

## Configuring your Beaglebone Black/Green for use with the ADC-001

Last edit: 1.9.2018 – SDB.

The code running the ADC-001 assumes you are using Debian 9.2 2017-10-10. It may fail on other Debian releases due to version skew. Use the following steps to reflash your Beaglebone for Debian 9.2 2017-10-10 prior to installing the ADC-001 software.

**Set-up.** The following instructions assume you have a PC (the “host”) connected to the internet. They also assume you can communicate from the host to your BB via the serial debug port, as well as a TCP/IP connection. Before reflashing your BB, make sure to remove all capes, including the ADC-001 from the BB.

### Download and burn the Debian image to the BB

1. Use your host computer to download the desired image from <https://beagleboard.org/latest-images>. The image to get is “bone-debian-9.2-iot-armhf-2017-10-10-4gb.img.xz”.
2. On the host computer, use the command “unxz” it to uncompress the image. This should leave an .img file in your working directory.
3. Now copy the image file to a microSD card. Note that the image contains both the boot and Debian partitions, so don't do any formatting of the microSD. Under Linux, become root and copy the image onto the microSD card using the command:  

```
dd bs=4M if=bone-debian-9.2-iot-armhf-2017-10-10-4gb.img of=/dev/sdb
```

This operation may take a couple minutes to complete. Note that you may find the SD card mounted on a different device. Look at the system log to see where it is mounted.
4. Hook up serial debug cable to debug header on BB. Start “minicom”, “screen”, or your preferred terminal program to watch the boot sequence and interact with BB.
5. Install the microSD into the BB, then hold down S2 (“reset”) while inserting power to the BB. Hold the button until LEDs flash, then release it. Let the BB boot up. Watch the BB debug spew in your terminal program. This can take many minutes. Once the login prompt appears log in as root, password = “root”.
6. Configure the BB to reflash eMMC upon boot. To do so, open the file /boot/uEnv.txt in a text editor and navigate to the bottom line, which reads  

```
#cmdline=init=/opt/scripts/tools/eMMC/init-eMMC-flasher-v3.sh.
```

 Delete the “#” at the beginning of this line, then save the file.
7. Reboot by power-cycling. Hold down S2 (“reset”) , insert the +5V power, let the board boot, then wait for it to flash eMMC. Reflashing will take many minutes.
8. Once flashed, the Beaglebone will power itself down.
9. Remove microSD card, then apply power again and watch reboot. Then log in as root, password = root. Verify your BB now runs the correct Debian release using the command “uname -a”. You should get the response “Linux beaglebone 4.4.91-ti-r133 #1 SMP Tue Oct 10 05:18:08 UTC 2017 armv7l GNU/Linux”.

### Configure networking on the BB

In order to talk to the BB while the ADC-001 cape is in place, you need to configure it to use IP over Ethernet. I prefer to use a static IP address for each of my Beaglebones. Here is how to set the static IP address.

10. Set the IP address by editing /etc/network/interfaces. I use 192.168.0.6 in the below example. You should set your preferred static IP address, which might be different from mine.

```
auto eth0
iface eth0 inet static
    address 192.168.0.6
    netmask 255.255.0.0
```

11. Plug in your Ethernet cable to the BB and make sure it's live (i.e. connected on the other end).
12. Also set up connman to use the same address. If you don't do this, then connman will fight to change the IP address, and unpredictable things can happen. Here are the commands:

```
connmanctl services
```

This will return the name of the Ethernet service, something like ethernet\_b0d5ccfe27c4\_cable. Now issue the command:

```
connmanctl config ethernet_b0d5ccfe27c4_cable ipv4 manual \
192.168.0.6 255.255.0.0 192.168.0.255
```

13. Now configure the BB to allow root login via ssh. Edit the file /etc/ssh/sshd\_config
- Change the line beginning with #PermitRootLogin to

```
PermitRootLogin yes
```

### **Enable pruss uio for communication to the PRU**

The pruss\_uio subsystem uses Linux's so-called "userspace IO" feature to allow the my programs to talk to the PRU.

14. Enable the uio subsystem to communicate with the PRU. Edit the file /boot/uEnv.txt. Comment out this line:

```
uboot_overlay_pru=/lib/firmware/AM335X-PRU-RPROC-4-4-TI-00A0.dtbo
```

Then comment in this line (i.e. remove the "#"):

```
uboot_overlay_pru=/lib/firmware/AM335X-PRU-UIO-00A0.dtbo
```

15. Now enable the device tree entry for the uio subsystem by editing the .dts file:

```
cd /opt/source/dtb-4.4-ti/src/arm/
vi am335x-bonegreen.dts
```

Comment out this line, so that it reads:

```
/* #include "am33xx-pruss-rproc.dtsi" */
```

Comment in this line:

```
#include "am33xx-pruss-uio.dtsi"
```

16. Rebuild the kernel:

```
cd /opt/source/dtb-4.4-ti
make && make install
```

17. Make sure rproc stuff does not get loaded by creating a blacklist.conf file:

```
vi /etc/modprobe.d/pruss-blacklist.conf
```

Add these lines to the file you have just created, then save it:

```
blacklist pruss
blacklist pruss_intc
blacklist pru-rproc
```

18. Reboot.

19. Check for success:

```
lsmod | grep uio
```

You should see:

```
uio_pruss          4629  0
uio_pdrv_genirq    4243  0
uio                11100  2 uio_pruss,uio_pdrv_genirq
```

The specific thing to look for is the presence of `uio_pruss` in the list of loaded kernel modules.

Next issue the this command:

```
ls /sys/class/uio/
```

You should see:

```
uio0 uio1 uio2 uio3 uio4 uio5 uio6 uio7
```

This means the filesystem holds an interface to the uio stuff. This interface is accessed in the file `__prussdrv.h` included in my include directory.

### Install and configure the ADC-001

20. Power the BB down and install the ADC-001 cape onto your BB.

21. Power up the BB and copy over the ADC-001 over project directory you downloaded from GitHub to your BB. Make sure to copy over all subdirectories as well as the main directory.

22. Log into the BB as root, enter the ADC-001 project directory, and issue the command “make”. The build should run to a successful completion. The build process will also install the device tree overlay file into the correct place (`/lib/firmware`).

23. Make sure the `.dtbo` file ends up in `/lib/firmware`. Issue the command “`ls /lib/firmware`” and verify the file `ADC_001-00A0.dtbo` is present in that directory. If it's not there, just copy it over by hand.

24. Tell the operating system to use the ADC-001 cape. Edit the file `/boot/uEnv.txt` and add this line (or modify the existing commented-out line):

```
uboot_overlay_addr0=/lib/firmware/ADC_001-00A0.dtbo
```

25. On the Beaglebone Black, remove support for HDMI on output conns. This is required so the ADC-001 cape may be recognized. (This is not required for the BB Green, which doesn't support video.) Edit the file `/boot/uEnv.txt` and comment out the following line (i.e. put a “#” at the beginning of the line):

```
#uboot_overlay_pru=/lib/firmware/AM335X-PRU-RPROC-4-4-TI-00A0.dtbo
```

26. Then reboot. The operating system should now pick up the presence of the ADC-001 cape. To check it is detected, issue this command:

```
cat /sys/kernel/debug/pinctrl/44e10800.pinmux/pinmux-pins | grep pru
```

You should see the following output, which indicates 5 pins are used to communicate with the board:

```
pin 56 (44e108e0.0): 4a300000.pruss (GPIO UNCLAIMED) ...
pin 101 (44e10994.0): 4a300000.pruss (GPIO UNCLAIMED) ...
pin 102 (44e10998.0): 4a300000.pruss (GPIO UNCLAIMED) ...
pin 103 (44e1099c.0): 4a300000.pruss (GPIO UNCLAIMED) ...
pin 105 (44e109a4.0): 4a300000.pruss (GPIO UNCLAIMED) ...
```

27. Finally, if everything has worked so far, run the program: “./main”. It should run through a set of tests, make some measurements, then finally run a loop, filling a buffer full of measured points once every 1/2 second. Hook up the microphones and speak into them. You should see the values of the measured points vary corresponding to the intensity of the sound waves detected by the microphones.

## References

- [https://elinux.org/Beagleboard:BeagleBoneBlack\\_Debian](https://elinux.org/Beagleboard:BeagleBoneBlack_Debian)
- <https://github.com/Scrashdown/PRU-Audio-Processing>
- <https://groups.google.com/forum/#!msg/beagleboard/l59Dx8ygxNg/GvIzOJSzDAAJ>