Herd Your Boards, Become a Farmer

Embedded Linux Conference Europe 2016

Geert Uytterhoeven
geert@linux-m68k.org

Glider bvba

Tuesday, October 11

Glider.be

Table of Contents

About Me (and Linux)

Hobbyist

1994 Linux/m68k on Amiga

1997 Linux/PPC on CHRP

1997 FBDev

Sony

2006 Linux on PS3/Cell

SONY

Glider bvba

2013 Renesas ARM-based SoCs

RENESAS

Why a Board Farm?

Board on Your Desk

Advantages

- ✓ Easy to setup
- ✓ Easy to interact with

Disadvantages

- X One too many boards (*boards are cheap*), outgrowing your desk
- X Too much noise
- X Home Office: Significant other, family members

How many boards on your desk?

Why a Board Farm?

Organizing Development Boards in a Board Farm

Advantages

- ✓ Less clutter on your desk
- ✓ Centralization
- ✓ Automation
- ✓ (Worldwide) Remote access
- ✓ Board sharing

~~Disadvantages~~Challenges

- X (More) Complex setup
- X How to interact with your board?

Board Farm Requirements

Basic

Power & Serial Console

Intermediate

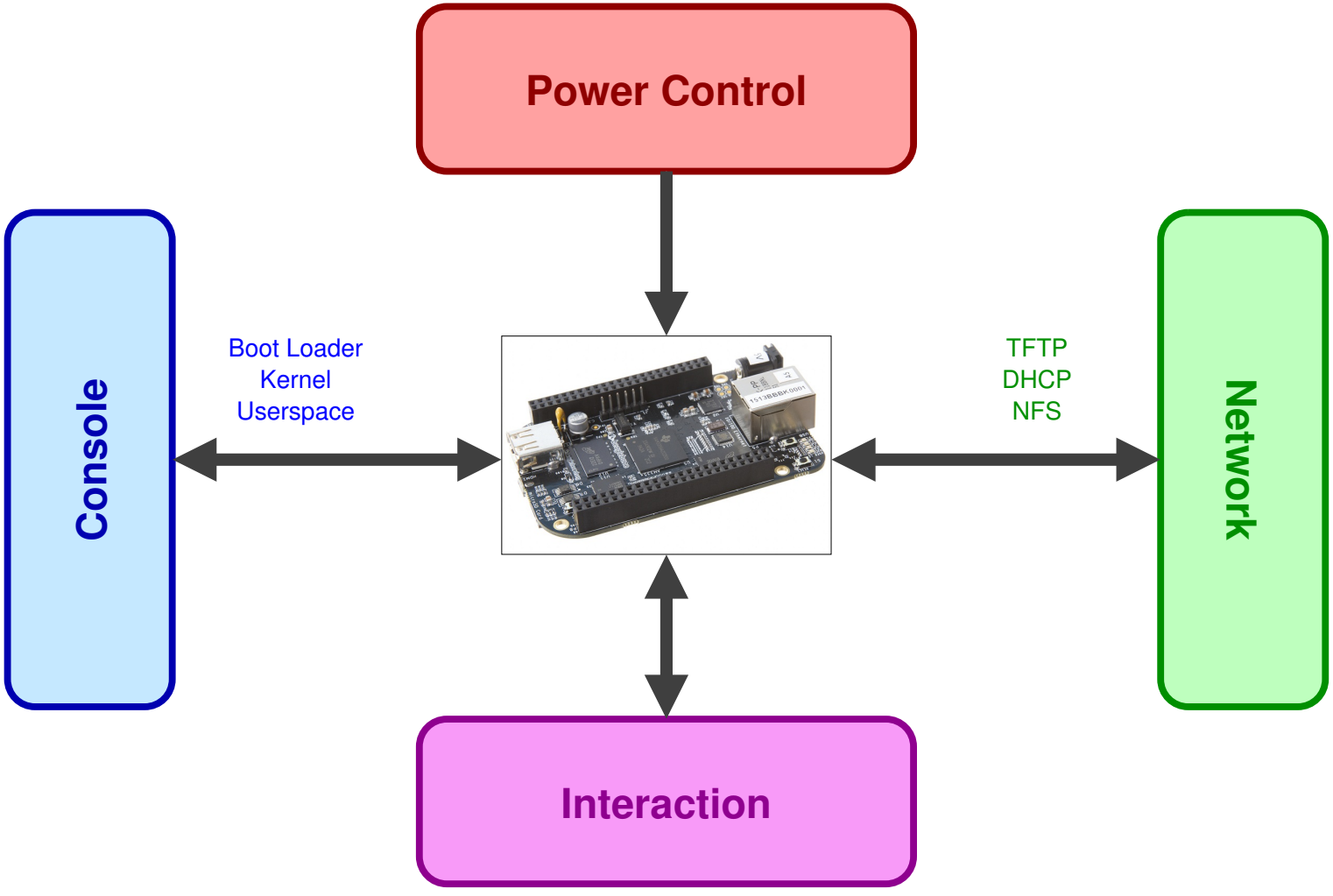
- Reset ($\neq$ powercycle $\neq$ software reboot)
- Wake-Up
- Soft Power-On
- Input buttons
- Measure Power Consumption

Advanced

- Video in & out
- . . . (add yours) . . .

Board Farm Building Blocks

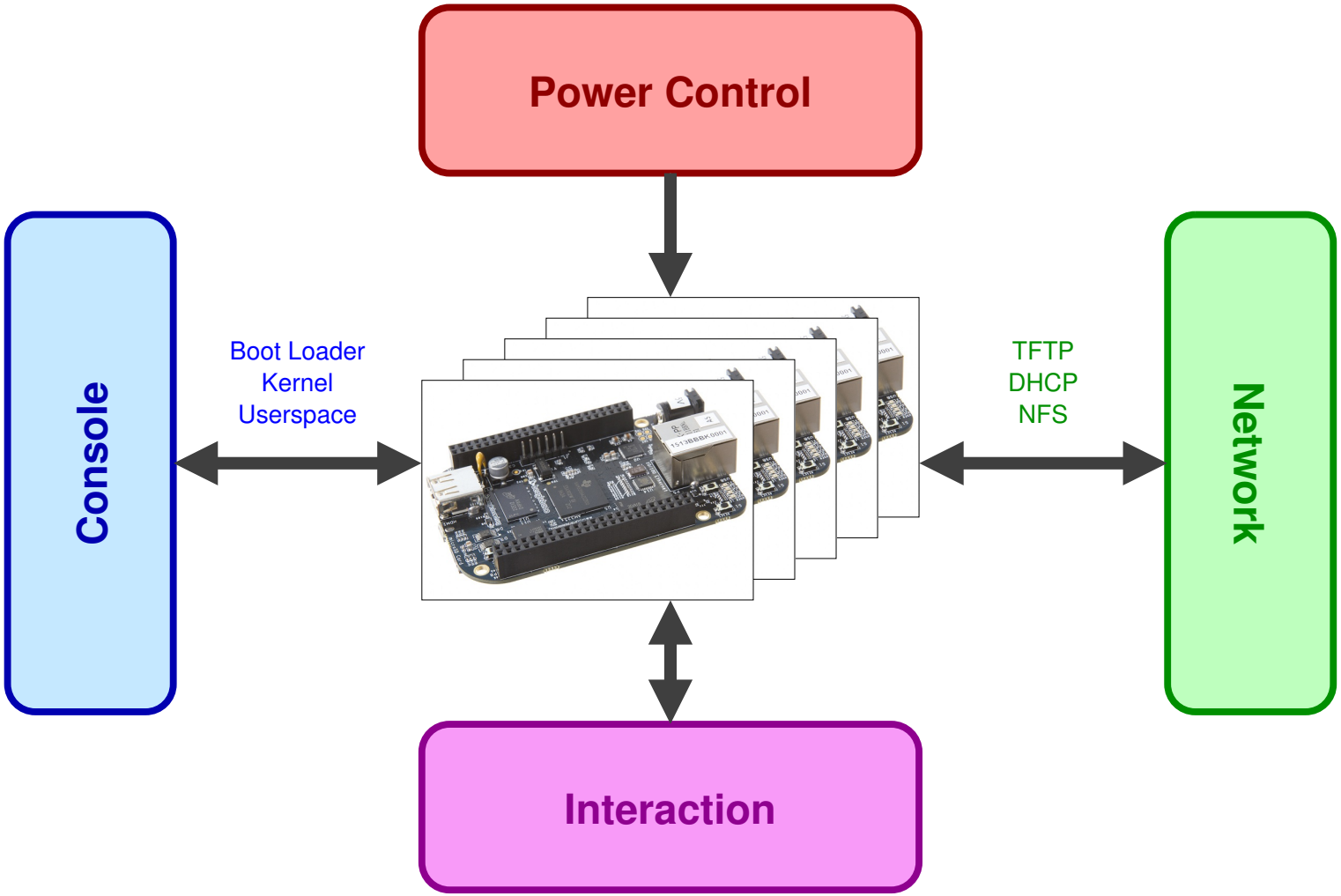
Single Development Board



Board image source: `http://http://elinux.org/`

Board Farm Building Blocks

Board Farm



Board image source: `http://http://elinux.org/`

Power Control

Power Outlet Control

- Classical solution
- Multiple interface options (Ethernet/serial)
- Metering
- Fine for your freezer or washing machine
- Overkill for many low-power embedded boards

Image source: `http://www.apc.com/`

Power Control

Relay Board

- Multiple interface options (GPIO, I2C, USB, Ethernet)
  - Beware multiple boards with identical MAC addresses!
- Overkill for many low-power embedded boards

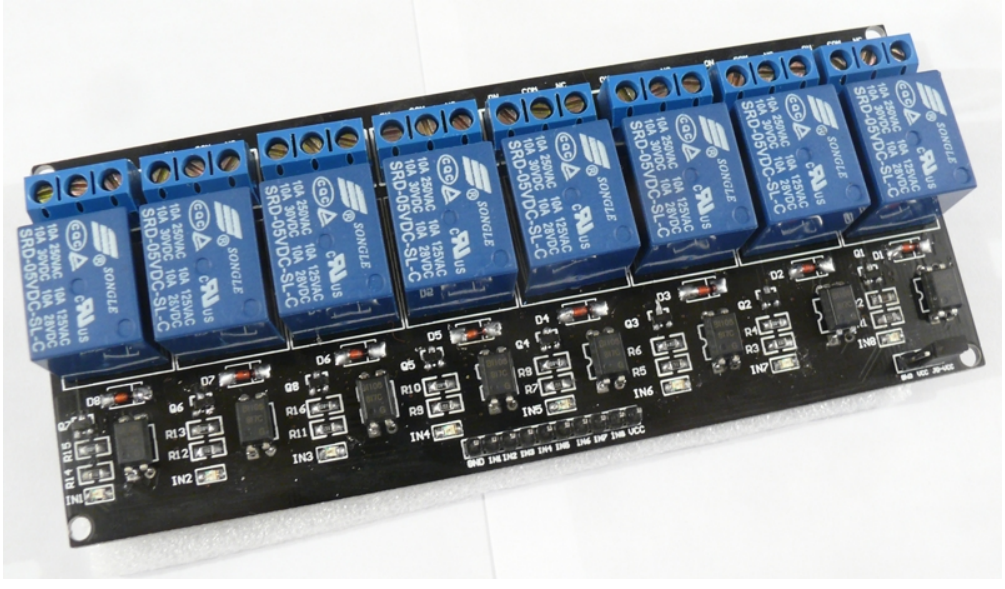


Image source: `http://www.yourduino.com/`

Power Control

BayLibre ACME

- Cape for BeagleBone Black
- OR . . . connect to anything that has an I2C bus
- Up to 8 channels ($\times$ max. 2 capes)

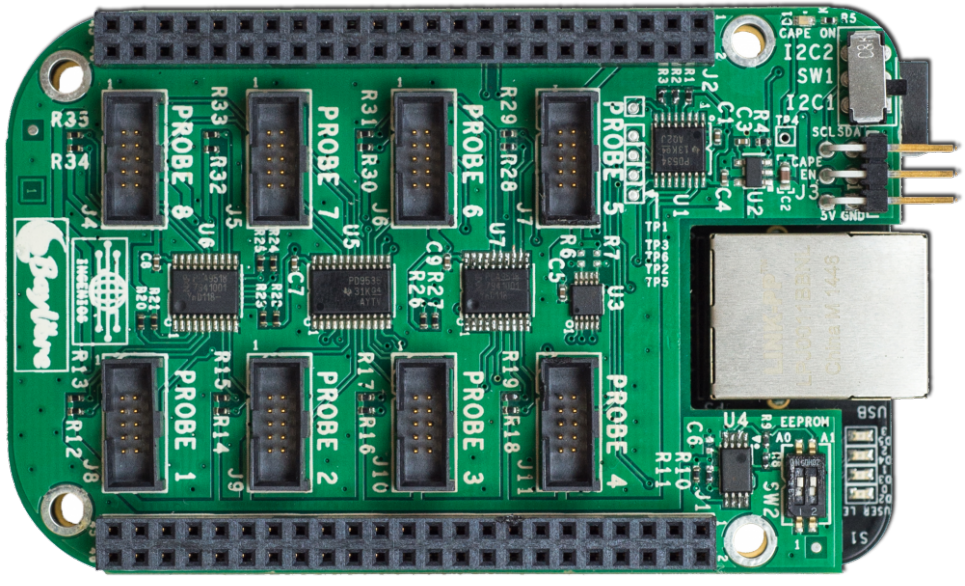
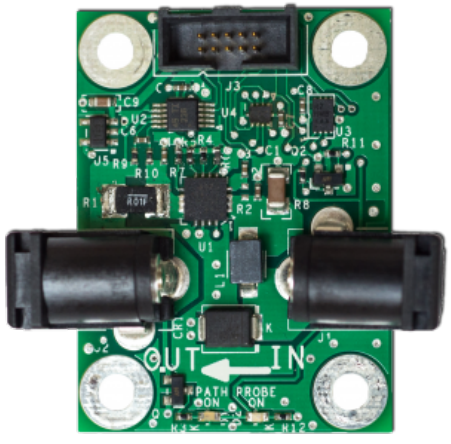


Image source: `http://baylibre.com/`

Power Control

BayLibre ACME

- Power Control and Voltage/Current Monitoring:
  - Jack Power Probe (2.1/5.5mm Center-pos, up to 20V/6A)
  - USB Power Probe (Mini-B, up to 1A)
  - Not good enough for the z890 ‿
- Measurement only:
  - HE10 Power Probe (up to 150mA/1.5A/10A)
  - Temperature Probe



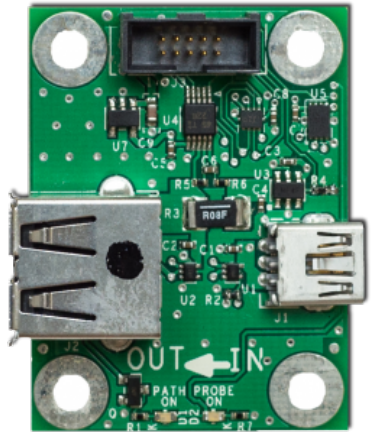
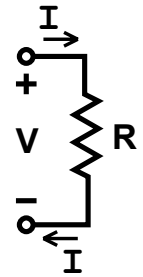


Image source: `http://baylibre.com/`

Power Measurement

Power Consumption $P = U \times I$

- Measure Voltage U
- Calculate Current I:
  - Measure Voltage U across a small Measurement Resistor R



  - Ohm's law:

$$U = R \times I \Leftrightarrow I = \frac{U}{R}$$

- Whole board / Subsystem power rails / Component

Image source: `http://www.wikipedia.org/`

Console

- Serial consoles
  - Legacy DB9/DB25 serial → USB-serial adapter
  - 1.8V/3.3V/5V UART → USB-serial adapter
  - On-board USB-serial chip
  - 96Boards UART
- USB hub with many ports (may need power)
- Use `udev` rules to pin names to your serial ports

  `/etc/udev/rules.d/99-usb-serial.rules`

```
SUBSYSTEM=="tty", ATTRS{idVendor}=="0403", ATTRS{idProduct}=="6001", \
    ATTRS{serial}=="A900YDVW", SYMLINK+="tty-ape6evm"
SUBSYSTEM=="tty", DRIVERS=="mos7840", ATTRS{port_number}=="0", \
    SYMLINK+="tty-aten0"
```

- Beware USB-serial chips with identical serial numbers! Fortunately USB devices can be addressed by topology

  `/dev/serial/by-path/...`

Network

- Ethernet Switch
  - Beware multiple boards with identical MAC addresses!
- Wireless Access Point
- Similar to networking *normal* Linux machines

Interaction

- Mostly input and control
- Output: complicated beyond console
- May require creative and custom solutions

Interaction

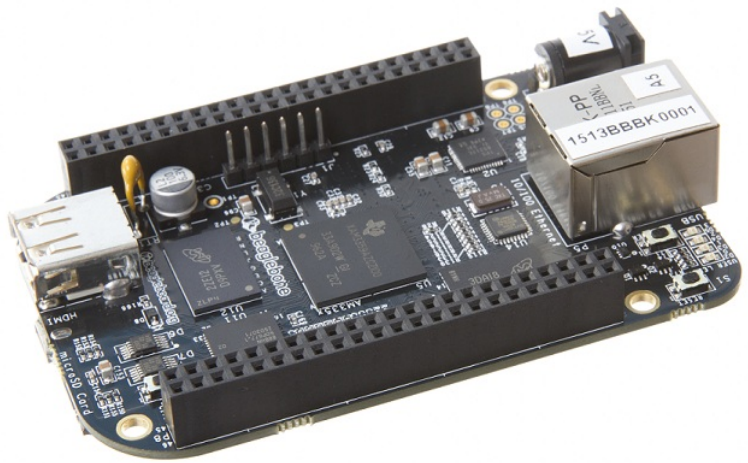
Switches and Buttons

- System Reset
- System Wake-Up
- Soft Power-On
- Generic input (e.g. keypad)
- ...

Interaction

Signal Inputs

- 2.54mm Female/Male Header
- 2 mm Female/Male Header



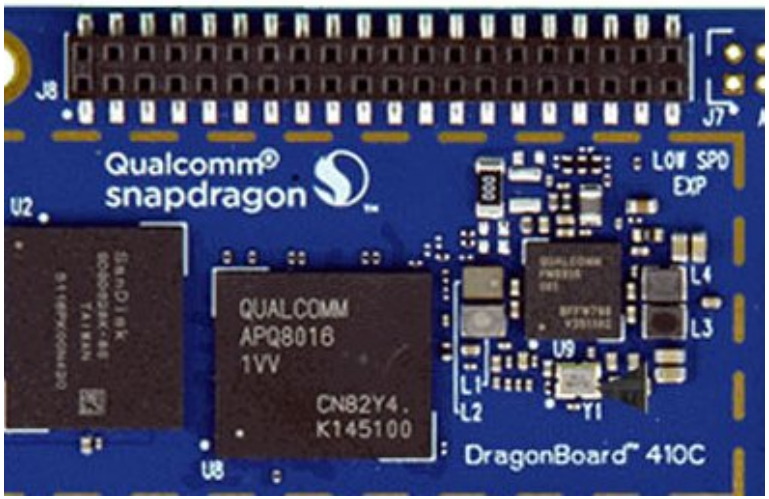
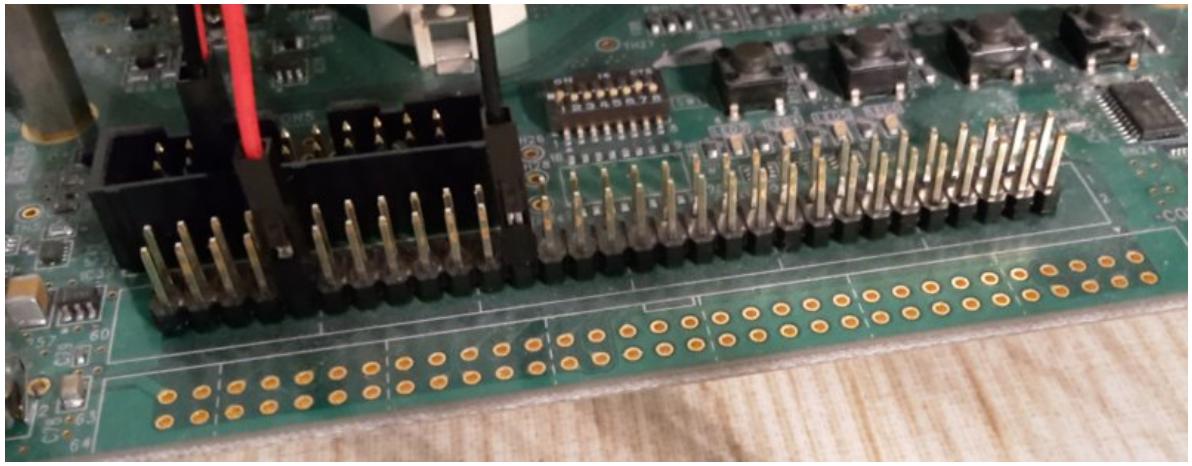


Image sources: `http://http://elinux.org/`, `http://www.96boards.org/`

Interaction

Signal Inputs

- Unpopulated Header $\rightarrow$ Solder header



- Test Point $\rightarrow$ Test Clip or Hook

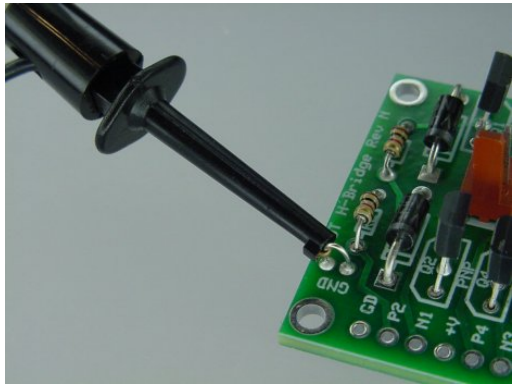


Image source: `http://www.robotroom.com/`

- High-Density Connector Breakout: Buy . . .

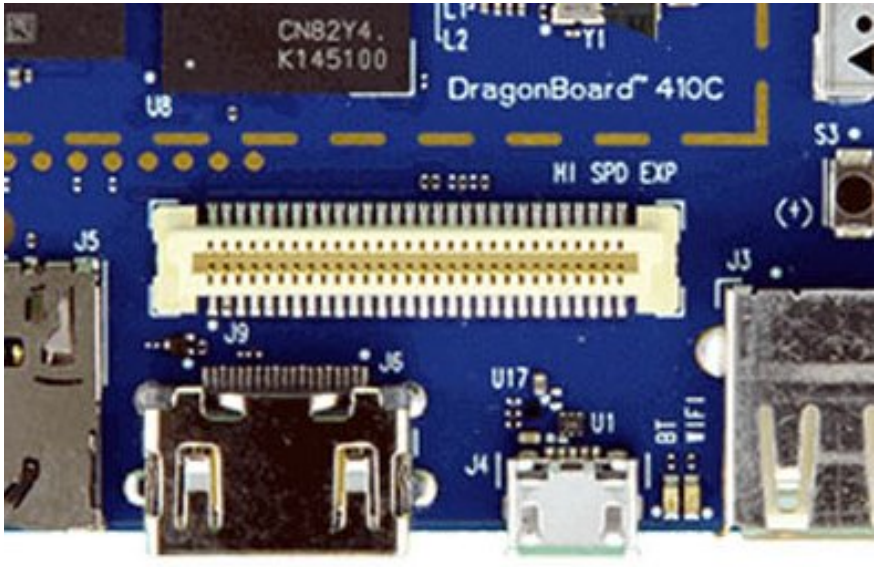
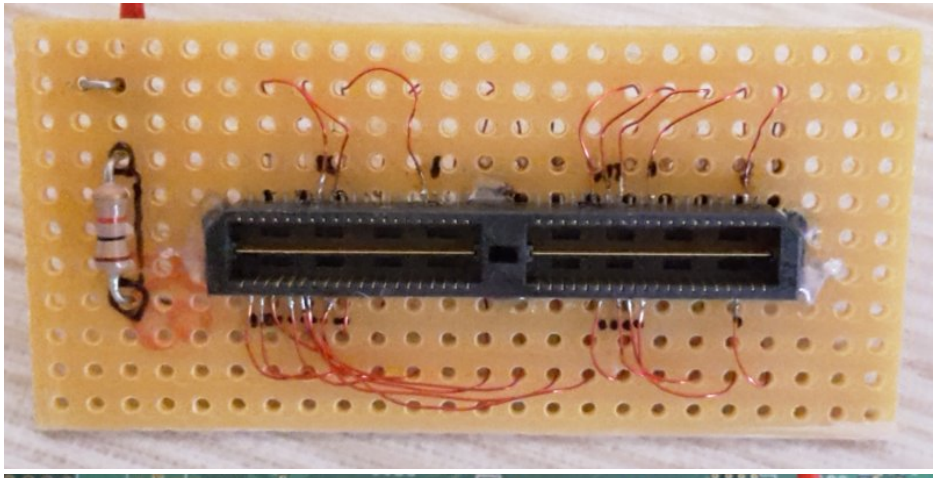
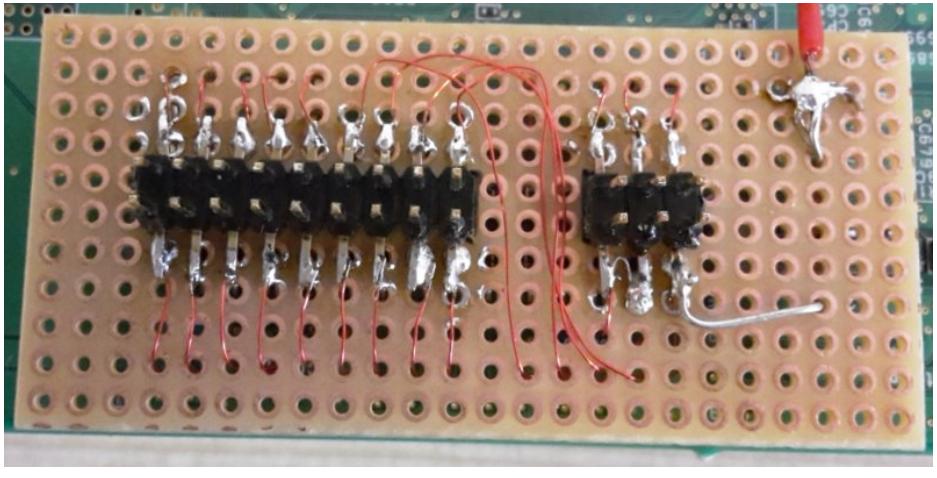


Image sources: `http://www.96boards.org/`, `http://zebax.com/`

- High-Density Connector Breakout: . . . or Build





- *Extreme Wiring on the Prototyping Board*

  `http://elm-chan.org/docs/wire/wiring_e.html`

- Solder to switch, or other component
- JTAG has reset
- Wake-Up needs an IRQ, or GPIO with interrupt capability
- Add any GPIO on expansion connector to gpio-keys in DT, to avoid having to solder a wire to a switch
- Caveats:
  - Signals are usually asserted by grounding
  - Sometimes asserted by pulling high (to which voltage?)
  - Positive voltage supply may be missing on connector ☹

Interaction

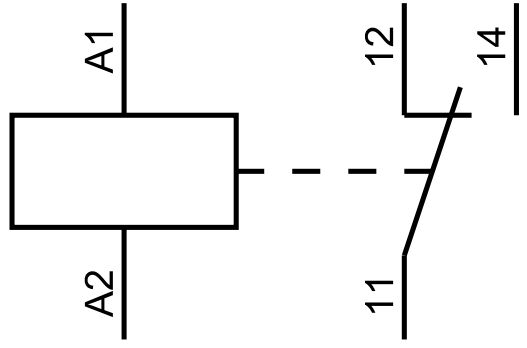
Switches and Buttons: GPIO

```
keyboard {
        compatible = "gpio-keys";
        pinctrl-names = "default";
        pinctrl-0 = <&keyboard_pins>;
        key-wakeup {
                gpios = <&gpio2 1 GPIO_ACTIVE_LOW>;
                label = "EXIO-D-50";
                linux,code = <KEY_WAKEUP>;
        };
};
&pfc {
        keyboard_pins: keyboard {
                pins = "GP_2_1";
                bias-pull-up;
        }
};
```

Interaction

How to Control All Those Signals?

- GPIOs with/without driving Transistors/MOSFETs
  - X No isolation
- Relays
  - Electromagnetically Controlled Switch



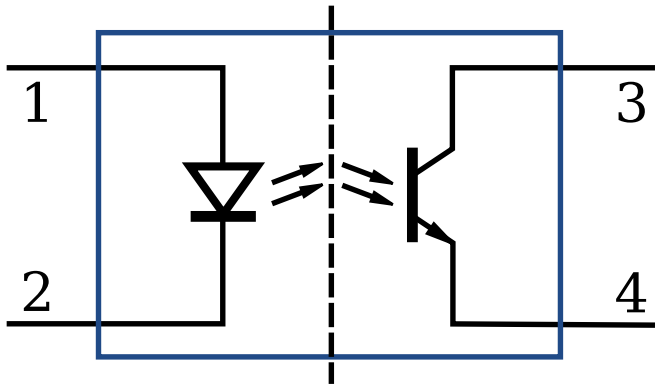
  - ✓ Isolation
  - ✓ Relay Boards readily available
  - X Noisy
  - X Overkill for most input signals

Image source: `http://www.wikipedia.org/`

Interaction

How to Control All Those Signals?

- Opto-Isolators
  - Light Controlled Switch

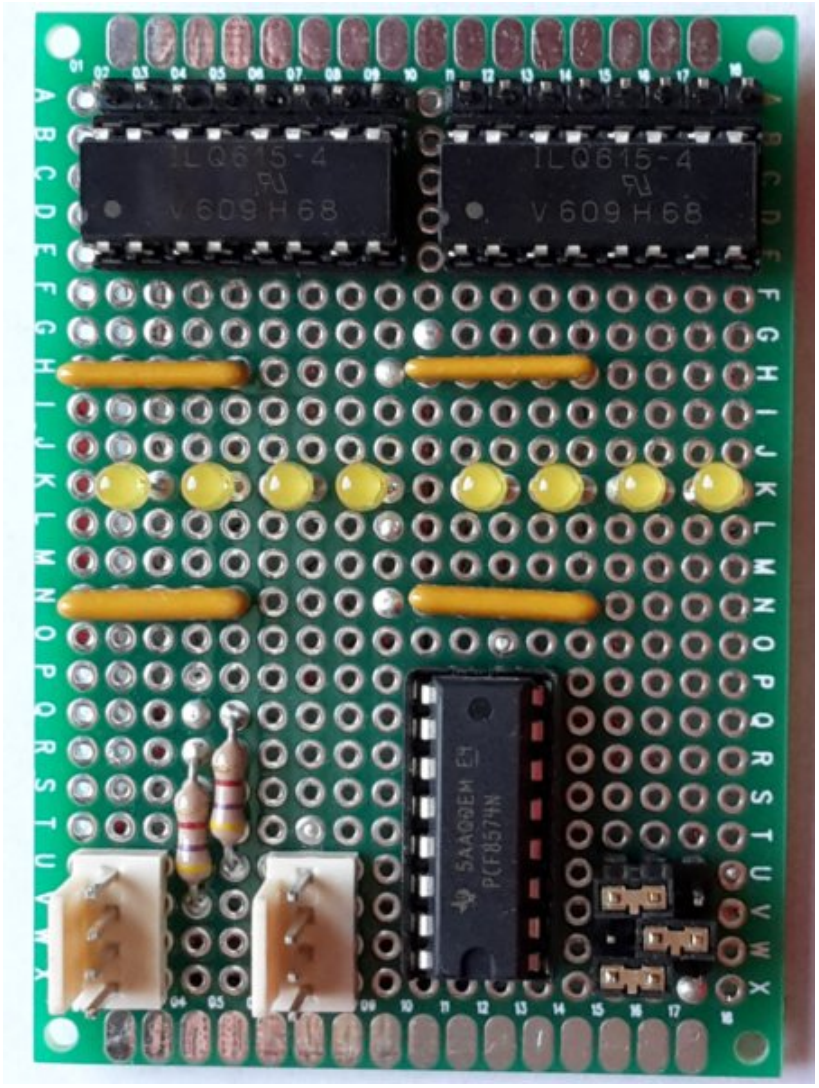


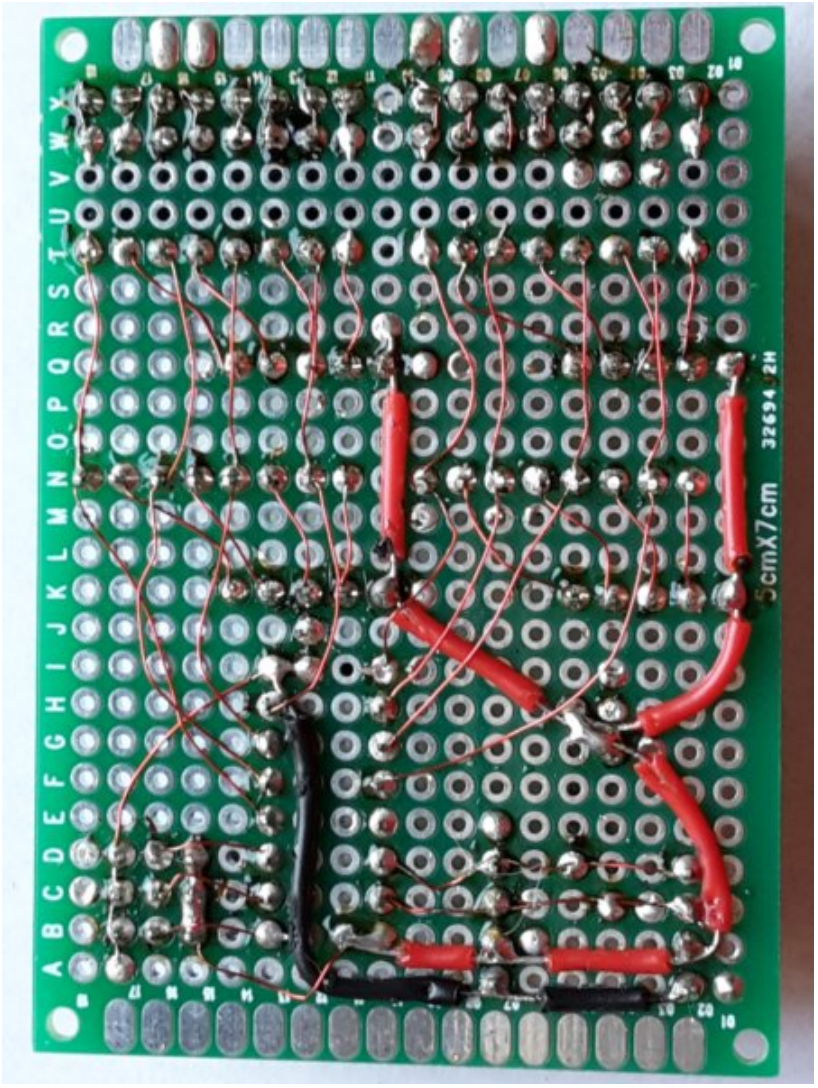
- ✓ Isolation
- ✓ Can switch +1.8V, +3.3V, +5V, . . .
- Add a relay if needed
- Polarity!

Image source: `http://www.wikipedia.org/`

Interaction

Eight Opto-Isolators Driven by I2C GPIO Expander

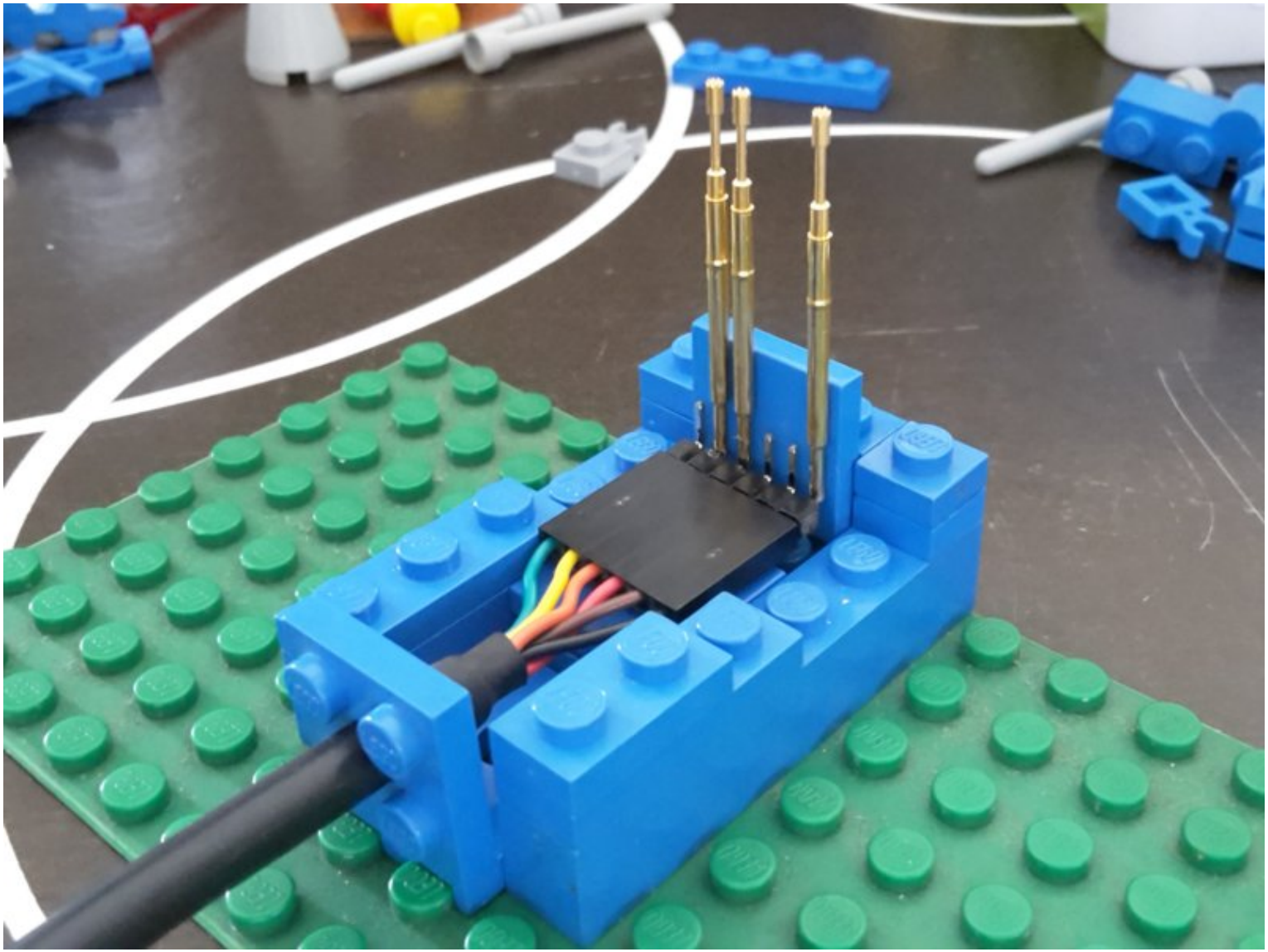




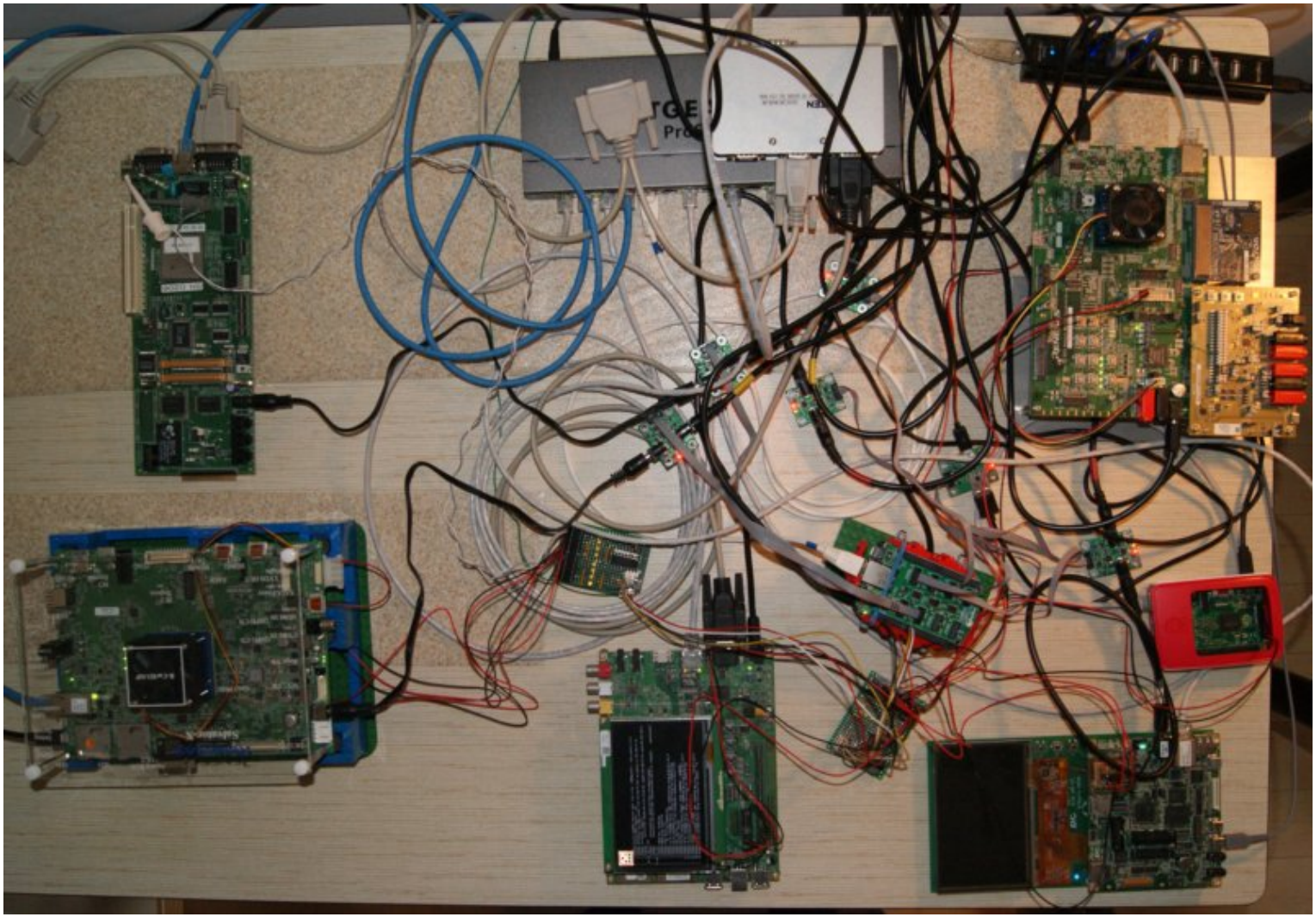
Management Host

- Control and monitor all blocks
- Provide services
- Old PC, embedded x86
- Embedded Development Boards becoming more powerful
  - E.g. BeagleBone Black, Raspberry Pi, . . .

Beagle Bone Black Console Bed



Board Farm Bird View



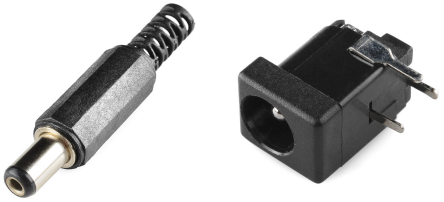
Power Supplies

- Each board comes with its own power supply
- Wall wart rats nest

- Can we improve upon?

Board Power Needs

- Most boards take either 5V or 12V
- Different connectors types, voltages, and polarities
  - 2.1/5.5mm or 2.5/5.5mm jack
    - 5V, 7.5V, 9V, 12V, . . . , up to 9A
    - Most (not all!) are Center-positive



  - EIAJ connector, Center-positive, 2A

#1	0–3.15V	2.35/0.7mm
#2	3.15–6.3V	4.0/1.7mm
#3	6.3–10.5V	4.75/1.7mm
#4	10.5–13.5V	5.5/3.4/1.0mm
#5	13.5–18V	6.5/4.4/1.4mm

← 96Boards 8–18V ☹

  - USB mini/micro-B
- Need for conversion when used with ACME

Image source: `https://www.sparkfun.com/`

Identical and Low Power Needs

- Powered USB hub for e.g. Beowulf of Raspberry Pis
- Barrel jack splitters (2-way, 4-way)

- Usually limited to 2A, maximum current is seldom advertized

Image source: `https://solarbotics.com/`

Single Power Supply

- My needs:
  - 8 Boards + Management Host & Control Hardware
  - 13A @ 5V $\rightarrow$ 65W
  - 28A @ 12V $\rightarrow$ 336W
  - Absolute maximum ratings!
- Lab power supply
- PC power supply

PC Power Supply

Multiple Output Voltages

+5V	For development boards
+12V	For development boards
	▸ Single or Dual Rail!
+3.3V	For an MSP430 LaunchPad?
-12V	Not so useful without real RS232

Management Host Features

+5Vsb	+2A is ample
PS_ON	Remote control

PC Power Supply

Minimum Load?

Older supplies may need some load to work

- Look for *Haswell C6/C7 Zero Load Support*
- Use e.g. Ethernet switch as load, or load resistors

Voltage Stability / Power Rail Cross Impact

- Not so much of an issue anymore
  - Single rail 12V with DC/DC converters for 3.3V and 5V
  - SoCs run at low voltages, board has own PMIC
  - Boards that need stable +5V signals typically run from 12V
  - Any boards that still need stable +12V signals?
- Power Supply needs some time to stabilize!
  - Turn power supply on first, individual boards last
  - Turn individual boards off first, power supply last
  - `PWR_OK` signal

PC Power Supply

Safety

Watch out for high currents!

- Typical 650W PC Power Supply
  - 52A @ 12V
  - 22A @ 5V
  - Low voltage, but high current, needs thick wires
  - My induction stove needs only 32A (@ 230V, though)
- Do not feed everything from one wire!
  - 4A per wire
  - Modular Power Supply can still be handy
- Fuses for individual boards
  - The PS should be designed to survive a short circuit
  - Your Raspberry Pi may not (@ 20A)

PC Power Supply

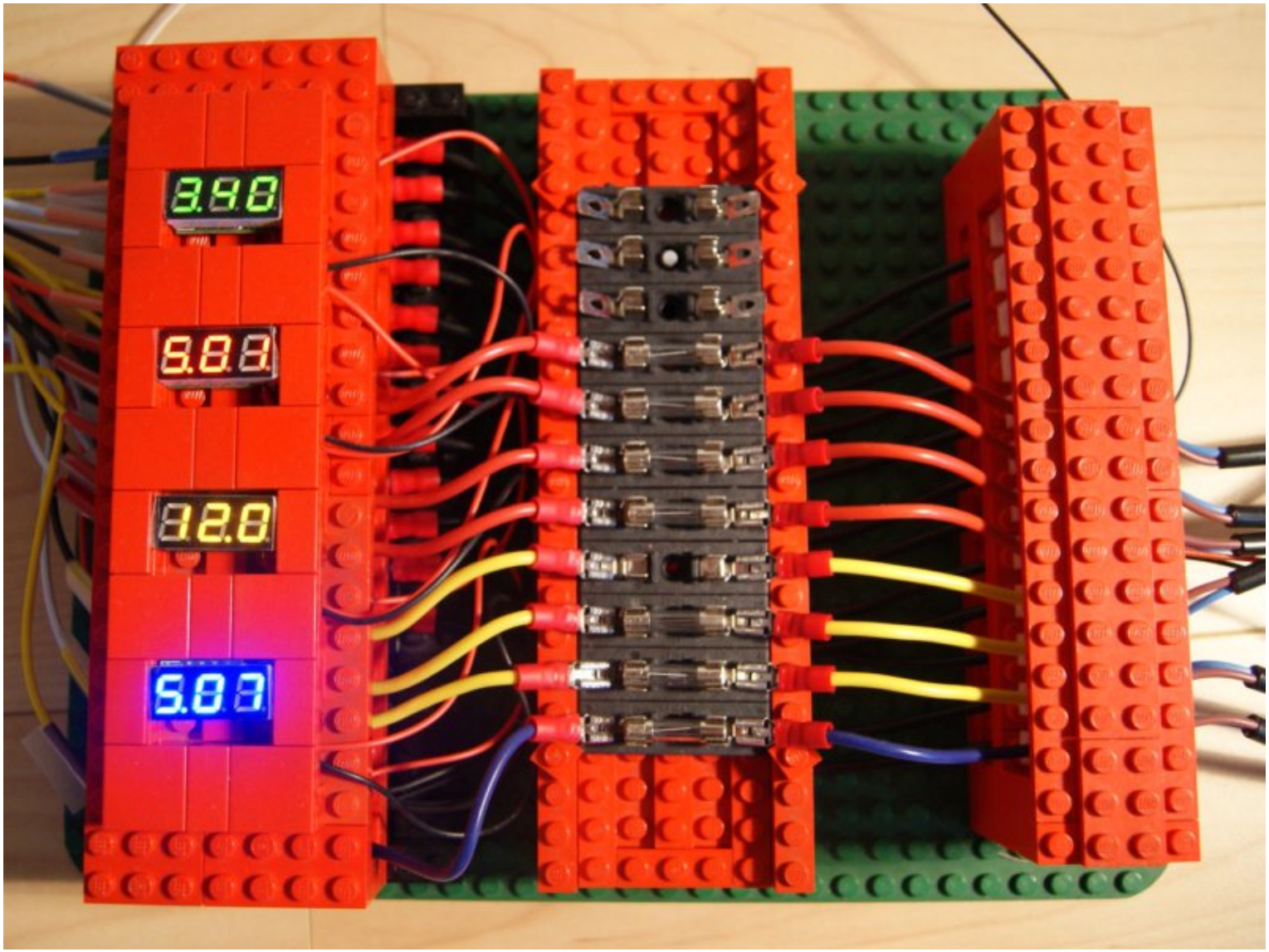
Tips & Tricks

- Use Ethernet switch with 12V input (some need e.g. 7.5V)
- Most USB hubs need 5V

- +12V rail is the major rail on modern supplies
- +5V limited to 20–25A, independent of total wattage
- Many boards that need 5V? $\rightarrow$ add a DC/DC converter

PC Power Supply

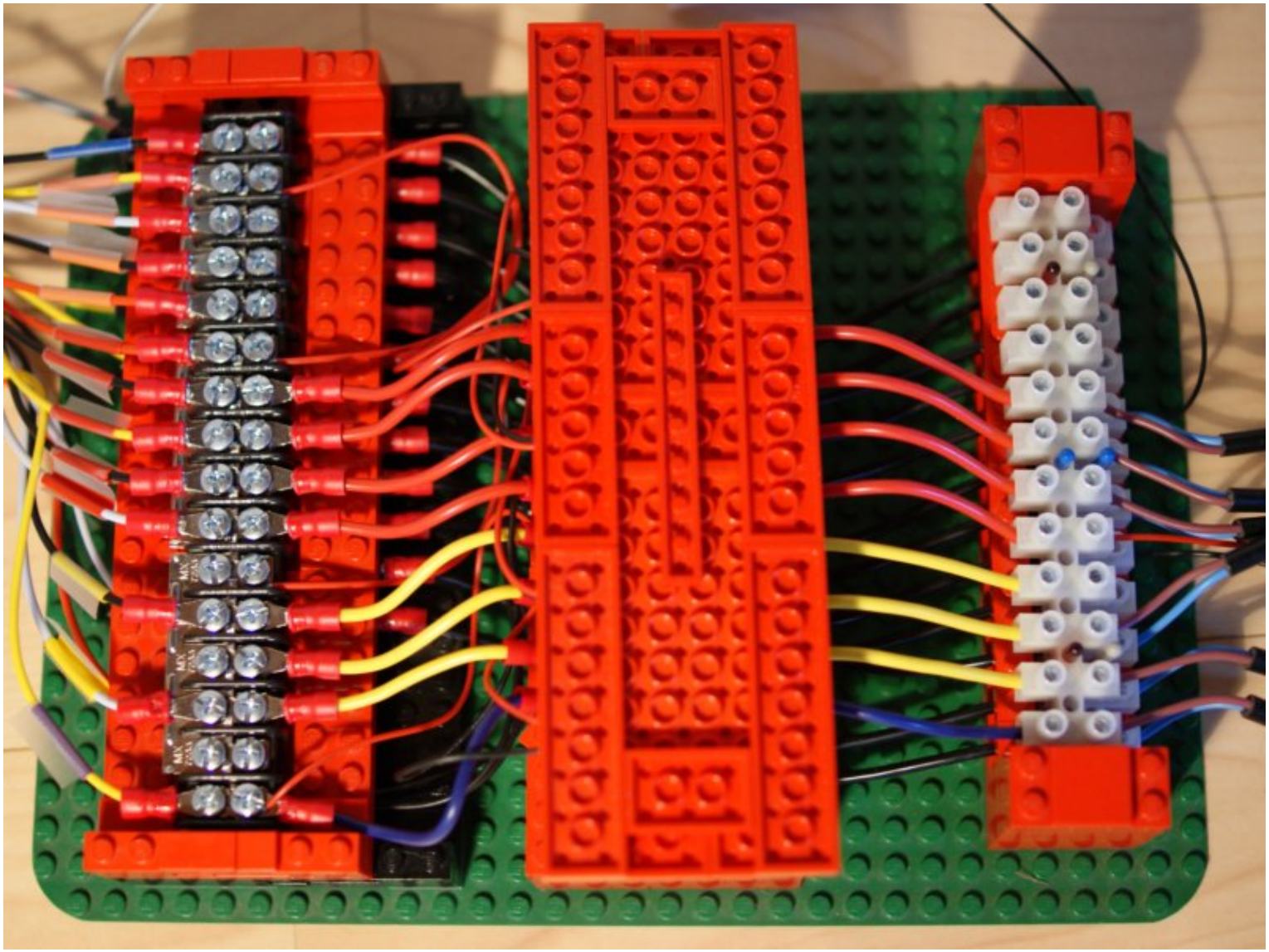
Before 3D Printing, There Existed LEGO



Beau Barrier Strips — Fuses — *Eurostyle* Barrier Strips

PC Power Supply

Before 3D Printing, There Existed LEGO



Beau Barrier Strips — Fuses — *Eurostyle* Barrier Strips

Software Side

GPIO Control

- Add PFC8574 GPIO expander at I2C address 0x24:

```
if i2cget -y 1 0x24 >& /dev/null; then
        echo pcf8574 0x24 > \
                /sys/class/i2c-adapter/i2c-1/new_device
        for i in $(seq 480 487); do
                echo $i > /sys/class/gpio/export
        done
fi
```

- Toggle reset line at gpio 480 using sysfs:

```
echo out > /sys/class/gpio/gpio480/direction
sleep 0.2
echo in > /sys/class/gpio/gpio480/direction
```

- Should try new GPIO chardev interface . . .

Software Side

ACME

- Sigrok integration
- PulseView GUI
- Command line

```
# sigrok-cli --driver=baylibre-acme --samples=1
FRAME-BEGIN
P1_ENRG_PWR: 1.225000 W
P1_ENRG_CURR: 0.246000 A
P1_ENRG_VOL: 5.019000 V
...
FRAME-END
```

- sysfs GPIO for board power control, or `sigrok-cli`
- ACME powers on all boards during boot up ☹
  → Edit `/etc/init.d/S95acme-init`:

```
- echo 1 > /sys/class/gpio/gpio$GPIO/value
+ echo 0 > /sys/class/gpio/gpio$GPIO/value
```

Software Side

Big collection of scripts:

- `main-power-{on,off,status}`
- `<board>-power-{on,off,status}`
- `<board>-acc-{on,off,status}`
- `<board>-{reset,wakeup}`
- `...`

```
ape6evm        :   4.850000 W    0.406000 A   11.928000 V
armadillo      :   3.025000 W    0.620000 A    4.893000 V
h3-salvator-x  :   9.625000 W    0.812000 A   11.851000 V
kzm9g          :   1.800000 W    0.364000 A    4.938000 V
rbtx4927       :   2.425000 W    0.490000 A    4.901000 V
rpi            :   1.225000 W    0.250000 A    4.879000 V
-----------------------------------------------------------
Total          :     22.95 W
Total at 5V    :     8.475 W       1.724 A
Total at 12V   :    14.475 W       1.218 A
```

Software Side

- Booting (TFTP / DHCP / NFS root): cfr. board on your desk
- TODO: Automated boot
  - Boot testing
  - Auto-bisecting regressions
  - Join `kernelci.org`?

Q *How can I participate in the boot test phase?*
A *The best way to participate is to send us your boards*

Source: `https://kernelci.org/faq`

Advanced Features

My Board Farm fulfills **my** requirements.

You may want:

- JTAG
- VNC for display output
- Board sharing
- Virtualization to isolate multiple users
- …

Final Words

What have I learned?

- Improved soldering skills
- Modern PC power supplies work with zero load
- It takes a while to get the details right

What can I improve?

- More automation
- Better UI for controlling boards
- Create my own PCB for the Opto-Isolator Board
- Get a real case for the power distribution parts
- Better furniture for the whole farm

Thanks & Acknowledgements

- **Renesas Electronics Corporation**, for contracting me for Linux kernel work, and supplying me with development boards,
- The **Linux Foundation**, for organizing this conference and giving me the opportunity to present here,
- **BayLibre**, for creating ACME,
- The **Renesas Linux Kernel Team**, for insights and discussions,
- The **Linux Kernel Community**, for having so much fun working together towards a common goal.

Questions?