BeagleBone Cookbook Webinar Series
Recipe #1 – Playing and Recording Audio

October 27, 2015

Jason Kridner

Co-author of BeagleBone Cookbook

Board member at BeagleBoard.org Foundation

Sitara Applications Engineering at Texas Instruments

BeagleBone Black

Ready to explore and use in minutes

Truly flexible open hardware and software development platform

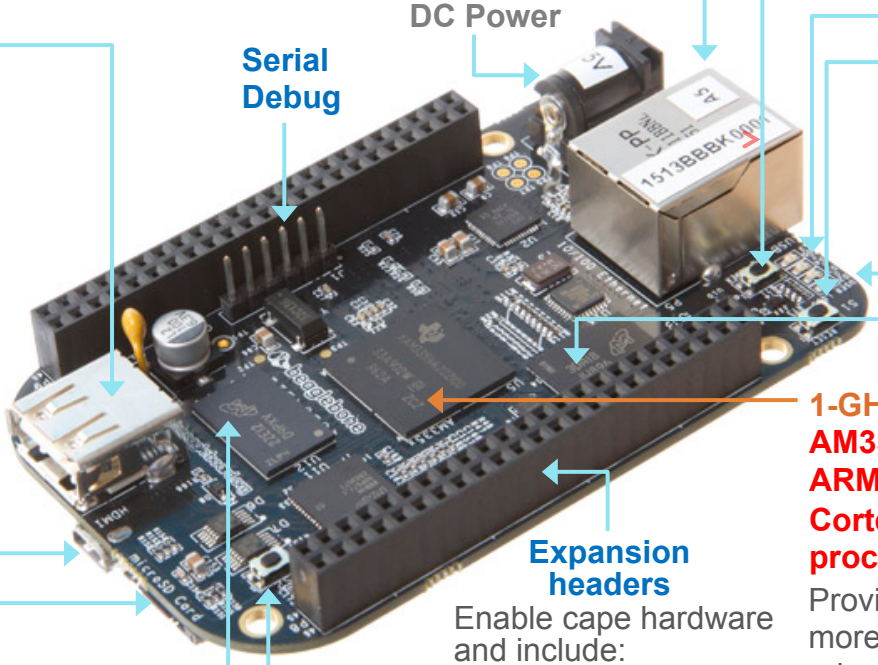
All you need is in the box

Proven ecosystem from prototype to product

- Ready to use
  - USB client network
  - Built-in tutorials
  - Browser based IDE
  - Flashed w/Debian
- Fast and flexible
  - 1-GHz Sitara ARM
  - 2x200-MHz PRUs
  - 512-MB DDR3
  - On-board HDMI
  - 65 digital I/O
  - 7 analog inputs
- Support for numerous Cape plug-in boards
  http://beaglebonecapes.com

BeagleBone Black – the most flexible solution in open-source computing

BeagleBone Black board features



10/100 Ethernet

USB Host
Easily connects to almost any everyday device such as mouse or keyboard

microHDMI
Connect directly to monitors and TVs

microSD
Expansion slot for additional storage

512MB DDR3
Faster, lower power RAM for enhanced user-friendly experience

Serial Debug

DC Power

Boot Button

Power Button

LEDS

Reset Button

USB Client
Development interface and directly powers board from PC

4-GB on-board storage using eMMC
- Pre-loaded with Debian Linux Distribution
- 8-bit bus accelerates performance
- Frees the microSD slot to be used for additional storage for a less expensive solution than SD cards

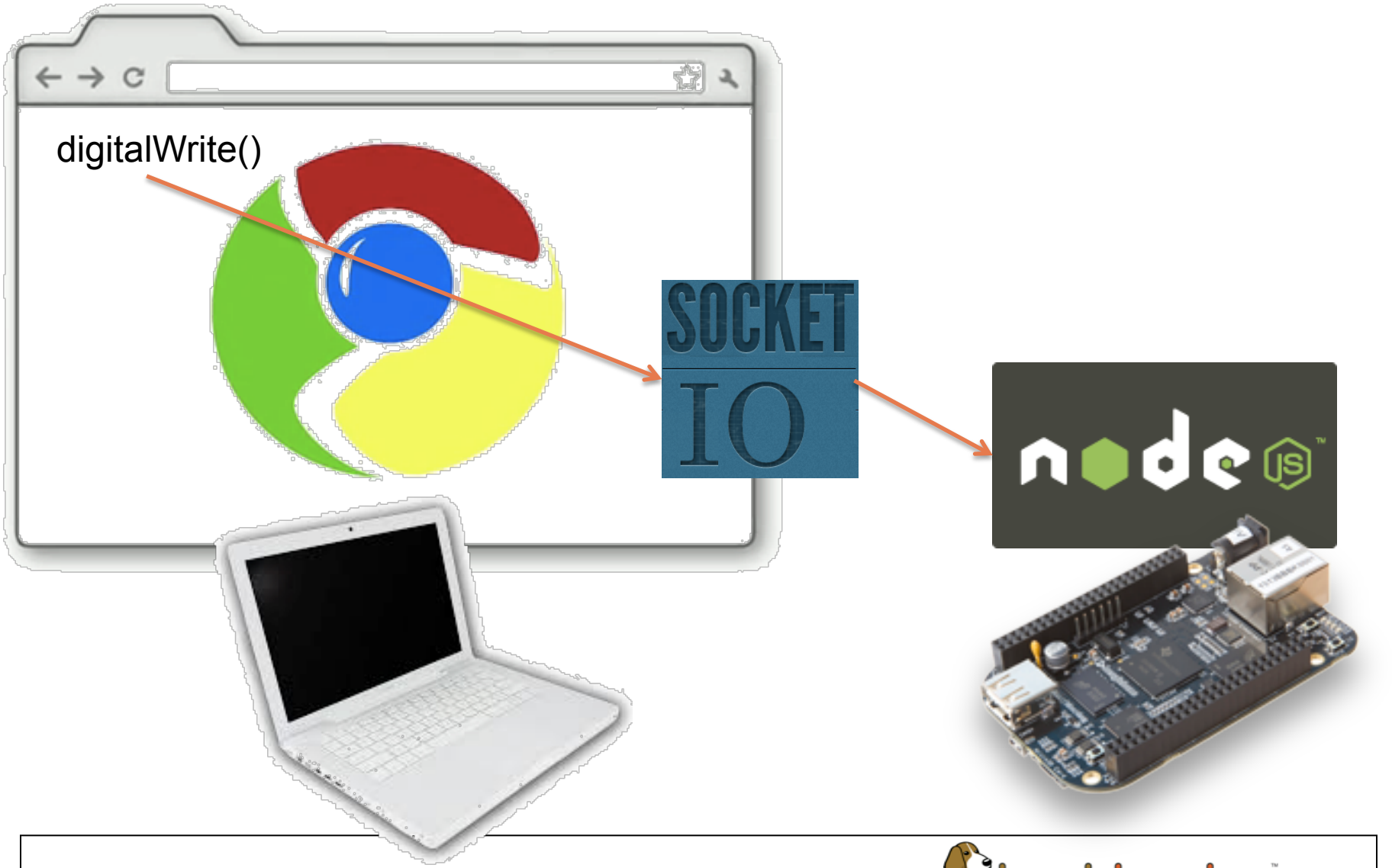
1-GHz Sitara AM335x ARM® Cortex™-A8 processor
Provides a more advanced user interface and up to 150% better performance than ARM11

Expansion headers
Enable cape hardware and include:
- 65 digital I/O
- 7 analog
- 4 serial
- 2 SPI
- 2 I2C
- 8 PWMs
- 4 timers
- And much much more!

Money saving extras:
- Power over USB
- Included USB cable
- 4-GB on-board storage
- Built-in PRU microcontrollers

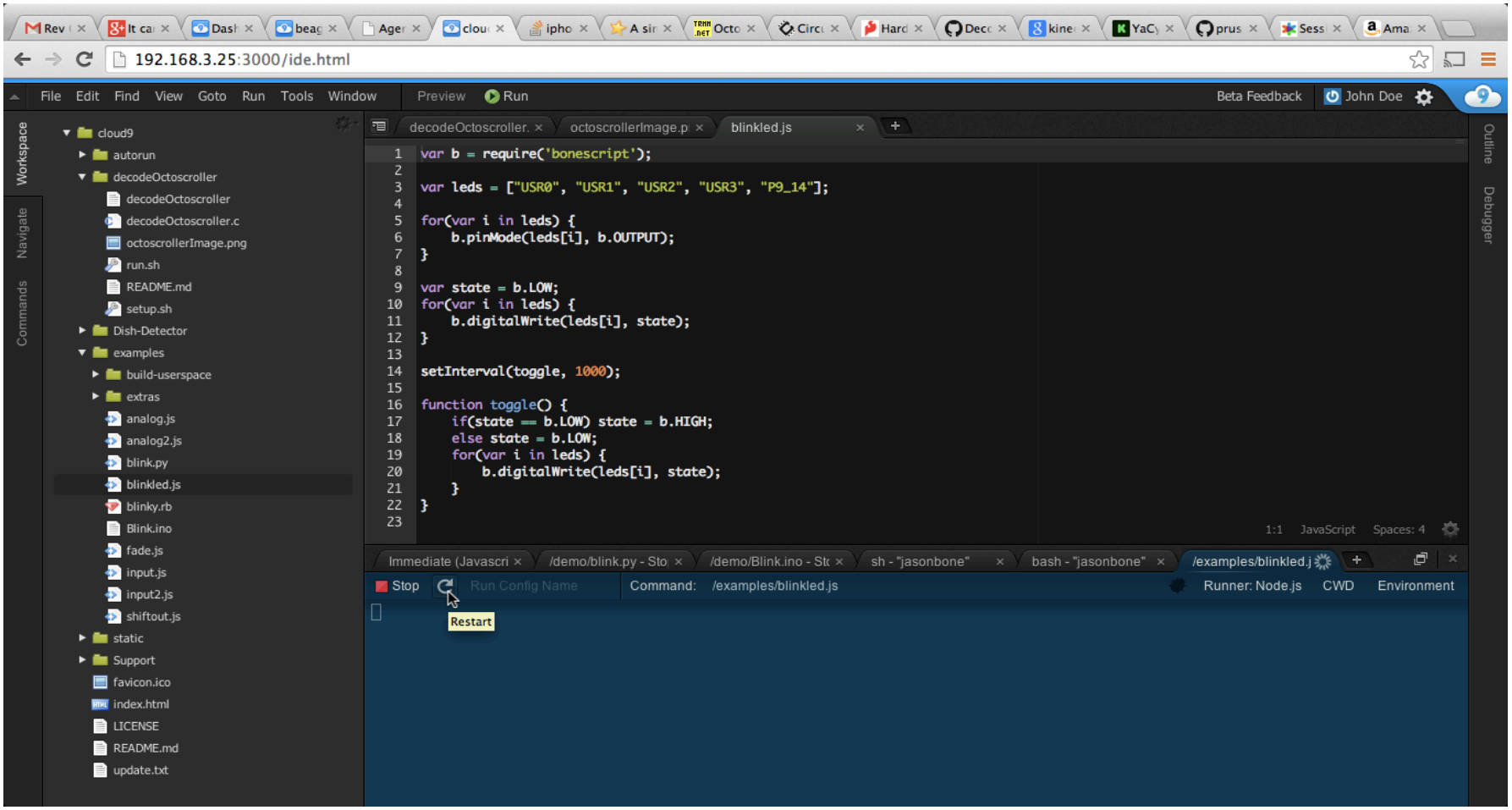
Simple browser-based interactions

http://beagleboard.github.io/bone101

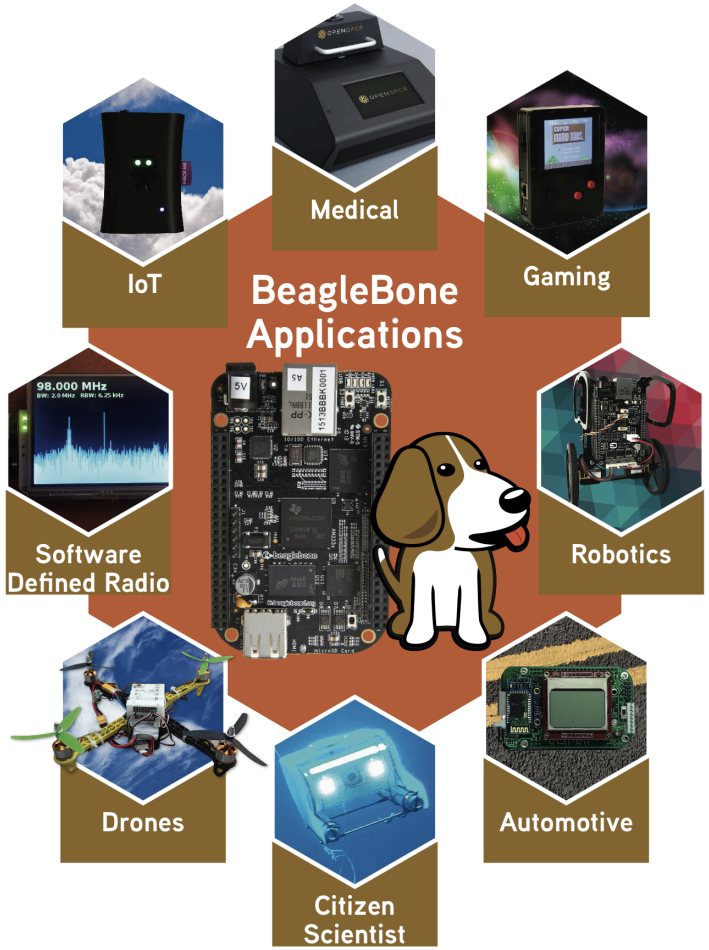


Cloud9 IDE hosted locally

Zero install and exposes command-line



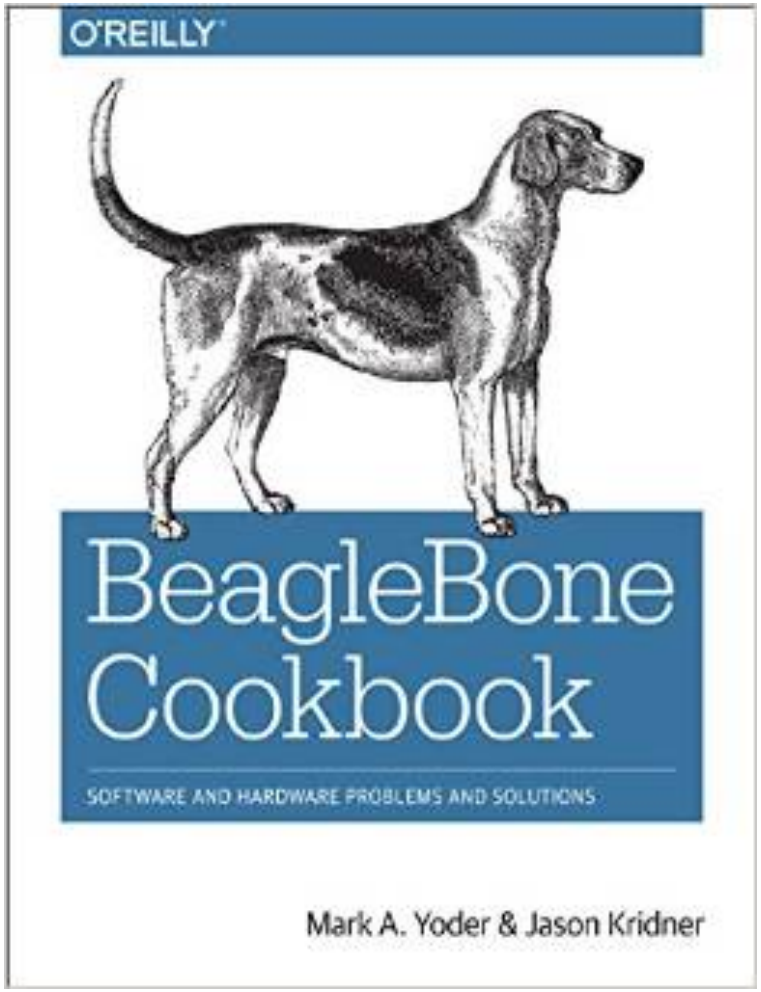
10,000s of developers building connected devices today



- Medical analysis, assistance and information management
- Home information, automation and security systems
- Home and mobile entertainment and educational systems
- New types of communications systems
- Personal robotic devices for cleaning, upkeep and manufacturing
- Remote presence and monitoring
- Automotive information management and control systems
- Personal environmental exploration and monitoring

BeagleBone Cookbook

http://beagleboard.org/cookbook



- 99 recipes covering
  - Basics
  - Sensors
  - Displays and outputs
  - Motors
  - Internet of things
  - Kernel
  - Real-time I/O
  - Capes

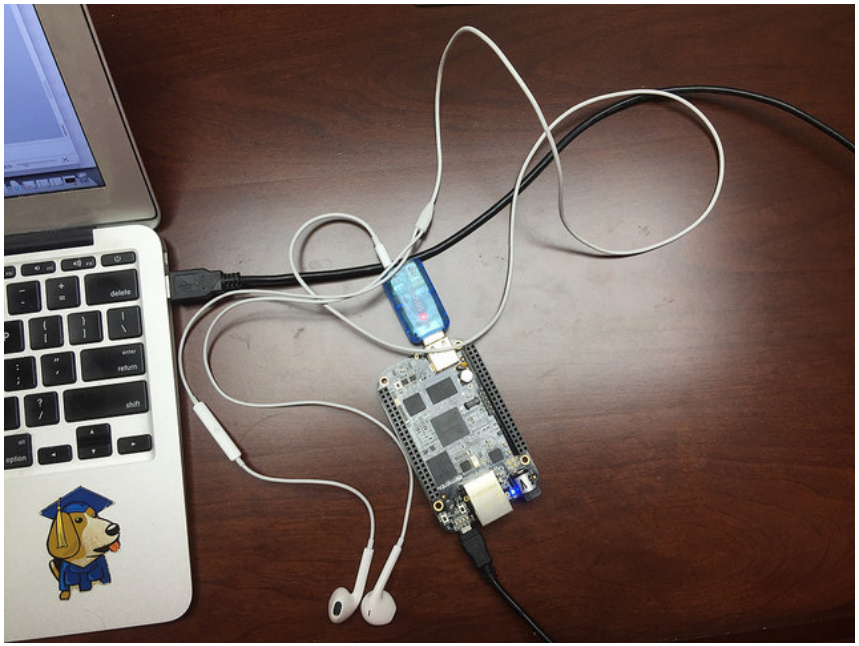
Possible audio solutions

- Built-in HDMI audio
  - connect to TV or HDMI-audio adapter
- Audio cape
  - SPI, I²S and I²C available
- USB Bluetooth dongles
  - BlueZ → https://wiki.debian.org/Bluetooth/Alsa
- USB audio adapter ← this will be our approach
  - Easy to find adapters on Amazon, etc.
    - http://www.amazon.com/s/ref=nb_sb_noss_2?url=search-alias%3Daps&field-keywords=linux+usb+audio

Step #0 – Prerequisites

- Connect to the board per recipe 1.2
  - http://beagleboard.org/getting-started
- Verify the software image per recipe 1.3 and potentially updating per recipe 1.9
  - http://beagleboard.org/latest-images

Step #1 – Boot with USB audio adapter



- Power up with USB audio adapter inserted
  - Some kernels don't like USB hotplugging
  - USB power typically sufficient, but add a power adapter if you see issues
- Verify driver loaded
  - lsusb
  - dmesg

Step #2 – Test playback

- Discover devices
  - man aplay
  - aplay -l
  - aplay -L
- Playback samples
  - aplay -D "default:CARD=Device" /usr/share/sounds/alsa/Front_Center.wav

Step #3 – Test record

- Use the mixer to set the input gain
  - alsamixer
- Record a sample
  - man arecord
  - arecord -f dat -D "default:CARD=Device" test.wav

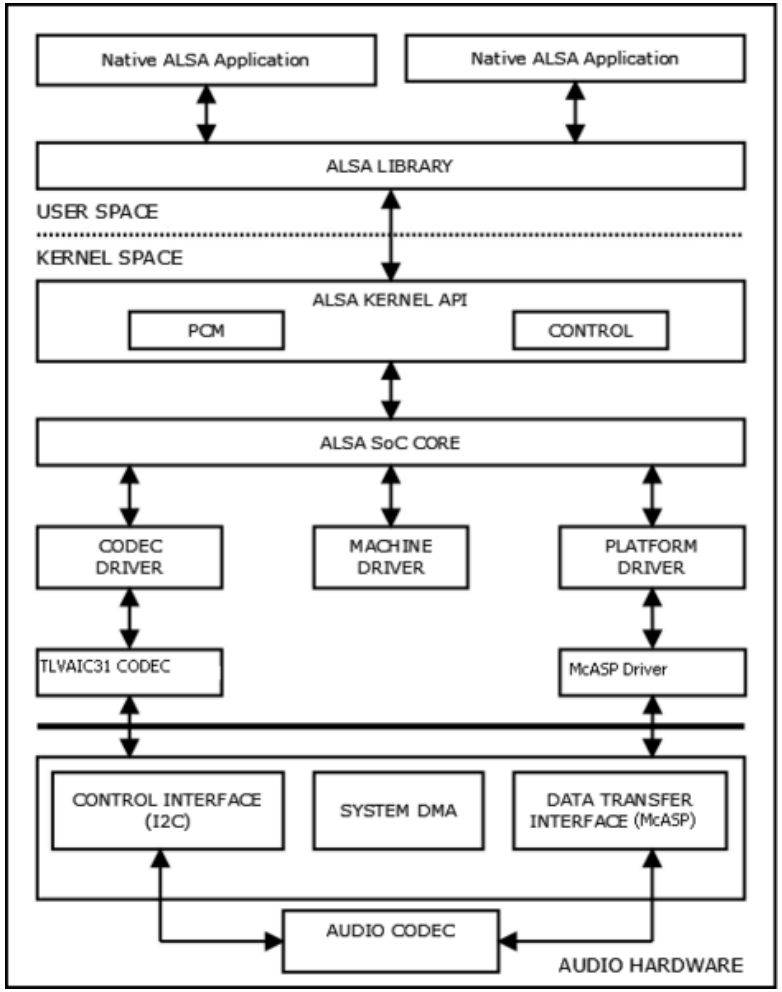
Step #4 – Set default audio

- Write to ~/.asoundrc
- Enables you to use applications without specifying the card each time
- Example
  requires ‘apt-get install flite’
  - flite –t "Hello!"

```
pcm.!default {
    type plug
    slave {
        pcm "hw:1,0"
    }
}
ctl.!default {
    type hw
    card 1
}
```

More about ALSA

Advanced Linux Sound Architecture - http://alsa-project.org



- Includes user space library for application programming
- Supports many devices
- ALSA SoC supports adding codecs to embedded boards

More

- Nice set of tutorials from 13-year old Alek Mabry
  - http://einsteiniumstudios.com/speak.html
- Shortcuts to updates and examples from the book
  - http://beagleboard.org/cookbook