.

5.3 First Boot

Remove the jumper from J2 and the micro-USB cable that was used to flash the RPi4CM.

Note: The function of J2 and the USB cable is described in above mentioned guide from Jeff Geerling.

Connect a HDMI monitor, a keyboard, a 12V power supply and power the board. You should see a screen like the following on your HDMI monitor connected to your RPi4CM:



Note: observe the "Please press Enter to activate this console" message.

6 Driver generation and Firmware placement

6.1 Cross-compiling the drivers

In the OpenWRT top directory make a directory to place the backport driver sources in and change to it:

```
alex@ubuntu:~/Projects/buildroot/openwrt$ mkdir backports
alex@ubuntu:~/Projects/buildroot/openwrt$ cd backports
alex@ubuntu:~/Projects/buildroot/openwrt/backports$
```

Create a file to place the necessary environment variables in and source it:

```
alex@ubuntu:~/Projects/buildroot/openwrt/backports$ cat openwrt_env
# RPi4cm
ARCH=arm64
CROSS_COMPILE=/home/alex/Projects/buildroot/openwrt/staging_dir/toolchain-aarch64_cortex-a72_gcc-
12.3.0_musl/bin/aarch64-openwrt-linux-
KLIB_BUILD=/home/alex/Projects/buildroot/openwrt/build_dir/target-aarch64_cortex-a72_musl/linux-
bcm27xx_bcm2711/linux-5.15.134
KLIB=/home/alex/Projects/buildroot/openwrt/staging_dir/target-aarch64_cortex-a72_musl/root-bcm27xx/lib/modules

alex@ubuntu:~/Projects/buildroot/openwrt/backports$ source openwrt_env
```

To check all environment variable are correct:

```
alex@ubuntu:~/Projects/buildroot/openwrt/backports$ echo $ARCH
arm64

alex@ubuntu:~/Projects/buildroot/openwrt/backports$ echo $CROSS_COMPILE
/home/alex/Projects/buildroot/openwrt/staging_dir/toolchain-aarch64_cortex-a72_gcc-12.3.0_musl/bin/aarch64-
openwrt-linux-

alex@ubuntu:~/Projects/buildroot/openwrt/backports$ echo $KLIB_BUILD
/home/alex/Projects/buildroot/openwrt/build_dir/target-aarch64_cortex-a72_musl/linux-bcm27xx_bcm2711/linux-
5.15.134

alex@ubuntu:~/Projects/buildroot/openwrt/backports$ echo $KLIB
/home/alex/Projects/buildroot/openwrt/staging_dir/target-aarch64_cortex-a72_musl/root-bcm27xx/lib/modules
```

Fetch the backport driver sources:

```
alex@ubuntu:~/Projects/buildroot/openwrt/backports$ wget https://github.com/LairdCP/Sterling-60-Release-
Packages/releases/download/LRD-REL-11.39.0.23/backports-laird-11.39.0.23.tar.bz2
--2023-12-12 15:20:06--  https://github.com/LairdCP/Sterling-60-Release-Packages/releases/download/LRD-REL-
11.39.0.23/backports-laird-11.39.0.23.tar.bz2
Resolving github.com (github.com)... 140.82.121.3
Connecting to github.com (github.com)|140.82.121.3|:443... connected.
HTTP request sent, awaiting response... 302 Found
...
Saving to: ‘backports-laird-11.39.0.23.tar.bz2’

backports-laird-11.39.0.23.tar.bz2
100%[=============================================================================>]  10,46M  12,3MB/s    in
0,9s

2023-12-12 15:20:08 (12,3 MB/s) - ‘backports-laird-11.39.0.23.tar.bz2’ saved [10965529/10965529]

alex@ubuntu:~/Projects/buildroot/openwrt/backports$ ls
backports-laird-11.39.0.23.tar.bz2  openwrt_env
```

Note: At the time of writing, backports version 11.39.0.23 was the latest. For other releases please adapt the “wget link”.

Un-zip and un-tar the archive:

```
alex@ubuntu:~/Projects/buildroot/openwrt/backports$ bunzip2 backports-laird-11.39.0.23.tar.bz2

alex@ubuntu:~/Projects/buildroot/openwrt/backports$ ls
backports-laird-11.39.0.23.tar  openwrt_env

alex@ubuntu:~/Projects/buildroot/openwrt/backports$ tar xf backports-laird-11.39.0.23.tar -C .

alex@ubuntu:~/Projects/buildroot/openwrt/backports$ ls
backports-laird-11.39.0.23.tar  laird-backport-11.39.0.23  openwrt_env
```

Change to the backport directory and issue:

```
alex@ubuntu:~/Projects/buildroot/openwrt/backports$ cd laird-backport-11.39.0.23/
alex@ubuntu:~/Projects/buildroot/openwrt/backports/laird-backport-11.39.0.23$

alex@ubuntu:~/Projects/buildroot/openwrt/backports/laird-backport-11.39.0.23$ make clean

alex@ubuntu:~/Projects/buildroot/openwrt/backports/laird-backport-11.39.0.23$ make defconfig-sterling60_nbt
cc -Wall -Wmissing-prototypes -Wstrict-prototypes -O2 -fomit-frame-pointer   -c -o conf.o conf.c
flex -ozconf.lex.c -L zconf.l
bison -ozconf.tab.c -t -l zconf.y
cc -Wall -Wmissing-prototypes -Wstrict-prototypes -O2 -fomit-frame-pointer   -c -o zconf.tab.o zconf.tab.c
cc   conf.o zconf.tab.o   -o conf
#
# configuration written to .config
#

alex@ubuntu:~/Projects/buildroot/openwrt/backports/laird-backport-11.39.0.23$ make
make[5]: 'conf' is up to date.
#
# configuration written to .config
#
Building backport-include/backport/autoconf.h ... done.
  CC [M]  /home/alex/Projects/buildroot/openwrt/backports/laird-backport-11.39.0.23/compat/main.o
  CC [M]  /home/alex/Projects/buildroot/openwrt/backports/laird-backport-11.39.0.23/compat/lib-crypto-arc4.o
  LD [M]  /home/alex/Projects/buildroot/openwrt/backports/laird-backport-11.39.0.23/compat/compat.o

...

  CC [M]  /home/alex/Projects/buildroot/openwrt/backports/laird-backport-11.39.0.23/net/wireless/cfg80211.mod.o
  LD [M]  /home/alex/Projects/buildroot/openwrt/backports/laird-backport-11.39.0.23/net/wireless/cfg80211.ko
  BTF [M] /home/alex/Projects/buildroot/openwrt/backports/laird-backport-11.39.0.23/net/wireless/cfg80211.ko
Skipping BTF generation for /home/alex/Projects/buildroot/openwrt/backports/laird-backport-
11.39.0.23/net/wireless/cfg80211.ko due to unavailability of vmlinux
alex@ubuntu:~/Projects/buildroot/openwrt/backports/laird-backport-11.39.0.23$ find . -name *.ko
./compat/compat.ko
./drivers/net/wireless/laird/lrdmwl/lrdmwl_sdio.ko
./drivers/net/wireless/laird/lrdmwl/lrdmwl.ko
./drivers/net/wireless/laird/lrdmwl/lrdmwl_pcie.ko
./drivers/net/wireless/laird/lrdmwl/lrdmwl_usb.ko
./net/mac80211/mac80211.ko
./net/wireless/cfg80211.ko
```

Notes: If an error is thrown because packages like “flex” or “bison” are missing on your development PC, simply install them with e.g., *sudo apt-get install flex bison*

When building for the first time the output might look slightly different.

The “nbt” in stands for “No-Blue-Tooth”. We only integrate for Wi-Fi; hence BT is not needed.

The “make” process might take quite a bit of time.

6.2 Transferring the drivers and firmware to the target

After build process has finished, issue:

```
alex@ubuntu:~/Projects/buildroot/openwrt/backports/laird-backport-11.39.0.23$ find . -name '*.ko' | xargs tar
cvf backports-openwrt-st60.tar
./compat/compat.ko
./drivers/net/wireless/laird/lrdmwl/lrdmwl_sdio.ko
./drivers/net/wireless/laird/lrdmwl/lrdmwl.ko
./drivers/net/wireless/laird/lrdmwl/lrdmwl_pcie.ko
./drivers/net/wireless/laird/lrdmwl/lrdmwl_usb.ko
./net/mac80211/mac80211.ko
./net/wireless/cfg80211.ko
alex@ubuntu:~/Projects/buildroot/openwrt/backports/laird-backport-11.39.0.23$ ls *.tar
backports-openwrt-st60.tar
```

Download the respective ST60-2230-P firmware file from Github:

```
alex@ubuntu:~/Projects/buildroot/openwrt/backports/laird-backport-11.39.0.23$ wget
https://github.com/LairdCP/Sterling-60-Release-Packages/releases/download/LRD-REL-11.39.0.23/laird-60-radio-
firmware-pcie-uart-11.39.0.23.tar.bz2
```

Extract the .bz2 file:

```
alex@ubuntu:~/Projects/buildroot/openwrt/backports/laird-backport-11.39.0.23$ bunzip2 laird-60-radio-firmware-
pcie-uart-11.39.0.23.tar.bz2
alex@ubuntu:~/Projects/buildroot/openwrt/backports/laird-backport-11.39.0.23$ ls *.tar
backports-openwrt-st60.tar  laird-60-radio-firmware-pcie-uart-11.39.0.23.tar
```

and copy the .tar files to a USB stick to copy them to your target:

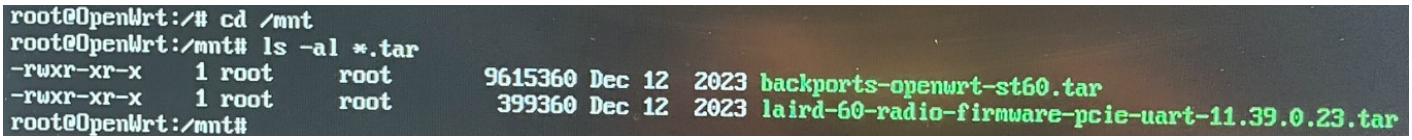
```
alex@ubuntu:~/Projects/buildroot/openwrt/backports/laird-backport-11.39.0.23$ sudo cp *.tar /media/alex/LAIRD/

alex@ubuntu:~/Projects/buildroot/openwrt/backports/laird-backport-11.39.0.23$ ls /media/alex/LAIRD/*.tar
/media/alex/LAIRD/backports-openwrt-st60.tar  /media/alex/LAIRD/laird-60-radio-firmware-pcie-uart-
11.39.0.23.tar
```

Mount that USB stick on your target RPi4CM and install both tar files:

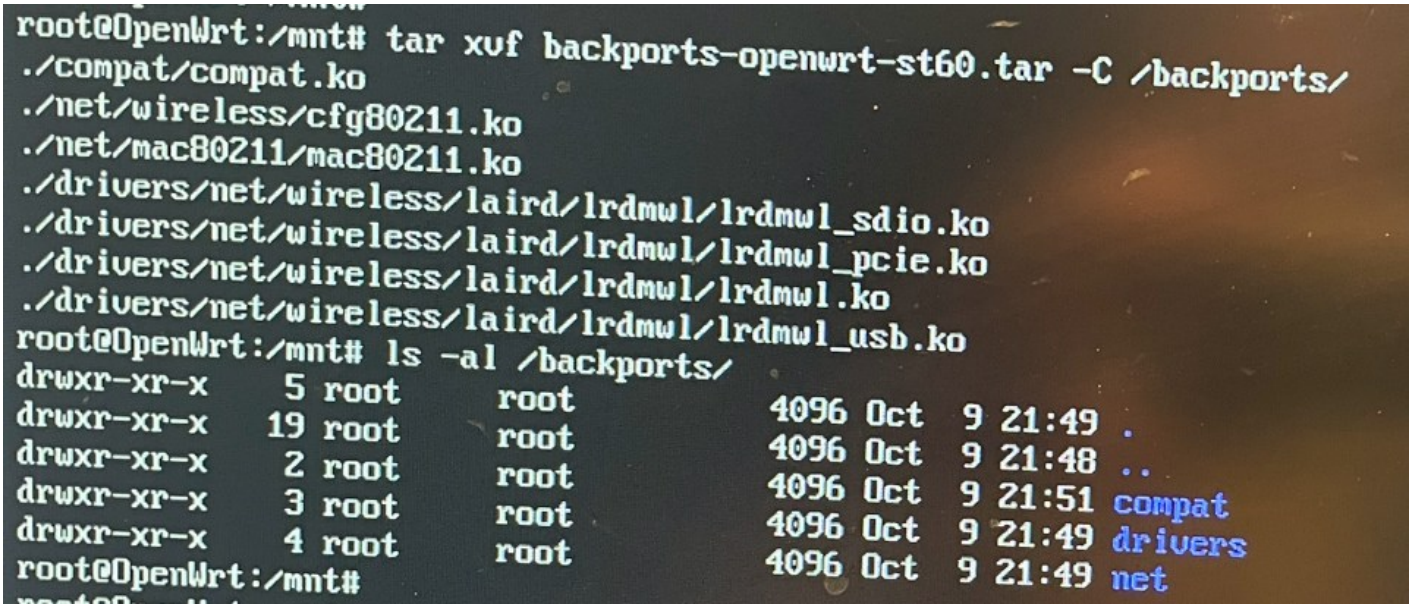
```
mount /dev/sda1 /mnt
```

```
root@OpenWrt:/# mount /dev/sda1 /mnt
[   60.351432] F2FS-fs (sda1): Magic Mismatch, valid(0xf2f52010) - read(0x0)
[   60.360171] F2FS-fs (sda1): Can't find valid F2FS filesystem in 1th superblock
[   60.369811] F2FS-fs (sda1): Magic Mismatch, valid(0xf2f52010) - read(0x0)
[   60.378054] F2FS-fs (sda1): Can't find valid F2FS filesystem in 2th superblock
[   60.389940] FAT-fs (sda1): Volume was not properly unmounted. Some data may be corrupt. Please run fsck.
root@OpenWrt:/#
```

```
cd /mnt
ls -al *.tar
```

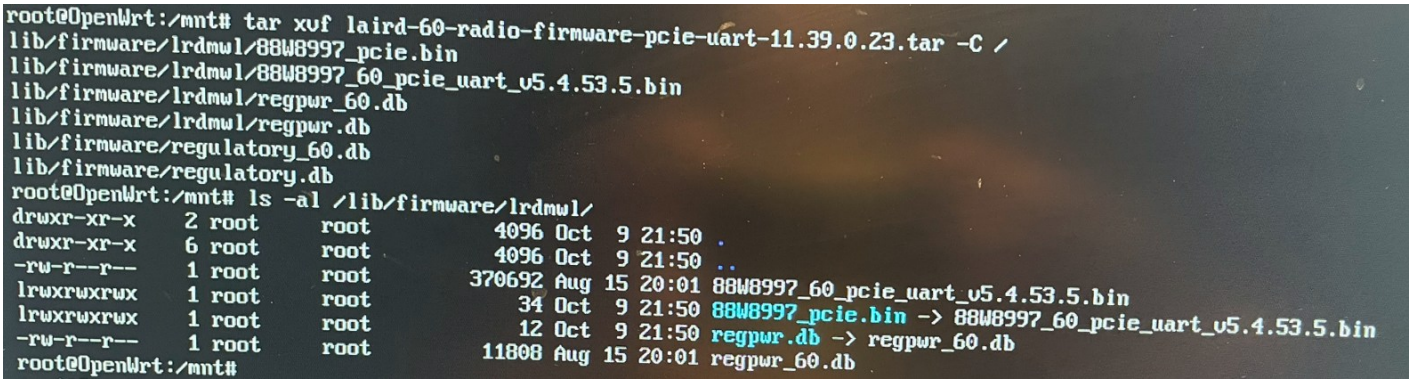



```
tar xvf backports-openwrt-st60.tar -C /backports/
```



Note: Files "lrdmwl_sdio.ko" and "lrdmwl_usb.ko" are not required and can be left out or deleted for the sake of saving space on the RPi4CMs eMMC storage.

```
tar xvf laird-60-radio-firmware-pcie-uart-11.39.0.23.tar -C /
```

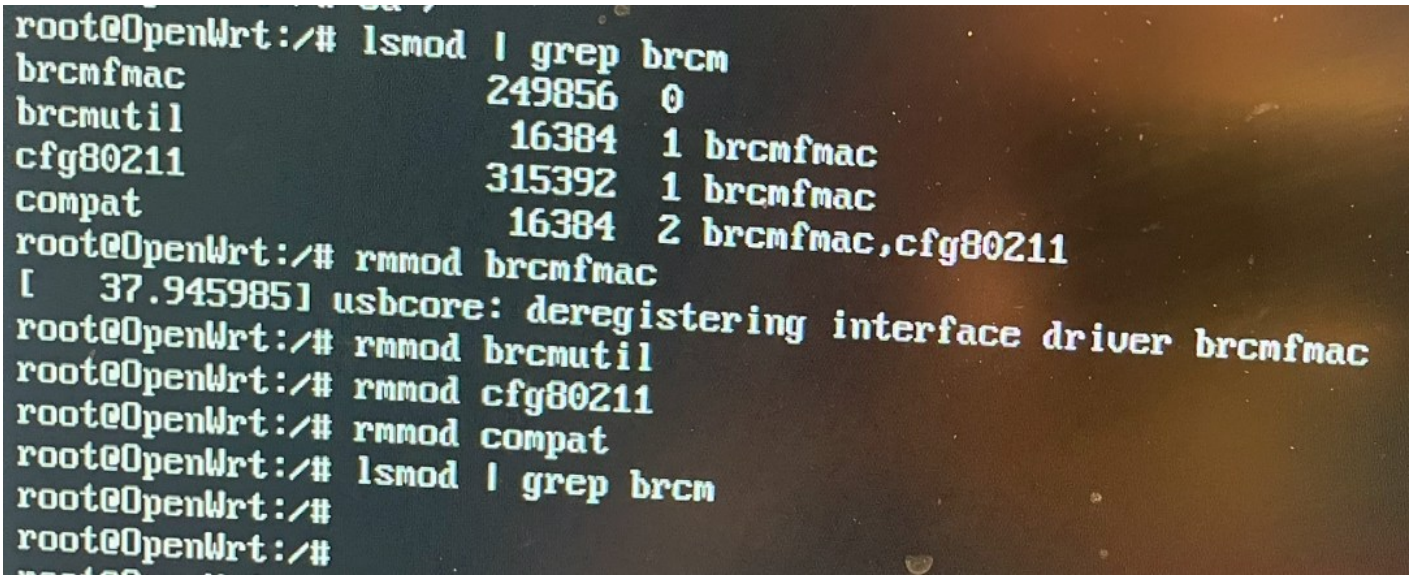


7 Bringing Up the Wi-Fi Interface

7.1 Removing already loaded drivers (aka kernel modules)

In order for the Ezurio backport drivers to be loaded, already inserted drivers need to be removed:

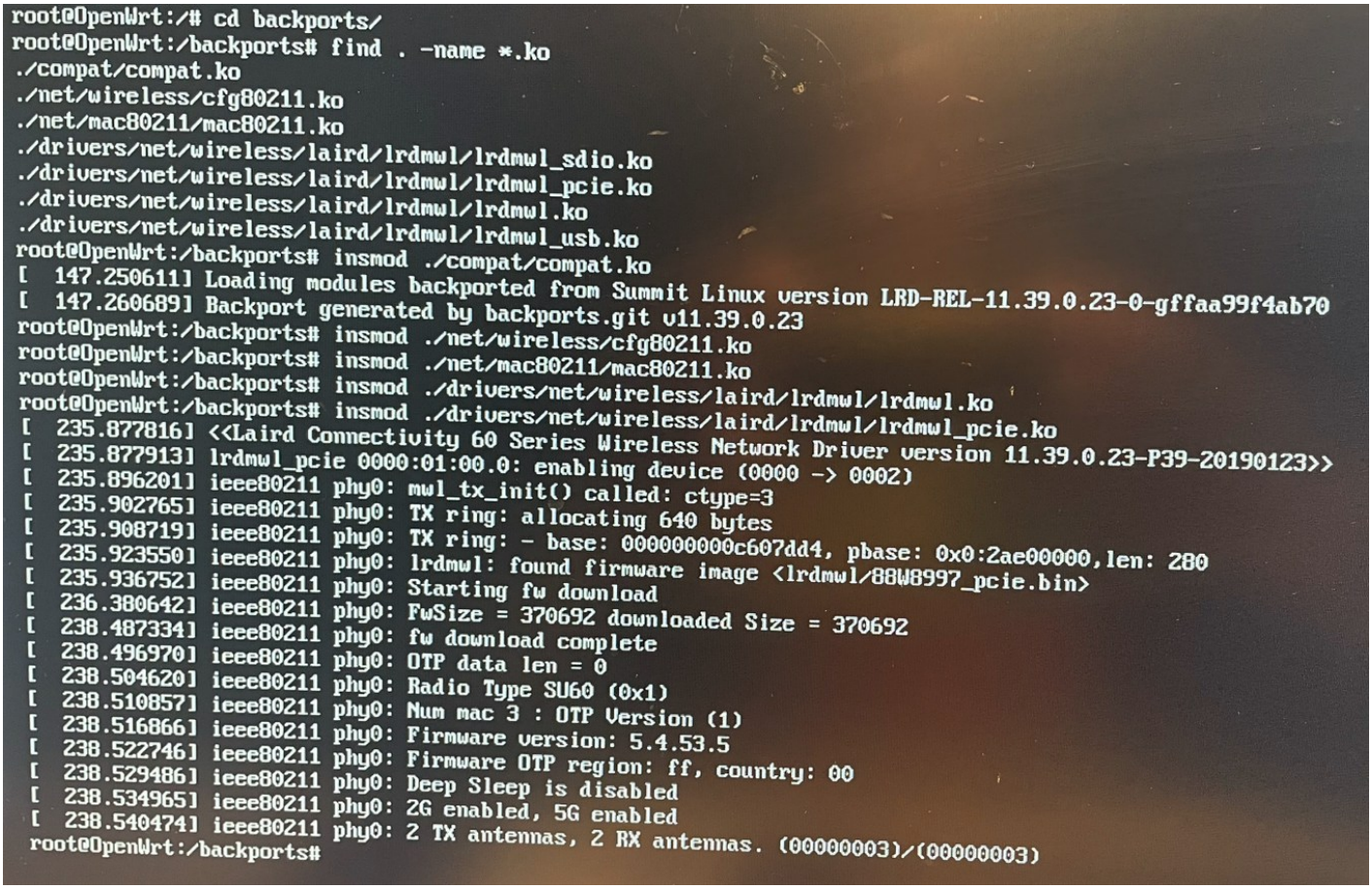
```
lsmod | grep brcm
rmmod brcmfmac
rmmod brcmutil
rmmod cfg80211
rmmod compat
lsmod | grep brcm
```



Note: There is no difference in the names of each driver between the originally loaded drivers and the Ezurio ones. Care needs to be taken not to confuse them.

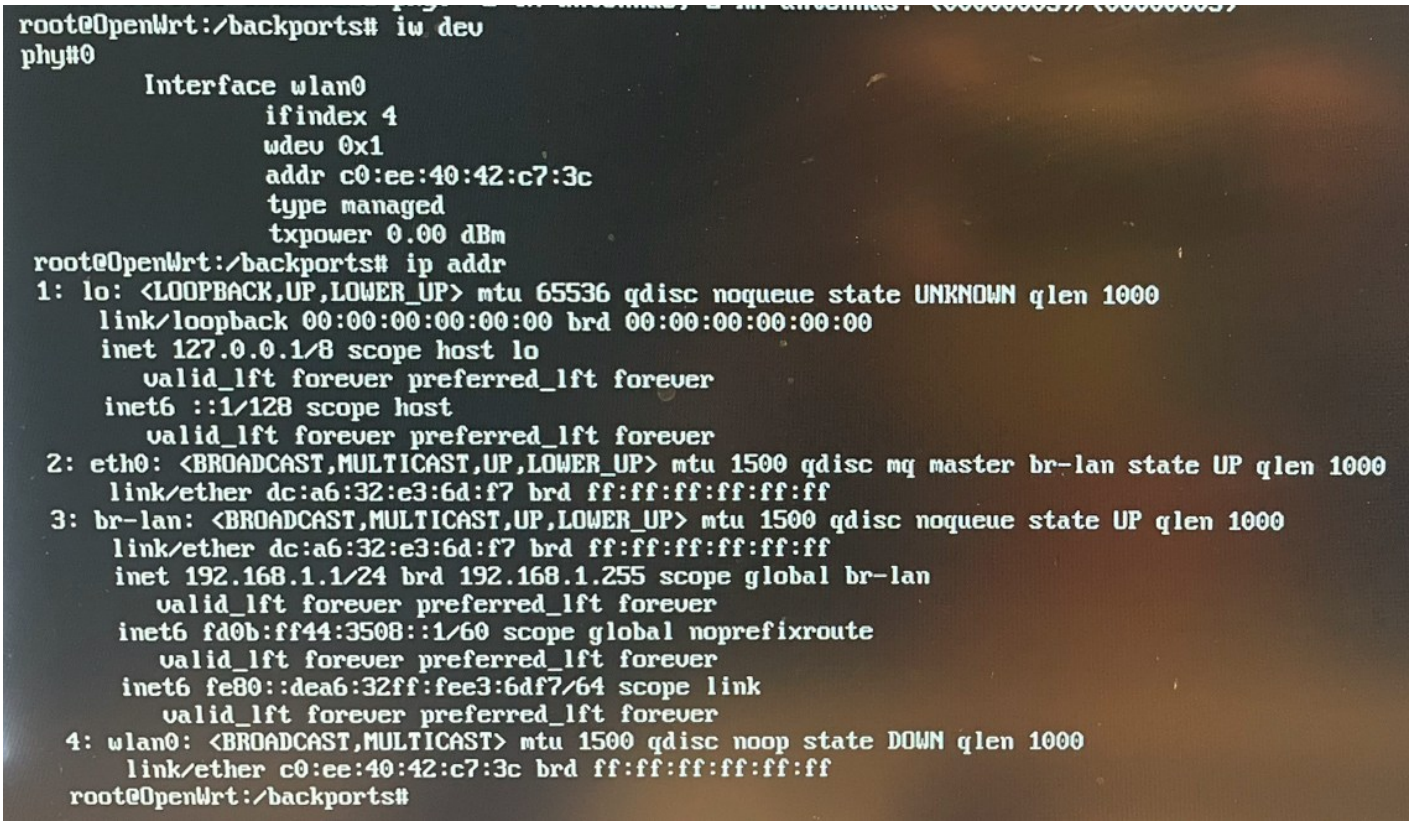
7.2 Inserting Ezurio's Backport Drivers

```
cd /backports
find . -name *.ko
insmod ./compat/compat.ko
insmod ./net/wireless/cfg80211.ko
insmod ./net/mac80211/mac80211.ko
insmod ./drivers/net/wireless/laird/lrdmwl/lrdmwl.ko
insmod ./drivers/net/wireless/laird/lrdmwl/lrdmwl_pcie.ko
```



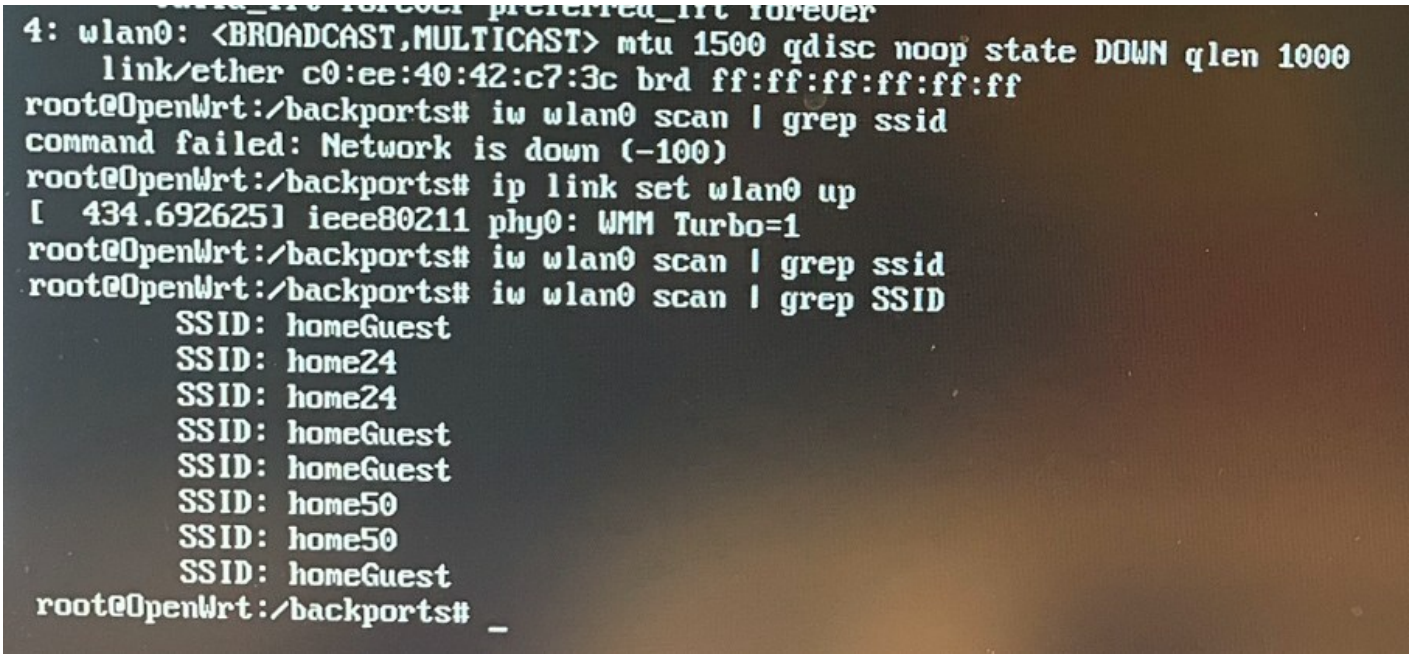
7.3 Bringing up the Wi-Fi Interface and Proof Of Work

```
iw dev
ip addr
```



Note: This now shows the presence of the “wlan0” Wi-Fi interface.

```
ip link set wlan0 up
iw wlan0 scan | grep SSID
```



8 When it May be Necessary to Change Kernel Config

8.1 Newer Kernel Versions

The main kernel config, as of 5.19, now requires that cfg80211 be included so that the networking core layer can support an external cfg80211 module.

The kernel version in this guide is 5.15.134:

```
root@OpenWrt:/backports# uname -a
Linux OpenWrt 5.15.134 #0 SMP Mon Oct 9 21:45:35 2023 aarch64 GNU/Linux
root@OpenWrt:/backports#
```

Hence it was not necessary to include “Wireless” as “M”.

However, kernel configuration can be changed with issuing this command:

```
make -j$(nproc) kernel_menuconfig
```

```
                                    Wireless
ts submenus ---> (or empty submenus ----).  Highlighted letters
xit, <?> for Help, </> for Search.  Legend: [*] built-in  [ ] ex

--- Wireless
[ ]   Wireless extensions
[ ]   WEXT_SPY
[ ]   WEXT_PRIV
<M>   cfg80211 - wireless configuration API
[ ]     nl80211 testmode command (NEW)
[ ]     enable developer warnings (NEW)
[ ]     cfg80211 certification onus
[*]     enable powersave by default (NEW)
[ ]     cfg80211 DebugFS entries (NEW)
[*]     support CRDA (NEW)
[ ]     cfg80211 wireless extensions compatibility (NEW)
< >   LIB80211
< >   LIB80211_CRYPT_WEP
< >   LIB80211_CRYPT_CCMP
< >   LIB80211_CRYPT_TKIP
< >   Generic IEEE 802.11 Networking Stack (mac80211)
```

8.2 If WEXT should become included OpenWRT default kerenl config as previously

It is a requirement to have WEXT disabled as it is not supported by Ezurio's backports.

```
--- Wireless
[ ]   Wireless extensions
[ ]   WEXT_SPY
[ ]   WEXT_PRIV
```

9 Revision History

Version	Date	Notes	Contributor(s)	Approver
1.0	21 December 2023	Initial Release	Alexander Mohr	Andy Ross
2.0	15 Apr 2025	Ezurio rebranding	Sue White	Dave Drogowski

Ezurio's products are subject to standard Terms & Conditions.