一. 项目简介

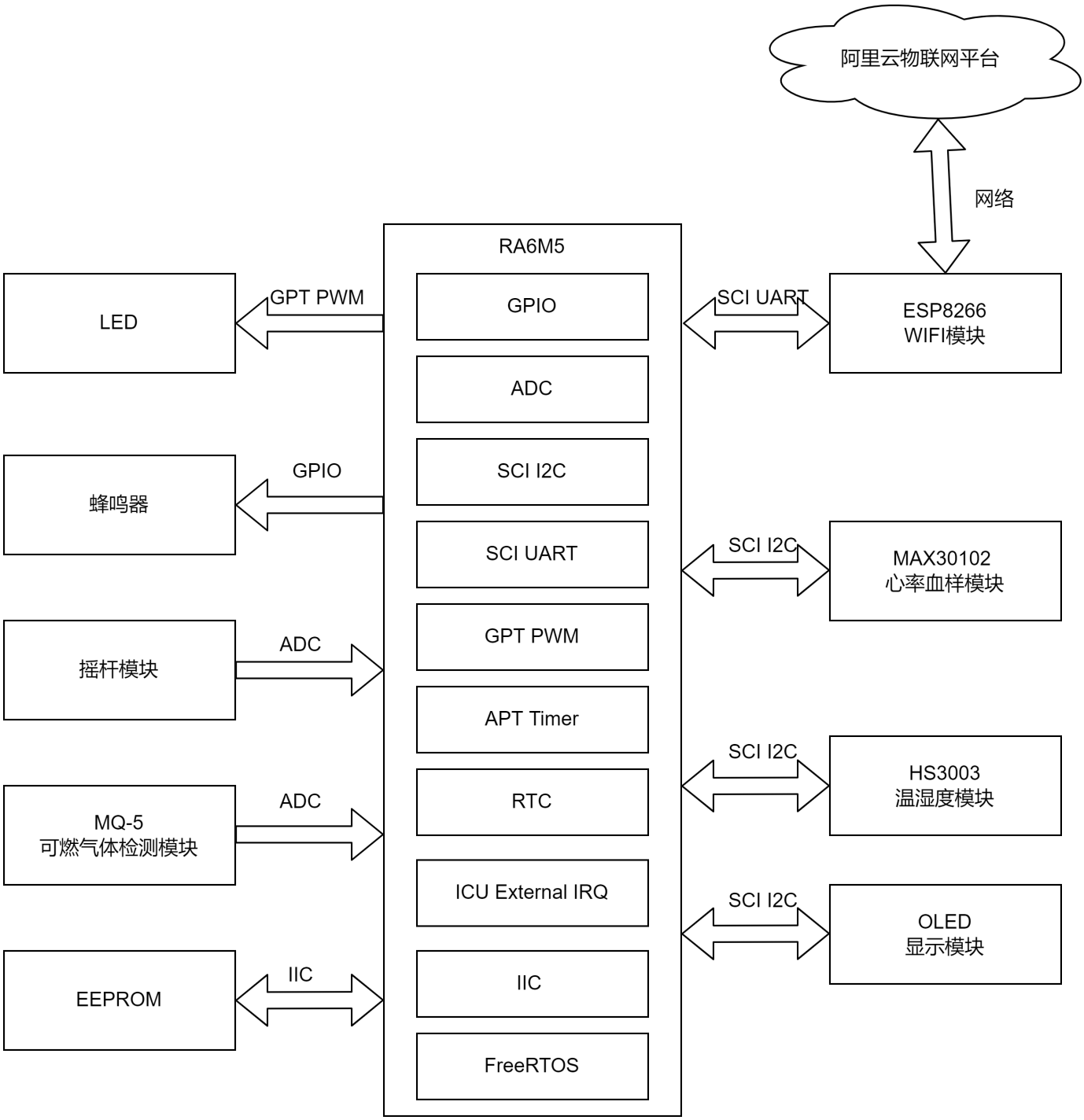
本项目实现了基于启明RA6M5的《物联网居家环境健康监控设备》。项目以启明RA6M5为主控，OLED为显示设备用于显示信息，HS3003传感器用于采集环境温湿度数据，MQ-5可燃气体传感器采集空气中可燃气体浓度，Max30102传感器采集人体心率以及血氧值，遥感模块为设备的本地控制端，内部RTC用于实时时钟，ESP8266模块用于连接物联网平台以及更新实时时钟时间，并且实现阿里云物联网平台的连接，用于数据上传与下发，蜂鸣器用于环境危险报警。本项目实现了集环境监控，健康监控以及物联网为一体的多功能监控装置。

二. 系统架构

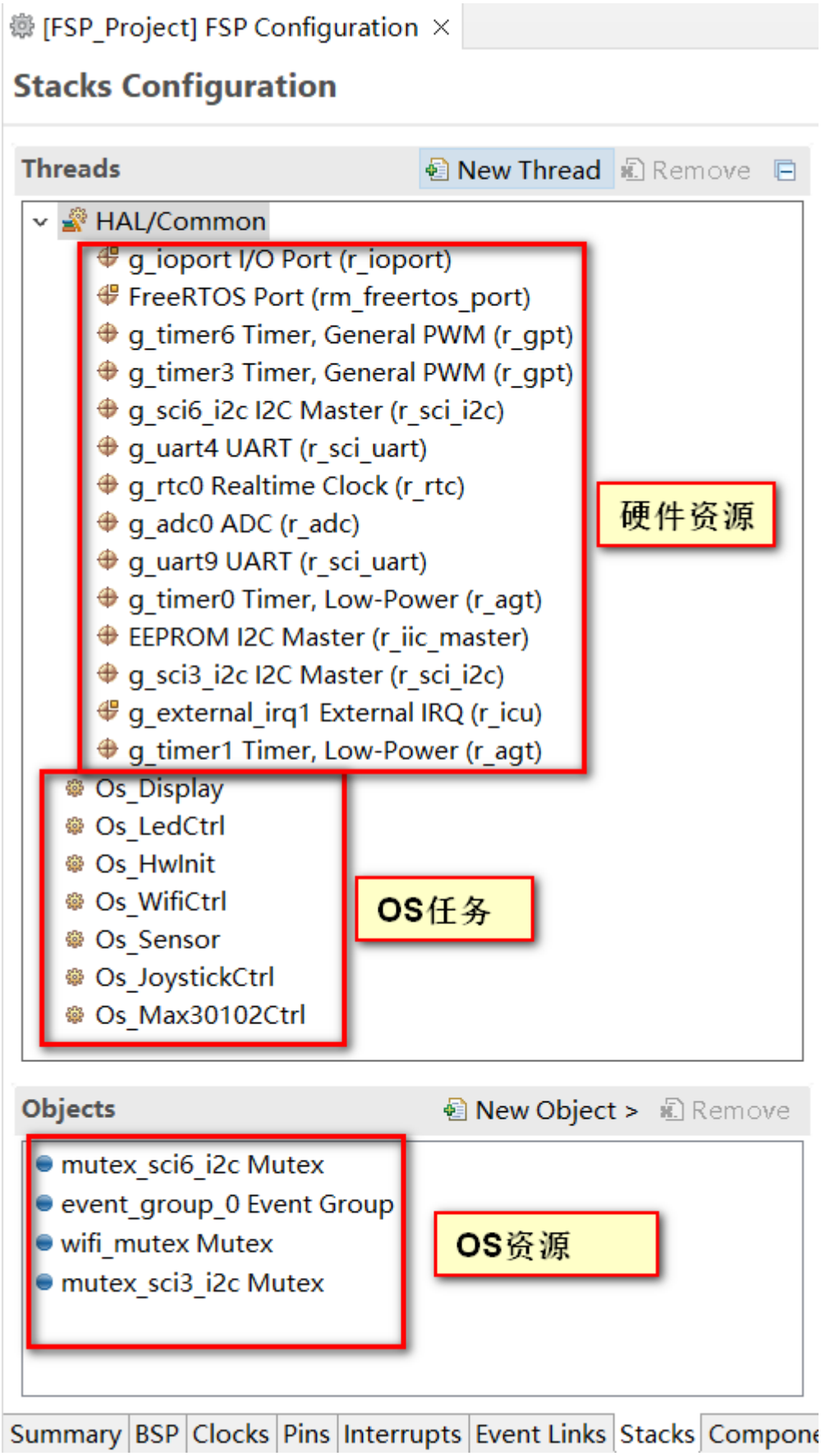
使用到的外设模块有：

- LED：红色用于模拟灯，可以被远程控制打开与关闭。绿色LED周期性呼吸，指示系统是否正常运行。
- 蜂鸣器：用于可燃气体超阈值报警。
- 摇杆模块：用于摇杆控制显示界面切换以及报警关闭。
- MQ-5可燃气体传感器：用于检测空气中可燃气体浓度值。
- EEPROM：用于保存MQ-5可燃气体浓度阈值。
- ESP8266 WIFI模块：用于更新网络时间以及连接物联网平台。
- MAX30102：用于采集人体的心率以及血氧值。
- HS3003：用于采集环境温湿度信息。
- OLED：用于显示时间，温湿度信息，可燃气体浓度，心率，血氧数值，物联网平台链接状态等信息。

详细结构描述见下图：



所使用的板载资源以及OS资源见下图所示：



三. 硬件资源说明

3.1 线路连接

板载资源不做说明(连线参考开发板原理图)，只说外接模块的连线。

3.1.1 OLED

使用了SCI I2C6:

- SCL - P505
- SDA - P506
- VCC - 3.3V
- GND - GND

3.1.2 HS3003

同OLED，使用了SCI6 I2C:

- SCL - P505
- SDA - P506
- VCC - 3.3V
- GND - GND

3.1.3 MQ-5

使用了ADC0 AN003:

- ADC- P003（AN003）

3.1.4 MAX30102

使用了SCI3 I2C：

- SCL - P408
- SDA - P409
- VCC - 3.3V
- GND - GND

INT引脚使用了外部中断：

- INT - P804

3.1.5 摇杆模块

使用了ADC0 AN001，AN002，AN010：

- X轴 - P001（AN001）
- Y轴 - P002（AN002）
- Z轴（按钮） - P010（AN010）
- +5V - 3.3V
- GND - GND

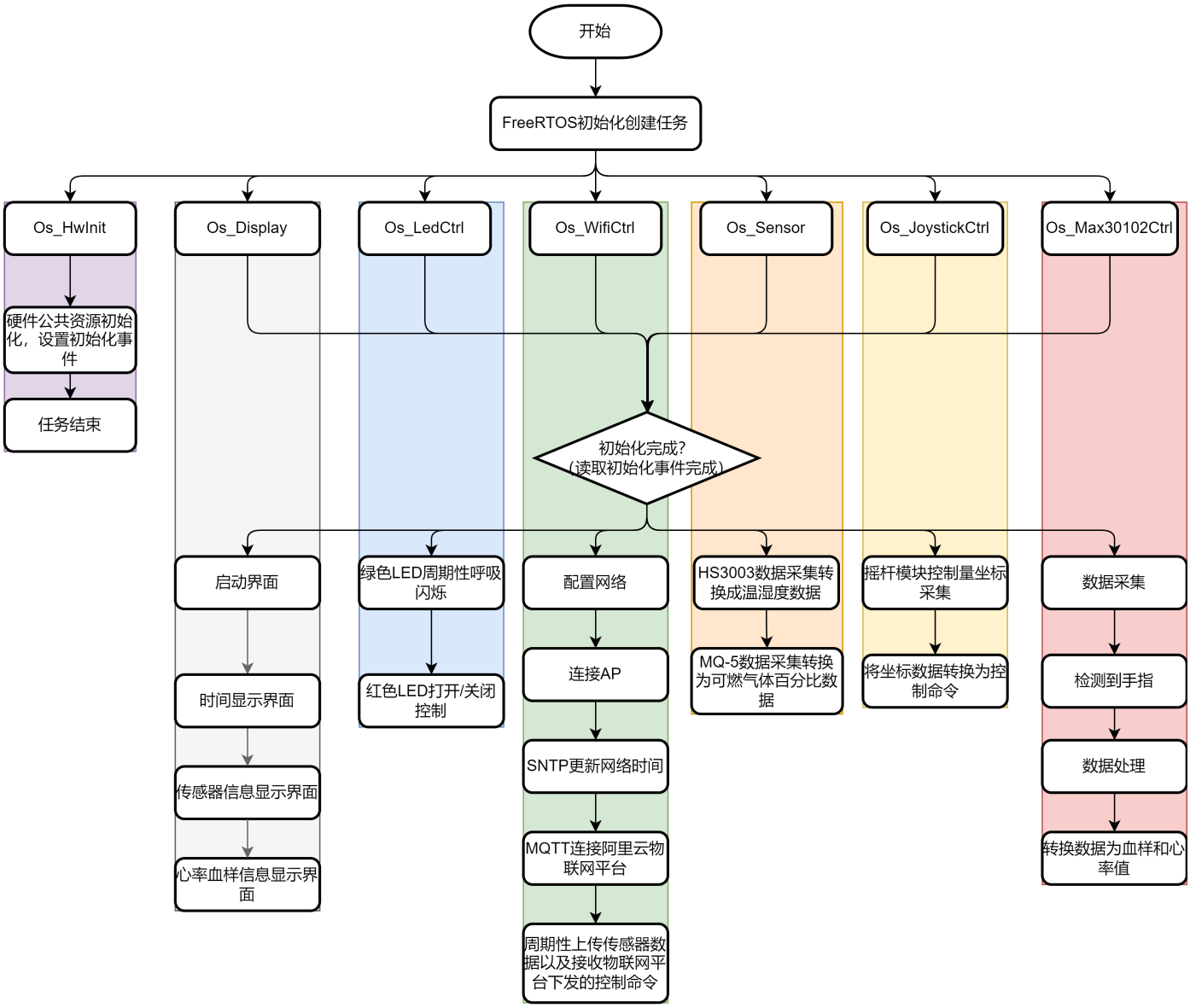
四. 软件实现

4.1 OS资源说明

软件使用了FreeRTOS实时操作系统，创建了7个任务，功能如下：

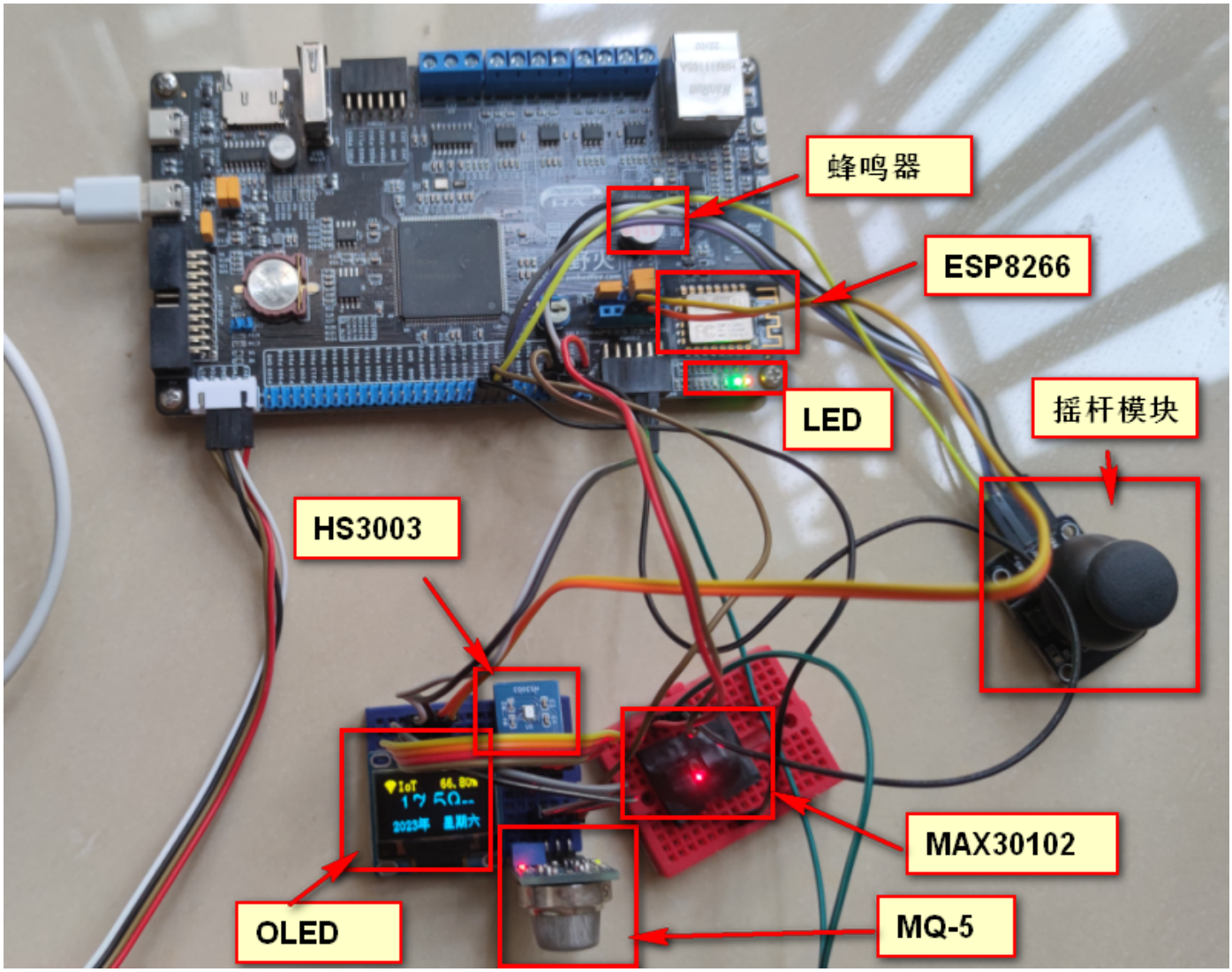
- Os_Display：用于控制OLED显示信息。
- Os_LedCtrl：用于控制LED功能。
- Os_HwInit：用于初始化公共资源，并且该任务是首先执行的，执行完毕之后其他任务才允许执行，避免资源在未初始化之前使用。
- Os_WifiCtrl：用于控制ESP8266模块，进行联网控制以及网络资源管理。
- Os_Sensor：用于采集传感器信息，包括HS3003，MQ-5。
- Os_JoystickCtrl：用于采集摇杆模块信息，并且判断摇杆模块的动作。
- Os_Max30102Ctrl：用于采集人体的心率以及血样数值。

下图简单描述了各个任务的工作，由于各个部分的关系型较为复杂，下面以文字和示例图详细说明。



4.2 功能说明

首先上一张全家福：



4.2.1 HS3003温湿度采集

通信方式：

- HS3003通过I2C通信，与OLED控制使用了同一路SCI的I2C，通过互斥量相互隔离，实现一路I2C控制两个从设备的目的。

功能：

- HS3003可以测量环境的温度以及湿度信息，通过计算公式进行转换成摄氏度以及百分比值。
- 周期性采集，每一秒钟采集一次。供其他模块使用。

4.2.2 MQ-5 可燃气体浓度采集

通信方式：

- MQ-5通过将采集到的可燃气体浓度值转换为ADC值输出，单片机周期性采集ADC信号用于计算空气中可燃气体浓度。

功能：

- MQ-5可燃气体浓度传感器可以采集空气中的可燃气体浓度，单位为ppm（百万分之一），每一秒钟更新一次数据。供其他模块使用。

4.2.3 MAX30102心率血样采集

通信方式：

- MAX30102使用I2C通信，为了精度以及采集速率，使用了单独的一路I2C来工作（其实和OLED使用同一路也是可以的）

功能：

- MAX30102上面的INT中断引脚用于判断数据是否更新，当有更新数据时，中断会拉高，数据被读取之后拉低。
- MAX30102会一直处于通信状态，当中断引脚变高之后，开始采集数据。当没有手指放上去的时候，采集到的传感器输出值RED以及IR数值，输出值大概在2000以内，但是当手指放上去之后，输出值会突变为100000以上，以此来判断是否有手指放在传感器上做数据采集。
- 采集到的RED以及IR值会周期性变化，以此来计算心率以及血样数值。
- 手指刚放上去的时候采集到的数值是不稳定的，需要等待一段时间才会采集到稳定的数值，血氧值大于95%，心率在60到120之前是较为正常的值。

4.2.4 摇杆模块控制功能

通信方式：

- 摇杆模块有两个方向（X轴和Y轴）以及一个按钮（Z轴），都是通过ADC进行采样。

功能：

- 对摇杆模块的控制，在X轴，Y轴和Z轴上都会输出一个ADC值，通过对采样值进行分析判断可以直到当前摇杆处于什么位置，以此来实现控制的功能。
- 摇杆模块用于对现实界面进行切换，当摇杆模块滑向右和上方时（即X轴以及Y轴的正方向），则会切换到传感器显示界面；当摇杆模块滑向左或者下方时（即X轴

和Y轴的负方向）则切换到时间显示界面。

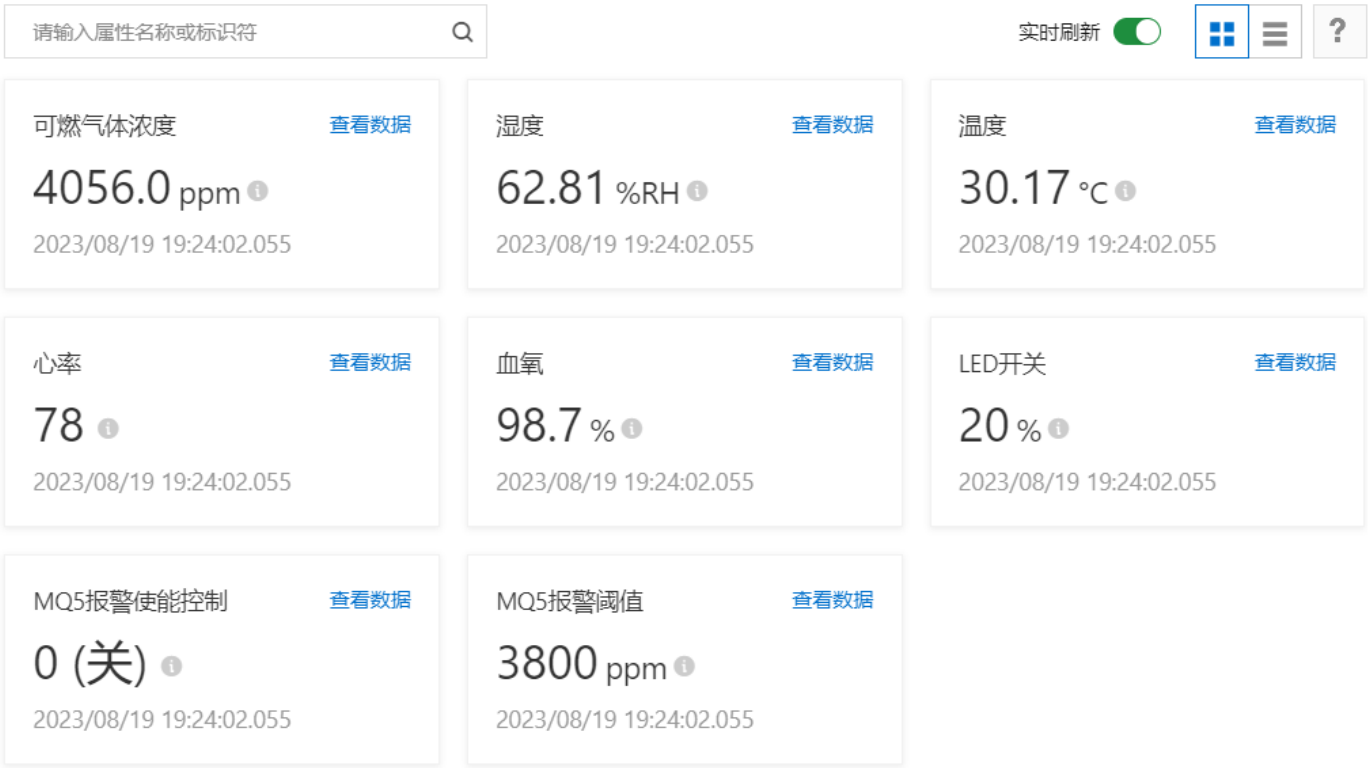
4.2.5 报警功能

- 报警使用了板载的蜂鸣器，当MQ-5采集到的空气中可燃气体浓度大于设定阈值的时候就会打开蜂鸣器报警，报警阈值可以通过物联网设置。也可以通过物联网控制报警功能打开/关闭。
- 当设定阈值更新时，会判断更新的阈值是否和当前阈值相等，不等则写入EEPROM，下次系统上电的时候会自动从EEPROM中读取阈值

4.2.6 ESP8266 网络通信

- 板载的ESP8266 WIFI模块用于物联网通信，通信平台为阿里云物联网平台，通信方式为MQTT。
- ESP8266上电工作流程：
  - 复位ESP8266
  - 关闭回显，连接AP
  - 连接AP完成之后更新SNTP时间，并将时间写入到RTC
  - 时间更新完成之后MQTT通信连接阿里云物联网平台
  - 数据通信（上载和下发），上载一秒钟更新一次
- 使用了一个定时器用来判断ESP8266 AT指令是否响应超时，如果超时，会重新发送。
- 连接了物联网平台之后，会将温度、湿度、可燃气体浓度、血氧数值，心率数值，可燃气体报警阈值、报警开关以及LED亮度上传到阿里云物联网平台。
- 物联网平台可以下发LED亮度百分比命令来控制本地LED的亮度。
- 物联网平台可以下发可燃气体浓度报警阈值到本地。
- 物联网平台可以下发可燃气体报警使能命令到本地。

下图为阿里云平台收到的数据：



下图为阿里云平台下发数据的调试界面：



4.2.7 OLED显示功能

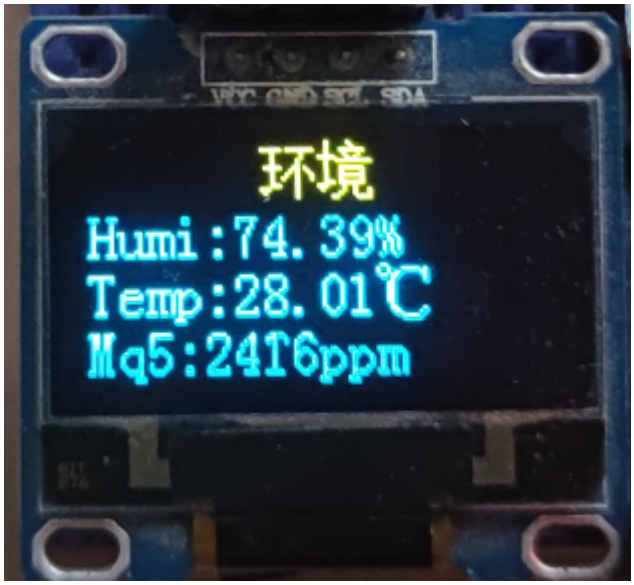
- OLED提供显示功能，包括：
    - 上电显示
    - 日期时间星期显示
    - wifi连接状态显示
    - IoT物联网平台连接状态显示
    - 温度、湿度、可燃气体浓度显示
    - 心率、血氧显示
- 上电之后会显示启动界面，1s之后切换到时间显示界面



- 时间显示界面下会显示WIFI连接状态，IoT物联网平台连接状态，日期时间星期，空气湿度和温度。wifi图标闪烁，表示在连接AP，常亮表示连接成功；IoT图标闪烁表示在连接物联网平台，常亮表示连接成功；时间或从RTC中读取并显示。



- 当摇杆模块滑向X轴或者Y轴正方向时，切换到传感器数据显示界面，此界面下，摇杆模块滑到X轴或者Y轴负方向时，切换到时间显示界面



- 当手指放到MAX30102传感器上进行测量时，切换到心率血样显示界面，此界面下，手指离开MAX30102，则退回到之前的界面



4.2.8 灯控制功能

- 板载的LED1来模拟普通LED灯，亮度控制位百分比，支持0~100：0为关闭LED；100为全亮度。
- 通过物联网平台下发控制亮度命令，本地接收到命令之后使用PWM控制LED的亮度。

4.2.9 RTC

- RTC用于本地实时时钟

- 上电会自动从上一次的时间开始计时，EAP8266成功连接AP之后之后会自动发起时间更新命令获取SNTP时间，并更新到本地RTC中。

五. 源码分享

源码上传到了gitee，链接如下：

https://gitee.com/hehung/envir_-body-heath

六. 视频演示

演示视频放置到了BiliBili平台，链接如下：

https://www.bilibili.com/video/BV1Jz4y1u7oc/

七. 总结

本人是第一次使用野火的开发板，之前听说过野火，但是一直没机会接触。无意间在公众号中看到了本次比赛，遂毫不犹豫的参加了。启明RA6M5给我的感觉就是精简，美观，板载资源丰富，扩展性强，给作品开发带来了很多便携性，像是板载ESP8266以及蜂鸣器，就减少了外扩模块的工作。历时三个月，终于算是完成了比赛，期间的心酸以及收获也难以述说，遇到了很多难题，也收获了很多知识。

简而言之，本次比赛学习到了很多东西，也让我了解了更多野火的产品。感谢野火提供的本次机会。祝愿野火越来越好，开发更多物美价廉的产品，方便广大爱好者以及工程师。