



**北京航空航天大学**  
B E I H A N G U N I V E R S I T Y

## **第十七届北航电子创新大赛项目设计报告**

## 摘要

近年来，智能可穿戴设备领域正在迅速升温。我们基于 ESP32 设计制作了一款智能手环，具有佩戴检测、体温检测、心率检测、久坐检测和计步功能，并可以通过手环搭载的 OLED 屏进行数据显示。同时，它能够将数据通过互联网（基于 MQTTs 协议）发送到手机 app 上。

**关键字：**可穿戴设备，物联网

## Abstract

In recent years, the field of smart wearable devices is rapidly heating up. We designed a smart band based on ESP32, which has the functions of wearing detection, body temperature detection, heart rate detection, sedentary detection and step counting, and it can display data through the OLED screen equipped on the band. At the same time, it can also send data to the mobile phone app via the Internet (based on the MQTTs protocol).

### **Keywords**

**Wearable Devices, IOT**

# 目录

摘要.....	i
Abstract.....	ii
第一章 背景介绍.....	1
1.1 ESP32.....	1
1.2 MAX30102.....	1
1.3 MPU6050.....	1
1.4 SSD1306.....	2
1.5 MQTT 相关.....	2
第二章 软件设计.....	2
2.1 总体设计.....	2
2.2 手环软件设计.....	4
2.2.1 项目结构.....	4
2.2.2 主要工作流程.....	5
2.2.3 测量准确性.....	6
2.2.4 UI 设计.....	7
2.2.5 异常处理.....	8
2.2.6 功耗管理.....	9
2.3 服务器和 app 设计.....	9
第三章 硬件设计.....	10
3.1 硬件总体设计.....	10
3.2 电路原理图.....	13
3.3 PCB 设计.....	14

结论.....	16
参考文献.....	17

## 第一章 背景介绍

### 1.1 ESP32

本作品选用的控制芯片为上海乐鑫公司研发的 ESP32 系列芯片。ESP32 是集成 2.4 GHz Wi-Fi 和蓝牙双模的单芯片方案，具备卓越的性能和丰富的外设，专为移动设备、可穿戴电子产品和物联网(IoT)应用而设计。ESP32 采用台积电(TSMC)超低功耗的 40 纳米工艺，具有超高的射频性能、稳定性、通用性和可靠性，以及超低的功耗，满足不同的功耗需求，适用于各种应用场景<sup>[1]</sup>。

### 1.2 MAX30102

本作品使用了两片 MAX30102 芯片，一片用于佩戴和体温检测，一片用于心率检测。MAX30102 是由 Maxim Integrated（美信）设计生产的高灵敏度脉搏血氧和心率传感器芯片。它通过内部 LED、光电探测器、光学元件等具有环境光抑制功能的低噪声电子设备来采集脉搏和血氧信号，通过标准的 I2C 兼容接口进行通信。为移动和可穿戴设备的设计提供了完整的系统解决方案<sup>[2]</sup>。

### 1.3 MPU6050

本作品使用 MPU6050 芯片来实现久坐检测和计步功能。MPU6050 是由 InvenSense（应美盛）设计生产的六轴（陀螺仪和加速度计）MEMS 运动跟踪传感器。MPU6050 在同一块硅片上结合了 3 轴陀螺仪和 3 轴加速度计，以及板载 Digital Motion Processor™（DMP™），后者可处理复杂的 6 轴 MotionFusion 算法，是世界上首款针对智能手机、平板电脑和可穿戴式设备设计的低功耗，低成本和高性能运动跟踪传感器<sup>[3]</sup>。

## 1.4 SSD1306

本作品使用 SSD1306 模块来实现数据显示。SSD1306 是由 Solomon Systech（晶门科技）设计生产的用于发光二极管矩阵图形显示系统的单芯片 CMOS OLED / PLED 驱动器和控制器。该 IC 通过 I2C 接口或串行外围接口与 MCU 通信。它适用于许多紧凑的便携式应用，例如手机子显示屏，MP3 播放器和计算器等<sup>[4]</sup>。

## 1.5 MQTT 相关

MQTT ( Message Queuing Telemetry Transport , 消息队列遥测传输协议 ) , 是一种基于发布/订阅 ( publish/subscribe ) 模式的"轻量级"通讯协议 , 该协议构建于 TCP/IP 协议上 , 由 IBM 在 1999 年发布。MQTT 最大优点在于 , 可以以极少的代码和有限的带宽 , 为连接远程设备提供实时可靠的消息服务。作为一种低开销、低带宽占用的即时通讯协议 , 其在物联网、小型设备、移动应用等方面有较广泛的应用。

ESP-MQTT 是乐鑫官方提供的 ESP-IDF 组件。Paho MQTT 是 Eclipse 基金会以各种编程语言提供的 MQTT 的开源实现。Mosquitto 是 Eclipse 基金会提供的实现 MQTT 协议版本 5.0、3.1.1 和 3.1 的开源消息代理。

# 第二章 软件设计

## 2.1 总体设计

软件系统的总体设计如图 1 所示。系统主要分为手环、云服务器、手机三个部分 , 其中手环是本项目的重点。

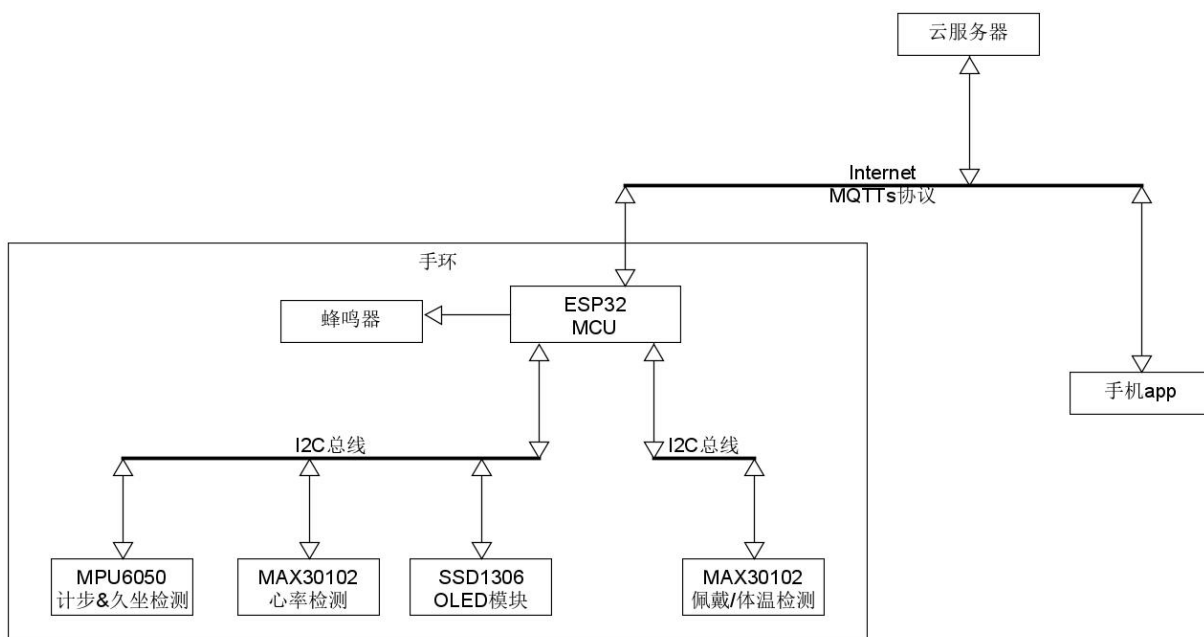


图 1 软件总体设计

蜂鸣器由 ESP32 的 GPIO（通过三极管）直接控制。ESP32 通过一条 I2C 总线与 MPU6050、MAX30102、SSD1306 模块通信，完成计步、久坐检测、心率检测和显示功能，通过另一条 I2C 总线与另一个 MAX30102 模块通信，完成佩戴和体温检测功能。由于 MAX30102 芯片在 I2C 总线上的地址不能更改，所以只能将两个 MAX30102 模块分在两条 I2C 总线上以避免冲突。

ESP32 通过自身集成的 2.4G WiFi 进行网络连接，并基于 ESP-MQTT 库建立客户端，与云服务器基于 MQTTs 协议通信。手机 app 基于 Paho MQTT 库建立客户端，与云服务器进行通信。云服务器上搭建有 Mosquitto 服务器，完成消息在 ESP32 和手机 app 之间中转的任务。

整个项目已经在 Gitee 上开源，仓库网址为 [https://gitee.com/narukara/BUAA\\_EC](https://gitee.com/narukara/BUAA_EC)。



## 2.2 手环软件设计

### 2.2.1 项目结构

手环软件项目结构如图 2 所示，整个项目采用了模块化的思想进行设计。项目主要包括 main 和 components 两个部分，main 文件夹下是项目的主程序，components 文件夹下是各个模块，包括 MAX30102、MPU6050、SSD1306 的驱动程序，以及对 GPIO、I2C、MQTT 相关操作的封装。

NAR\_GPIO 文件夹下的源文件提供了对 LED、蜂鸣器、按钮等功能的封装。NAR\_I2C 提供了对两条 I2C 总线的初始化和读写操作的封装。NAR\_MQTT 提供了对网络连接和 MQTT 收发操作的封装。

SSD1306 的驱动程序移植自 GitHub 上的项目 Minimal SSD1306 display driver[5]，并在此项目基础上扩展了大量实用功能，其中 font.h 为本项目使用的字/图形库，通过字模提取软件获得。

MPU6050 的驱动程序移植自 InvenSense 官方提供的驱动程序 eMD 5.1.3[6]，并在其基础上增加了一层封装。计步功能由 MPU6050 内置的 Digital Motion Processor™(DMP™)完成，系统初始化时，ESP32 将 DMP 所需固件写入 MPU6050 芯片，之后 DMP 可以独立工作并完成计步，ESP32 只需简单地向它查询当前步数即可。

MAX30102 的驱动程序参考和移植自 Maxim Integrated 官方提供的参考设计 MAXREFDES117#[7]，并在其基础上增加了一层封装。由于 MAX30102 是通过内置 LED 发光，然后采集人体组织的反射光来计算心率等数据的，因此可以通过判断反射光的强度来判断手环是否被佩戴：当手环处于佩戴状态时，反射光较强。另外，

MAX30102 本身也提供了温度测量功能。

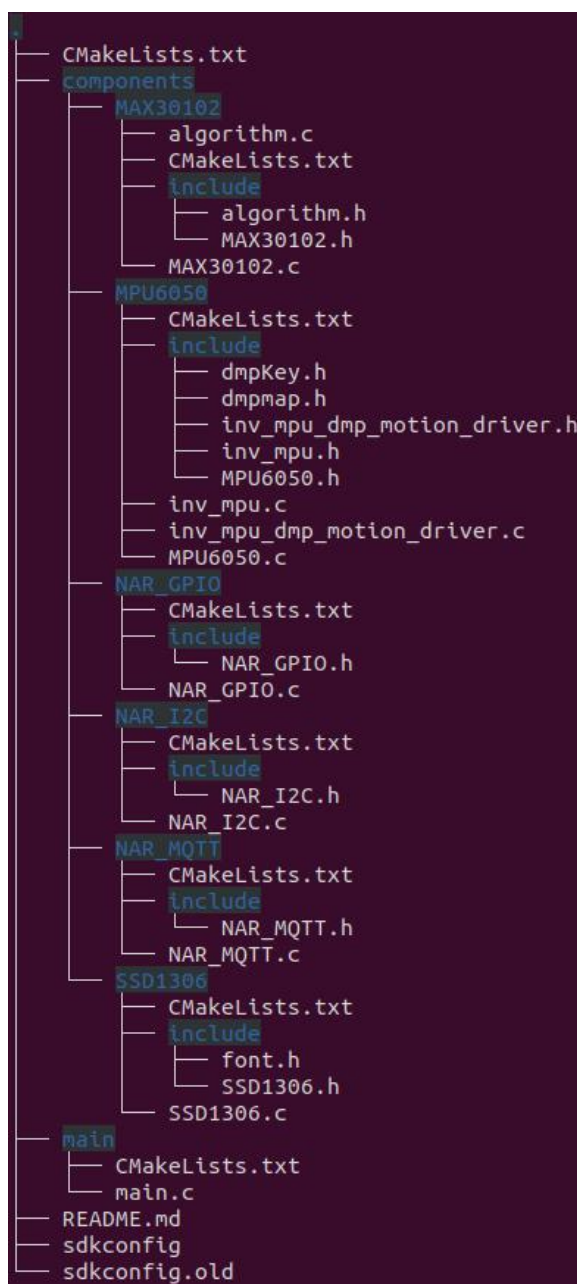


图 2 项目文件结构

### 2.2.2 主要工作流程

系统的主要工作流程由 main.c 控制，流程图如图 3 所示。ESP-IDF 内置对 FreeRTOS 的支持，因此可以很方便地进行线程管理。上电后，主线程调用各模块提供的初始化接口，完成初始配置工作，然后创建一个新线程负责网络管理。之后主线

程根据佩戴与否，在休眠模式和佩戴模式间切换，并调用各模块提供的接口完成数据采集、显示等工作。具体的实现和流程参考 main.c 中源代码。

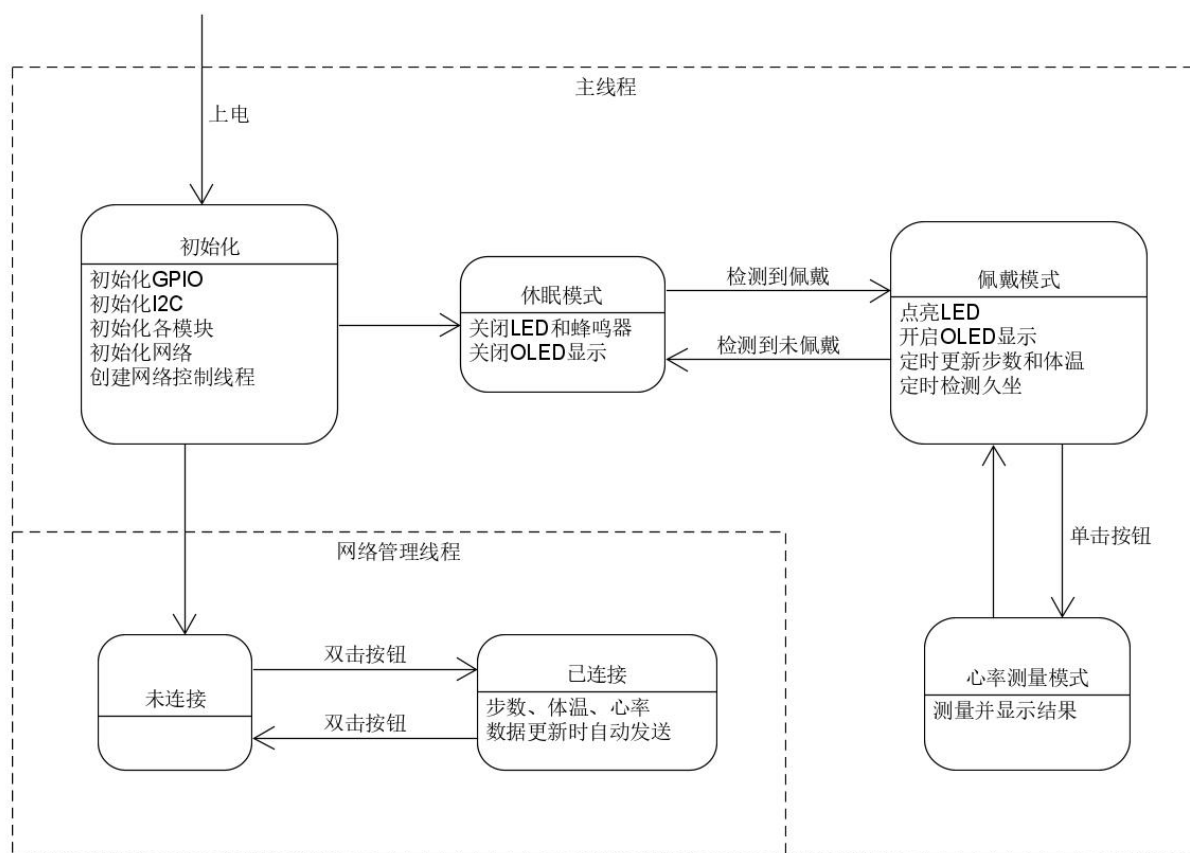


图 3 流程图

### 2.2.3 测量准确性

计步功能是由 MPU6050 芯片中的 DMP 实现，其使用的算法不得而知。但经过实际测试，计步准确度极高：一次行走中无论走了多少步（大于 7），一般只会有一到两步的误差。也就是说，一般只会在开始和结束行走时产生一步左右的误差，在行走的途中基本无误差。另外，根据官方的说明，低于 7 步的行走将被忽略，以减小误差 [8]。

心率测量所使用的算法由 Maxim 官方提供的算法改进而来，但准确度依然不够理想，且会受到测量时用户手部运动的强烈影响，因此建议在使用时尽可能保持稳定。

以非入侵的方式测量人体体温一般而言是比较困难的。事实上，手环测量的仅是手腕部的体表温度，往往低于人体核心温度，而且易受环境影响，想要以此推算出人体核心温度极为困难。目前我们采取的做法仅是显示直接测得的体表温度。

久坐检测理论上也难以实现，因为在久坐时，人的手臂一般也不会静止不动，且站姿和坐姿也难以区分。目前我们采用的方法较为简单，只要一段时间内步数没有增加，就判定为久坐。

佩戴检测是通过将反射光的强度与某一阈值进行比较而实现的，这一阈值是通过实际测试确定的。一般而言，佩戴时皮肤距 MAX30102 模块较近，反射光强度较高，摘下时将手环置于桌面上，桌面距模块较远，反射光强度较低。但实际情况是，部分材质的桌面反光能力很强，以至于不能很好地区分是否佩戴。目前设置的阈值可以准确识别佩戴，但手环置于桌面上时，有时会误识别为佩戴，且阈值已经无法继续上调，否则会导致将佩戴状态识别为未佩戴，这一问题暂时难以解决。

#### 2.2.4 UI 设计

在佩戴模式下，用户可以通过单击手环上按钮开始心率测量，或者双击按钮启动、关闭网络连接。

下面展示手环各状态下 OLED 屏的 GUI 设计：

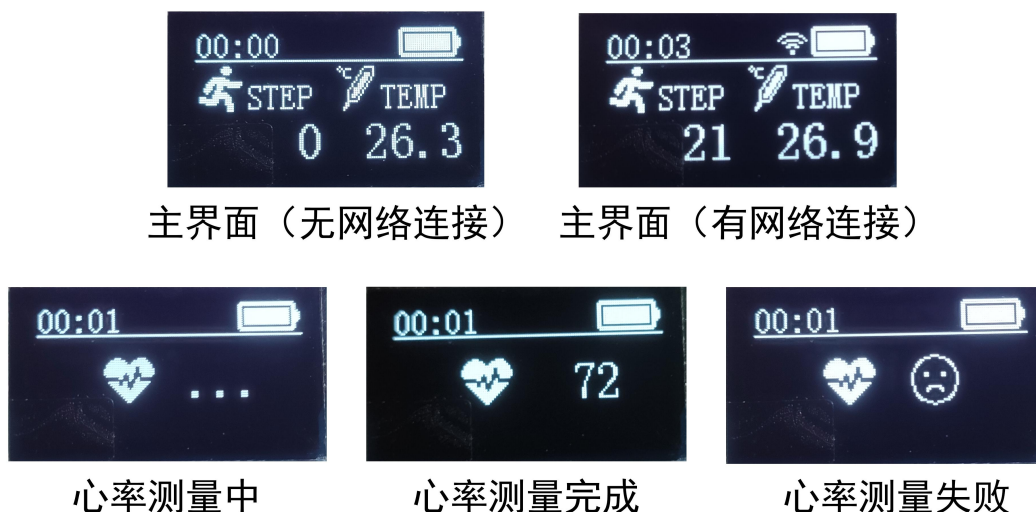


图 4 OLED 屏 GUI

### 2.2.5 异常处理

虽然项目使用的是在异常处理方面较为原始的 c 语言，但得益于模块化的设计，本项目在异常处理方面仍然有较好的实现。

本系统异常的主要来源有 I2C 通信和其他底层库的调用。异常的处理方法如下：每一个模块各自设置了一个标志变量，当某一模块产生无法处理的异常时（往往是由于硬件故障），将会把本模块标记为故障并产生日志信息，之后所有对这一模块的调用都会跳过正常的流程，直接返回明显异常，但又不影响上层逻辑的值。所有异常都在模块层上完成了捕获和处理，因此在编写主程序 main.c 时，基本无需考虑可能的异常，即对上层实现了屏蔽。而且，每个模块独立处理，一个模块的故障就不会影响另一个模块的正常工作（故障的模块被另一模块依赖除外）。我们在制作和调试时就利用了这种特性，在系统未完全组装好的情况下，就可以直接上电运行，非关键模块的缺失不会影响整个系统的工作。

### 2.2.6 功耗管理

可穿戴设备一般不能连接固定电源，或配备较大的电池，这就要求设备具有足够低的功耗。本项目手环的功耗管理措施主要有：在休眠模式下关闭 OLED 显示，仅需要时（测量心率和佩戴检测时），启用 MAX30102 模块，其他时候都将其配置为省电模式。实测数据如下（供电电压约 5V）：

**表 1 各模式功耗**

情况	电流（实际情况可能有波动）
休眠模式	24mA
佩戴模式	29mA
心率测量模式	43mA
射频芯片工作时	100mA

在手环尝试进行 Wifi 连接但尚未连接上，以及传输数据时，由于射频芯片的启动，功耗将大幅上升。而在已经连接上，但不进行数据传输的情况下，与未连接状态下的功耗近似相等。

目前，手环配备的电池容量为 900mAh（5V），假定手环工作的平均电流为 30mA，那么理论续航为 30h。

## 2.3 服务器和 app 设计

云服务器为 CentOS 系统，其上搭建了 Mosquitto 服务器，并配置为了单向 SSL 认证+用户名密码的形式，即服务器使用数字证书向客户端证明自己的身份，客户端通过用户名和密码登录到服务器上，这一认证形式具有较好的安全性和实用性。双向 SSL 认证虽然更加安全，但是在实际应用中，给每一个设备签发一个数字证书成本过

高，因此不采用。

App 为安卓平台设计，在 Paho MQTT 库的基础上实现了基本的连接、数据接收和显示功能。下面是 app 界面截图：

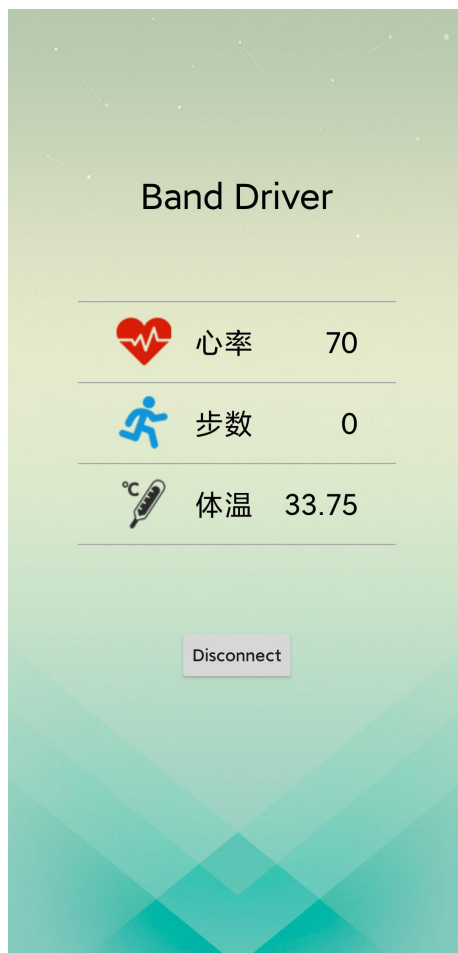


图 5 App 界面

## 第三章 硬件设计

### 3.1 硬件总体设计

硬件主要搭建了一个 PCB 平台将各个模块稳定连接在一起，以达到稳定性强，便于携带的目的。整个硬件系统的核心是 ESP32 模块，周边根据单片机资源的调度情况依照布线简洁、方便使用（如显示屏角度，心率测量模块）和满足其他硬件布局需

要（如电池安放）的原则进行了电路安排。

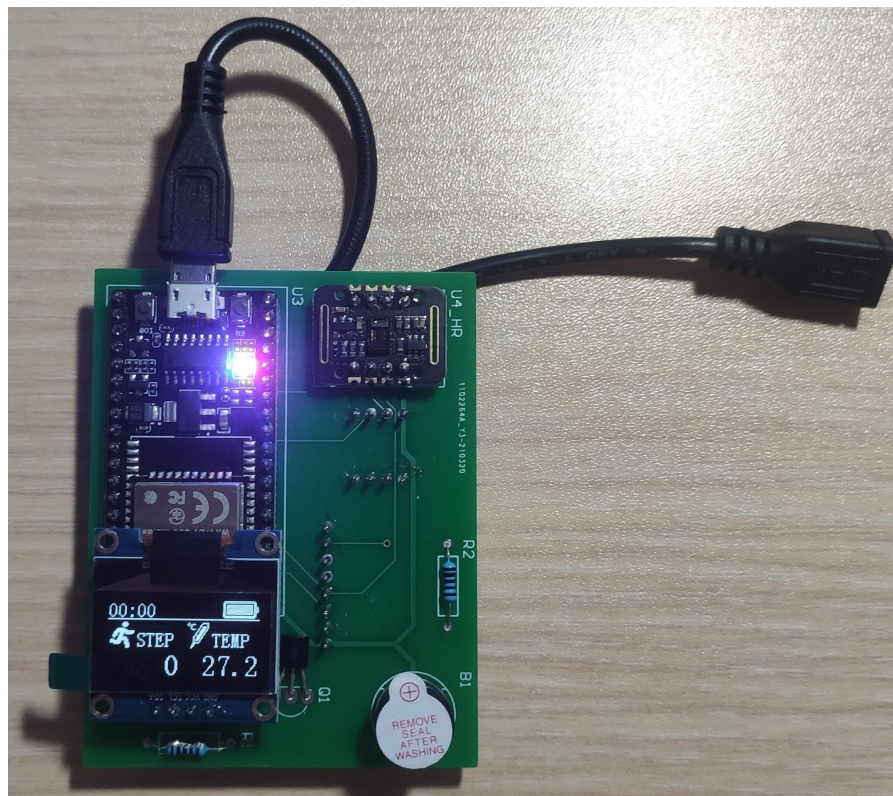


图 6 实物正面



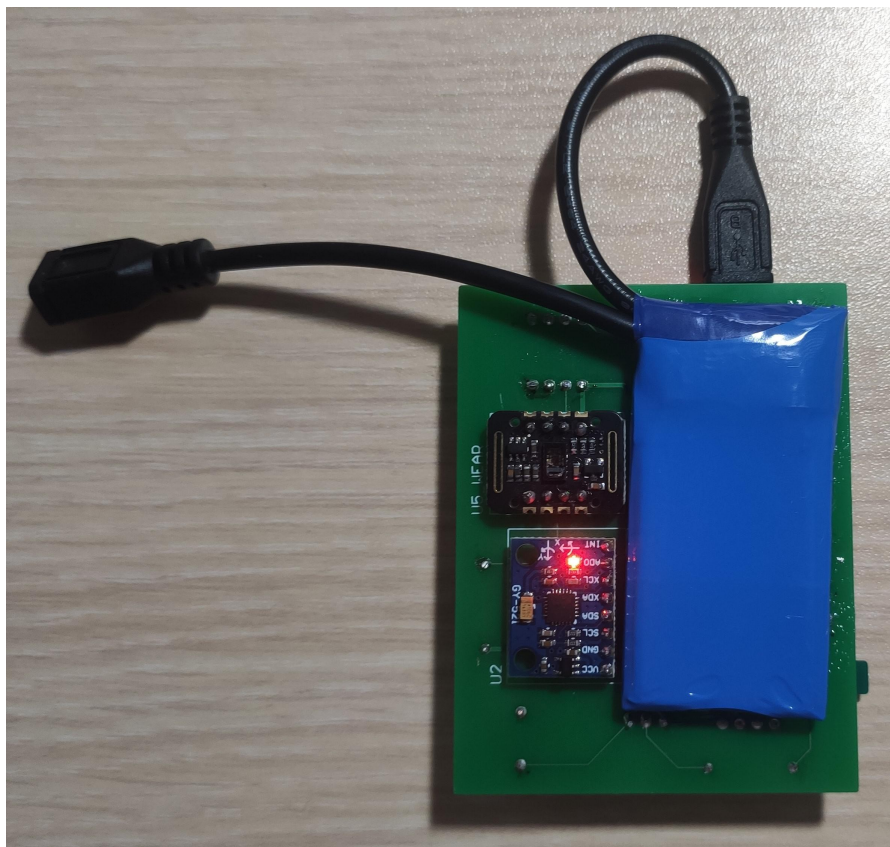


图 7 实物背面

显示屏安放在了电路板的侧边，可以让用户抬起手臂的时候显示屏的显示方向是正向的；心率监测模块放到了显示屏的大致为对角线的位置，尽量避免心率监测时手臂阻挡显示屏的观察；在 ESP32 模块的背面安排了电池的位置，这样有利于供电的走线，同时也不会让电池阻碍了佩戴监测模块的正常运行。

经过焊接后测试，系统工作良好。

### 3.2 电路原理图

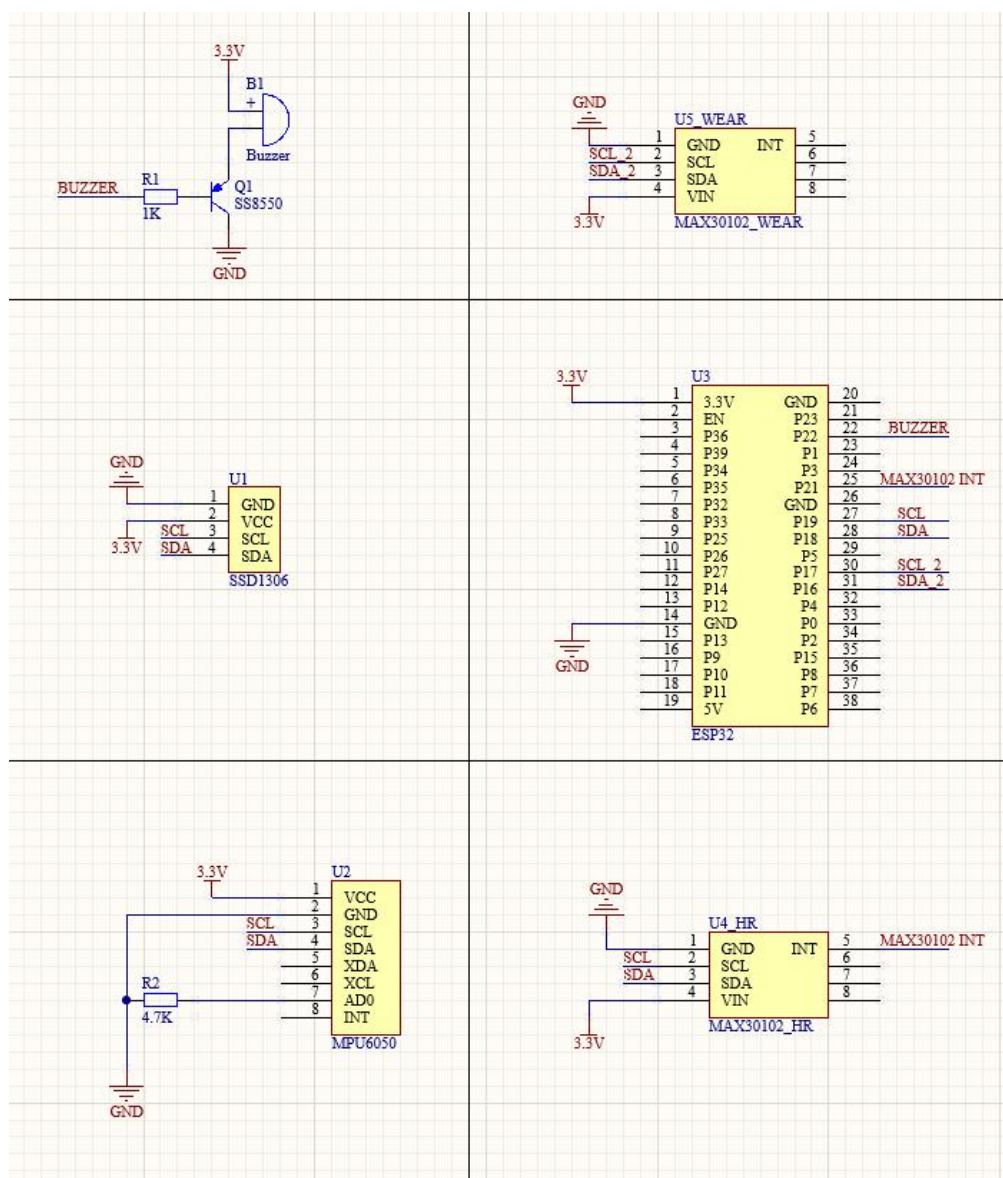


图 8 电路设计原理图

由原理图可见系统由六个模块组成——ESP32 控制模块、蜂鸣器、心率检测、佩戴检测、显示屏、加速度传感器。这些模块通过 I2C 总线和作为核心的 ESP32 连接。

### 3.3 PCB 设计

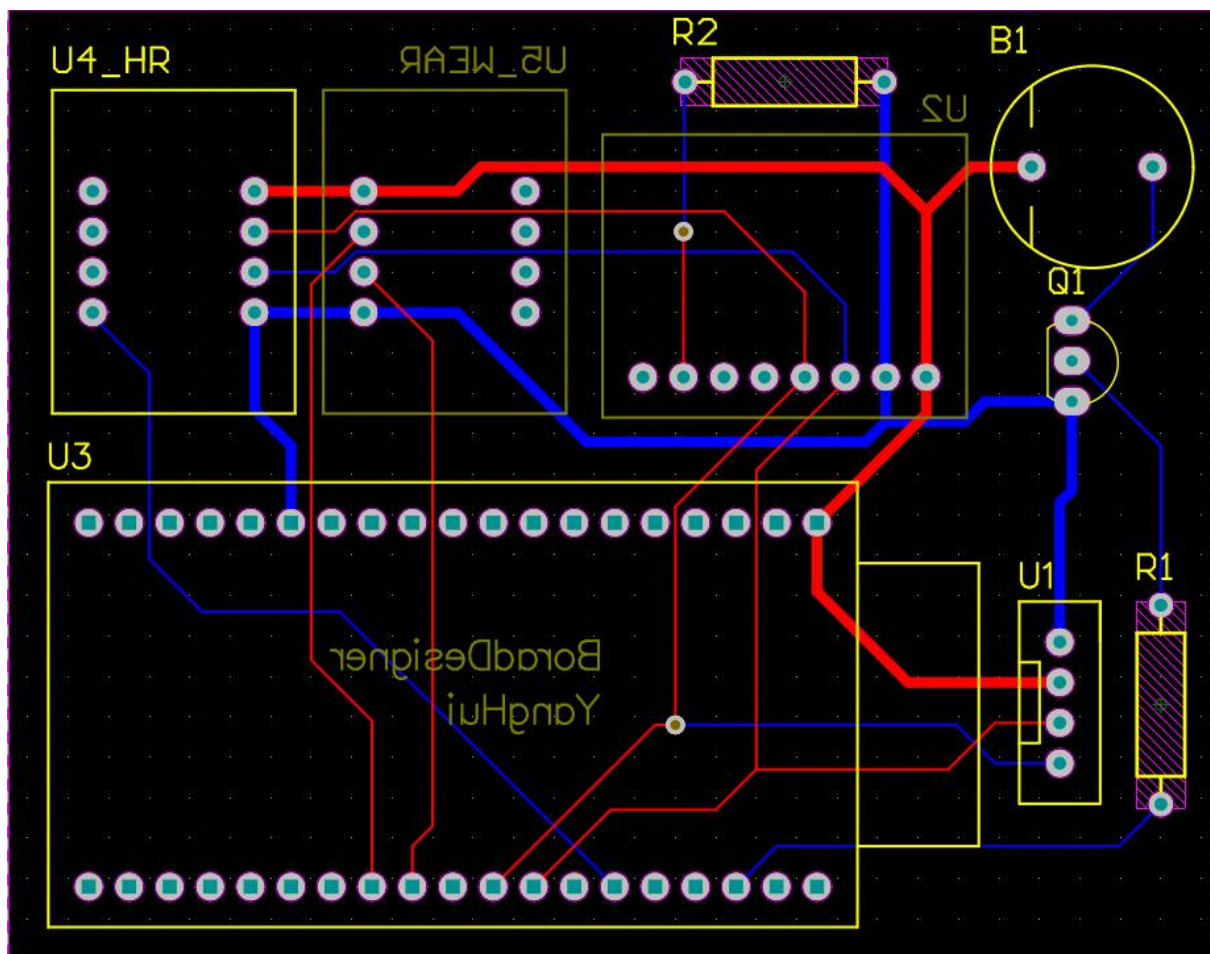


图 9 PCB 设计

PCB 的设计目的是作为一个平台将各个现成的模块稳定连接在一起，提高系统的稳定性和便携性。由于各个模块都已经集成在了各自的开发板上，并且使用的非高速 I2C 的频率仅有 100kHz，不需要严格地考虑信号完整性的问题，因此 PCB 的设计中没有进行地平面铺铜、严格信号完整性保证等设计，仅仅是通过电路板上的走线完成电路的连接。由于使用的是现成的模块，体积已经很大，没有必要单独采购使用贴片元件，因此电阻、三极管等均采用了有现成元件的直插设计。

整个电路板唯一特殊设计的是相对较粗的供电电路，根据上表 1 中测得的功耗，我们决定使用 30mil 的供电线粗细，以保证供电电路的正常工作。

整个板子的大小为  $76.2 \times 59.69 \text{ mm}^2$ ，制作出来以后大致可以覆盖整个手腕，如果需要减小面积则需要将各个模块的电路在自己的电路板上实现。这显然是一个很复杂的工程，如果日后有减小体积的需要，这方面还有很大的升级空间。

## 结论

本项目设计了一种可穿戴智能手环，实现了佩戴、体温、心率、久坐检测和计步功能，支持在搭载的 OLED 屏上直接显示数据，也可通过互联网，在 MQTTs 协议的基础上将数据发送至手机 app 上。进一步的改进方向有：添加对多手环、多手机端（即多用户）的支持，进一步减小手环体积、美化手环封装，进一步减小系统功耗，优化已有的测量准确性，添加更多实用功能，如蓝牙、时钟、电量显示、天气预报等。

## 参考文献

- [1] 乐鑫信息科技. ESP32 系列芯片技术规格书[DS/OL]. [https://www.espressif.com/sites/default/files/documentation/esp32\\_datasheet\\_cn.pdf](https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_cn.pdf).
- [2] Maxim Integrated. MAX30102 Data Sheet[DS/OL]. <https://datasheets.maximintegrated.com/en/ds/MAX30102.pdf>.
- [3] InvenSense. MPU-6050[EB/OL]. <https://invensense.tdk.com/products/motion-tracking/6-axis/mpu-6050/>.
- [4] Solomon Systech. SSD1306[EB/OL]. <https://www.solomon-systech.com/product/ssd1306/>.
- [5] uNetworking. SSD1306 in less than 100 lines[Z/OL]. <https://github.com/uNetworking/SSD1306>.
- [6] InvenSense. Software Downloads[DB/OL]. <https://invensense.tdk.com/developers/software-downloads/>.
- [7] Maxim Integrated. MAXREFDES117# : 心率和脉搏血氧监测器[DB/OL]. <https://www.maximintegrated.com/cn/design/reference-design-center/system-board/6300.html>.
- [8] InvenSense. Embedded Motion Driver 5.1.1 Tutorial[DS].