

气体流量在线监测系统设计报告

版本：V1.0

发布说明：

日期	版本	说明
2019.01	V1.0	首次发布

目录

1、	项目介绍.....	2
2、	硬件说明.....	3
2.1	电源模块.....	3
2.2	STM32F429 核心板.....	4
2.3	ESP32 模块.....	5
2.4	FS2012 模块.....	6
2.5	系统硬件 PCB 设计.....	6
3、	软件说明.....	8
3.1	软件整体设计方案.....	9
3.2	系统初始化.....	9
3.3	FS2012 采集模块.....	10
3.4	EEPROM 存取模块.....	10
3.5	ESP32 AT 指令模块.....	10
3.6	MQTT 通信模块.....	11
3.7	中国移动 OneNET 平台使用.....	12
4、	效果演示.....	12
5、	总结.....	13

1、 项目介绍

气体流量在线监测系统基于 IDT 气体流量传感器 FS2012 模块而设计。主控芯片采用高性能意法半导体公司 Cortex-M4 系列微控制器 STM32F429IGT6，并采用 ESP32 无线 2.4GHz Wi-Fi 模块实现系统与互联网的连接。

本方案专为改变家庭传统燃气计费抄表方式，采用物联网+云平台方式实现智能在线监测和在线计费等功能。在断网断电等离线情况下可通过系统 EEPROM 和 Nand Flash 存储流量总量，待恢复网络连接情况下可继续记录并实时监测。具有实用性、便捷性、安全性等优点。

本项目采用模块化方案设计，主要构成系统的模块有 STM32F429 核心板、底板和各模块等。便于扩展各类传感器模块，实现家庭燃气、水、温湿度、空气质量等的实时在线监测功能。软件采用开源嵌入式实时操作系统 RT-Thread，具有组件丰富、可靠性高等诸多优点。

系统主要模块实物图如下图所示。

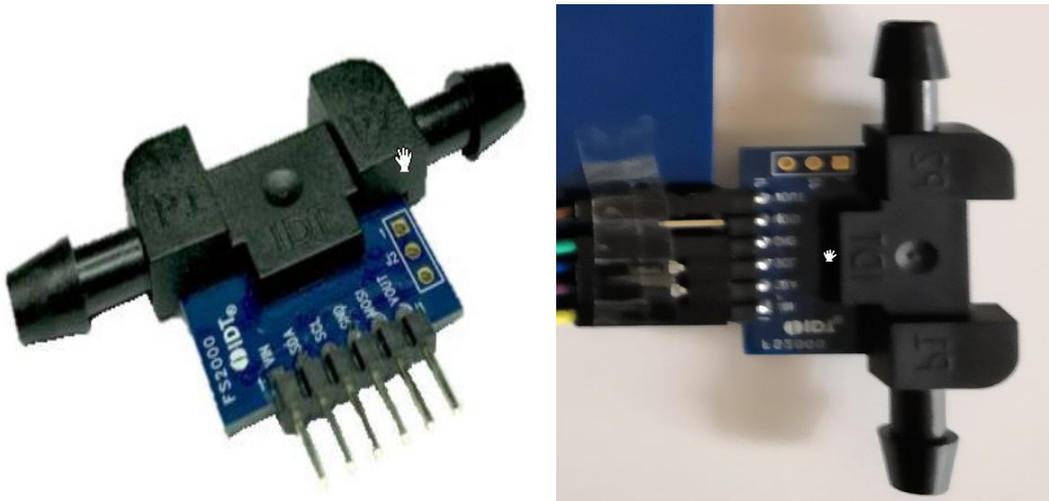


图 1 FS2012 气体流量传感器模块



图 2 STM32F429 核心板

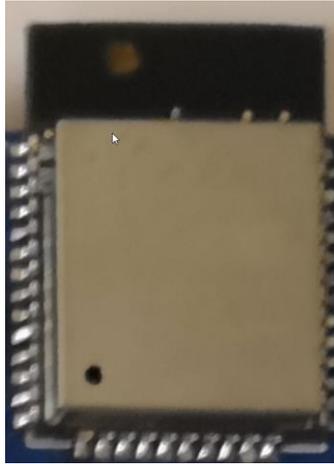


图 3 ESP32 Wi-Fi 模块

2、 硬件说明

本项目硬件主要有电源模块、STM32 核心板（含 SDRAM、EEPROM、Nand Flash、USB 通信等功能部分）、ESP32 模块电路、和 FS2012 模块。各部分电路介绍如下。

2.1 电源模块

通过分析，此项目所需功耗约为 2W。电源模块采用宽输入电压，低噪声 500mA，5V 固定输出的 LTC1763 芯片，再将 5V 电压使用 LM1117 低压差稳压器转换为 3.3V 电压给整个系统供电。电源电路原理图如图 4 所示。

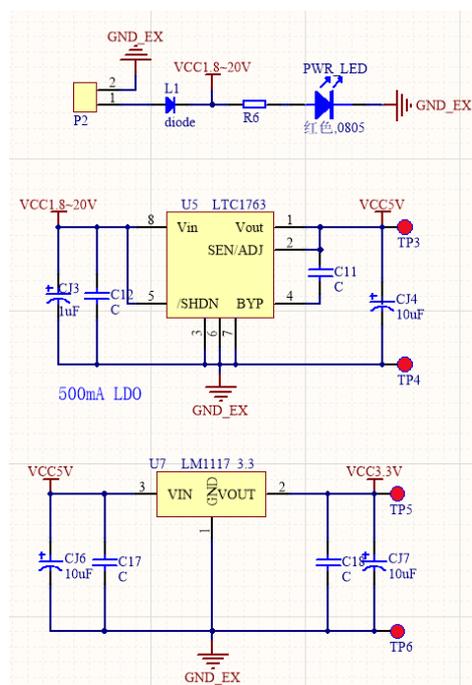
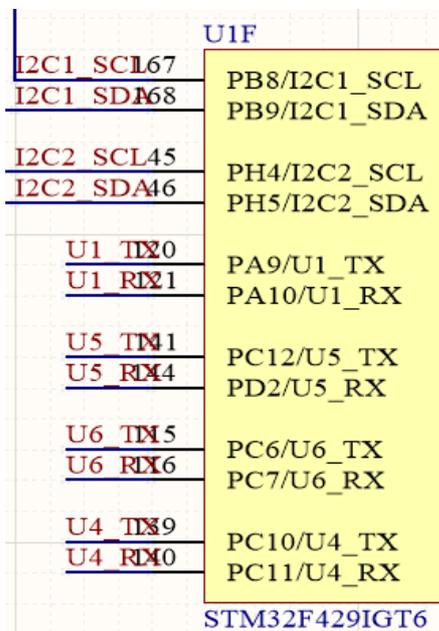
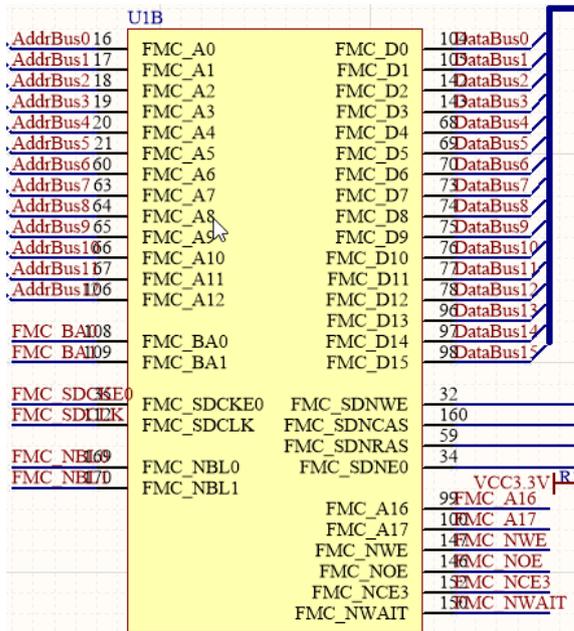
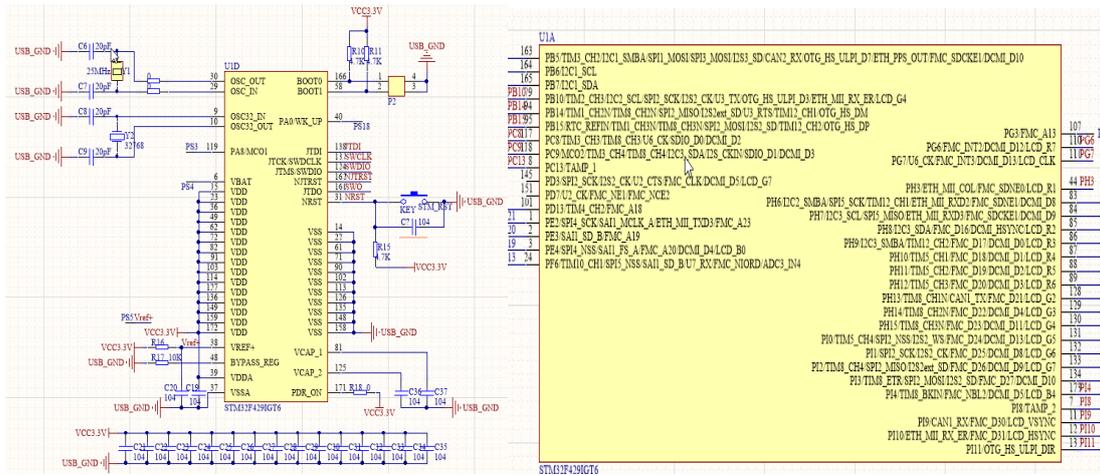


图 4 系统电源电路原理图

2.2 STM32F429 核心板

为满足系统高性能和适应嵌入式 RTOS 的需求和后续功能扩展，本项目采用 STM32F429 高性能微控制器。并配有 SDRAM、Nand Flash、EEPROM 等芯片保证系统性能。STM32F429 运行主频最高 180MHz，内部集成了多达 20 个通信模块，能支持 Compact Flash、SRAM、PSRAM、NOR、NAND 以及现在的 SDRAM 存储器，内部定时器多达 17 个，完全满足本项目需求。核心板电路原理图如图 5 所示。



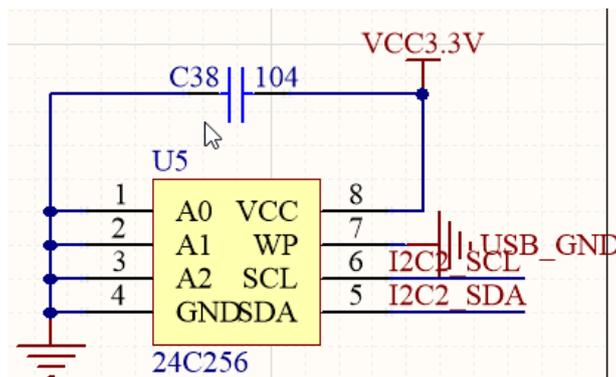
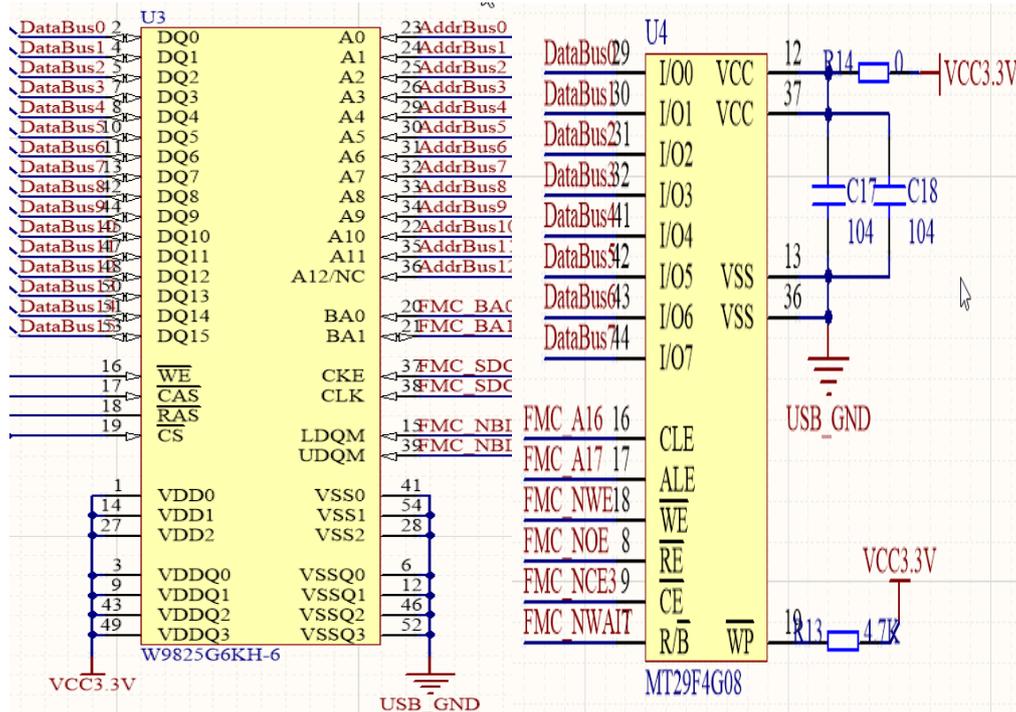


图 5 STM32 核心板原理图

2.3 ESP32 模块

ESP32 是集成 2.4GHz Wi-Fi 和蓝牙双模的单芯片方案，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。

ESP32 专为移动设备、可穿戴电子产品和物联网 (IoT) 应用而设计。本项目中使用 ESP32 模块连接系统与互联网，实现流量数据实时上传功能。ESP32 模块外围电路如图 6 所示。

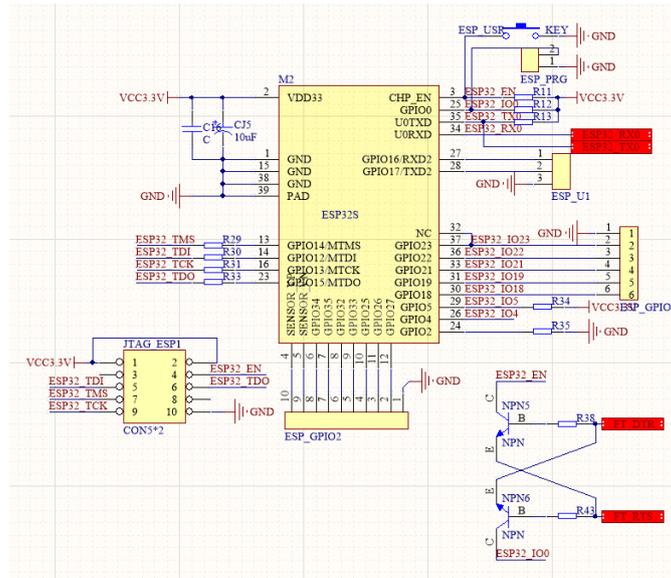


图 6 ESP32 模块外围原理图

2.4 FS2012 模块

气体流量传感器模块采用 IDT 公司高性能 FS2012 模块，实现对气体流量的采样，工作电流仅为 30mA(5V 电压)，其精度达到 2%，具有数字 I2C 接口和模拟输出接口，是实现气体流量检测的最佳选择。本项目采用排线外接 FS2012 模块，通信接口采用模拟输出接口方式与 STM32F429 连接。由于 FS2012 输出电压 0~5V，STM32 内部 ADC 采样电压最大 3.6V，因此通过两个 1K 电阻分压后输入到 STM32 的 ADC 通道。

2.5 系统硬件 PCB 设计

系统硬件采用核心板+底板方案。核心板包含 STM32F429、SDRAM、Nand Flash、EEPROM、调试接口、USB 通信接口等。为实现存储芯片的可靠工作，本项目核心板 PCB 采用 4 层 PCB 设计，其层叠设计如图 7 所示。

Layer Name	Type	Material	Thickness (mm)	Dielectric Material	Dielectric Constant	Pullback (mm)	Orientation	Coverlay Expansion
Top Overlay	Overlay							
Top Solder	Solder Mask/C...	Surface Material	0.01016	Solder Resist	3.5			0
1 Top Layer	Signal	Copper	0.03556				Top	
Dielectric 1	Dielectric	Core	0.254	FR-4	4.2			
2 GND	Internal Plane	Copper	0.036			0.508		
Dielectric 3	Dielectric	Prepreg	0.127		4.2			
3 VCC3.3V	Internal Plane	Copper	0.036			0.508		
Dielectric 2	Dielectric	Core	0.254		4.2			
4 Bottom Layer	Signal	Copper	0.03556				Bottom	
Bottom Solder	Solder Mask/C...	Surface Material	0.01016	Solder Resist	3.5			0
Bottom Overlay	Overlay							

图 7 核心板 PCB 层叠设计

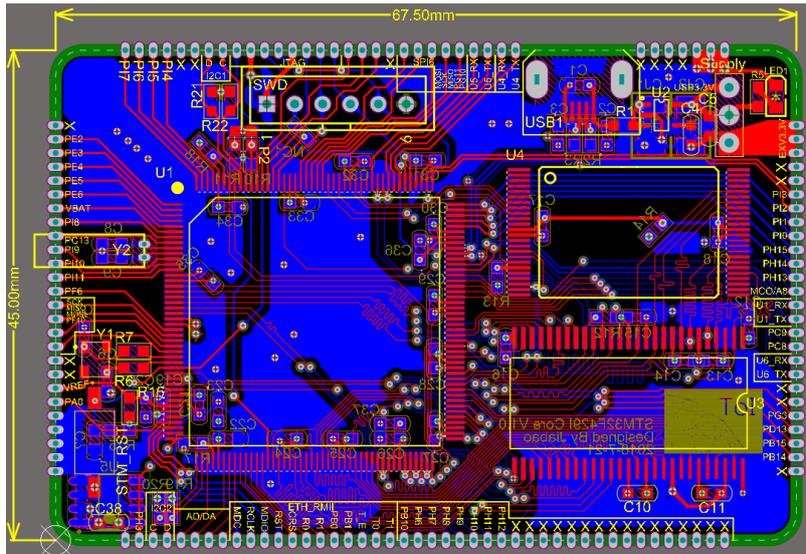


图 8 核心板 PCB 设计图

底板包含核心板接口和各种模块接口。其 PCB 如图 9 所示。

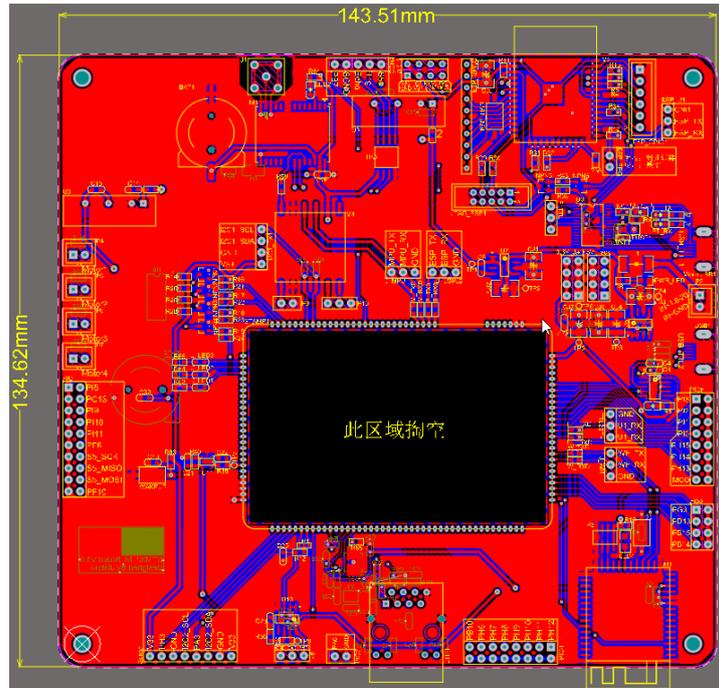


图 9 底板 PCB 设计图

经过焊接并调试各功能模块，最终制作出本项目硬件电路如图 10 所示。

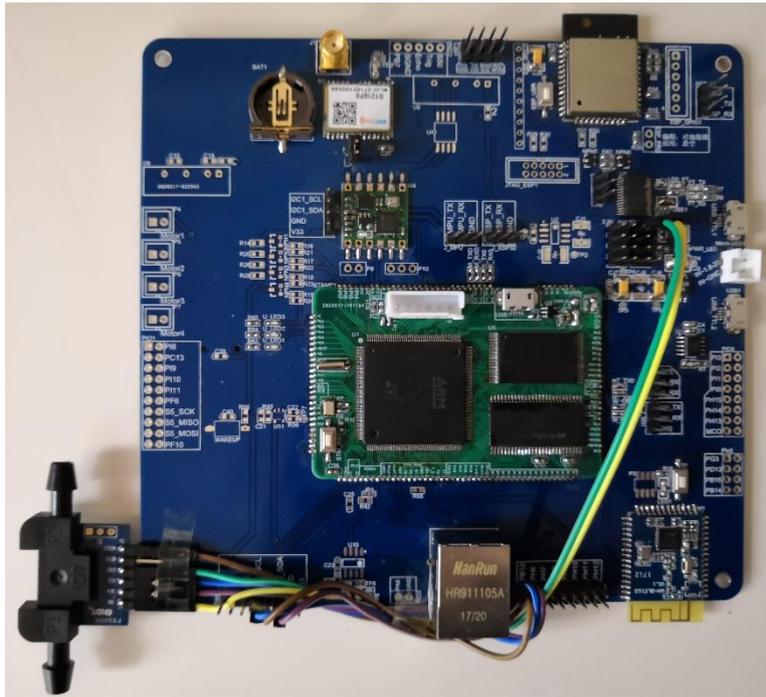


图 10 项目硬件电路实物图

3、 软件说明

本项目软件使用的功能模块较多，软件功能模块如表 1 所示。

表 1 软件功能模块

序号	软件模块	功能	备注
1	系统初始化	实现系统和各模块驱动初始化	
2	FS2012 采集模块	实现对气体流量的 AD 采集	
3	EEPROM 存取模块	实现对流量数据的存储与读取	
4	ESP32 AT 指令模块	实现对 ESP32 的控制	
5	MQTT 通信模块	实现将气体流量数据上传至互联网云平台	

若使用裸机程序编写，将花费大量时间安排各模块工作流程，以保证各模块之间无缝配合。因此本项目采用嵌入式 RTOS 进行程序设计，各个软件功能采用模块化、多任务设计。

本项目采用 RT-Thread 物联网操作系统。RT-Thread 物联网操作系统是由上海睿赛德电子科技有限公司研发的稳定可靠、简单易用、高度可伸缩、组件丰富的嵌入式 RTOS。RT-Thread 提供了专为针对物联网开发的各种组件，其中 AT 组件、SAL 组件、onenet 软件包、MQTT 软件包和各种设备驱动可用于本项目开发。并且具有简单易用的调试组件如 finsh 组件、uLog 组件，大大提高了软件开

发效率。

3.1 软件整体设计方案

软件设计采用开源嵌入式系统 RT-Thread 作为运行基本系统。在此基础上增加各模块使用的驱动和线程函数，系统每 10ms 进行一次进行线程调度，保证系统实时性。软件整体设计流程如图 11 所示。

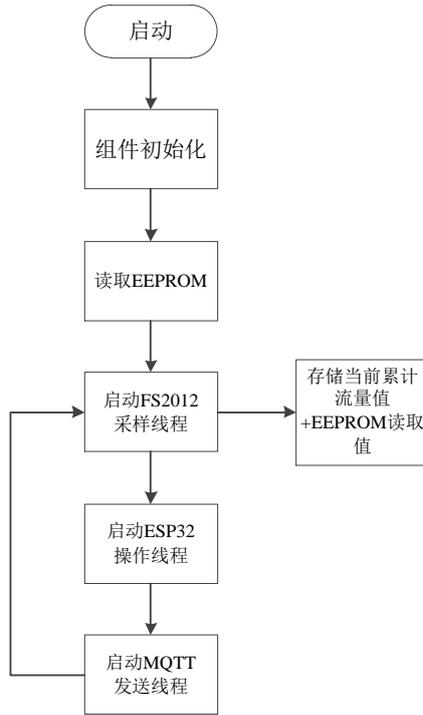


图 11 软件流程图

3.2 系统初始化

系统初始化包含系统时钟、IO、定时器、AD 采样、串口、I2C、ESP32、onenet 等初始化，不同的是，RT-Thread 操作系统提供组件初始化功能，无需在主函数初始化时初始化系统外设，应用起来十分灵活。本项目各初始化函数如表 2 所示。

表 2 组件初始化函数

序号	驱动模块	初始化函数	备注
1	板级初始化	rt_hw_board_init();	系统初始化
2	串口初始化	INIT_BOARD_EXPORT(stm32_hw_usart_init);	组件初始化
3	I2C 初始化	INIT_BOARD_EXPORT(hw_i2c1_init);	组件初始化

4	ESP32 初始化	INIT_APP_EXPORT(at_socket_device_init);	组件初始化
5	onenet_mqtt 初始化	INIT_APP_EXPORT(onenet_mqtt_init);	组件初始化

3.3 FS2012 采集模块

程序启动后即刻通过 STM32 内部 12bit ADC 采样 FS2012 输出的模拟信号，调用函数为 fs2012_mess_get_adc(rt_int16_t *mess)实现气体流量的采集。采样后通过计算得出当前实时的气体流量值。为保证长时间采集的气体流量准确性，原则上采样周期越大，计算出的流量总和越准确。本项目采样周期 1s，连续采样 10 次后计算一次当前流量平均值和总流量值。

```

while (1)
{
    fs2012_mess_get_adc(&temp[i]);
    i++;
    if(i==10)
    {
        bubbleSort(temp,10);
        temp_average = (((int)((temp[1]+temp[2]+temp[3]+temp[4]+temp[5]+temp[6]+temp[7]+temp[8])/8*1.32/245760.0-57*1.32/245760.0)*1000.0))/1000.0;
        mess_value = mess_value + temp_average*10.0;
        mess_current = temp_average;
        sprintf(s_temp, "%2.3fL/s  %3f L\r\n", mess_current, mess_value);
        rt_kprintf("%s", s_temp);
        memset(temp, 0, 10);
        eeprom_write_reg(0x10, (char)(mess_value*1000)); //mL低8位
        eeprom_write_reg(0x11, (char)(mess_value*1000/256)); //mL高8位
        i = 0;
    }
    rt_thread_delay( RT_TICK_PER_SECOND / 1 );
}

```

为保证数据准确性，对 10 次数据进行排序，去除最小值和最大值后计算 8 次数据的平均值。计算完成后向 EEPROM 写入当前总流量的值。

3.4 EEPROM 存取模块

为保证断电重启后前面的总流量数据准确，程序启动后读取一次 EEPROM 内存储的重启前的总流量值。

```

void rt_init_thread_entry(void* parameter)
{
#ifdef RT_USING_COMPONENTS_INIT
    rt_components_init();
#endif
#ifdef RT_USING_EEPROM
    eeprom_read_data();
#endif
}

```

3.5 ESP32 AT 指令模块

STM32 通过串口 AT 指令对 ESP32 模块进行控制。其模块配置过程如下所示。

```

static void esp32_init_thread_entry(void *parameter)
{
    at_response_t resp = RT_NULL;
    rt_err_t result = RT_EOK;
    rt_size_t i;

    resp = at_create_resp(2048, 0, rt_tick_from_millisecond(10000));
    if (!resp)
    {
        LOG_E("No memory for response structure!");
        result = -RT_ENOMEM;
        goto __exit;
    }

    AT_SEND_CMD(resp, "AT+RST");
    rt_thread_delay(rt_tick_from_millisecond(10000));
    AT_SEND_CMD(resp, "ATE0");
    AT_SEND_CMD(resp, "AT+CWJAP=1");
    AT_SEND_CMD(resp, "AT+GMR");
    for (i = 0; i < resp->line_counts - 1; i++)
    {
        LOG_D("%s", at_resp_get_line(resp, i + 1));
    }
    if (at_exec_cmd(at_resp_set_info(resp, 1024, 0, 20 * RT_TICK_PER_SECOND), "AT+CWJAP=\"%s\", \"%s\"",
        AT_DEVICE_WIFI_SSID, AT_DEVICE_WIFI_PASSWORD) != RT_EOK)
    {
        LOG_E("AT network initialize failed. check ssid(%s) and password(%s).", AT_DEVICE_WIFI_SSID, AT_DEVICE_WIFI_PASSWORD);
        result = -RT_ERROR;
        goto __exit;
    }

    AT_SEND_CMD(resp, "AT+CIPMUX=1");

__exit:
    if (resp)
    {
        at_delete_resp(resp);
    }

    if (!result)
    {
        LOG_I("AT network initialize success!");
    }
    else
    {
        LOG_E("AT network initialize failed (%d)!", result);
    }
}

```

3.6 MQTT 通信模块

配置完成 ESP32 模块后，ESP32 作为 TCP Client，通过 MQTT 协议与中国移动 OneNET 平台进行数据传输，主要将计算得到的气体流量当前流量速度与总流量上传至 OneNET 平台。

```

static void onenet_upload_entry(void *parameter)
{
    short value = 0;
    char s_temp[30];
    rt_thread_delay(rt_tick_from_millisecond(10 * 1000));
    while (1)
    {
        value = rand() % 100;
        if (onenet_mqtt_upload_digit("mess", mess_value) < 0)
        {
            log_e("upload has an error, stop uploading");
            break;
        }
        else
        {
            sprintf(s_temp, "buffer : {\"mess\":%.3f}", mess_value);
            log_d("%s", s_temp);
        }
        rt_thread_delay(rt_tick_from_millisecond(10 * 1000));
    }
}

```

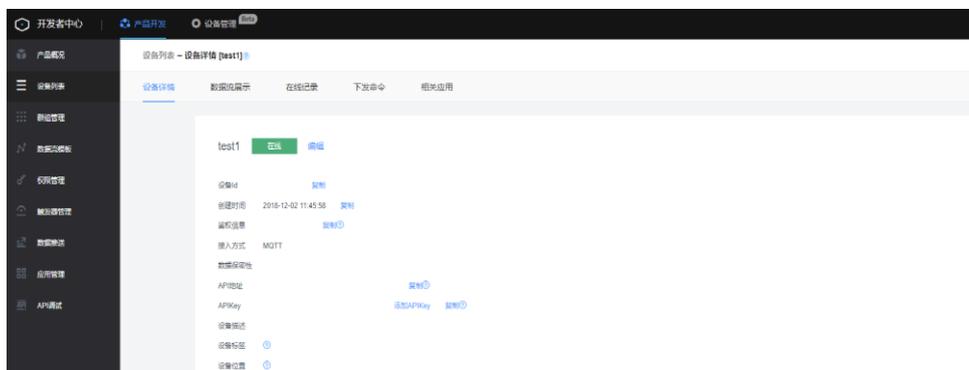
```

static void onenet_upload_entry2(void *parameter)
{
    short value = 0;
    char s_temp[30];
    rt_thread_delay(rt_tick_from_millisecond(10 * 1000));
    while (1)
    {
        if (onenet_mqtt_upload_digit("mess_curr", mess_current) < 0)//mess_current
        {
            log_e("upload has an error, stop uploading");
            break;
        }
        else
        {
            sprintf(s_temp, "buffer : {\\"mess_curr\\":%2.3f}", mess_current);
            log_d("%s", s_temp);
        }
        rt_thread_delay(rt_tick_from_millisecond(10 * 1000));
    }
}

```

3.7 中国移动 OneNET 平台使用

OneNET 云平台是由中国移动打造的 PaaS 物联网开放平台。能够帮助开发者轻松实现设备接入与设备连接，快速完成产品开发部署，为智能硬件、智能家居产品提供完善的物联网解决方案。本项目使用 MQTT 协议接入 OneNET 平台，实现流量数据的在线监测。需要在 OneNET 平台上创建账号并创建产品，复制设备 ID、鉴权信息、APIkey 等信息，将其配置进 onenet 组件包对应的位置。

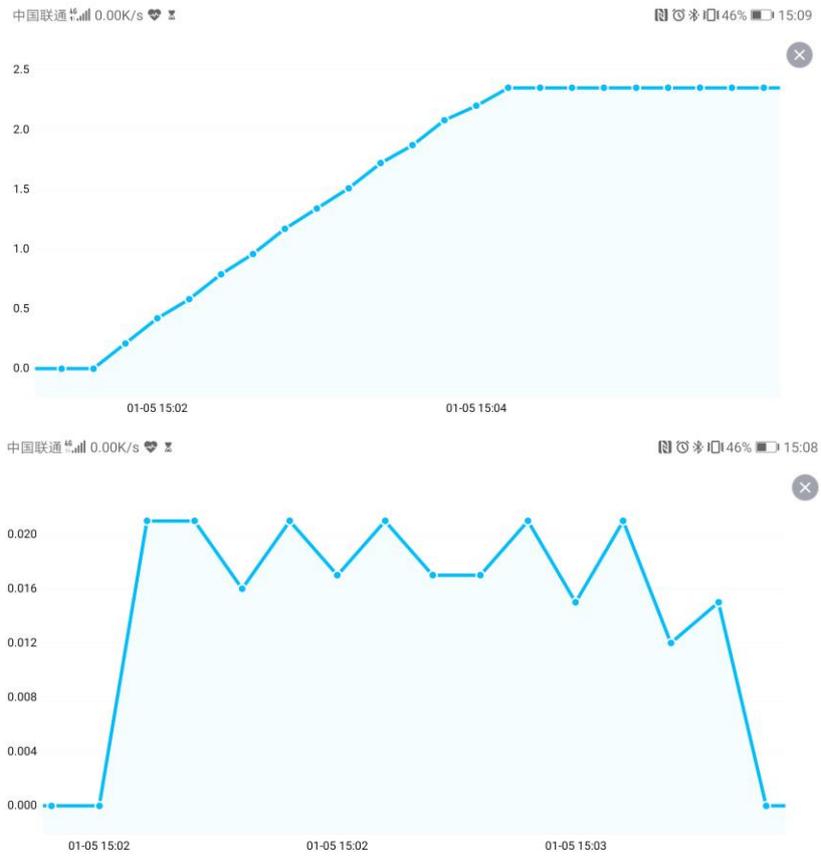


4、 效果演示

完成项目硬件设计和软件设计后，进行气体流量在线监测测试。由于目前未制作连接于燃气管道的结构工装，因此测试时使用软管对接流量传感器 P1 管道并进行吹风，获取 2 分钟测量数据。OneNET 云平台显示总流量数据和当前流量速度如下所示。



使用手机客户端查看的数据如下所示。



优酷演示视频链接：

https://v.youku.com/v_show/id_XMzk5OTY2MDgyNA==.html?spm=a2h3j.8428770.3416059.1

观看密码：messmonitor

5、 总结

本项目结合 IDT 公司提供的 FS2012 气体流量传感器套件，通过 STM32 内

部 12 位 ADC 实时采样流量数据并计算当前流量速度和总流量值。使用 ESP32 模块与中国移动 OneNET 平台通过 MQTT 协议实现数据的周期上传,在 OneNET 云平台或者手机客户端可随时进行查看当前气体实时流量和总流量。该项目可用于家庭燃气的在线监测功能。并且系统采用模块化设计,软件采用开源嵌入式实时操作系统 RT-Thread,可轻松进行功能扩展,实现智能家庭燃气、水、电、温湿度等信息的在线监测和统计,为家庭大数据提供稳定易用的数据采集上传的综合平台。