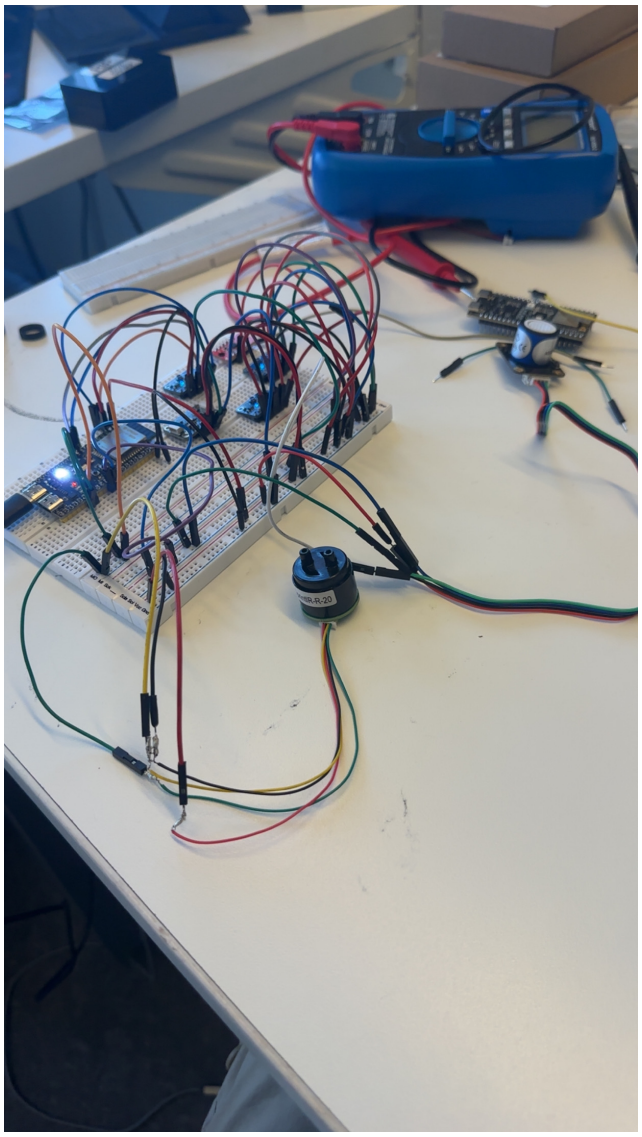
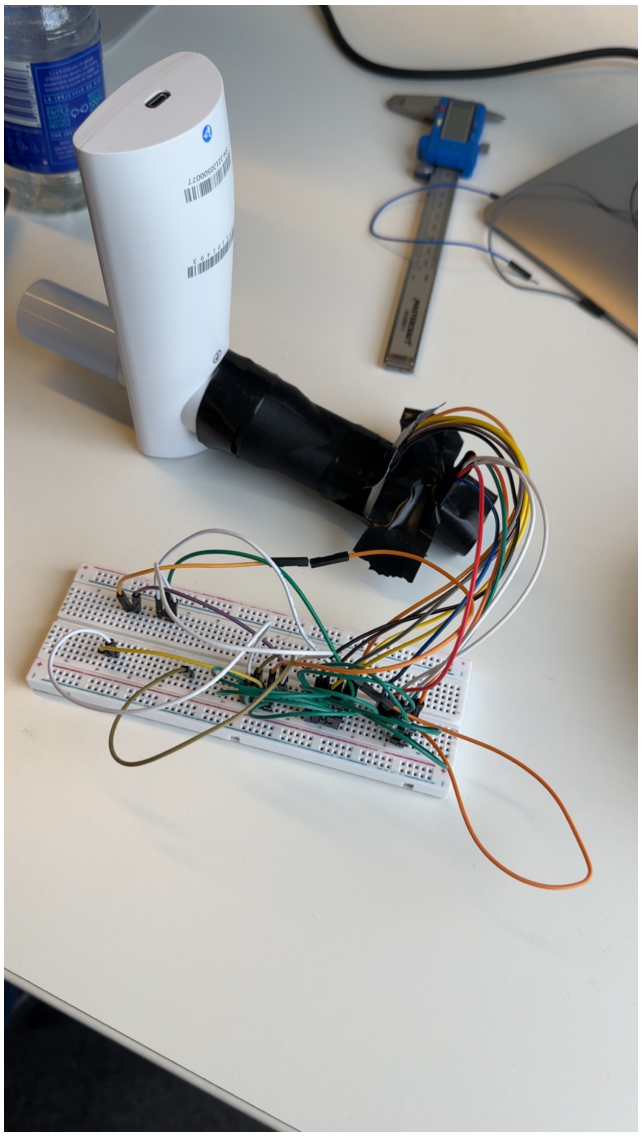
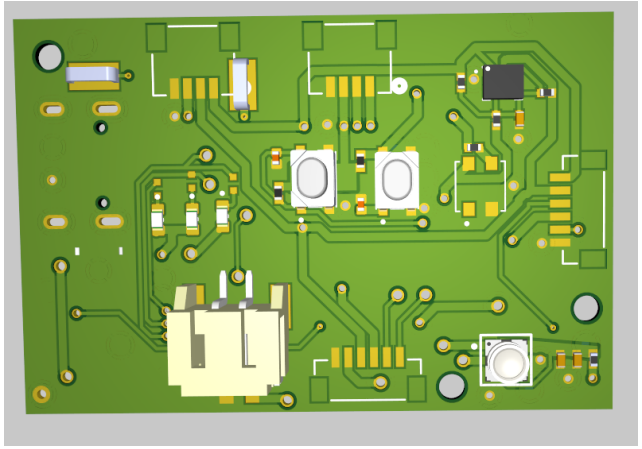
ESP32 S3 Wearable Health and Respiratory Device



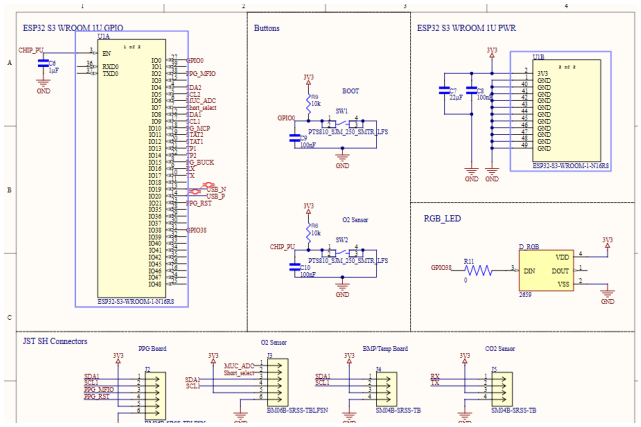
(a) Breadboard prototype



(b) Sensor and flow setup



(a) Custom Integrated ESP32-S3, Sensors, Power PCB



(b) ESP32-S3 Schematic

Overview

Developing a next generation wearable that monitors cardiovascular, respiratory, and metabolic health. The device combines custom low noise PCBs around an ESP32 S3 with biomedical and environmental sensors to capture high resolution data in real time.

Hardware Design

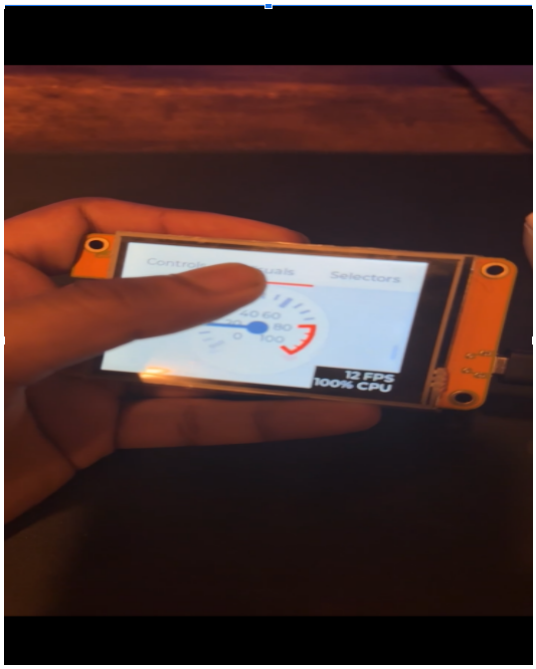
- Assembled multilayer PCBs integrating BMP585 pressure and temperature sensors, Max30102 optical pulse oximetry, SprintIR CO_2, Gravity O_2, and a compact IMU.
- Optimized analog layout and decoupling to reduce noise and improve wireless signal integrity while cutting board weight by twenty percent.
- Implemented USB C power management with ESD protection and verified production readiness with DFM checks.

Firmware and Data Pipeline

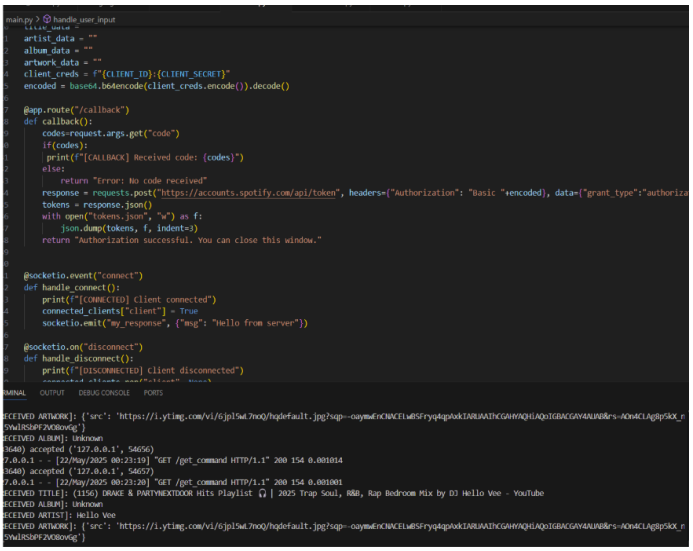
- Wrote multi sensor firmware in FreeRTOS with ESP IDF to poll I^2C and UART devices and stream synchronized data to a mobile backend through BLE with latency under fifty milliseconds.
- Built power efficient dual core task scheduling on the ESP32 S3 for continuous acquisition of high rate biomedical signals without packet loss.
- Calibrated cardiac, respiratory, and gas sensors against lab instruments to keep measurements within plus or minus two percent of reference.

ESP32 Media Controller Dashboard

GitHub



(a) Media Dashboard



(b) Backend Python Server

Overview

Handheld touchscreen device that displays and controls YouTube, Spotify, and Discord media without touching the computer.

Hardware and UI

ESP32 drives a 2.8 inch SPI TFT display and capacitive touch panel, with LVGL layouts for play, pause, skip, and volume.

Browser Extension

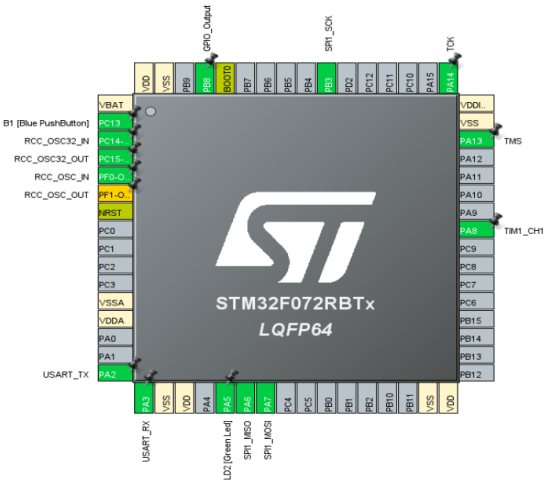
Chrome extension calls the YouTube metadata API to pull video title, playback state, and timestamp, then packages JSON and sends it with WebSockets.

Communication

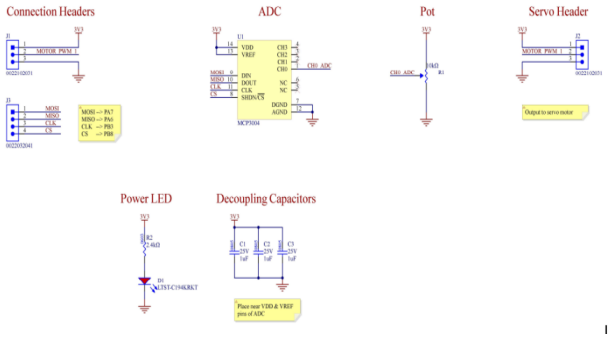
- **Incoming**: Flask WebSocket server relays JSON from the browser to the device.
- **Outgoing**: Device sends button press commands back to the server over UART to trigger play, pause, or skip.

Motor Tester

GitHub



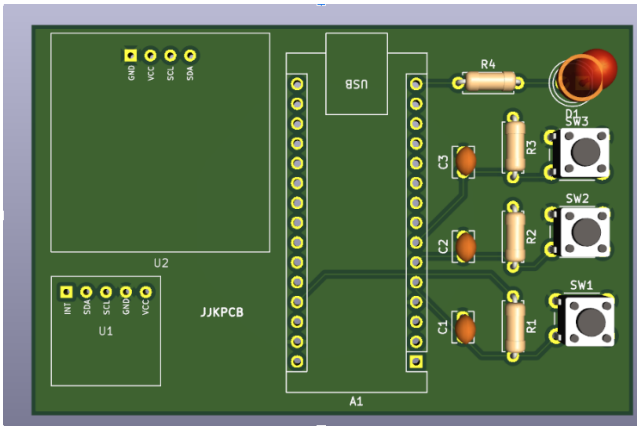
(a) STM32 pin mapping



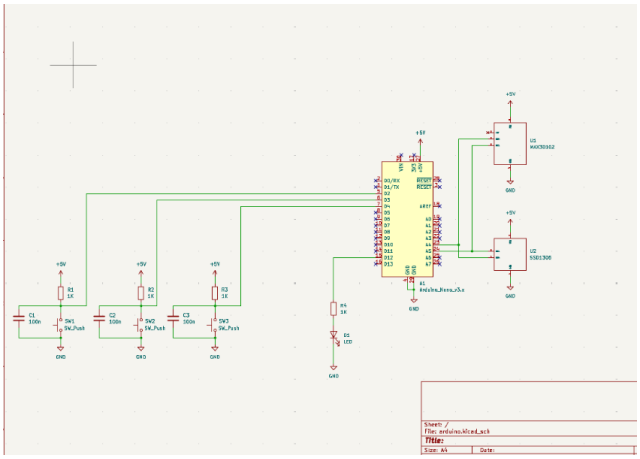
(b) ADC and servo schematic

- Automated servo and continuous rotation motor tests by reading potentiometer positions through an SPI connected ADC and generating PWM control signals.
- Wrote SPI drivers to sample external ADC voltages and tuned timer based PWM to convert readings into speed and direction.

Heartbeat Monitor PCB Design



(a) PCB render



(b) Circuit schematic

- Designed a custom PCB in KiCad with the MAX30102 for heartbeat and oxygen saturation.
- Integrated I^2C with an Arduino for live BPM and SpO_2 on an OLED display.
- Optimized placement and decoupling to improve signal quality and reduce noise.
- Implemented USB C power input with ESD protection and verified manufacturability with production Gerbers.