

基于单片机的热能水表检测系统的设计

【原文对照报告-大学生版】

报告编号: 810639a61254ff93

检测时间: 2019-05-10 12:17:33

检测字数: 19,059字

作者名称: 蒋艳煜

所属单位: 宿迁学院

检测范围:

- | | | |
|------------------|-----------------|-------------------|
| ◎ 中文科技期刊论文全文数据库 | ◎ 中文主要报纸全文数据库 | ◎ 中国专利特色数据库 |
| ◎ 博士/硕士学位论文全文数据库 | ◎ 中国主要会议论文特色数据库 | ◎ 港澳台文献资源 |
| ◎ 外文特色文献数据全库 | ◎ 维普优先出版论文全文数据库 | ◎ 互联网数据资源/互联网文档资源 |
| ◎ 高校自建资源库 | ◎ 图书资源 | ◎ 古籍文献资源 |
| ◎ 个人自建资源库 | ◎ 年鉴资源 | ◎ IPUB原创作品 |

时间范围: 1989-01-01至2019-05-10

检测结论:

全文总相似比	=	复写率	+	他引率	+	自引率	+	专业术语
23.92%		23.92%		0.00%		0.00%		0.00%

其他指标:

自写率: 76.08%

专业用语: 0.00%

高频词: 系统, 单片机, 数据, 可以, 采用

典型相似性: 无

指标说明:

复写率: 相似或疑似重复内容占全文的比重

他引率: 引用他人的部分占全文的比重, 请正确标注引用

自引率: 引用自己已发表部分占全文的比重, 请正确标注引用

自写率: 原创内容占全文的比重

专业用语: 公式定理、法律条文、行业用语等占全文的比重

典型相似性: 相似或疑似重复内容占互联网资源库的比重, 超过60%可以访问

总相似片段: 382

期刊: 38 博硕: 163 外文: 0 综合: 2 自建库: 0 互联网: 179

颜色标注说明:

- 自写片段
- 复写片段 (相似或疑似重复)
- 引用片段
- 引用片段(自引)
- 专业用语 (公式定理、法律条文、行业用语等)

京 江 学 院

JINGJIANG COLLEGE OF J I A N G S U U N I V E R S I T Y

本 科 毕 业 论 文

基于单片机的热能水表间的系统的设计

Design of Thermal Energy Meter Based on ARM

专 业: 通信工程

班 级: 15通信2班

学 号: 20150303232

姓 名: 蒋艳煜

指导教师姓名: 袁静

指导教师职称: 副教授

2019年 5 月

基于单片机的热能水表检测系统的设计

摘要: 热能表是用于测量及显示水流经热交换回路后吸收或者释放的热量的仪表, 长期以来, 我国北方地区城镇居民采暖一般按住宅面积而不是实际用热量收费, 导致用户节能意识差, 造成严重的资源浪费。显然该计量方法缺乏科学性。本系统设计一款基于STM32ZET6单片机的热能表检测系统。

论文主要工作如下:

1. 硬件部分:

热能表的设计基于防水式DS18B20传感器为基础, 测量入水口以及出水口的水温变化情况, 利用STM32ZET6单片机进行数据的存储以及运算处理, 本系统的硬件主要选择STM32ZET6单片机、二个防水式DS18B20传感器、RC522射频模块、MQ2气体浓度传感器、水流量传感器、TFT显示屏以及水泵等模块。系统通电后, 需要先用门禁卡登录, 才能进入系统, 然后两个防水式DS18B20来检测入水口和出水口的水温, 将水温数据传送给单片机, 单片机处理后在TFT屏上显示出来, 并且单片机将近五次的水温数据保存至AT24C02外部存储器中, 以保证在下次通电之后, 数据依然不丢失。其次, 主页面还显示有水流量数据以及热能表附近环境的气体浓度数据等, 最后还设计有水泵管理操作, 可以手动打开水泵或者关闭水泵。

2. 软件部分:

本文采用的是在Keil软件里面使用C语言编写程序代码, 介绍了Keil软件的设计思想。主函数主要定义了引脚, 并将各子函数初始化。并且接受各个子函数及主函数的设计流程图。了解下载串口与程序烧录软件, 将下载串口与电脑连接, 在电脑的设备管理器里找串口。完成驱动的安装。然后在烧录软件里面选择芯片以及找路径, 将代码烧录到硬件当中。

关键词: 热能表 水流量传感器 TFT显示屏 STM32ZET6 DS18B20

Design of Thermal Energy Meter Based on ARM

Abstract: Thermal energy meter is an instrument used to measure and display the heat absorbed or released by water flow through heat exchange circuit. For a long time, urban residents in northern China generally charge for heating according to residential area rather than actual heat consumption, resulting in poor awareness of energy saving and serious waste of resources. Obviously, the measurement method is not scientific. This system designs a heat meter detection system based on STM32ZET6 single chip computer.

The main work of the thesis is as follows.

1. Hardware:

The design of the heat meter is based on the waterproof DS18B20 sensor. It measures the water temperature change of the water inlet and the water outlet. The data is stored and processed by the STM32ZET6 single-chip microcomputer. The hardware of the system mainly chooses STM32ZET6 single-chip microcomputer and two waterproof DS18B20 sensors. RC522 RF module, MQ2 gas concentration sensor, water flow sensor, TFT display and pump module. After the system is powered on, you need to log in with the access card before entering the system. Then two waterproof DS18B20s are used to detect the water temperature of the water inlet and the water outlet, and the water temperature data is transmitted to the single-chip microcomputer. After processing by the single-chip computer, it is displayed on the TFT screen, and the single-chip microcomputer The water temperature data of nearly five times is saved to the AT24C02 external memory to ensure that the data is not lost after the next power-on. Secondly, the main page also displays water flow data and gas concentration data of the environment near the heat meter. Finally, a pump management operation is also designed, which can be manually turned on or turned off.

2. Software part:

This article uses Keil software to write program code in C language and introduces the design idea of Keil software. The main function mainly defines the pin and initializes the timer. Infrared counting and distance detection module initialization and definition. The distance sensor of the buzzer section is defined. The distance control flag is cleared and the buzzer stops the alarm. The LED lamp acceptance parameters are set. Understand the download of the serial port and program burning software, will download the serial port and the computer connection, in the computer's device manager to find the serial port. Complete the installation of the driver. Then select the chip and find the path in the burning software, and burn the code into the hardware.

Keywords: Heat meter water flow sensor TFT STM32ZET6 DS18B20

目录

1.1课题背景及其意义 1

1.2国内外的研究状况 1

1.3论文主要内容及结构安排 2

1.4本章小结 2

第二章 热能表的系统方案设计 4

2.1系统的功能分析及体系结构设计 4

2.1.1 系统功能分析 4

2.1.2 系统总体结构 4

2.2控制方式的选择 5

2.2.1 单片机的型号选择 5

2.2.2 气体浓度检测的选择 6

2.2.3 显示屏的选择 8

2.2.4 水温检测模块的选择 9

2.2.6 水流量计模块选择 11

2.3本章小结 12

第三章 硬件系统的设计 13

3.1模块电路的设计 13

3.1.1 STM32F103ZET6单片机核心系统电路的设计 13

3.1.3 TFT显示屏电路设计 20

3.1.4 MQ2气体浓度电路设计 22

3.1.5 RC522登录管理电路设计 23

3.1.6 水流量计电路设计 25

3.1.7 蜂鸣器报警电路设计 25

3.2本章小结 26

第四章 软件设计 27

4.1编程语言选择 27

4.2软件流程图 27

4.3程序烧录及调试 29

4.4 部分程序代码 32

4.4.1 主控程序部分代码 32

4.4.2 显示屏部分程序代码 34

4.4.3 传感器驱动部分程序代码 38

4.4本章小结 42

第五章 系统运行结果 43

5.1系统测试情况分析 43

5.2本章小结 47

总结与展望 48

第一章 绪论

1.1课题背景及其意义

热能表主要用于供热系统中，目前，北方主要集中供暖的省份都发布了新的热量计的相关收费标准，比如黑龙江省现在的供热收费就采用了定量收费，这类法规的发布无疑会给热能表行业带来了巨大的商机。但是，我国目前的热流量计设备主要是购买国外的，我国自主研发的流量计系统质量参差不齐，生产还不成熟，在施工时存在着很大的问题，从某种程度说，这些次品也限制了供热行业的发展。

热流量计的发展对应计量收费的主要依据，而长期以来，我国北方城市的供暖计费主要依据住宅面积进行收费，这样会照常诸多不合理之处，导致用户的节能意识差，造成大量的能源浪费。为了改进这种原始的计费方式，我国于2001年12月4日，发布了JJG-225-2001《热能表检定规则》，其中对供热收费方式规定了按量收费，而不是再按以前的住宅面积收费。

作为供热系统改革的主要环节，流量计的设计也越来越重要，基于此，本课题系统主要设计了一个基于单片机的热能表系统，以单片机和防水式DSS18B20模块为核心，对入水口和出水口的水温进行监控，并在TFT屏上显示出来。

1.2国内外的研究状况

国外的供热收费计量系统主要起源于上世纪70年代中期，在欧洲最早开始运营，而热能表的发展也由最原始的机械式发展到电子模拟

式，然后又研发出了电子数字式，到目前，主流的热能表系统都是采用微处理器为核心的智能式系统。

1997年4月，欧洲各国共同规定了统一的热能表标准，代号为EN1434。并且现在国外的热能表一律采用K系数补偿的方式进行计算，该方式的缺点也很明显，就是位置是固定的，测温元件主要采用PT100传感器进去测温。

中国的热能表自主研制开始于上世纪90年代，中国最早的热能表是由山西的一位大学老师设计的。中国热能表的发展过程中，认真学习了国外的先进成熟的技术然后结合我国的具体国情进行设计。

目前我国的热能表主要特点有：

测温传感器采用PT1000，相比较国外采用的PT100测温传感器，PT1000的信噪比更高；

我国的热能表既可以按照在进水管道里也可以按照在回水管道里，这样可以有效的遏制盗水现象；

我国的热能表不仅可以测量供热量，还可以测量吸热量，一表多用，用途更加广泛；

（4）我国研制的热能表的价格比国外更低，市场竞争力也更强。

虽然我国的热能表无论在性能上还是价格上都比国外优势更大，但是由于热能表的主要元件水流量计还是主要购买国外的，自主设计相对复杂，问题颇多。所以，在热能表系统的研究上，我国还有很大的进步空间。

1.3 论文主要内容及结构安排

主要内容：系统通电后，会先进入刷卡登录的界面，当使用门禁卡刷卡登录后，就进入了主显示页面，该页面中有入水口温度数据、出水口温度数据以及水泵控制开关等按键，还有其他浓度信息以及水流量信息。点击入水口温度数据将会进入入水口温度数据显示的界面，该界面显示了近五次的入水口温度数据，并且将这些数据保持至AT24C02中，及时掉电，这些温度数据也不会丢失。点击出水口温度数据也会进入相应的温度显示的界面，同时还可以通过TFT的触摸功能来进行水泵开关的控制。

本文章一共分为五个章节，各个文章内容具体如下：

第一章：主要介绍了选题的研究意义以及国内外的研究现状；

第二章：介绍了控制方案的选择，具体介绍了单片机芯片及选择、水温检测传感器的选择、气体浓度检测传感器的选择、显示屏的选择以及水流量计的选择等构成，并介绍了相关传感器的实物图等。

第三章：介绍了系统硬件的设计方案以及各种电路的原理图，并介绍了主控单片机的各种外设功能等。

第四章：主要介绍程序流程图以及Keil软件的配置和部分程序代码等。

第五章：主要是介绍了系统的实际测试过程，并附有相关的实物图等。

最后一部分是此次毕业设计的总结与展望和心得体会。

1.4 本章小结

本章主要介绍了热能表的研究背景及意义，并根据国内外研究现状讨论了研究的目的，现在热能表已经成为供热系统中必不可少的一环，本课题从研究意义方面对热能表进行讨论，并制定出研究方案。

热能表的系统方案设计

2.1 系统的功能分析及体系结构设计

2.1.1 系统功能分析

本设计由STM32F103ZET6单片机最小系统电路、防水式DS18B20温度检测电路、环境气体浓度检测电路、水泵控制电路、RC522登录管理控制电路、水流量测量电路以及TFT屏显示电路组成。

（1）系统通电后进入登录界面，使用门禁卡登录才能进入监控界面。监控界面主要有进入入水口温度数据的接口、进入出水口温度数据的接口、室内气体浓度数据、水流量数据、水泵控制开关以及登陆者ID信息。进入该界面后，单片机通过接收MQ2、RC522射频模块以及水流量计传来的数据，进行信息显示。并且还可以在该页面控制水泵的开关以及退出登录。

（2）点击进入入水口温度数据，可以进入入水口温度数据显示界面，该界面主要有近五次的入水口温度数据，以及数据清理按键和返回主页面按键，点击数据清零，将会使所有温度数据清零，点击返回按键，可以返回主页面。

2.1.2 系统总体结构

本系统具体框图如图2.1所示。

串口

图2.1 系统框图

2.2控制方式的选择

2.2.1 单片机的型号选择

选择一

STC89C52单片机拥有4组共32个I/O口，并且其P1、P2、P3组I/O口都已进行准双向上拉，可以直接使用，但是P0口属于弱上拉，使用前需要先用103的排阻进行上拉。并且其拥有2个外部中断、3个16位定时器以及一组串口，这些都属于单片机的基本功能，另外其晶振频率高达12MHz，运算速度也可以。

另外其材料工艺采用高性能的互补金属氧化物半导体材料，是一个8位的单片机，并且拥有可编程的Flash。其最小系统结果也很简单。

但是光本系统使用的TFT就需要多达34个引脚。MQ2传感器为系统输入的数据为模拟值，但是STC89C52单片机没有ADC接口，还需要再设计专门的AD转换电路。RC522射频模块使用的是SPI通信，而STC89C52单片机也没有SPI外设接口，还需要模拟SPI接口进行通信使用。所以本系统不选择STC89C52单片机作为主控芯片。

选择二

STM32单片机系列，本系统主要采用STM32F103ZET6单片机作为主控，STM32单片机具有高性能、成本低以及功耗低等优点，内核基于ARM Cortex-M0。

另外STM32单片机系列的STM32F103ZET6拥有7组I/O口，每组I/O口有16个引脚，每个引脚都可以被配置为8种模式，分别为：浮空输入、下拉输入、上拉输入、模拟输入、开漏输出、开漏复用输出、推挽输出以及复用推挽输出，可谓是接口资源居多。并且STM32ZET6单片机编程使用C语言编程，开放软件又有专门的Keil软件，开放难度也不难，并且配有专门的中文技术手册，将STM32F103ZET6单片机的所有寄存器都介绍的十分清楚，进一步的降低了开放难度，可以直接通过Keil软件对STM32的相关寄存器进行编程来达到相应的功能输出。

其次，STM32系列单片机拥有72MHz的晶振频率，内存为256KB的闪存和64KB的嵌入式SRAM，封装采用LFBGA10，使系统工作更加稳定可靠。另外，STM32的工作电压仅为3.3V，低电压使用更安全。并且本系统采用的STM32F103ZET6单片机最小系统板上还有丰富的外设接口，这些外设接口大大简化了系统开发的复杂度。

本系统选用的STM32F103ZET6单片机的外设接口如下表：

表2.2 STM32F103ZET6单片机的基本外设接口

外设接口	数量
12位ADC接口	21路
12位DAC接口	2路
定时器	8个（其中TIM1和TIM8为高级定时器）
CAN接口	1个
SPI接口	3个
IIC接口	2个
IIS接口	2个
USART（串口）	3个
UART	2个
外部中断接口	144个I/O口均可作为外部中断输入端

最后之所以选择STM32单片机而不选择STC89C52单片机，一方面是因为STC89C52单片机没有足够的外设资源，另一方面，STM32单片机的价格也不比STC89C52单片机贵多少，所以STM32单片机具有更高的性价比。

故选择二。

2.2.2 气体浓度检测的选择

选择一

采用由华控兴业公司生产的气体浓度检测仪，该检测仪集成度高，自带显示以及声光报警功能，可检测可燃气体、硫化氢、氨气、氧气、一氧化碳、二氧化碳、二氧化硫、二氧化氮以及臭氧等多种气体，并且其外壳采用压铸铝防爆壳体其主要产品参数如下：

- (1) 采样方式：自然扩散
- (2) 额定功率：3W
- (3) 数据误差： $\pm 5\% F \times S$
- (4) 工作温度： $-40^{\circ}\text{C} \sim 55^{\circ}\text{C}$
- (5) 工作电压：24V DC
- (6) 工作湿度： $\leq 95\% \text{RH}$
- (7) 传输距离： ≤ 1200 米
- (8) 防爆等级：ExdIICT6

并且其操作简单，可以多传感器探头组网进行监控，其组网监控布线图如下：

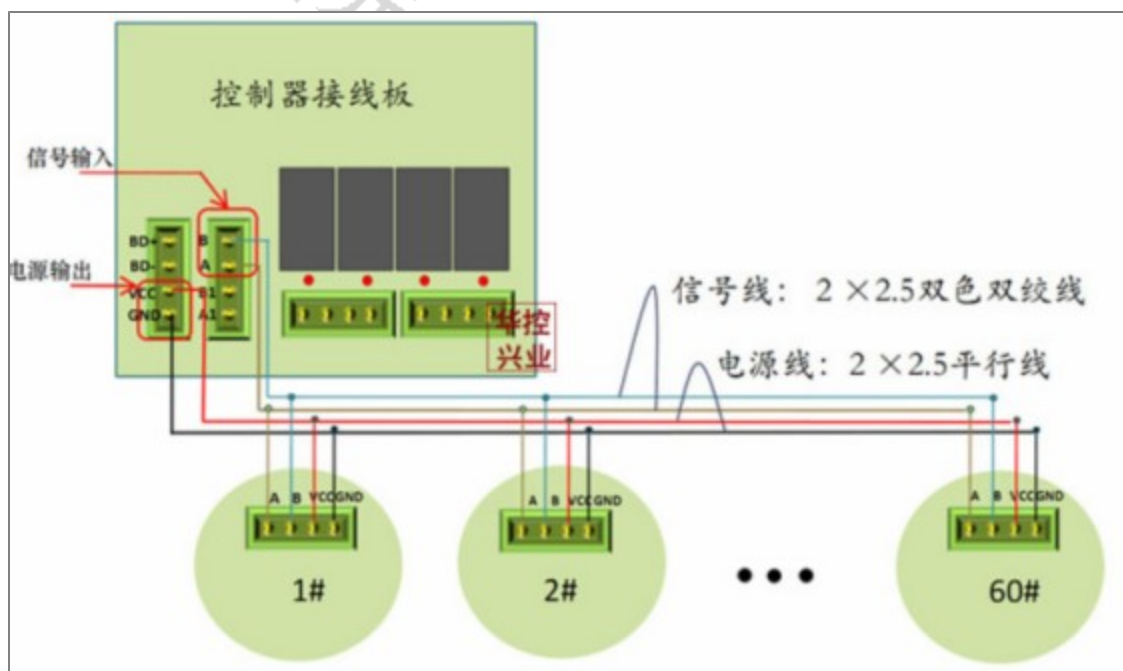


图2.3 组网监控布线图

但是由于其制作工艺和材料，使其价格昂贵，一般售价在1000元人民币左右，并且该传感器为工业级传感器，适用于高精度测量场所，本系统模拟的气体浓度检测传感器不需要那么高的精度，考虑实际性价比，故不采用。

选择二

采用由TELESKY公司生产的MQ2气体浓度检测传感器，该传感器具有探测范围广、灵敏度高、响应速度快、稳定性高、使用寿命长以及驱动电路简单等特点。并且MQ-2型烟雾传感器的材质采用了二氧化锡半导体气敏材料，该材料属于覆盖面离子式的N型半导体，利用该材料的材料特性就可以检测环境中的相关气体的浓度，具体工作原理为，MQ2中的二氧化锡材料在通电加热后会吸附空气中的氧，形成氧的负离子吸附，从而使半导体材料中的电子密度减少，导致MQ2回路电阻值增加，当MQ2的检测面与相关气体接触时，会引

起MQ2覆盖面表面导电率的变化。然后单片机通过检查回路电流值来获得这种气体浓度大小的信息。

其实物图如下：



图2. 4 MQ2实物图

其基本工作参数如下：

表2. 4 MQ2基本工作参数

工作参数名称	数据
工作温度	-20℃-100℃
工作湿度	0%RH-60%RH
工作电压	3. 3V-5V
检测气体浓度范围	300Ppm-10000Ppm
输出数据类型	模拟量

另外MQ2传感器主要输出模拟值，而STM32单片机又有21路ADC接口，可以直接将MQ2的输出引脚与STM32的ADC输入引脚相连，来测量相关气体浓度的大小。

故选择二。

2. 2. 3 显示屏的选择

选择一

采用12864液晶显示屏，12864的主要优点为其内部自带汉字字库128X64-0402B型字库，其他屏幕基本都没有字库，需要专门的字库生成软件来生成字库。并且12864的分辨率可达128×64，每个显示单元可以显示一个汉字或者2个字符。显示原理主要通过将字符的显示编码写进12864的显示缓存ARM中，并且根据显示内容的差异，12864可以显示CGROM（中文字库）、HCGRAM（ASCII码字库）及CGRAM（自定义字形）的相关内容。

但是12864字库大小是固定的，并且最多只能显示十几个汉字或字符，而且引脚多达20个，控制繁琐，性价比不高，故本系统不采用。

选择二

采用TFT显示屏，TFT式显示屏是目前各种主流显示设备的首要选择，比如笔记本电脑和台式机屏幕基本都是TFT式显示屏，该显示屏显示时是由每个液晶像素点后面的薄膜晶体管来驱动，因此TFT显示屏也属于有源液晶。

另外TFT屏具有式显示器具有高响应度、高亮度、高对比度等优点，其显示效果接近CRT式显示器。并且TFT屏的显示色度可以有65536色、16万色以及更高的1600万色，真彩显示效果极好。本系统主要采用2. 8寸的TFT显示屏。

本系统2. 8寸TFT显示屏的基本产品信息如下：

表2. 5 2. 8寸TFT显示屏基本信息

名称	介绍

分辨率	320×240
显示	16位真彩显示
输入	电阻式触摸输入

之所以选择TFT屏作为系统显示屏而不用OLED屏，是应为后者只能显示单色/双色，并且常用的OLED尺寸为0.96寸等，尺寸较小，本系统采用的是2.8寸的显示屏，尺寸更大，并且可以显示16位色的真彩图片，综合而言，TFT屏无论在尺寸上还是在功能上都比OLED好得多，所以本系统采用TFT屏作为系统显示屏。

故选择二。

2.2.4 水温检测模块的选择

选择一

采用由Sinomeasure公司制造的PT100温度传感器，该类型传感器是目前国外热能表系统的主要选择传感器，该传感器的探头由铂热电阻组成，感温效果极好，响应速度也快，其测温原理采用了高精度K型电偶温度变送器，可以将温度数据直接转换为易传输的数字信号。

另外该传感器主要由两部分组成，分别为：测温传感器以及信号转换器。其中，测温传感器主要为热偶电阻，信号转换器主要为：测量单元、信号处理单元以及信号转换单元组成。并且该传感器测温范围可达：-200℃～+850℃，应用场景十分广泛，但是由于其材料的价格比较昂贵，并且处理深度的加深，使其造价十分昂贵，一般该传感器主要用于对温度测量要求极高的场所，而本课题设计的停车场检测系统不需要这么高的精度，考虑性价比的情况下，不采用本传感器。

选择二

由TELESKY公司生产的DS18B20 T0-92型温度传感器，DS18B20是最常用的数字温度传感器，其输出的是数字信号，具有体积小、硬件开放难度低、抗干扰能力强、精度高的特点。

另外其读写时序和测温原理与DS1820相同，区别在于得到的数据精度不同，并且温度转换的延迟时间也从2S降低到750ms，响应更快。

DS18B20的基本产品参数如下表所示：

表2.6 DS18B20基本产品参数

名称	描述
接口方式	单线通信
测温范围	-55℃～+125℃
工作电压	3.0V～5.5V
输出	数字串行输出
精度	9～12位输出
存储器	高速暂存RAM和可电擦除RAM
ROM	64位光刻ROM

由表易知，DS18B20传感器虽然在精度和测温范围上没有TP-100传感器高，但是其价格便宜，操作简单，并且精度也满足本课题的基本要求，故本系统决定采用DS18B20温度传感器来作为测温传感器。

故选择二。

2.2.6 水流量计模块选择

选择一

采用由上海驰控公司生产的电磁流量计来检测水流量，该传感器具有抗干扰能力强、瞬时流量快以及多单位切换的特点，其集成度高

，表头抗干扰能力强，精度可达千分之五。

另外其主要测量介质为：污水废水、强酸强碱、导电液体以及冷凝水等，其信号输出为4mA-20mA的电流，通信采用485通信方式通信，工作电压为24V。

其电路接线图如下图：

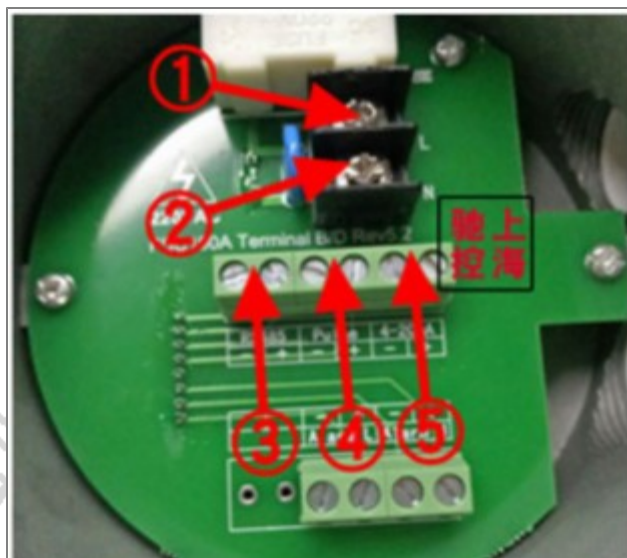


图2.7 电磁流量计接线图

由于工作电压要24V直流电，本系统模拟电路提供的电压仅为5V直流电，并且其集成度过高，可以独立工作，不符合本系统的模拟需求，所以不采用该传感器。

选择二

采用优信电子公司生产的3Y水流量计，该流量计内部设置有霍尔开关涡轮流量计，并且该产品具有体积小、密封严等特点，其适用范围很广，比如：应用于热水器、刷卡机、自动售水机以及其他流量计设备。

其主要工作参数如下：

表2.8 电磁流量计主要工作参数

工作参数	数据
工作电压	5V
工作电流	15mA
工作温度	$\leq 80^{\circ}\text{C}$
工作湿度	35%RH-90%RH
耐压	$\leq 1.75\text{Mpa}$
频率	$F=7.5 \times Q$ （单位：L/min）
流量监测范围	1-30L/min

故选择二

2.3本章小结

本章首先介绍了实行方案，然后介绍了各种控制方式的选择，包括单片机、水流量计、气体浓度检测传感器、显示屏以及水温检测传感器的比较以及选择。另外还介绍了相关传感器模块的工作参数数据等。

硬件系统的设计

3.1 模块电路的设计

3.1.1 STM32F103ZET6单片机核心系统电路的设计

在本次设计中，我所采用的主控模块是STM32ZET6的最小系统板，其中STM32则是熟知的意法半导体公司所设计的主控芯片主要对其性能的提高、降低成本、功耗的降低采用嵌入式应用开发设计的一款芯片。这款芯片的内存有64KB至256KB的闪存以及64KB的嵌入式SRAM。它的新系列采用三种封装方式包括LQFP64、LQFP100和LFBGA100，虽然采用不同的封装方式，但其引脚的封装却是一致的，我们与STM32平台的设计理念相结合，通过对产品的选择可以使其功能、存储器、性能以及引脚数等各方面得到重新的优化与完善，通过最少的改变硬件来满足各方面灵活的应用需求。

一、STM32单片机的主要特征：

72MHz的主频；

32位的机器字长；

512KB内部Flash；

可用J-LINK、ST-LINK以及串口进行程序的烧录和调试。

STM32单片机的基本工作参数：

表3.1 STM32单片机基本工作参数

名称	介绍
程序存储器容量	64KB
RAM容量	20KB
工作电压	2V~3.6V
数据转换器	12位ADC、12位DAC
封装/外壳	144-LQFP
控制器	DMA

三、STM32单片机相关引脚说明：

STM32单片机开发板，共引出了144个IO口，STM32内部提供了8个定时器，其中有2个高级定时器，4个通用定时器，开发者可以使用这些定时器对外部脉冲进行计数以及可以通过配置预分频寄存器和配置自动重装载寄存器来输出PWM等等。另外STM32ZET6内部还集成了12路的ADC和DAC，开发者在对外部模拟输入和对外输出模拟值时不再需要设计AD转换电路和DA转换路，大大减少了系统开发的繁杂程度。同时STM32ZET6还已经集成了多种通信接口。

熟知的CAN通信接口，设计此通信接口主要是通过最小的CPU负荷采取最高的处理效率处理收到的大量报文。我们可以用软件进行配置以及SPI通信接口的改变来实现优先级特效，SPI接口的全称为串行外设接口，此接口可以允许STM32和它外面的设备通过半双工以及全双工和同步串行等方式进行通信，另外SPI可以被配置为主模式，并且可以给外部的设备提供通信时钟等等各方面的需求。

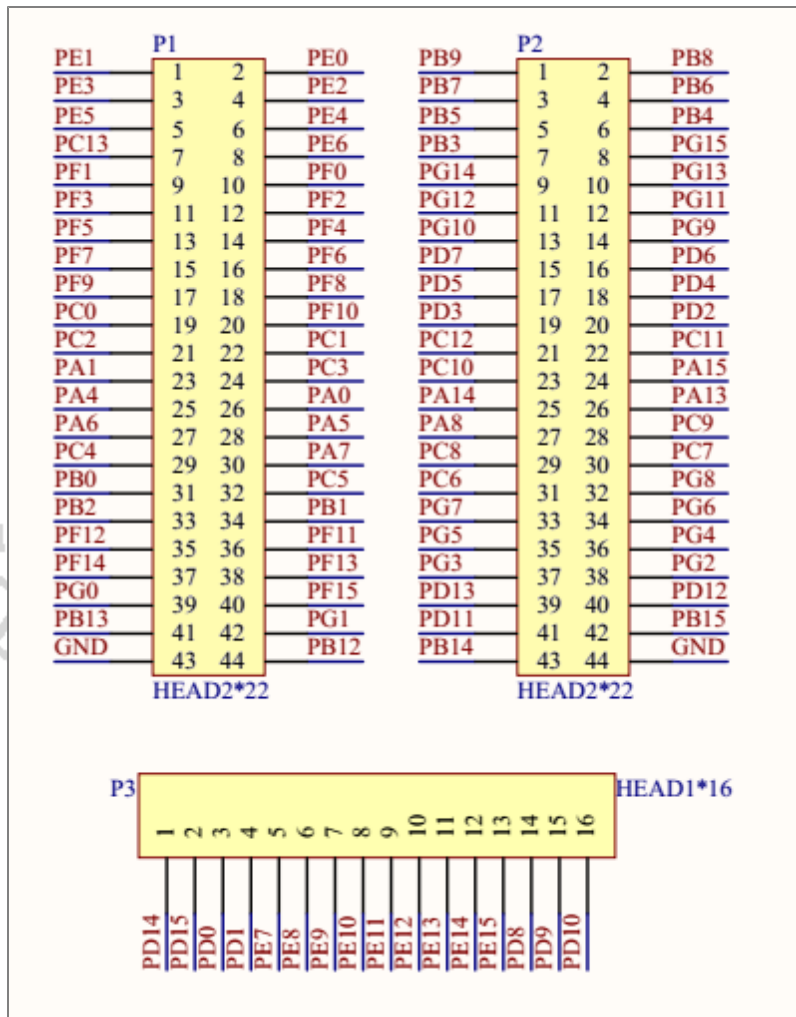


图3.2 STM32单片机引脚图

四、STM32单片机开发板说明：

STM32单片机开发板电路的组成部分就由时钟晶振电路、复位电路、电源电路以及通信电路组成。在这几部分的带动下，单片机即可正常工作。

1：STM32时钟晶振电路

我们这次设计的系统采用的是拥有3种不同时钟源的STM32开发板，这个开发板可以用来驱动时钟，其中包括HIS振荡器时钟、HSE振荡器时钟和PLL时钟，如果我们在用到PLL时钟时采用HIS振荡器作为它的输入，那么这个系统所能得到的最大的时钟频率则为36MHz。在系统复位以后，这里会选用HIS振荡器来作为此系统的系统时钟。这是不管时钟源是被直接应用为系统时钟或者是通过PLL间接得到，两种情况均不能停止。然而只有目标时钟源一切准备就绪的情况下即在PLL稳定时或者启动稳定之后产生延时，这是才会发生从一个时钟源到另一个时钟源切换。那么如果所选择的时钟源没有准备好的情况下，那么系统时钟是不会发生切换的。除非时钟源一切准备完成，切换才会发生。

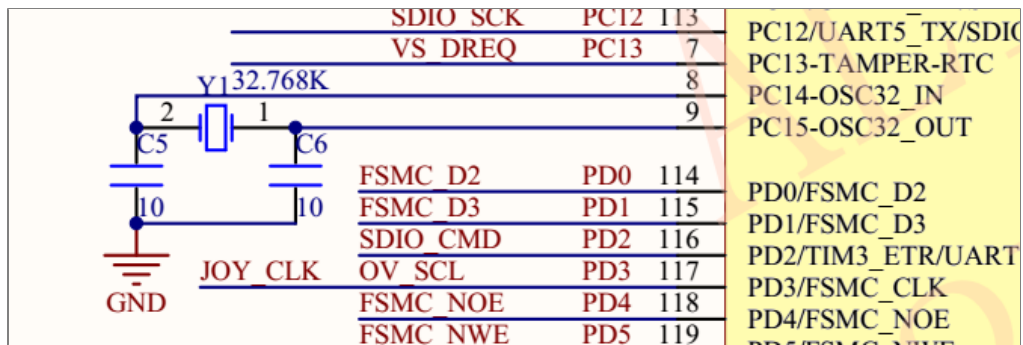


图3.3 STM32单片机时钟晶振电路

2: STM32的复位电路

这里介绍的STM32开发板，它的复位功能有三种，包括系统复位、电源复位和最后一种后备域复位。这里面的系统复位可以把除了时钟控制寄存器CSR里面的复位标志以及备份区域里面的寄存器以外，所有的寄存器均为它们的复位数值。

当以下事件中的一件发生时，产生一个系统复位：

NRST引脚上的低电平(外部复位)

窗口看门狗计数终止(WWDG复位)

独立看门狗计数终止(IWDG复位)

软件复位(SW复位)

低功耗管理复位

当遇到上电和掉电两种情况时系统都会自动产生电源的复位，还有一种进入电源复位的情况就是从代价模式中返回，这些都会引起电源复位清除备份区以外的任何寄存器。

然而所说的备份域只能够影响到备份区域，这里面复位是由软件进行的。

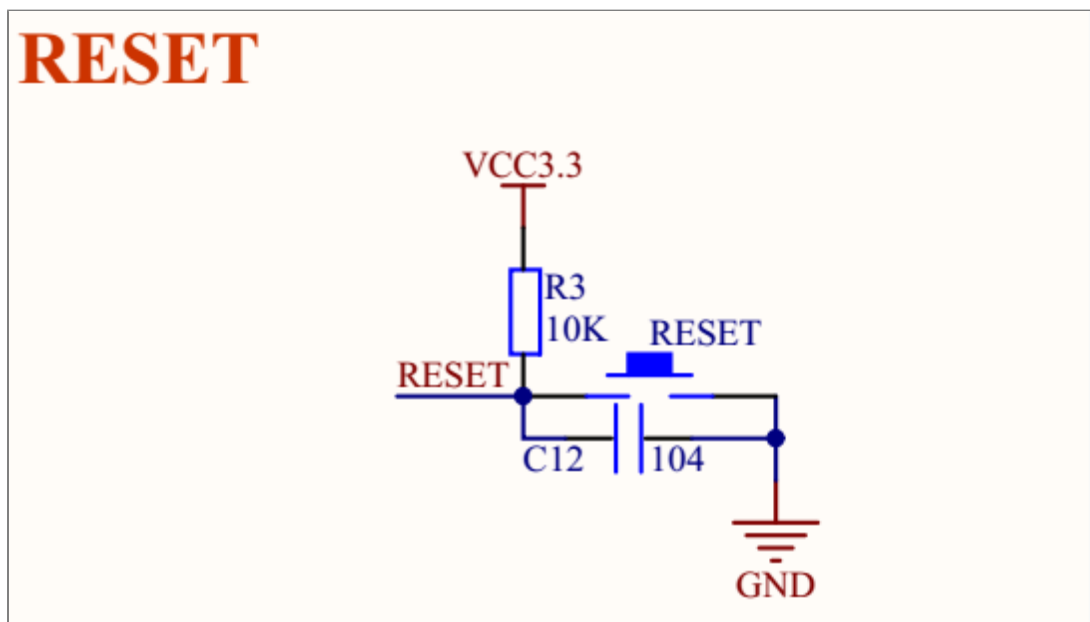


图3.4 STM32单片机复位电路

3: 通信接口电路

STM32配备的主要通信方式及接口有：串口通信、RS485通信、CAN通信、SPI通信、JTAG接口以及RS232接口。

上文提到的串口通信可以最终转换成需要的连续的串行数据流，它是通过来自CPU的串行数据字符进行转换最终发送出去的，与此同时，CPU器件所需要的串行数据字符可以由串行数据流交换得到。这种能够完成所描述功能的电路，我们一般称它叫做串行接口电路。

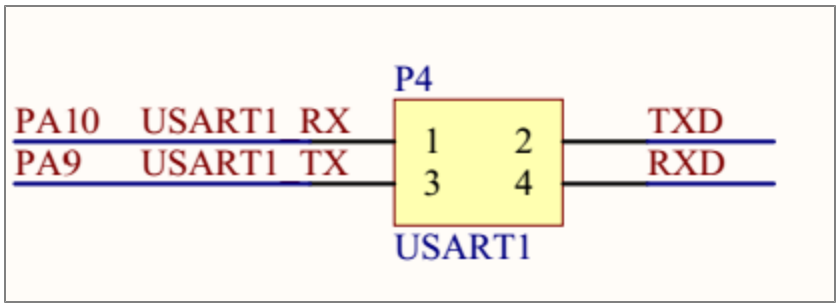


图3.5 STM32单片机串口1接口图

RS485通信由电气特征的标准可以把它定义为是在平衡数字系统中的一种驱动器以及接收器，这个标准是由权威的行业协会所定义的。所有在使用这种标准的数字通信网络都能够在距离远的和噪声大的情况下进行高效的数据传输。在使用RS-485时，能够使网络变得更加便宜，增加了更多的可能性。拥有两线制和四线制两种连接方式的RS485中，现在很少采用四线制，因为受到制约只能完成点对点的通信方式，最多采用的二线制则不同，通过总线式的拓扑结构，同样在一条线上可以同时连接32个节点。在RS485中，通常都采用主从的方式进行通信，也就是说可以由一个主机带动多个从机。

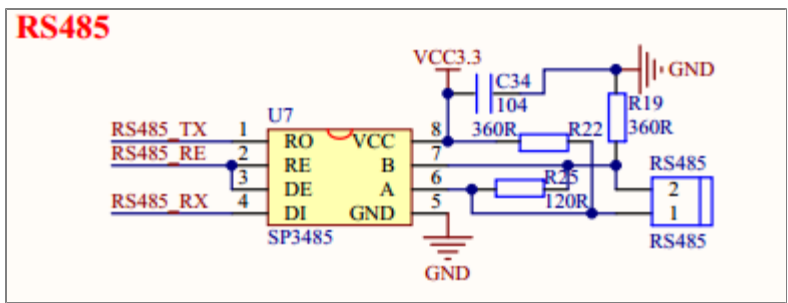


图3.6 STM32单片机485通信接口图

CAN 是Controller Area Network 的缩写（以下称为CAN），是ISO国际标准化的串行通信协议。在我们所熟知的各个产业中尤其是汽车产业更为明显，我们不断地开发各种电子控制系统对汽车所需的安全舒适方便以及低公害还有要求最多的低成本进行优化。因为不管在哪方面对于系统的通信方面的要求基本相同，多种情况中总线数目在不断增加，与此同时，为了满足减少总线数目的要求，对通信数据的速度以及通信信息的总量增加的需求，在1986年公开了对于汽车的CAN通信协议要求。以后，对于不断地需求，协议也越来越标准，出现了标准协议，以及协议的标准化。

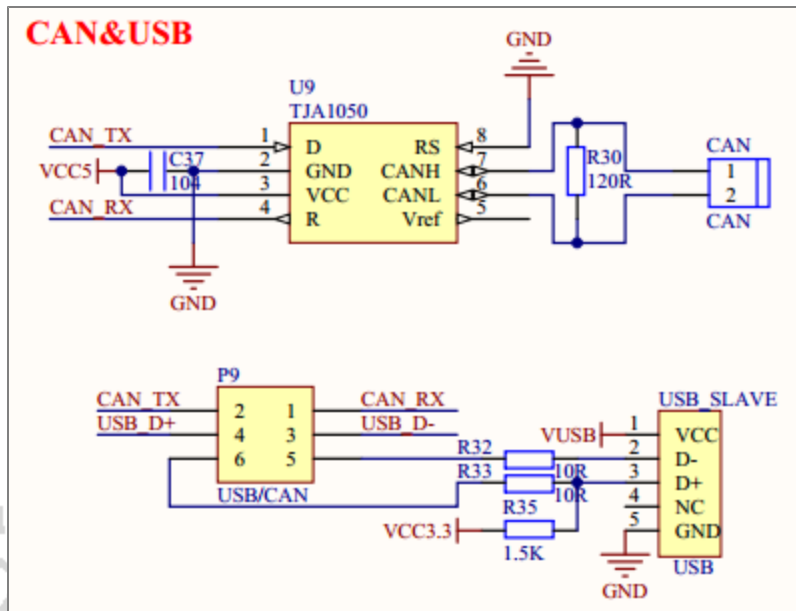


图3.7 STM32单片机CAN通信接口图

SPI是串行外设接口 (Serial Peripheral Interface) 的缩写。所说的SPI则是基于速度要求高以及全双工的一种同步的通信总线。它的原理也很容易理解，它的工作方式是我们上面提到过的主从工作方式，可以有一个设备也可以有多个设备，但是其中至少包括四根线，然而在单向传输时三根也是可以实现的。所有的SPI的共同点就是他们拥有数据的传输以及输出，还有时钟和片选。片选主要看主芯片是否可以控制从芯片，这样就是说一条总线上连接多个设备的可能性就增加了。

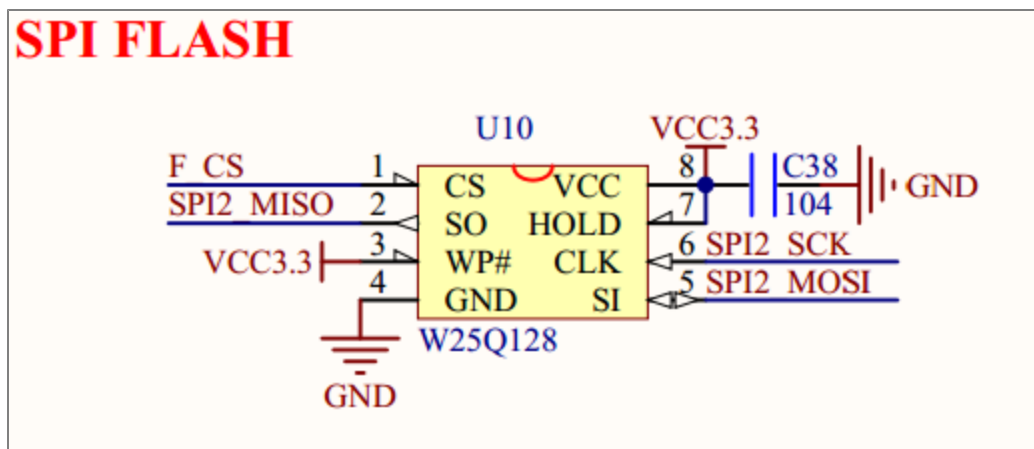


图3.8 STM32单片机SPI通信接口图

需要内部芯片测试时JTAG就被采用。现阶段我们知道的大多数的高级器件都拥有JTAG协议，比如熟知的DSP、FPGA器件等等。一般情况JTAG拥有四个接口线。它的编程方式也很特别是在线的，传统的则一般是对芯片进行预编程最后再装到板子上，简化后则是完全相反的先装到板子再进行编程，从而使得我们的速度得到了提高。DSP内部的所有部件都可以由他进行编程。它的内部还有一个状态机被叫做控制器。可以改变状态机的状态，从而输入数据以及指令实现功能。

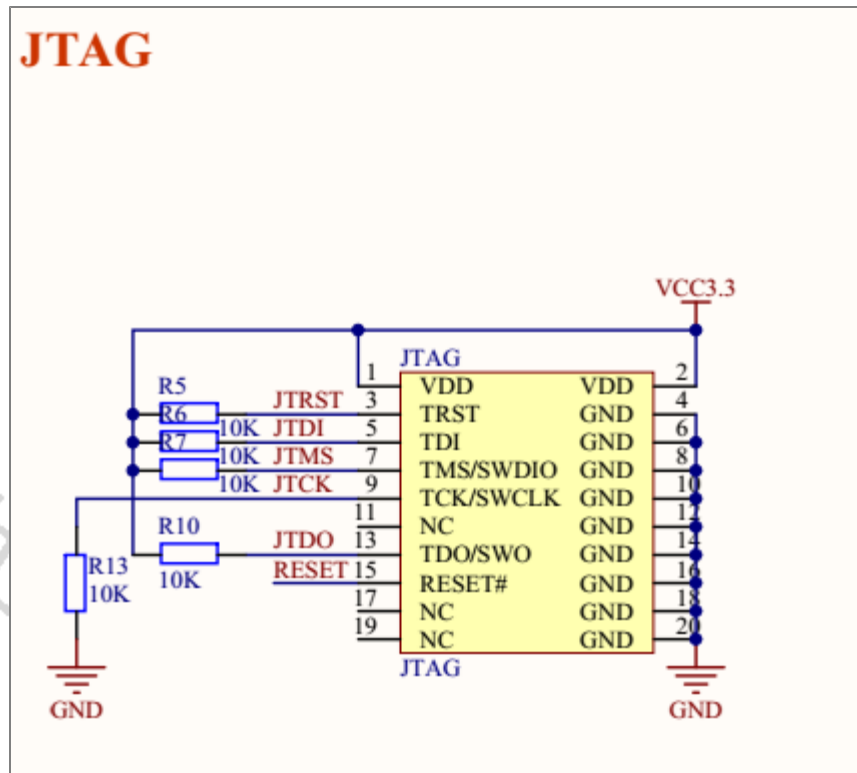


图3.9 STM32单片机JTAG接口图

RS-232接口原始编号全称是EIA-RS-232（简称232，RS232）。计算机的串行接口普遍采用它进行应用。可以对各种设备的信号等进行传送。RS-232是当下最流行的通信接口。但也因为他的出现时间比较早，所以还是存在许多不足，下面列举了四点不足：

- （1）此接口的传输信号较高，很容易造成接口的损坏，此接口每个信号线的电压都可以看做负逻辑。
- （2）他拥有很低的传输速度，在通常的异步传输，比特率则为20Kbps；所以我们在开发板中采用的波特率都只能是19200。
- （3）在传输形式上采用的是用一个信号线和一个返回线来构成，这样的传输形式是容易产生共模干扰的，所以抗噪声的能力是很弱的。
- （4）在传输距离方面，最大也不会超过50英尺，实际应用时最多也只能在15米左右，所以传输的距离也是比较有限的。

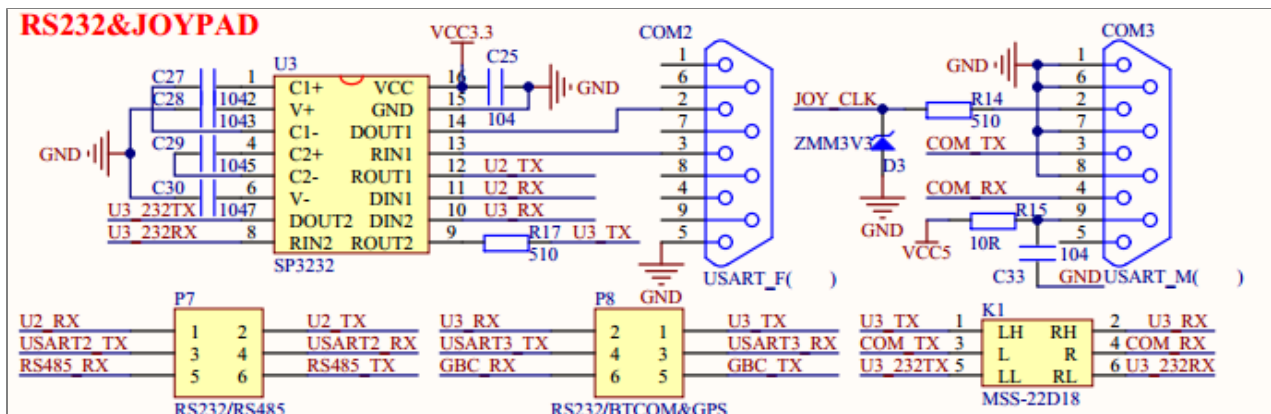


图3.10 STM32单片机RS232接口图

3.1.2 5V电源电路设计

本系统选择5V直流电源作为总电源，为整个系统供电，电路简单、稳定。DC为电源的DC插座，LED为红色LED灯，作为系统是否有电的

指示灯，电阻为1K电阻，起到限流作用，保护LED灯，以防电流过大，烧坏LED灯。SW为自锁开关，开关按下后，红灯亮，此时系统电源5V直流输出。开关再次按下后，红灯灭，此时系统电源5V电源输出。



图3.11 5v电源电路原理图

3.1.3 TFT显示屏电路设计

此次设计时所采用的显示屏是2.8寸的LCD_TFT屏，就是通常所说的液晶显示屏。它的英文全称是Thin Film Transistor-Liquid CrystalDisplay。他和其他的简单助阵是不同的，他可以很有效的克服在没有选通时所带来的干扰，因为他在整个显示屏上小到每个像素点上都设置了一个晶体管，这样就使得屏幕的静态和线性扫描没有联系，所以图像的质量会很好。

该TFT屏的驱动芯片采用ILI9314芯片，ILI9341 液晶控制器自带显存，其显存总大小为 172800（240*320*18/8），即 18 位模式（26万色）下的显存量。在 16 位模式下，ILI9341 采用 RGB565 格式存储颜色数据，此时 ILI9341的 18 位数据线 与 MCU 的 16 位数据线以及 LCD GRAM 的对应关系如下表所示：

表 3.12 ILI9341数据线 与 MCU数据线以及GRAM的对应关系

9341总线	MCU数据（16位）	LCD GRAM（16位）
D17	D15	R4
D16	D14	R3
D15	D13	R2
D14	D12	R1
D13	D11	R0
D12	NC	NC
D11	D10	G5
D10	D9	G4
D9	D8	G3
D8	D7	G2
D7	D6	G1
D6	D5	G0
D5	D4	B4

D4	D3	B3
D3	D2	B2
D2	D1	B1
D1	D0	B0
D0	NC	NC

从表中可以看出，ILI9341 在 16 位模式下面，数据线有用的是：D17~D13 和 D11~D1，D0和 D12 没有用到，实际上在我们 LCD 模块里面，ILI9341 的 D0 和 D12 压根就没有引出来，这样，ILI9341 的 D17~D13 和 D11~D1 对应 MCU 的 D15~D0。这样 MCU 的 16 位数据，最低 5 位代表蓝色，中间 6 位为绿色，最高 5 位为红色。数值越大，表示该颜色越深。另外，特别注意 ILI9341 所有的指令都是 8 位的（高 8 位无效），且参数除了读写 GRAM 的时候是 16 位，其他操作参数，都是 8 位的。

在本系统采用的STM32F103ZET6最小系统板上，有专门的TFT插孔，TFT与STM32连接示意图如下：

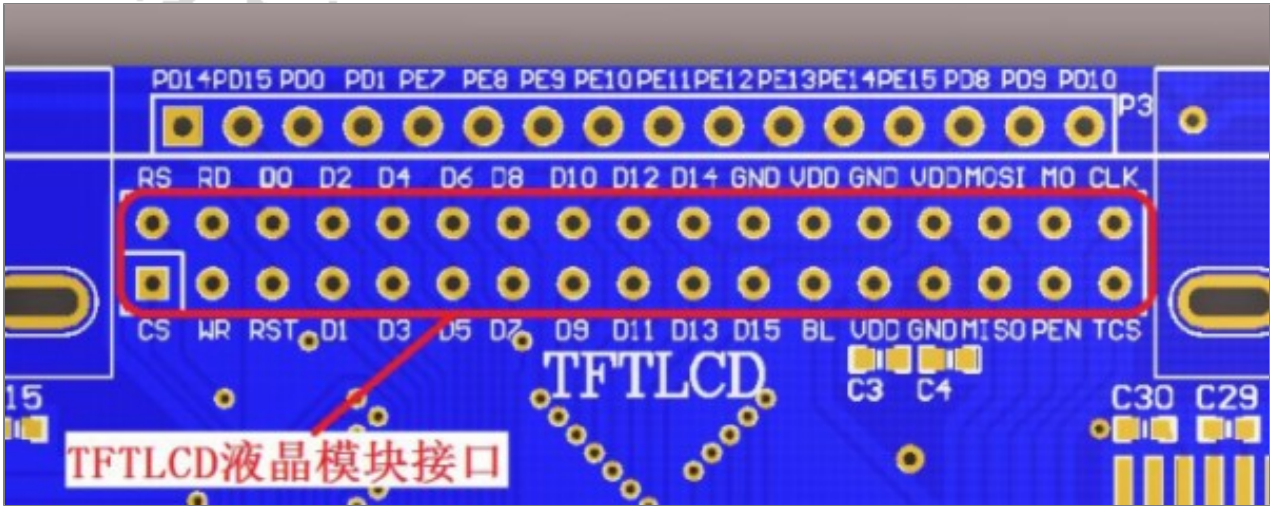


图3.13 TFT与STM32的接口图

其中TFT的模块原理图如下：

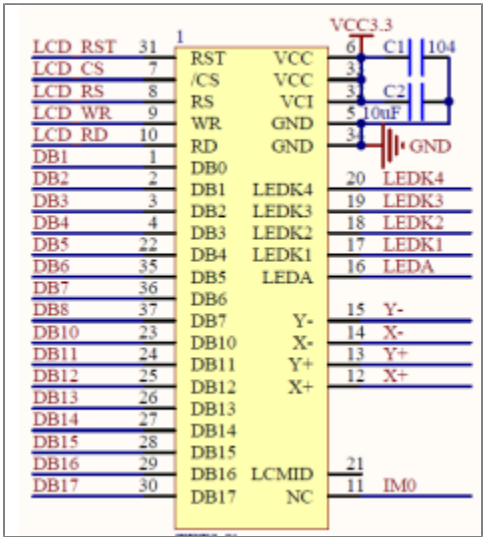


图3.14 TFT的原理图

3.1.4 MQ2气体浓度电路设计

本系统采用的气体浓度检测传感器为MQ2，主要应用于热能表附近的易燃气体检测，保障安全。

其基本工作参数如下表所示：

符号	参数名称	技术条件	备注
Vc	回路电压	$\leq 15V$	AC或DC
Vh	加热电压	$5.0V \pm 0.2V$	AC或DC
R1	负载电阻	可调	
Rh	加热电阻	$31\Omega \pm 3\Omega$	室温条件下
Ph	加热功耗	$\leq 900mW$	

图3.15 MQ2基本工作参数

MQ2输出为模拟值，本系统采用的STM32单片机拥有21路ADC接口，所以可以直接将MQ2的输出端与STM32单片机相连，其工作原理图如下：

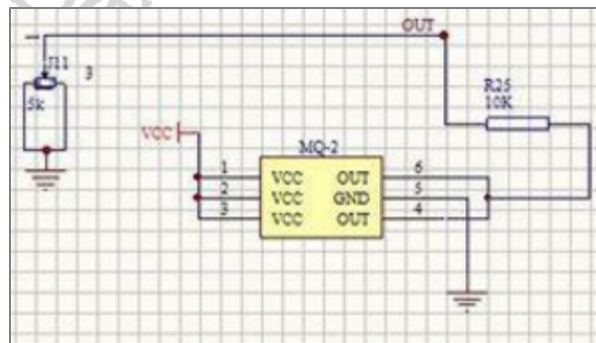


图3.16 MQ2原理图

3.1.5 RC522登录管理电路设计

使用的RC522是我们熟知的高集成系列芯片中的非接触式通信。它采用了调制和解调的概念，使非接触式和协议完全体现在上面。它包括了发送部分和可以驱动的读写器的天线通信，根本不需要其他的电路就能实现所需的功能。上面提到的调制和解调就被应用在了接受端，用来处理信号的兼容问题。它其实还可以运行高速的加密算法，可以用来验证一些产品。

RC522的主要特性为：

高集成度的调制解调电路

支持SPI通信接口

灵活的中断模式

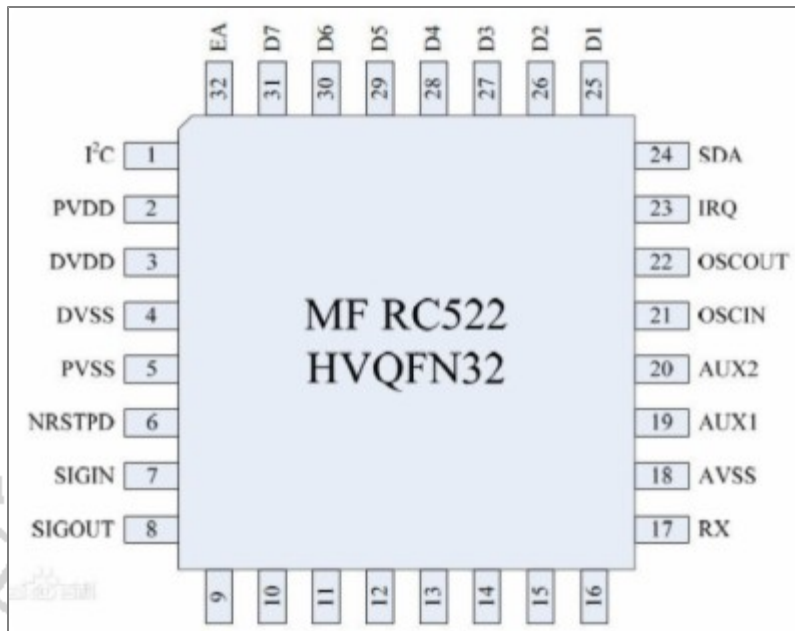
可编程的定时器

内置温度传感器

内置振荡器

工作温度：-30℃-85℃

RC522芯片引脚图如下：



3.17 RC522引脚图

RC522采用SPI通信，而本系统采用的STM32单片机的基本外设接口中就有SPI接口，使用本系统可以直接连接STM32单片机的SPI外设通信接口，通过软件的编程来控制RC522。

RC522电路原理图如下：

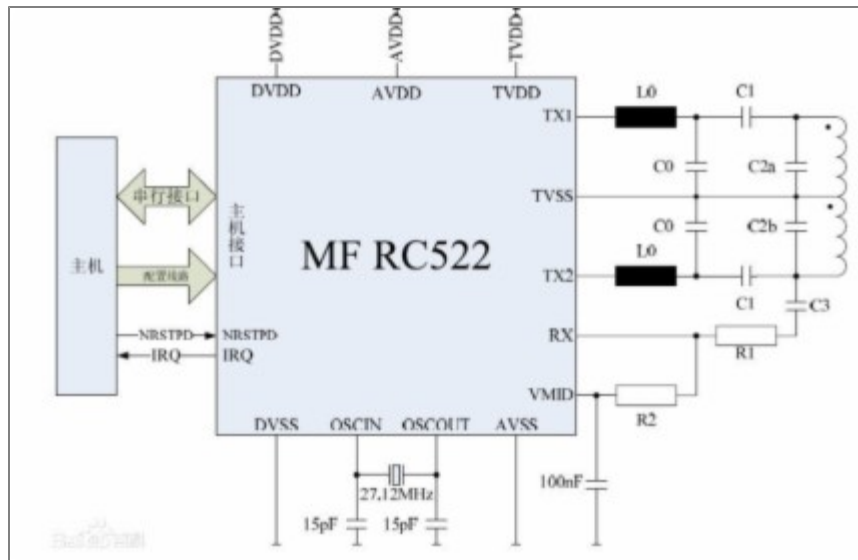


图3.18 RC522电路原理图

3.1.6 水流量计电路设计

这个系统是用于测量水流量的传感器为Water Flow Sensor，塑料阀体、水流转子组件和霍尔传感器主要是由这几部分组成的。这个系统可以应用在测量水的流量，平时最常见的就是热水器，可以安装在它的进水端。它的工作原理也很简单，可以通过水流的流量以及速度的变化对器进行测量，根据相应的输出脉冲信号，由控制器进行感应，从而判断水流量的大小多少，最终实现调节的功能。

Water Flow Sensor与国外同类 产品相比具有以下特点：

一、外观轻巧灵便，体积小，便于安装。

- 二、叶轮内部镶有不锈钢珠，永久耐磨。
 - 三、密封圈采用上、下受力的结构不漏水。
 - 四、霍尔元件采用德国进口，并且用灌封胶封装防止进水，永不老化。
 - 五、所有原材料均有符合ROHS检测标准。
 - 六、适合各种控制器和开发板，如：Arduino控制器、STC单片机、AVR单片机等。
- 其基本参数如下表所示：

表3.19 水流量计基本参数

工作参数名称	技术指标
进水口内径	8.5mm
出水口外径	20.12mm
输出波形	方波
工作电压	5V
工作电流	15mA
负载能力	≤10mA
工作温度范围	≤80℃
工作湿度范围	35%RH-90%RH

3.1.7 蜂鸣器报警电路设计

在所选择的蜂鸣器上我们私用直流电，采用直流电在很多电路以及设备中都是很常见的。我们也可以看到，蜂鸣器模块采用的也是有源的，这里电压设置为5v，这里的电路则使用三极管来驱动。如果蜂鸣器遇到低电平时，就会出现报警现象。同样道理，当遇到高电平时报警就不会出现。我们若想控制蜂鸣器是否报警就可以根据刚分析的对他的引脚电平进行控制。



图3.20 蜂鸣器报警电路原理图

3.2本章小结

本章首先介绍了系统的可行性以及系统的框图。为系统的总设计提出了方案，然后接着介绍了各个硬件模块的功能以及他们的特点。分析了他们的引脚图以及工作原理图。

软件设计

4.1编程语言选择

本系统设计采用的编程语言为C语言，编程软件为Keil软件，之所以采用C语言是因为C语言程序易读，并且对处理指令和存储结构要

求不高。另外本系统的复杂度不高。采用C语言足以满足本系统的逻辑结构需求，

4.2 软件流程图

软件设计部分采用的是结构化程序的方法，该方法条理结构易懂。任何程序都是由顺序、选择、循环三种方式构成，所以如何规划这三种方式尤为重要。它具有自顶向下的原则，即从整体到细节，慢慢将问题具体化，结构化程序采用模块形式，一步步将问题模块化，具体问题具体分析。

系统工作时，先是运行初始化子程序，将各个模块的初始化函数进行调用，来实现系统的总体初始化过程，初始化之后，开始运行RC522门禁登录函数，显示刷卡登录界面，直到有刷卡的行为，才进入信息显示主界面，信息显示主页面有入水口温度数据、出水口温度数据、退出登录以及水泵控制等按钮，点击不同的按钮，将进入不同的显示页面。其中主页面工作流程图如下图：

图4.1 主页面显示函数工作流程图

数据信息管理界面主要为出、入水口温度数据显示界面，该页面工作流程图如下：

图4.2 数据信息管理界面流程图

4.3 程序烧录及调试

1. 首先，安装好JLINK驱动安装过后将仿真器的 USB 口插入电脑的 USB，然后打开电脑的设备管理器，在设备管理器查看是否能检测到JLINK。如图4.3示：

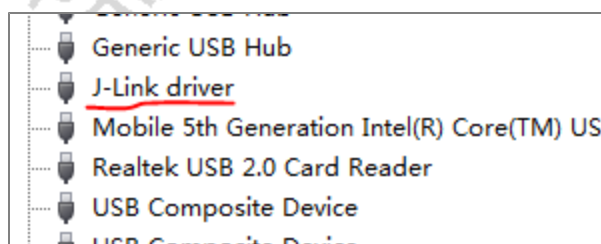


图4.3 设备管理器监测JLINK

2. 然后打开Keil软件，对下载方式进行配置，界面如图4.4所示

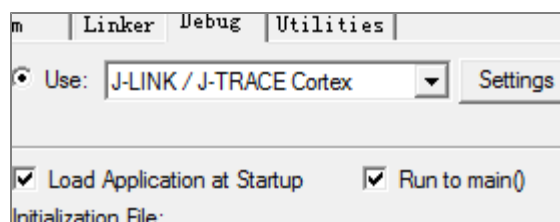


图4.4 下载配置界面

继续配置Keil，将C/C++中的Define配置为STM32F10X_HD，如图4.5所示。

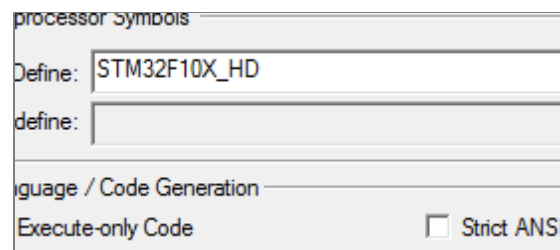


图4.5 选择单片机型号图

4. 配置单片机内部flash型号，如图4.6所示。

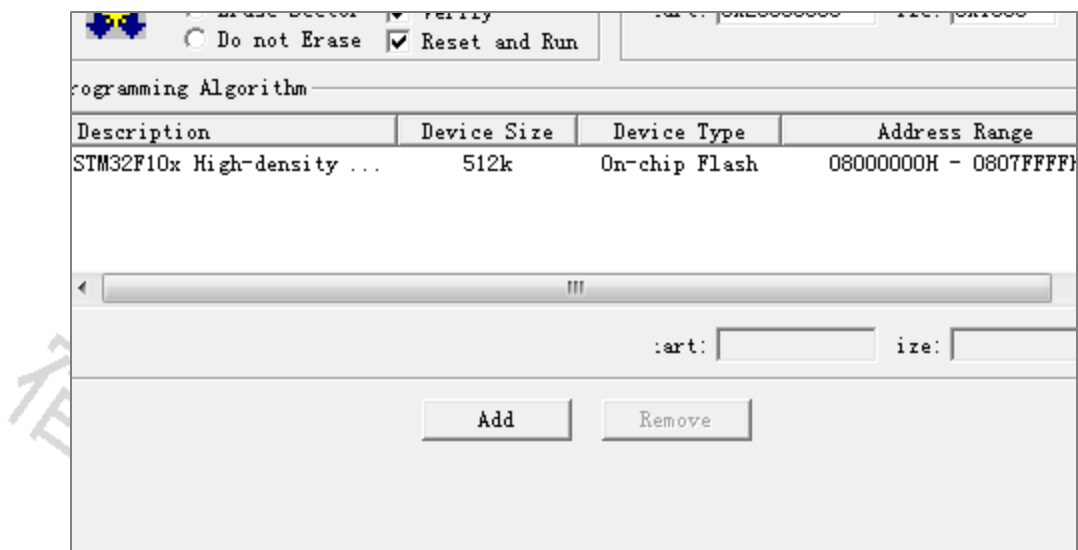


图4.6 单片机内部flash配置界面

完成以上设置后，开始将各只文件导入工程系统中，并将根目录也导入系统中，具体方法如下图所示：

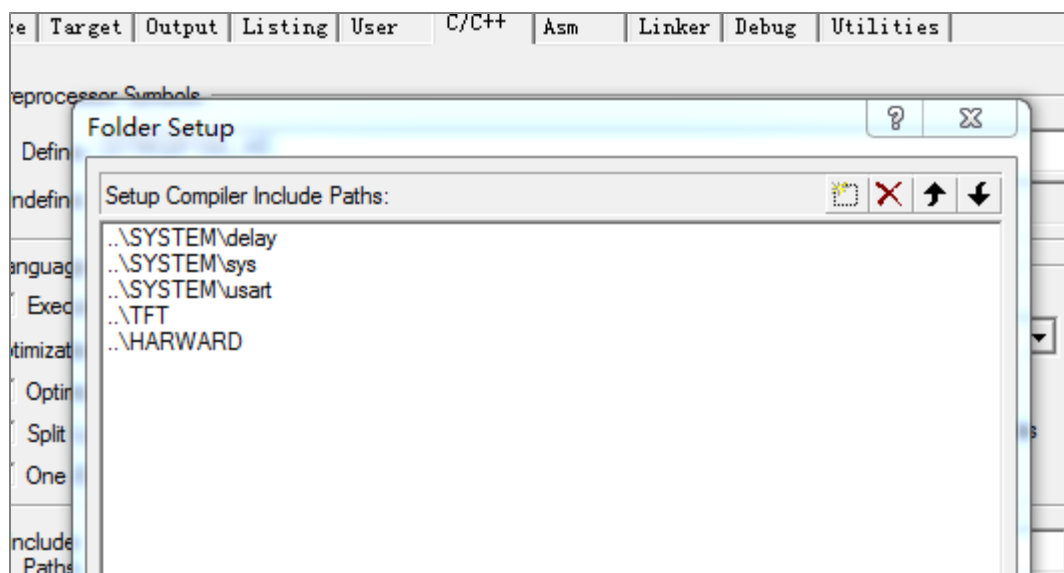


图4.7 子文件根目录导入工程系统配置界面

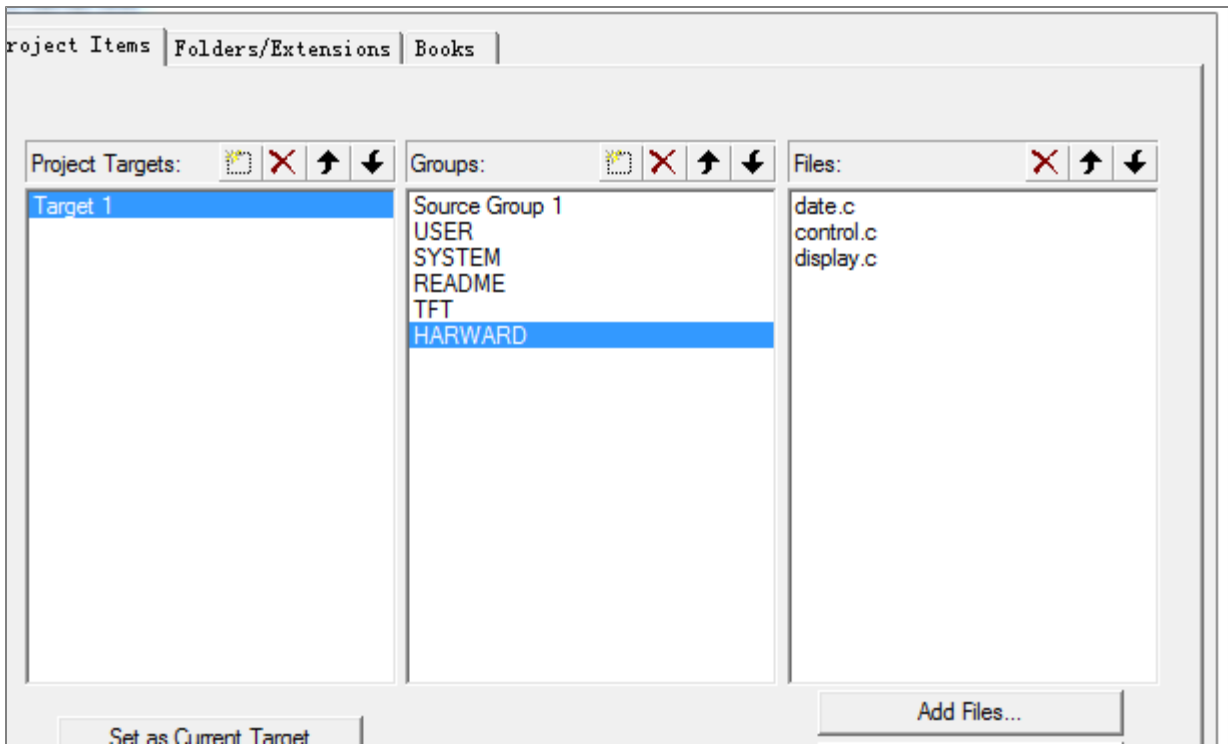


图4.8 子文件文件导入工程系统配置界面

6. 点击下载后，系统进入以下下载状态，下载成功之后显示如下图4.6所示。



图4.9 程序下载按钮

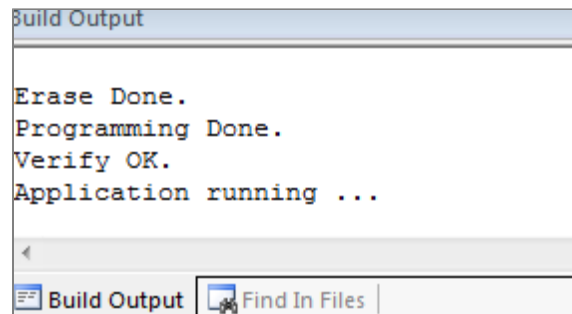


图4.10 程序下载成功显示界面

4.4 部分程序代码

4.4.1 主控程序部分代码

本部分代码主要为各种函数的初始化调用以及逻辑结构的设计代码。

```
#include "sys.h"
#include "usart.h"
#include "delay.h"
#include "lcd.h"
```

```
#include "touch.h"
#include "w25q128.h"
#include "text.h"
#include "ds18b20.h"
#include "ds18b202.h"
#include "memory.h"
#include "rc522.h"
#include "timer.h"
#include "adc.h"
#include "led.h"
#include "control.h"
#include "at24c02.h"
u8 tim=0; //水泵开启标准量
u8 tempdate1,date1,tempdate2,date2,tempdate3,date3,tempdate4,date4,tempdate5,date5;
u8 tempoutdate1,outdate1,tempoutdate2,outdate2,tempoutdate3,outdate3,tempoutdate4,outdate4,tempoutdate5,outdate5;
u16 SL;
u8 page=1;
int main(void)
{
    Stm32_Clock_Init(9); //系统时钟设置
    delay_init(72); //延时初始化
    uart_init(72,115200); //串口初始化为115200
    LCD_Init();
    W25QXX_Init(); //初始化W25Q128 和SPI2
    tp_dev.init(); //触摸屏初始化
    InitRc522();
    Adc_Init(); //ADC初始化
    Adc2_Init();
    LED_Init();
    tempdate1=AT24CXX_ReadOneByte(60); //读取近5次数据
    date1=AT24CXX_ReadOneByte(61);
    tempdate2=AT24CXX_ReadOneByte(62);
    date2=AT24CXX_ReadOneByte(63);
    tempdate3=AT24CXX_ReadOneByte(64);
    date3=AT24CXX_ReadOneByte(65);
    tempdate4=AT24CXX_ReadOneByte(66);
    date4=AT24CXX_ReadOneByte(67);
    tempdate5=AT24CXX_ReadOneByte(68);
    date5=AT24CXX_ReadOneByte(69);
```

```
tempoutdate1=AT24CXX_ReadOneByte(70); //读取近5次数据
outdate1=AT24CXX_ReadOneByte(71);
tempoutdate2=AT24CXX_ReadOneByte(72);
outdate2=AT24CXX_ReadOneByte(73);
tempoutdate3=AT24CXX_ReadOneByte(74);
outdate3=AT24CXX_ReadOneByte(75);
tempoutdate4=AT24CXX_ReadOneByte(76);
outdate4=AT24CXX_ReadOneByte(77);
tempoutdate5=AT24CXX_ReadOneByte(78);
outdate5=AT24CXX_ReadOneByte(79);
init_display();
TIM2_Int_Init(1999, 7199); //10Khz的计数频率, 200ms
TIM3_Int(65535);
tim=0;
SB=1;
while(1)
{
control();
}}
```

4.4.2 显示屏部分程序代码

本部分代码主要为显示信息的设计。

```
#include "control.h"
#include "lcd.h"
#include "touch.h"
#include "w25q128.h"
#include "text.h"
#include "ds18b20.h"
#include "ds18b202.h"
#include "rc522.h"
#include "timer.h"
#include "adc.h"
#include "led.h"
#include "delay.h"
#include "memory.h"
u16 adcx1; //烟雾
u16 adcx2; //液位
u16 LL; //水流量
extern u16 SL;
extern u8 nn;
```

```
extern u8 page;

unsigned char CT[2]; //卡类型

unsigned char SN[4]; //卡号

u8 KEY[6]={0xff, 0xff, 0xff, 0xff, 0xff, 0xff};

void init_display(void) //初始化显示 并对出水口DS18B20进行延迟检测
{
while(font_init()) //检查字库
{;}

POINT_COLOR=BLUE;
Show_Str(24, 5, 240, 24, "热能水表检测系统", 24, 0);
POINT_COLOR=BLACK;
Show_Str(48, 35, 240, 24, "检测系统", 24, 0);
delay_ms(3000);
LCD_Fill(5, 65, 240, 24, WHITE);
POINT_COLOR=BLUE;
Show_Str(24, 5, 240, 24, "热能水表检测系统", 24, 0);
POINT_COLOR=BLACK;
POINT_COLOR=BLACK;
Show_Str(5, 65, 240, 24, "TFT屏触摸功能正常", 24, 0);
Show_Str(5, 95, 240, 24, "入水口DS18B20正常", 24, 0);
Show_Str(5, 125, 240, 24, "出水口DS18B20正常", 24, 0);
Show_Str(5, 155, 240, 24, "水泵控制电路正常", 24, 0);
Show_Str(5, 185, 240, 24, "蜂鸣器功能正常", 24, 0);
Show_Str(5, 215, 240, 24, "RFID读写卡功能正常", 24, 0);
Show_Str(5, 245, 240, 24, "烟雾传感器功能正常", 24, 0);
Show_Str(5, 245, 240, 24, "信号灯功能正常", 24, 0);
delay_ms(5000);
LCD_Clear(WHITE);
}

void main_display(void) //主页面 page=0
{
date_read();
POINT_COLOR=BLUE;
if(tim==0)
{
Show_Str(125, 155, 96, 24, "水泵关闭", 24, 0);
}
else if(tim==1)
{

```

```
Show_Str(125, 155, 96, 24, "水泵开启", 24, 0);  
}  
POINT_COLOR=BLUE;  
Show_Str(10, 245, 240, 24, "管理者ID信息: ", 24, 0);  
POINT_COLOR=BLACK;  
LCD_ShowNum(10, 270, SN[0], 3, 16);  
LCD_ShowNum(58, 270, SN[1], 3, 16);  
LCD_ShowNum(106, 270, SN[2], 3, 16);  
LCD_ShowNum(154, 270, SN[3], 3, 16);  
Show_Str(10, 295, 96, 24, "退出登录", 24, 0);  
}  
void date_read(void)  
{  
    adcx1=Get_Adc_Average(ADC_CH1, 10);  
    // adcx2=Get_Adc_Average(ADC_CH2, 10);  
    LL=(SL*40)/16;  
}  
void RFID_read(void)  
{  
    unsigned char status;  
    status = PcdRequest(PICC_REQALL, CT); /*尋卡*/  
    if(status==MI_OK) //尋卡成功  
    {  
        status=MI_ERR;  
        status = PcdAnticoll(SN); /*防冲撞*/  
    }  
    if(status==MI_OK)  
    {  
        status=MI_ERR;  
        status =PcdSelect(SN);  
    }  
    if(status==MI_OK) //選卡成功  
    {  
        status=MI_ERR;  
        status =PcdAuthState(0x61, 0x06, KEY, SN);  
    }  
    if(status==MI_OK) //驗證成功  
    {  
        status=MI_ERR;
```

```

page=0; //如果登录成功 进入主页面
LCD_Clear(WHITE);
delay_ms(10);
}
}

void DL_display(void) //page=1
{
POINT_COLOR=BLUE;
Show_Str(60,100,240,24,"请刷卡登录",24,0);
POINT_COLOR=BLACK;
RFID_read();
LED_RED=0;
LED_YELLOW=0;
LED_GREEN=0;
}

```

4.4.3 传感器驱动部分程序代码

本部分代码主要为各传感器的控制函数。

```

#include "adc.h"
#include "delay.h"

//初始化ADC1
//这里我们仅以规则通道为例
//我们默认仅开启通道1

void Adc_Init(void)
{
//先初始化IO口
RCC->APB2ENR|=1<<2; //使能PORTA口时钟
GPIOA->CRL&=0xFFFFF0F; //PA1 analog输入
RCC->APB2ENR|=1<<9; //ADC1时钟使能
RCC->APB2RSTR|=1<<9; //ADC1复位
RCC->APB2RSTR&=~(1<<9); //复位结束
RCC->CFGR&=~(3<<14); //分频因子清零
//SYSCLK/DIV2=12M ADC时钟设置为12M,ADC最大时钟不能超过14M!
//否则将导致ADC准确度下降!
RCC->CFGR|=2<<14;
ADC1->CR1&=0XF0FFFF; //工作模式清零
ADC1->CR1|=0<<16; //独立工作模式
ADC1->CR1&=~(1<<8); //非扫描模式
ADC1->CR2&=~(1<<1); //单次转换模式
ADC1->CR2&=~(7<<17);

```



```

ADC1->CR2|=7<<17; //软件控制转换

ADC1->CR2|=1<<20; //使用用外部触发(SWSTART)!!! 必须使用一个事件来触发

ADC1->CR2&=~(1<<11); //右对齐

ADC1->SQR1&=~(0XF<<20);

ADC1->SQR1|=0<<20; //1个转换在规则序列中 也就是只转换规则序列1

//设置通道1的采样时间

ADC1->SMPR2&=~(3*1); //通道1采样时间清空

ADC1->SMPR2|=7<<(3*1); //通道1 239.5周期, 提高采样时间可以提高精确度

ADC1->CR2|=1<<0; //开启AD转换器

ADC1->CR2|=1<<3; //使能复位校准

while(ADC1->CR2&1<<3); //等待校准结束

//该位由软件设置并由硬件清除。在校准寄存器被初始化后该位将被清除。

ADC1->CR2|=1<<2; //开启AD校准

while(ADC1->CR2&1<<2); //等待校准结束

//该位由软件设置以开始校准, 并在校准结束时由硬件清除

}

void Adc2_Init(void)
{
//先初始化IO口

RCC->APB2ENR|=1<<2; //使能PORTA口时钟

GPIOA->CRL&=0xFFFF00FF; //PA1 analog输入

RCC->APB2ENR|=1<<9; //ADC1时钟使能

RCC->APB2RSTR|=1<<9; //ADC1复位

RCC->APB2RSTR&=~(1<<9); //复位结束

RCC->CFGR&=~(3<<14); //分频因子清零

//SYSCLK/DIV2=12M ADC时钟设置为12M, ADC最大时钟不能超过14M!

//否则将导致ADC准确度下降!

RCC->CFGR|=2<<14;

ADC1->CR1&=0XF0FFFF; //工作模式清零

ADC1->CR1|=0<<16; //独立工作模式

ADC1->CR1&=~(1<<8); //非扫描模式

ADC1->CR2&=~(1<<1); //单次转换模式

ADC1->CR2&=~(7<<17);

ADC1->CR2|=7<<17; //软件控制转换

ADC1->CR2|=1<<20; //使用用外部触发(SWSTART)!!! 必须使用一个事件来触发

ADC1->CR2&=~(1<<11); //右对齐

ADC1->SQR1&=~(0XF<<20);

ADC1->SQR1|=0<<20; //1个转换在规则序列中 也就是只转换规则序列1

//设置通道1的采样时间

```

```

ADC1->SMPR2&=~(6*1); //通道1采样时间清空

ADC1->SMPR2|=7<<(6*1); //通道1 239.5周期, 提高采样时间可以提高精确度

ADC1->CR2|=1<<0; //开启AD转换器
ADC1->CR2|=1<<3; //使能复位校准
while(ADC1->CR2&1<<3); //等待校准结束
//该位由软件设置并由硬件清除。在校准寄存器被初始化后该位将被清除。
ADC1->CR2|=1<<2; //开启AD校准
while(ADC1->CR2&1<<2); //等待校准结束
//该位由软件设置以开始校准, 并在校准结束时由硬件清除
}_
//获得ADC1某个通道的值
//ch:通道值 0~16
//返回值:转换结果
u16 Get_Adc(u8 ch)
{
//设置转换序列
ADC1->SQR3&=0xFFFFFE0;//规则序列1 通道ch
ADC1->SQR3|=ch;
ADC1->CR2|=1<<22; //启动规则转换通道
while(!(ADC1->SR&1<<1)); //等待转换结束
return ADC1->DR; //返回adc值
}.
//获取通道ch的转换值, 取times次, 然后平均
//ch:通道编号
//times:获取次数
//返回值:通道ch的times次转换结果平均值
u16 Get_Adc_Average(u8 ch,u8 times)
{
u32 temp_val=0;
u8 t;
for(t=0;t<times;t++)
{
temp_val+=Get_Adc(ch);
delay_ms(5);
}
return temp_val/times;
}

```

4.4本章小结

本章节主要介绍了软件编程语言的选择为Keil软件, 设计了软件实行的方案, 并且详细介绍了程序的烧录过程。以及如何对产品信息

进行相应的配置和如何将各个子文件导入到工程系统之中，并在最后给出了程序的部分代码。

第五章 系统运行结果

5.1 系统测试情况分析

本次测试一共测试了几种情况：

RC522管理员登录系统，当无人刷卡时，一直显示登录界面，直到有人刷卡成功，才进入主页面显示。

主页面显示主要分为数据管理界面进入按钮、水泵控制按钮、环境气体浓度数据显示、水流量数据显示已经管理员ID信息。通过点击屏幕上的各种按钮，可以实现相应的功能。

数据管理界面主要显示有近五次的温度数据、返回主页面显示按钮已经数据清零按钮，通过点击返回按钮，可以返回主页面。点击数据清零按钮，可以将数据清零。

（1）经过测试，当接通电源时，系统先自检，以保证各个模块都能正常工作，然后进入登录界面，等待刷卡登录，如图5.1所示。

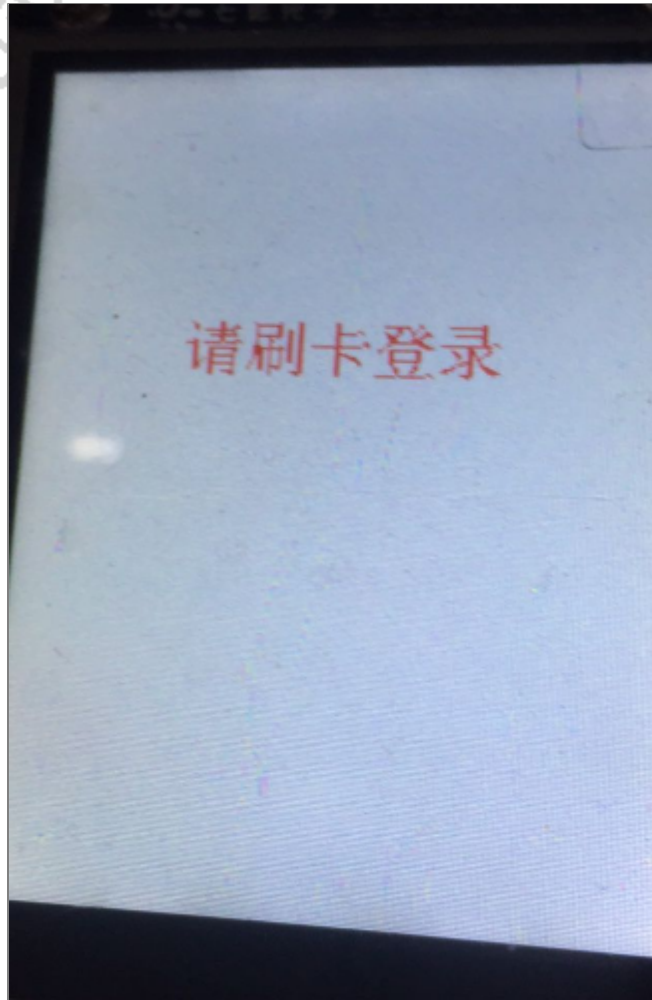


图5.1 接通电源图

（2）当用门禁卡成功刷卡后，将进入主页面显示，如图5.2。

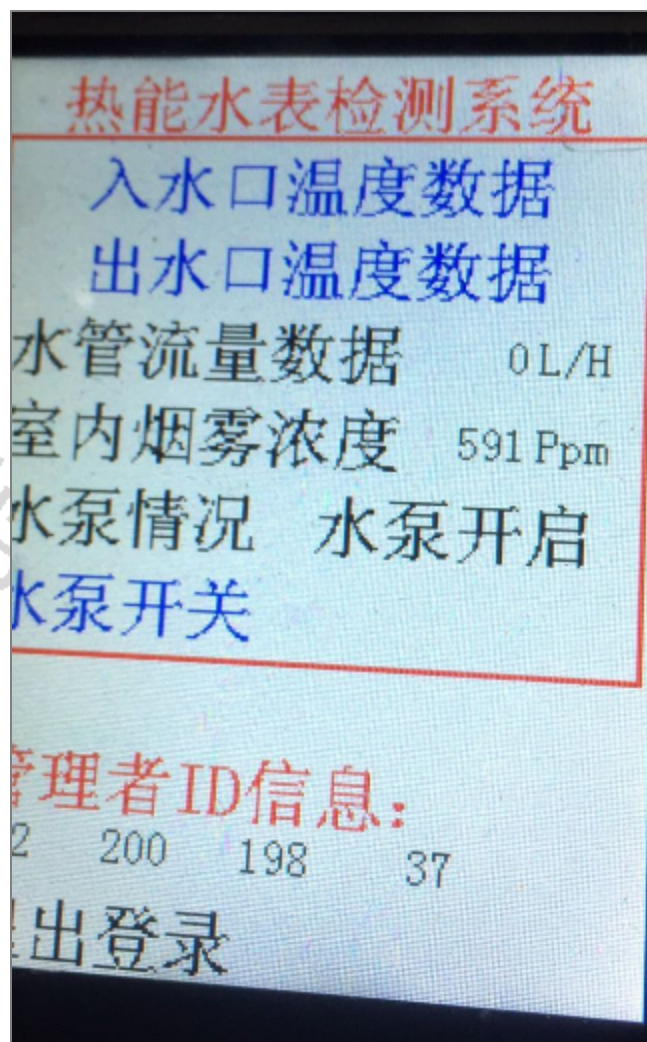


图5.2 主页面显示图

(3) 通过点击水泵开关按钮，可以关闭或者开启水泵，如图5.3:

图5.3 水泵开关控制图

(4) 点击入水口温度数据按钮，可以进入入水口温度数据管理界面，如图5.4。

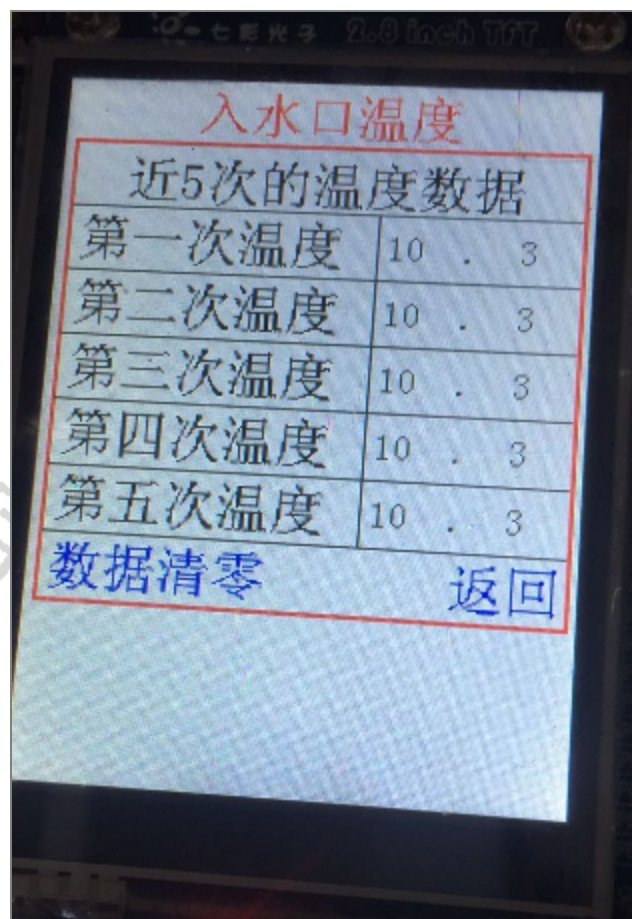


图5.4 入水口温度数据管理界面

(5) 点击返回按钮，可以返回主页面。如图5.5所示。

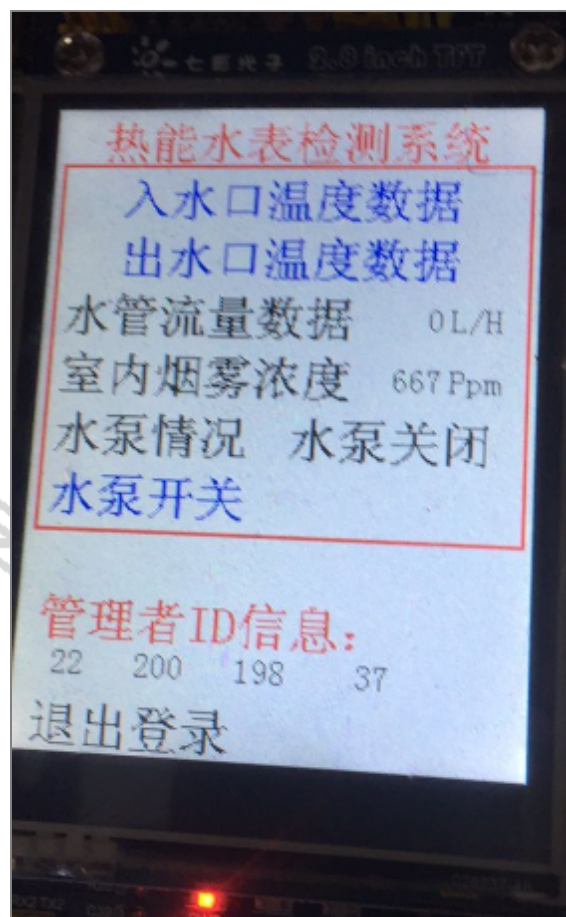


图5.5 返回主页面显示图

(6) 此时点击出水口温度数据，可以进入出水口温度数据显示界面。如图5.6所示。

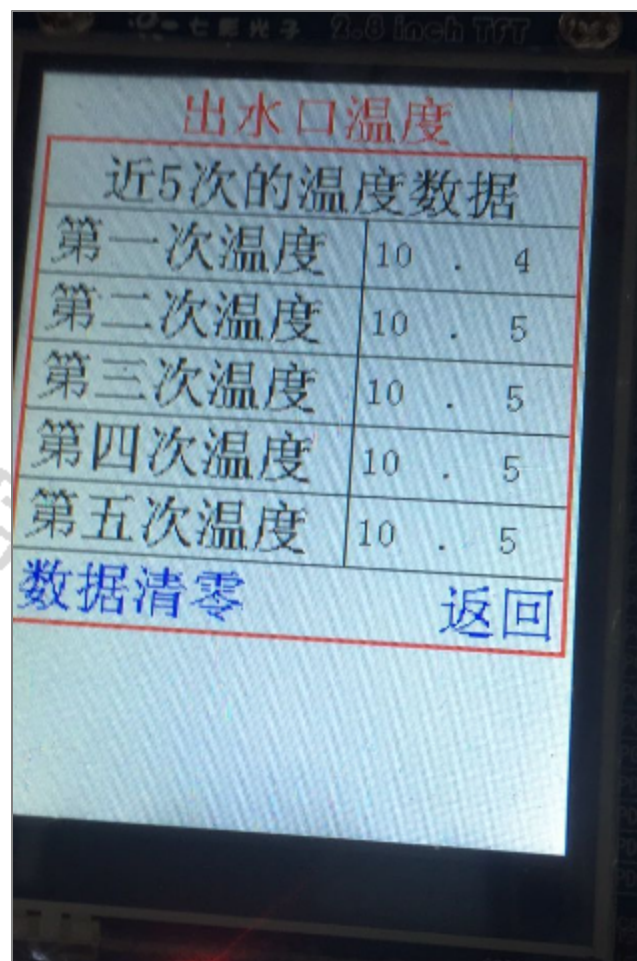


图5.6 出水口温度数据

(7) 点击数据清零按钮，可以将数据全部清零，如图5.7所示。

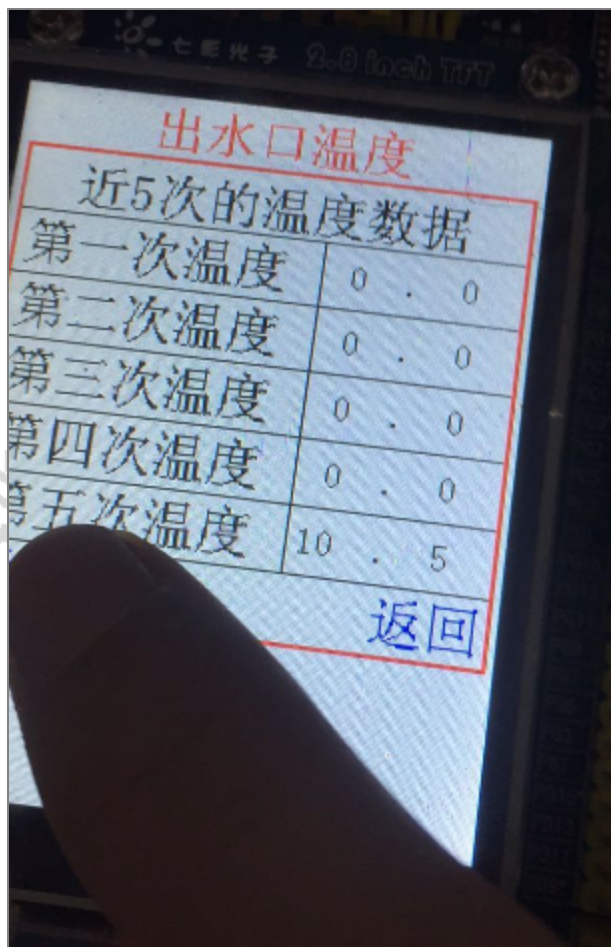


图5.7 出水口温度数据清零

5.2本章小结

本章节主要详细介绍了系统的各个显示界面的情况，以及在各个界面的基本操作过程，测试过程中有时会发生刷卡刷不上的情况，据查阅资料，总结出以下三点可能：

电流不够，导致RC522工作不正常。

短时间刷同一张卡会使RC522内部防冲撞处理函数出错。

门禁卡有问题，不能每次都触发RC522的检测中断。

但出问题后，将系统进行复位处理，又能正常使用，具体是什么原因，还需要大量的调试以及查阅资料才能找到。

• 说明：

相似片段中“综合”包括：

《中文主要报纸全文数据库》 《中国专利特色数据库》 《中国主要会议论文特色数据库》 《港澳台文献资源》
《图书资源》 《维普优先出版论文全文数据库》 《年鉴资源》 《古籍文献资源》 《IPUB原创作品》

• 声明：

报告编号系送检论文检测报告在本系统中的唯一编号。

本报告为维普论文检测系统算法自动生成，仅对您所选择比对资源范围内检验结果负责，仅供参考。

客服热线：400-607-5550 | 客服QQ：4006075550 | 客服邮箱：vpcs@cqvip.com

唯一官方网站：<http://vpcs.cqvip.com>



关注微信公众号