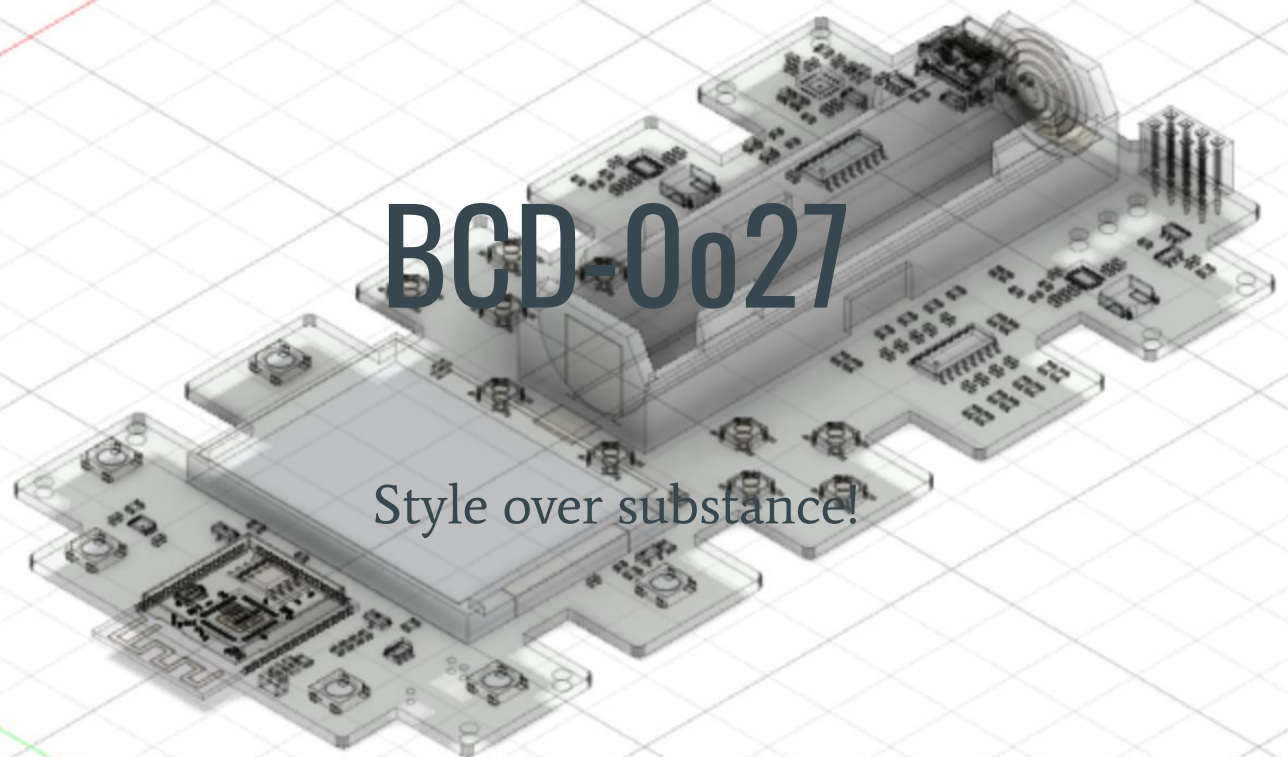


BCD-0027

Style over substance!

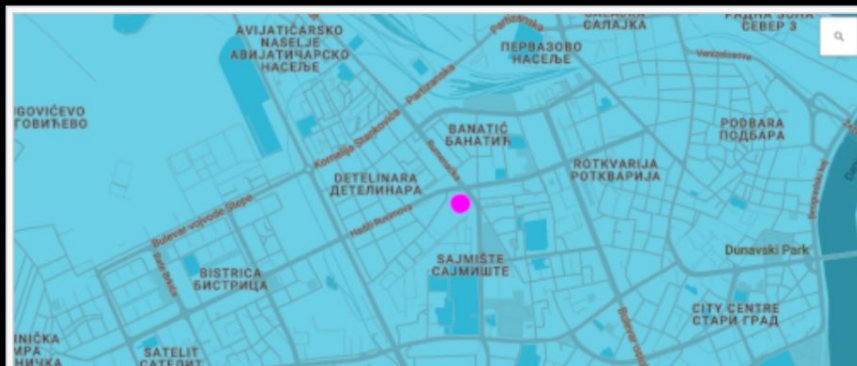


You may have seen...

BaltConZk23 Badge

Secret Prototypes Leaked

Greeting fellow console cowgirls and -boys. Today I will reveal some real nuggets. So put your privacy shield up and get into a quiet corner, as you sure don't want to spread word too far. Our special envoy going by the handle of Acid was not expecting much when scouting out a small black market in Serbia. However, after downing a couple of rakijas to blend in with the locals, she found out that a small group of edge runners is working on a rogue cyberdeck. While corpos sure won't like that, they managed to stay undetected so far.



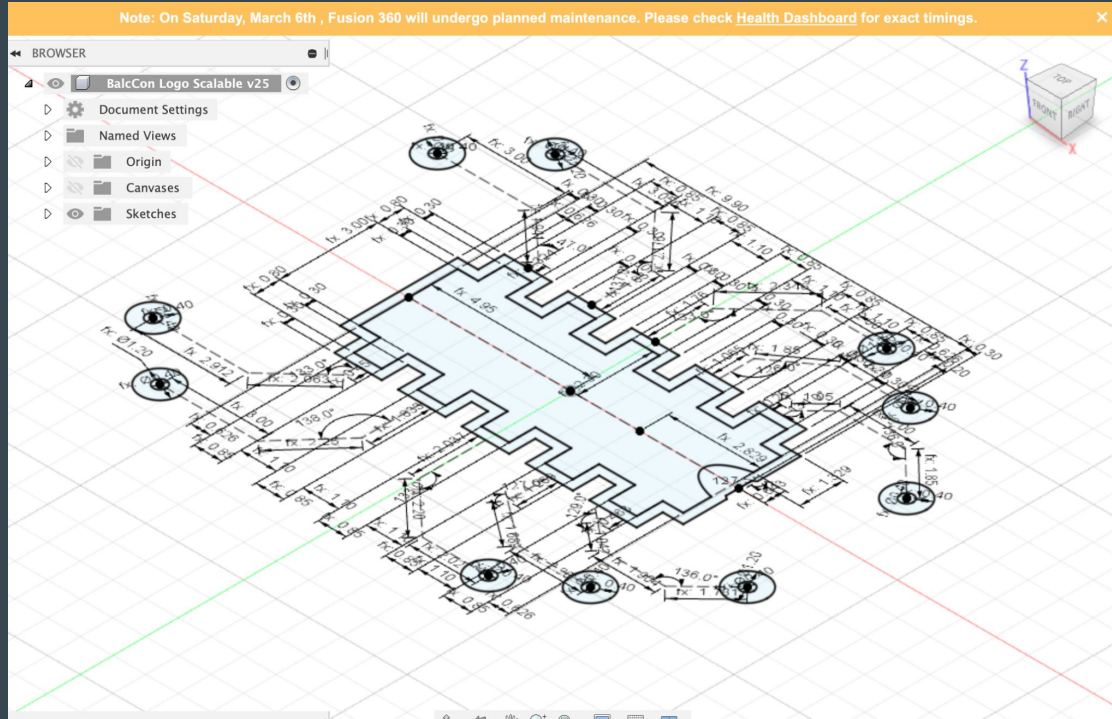
TL;DR

The cyberdeck comes with a simplistic firmware. This is because - as any console cowgirl or cowboy knows - you need to make a cyberdeck your own. There is a framework that makes it easier for you to develop your own customisations and extensions.

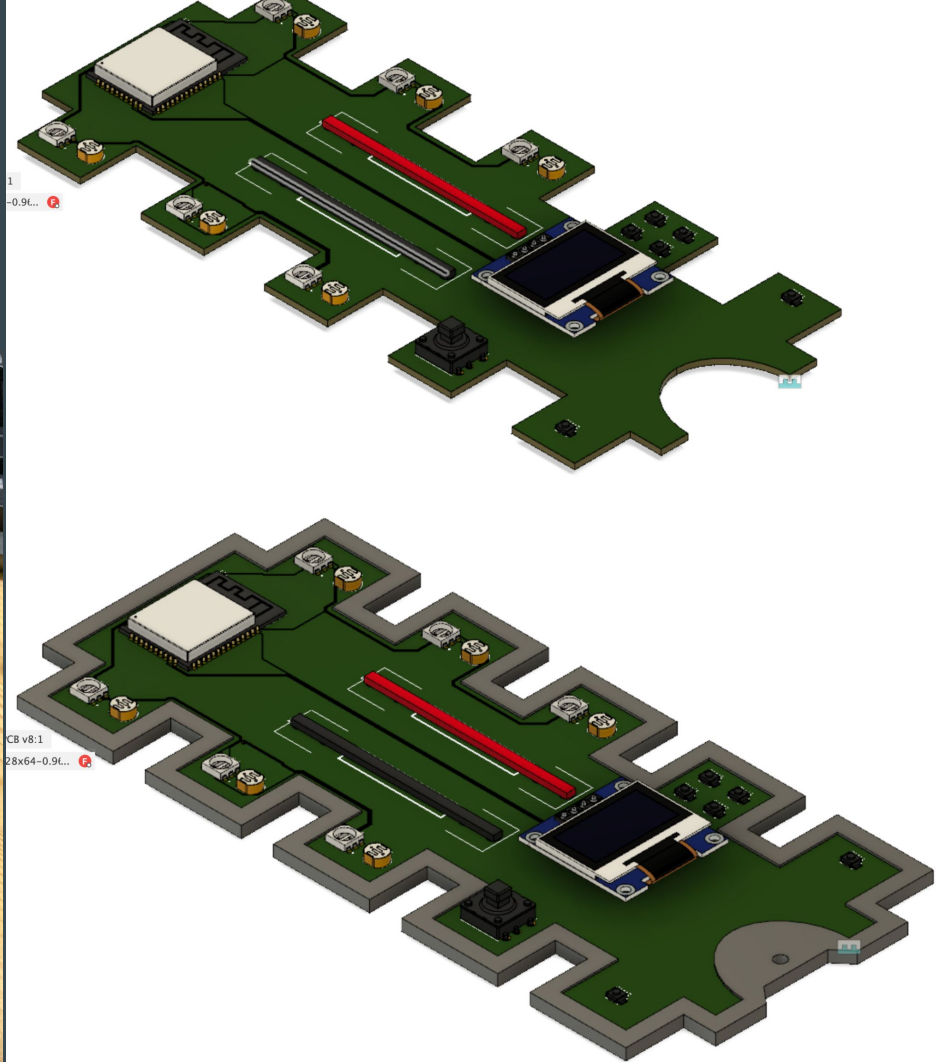
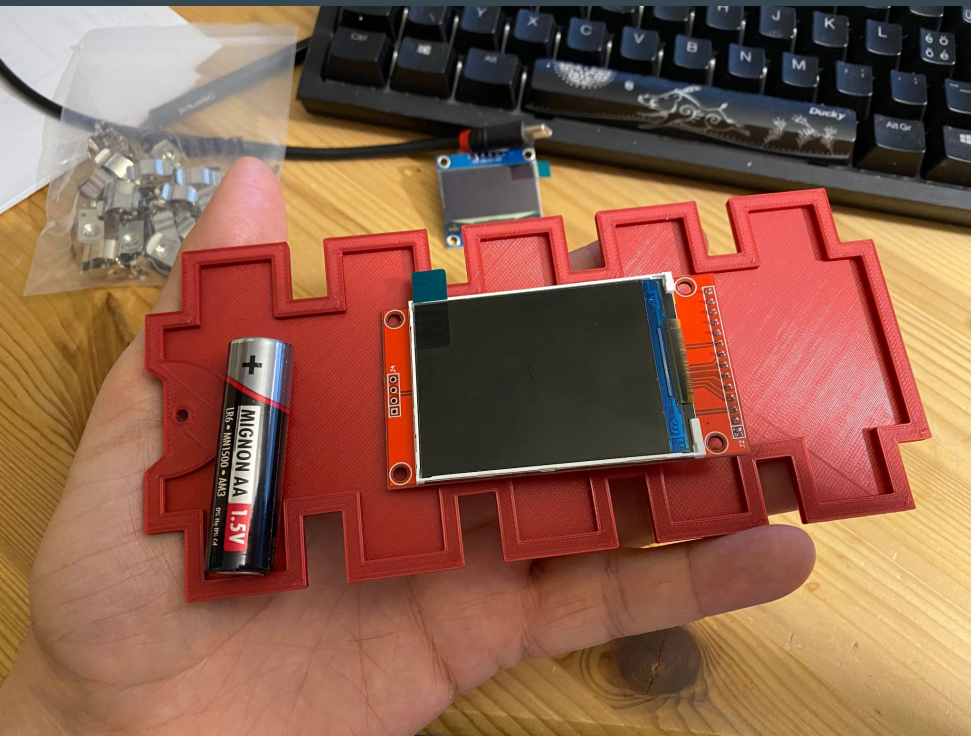
Your cyberdeck might hide some secrets. Find them and help Acid to accomplish her mission.

Also, there is a developer challenge. Stick to the end of the talk to learn more about it.

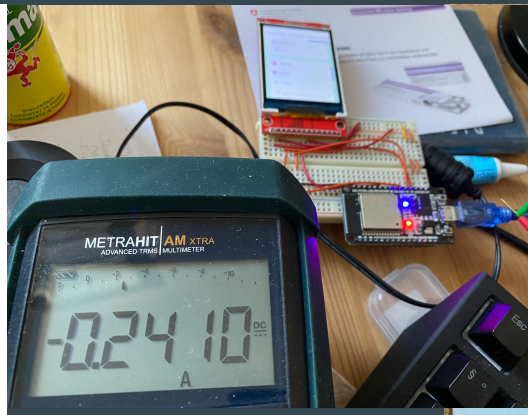
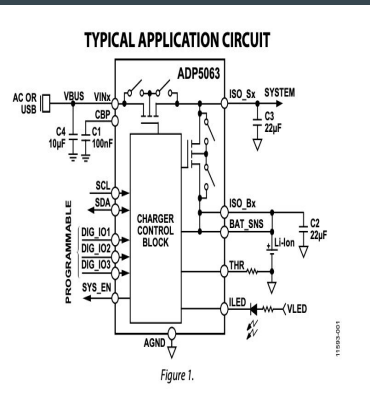
First idea of a Badge



And some Concept Sketches



Sketching Parts of the Design

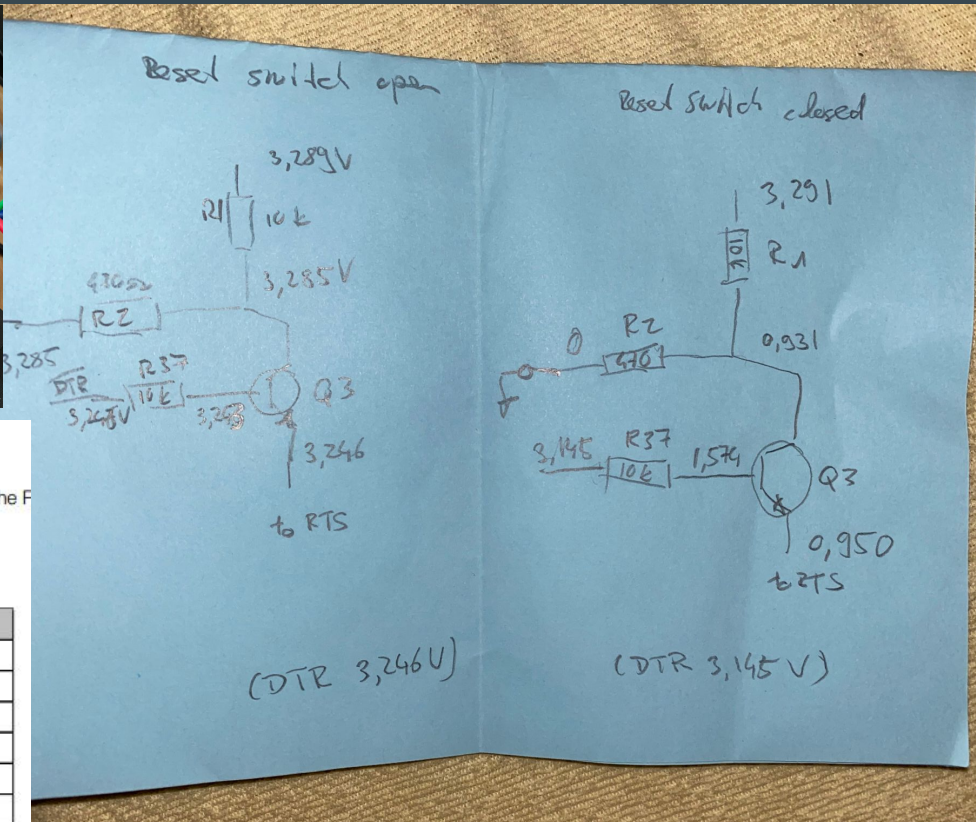


5.5 RF Power-Consumption Specifications

The power consumption measurements are taken with a 3.3 V supply at 25 °C of ambient temperature at the F port. All transmitters' measurements are based on a 50% duty cycle.

Table 15: RF Power-Consumption Specifications

Mode	Min	Typ	Max	Unit
Transmit 802.11b, DSSS 1 Mbps, POUT = +19.5 dBm	-	240	-	mA
Transmit 802.11g, OFDM 54 Mbps, POUT = +16 dBm	-	190	-	mA
Transmit 802.11n, OFDM MCS7, POUT = +14 dBm	-	180	-	mA
Receive 802.11b/g/n	-	95 ~ 100	-	mA
Transmit BT/BLE, POUT = 0 dBm	-	130	-	mA
Receive BT/BLE	-	95 ~ 100	-	mA



Prototype Parts of the Circuit

Conductor Characteristics

Solve For: Amperage Conductor Width Conductor Length

Plane Present? No Yes

Parallel Conductors? No Yes

Conductor Width: mm

Conductor Length: mm

PCB Thickness: mm

Frequency: AC DC

Distance to Plane: mm

IPC-2152 with modifiers mode Etch Factor: 1:1

Power Dissipation: Watts

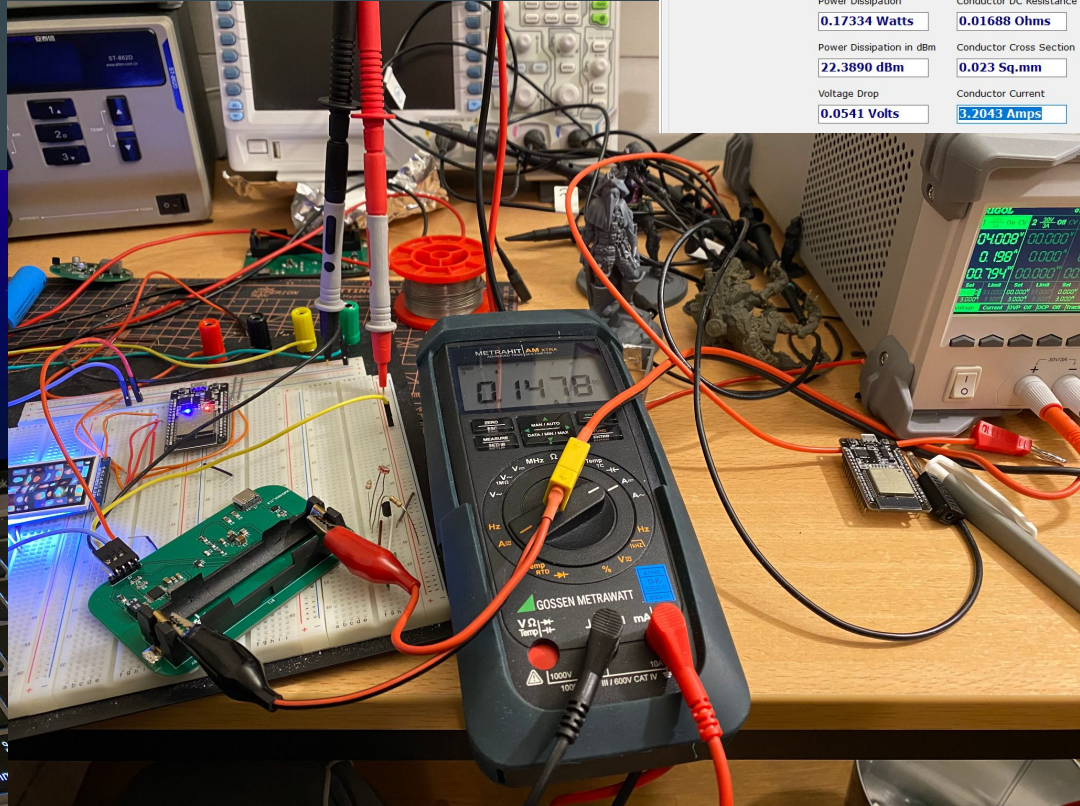

Power Dissipation in dBm: dBm

Voltage Drop: Volts

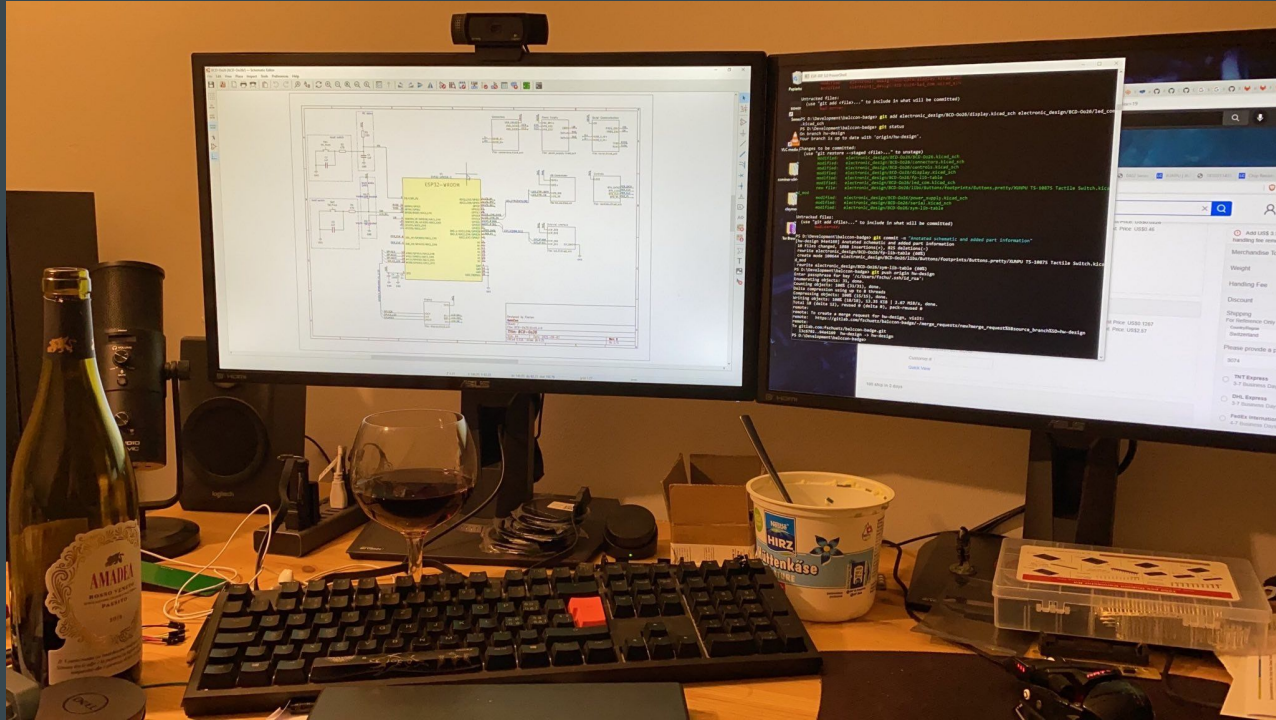
Conductor DC Resistance: Ohms

Conductor Cross Section: sq.mm

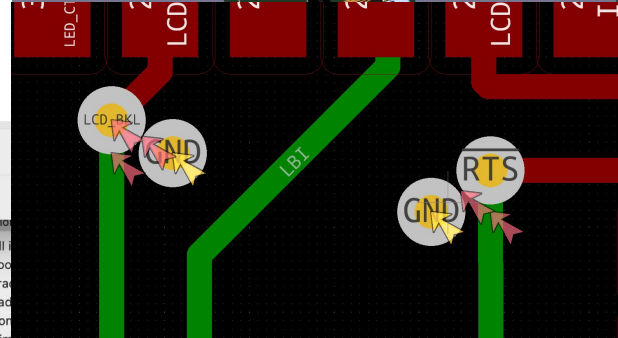
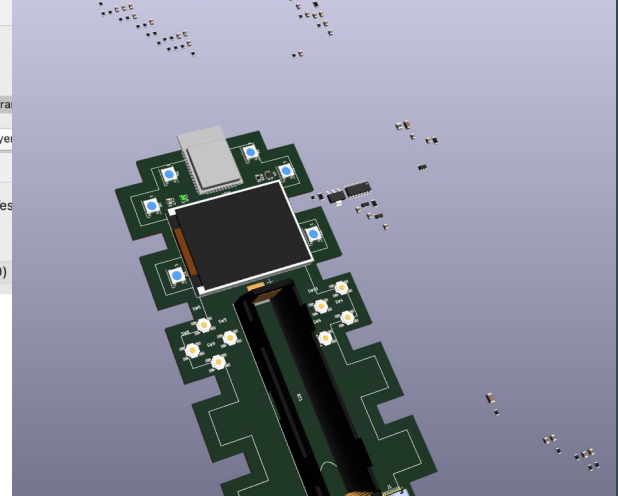
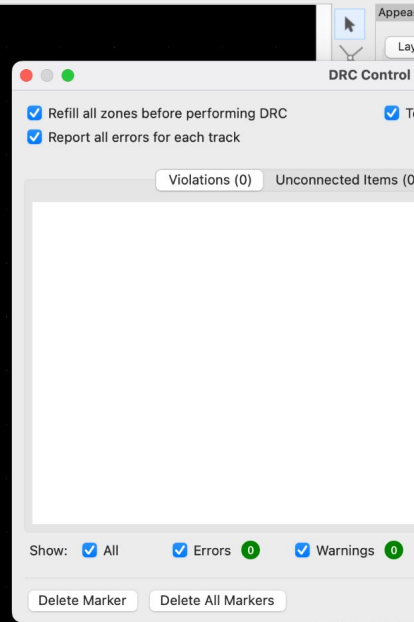
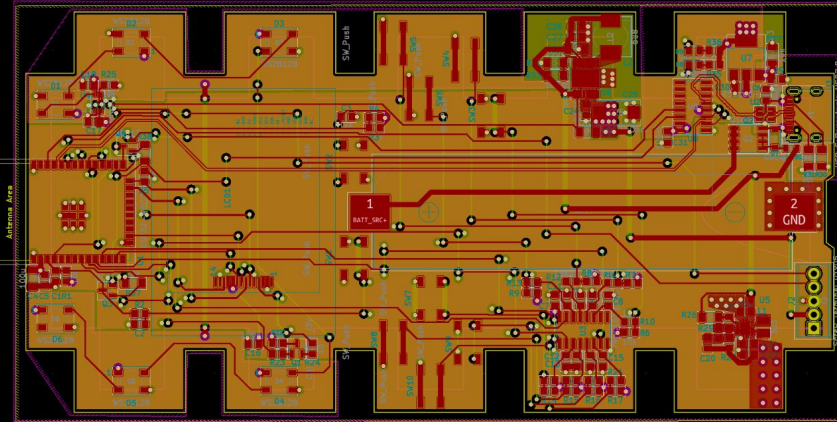
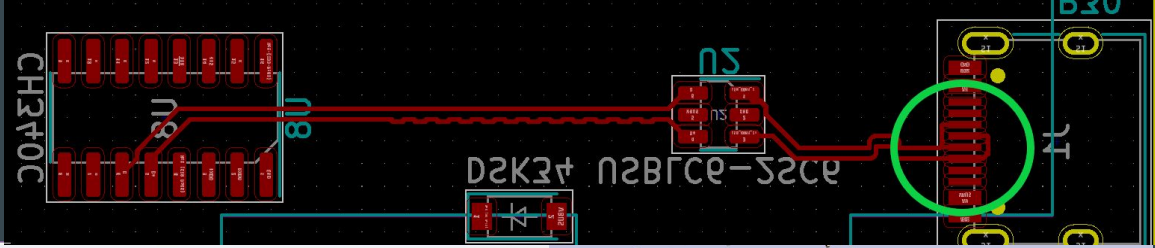
Conductor Current: Amps



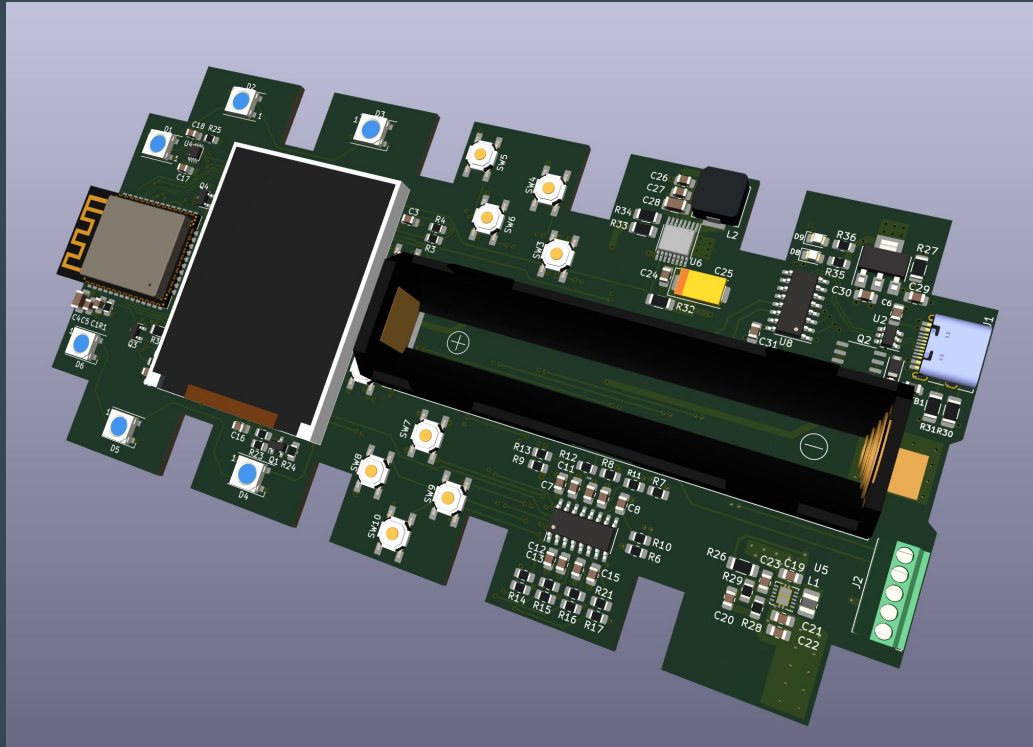
Making the Schematic



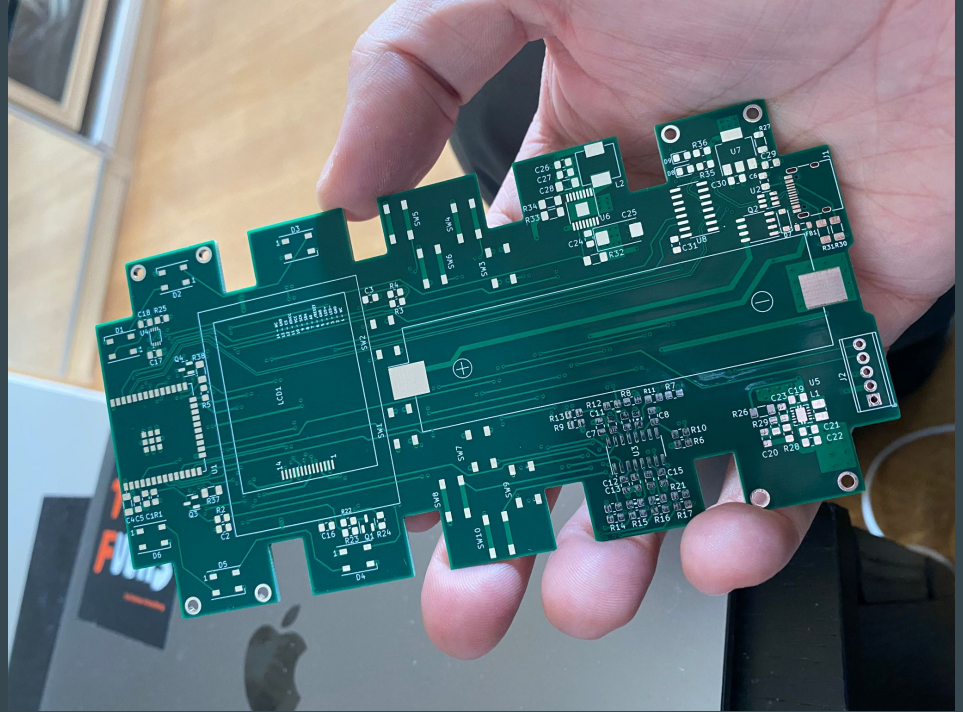
Routing



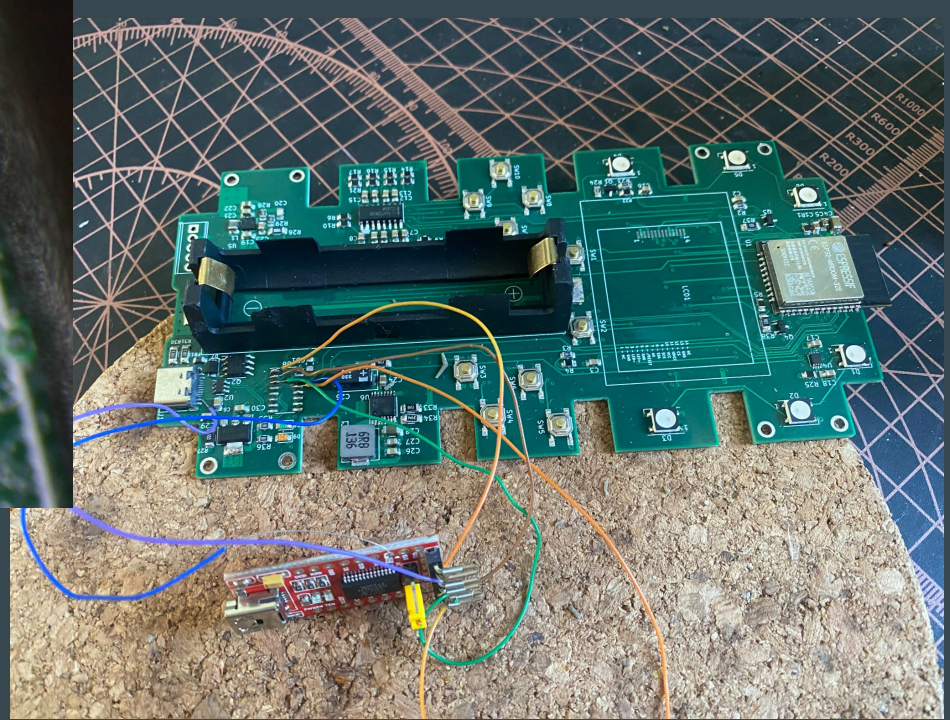
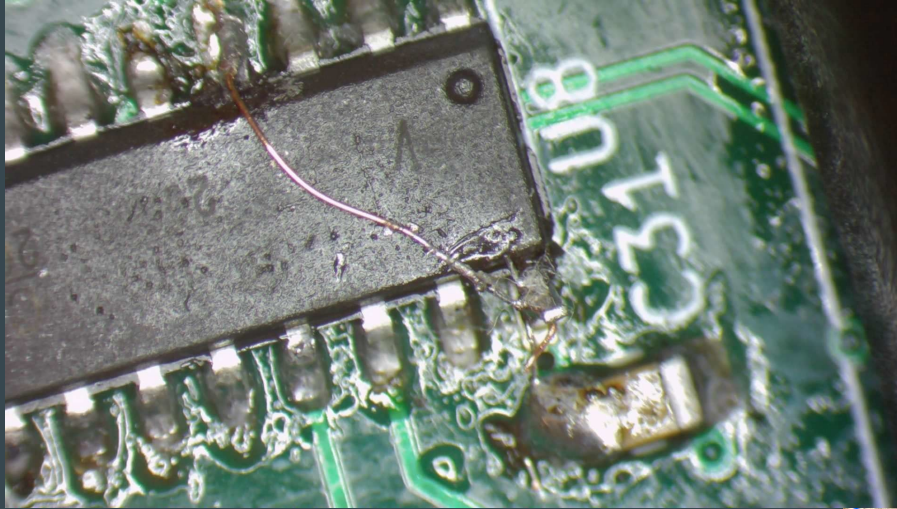
Design Finalised



First Prototype

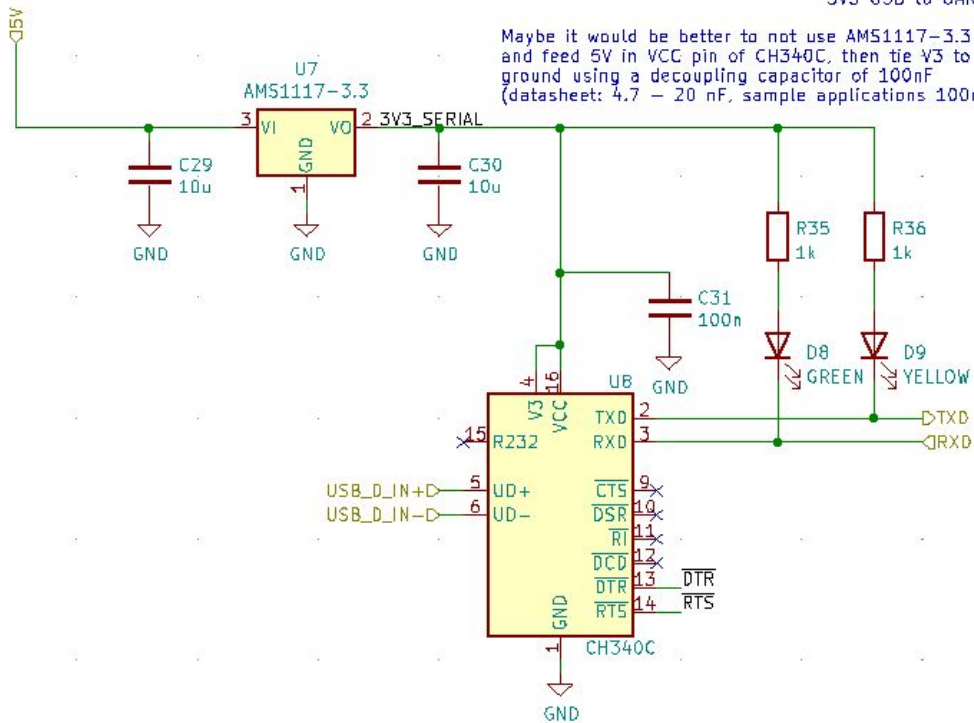


Fixing the Serial Programmer

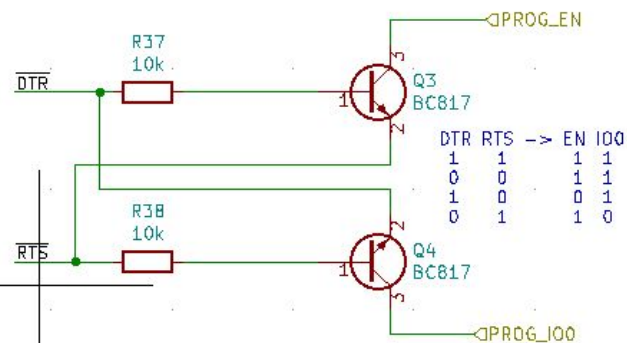


3V3 USB to UART

Maybe it would be better to not use AMS1117-3.3 and feed 5V in VCC pin of CH340C, then tie V3 to ground using a decoupling capacitor of 100nF (datasheet: 4.7 – 20 nF, sample applications 100nF)

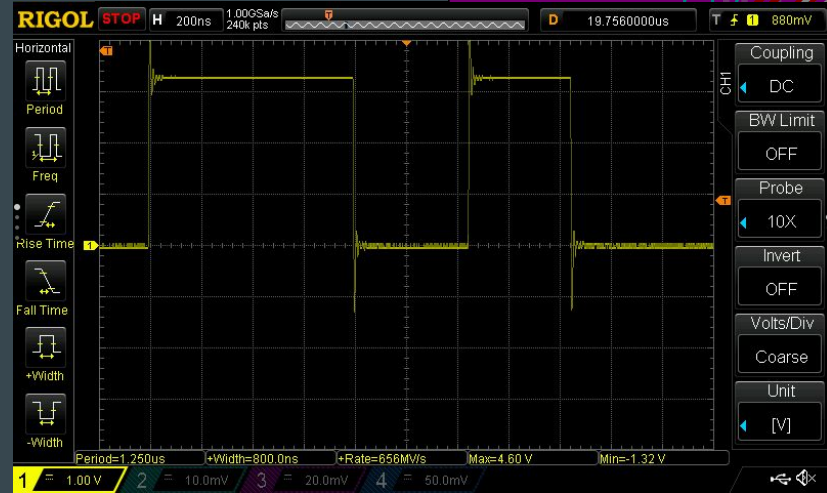
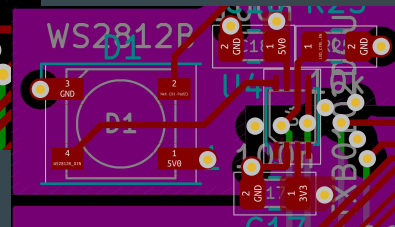
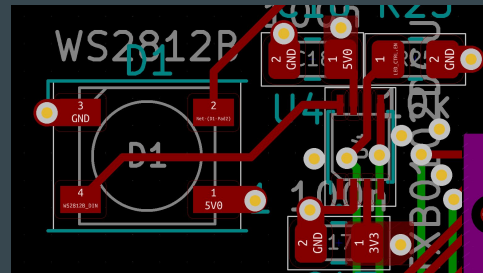
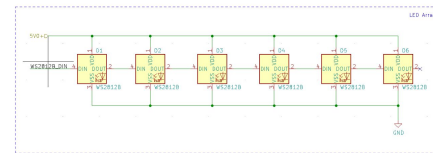
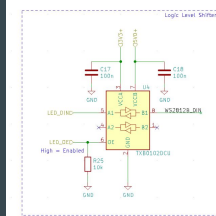


Auto programmer

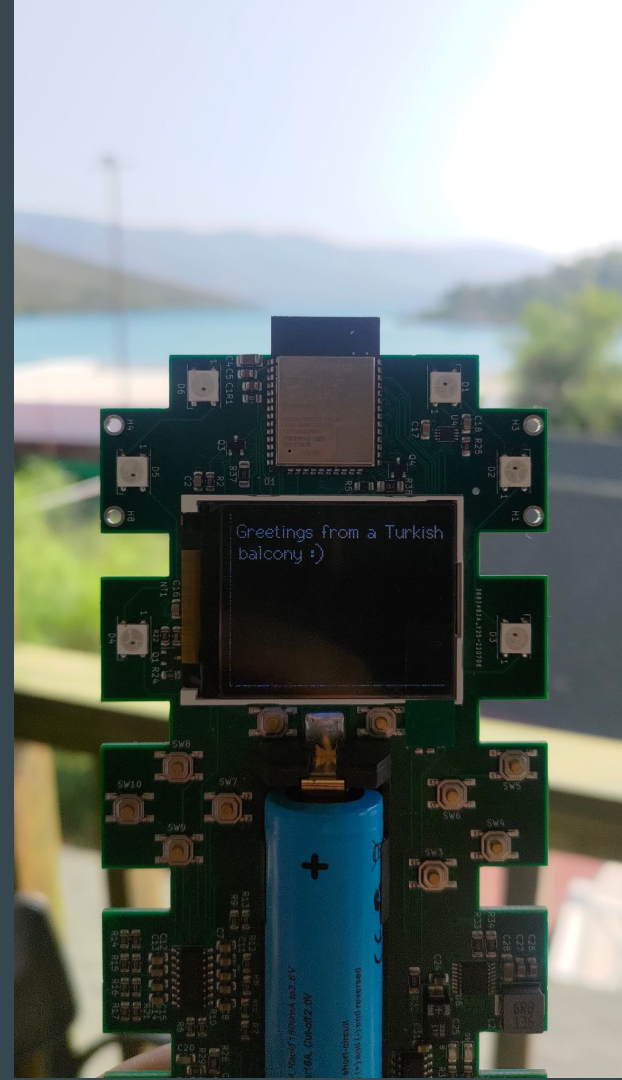
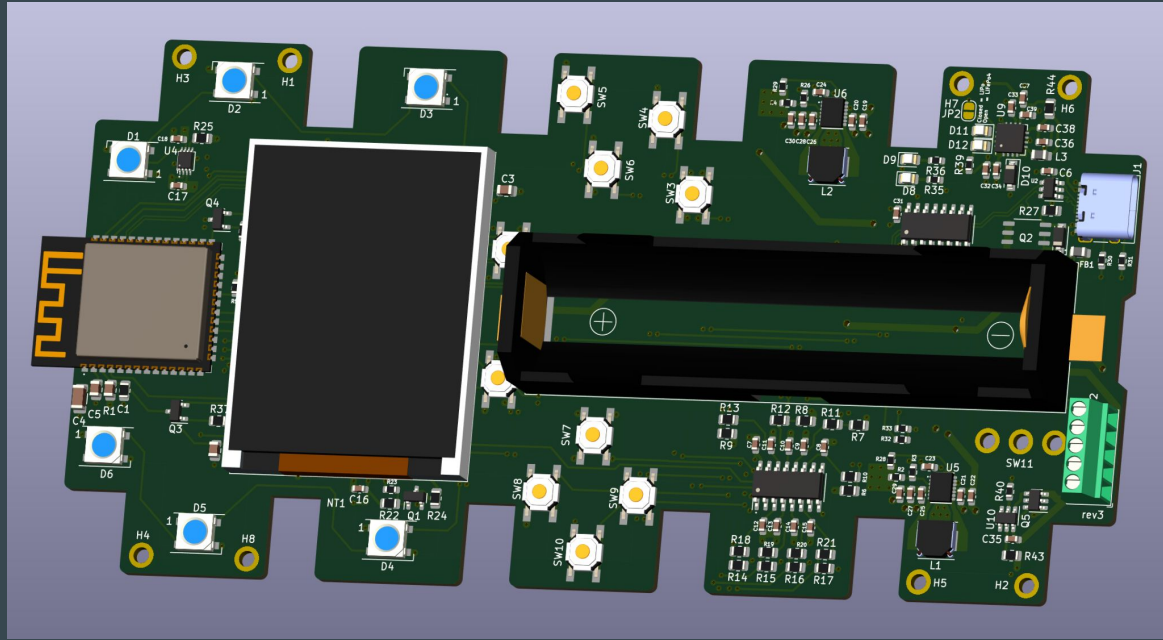


DTR	RTS	-> EN	I/O
1	1	1	1
0	0	1	1
1	0	0	1
0	1	1	0

Fixing the LED circuit



Ok, it's working.

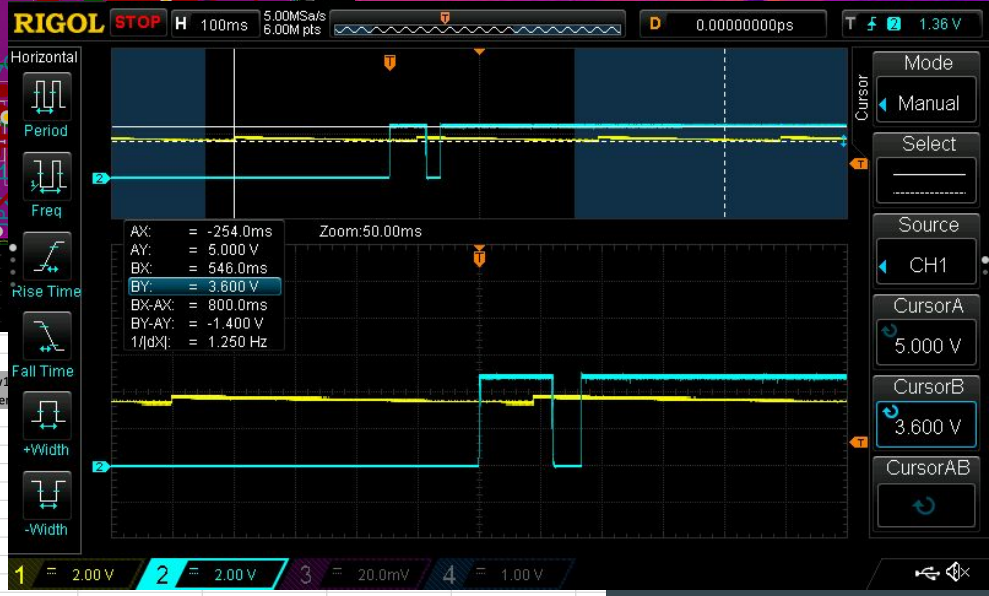
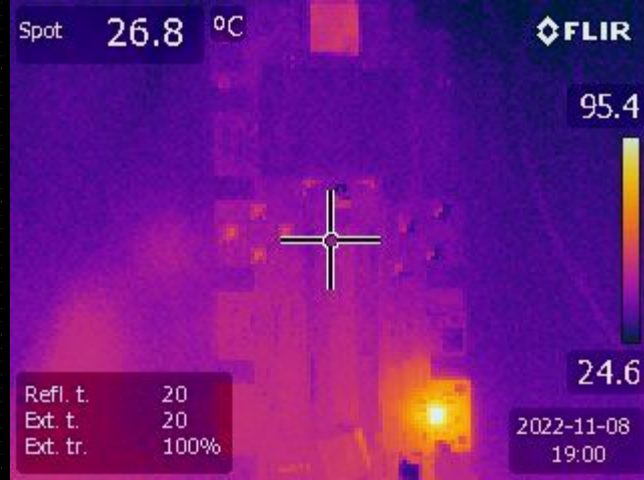
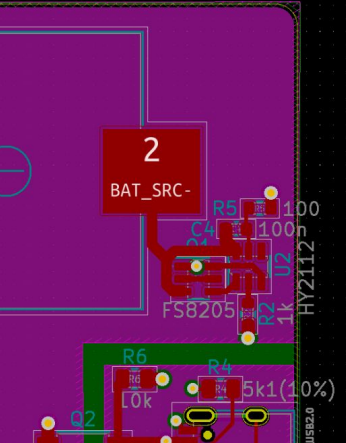
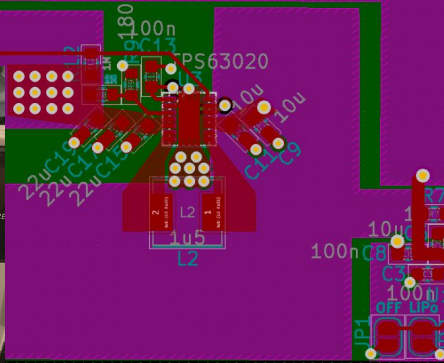


Except...

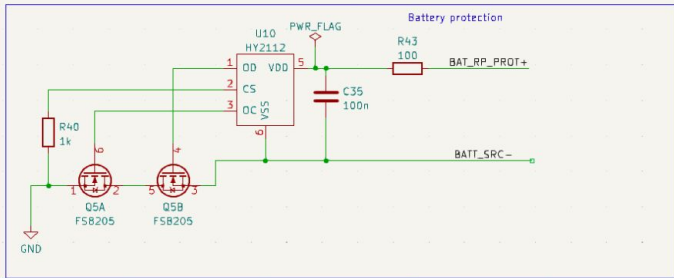
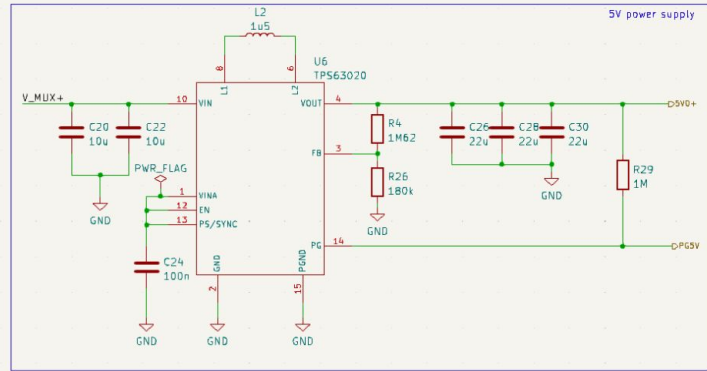
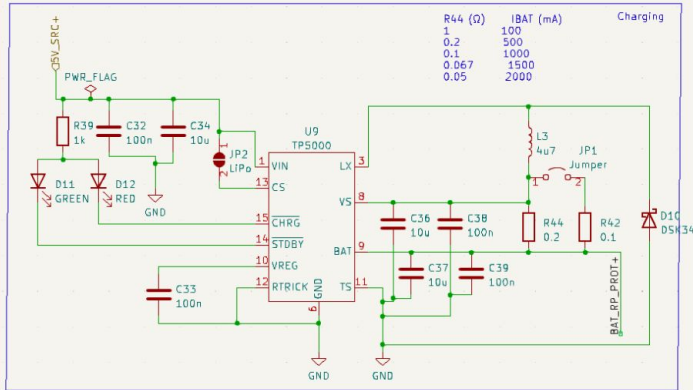
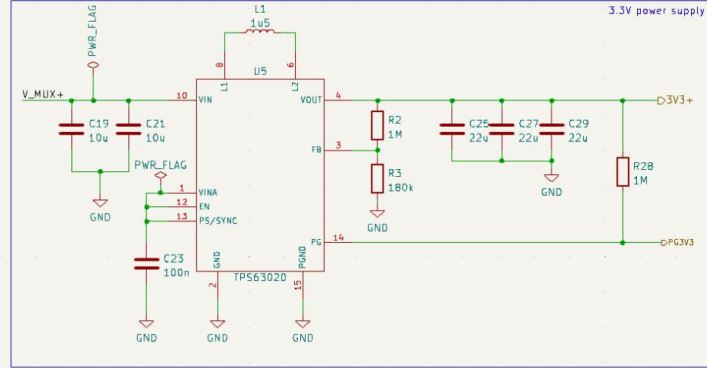
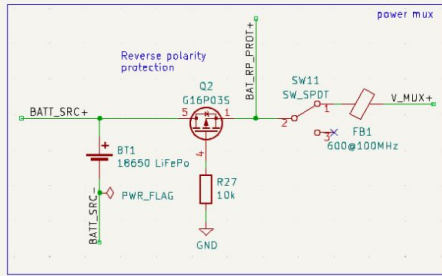
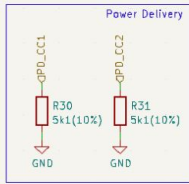
1
BAT_SRC+

2
BAT_SRC-

BT1
18650 LiFePo4



Voltage Measurements			
		USB, sw11 off	USB, sw2
		No battery present	No batter
V_USB		5.133V	5.133V
V_BAT (disconnected)			
V_BAT (in circuit)	or terminals if no bat	3.593V	3.593V
V_TPS000_BAT	pin 9, charge current	3.594V	5.593V
V_TPS000_VREG	pin 10	3.861V	3.861V
V_Q2_G	gate	5.133V	5.129V
V_Q2_S	source	5.164V	4.800V
V_Q2_D	drain	3.594V	3.593V
V_MUX+	after fb1	0.14V (falling)	4.789V



Florian Schuetz

BalcCon

Sheet: /Power Supply/

File: power_supply.kicad_sch

Title: BCD-0a26 - Power Supply

Size: A4

Date: 2021-08-20

Rev: 2

KiCad E.D.A. kicad (7.0.0-0)

Id: 7/8

Writing the Framework

The screenshot displays an IDE window with a disassembler view of C++ code. The main window shows the disassembly of a function, with the following code visible:

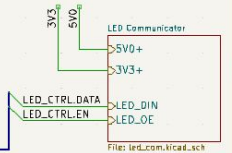
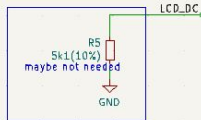
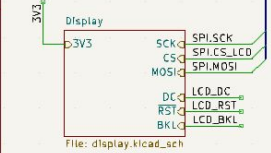
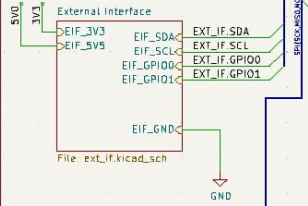
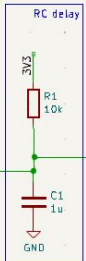
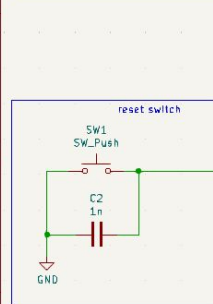
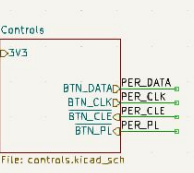
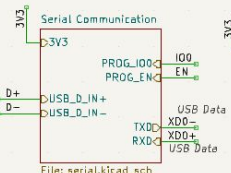
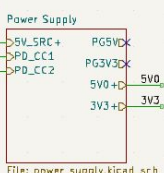
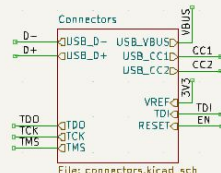
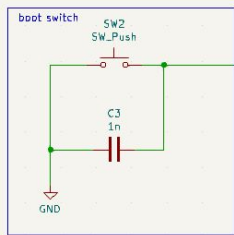
```
return r;
    ++x2;
    }
    x2&=
    ++y2;
    }
    r=destination.commit_batch_async();
    if(gfx_result::success!=r) {
        return r;
    }
    return gfx_result::success;
};
template<typename Destination,typename Source>
static gfx_result draw_bitmap_impl(Destination& destination, const rect16& dest_rect, Source& source, const r
using batch=helpers::batcher<Destination, Destination::caps::batch, Destination::caps::async>;
gfx_result r;
rect16 src=source_rect.normalize().crop(source.bounds());
rect16 dsr=dest_rect.crop((rect16)destination.bounds()).normalize();
if(!nullptr!=clip) {
    dsr=dsr.crop(*clip);
}
rect16 ddr = (rect16)dsr;
int o = (int)dest_rect.orientation();
const int W = dest_rect.width(), H=dest_rect.height();
// suspend if we can
helpers::suspender<Destination, Destination::caps::suspend, Destination::caps::async> stok(destination, async
```

The IDE interface includes a left sidebar with project files, a top toolbar, and a bottom status bar. The bottom status bar shows the current project and build status.

Make it your own...

Overview

- The badge is intended to be programmed. To make this easier there is a framework.
- Programming is easiest if you use Visual Studio Code and Platformio. Check the documentation how to set it up:
https://gitlab.com/fschuetz/bcd-0o27/-/tree/main/documentation?ref_type=heads
- There is a programming framework that provides drivers and abstraction layers that make it easier to program the badge in c++. Its built on top of espidf.
- The operating system used by espidf - and thus the cyberdeck - is FreeRTOS



Designed by Florian
BalcCon
 Sheet: 7
 File: BCD-0a26.kicad_sch
Title: BCD-0a26
 Size: A4 Date: 2021-08-07 Rev: 0
 KiCad E.D.A. kicad (7.0.0-0) Id: 1/8

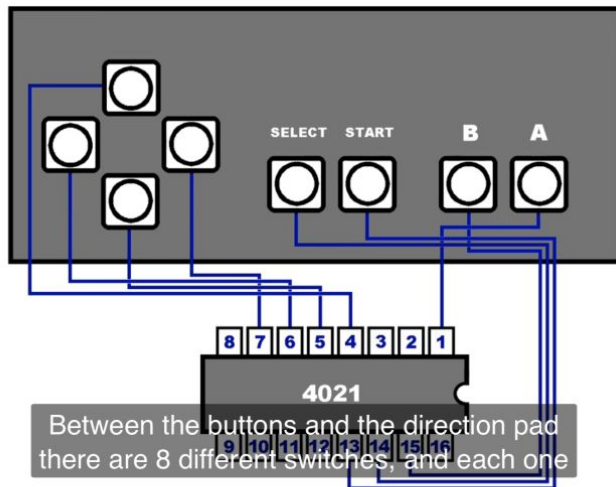
Inputs

Inputs



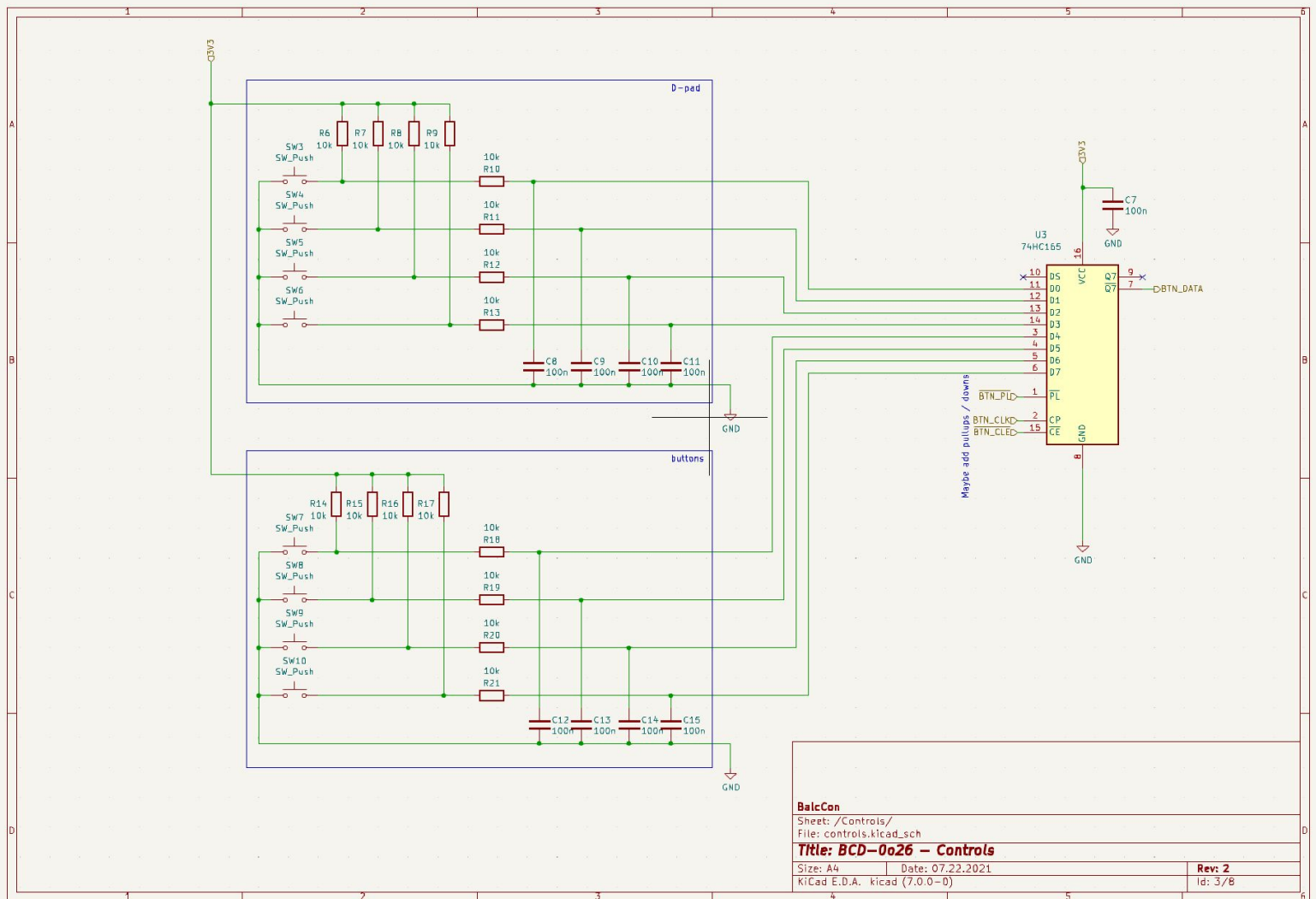
Grabmoix 21.07.2021 14:57

I think for reading the buttons we will do similar as NES and SNES as sort of a tribute.



Between the buttons and the direction pad there are 8 different switches, and each one

(shift register).



BalcCon

Sheet: /Controls/
 File: controls.kicad_sch

Title: BCD-0a26 - Controls

Size: A4 Date: 07.22.2021
 KiCad E.D.A. kicad (7.0.0-0)

Rev: 2
 Id: 3/8

Reading inputs

`espconsole::consoleController& console = espconsole::consoleController::getInstance();` - Get controller instance

`controller.capture()` - Read button states, overwrite previous state

`controller.sample()` - Read button states, or it with previous state

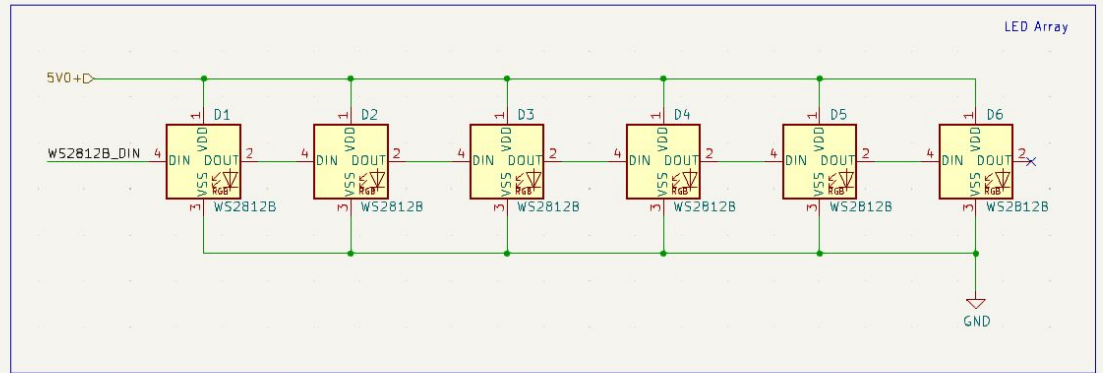
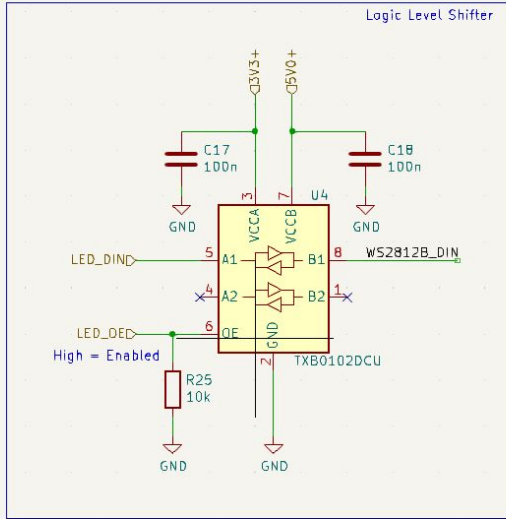
`controller.clear()` - Clear out all button presses

`controller.getButtonState(BUTTON_A)` - Get state of buttons (`_LEFT`, `_RIGHT`, `_UP`, `_DOWN`, `_A`, `_B`, `_X`, `_Y`)

```
1843 //controller.capture();
1844 controller.clear();
1845 controller.sample(8, 50);
1846 if(controller.getButtonState(BUTTON_A) || controller.getButtonState(BUTTON_B)) {
1847     // Depending on selection execute
1848     switch(selected_entry) {
1849         case 1:
1850             run_game(lcd, controller, screenbuffer, screenbuffer_buf);
1851
1852             // Draw titlescreen and menu (we know 1 is selected)
1853             draw::bitmap(*lcd, screen_bounds, titlescreen_bmp, screen_bounds);
```

LED

Circuit



Programming LEDs

- LEDs flash concurrent to everything else. They can be steered individually or patterns can be scheduled.
- LEDPatternGenerator generator; - Create a new generator.
- generator.setRepetitions(1); - Set number of repetitions.
- generator.addState(s); - Add state to pattern.
- ledPattern lp; generator.generate(&lp);
- generator.reset(); - reset the generator.
- led_err_t error = led.patternSchedule(lp);

```
41 // Generate first disco pattern
42 LEDPatternGenerator generator;
43
44 // First, we start with a very simple pattern of just blue, then
45 // green, then red. End state is to switch all leds off.
46 generator.setInterruptable(false);
47 generator.setRepetitions(2);
48
49 ledStates pstates;
50 for(int i = 0; i < LED_IF_NUM_LED; i++) {
51     pstates.led[i] = {
52         .red = 0x00,
53         .green = 0x00,
54         .blue = 0x00
55     };
56 }
57 generator.addState(pstates, pdMS_TO_TICKS(500));
```


SAO++

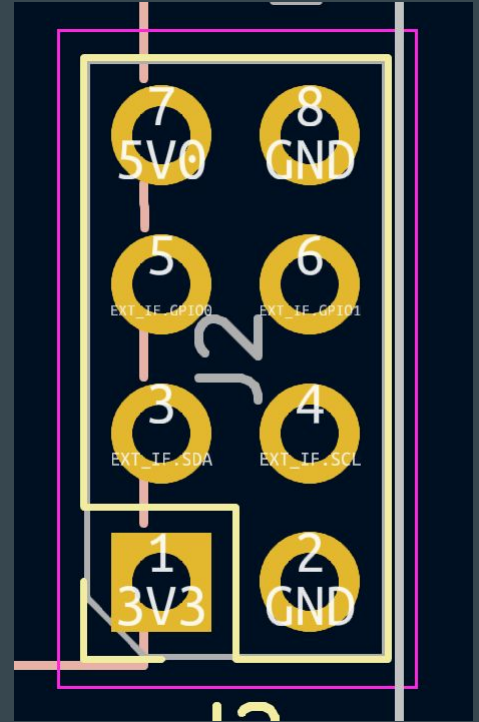
SAO++

```
template<typename Destination>
int saoBlink(void *param) {

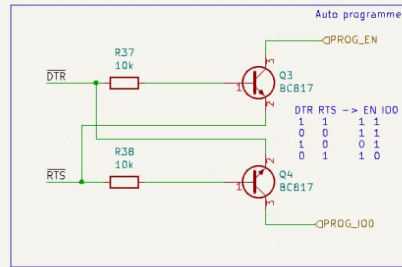
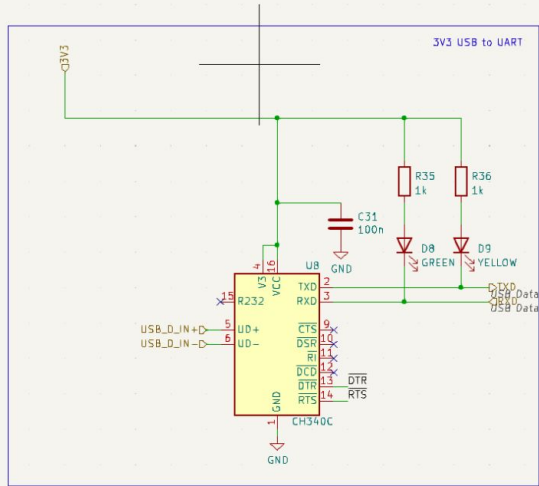
    // Configure the GPIOs
    gpio_config_t io_conf = {};
    io_conf.intr_type = GPIO_INTR_DISABLE;
    io_conf.mode = GPIO_MODE_OUTPUT;
    io_conf.pin_bit_mask = ((1ULL<<CONFIG_SAO_PIN0)
                             | (1ULL<<CONFIG_SAO_PIN1)
                             | (1ULL<<CONFIG_SAO_SCL)
                             | (1ULL<<CONFIG_SAO_SDA));
    io_conf.pull_down_en = GPIO_PULLDOWN_DISABLE;
    io_conf.pull_up_en = GPIO_PULLUP_DISABLE;

    esp_err_t error = gpio_config(&io_conf);
    if(error != ESP_OK) {
        return -1;
    }

    gpio_set_level(SAO_PIN0, 0);
    gpio_set_level(SAO_PIN1, 0);
    gpio_set_level(SAO_SDA, 0);
    gpio_set_level(SAO_SCL, 0);
}
```



Console



Florian Schuetz

Sheet: /Serial Communication/

File: serial.kicad_sch

Title: Serial Communication

Size: A4 Date: 2021-09-12

Rev: 0

KiCad E.D.A. kicad (7.0.0-0)

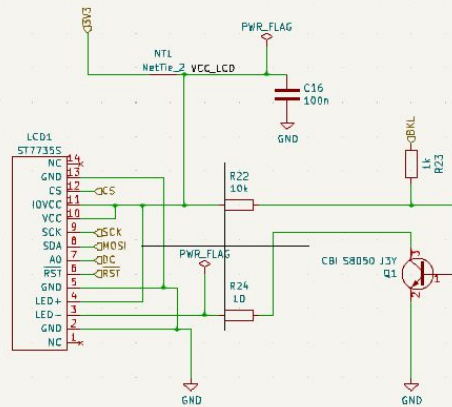
Id: 8/8

Console / Serial Connection

- Has a bug that if not connected during bootup will not accept input.
- Supports forwarding input to other threads.
- `console.registerCommand(...)` - registers a new command with the console
- `consoleController &c = consoleController::getInstance();` - It is a singleton.
- `console_err_t cerr = console.requestInputRedirect(qData, qSignal, &hForwarder, true);` - Register a forwarder.
- `console.stopCommandExecution(&hForwarder);` - Stop default console behaviour.
- `console.enableCommandExecution(&hForwarder);` - Enable console behaviour.

```
// <--- Register console commands below -->
console.registerCommand("lsap", &lsap, "List available access points.");
console.registerCommand("wificfg", &wificfg, "Configure wifi.");
console.registerCommand("apinfo", &apinfo,
    "Show information about connected access point.");
```

Display



BalcCon

Sheet: /Display/
File: display.kicad_sch

Title: BCD-0a26 - Display

Size: A4 Date: 2021-10-07
KiCad E.D.A. Kicad (7.0.0-B)

Rev: 2
Id: 4/8

Display

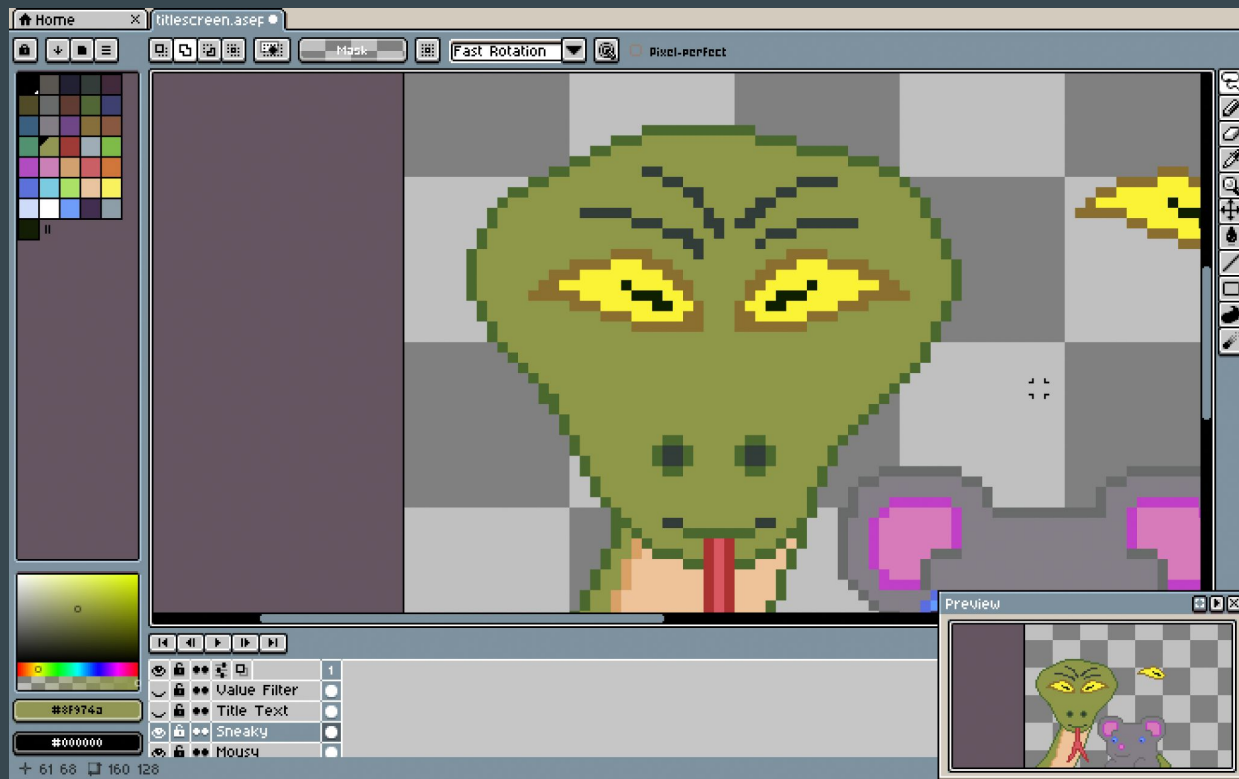
- Uses htcw gfx: <https://honeythecodewitch.com/gfx/wiki/index.md>
- Custom driver, based on the driver of Bodmer: https://github.com/Bodmer/TFT_eSPI/tree/master/TFT_Drivers
- Library to build menus:
 - MenuController<lcd_type, lcd_type::pixel_type> mc; - Controller object to build menus
 - mc.cursor->addEntry(mc.createSubmenu("Games")); - Add an entry (submenu in this case)
 - mc.createActionItem("Sneaky & Mousy", bcd_mod_snake::module_main<lcd_type>, &lcd) - Action
 - mc.cursor->enter(); / mc.cursor->leave(); - Enter / leave a submenu
 - mc.cursor->drawMenu(*bmp, bounds, lcd_color::alice_blue, lcd_color::dark_goldenrod); - Draw it.
 - int return_code = ((ActionItem<bmp_type, bmp_type::pixel_type> *)e)->execute(); - Execute

194
195
196
197
198
199
200

```
// Settings Submenu
mc.cursor->addEntry(mc.createSubmenu("Settings"));
mc.cursor->enter();
mc.cursor->addEntry(mc.createActionItem("Connect WiFi", selectWifi<lcd_type>, &lcd));
mc.cursor->leave();
```

> console Ad du ? 01 42 | v = x

Pixel Art / Sprites

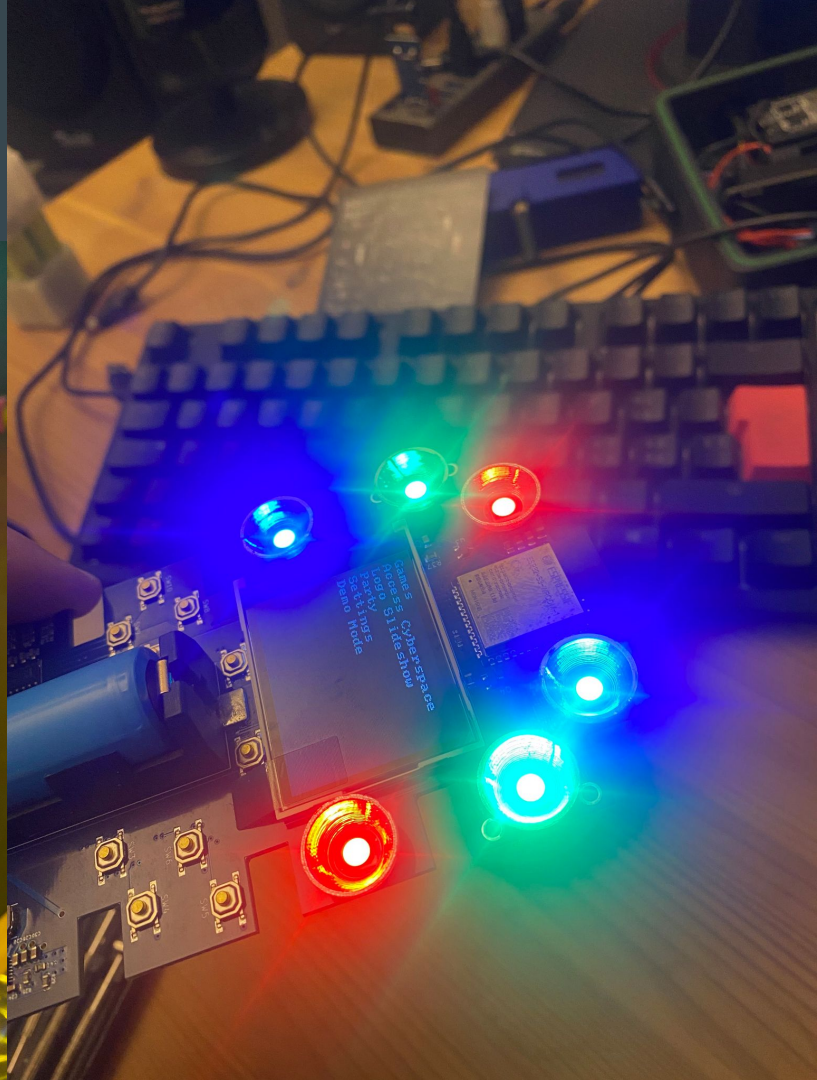


More Information

<https://hackaday.io/project/192371-balcon-cyberdeck-0o27-aka-bcd-0o27>

Note: There will be updates on documentation and the framework in the next hours.

Cases



Challenge I

Exploration challenge:

- Acid hit some hints in the badge that will allow you to find out important information that should be shared with Direct||.
- A little hint for the audience: The formula for the score in the Sneaky & Mousy game is as follows: `score += score_base + score_multiplier`
→ First one to bring the information will receive a black badge.

Challenge II

Development challenge:

- Fix the bug in the console driver → Earns you a beer in any case
 - Rewrite the LED driver to support the new rmt driver of espidf
 - Port Doom on the badge
 - Clone any classic game or make your own
 - Do something else that is really cool
- Present your project at the closing ceremony and the one getting the most applause will be rewarded a black badge.

Black Badge

Most important, the black badge earns you fame ;-).

The black badge will also earn you five free beverages of your choice at each upcoming BalCCon.

Maybe there will be even more goodies in the future - like special events or so. Who knows....

Questions?

