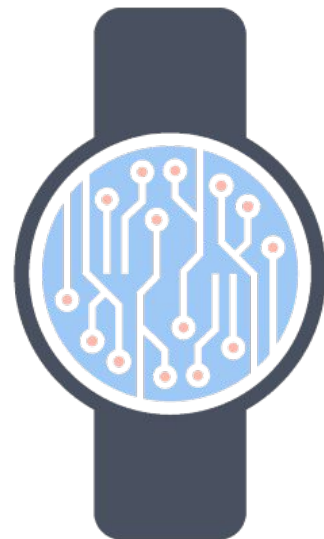


# ZSWatch

The Open-Source Zephyr™-based Smartwatch

FOSDEM - Brussels 2025

Daniel Kampert  
[github.com/kampi](https://github.com/kampi)



# Agenda

- 1 Who am I?
- 2 What is “ZSWatch”?
- 3 Let's talk about the hardware
- 4 The software in a nutshell
- 5 How to write a simple app
- 6 Many ideas for the future
- 7 Support us!
- 8 (Optional) Short Demo
- 9 QA

# Who am I?

- Daniel Kampert - You can call me Kampi
- 35 years old
- From Hagenbüchach (near Nuremberg)
- Passionate electronic engineer
- Currently working as an Electronics Engineer at ETO Sensoric
- Joined the project in August 2023
- Never worked with Zephyr before

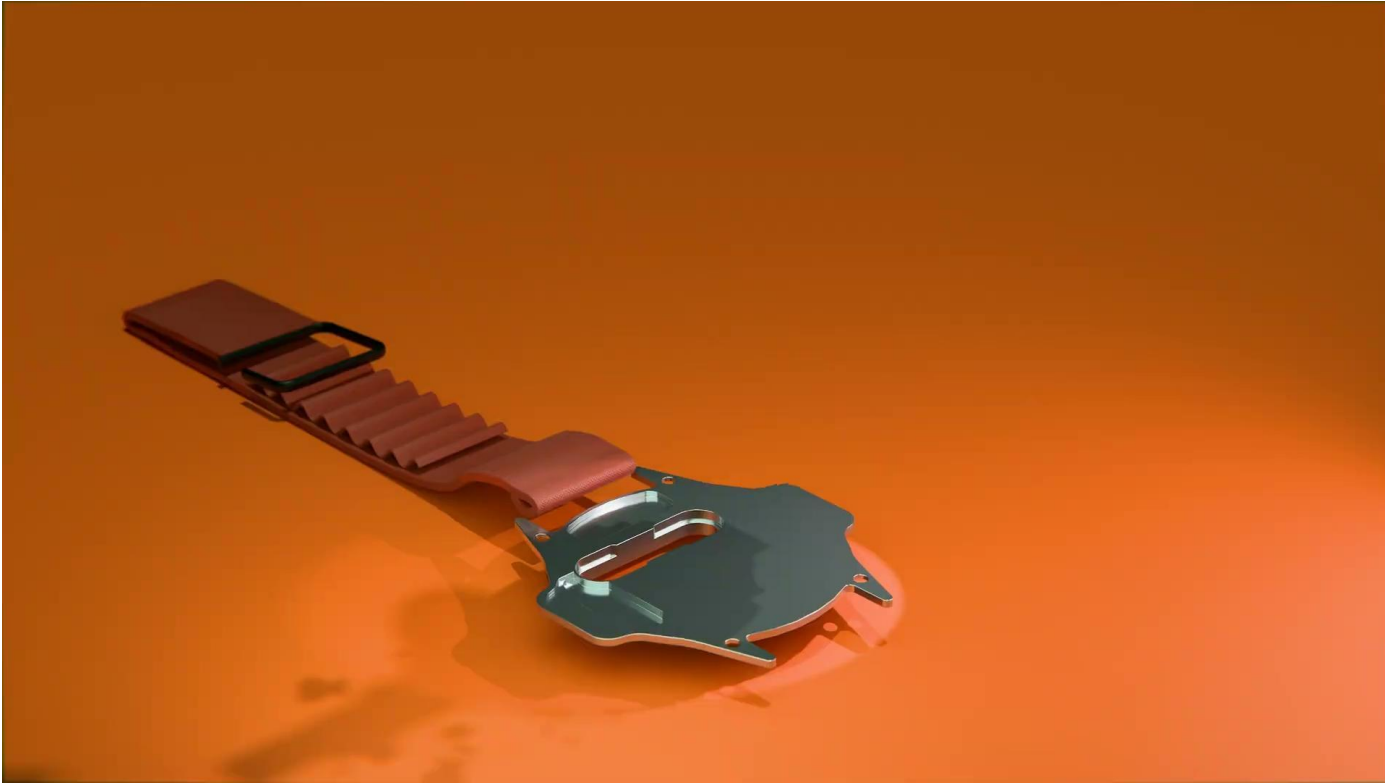


# What is “ZSWatch”?

- Zephyr based Smartwatch
- Started by Jakob Krantz
- All parts are Open-Source
- Everything belongs to you
- You can modify and repair every part
- Everything needed is on-board
- State-of-the-art smartwatch
- Nice looking design



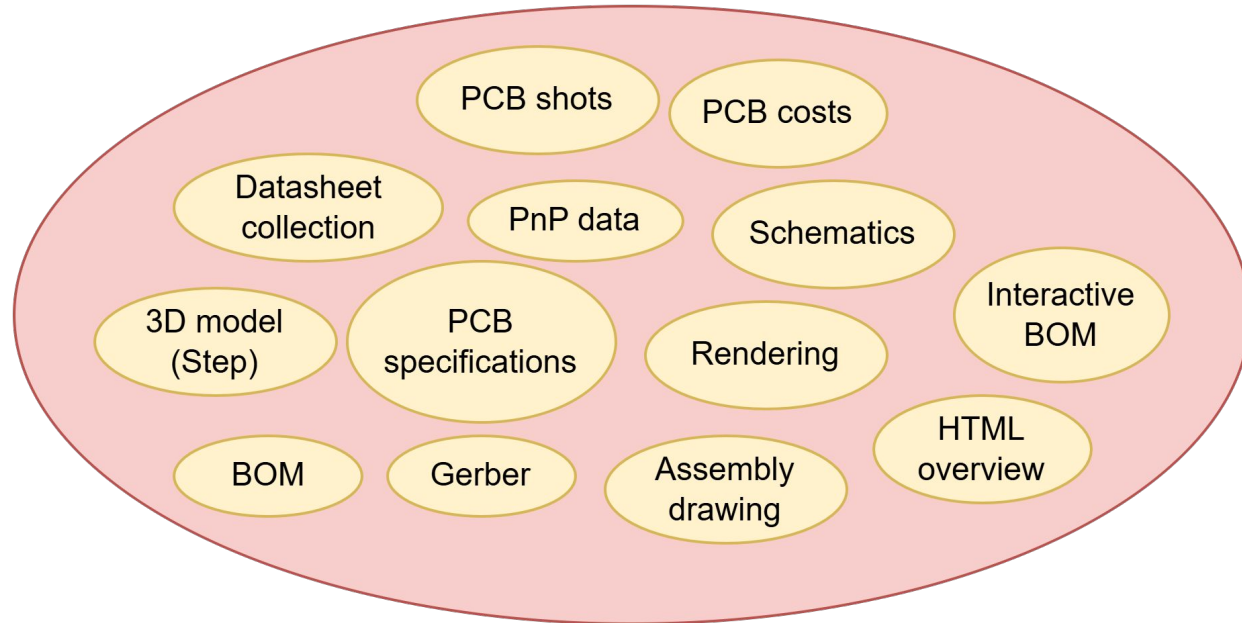
# What is “ZSWatch”?



© [Samer Aldaheer](#)

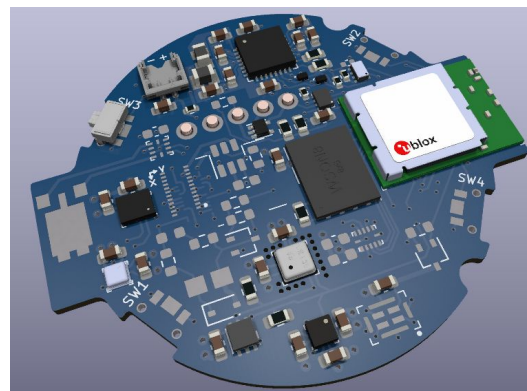
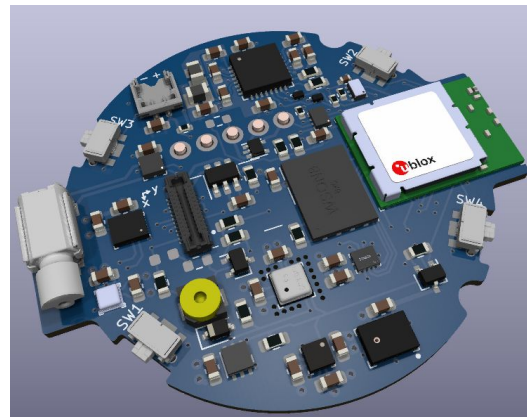
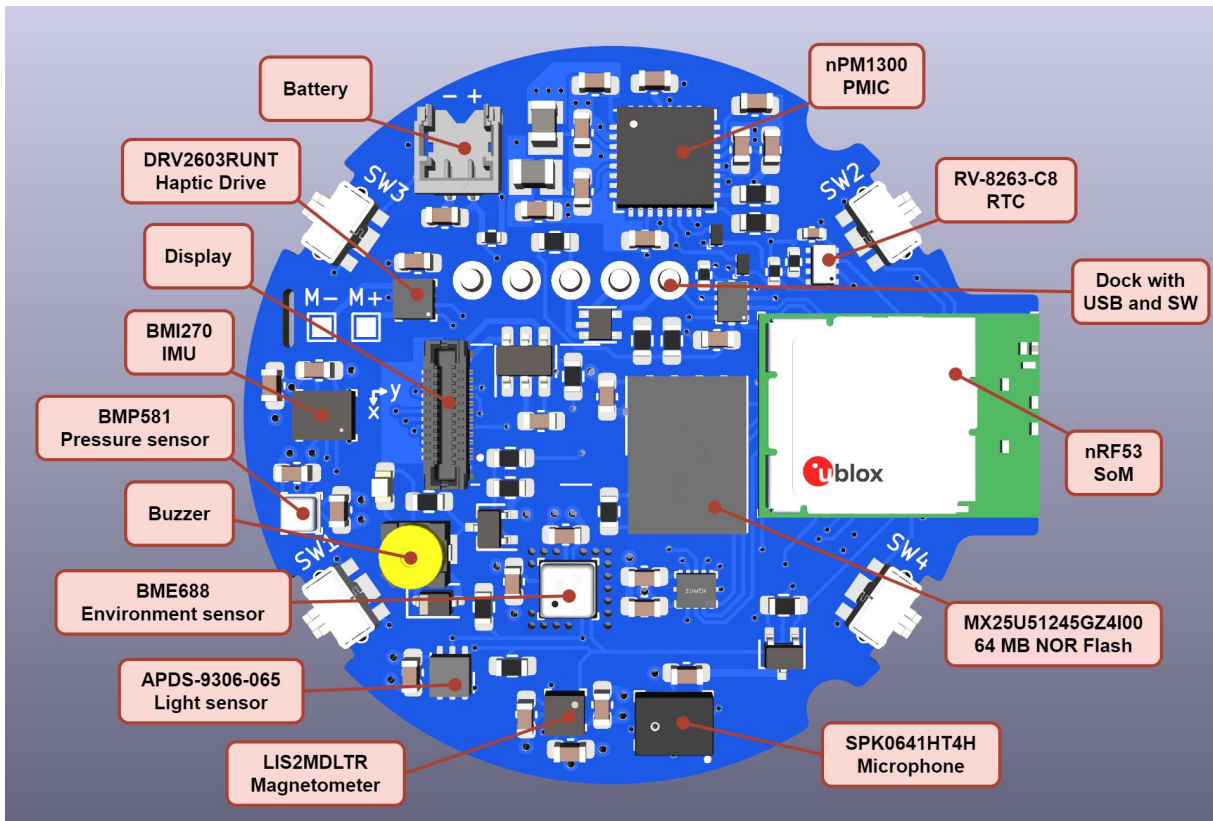
# Let's talk about the hardware

- Electronics Design is done with KiCad
- KiBot is used for CI/CD based production data generation
- Reduced PCB can be used as ZigBee End Device for Home Automation
- Mostly single sided assembly

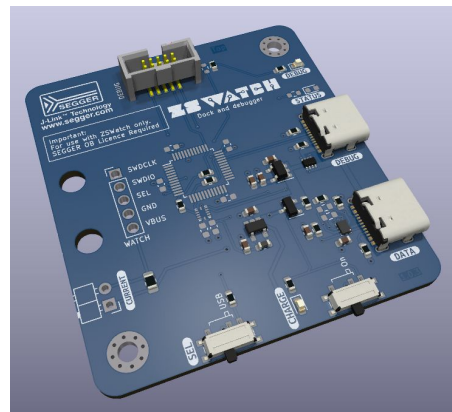
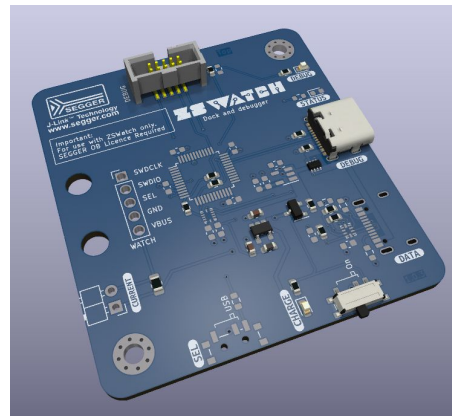
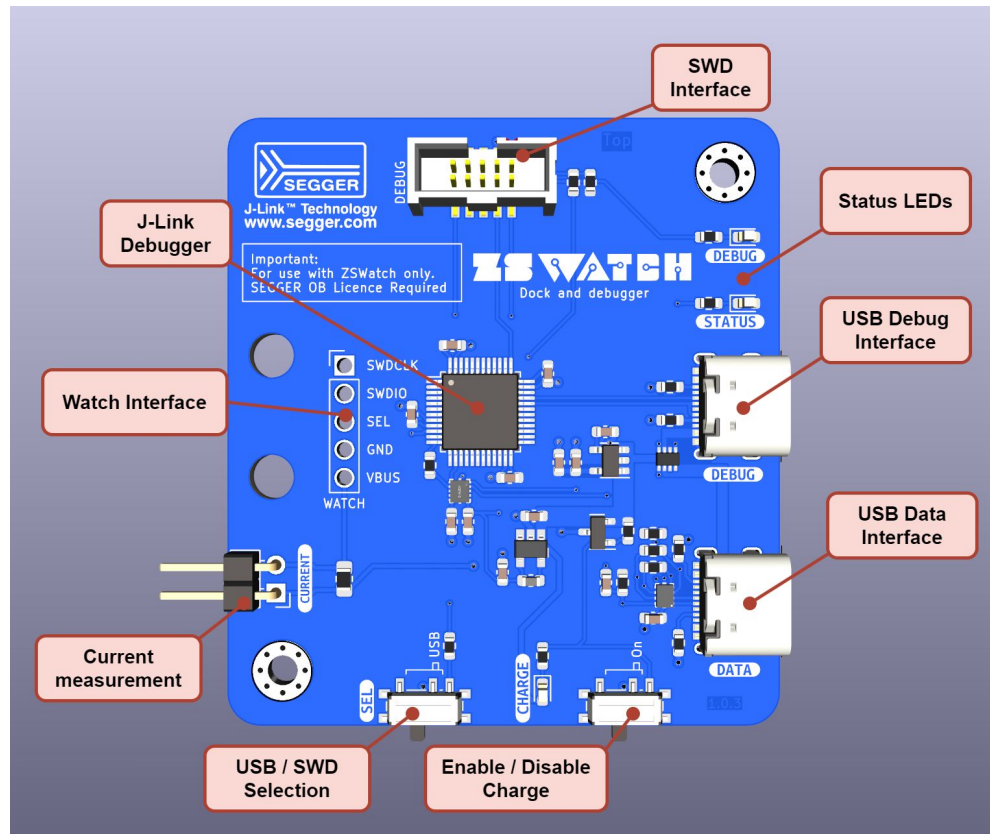




# Let's talk about the hardware

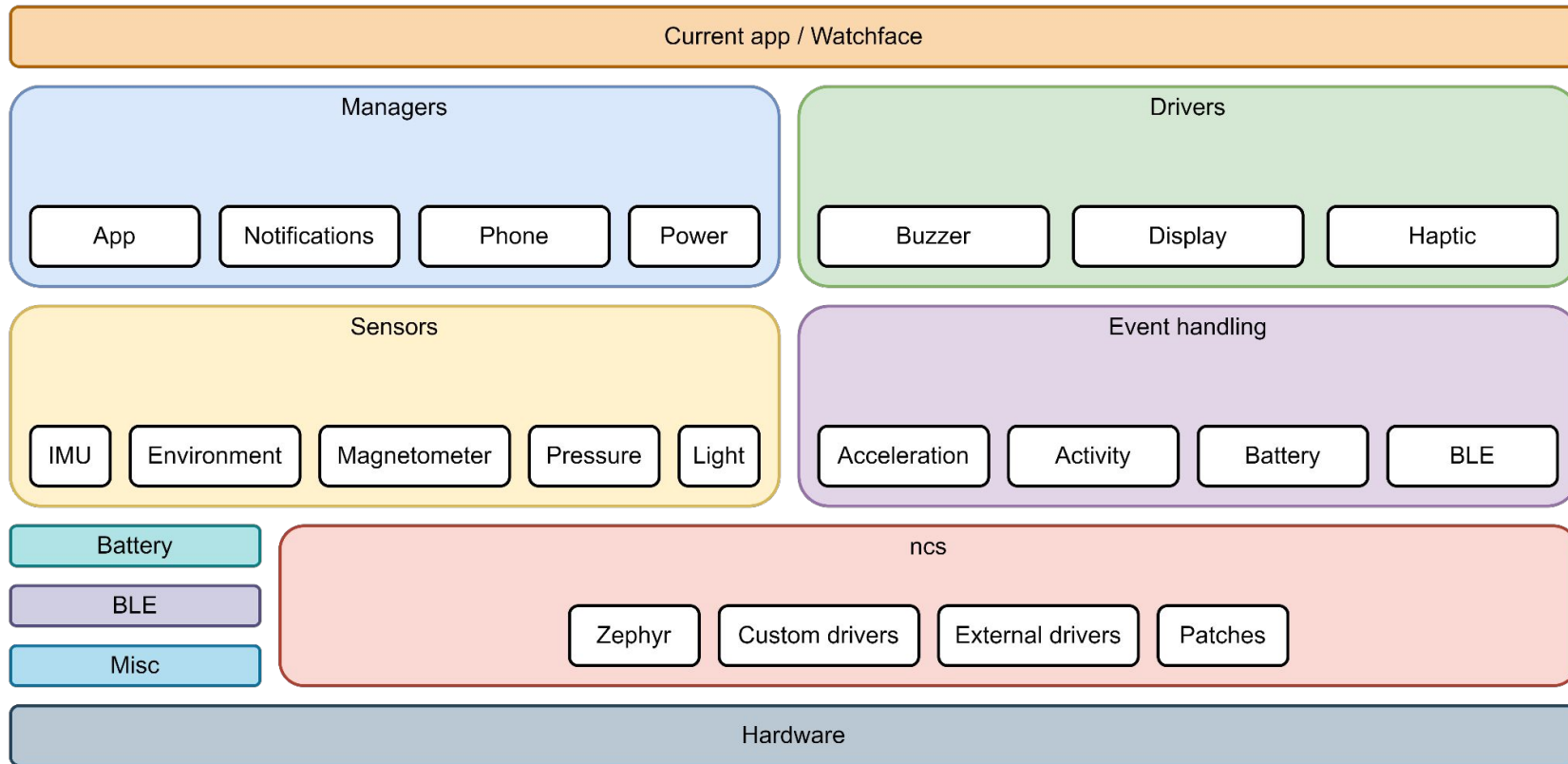


# Let's talk about the hardware



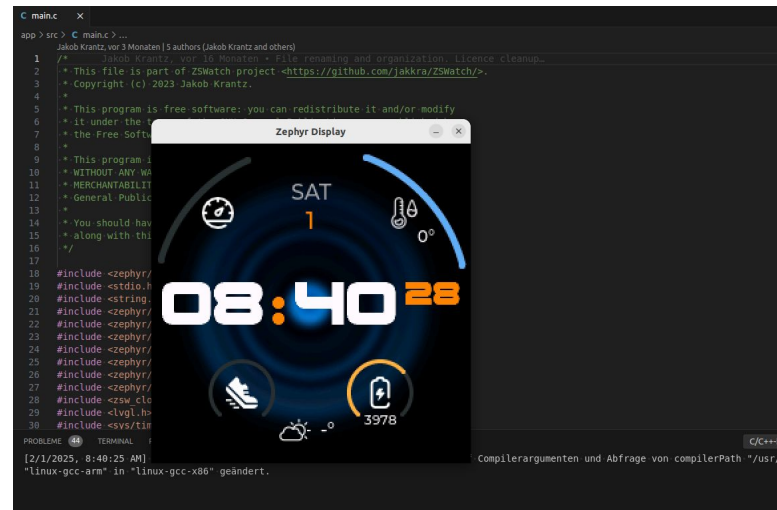


# The software in a nutshell



# The software in a nutshell

- One software for all hardware revisions thanks to device tree overlays
- The software is built with CI/CD
- Core dump app in debug builds to track crashes during the testing
- Different watch faces based on esp-lvgl-watchfaces from Felix Biego
- POSIX build to develop and test new UIs on a desktop PC



```
C main.c x
app > src > C main.c > ...
Jakob Krantz, vor 3 Monaten | 5 authors (Jakob Krantz and others)
1 /*      Jakob Krantz, vor 16 Monaten · File renaming and organization, Licence cleanup.
2  * This file is part of ZSWatch project <https://github.com/jakkr/ZSWatch>.
3  * Copyright (c) 2023 Jakob Krantz.
4  *
5  * This program is free software: you can redistribute it and/or modify
6  * it under the terms of the GNU General Public License as published by
7  * the Free Software Foundation, either version 3 of the License, or
8  * (at your option) any later version.
9  * This program is distributed in the hope that it will be useful,
10 * WITHOUT ANY WARRANTY; without even the implied warranty of
11 * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
12 * GNU General Public License for more details.
13 *
14 * You should have received a copy of the GNU General Public License
15 * along with this program. If not, see <http://www.gnu.org/licenses/>.
16 */
17
18 #include <zephyr/
19 #include <stdio.h
20 #include <string.
21 #include <zephyr/
22 #include <zephyr/
23 #include <zephyr/
24 #include <zephyr/
25 #include <zephyr/
26 #include <zephyr/
27 #include <zephyr/
28 #include <zsw_cle
29 #include <lvgl.h>
30 #include <sys/tim
```

The screenshot shows a terminal window with a C++ source file for a smartwatch display. The code includes headers for Zephyr, stdio, string, and lvgl. A preview window titled "Zephyr Display" shows a watch face with a dark blue background, a large digital clock displaying "08:40:28", and various icons for battery, signal, and weather. The battery icon shows 3978. The watch face also displays "SAT 1" and "0°".

# How to write a simple app

```
static application_t app = {
    .name = "Battery",
    .icon = ZSW_LV_IMG_USE(battery),
    .start_func = battery_app_start,
    .stop_func = battery_app_stop
};

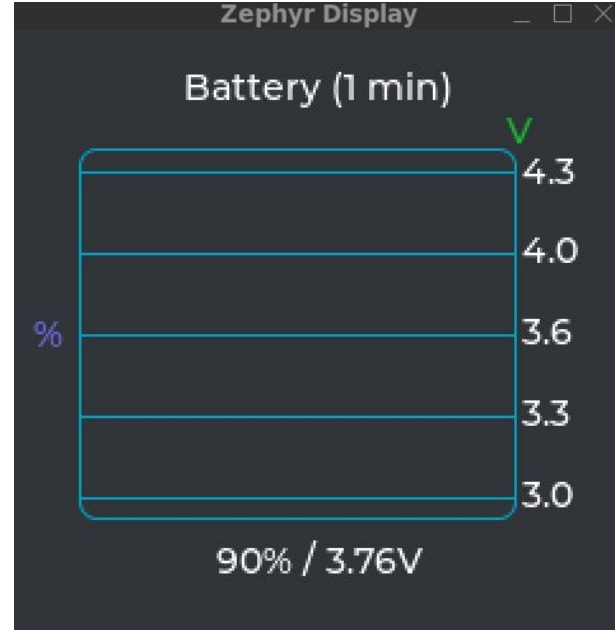
static void battery_app_start(lv_obj_t *root, lv_group_t *group) {
    battery_ui_show(root, on_close_battery);
}

static void battery_app_stop(void) {
}

static void on_close_battery(void) {
}

static int battery_app_add(void) {
    zsw_app_manager_add_application(&app);
    return 0;
}

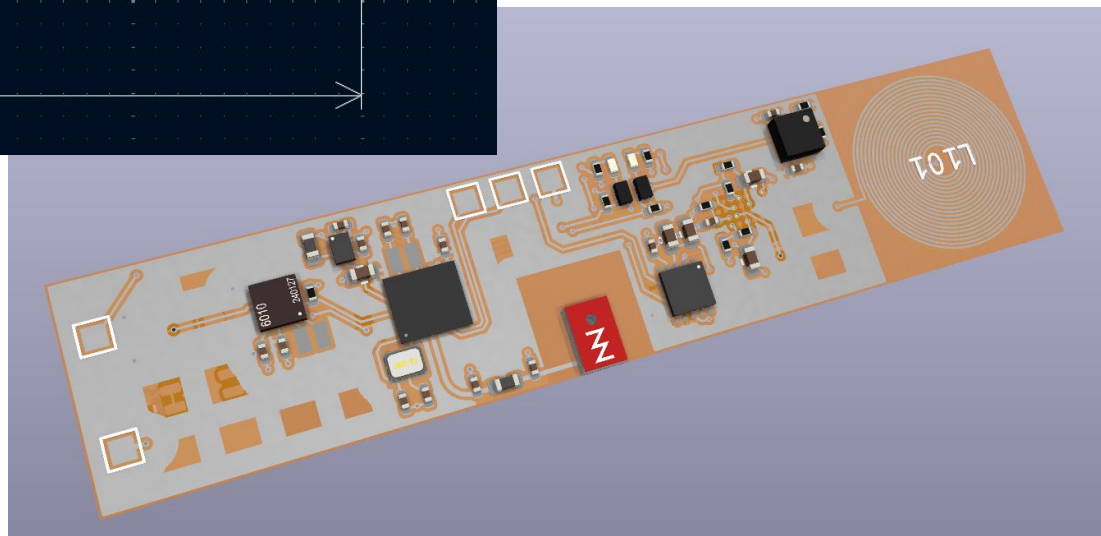
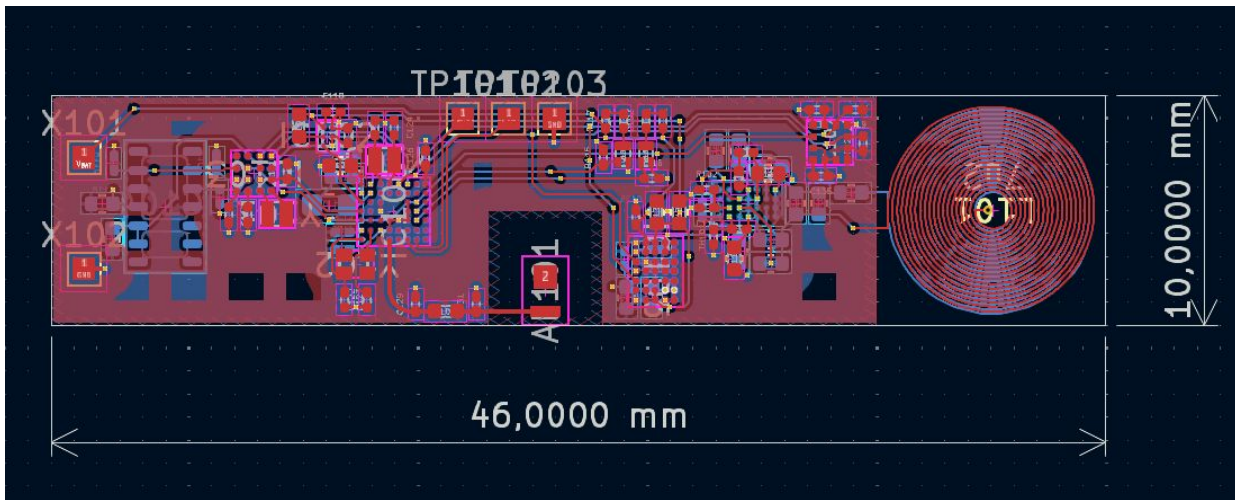
SYS_INIT(battery_app_add, APPLICATION, CONFIG_APPLICATION_INIT_PRIORITY);
```



# Many ideas for the future

- Running Doom
- Heart Rate sensor and fitness app
- LE audio with AI integration
- Optimized smartphone companion app for the watch
- New housing
- Switch to nRF54
- Finding someone who helps us to sell the watch
- Clean up code, update documentation, more templates, etc.
- etc.

# Idea: ZSRing



# Project support

- Since December 2024 the project is officially supported by the 2024 NGI0 Commons Fund
- The project is also supported by SEGGER who support us with OB licenses for our programmer
- Everyone can support on GitHub
- Always looking for additional support



**The Embedded Experts**



# Support us!



# Short Demo



# Thanks for your attention!



**STAY IN TOUCH**