Mezzanine Enablement

Hardware for Software engineers
“People who are really serious about software should make their own hardware”

Alan Kay, Creative Think Seminar, 1982
Why 96Boards?
96Boards is a **Software** Project

- Hardware to support our software goals
- Reference Platform Builds
  - 16.03 release single kernel binary & distro
    - supports many platforms out of the box
  - Software base for applications
    - Including support for IO expansion
- Same form factor, same software, same use cases, different hardware
96Boards are Real Products

- Market Availability
- Directly Usable in Products
- Platforms are First Class Citizens
  - Support from Reference Platform Builds
  - Support from Distros expected
Why Build a Mezzanine?

1. Idea for hardware
2. ???
3. Profit!
DISCLAIMER

I am *not* saying everyone should design hardware!

Adverse effects such as bricked boards, soldering iron burns, or slipped schedules are on your own head!
Why Build a Mezzanine?

- Hardware design has never been easier
- Consider solving SW problems with HW
  - Test jigs
  - Expand capabilities of hardware
Start With The Basics

Getting started can be intimidating, but fear not! Ask the Internet!

- Find Examples showing Basic Electronics
  - Arduino
  - Starter Kits
- Standard Solderless Breadboards*
- Build something that interests you
  - Lightsabers!
Free Software Design Tools

We’ve got Open Source tools for every stage of design

- **KiCad**
  - Schematic & PCB
  - Advanced routing tools

- **Fritzing**
  - Breadboard, Schematic & PCB
  - “Lego” building blocks
  - Library of common maker modules
Free Software Design Tools

- **OpenSCAD**
  - Solid modeling for programmers
  - 3D case design
- **Inkscape**
  - Laser cut designs
- **LibreOffice**
  - Packaging and printed documentation
- **Git, Make, XML tools**
  - Good software practice applies to hardware
Prototyping

- PCB printing
  - DirtyPCBs
  - OSHPark
  - Seeed Fusion PCB

- Assembly
  - Do it yourself
    - Learn how to solder - lots of guides online
  - Prototype assembly services
Manufacturing

- Find a manufacturing partner
  - Doesn’t have to be local
  - Should provide
    - Design for Manufacturing Review
    - Testing
Sales & Marketing

● Don’t ask me, I’m just an engineer
Design for 96Boards

- Start with KiCad Template
  - https://github.com/96boards/96boards-kicad-mezzanine-template

- Or fork an existing design
  - 96Boards UART
  - Sensors Mezzanine
  - Robomezzi
Mechanical

- 54x85mm
- Optionally stackable
  - Check baseboard height requirements
- 7mm minimum separation
  - LS connector only insufficient mechanically
    - Mount with standoffs
  - LS+HS mounts solidly
- Prefer surface mount
  - Don’t short against baseboard connectors
Components and Footprints

- 0603 footprints are a good choice
  - Resistors
  - Capacitors
  - Diodes
- Look in datasheets for recommended footprint
- Look at other open hardware designs
Power

- Supply
  - SYS_DCIN (8-18V) - pins 36, 38
    - Careful!
  - Supply from either baseboard or mezzanine

- Regulated Power
  - 5V/1A - Pin 37
  - 1.8V/0.1A - Pin 35

- Typical mezzanine will regulate from 5V rail
- Use 1.8V (pin 35) as VIO reference
1.8V IO

Transparent level shifting is tricky. Options:

- One-way Buffers
  - When the signal is unidirectional
- Simple Clamp Style
  - BSS138 MOSFETs
  - PCA9306 dual channel (made for I²C)
- High Density Multi-Channel
  - Low drive strength
  - Good for short traces, poor for off-board
  - TXS010x - Clamp style
  - TXB010x - Push/pull
I\textsuperscript{2}C

- Test design on breadboard before building
- BSS138 or PCA9306 works best
- Not all devices play well together
- Check drive strength and choose pull-ups appropriately
- ie. 1k pull-up is typical, but the Grove RGB LCD cannot drive it through the shifter
UART

- UART0
  - Available for applications
  - RTS/CTS lines

- UART1
  - Default Linux console
  - Pass through to a header if not using
SPI

- Pins defined for MISO, MOSI, CLK and one CS line
- Can use GPIOs as additional CS lines
GPIO

- 12 Lines
- Pins 23 through 34
PWR_BTN & RST_BTN

- Power and Reset button inputs
- Pulled high
- Drive low to activate
  - ex. Short to GND with momentary button
- Behaviour depends on PMIC/SoC
  - Artifact of using Mobile SoCs & PMICs
  - Looking to resolve in next rev of spec
PCM/I$^2$S Audio

- Should be usable with I2S codec
Questions?