

Developing a Keyboard from Scratch

Danny Robson

<DrHeroic> Do you want to make a mechanical keyboard?
<gim> How hard could it be?



Requirements

- Sufficiently affordable.
- Customisable.
- Pretty.
- Suitable for gaming.

Hardware

- *Somewhat* off the shelf components
 - Switches
 - Keycaps
 - PCB
 - Case
 - Cables

colour

clicky

tactile

linear

Switches

colour

clicky

tactile

linear

brown

yes

red

yes

blue

yes

yes

Keycaps

ABS

cheapest, moderately durable, yellows

PBT

expensive, durable

Polycarb

transparent

PCB

- Based off the GH60 design
 - ATmega32 for USB, keys
 - ATtiny85 for RGB LED control
 - 8 RGB LEDs over I2C

Case

- 4 layers of laser cut acrylic
 - Wildly divergent quality in suppliers
 - Moving to water cut polycarb
 - Somewhat finnickly to `get right'



USB Cables

- Paracord
- Cat6 cable
- Time

Construction

1. Solder switches onto PCB
2. Assemble case around PCB
3. Attach keycaps
4. Cover cable

Development

- avr-gcc
- x86_64-w64-mingw32-gcc
- make, autotools
- wireshark
- Scotch

Development

- Goals
 1. Fix the core firmware functionality
 2. Control the LEDs from the host machine

Firmware

- Based off ps2avrU
 - Based off ps2avr
 - For the `IBM Model M'
- But it works.
 - Mostly.

Firmware



The single worst code base I've ever laid eyes on.

Firmware



8 RGB LEDs, but 7 colours.

Firmware



What does RGB LED mode 4 do?

Firmware



Is there an RGB LED mode 5? 6? 7?

Firmware



Why is requesting 100 bytes of data the trigger to send the 52 bytes of configuration data?

Firmware



Why does holding 6 keys down crash the firmware?

Firmware

- Nothing terrifically interesting. Just lots of staring at a console
 - wireshark
 - ctags
 - Google

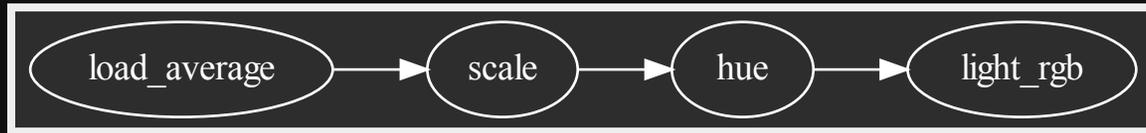
Blinkenlights

- RGB backlights are fun, but can they be made useful?
- Idea: change the hue based on loadaverage.

Blinkenlights

- light_daemon
 - Devices & lights expose a list of controls
 - Sources generate values over time
 - Sinks accept values

Blinkenlights



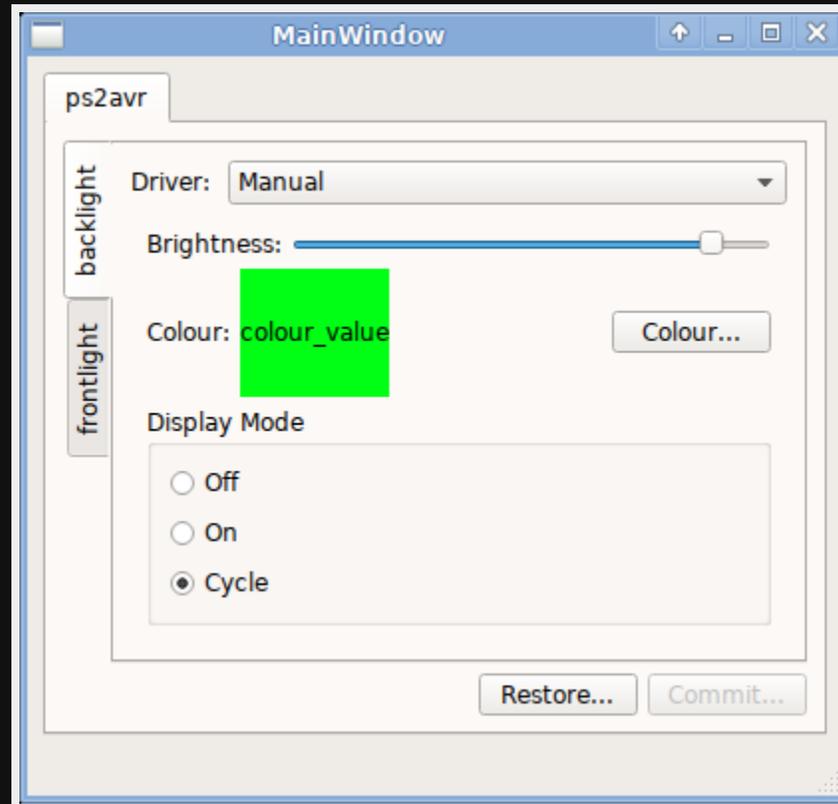
Blinkenlights: Attempt #1

- Use libusb to send settings data
 - Detaches devices under Linux
 - Doesn't work for keyboards/mice under Windows

Blinkenlights: Attempt #2

- hidapi
 - Very restrictive API.
 - Works under Windows

Blinkenlights



TODO

- Finish lighting automation
- Robustify the firmware
- `Proper' user interface

