Embedded Apprentice
Linux Engineer
Our Conference Sponsors

- THE LINUX FOUNDATION
- Intel
- yocto PROJECT
- SCALE 16x
Our Training Sponsors
Before we do anything....

# This is going to take a little bit and we need to start now.
chmod +x e-ale-setup.sh
./e-ale-setup.sh
Quick Introduction

- Beth ‘pidge’ Flanagan
- CTO of Togán Labs (www.toganlabs.com)
- Former release engineer for the Yocto Project
- SW dev for 25 years, embedded for 10
- If anything goes wrong with this talk, it’s Behan’s fault.
Now we can open things up

- 2x microUSB cables
- 1 uSD reader/writer
- 1 pocketbeagle
- 1 BaconBits cape
- 1 uSD card
- 1 uSD to SD adapter
The hardware

- BaconBits on the top
- Pocketbeagle on the bottom
Putting the hardware together

- USB type-A (the big one) on the cape is on the same end as the micro-USB (the wee one) on the pocketbeagle
- If you get this wrong they’re difficult to get apart.
What it should look like now.

- CAREFULLY align the pins, push them together.
What mine looks like

- I dislike Altoids tins
- I also approve of waterproofing things
- And dedicated USB batteries
- Provides opportunities to explain embedded linux to airport security.
Comms

- BaconBits provides serial console (we’ll be using this)
- Pocketbeagle provides Ethernet over USB
- Both ports can provide power.
Let’s blast some things to the uSD!
Why I use dd but I’m going to ask you to use etcher.io

- dd is a wonderful command line tool to create sd cards. BUT.
- Long ago, I use to sync .bash_history between my dev machines
- laptop’s usb was listed as /dev/sdb.
- Desktops / was /dev/sdb.
- Ctrl+shift+r for dd and BOOM!
- Hybrid SuSE/poky generic-x86-64 OS until I could reinstall.

Conclusion. Use Etcher.io today EVEN IF YOU KNOW dd.
Let’s blast some things to the uSD!

- Insert sd card reader + sd card
- From terminal: ~/e-ale-intro/etcher-electron-1.4.5-x86_64.AppImage
Let’s blast some things to the uSD!
Let’s blast some things to the uSD!

- Insert sd card reader + sd card
- From terminal: ~/e-ale-intro/etcher-electron-1.4.5-x86_64.AppImage
And now we wait.
When etcher is done

- Remove uSD card from reader/writer
- Insert into the pocketbeagle uSD slot gold pads down.
- You should feel a click.
- Take one USB cable and insert into your laptop and the other end into the microusb of the BaconBits (the top board).
- This will give us a serial console
I’m going to teach minicom and moving files manually:
  • There are a bunch of ways to do this (kermit, screen, ethernet over USB, uboot, nfs, tftp, etc. etc.)
  • But this is the dirt simple thing that doesn’t require setup and just about every board in the world with an SD card also supports a serial console.
  • But it also annoys Behan, which is a bonus
  • Run “sudo minicom” in a terminal
BOOT

- Take the second USB cable and plug it into the pocketbeagle and your laptop.
- You should see a login prompt!
- Username: root
- Password: <empty>
It’s ALIVE!
Now, let’s do something with it
Setting up the toolchain

cd ~/e-ale-intro
tar xvf gcc-linaro-7.3.1-2018.05-x86_64_arm-linux-gnueabihf.tar.xz
export PATH=~/e-ale-intro/gcc-linaro-7.3.1-2018.05-x86_64_arm-linux-gnueabihf/bin:$PATH
arm-linux-gnueabihf-gcc -v
Let’s make a kernel module

cd ~/e-ale-intro/basic-kernel-module;
make KERNEL=~/e-ale-intro/linux-kernel/ CROSS=~/e-ale-intro/gcc-linaro-7.3.1-2018.05-
x86_64_arm-linux-gnueabihf/bin/arm-linux-gnueabihf-
ls
Looking deeper

PWD := $(shell pwd)
obj-m += hello.o

all:
    make ARCH=arm CROSS_COMPILE=$(CROSS) -C $(KERNEL) M=$(PWD) modules

clean:
    make -C $(KERNEL) M=$(PWD) clean
Looking deeper

PWD := $(shell pwd)
obj-m += hello.o

all:
    make ARCH=arm CROSS_COMPILE=<location of toolchain and prefix> -C <location of kernel source (change directory, pull in that Makefile)> M=<location of external module> modules

clean:
    make -C $(KERNEL) M=$(PWD) clean
Getting it onto the device the lazy way

Unplug just the pocketbeagle (Leave the baconbits cape plugged in)
Eject uSD card and put it into your laptop

cp ~/e-ale-intro/basic-kernel-module/hello.ko <whereever your laptop mounted the sd card>/tmp

#very important!
#
# See: Eat My Data: How Everybody gets File IO Wrong
# by Stewart Smith
sync
Getting it onto the device the lazy way

Put SD card back in. Plugin the pocketbeagle. Wait for everything to boot and then log in

cd /tmp
  # hello.ko should be here in tmp
ls
  # hello shouldn't be here. If so, you are MAGIC or I did something wrong
lsmod |grep hello
  # Let's install it
insmod /tmp/hello.ko
  # hello should be here. If not, either you or I did something wrong
lsmod |grep hello
  # Let's remove it
rmmod hello; lsmod |grep hello
More Resources

https://github.com/e-ale/meta-pocketbeagle
https://github.com/e-ale/basic-kernel-module
https://github.com/e-ale/linux-kernel

Special thanks for Jason Kridner’s images (and hands)

(One of the best talks I’ve ever seen)