



电工电子实验教学中心

嵌入式系统课程设计

设计报告

设计题目：智能家居温度控制器

姓名：孙天一

学号：17211316

专业：电子科学与技术

姓名：高成鑫

学号：17211270

专业：电子科学与技术

选课教师：赵翔

2020 年 11 月 1 日

目 录

1	设计任务要求.....	4
1.1	基础部分（必做，60 分）	4
1.2	提高部分（选做，60 分）	4
2	系统设计方案.....	5
2.1	系统顶层设计（15 分）	5
2.2	小组人员分工（5 分）	6
2.3	分工接口定义（5 分）	6
3	功能模块设计方案（30 分）	7
3.1	模块 1：时间显示模块.....	7
3.1.1	模块功能要求.....	7
3.1.2	模块设计思路.....	7
3.1.3	CubeMX 设计方案.....	7
3.2	模块 2：温度湿度采集显示模块.....	8
3.2.3	CubeMX 设计方案.....	9
3.3	模块 3：设置温度和工作模式功能模块.....	10
3.3.1	模块功能要求.....	10
3.4	模块 4（定时器输出 PWM 控制各模式的功率模块）	13
3.4.1	模块功能要求.....	13
3.4.2	模块设计思路.....	13
3.4.3	CubeMX 设计方案.....	14
3.4.4	模块内部结构.....	14
3.4.5	模块软件流程图.....	14
3.5	模块 5（红外控制模块）	15
3.5.1	模块功能要求.....	15
3.5.2	模块设计思路.....	15
3.5.3	CubeMX 设计方案.....	15
3.5.4	模块内部结构.....	16
3.5.5	模块软件流程图.....	16
3.6	模块 6（预约开关机和达到预设温度开关机）	16
3.6.1	模块功能要求.....	16
3.6.2	模块设计思路.....	16

3.6.3 CubeMX 设计方案.....	17
3.6.4 模块内部结构.....	17
3.6.5 模块软件流程图.....	17
4 系统设计中问题及解决（10 分）	17
4.1 问题 1（每人至少一项）	17
4.2 问题 2.....	18
4.3 问题 3.....	19
5 系统使用说明.....	19
5.1 系统外观及接口说明（5 分）	19
5.2 系统操作使用说明（15 分）	19
5.3 技术指标（5 分）	20
5.4 其他（可选）	22
6 总结.....	22
6.1 设计结果（5 分）	22
6.2 心得体会（5 分）	22
6.3 意见建议.....	23

1 设计任务要求

使用本课程提供的 STM32F746 开发平台和配套电路模块设计并制作一个智能家居温度控制器，具有以下功能：

1. 测量并显示当前室内环境的温度和湿度。
2. 显示日期、时间。
3. 红外遥控功能，可以遥控本控制器开关机、调整空调温度、风速和工作模式。
4. 屏幕保护功能，可以以屏幕保护的形式显示存储在 TF 卡中的一组照片。
5. 可以通过 USB 接口修改 TF 卡中保存的照片。
6. 预约功能，可以预约定时开关机，预设温度开关机等功能。
7. 网络控制功能，可以通过以太网实现远程控制。

1.1 基础部分（必做，60 分）

（1）温度测量与显示（10 分）

系统可以测量室内温度并在屏幕上显示。

（2）日期、时间显示功能（10 分）

在屏幕上显示当前日期、时间。

（3）空调设置功能（10 分）

系统可以通过触摸屏设置空调的温度、风速、工作模式，并在屏幕上显示当前的设置。

温度可以在 18~30℃之间调节，模式可配置为“制冷”、“制热”和“通风”，风速可配置为“高”、“中”、“低”三档。

（4）空调控制功能（15 分）

在功能 2 的基础上增加空调变频控制功能，使用一个蓝色 LED 灯和一个红色 LED 灯表示空调制冷功率和制热功率和风速。LED 灯的亮度表示变频压缩机转速，室内温度与设置温度温差越大，LED 亮度越高，表示制冷/制热功率越大。

使用一个黄色 LED 灯表示风速，LED 灯亮度越高表示风速越大。

（5）红外遥控开关机功能（15 分）

系统可以使用红外遥控器控制开启和关闭，按下遥控器开关可以切换系统是开关状态。

1.2 提高部分（选做，60 分）

（1）温湿度测量功能（5 分）

系统可以测量室内温度和湿度并在屏幕上显示。

（2）全功能红外遥控（15 分）

系统可以解码红外遥控器的输出，可以通过红外遥控器设置温度、风速和工作模式。

（3）预约功能（10 分）

预约定时开关机，系统可以在指定时间按预设模式开始工作。预设温度开关机，系统

可以在室内温度超出指定范围时自动开始工作。

(4) 屏幕保护功能 (10 分)

可以显示一副图片作为屏保, 触摸屏或红外无操作 20 秒后系统进入屏幕保护模式, 屏幕显示存储在 FLASH 或 TF 卡中的一组照片。触摸屏或红外有操作时自动退出屏幕保护

(5) 可以通过 USB 修改屏幕保护图片 (10 分)

USB 接口可以设置为大容量存储设备模式 (Mass Storage Device), 可以在 PC 机上通过 USB 修改 FLASH 或 TF 卡保存的屏幕保护图片。

(6) 网络控制功能 (10 分)

可以通过以太网实现远程控制, 可以使用调试软件发数据包, 网页配置、APP 配置等方式, 形式不限。

1.3 注意事项

(1) 不允许使用除课程提供的开发板和模块之外的任何模块或电路。

(2) 红外遥控器统一使用宿舍空调遥控器, 制作调试时注意防止相互干扰。

(3) 系统界面上应显示所有组员的姓名学号。

(4) 如使用 TF 卡保存图片数据, TF 卡自备。

2 系统设计方案

2.1 系统顶层设计 (15 分)

说明: 分析设计任务, 提出设计方案, 给出完整的系统分析和结构框图。

设计任务: 利用嵌入式系统设计一个可以测量室内环境温湿度, 控制新风系统, 家用空调的运转, 实现对室内温度的智能控制。对于这一任务, 老师进行了难度上的划分, 分为基础和提高。

温度的单独测量是采用模拟温度感应器通过 ad 转换得到所测的温度的值, 而温湿度的一起测量则是采用数字温湿度感应器, 使用 DHT11, 通过 GPIO 口传输测量的数据。

时间的显示是利用开发板本身带的 RTC 进行时间计时, 并利用 WM_TIMER 函数进行刷新, 实现时间的不断增加, 之后又在此基础上添加了 time.h 函数库里面的函数从电脑中获取当前时间, 将值赋给 RTC, 进而实现时间的实时显示。

空调设置功能分为两部分, 一部分是屏幕显示的 LTDC 函数, 利用工程所自带 STemWin 工具, 构建出所需要的界面和功能, 并在其中对界面上不同的按钮、文本显示进行功能编写; 另外一部分是空调模式的切换, 设置出三种模式, 分别用 PWM 来控制三个灯的亮度来代表变频控制功能, 其中蓝色和红色为设置温度与室内温度差值来决定。

红外遥控部分是首先对于红外信号进行接收和解码, 分析出红外各部分代码所代表的含义和控制的功能, 对解码结果进行分析之后, 对于不同的按键实现不同的功能, 实现通过红外遥控设置温度等参数。

预约关开机部分包括定时开关和预设温度开关，通过预设阈值实现对开关机功能。

结构框图：

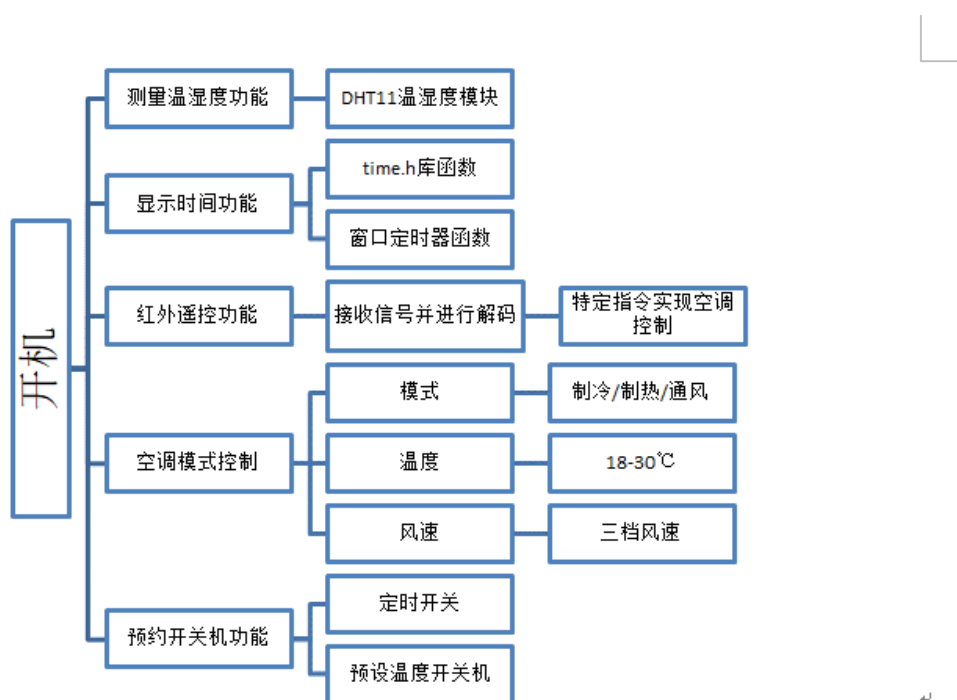


图 1 系统结构框图

2.2 小组人员分工（5 分）

说明：本人在小组中的分工和实际完成的工作，详细说明每人负责的功能模块。

模块 1（时间显示模块）：主要由孙天一负责，次要由高成鑫负责。

模块 2（温度湿度采集显示模块）：主要由高成鑫负责，次要由孙天一负责。

模块 3（设置温度和工作模式功能模块）：主要由孙天一负责，次要由高成鑫负责。

模块 4（定时器输出 PWM 控制功率模块）：主要由高成鑫负责次要由孙天一负责。

模块 5（红外控制模块）：主要由高成鑫负责，次要由孙天一负责。

模块 6（预约开关机和达到预设温度开关机）：主要由高成鑫负责，次要由孙天一负责。

2.3 分工接口定义（5 分）

说明：详细说明功能模块之间如何接口。

温度测量（模拟温度模块）、红外接受，模块由系统定时器控制，当收到数据产生中断进入。

屏幕触摸检测、温湿度（数字温度模块），在主循环中由 CPU 循环执行（数字温度模块还受到定时器的限制，每 1s 执行一次）

预约定时、按键检测、空调变频控制等放在创建窗口的回调函数中，每次当有按键按下

或 RTC 定时产生时执行。

日期时间显示，刷新部分在窗口的回调函数中，对于最初的赋值同窗口创建时的初始化一起，之后便由 RTC 时钟自动递增。

3 功能模块设计方案（30 分）

3.1 模块 1：时间显示模块

说明：详细说明每个功能模块的设计的方法和过程。

3.1.1 模块功能要求

日期、时间显示功能，在屏幕上显示当前日期、时间。

3.1.2 模块设计思路

首先应该找到时间生成的来源以及时间递增的方式，最终选择通过 `time.h` 库中的函数 `time_t time(time_t* t)`；取得从 1900 年 1 月 1 日至今的秒数，然后使用 `struct tm* gmtime(const time_t* timep)`；函数将 `time_t` 表示的时间转换为没有经过时区转换的 UTC 时间，是一个 `struct tm` 结构指针，UTC 时间可以格林威治时间，与我们的当地时间相差 8 小时，对上述结果加 8（并判断是否过了一天、过了一个月……）即可得到现在的时间。当我们把这个时间赋给 RTC 时要注意此时的年、月需要处理。具体的就是年等于获得结构体中年份加 1900 然后对 1000 取余，RTC 时间的月等于获得结构体月加一。将此值赋给 RTC 时钟之后就可以获得当前日期时间，通过 `TEXT_SetText` 函数按照字符串的形式输出时间和日期。

3.1.3 CubeMX 设计方案

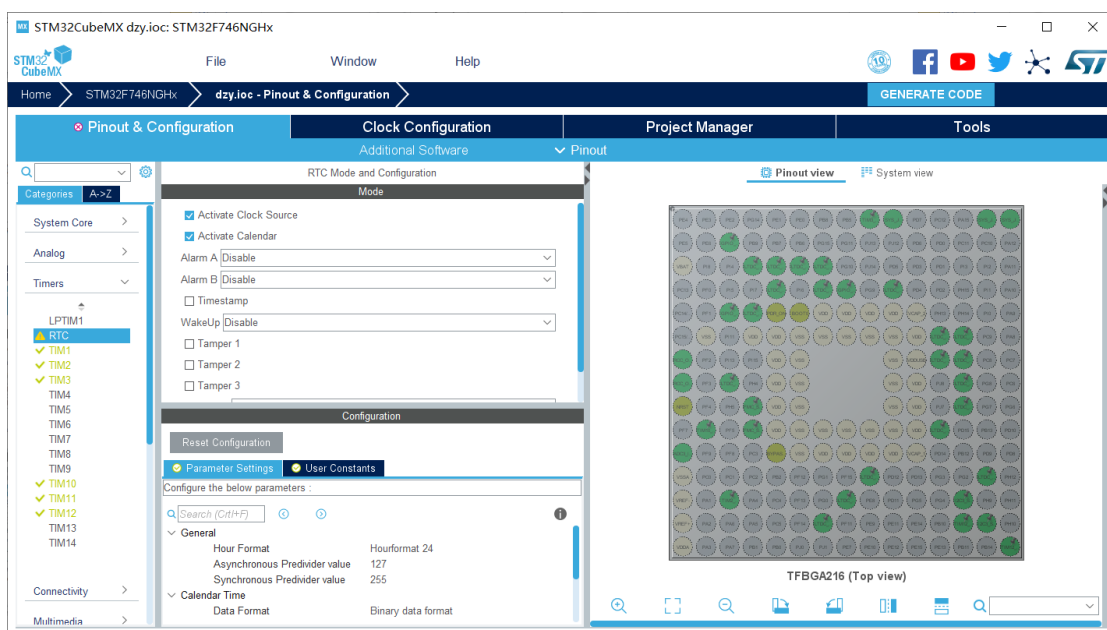


图 2 RTC 时钟配置

3.1.4 模块内部结构

没有外部模块

3.1.5 模块软件流程图

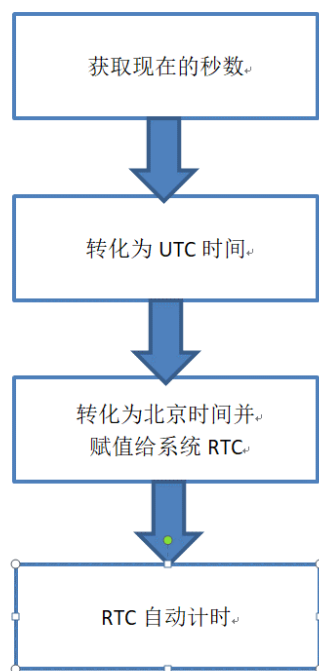


图 3 利用 RTC 计时

3.2 模块 2：温度湿度采集显示模块

3.2.1 模块功能要求

温湿度测量功能，系统可以测量室内温度和湿度并在屏幕上显示。

3.2.2 模块设计思路

基础的单独温度测量主要是利用模拟温度感应器采用 AD 转换，得到值，之后经过公示计算转换实现所要测的温度值。

提高部分的温湿度测量则是利用 DHT11 来实现定义一个 us 级别的时延函数，然后按照 DHT11 的要求主机先发送下拉和上拉，然后接收来自 DHT11 的数据，数据是 40 位的，其中包括 8bit 湿度整数数据+8bit 湿度小数数据+8bit 温度整数数据+ 8bit 温度小数数据+8bit 校验位。将得到的值进行校验和输出即可。

利用 DHT11 来实现，由于 DHT11 的总体通信流程三步，第一步：主机先发送开始信号，从机会返回一个相应信号进行应答。第二步：主机信号线拉高准备接收数据。第三步：开始接收数据（一次接收 40 位）。查阅资料得到时序图：

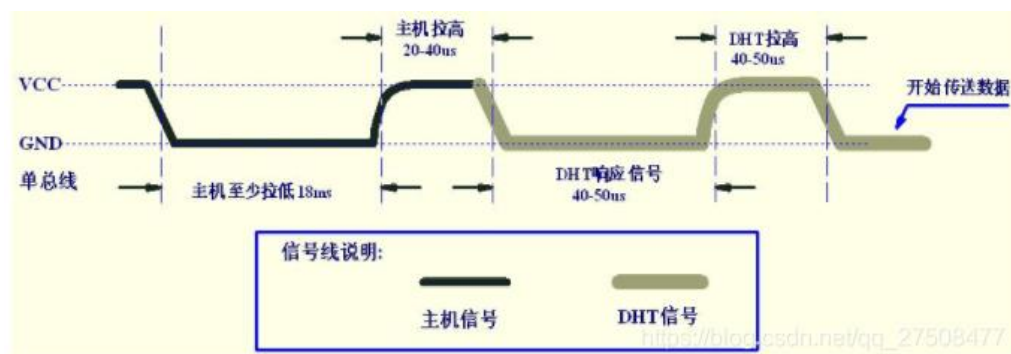


图 4 DHT11 控制时序

由于需要用到 us 级别的时间，所以要定义一个 us 级别的时延函数，然后按照 DHT11 的要求主机先发送下拉和上拉，然后接收来自 DHT11 的数据，根据手册传输数据可以根据高电平持续时间读出，我们对于数据在高电平时统一延时 40us(正常传输 0 的高电平最多 28us)，读取电平状态，低则为 0 高则为 1，数据是 40 位的，而且查阅资料得到：

DATA 用于微处理器与 DHT11 之间的通讯和同步,采用单总线数据格式，一次传送 40 位数据，高位先出。

数据格式:

8bit 湿度整数数据 + 8bit 湿度小数数据+8bit 温度整数数据 + 8bit 温度小数数据+8bit 校验位。

◎校验位数据定义

“8bit 湿度整数数据 + 8bit 湿度小数数据+8bit 温度整数数据 + 8bit 温度小数数据” 8bit 校验位等于所得结果的末 8 位。

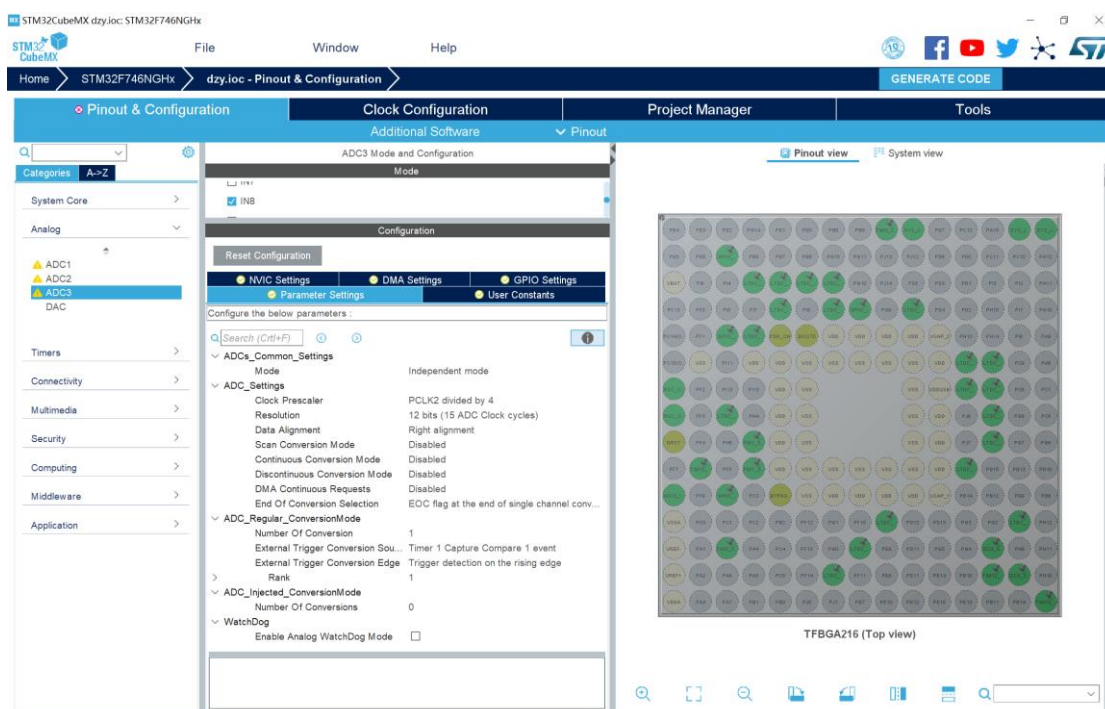
https://blog.csdn.net/qq_34952376

图 5 DTH11 数据格式

然后对其进行校验和输出即可。

3. 2. 3 CubeMX 设计方案

使用模拟温度传感器，则使用 ADC3IN8 作为输入，使用 TIM1CH1，作为 AD 采样的外部触发。



使用开发板上 PF8，初始设置为普通输出模式，作为发送信号与接收信号的引脚。

us 级别的时延使用系统滴答定时器 SysTick 完成，在设定系统分频系数为 $\text{SystemCoreClock}/1000000$ 从而得到 1us 的定时中断，在中断中对延时时间进行自减，从而得到期望的定时。（因而不需其他定时器配置）

3.2.4 模块内部结构

3.2.5 模块软件流程图

如图 7 所示。

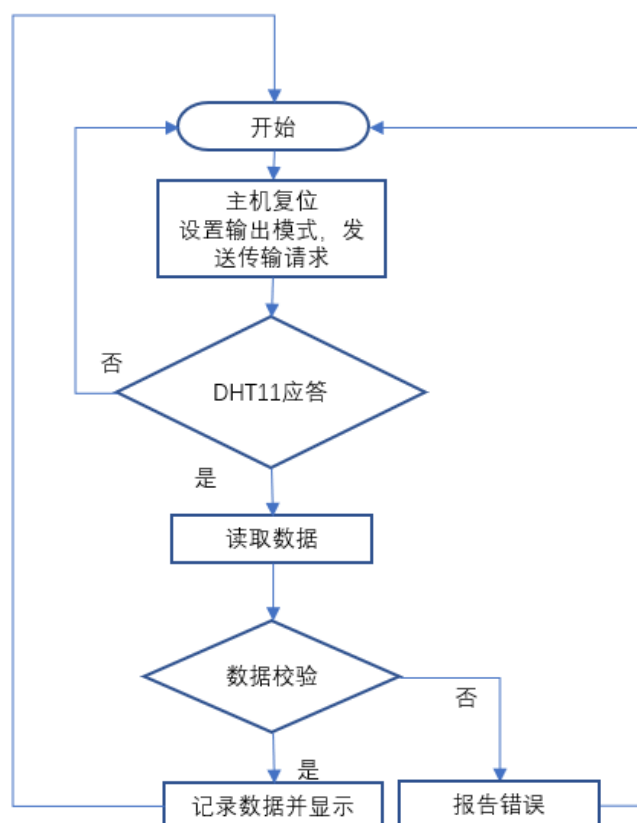


图 7 利用 DHT11 进行温度湿度采集

3.3 模块 3：设置温度和工作模式功能模块

3.3.1 模块功能要求

空调设置功能，系统可以通过触摸屏设置空调的温度、风速、工作模式，并在屏幕上显示当前的设置。温度可以在 18~30℃ 之间调节，模式可配置为“制冷”、“制热”和“通风”，风速可配置为“高”、“中”、“低”三档。

3.3.2 模块设计思路

该模块主要是屏幕显示的设计和操作窗口设置。主要需要实现的功能和要求是实现屏幕显示和液晶屏的触屏输入控制。温度和工作模式都是添加按键，检测到触摸按键借由回调函数实现，对每个不同模块所实现的功能进行编程即可。

3.3.3 CubeMX 设计方案

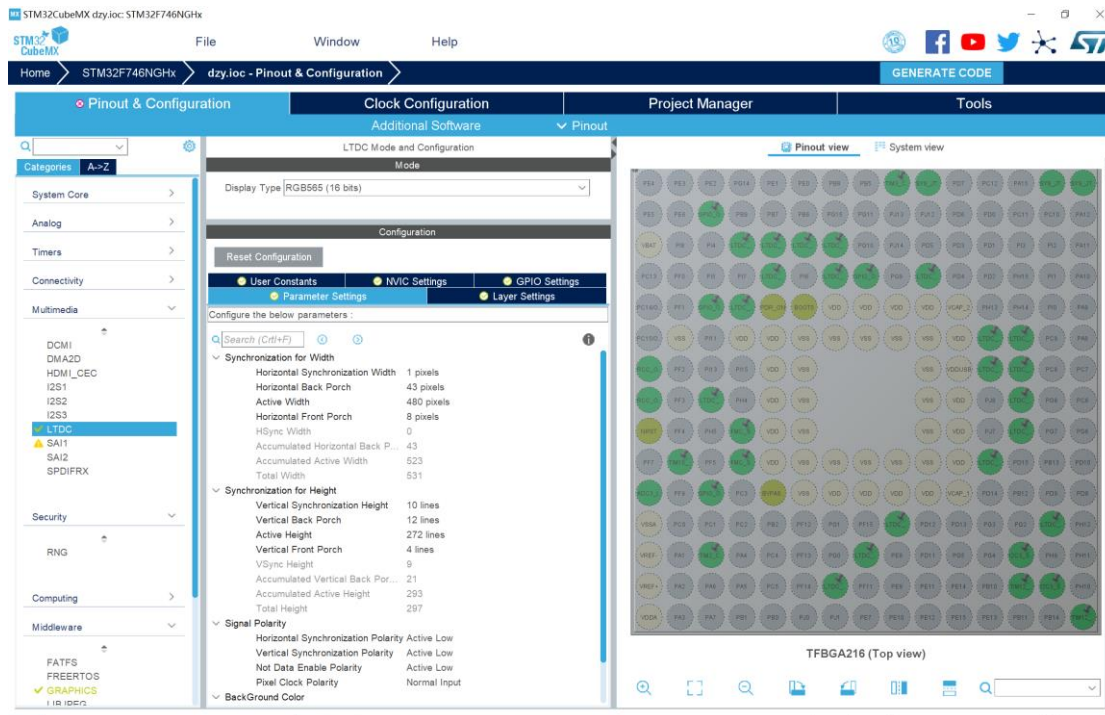


图 9 LTDC 配置

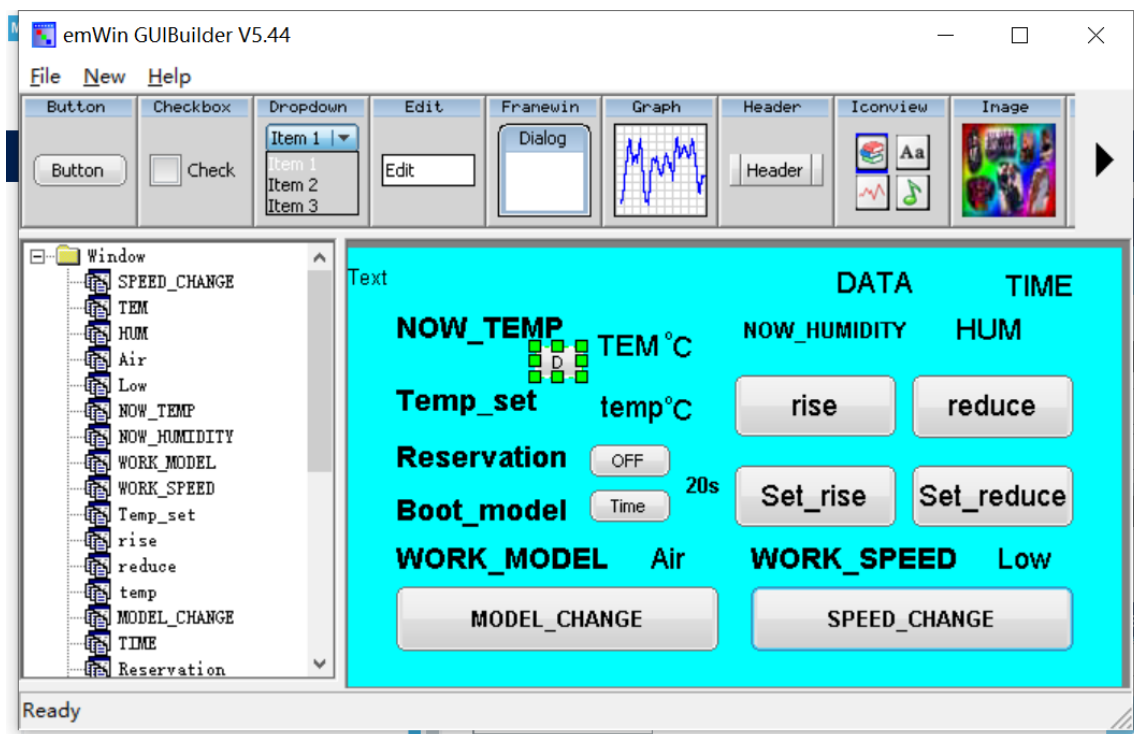


图 8 STemWin 窗口绘制

对于液晶屏的控制主要依靠液晶屏控制器 LTDC，按手册中参数在 CubeMX 中对相关引脚和时钟进行配置，利用 StemWin 绘制窗口，在其中布置按钮与文本显示。

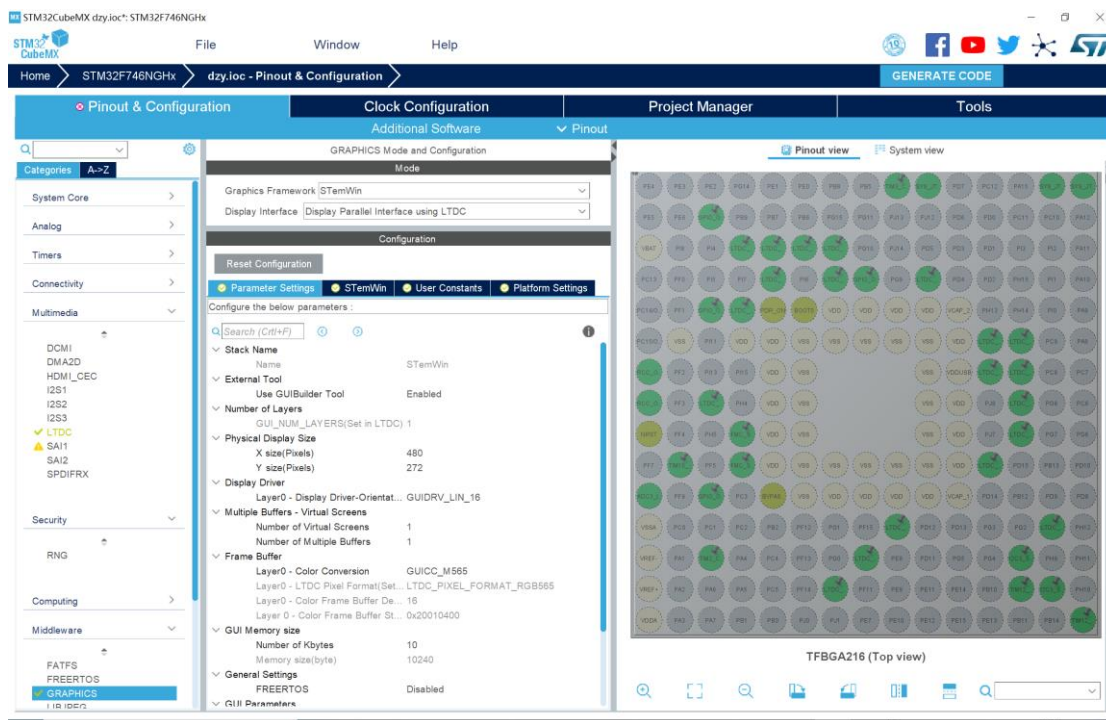


图 10 STemWin 配置

3. 3. 4 模块内部结构

屏幕显示无外部附加模块

3. 3. 5 模块软件流程图

如图 11 所示。

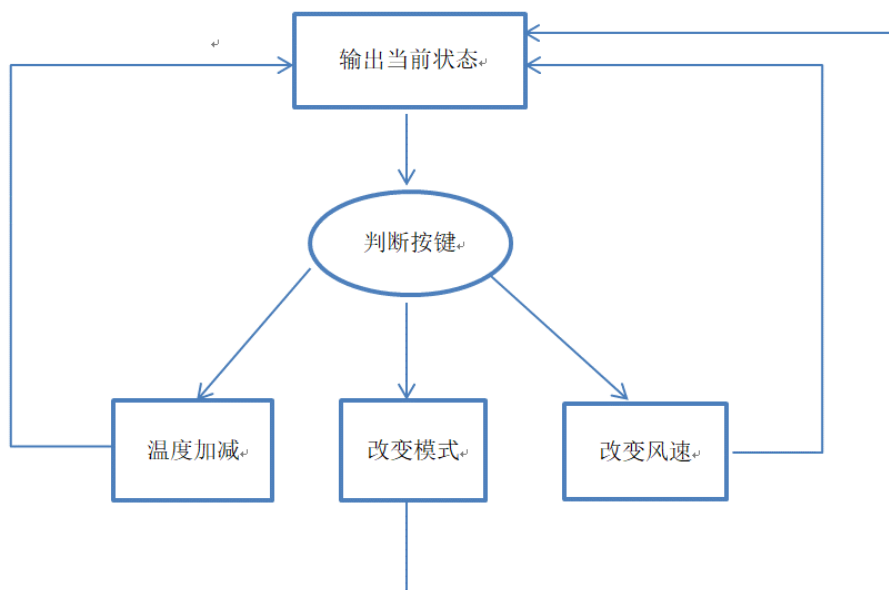


图 11 空调控制流程图

3.4 模块 4（定时器输出 PWM 控制各模式的功率模块）

3.4.1 模块功能要求

空调控制功能，在功能 3 的基础上增加空调变频控制功能，使用一个蓝色 LED 灯和一个红色 LED 灯表示空调制冷/制热功率和风速。LED 灯的亮度表示压缩机转速，室内温度与设置温度温差越大，LED 亮度越高，表示制冷/制热功率越大。使用一个黄色 LED 灯表示风速，LED 灯亮度越高表示风速越大。

3.4.2 模块设计思路

该模块根本设计思路就是通过控制 pwm 的占空比进而控制 led 的亮度。根本温差信息线性的改变 pwm 的占空比，从而控制 LED 灯的亮度；风速的信息则是用 LED 灯的亮度表示空调的输出功率，对应 PWM 的三个档位的不同占空比。

PWM 的占空比可以设三个 int 类型变量对应输出亮度，对风速可以直接根据变量的值建立对应的函数得到 PWM 占空比，对于制冷、制热，则需要将设定温度与当前温度做差，利用差值建立函数得到 PWM 占空比。

3.4.3 CubeMX 设计方案

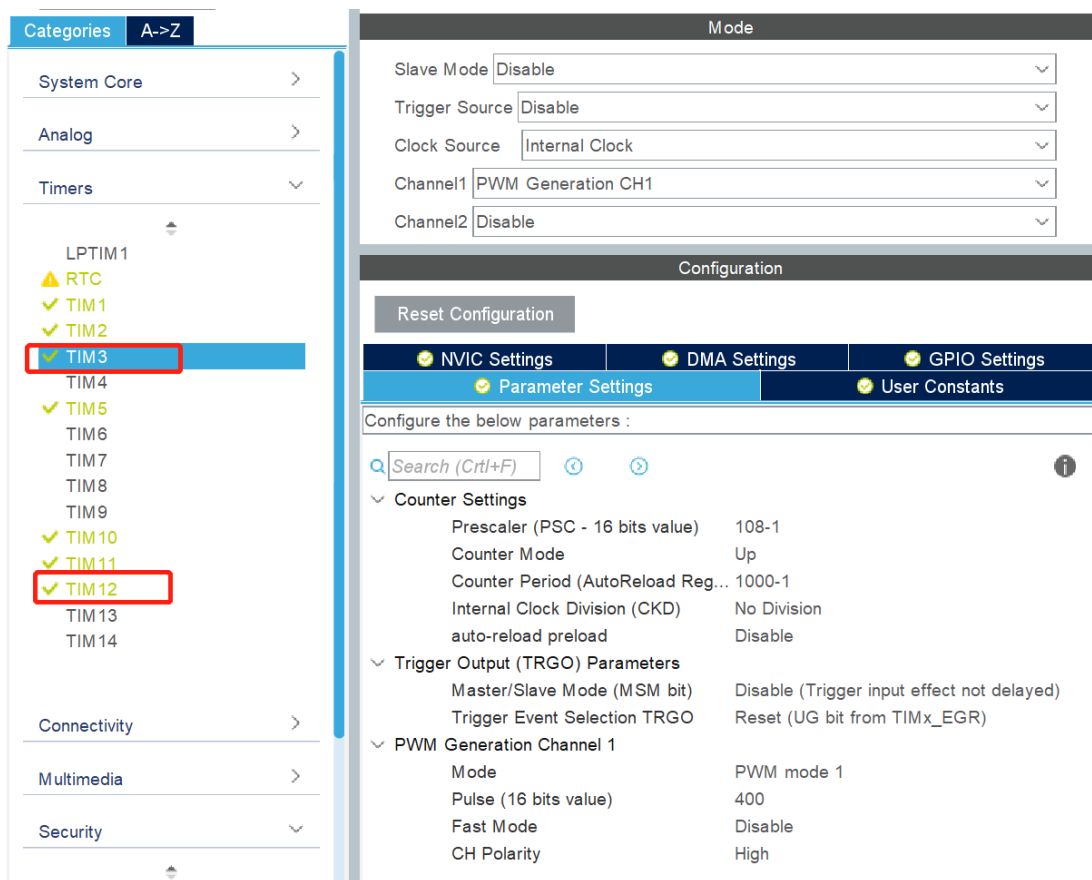


图 12 利用 TIM3 通道一、TIM12 通道一、二产生 PWM 波

开发板上 TIM3CH1 通道对应 PB4 口,产生 PWM 输出控制黄灯表示风速,开发板上 TIM12, CH1 通道对应 PH6、CH2 通道对应 PB15,控制蓝、红灯,表示制冷、制热功率。

3.4.4 模块内部结构

3.4.5 模块软件流程图

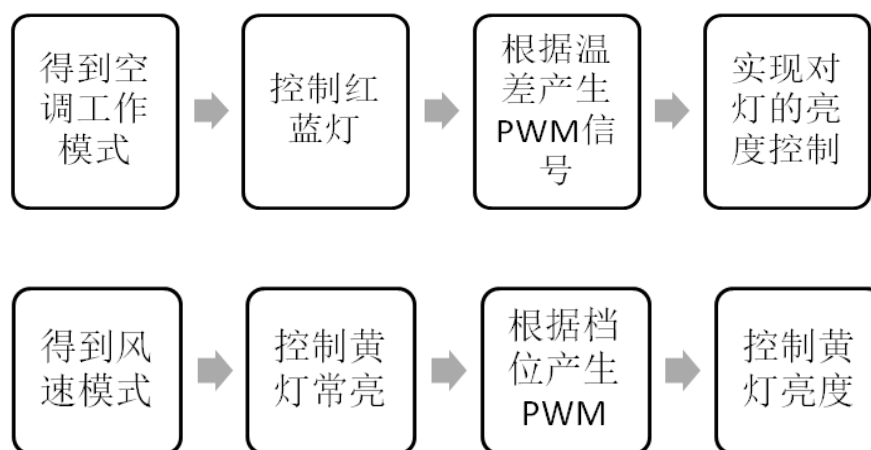


图 13 定时器输出 PWM 控制各模式的功率

如图 13 所示。

3.5 模块 5（红外控制模块）

3.5.1 模块功能要求

全功能红外遥控，系统可以解码红外遥控器的输出，可以通过红外遥控器控制系统的开关以及设置温度、风速和工作模式。

3.5.2 模块设计思路

红外控制功能主要包括两部分，其中最重要的难点就在于解码，先设计好红外接收模块，识别出信号，对信号进行汇总，找到规律，确定不同编码控制的功能，进而实现对应功能。第二部分就是通过修改指定的编码实现所要求的的功能，包括开关机以及相关参数设置，这一部分是讲红外信号识别出来并与之前设计好的功能所匹配。

3.5.3 CubeMX 设计方案

红外的解码是使用定时器的输入捕获功能，我们使用定时器 2 的 ch1 主动捕获下降沿作为周期的开始，ch2 被动捕获上升沿作为低电平时间的计数，我们最后只需要用到 ch1 的数据。

配置信息如图 10 所示。

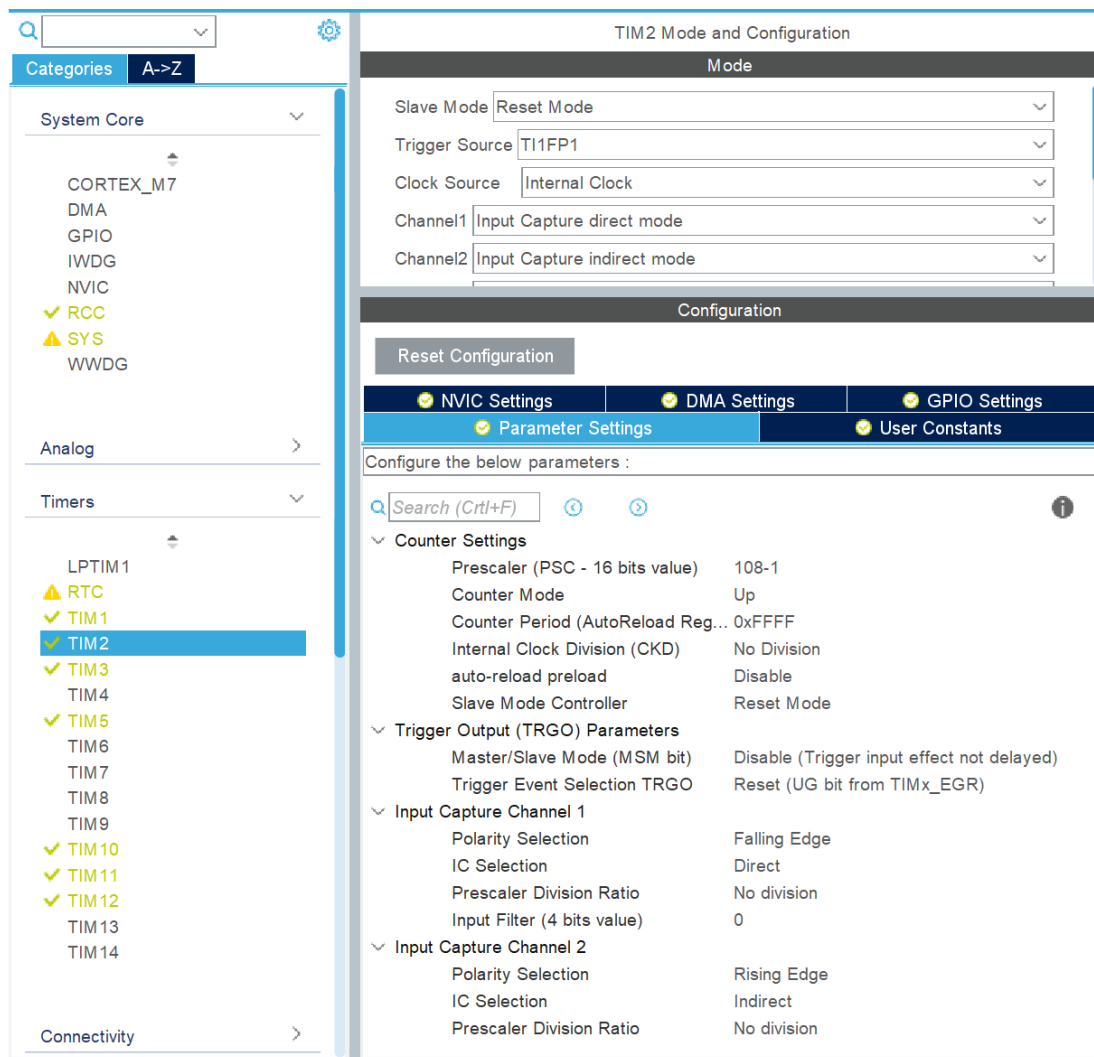


图 14 定时器 2 输入捕获配置

3.5.4 模块内部结构

3.5.5 模块软件流程图

如图 15 所示。

3.6 模块 6（预约开关机和达到预设温度开关机）

3.6.1 模块功能要求

预约功能，预约定时开关机，系统可以在指定时间按预设模式开始工作。预设温度开关机，系统可以在室内温度超出指定范围时自动开始制冷/制热。

3.6.2 模块设计思路

预约开关机主要包括定时开关机和预设温度关机。其中定时为确定进行定时设置的起始时间点，并在经过所设置的时间后，开始执行开关机程序功能。预设温度则是检测室内温度

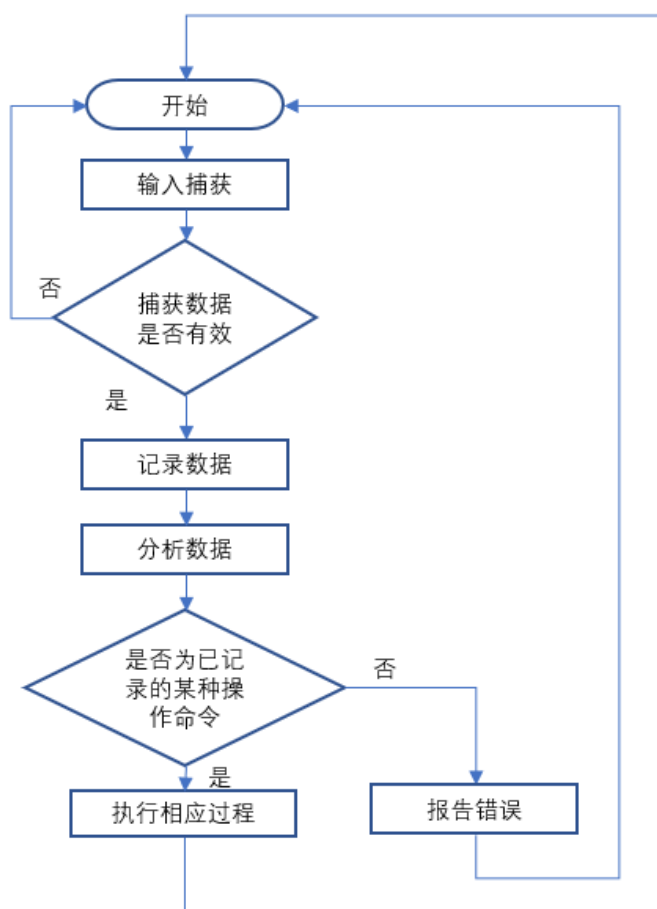


图 15 红外信号判断

与预设温度值的大小，超过大的预设温度，执行开机操作，工作模式为制冷。低于小的预设温度，执行开机操作，工作模式为制热。

此项功能也可在遥控器上进行操作，“静眠”按钮对应定时开关机启动，“健康气流”按钮对应定时开关机模式切换，“健康”+“温度”加减，可以调节设定时间/温度值。

3.6.3 CubeMX 设计方案

此项功能直接在之前 StmWin 创建的窗口中进行补充绘制，之前的图 5 中已包含全部组件。主要包括定时开/关机启动模式（定时启动、定温度启动）、定时开启与否按钮、定时改变（加减）。

3.6.4 模块内部结构

3.6.5 模块软件流程图

如图 16 所示。

4 系统设计中问题及解决（10 分）

4.1 问题 1（每人至少一项）

说明：在设计和调试过程中遇到的问题、产生原因、解决方法。

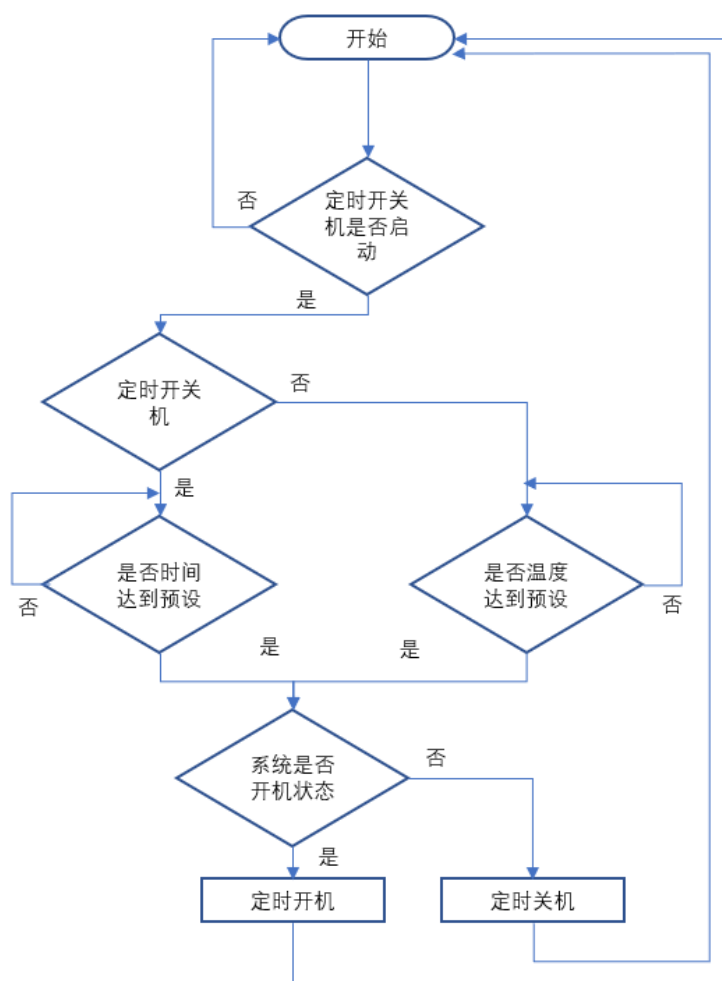


图 16 定时开关机

在使用 RTC 时钟实时显示时间时，最开始通过窗口将时间显示到屏幕上时，时间无法自动计时，但是其他按键没有任何影响。从 RTC 获得了时间但是在显示时无法正常进行累加，通过检验程序发现，这是因为时间的输出函数只调用了一次，而对于之后进行自动计时的函数却没有再次进行输出。意识到缺乏重复执行输出功能程序，尝试了 while 函数等其他方法，最后从芯片手册中，找到了 WM_TIMER 函数，采用定时器的方法进行输出，这时进行执行发现函数可以产生计时，但是自动更新还是有一定问题，通过查询手册发现是对于定时器的使用存在问题，通过调整定时器的相关函数，设置低于 1s 的周期，使得定时器在低于时间改变的周期下不断重复执行，使得屏幕显示区域可以不断接收到自动计时的时间。

4.2 问题 2

在使用 time() 函数获得时间时按遥控器，红外接受模块无法正确译码。在开始的时候时间显示是使用 time() 函数，在 RTC 时钟的刷新下不停地获得，在这个过程中获取红外只能解码得到部分值，大概几十个数，甚至只有几个，无法再对里面各个部分的内容进行有效

的控制。联想到之前在中断函数中包含 `printf` 时解码出现问题的情况，尝试让使用系统函数和中断不再同时进行，最终将 `time()` 函数只调用一次，就是在开始创建的 windows 初始化的时候利用函数得到时间，然后将当前时间值赋给 RTC 时钟，让 RTC 自增而不是反复使用 `timer()` 获得时间。

但是，这样产生的另一个问题是，RTC 时钟使用的是内部时钟源 LSI，根据网上资料显示，它受温度等影响，RTC 定时不够精确，如果条件允许应该设置 RTC 自增与定时调用 `time()` 函数相结合。

最后，由于使用了 `time()` 函数，那么板子工作时就无法离开电脑，目前想到的是在利用 `time()` 给 RTC 赋值语句的周围加上条件编译语句，如果能自动判断能否根据 `time()` 函数获得时间，如果不能则跳过 `time()` 函数部分——RTC 计时也就会从 0 开始，需要用户手动校准时间。这样就确保程序能在离开调试环境下正常工作。

4.3 问题 3

（目前问题仍待解决）

在使用 DHT11 做测温时，第一次测温是正常的，可以看到显示出的温度，但之后就程序每次运行时都无法得到正确的返回值，由于 DHT11 在测温时有严格的时序控制流程，对于调试时也不好找出原因，目前只能使用手动复位让程序重新开始，从而获得新的温湿度。

5 系统使用说明

5.1 系统外观及接口说明（5 分）

说明：应附实物作品照片 2~3 张，说明作品中按键、显示单元、接口等输入输出单元的位置和功能。

如图 16 所示。

5.2 系统操作使用说明（15 分）

说明：作品的操作使用说明。

本系统为智能家居温度控制器，可以使用屏幕触摸或遥控控制模式、风速、设定温度等，并可以根据预设温度进行温度的变频控制。

在系统的工作模式设置下，默认的开启为通风状态。有 Air(通风)，Warm(加热)，Cold(制冷)三档，当使用通风时，只有代表送风的黄灯亮，在加热与制冷两档时，对应加热的红灯或制冷的蓝灯与黄色灯一起亮，根据设定温度与房间内温差大小实现智能控制。

系统的风速有低、中、高三档，对应黄灯的不同亮度，可以在工作时调节。

点击温度设定右边的 rise 与 reduce 可以调节设定温度，调节范围是 18-30。

温度设定下方是预约开关机按钮，点击 OFF 按钮可切换到启动预约开关机，调节

set_rise 与 set_reduce 可以加减数值，点击 Time 可以切换到定温度开关机。如果设定好后系统被关机，则认为是定时/定温度开机，否则则为定时/定温度关机，系统在达到条件后执行相应操作。定时开关机的单位是秒，为简单操作起见，设定的最大范围是 59s。

系统除了可使用触屏控制，也可以使用宿舍空调遥控器控制，遥控器开关键对应系统开启关闭，制冷、制热、除湿对应系统中制冷、加热、通风模式，温度加减是被设定温度与预约开关机共同使用，如果点击健康（屏幕上出现小叶子）后按温度加减，则为调节定时开关机，否则则为普通温度设定改变，风速按键可控制风速切换，“静眠”、“健康气流”用于控制定时开关机的启动与模式改变。

空调遥控器控制温度加减或预约开关机范围有一定受限，具体调节范围以屏幕上示数范围为准。

5.3 技术指标（5 分）

说明：作品的性能、规格等。

作品的性能基本完成所设计的要求，且失误率低于千分之一，可以投入正常使用

对于屏幕的操作，能在 100ms 以内时间给出反应，使用遥控器可以达到更快的速度。

规格：长宽不超过 20cm，高不超过 5cm。

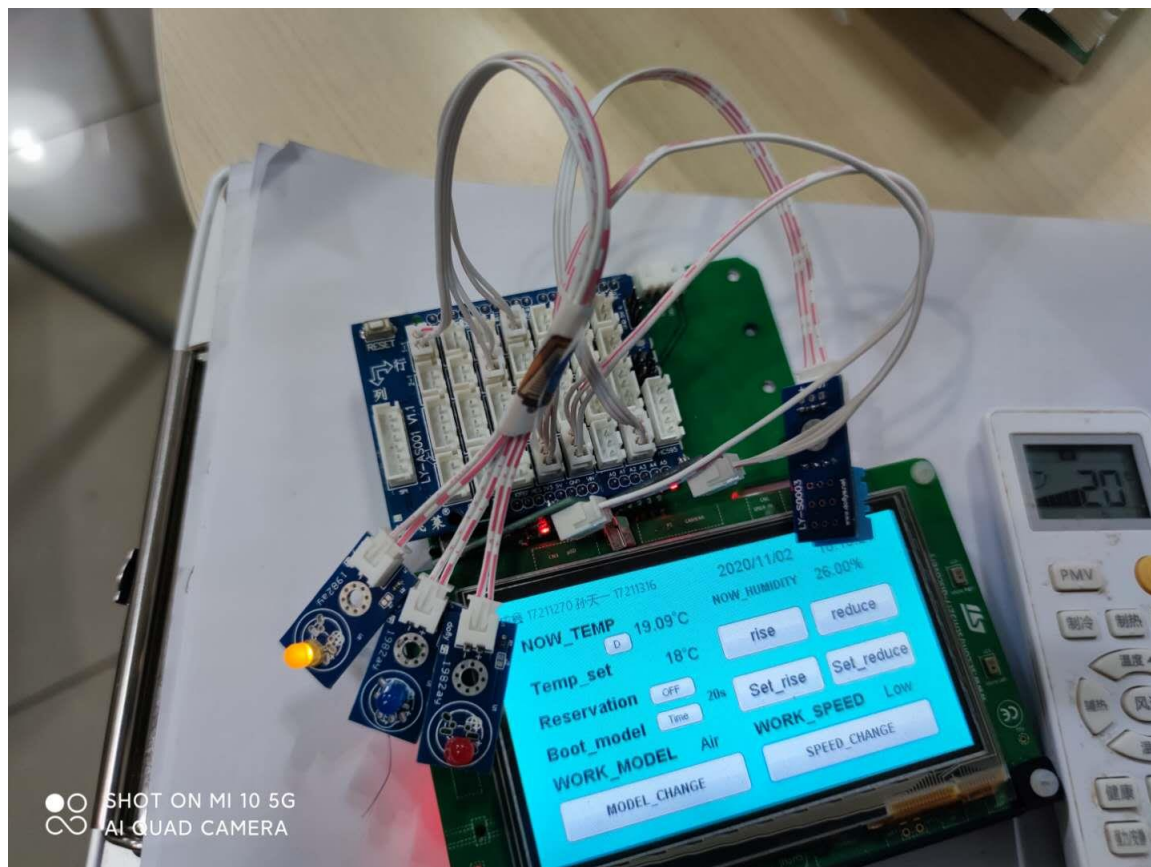
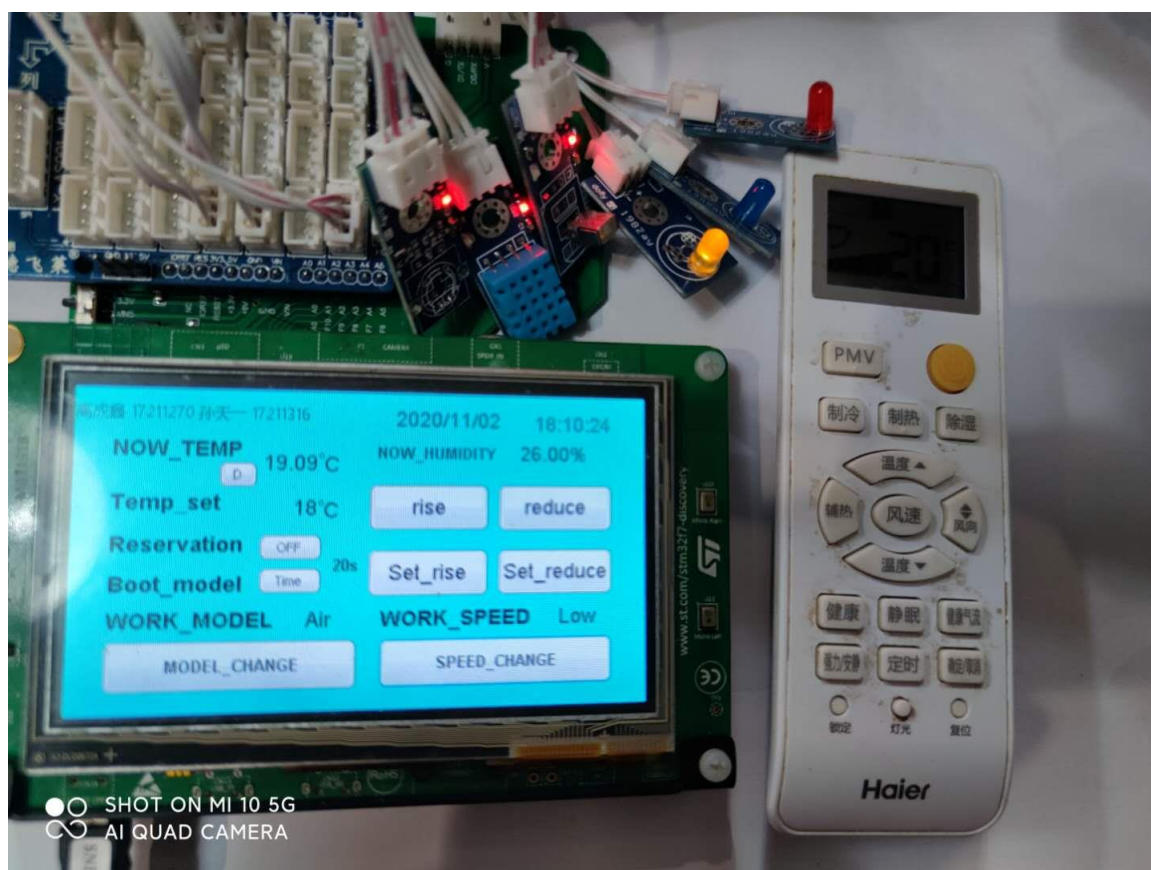


图 17 系统整体外观+传感器接口

5.4 其他（可选）

说明：对作品的其他说明。

6 总结

6.1 设计结果（5 分）

说明：对本作品设计结果总结。

本次设计结果，基本上完成了老师要求的基础部分功能并对提高部分做了部分涉及，基本完成了任务。对于作品美化上还缺乏尝试，用户界面存在相似的功能占用多个模块（设定温度与设定定时时间、设定温度）等，屏幕上各个按钮排列比较单调，缺乏主次感，用户界面算不得友好。对于系统中存在的一些问题也没有进行深究，系统还存在上电后如果接触不良等，程序就无法复位后正常运行的问题等。总体来说，对于设计结果还算差强人意。

6.2 心得体会（5 分）

说明：本人在本课程中的收获与体会。

孙天一：通过本次课程的学习，我主要收获了两方面的内容，一方面获得对于嵌入式系统相关的基础知识和相关体系，另一方面获得了对嵌入式系统开发的整体结构和程序编写的实践。其中第二方面的收获，让我收获颇多。

作为一门实践性的课程设计，在课程学习过程中，从最开始的实验教学到后面的课程设计，极大的提升了我的编程能力，让我对于嵌入式的了解进一步加深，通过实践的项目编程，我更深刻的理解前后端的联系和区别以及硬件和软件的交互性。其中，尤以硬件上和工程设置上的相关端口和功能的实现，使得我可以从程序最开始的执行，如何一步步运行，在不同的阶段是什么部分产生怎样的交互有了更深层次的理解。在最后的大作业中，我深刻体会到工程文件和库函数的便捷和重要，了解到基础知识的扎实的重要性，在设计过程遇到的很多问题，出现的很多 bug，发生的很多原因其实都是很简单，很基础的问题，其中知识的融会贯通显得尤为重要。

这次课程我最大的收获有两个：1. 学会如何从别的地方吸取学习，如何从其他的地方发展对自己有用的内容，进行借鉴改编，从而实现自己的功能，提高了我的学习能力和搜索能力，学会了如何举一反三。2. 培养了我良好的编程习惯，因为程序的复杂，我养成了一步一个脚印的编程习惯，当上一个功能或部分程序完全运行没有问题的时候再进行下一步的编程，这样再出现问题的时候，可以更快的找到问题，极大的提高了编程效率。

高成鑫：在以前的学习，我们编写的代码大多数是运行在 Windows 或者安卓下，使用 C++ 或 Java 这种离底层更远的语言。我们通常只用考虑程序是怎样实现的，而不用考虑程序是如何，本次课程设计我们使用 C 对 GPIO，或者说寄存器、内存单元进行访问时，我们要

考虑的就程序如何一步步执行，在不同的阶段会发生怎样的事情，特别的，由于我们是对内存单元进行直接地控制，我们就要特别小心的寻找可能地栈溢出或野指针产生的影响。

在使用库函数时，有一种很特别的感觉，一方面库函数毕竟是在 C 语言下的操作，许多过程都显得琐碎却重要，另一方面对一些复杂操作提供的简单的接口，对于屏蔽下层的实现细节，给我们了很多使用高级语言的熟悉感，是一种很特别的感觉。最后在课程设计的过程中，也遇到过很多费解的问题，虽然最后通过尝试提出了解决问题的方案，但是对于其中原理或有没有更优的方案没有更深的考虑，这些问题在以后的学习生活里需要注意。

6.3 意见建议

说明：本人对本课程教学的意见和建议。

孙天一：课程设计的题目非常合适，难度和内容都适当，相信课程组的老师花费了很多心思，我希望老师可以提供相关的编程教程资源，方便学生按需进行深一步学习。

高成鑫：无，课程设计的很好。