摘 要

本次实训主要分析了采用ARM Cortex-M3内核的STM32F103ZET6作为微控器的无线遥控智能电子时钟系统的设计过程。此智能系统的组成主要由蜂鸣器、按键、NRF24L01串口转USB无线模块、NRF24L01无线模块、LCD1602显示屏等组成。完成时钟计时，定时，蜂鸣器奏乐、切歌等一系列功能。通过python设计GUI界面实现上位机、下位机互通，通过NRF24L01无线模块实现上位机控制下位机，实现时钟计时，定时，蜂鸣器奏乐、切歌等一系列功能。

由单片机控制LCD1602显示屏的显示，修改设置时间采用操作方便的按键,定时奏乐系统使用有源蜂鸣器,通过调节输入方波的占功比，达到美妙旋律的声音实现报时的功能，整个系统设计比较完善，有显示时间、报时、时间调节设置的功能，除些之外还介绍了单片机和汇编程序一些经验和方法，以便将单片机的汇编程序经过局部的修改就可以移植到单片机上，实现不同的功能，节约重新开发的时间。

本系统设计并实现了集音乐播放，时间显示，时间设置为一体的音乐闹钟，解决了闹钟铃声单调，重复的苦恼。

关键词：STM32；NRF24L01无线模块；蜂鸣器；LCD1602显示屏

**Abstract**

This training mainly analyzes the design process of wireless remote control intelligent electronic clock system using stm32f103zet6 of arm Cortex-M3 core as microcontroller. The intelligent system is mainly composed of buzzer, key, nRF24L01 serial port to USB wireless module, nRF24L01 wireless module, LCD1602 display screen, etc. Complete the clock timing, timing, buzzer play, cut song and a series of functions. The GUI interface is designed by Python to realize the intercommunication between the upper computer and the lower computer, and the nRF24L01 wireless module is used to realize the functions of the upper computer controlling the lower computer, such as clock timing, timing, buzzer playing and song cutting.

The display of LCD1602 display screen is controlled by single chip microcomputer. The convenient buttons are used to modify the setting time. The active buzzer is used in the timing playing system. By adjusting the power ratio of the input square wave, the beautiful melody sound is achieved, and the function of time reporting is realized. The whole system design is relatively perfect, with the functions of time display, time reporting and time adjustment setting, In addition, it also introduces some experience and methods of MCU and assembler, so that the assembler of MCU can be transplanted to MCU after partial modification, so as to realize different functions and save the time of redevelopment.

This system designs and implements a music alarm clock which integrates music playing, time display and time setting, and solves the problem of monotonous and repetitive alarm.

**Keywords:** STM32; NRF24L01 Wireless Module; Buzzer; LCD1602 Display

目 录

**[摘 要](#_Toc6492)** [I](#_Toc6492)

**[Abstract](#_Toc7305)** [II](#_Toc7305)

**[绪 论](#_Toc3172)** [1](#_Toc3172)

**[1 硬件部分设计](#_Toc1951)** [3](#_Toc1951)

[1.1硬件结构设计 3](#_Toc20663)

[1.2 模块电路设计 3](#_Toc13189)

[1.2.1 NRF24L01无线模块的设计 3](#_Toc17803)

[1.2.2 蜂鸣器的设计 4](#_Toc16922)

[1.2.3 LCD1602显示屏的设计 5](#_Toc20153)

[1.3 无线遥控智能电子时钟主控单元 6](#_Toc4964)

**[2 软件部分设计](#_Toc29054)** [7](#_Toc29054)

[2.1 下位机软件编程环境 7](#_Toc20433)

[2.1.1 单片机最小系统简介 7](#_Toc5157)

[2.1.2 程序下载方法 7](#_Toc3502)

[2.2 主体程序设计 8](#_Toc24629)

[2.2.1 无线模块传输程序设计 9](#_Toc13099)

[2.2.2 LCD1602显示屏程序设计 10](#_Toc726)

[2.2.3 蜂鸣器程序设计 10](#_Toc417)

[2.3 上位机软件编程环境 11](#_Toc9605)

[2.3.1 Python软件介绍 12](#_Toc23551)

[2.3.2 QT Creator软件介绍 13](#_Toc2388)

[2.3.3 GUI界面 13](#_Toc10617)

**[3 系统测试](#_Toc16716)** [14](#_Toc16716)

[3.1 软件测试 14](#_Toc32508)

[3.2 硬件测试 14](#_Toc11429)

**[结 论](#_Toc19695)** [15](#_Toc19695)

**[致 谢](#_Toc6127)** [16](#_Toc6127)

**[参考文献](#_Toc31816)** [17](#_Toc31816)

**[附 录](#_Toc25702)** [18](#_Toc25702)

绪 论

20世纪末，电子技术获得了飞速的发展，在其推动下，现代电子产品几乎渗透到社会的各个领域，有力的推动和提高社会生产力的发展与信息化程度，同时也使现代电子产品性能进一步提升，产品更新换代的节奏也越来越快。自单片机出现至今，其应用领域随着人们的需求越来越大。现今，单片机技术已经遍满人类生活的各个领域，定时控制是不可缺少的条件，未来的物联网等对单片机的要求越来越高，智能化的道路定时必不可少。工业自动化中的数据处理以及实时控制，电子玩具，智能手机，游戏等等，定时的应用领域非常广泛。在以后物联网的发展中定时也是发挥着很大的作用，智能家居，智能交通系统，智能仪表等领域，或许这一刻的研究正是下一个难题的突破。

因此对于定时器的学习是非常重要的。但仅仅只有单片机的只是远远不够，针对实际生活，软硬件结合。1964 年，我们的前辈承担着东京奥运会的计时工作，此后，电子闹钟给我们的生活带来了意外的结果。

时间对于人们来说总是那么宝贵，工作的忙碌和繁杂容易使人忘记当前的时间。然后遇到重大事情的时候，一旦忘记时间，就会给自己或他人造成很大麻烦;工业生产中，很多环节都要用时间来确定工序替换时刻。所以随时知道准确的时间重要性已不言而喻，这是我们生活和工作中必不可少的。

随着生活节奏加快，人们对时间的依赖越来越高，对时间的观念越来越强，无论是职工上班，学生上学，还是火车到点，航班的起飞，都离不开对准确时间的要求。譬如，银行、医院、车站等公共场所可以远眺的时钟，就给出行人带来了极大方便。因而公共场所时钟的准确性，对于有时间要求的路人有着非常重要作用和意义。

目前市场上提供钟的类型主要有机械钟和石英钟。随着电子技术的飞速发展，石英钟在市场份额的比重越来越大，其性能和质量也不断提升。对于石英电子钟，其特点是:振动系统一般选用振动频率32.768KHz或是4.194304MHz;走时精度高，日差一般为0.5秒，即15秒/月;高精度的年差1~2秒;机芯可以设计制造很小很薄，最小机芯3.5令(1令=2.226毫米);最薄机芯为1.2毫米;机芯零件少，指针式石英电子钟表约为机械钟表的30%;显示式石英钟表零件更少，没有金属传动件，仅有少量按钮、簧片和螺钉;用电方式灵活，即可用市供电网供电，也可用一般的电池供电;装配、维修比较方便。由于有上述优点，所以深受广大消费者欢迎。

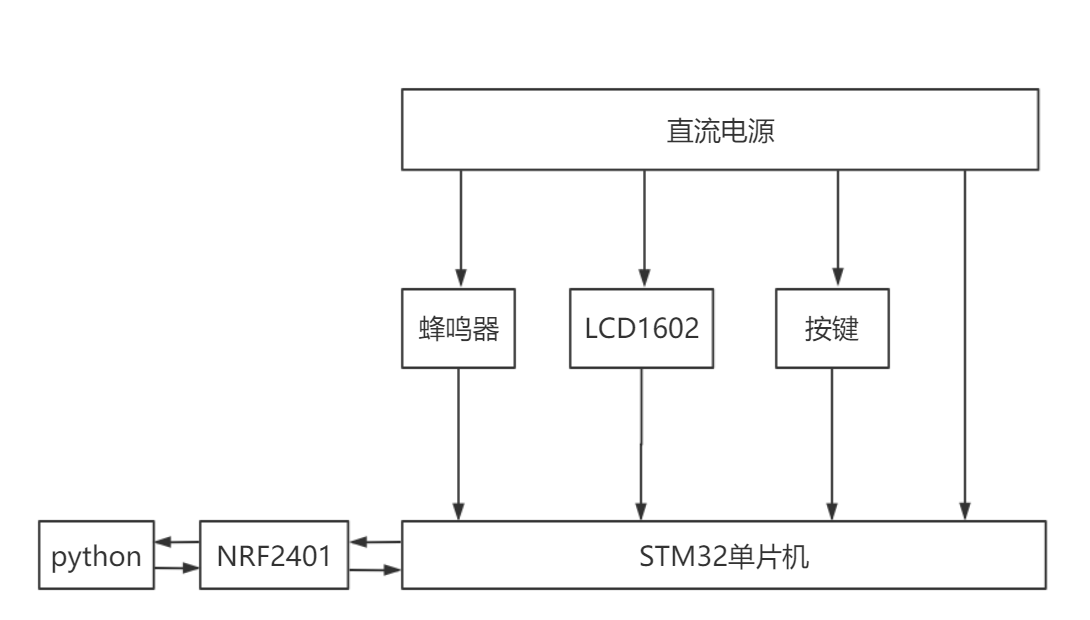
在日常生活中，为了给忙碌的人们提供方便，往往会在公共场合的某一高处挂一个时钟，以便大家都能看清此时此刻的时间。每当时钟上的时间与标准时间出现偏差时，工作人员不得不经过搬梯、爬高、扶钟、按键等一系列动作来调整时间，这既不方便，又存在安全隐患。如果用GPS进行校时，价格昂贵。本设计利用无线遥控进行校时，具有传输距离远、价格廉价、功能好等优点。

随着需求的增加，51单片机已经不能满足社会的发展，STM32系列专为低功耗、高性能、低成本的嵌入式设计的 ARM Cortex-M内核。本设计用的是STM32F103ZET6是一款时钟频率能达到72MHz的增强型内核。由于其体积小、价格低廉因此可非常方便的移植到遥控机器人、遥控小车上等，实现远距离控制。

1 硬件部分设计

1.1硬件结构设计

本设计主要采用STM32单片机作为多功能电子时钟的控制核心，结合蜂鸣器、按键、NRF24L01串口转USB无线模块、NRF24L01无线模块、LCD1602显示屏等组成。完成时钟计时，定时，蜂鸣器奏乐、切歌等一系列功能。通过python设计GUI界面实现上位机、下位机互通，通过NRF24L01无线模块实现上位机控制下位机，实现时钟计时，定时，蜂鸣器奏乐、切歌等一系列功能。系统整体框架如图1.1所示。

图1.1系统整体框架

1.2 模块电路设计

1.2.1 NRF24L01无线模块的设计

NRF24LO1是一款工作在2.4~2.5GHz世界通用ISM频段的单片无线收发器芯片。无线收发器包括:频率发生器、增强型SchockBurstTM模式控制器、功率放大器、晶体振荡器、调制器、解调器。输出功率、频道选择和协议的设置可以通过SPI接口进行设置。

极低的电流消耗：当工作在发射模式下发射功率为-6dBm时电流消耗为9mA，接收模式时为12.3mA。掉电模式和待机模式下电流消耗更低。

1. VCC：电源；
2. GND：接地；
3. CSN：芯片的片选线，CSN为低电平工作，连接单片机PG1引脚。
4. SCK：芯片控制的时钟线，连接单片机PG2引脚。
5. MISO：芯片控制数据线，主输出从输入，连接单片机PB14引脚。
6. MOSI：芯片控制数据线，主输入从输出，连接单片机PB15引脚。
7. IRQ：中断信号，无线通信过程中MCU主要是通过IRQ与NRF2401进行通信，连接单片机PB15引脚。
8. CE：芯片的模式控制线，在CSN为低的情况下，CE协同NRF2401的寄存器共同，连接单片机PG0引脚。无线模块原理图如图1.2所示。

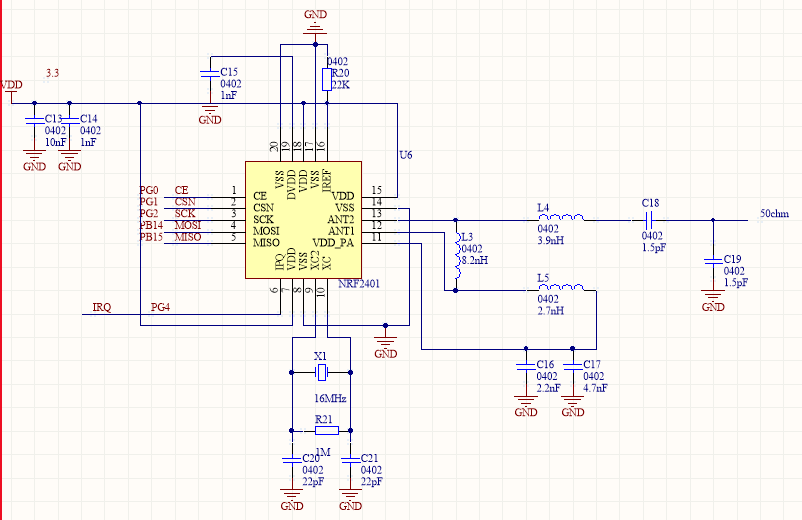


图1.2 NRF24L01无线模块原理图

1.2.2 蜂鸣器的设计

蜂鸣器是一种一体化结构的电子讯响器，采用[直流电压](https://baike.baidu.com/item/%E7%9B%B4%E6%B5%81%E7%94%B5%E5%8E%8B/8640960" \t "https://baike.baidu.com/item/%E8%9C%82%E9%B8%A3%E5%99%A8/_blank)供电，广泛应用于计算机、打印机、复印机、报警器、[电子玩具](https://baike.baidu.com/item/%E7%94%B5%E5%AD%90%E7%8E%A9%E5%85%B7/10753789" \t "https://baike.baidu.com/item/%E8%9C%82%E9%B8%A3%E5%99%A8/_blank)、汽车电子设备、电话机、[定时器](https://baike.baidu.com/item/%E5%AE%9A%E6%97%B6%E5%99%A8/5109454" \t "https://baike.baidu.com/item/%E8%9C%82%E9%B8%A3%E5%99%A8/_blank)等电子产品中作发声器件。蜂鸣器主要分为[压电式蜂鸣器](https://baike.baidu.com/item/%E5%8E%8B%E7%94%B5%E5%BC%8F%E8%9C%82%E9%B8%A3%E5%99%A8/3936962" \t "https://baike.baidu.com/item/%E8%9C%82%E9%B8%A3%E5%99%A8/_blank)和[电磁式蜂鸣器](https://baike.baidu.com/item/%E7%94%B5%E7%A3%81%E5%BC%8F%E8%9C%82%E9%B8%A3%E5%99%A8/6597409" \t "https://baike.baidu.com/item/%E8%9C%82%E9%B8%A3%E5%99%A8/_blank)两种类型。蜂鸣器在电路中用字母“H”或“HA”（旧标准用“FM”、“ZZG”、“LB”、“JD”等）表示。

1. VCC：电源；
2. GND：接地；
3. I/O口：连接单片机PB1引脚。蜂鸣器原理图如图1.3所示。

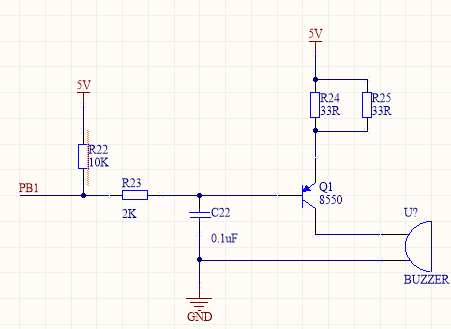


图1.3蜂鸣器原理图

1.2.3 LCD1602显示屏的设计

LCD1602液晶显示器是广泛使用的一种字符型液晶显示模块。它是由字符型液晶显示屏（LCD）、控制驱动主电路HD44780及其扩展驱动电路HD44100，以及少量电阻、电容元件和结构件等装配在PCB板上而组成。不同厂家生产的LCD1602芯片可能有所不同，但使用方法都是一样的。LCD1602的8根数据线和3根控制线E，RS和R/W与单片机相连后即可正常工作。一般应用中只须往LCD1602中写入命令和数据，因此，可将LCD1602的R/W读/写选择控制端直接接地，这样可节省1根数据线。VO引脚是液晶对比度调试端，通常连接一个10kΩ的[电位器](https://baike.baidu.com/item/%E7%94%B5%E4%BD%8D%E5%99%A8/6530251" \t "https://baike.baidu.com/item/LCD1602/_blank)即可实现对比度的调整；也可采用将一个适当大小的电[阻](https://baike.baidu.com/item/%E9%98%BB/5874163" \t "https://baike.baidu.com/item/LCD1602/_blank)从该引脚接地的方法进行调整，不过电阻的大小应通过调试决定[1]。

（1）引脚1：VSS为地电源。

（2）引脚2：VDD接5V正电源。

（3）引脚3：V0为[液晶显示器对比度](https://baike.baidu.com/item/%E6%B6%B2%E6%99%B6%E6%98%BE%E7%A4%BA%E5%99%A8%E5%AF%B9%E6%AF%94%E5%BA%A6/1162217" \t "https://baike.baidu.com/item/LCD1602/_blank)调整端，接正电源时对比度最弱，接地时对比度最高，对比度过高时会产生“鬼影”现象，使用时可以通过一个10kQ的电位器调整其对比度。

（4）引脚4：RS为寄存器选择脚，高电平时选择[数据寄存器](https://baike.baidu.com/item/%E6%95%B0%E6%8D%AE%E5%AF%84%E5%AD%98%E5%99%A8/1944231" \t "https://baike.baidu.com/item/LCD1602/_blank)、低电平时选择[指令寄存器](https://baike.baidu.com/item/%E6%8C%87%E4%BB%A4%E5%AF%84%E5%AD%98%E5%99%A8/3219483" \t "https://baike.baidu.com/item/LCD1602/_blank)。连接单片机PF0引脚。

（5）引脚5：R/W为读/写信号线，高电平时进行读操作，低电平时进行写操作。当RS和R/W共同为低电平时可以写入指令或显示地址；当RS为低电平，R/W为高电平时，可以读忙信号；当RS为高电平，R/W为低电平时，可以写入数据。连接单片机PF1引脚。

（6）引脚6：E端为使能端，当E端由高电平跳变为低电平时，液晶模块执行命令。连接单片机PF2引脚。

（7）引脚7~14：D0~D7为8位双向数据线。连接单片机PF2-PF10引脚。

（8）背光源正极。

（9）背光源负极。LCD1602显示屏原理图如图1.4所示。

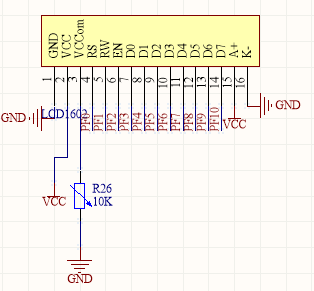


图1.4LCD1602显示屏原理图

1.3 无线遥控智能电子时钟主控单元

主控单元是STM32f103ZET6芯片，它负责将配置程序传输给按键，蜂鸣器并且将他们反馈回的数据显示在LCD1602屏幕或电脑的窗口上。主控单元图如图1.5所示。

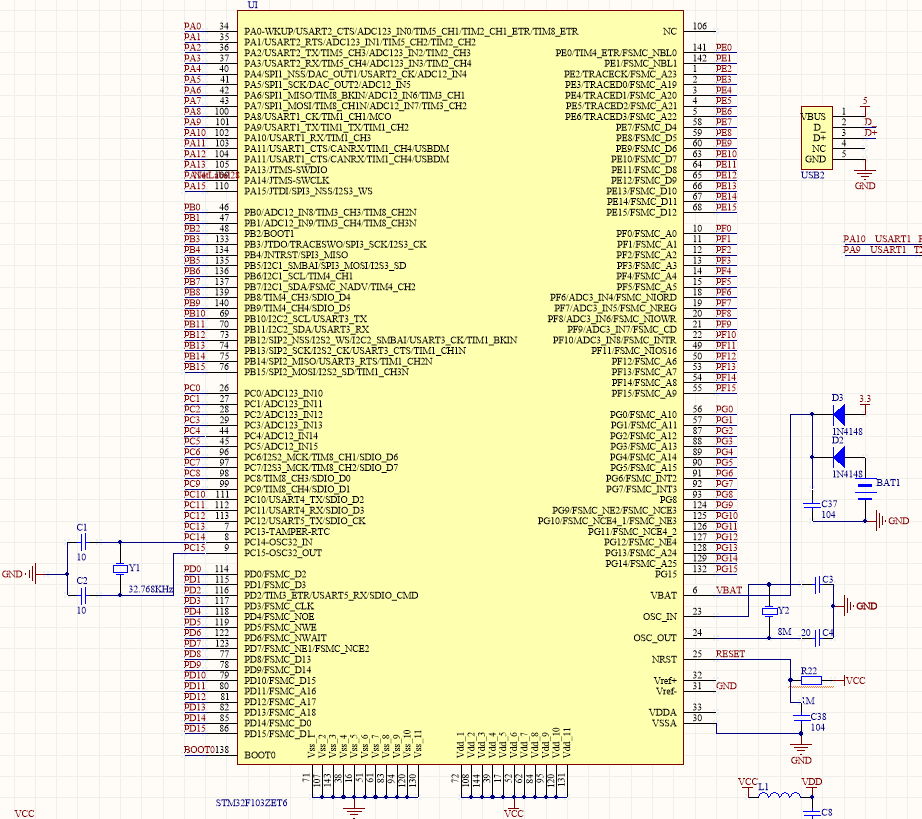


图1.5主控单元图

2 软件部分设计

2.1 下位机软件编程环境

本设计采用Keil Software公司出品的集成开发环境进行编程，其版本为Keil μVision5。Keil μVision5是2013年10月发布的最新版本，是针对Windows桌面平台所研发的一套完整的集成开发环境，具有可视化、灵活的Windows窗口管理。该软件不仅给用户提供了丰富的库函数，而且提供了功能强大的开发测试工具；既支持C语言编程，也支持汇编语言编程；可以完成编辑、编译、连接、测试、仿真等整个开发流程；此外，在程序被编译之后，即可生成相应的汇编语言代码，使用户可以切身感觉到该款软件的生成目标的代码效率是相当高的，并且多数语句生成的汇编代码更为紧凑和容易理解，进而使得编程效率更为高效[2]。

2.1.1 单片机最小系统简介

（1）32单片机最小系统复位电路的极性电容C1的大小直接影响单片机的复位时间，一般采用10~30uF，32单片机最小系统容值越大需要的复位时间越短。

（2）32单片机最小系统晶振Y1也可以采用6MHz或者11.0592MHz典型的晶振取11.0592MHz（因为可以准确地得到9600波特率和19200波特率,用于有串口通讯的场合）在正常工作的情况下可以采用更高频率的晶振，32单片机最小系统晶振的振荡频率直接影响单片机的处理速度，频率越大处理速度越快。

（3）32单片机最小系统起振电容C2、C3一般采用10~40pF，并且电容离晶振越近越好，晶振离单片机越近越好。

（4）P0口为开漏输出，作为输出口时需加上拉电阻，阻值一般为10k。

2.1.2 程序下载方法

（1）在单片机型号栏中选择正确的单片机型号，我们所用的为STM32F103C8T6，串口栏中选择正确的通讯端口，一般情况下会自动查询，占用的情况下将需要重新选择可用端口。

（2）点击“打开工程文件”栏进入文本对话框选择需要烧录的“\*\*\*.hex”文件，确定即可。

（3）点击“下载/编程”栏，此时整个窗口呈现灰色，进入等待烧录状态。

（4）断开PCB的电源重新上电进入烧录状态等待完成即可。

2.2 主体程序设计

下位机主体程序设计主要通过Keil uVision5软件进行设计，围绕四个模块进行设计。

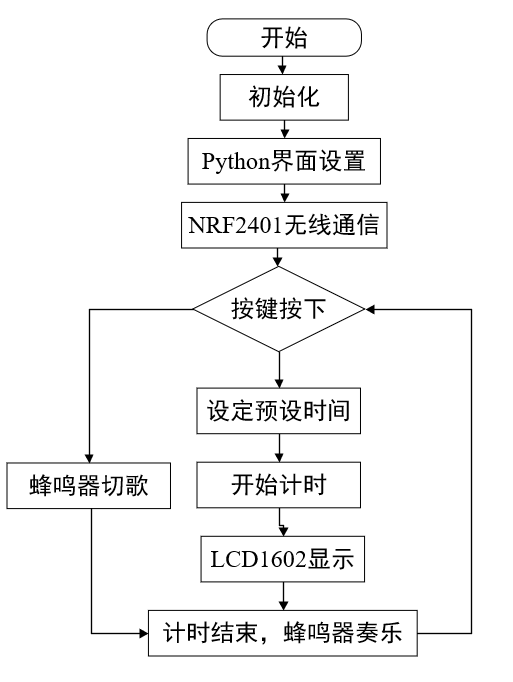
（1）主函数的设计

（2）NRF2401的配置

（3）蜂鸣器音乐的配置和切换

（4）LCD1602的显示

当程序执行时，蜂鸣器、NRF2401无线模块、LCD1602显示屏进行初始化，初始化完毕后，电子钟在不按下任何按键的情况下，LCD1602显示屏不显示时间。当有按键按下，LCD1602显示屏显示预设时间并开始计时。到达预设时间计时结束同时蜂鸣器奏乐。主体程序框图如图2.1所示。

图2.1主体程序框图

2.2.1 无线模块传输程序设计

模块初始化。初始状态时，LED、CE、SCK引脚均拉低，CSN 拉高，SPI初始完成。随后分别设置SPI接收，发送的地址。设置工作频道、数据长度、发射频率等信息。模块随即进入等待状态。设置工作模式。拉低 CE 设置为stand by l模式，配置SPI读写寄存器，随后拉高CE，短暂延时。配置好工作模式。发送：打包好数据，通过NRF24LO1\_TxPacket( )函数发送数据包；接收：通过NRF24L01\_RxPacket( )等待数据的到来[3]。无线模块传输发送框图如图2.2所示。

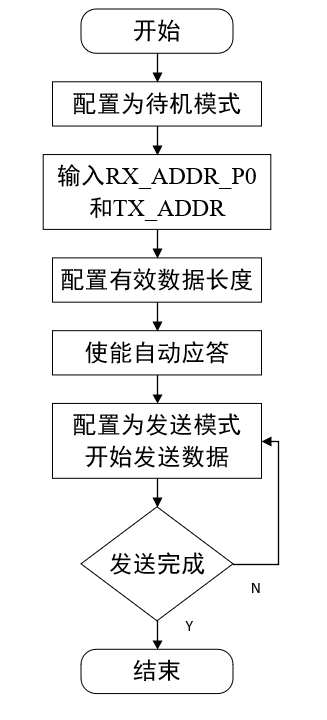


图2.2 无线模块发送传输框图

无线模块传输接收框图如图2.3所示。

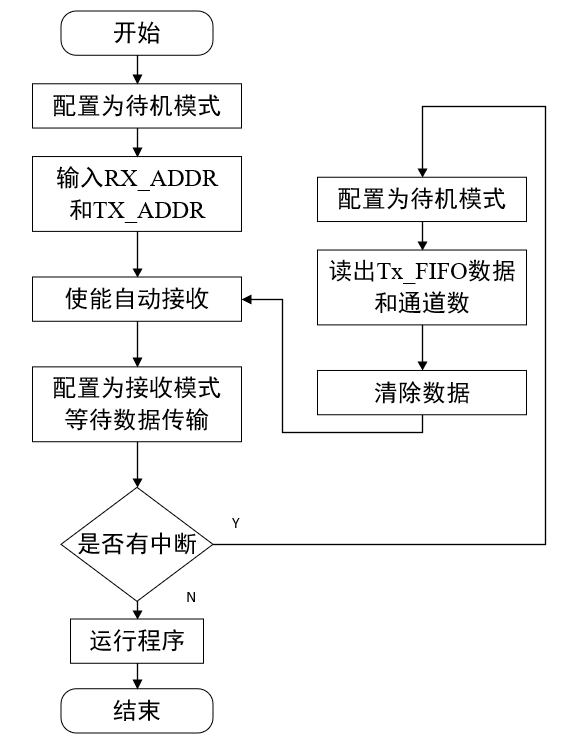


图2.3 无线模块传输接收框图

 if(NRF\_IRQ==0)

{  
            if(NRF24L01\_RxPacket(tmp\_buf)==0)

{  
  if( tmp\_buf[1]=='1')

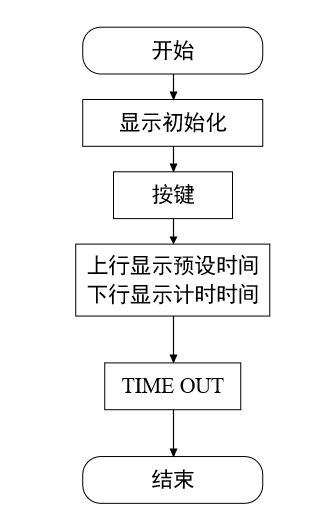
  printf("sd");

if( tmp\_buf[1]=='2')

 printf("2";  
  }   }

2.2.2 LCD1602显示屏程序设计

LCD1602的8根数据线和3根控制线E，RS和R/W与单片机相连后即可正常工作。一般应用中只须往LCD1602中写入命令和数据，因此，可将LCD1602的R/W读/写选择控制端直接接地，这样可节省1根数据线。VO引脚是液晶对比度调试端，通常连接一个10kΩ的[电位器](https://baike.baidu.com/item/%E7%94%B5%E4%BD%8D%E5%99%A8/6530251" \t "https://baike.baidu.com/item/LCD1602/_blank)即可实现对比度的调整；也可采用将一个适当大小的电[阻](https://baike.baidu.com/item/%E9%98%BB/5874163" \t "https://baike.baidu.com/item/LCD1602/_blank)从该引脚接地的方法进行调整，不过电阻的大小应通过调试决定[4]。LCD1602显示屏框图如图2.3所示。

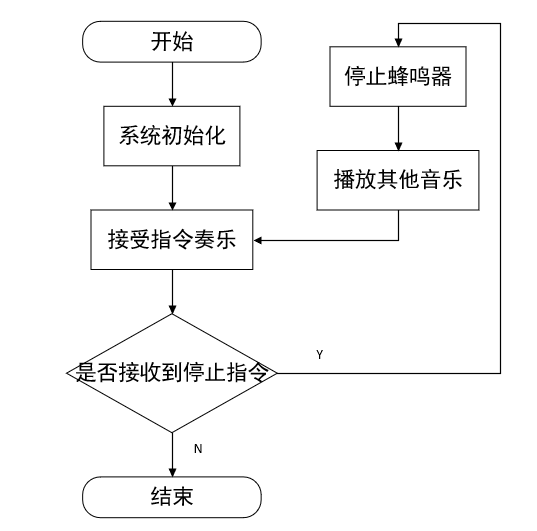
图2.3 LCD1602显示屏框图

sprintf((char\*)ST,"%2d:%2d:%2d",S\_Hr,S\_Min,S\_Rcount)；  
  LCD1602\_Show\_Str(1,0,"             ")；  
  LCD1602\_Show\_Str(1,0,ST)；

2.2.3 蜂鸣器程序设计

利用I/0 通过定时翻转电平产生符合蜂鸣器要求的频率的波形对蜂鸣器进行驱动。

由于蜂鸣器的工作电流一般比较大，以致于单片机的I/0口是无法直接驱动，所以要利用放大电路来驱动，一般使用三极管来放大电流就可以了。蜂鸣器驱动电路一般都包含以下几个部分：一个三极管、一个蜂鸣器、一个续流二极管和一个电源滤波电容[5]。蜂鸣器框图如图2.4所示。

图2.4蜂鸣器框图

if(music==0)  
{     
    play\_music();  
   if(KEY3==0)  
   {  
     GPIO\_SetBits(GPIOB,GPIO\_Pin\_1);}  
     }

2.3 上位机软件编程环境

Python是一种解释型、面向对象、动态数据类型的高级程序设计语言。Python由Guido van Rossum于1989年底发明，第一个公开发行版发行于1991年。像Perl语言一样, Python 源代码同样遵循 GPL(GNU General Public License)协议。

Qt Creator是全新的跨平台 Qt IDE（集成开发环境），可单独使用，也可与 Qt 库和开发工具组成一套完整的 SDK（软件开发工具包）。 其中包括：高级 C++ 代码编辑器，项目和生成管理工具，集成的上下文相关的帮助系统，图形化调试器，代码管理和浏览工具。

2.3.1 Python软件介绍

Python具有丰富和强大的库。它常被昵称为胶水语言，能够把用其他语言制作的各种模块（尤其是C/C++）很轻松地联结在一起。常见的一种应用情形是，使用Python快速生成程序的原型（有时甚至是程序的最终界面），然后对其中有特别要求的部分，用更合适的语言改写，比如3D游戏中的图形渲染模块，性能要求特别高，就可以用C/C++重写，而后封装为Python可以调用的扩展类库。需要注意的是在您使用扩展类库时可能需要考虑平台问题，某些可能不提供跨平台的实现。Python语法简洁清晰，特色之一是强制用空白符(white space)作为语句缩进。

语言优势：

（1）作为零基础非科班出身，python语法简单，适合我们阅读。Python本身就是一种脚本语言，简单易懂，Python的这种伪代码本质是它最大的优点之一。

（2）通俗易学，python放弃了c中非常复杂的指针，简化了python的语法。

（3）Python既支持面向过程的函数编程也支持面向对象的抽象编程。在面向过程的语言中，程序是由过程或仅仅是可重用代码的函数构建起来的。在面向对象的语言中，程序是由数据和功能组合而成的对象构建起来的。与其他主要的语言如C++和Java相比，Python以一种非常强大又简单的方式实现面向对象编程。

（4）Python标准库确实很庞大。python有可定义的第三方库可以使用。它可以帮助你处理各种工作，包括正则表达式、文档生成、单元测试、线程、数据库、网页浏览器、CGI、FTP、电子邮件、XML、XML-RPC、HTML、WAV文件、密码系统、GUI（图形用户界面）、Tk和其他与系统有关的操作。记住，只要安装了Python，所有这些功能都是可用的。这被称作Python的“功能齐全”理念。除了标准库以外，还有许多其他高质量的库，如wxPython、Twisted和Python图像库等等。

（5）可扩展性和可嵌入性。如果你需要你的一段关键代码运行得更快或者希望某些算法不公开，你可以把你的部分程序用C或C++编写，然后在你的Python程序中使用它们。你可以把Python嵌入你的C/C++程序，从而向你的程序用户提供脚本功能。

（6）由于它的开源本质，Python已经被移植在许多平台上（经过改动使它能够工作在不同平台上）。如果你小心地避免使用依赖于系统的特性，那么你的所有Python程序无需修改就可以在下述任何平台上面运行。

2.3.2 QT Creator软件介绍

Qt是一个跨平台应用程序和UI开发框架。使用Qt只需一次性开发应用程序，无须重新编写源代码，便可跨不同桌面和嵌入式操作系统部署这些应用程序。Qt Software的前身为创始于1994年的Trolltech（奇趣科技），Trolltech于2008年6月被 Nokia收购，加速了其跨平台开发战略。

Qt Creator是全新的跨平台Qt IDE（集成开发环境)，可单独使用，也可与Qt库和开发工具组成一套完整的SDK（软件开发工具包）。其中包括:高级C++代码编辑器、项目和生成管理工具、集成的上下文相关的帮助系统、图形化调试器、代码管理和浏览工具。

Qt应用程序可以移植到不同平台。根据不同平台的本地观感生成相应的本地应用。

2.3.3 GUI界面

用Qt Creator软件设计GUI界面，通过pyuic5 -o q.py mainwindow.ui指令转换为python代码。实现python设计GUI界面。GUI界面如图2.5所示。



图2.5 GUI界面

3 系统测试

## 3.1 软件测试

完成python代码编写，运行程序后，连接单片机与上位机，进入测试界面，打开串口，系统自动选择正确的串口与波特率，可设置时间选择班级，软件测试结果图如图3.1所示。

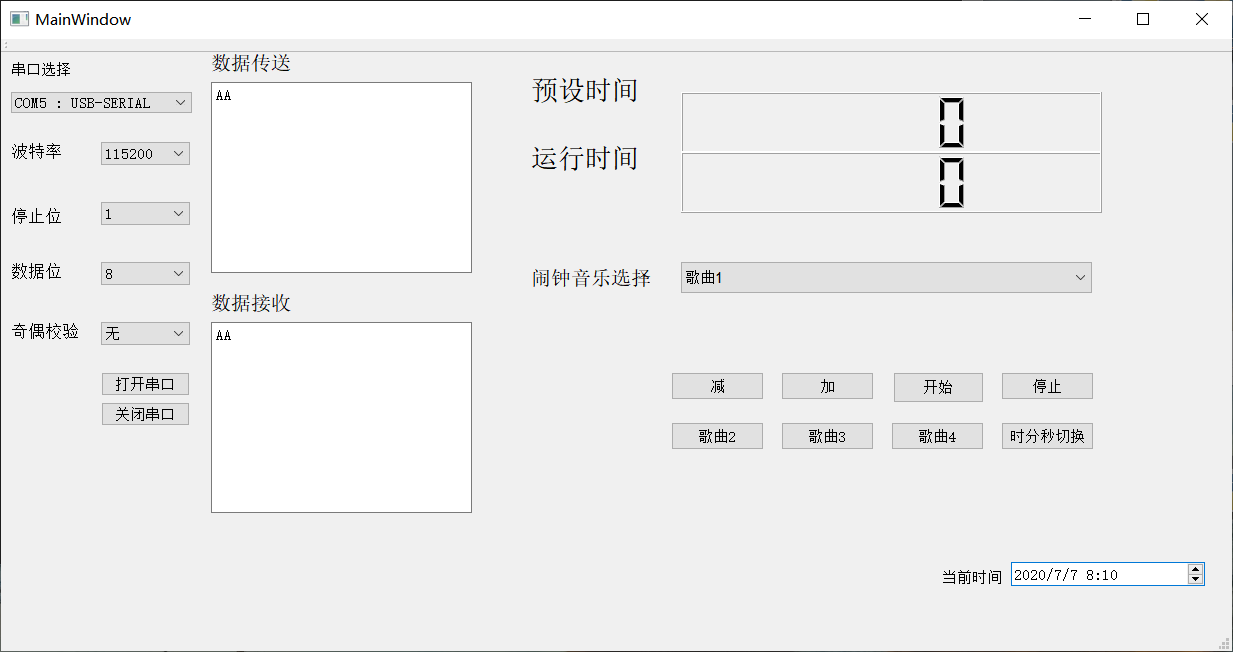


图3.1 软件测试结果图

## 3.2 硬件测试

下位机串口连接好，运用Keil-v5单片机程序编写软件进行主要功能的编写并且进行编译和烧录，与python程序配合编写从发射端接收过来的数据，在LCM1602模块上显示，通过按键可实现时间的加减、预设、开始、停止、时分秒切换和切歌。硬件测试结果图如图3.2所示。

图3.2 硬件测试结果图

结 论

本实训的研究工作已经基本达到预期的要求，通过了最后的硬件和软件的模拟调试。归纳起来有以下几点：

1. 本课题实现了通过STM32F103ZET6单片机实现了时钟计时，定时，蜂鸣器奏乐、切歌等一系列功能。
2. 系统软件设计包括程序初始化、模块摆放、代码修改、定时、功能处理以及ui转py等，软件操作方便和容易修改。
3. 系统硬件设计通过重点使用单片机设计下位机、python设计上位机，实现上下位机互通。
4. 硬件由蜂鸣器、按键、NRF24L01串口转USB无线模块、NRF24L01无线模块、LCD1602显示屏等组成。完成时分秒显示，时钟计时，定时，蜂鸣器奏乐、切歌等一系列功能。

在这次的实训设计中，设计也存在不足之处，在硬件设备上没有更好的得到实际操作，程序可能会在功能上出现缺失或不足。这确实是为由将理论知识切实应用时才能找出自己的不足之处。在今后的学习过程中唯有不断地发现错误，多进行动手操作，。以便可以更好的实际与理论相结合，使同学们能更深切的体会到不仅要有理论知识也要有实际操作能力，能灵活的学以致用，才能成为更好的自己。

本系统还可以进行改装，可以增加按键的使用进一步完善切歌功能实现一个按键切歌。进一步完善GUI界面的设计，更好的实现上、下位机的互通。

致 谢

感谢老师对我这次实训的悉心指导，对我的帮助很大。从实训开始到实训结束，经过老师的悉心教导，我完成了本次实训的内容。她严谨的治学作风给予我深深的影响，促使我在实训过程中精益求精。对老师的辛勤指导，呈上我最诚挚的谢意。

感谢两位老师，在实训这一个月中，不厌其烦的向我讲解实训，在您们真心的帮助下，使这次实训对我的意义更大了，您们不仅帮助我完成了这次实训，还为我们以后的毕业设计打下了良好的基础。您们不仅对我的课内知识进行了指导，还在思想以及道德上为我上了一堂课。

诚挚的感谢老师在这次实训中对我们得帮助，感谢老师在这一学期中对我们的支持与关怀。

参考文献

[1] 冯民昌.模拟集成电路系统[M].北京:中国铁道出版社，2016.8:15-17

[2]刘一.基于STM32的嵌入式系统设计[M].北京:中国铁道出版社，2016.9:33-35

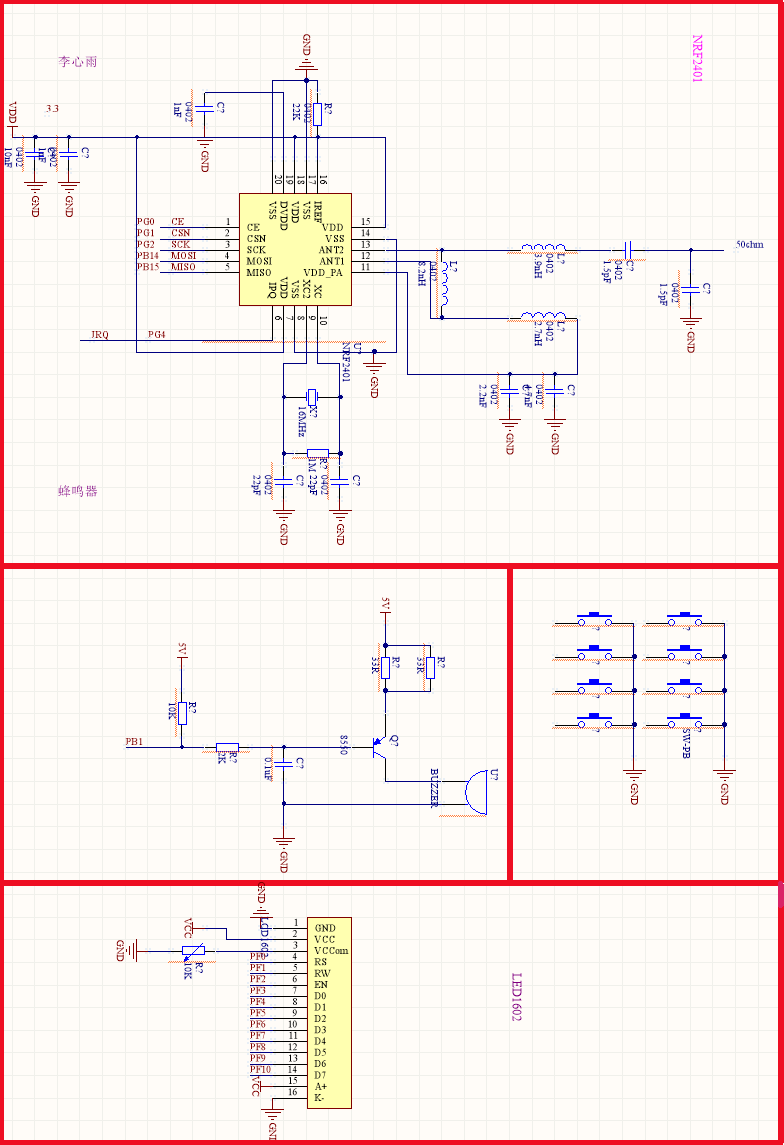
[3]陈明荧. ATmega16 单片机课程设计实训教材[M].北京:清华大学出版社，2018.8:36-39

[4]雷钢，王宏远.基于AVR单片机的控制系统设计[J].华中科技大学学报，2017.41(3):137-139

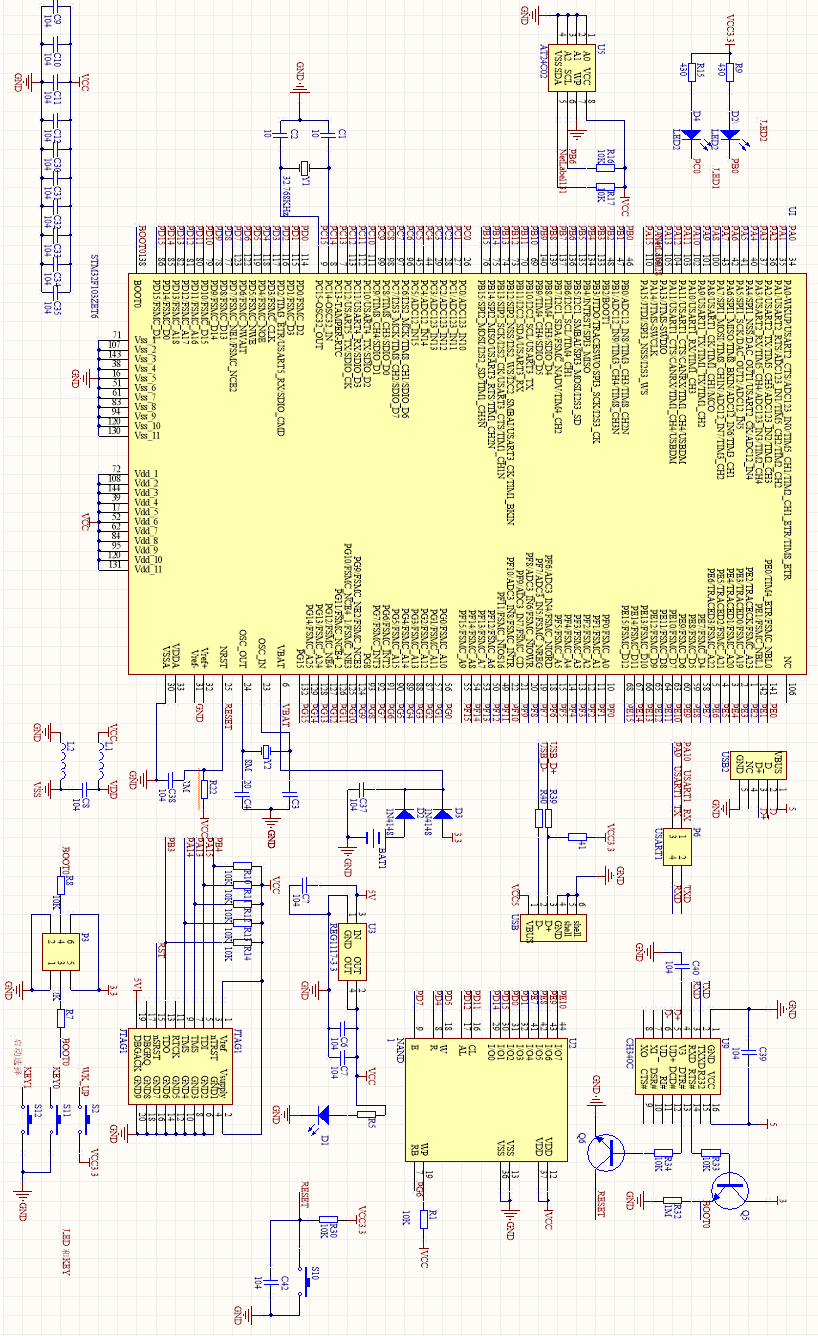
[5]陈朝大，韩剑.单片机原理与应用实验实训和课程设计[M].武汉:华中科技大学出版社，2017.6:52-53

# 附 录

硬件设计原理图如下图所示。



STM32F103ZET6设计原理图如下图所示。



程序主体代码：

while(1)  
 {  
    if(NRF\_IRQ==0)   //接收到数据

        {  
            if(NRF24L01\_RxPacket(tmp\_buf)==0)  //ÉèÖÃÎª½ÓÊÕÄ£Ê½  ½ÓÊÕÍê±Ï  
            {  
                if( tmp\_buf[1]=='1')   
                    printf("cxk");              
if( tmp\_buf[1]=='2')

 printf("2");  
            }    }  
      if(flag==1)  
{  
if((S\_Rcount+S\_Min\*60+S\_Hr\*3600)==(Rcount+B\*60+A\*3600))  
{  
TIM\_Cmd(TIM3,DISABLE);  
LCD1602\_Show\_Str(1,1,"                ");  
LCD1602\_Show\_Str(1,1,"TIME OUT!");  
printf("TIME OUT");  
printf("++++++++++++++++++++++++++++++++++++++++++++++");  
printf("\r\n");  
if(music==0)  
{     
    play\_music();  
   if(KEY3==0)  
   {  
     GPIO\_SetBits(GPIOB,GPIO\_Pin\_1);}  
     }  
if(music==1)  
{  
     play\_music2();  
     if(KEY3==0)  
     {  
      GPIO\_SetBits(GPIOB,GPIO\_Pin\_1);}  
    }  
if(music==2)  
{  
     play\_music2();  
     if(KEY3==0)  
     {  
      GPIO\_SetBits(GPIOB,GPIO\_Pin\_1);}  
    }  
if(music==3)  
{  
     play\_music2();  
     if(KEY3==0)  
     {  
      GPIO\_SetBits(GPIOB,GPIO\_Pin\_1);}  
    }  
S\_Hr=0;  
        S\_Min=0;  
        S\_Rcount=0;  
flag=0;  
Rcount=0;  
count=0;  
}}  
else  
{  
  if(KEY7==1) //启动  
  {  
  TIM\_Cmd(TIM3,ENABLE);  
flag=1;  
}  
if(KEY8==1) //增加预设时间  
  {  
  sprintf((char\*)ST,"%2d:%2d:%2d",S\_Hr,S\_Min,S\_Rcount);  
  LCD1602\_Show\_Str(1,0,"             ");  
  LCD1602\_Show\_Str(1,0,ST);  
  if(count==0)  
  {  
  S\_Rcount+=1;  
  }  
  if(count==1)  
  {  
  S\_Min+=1;  
  }  
  if(count==2)  
  {  
  S\_Hr+=1;  
  }  
printf("SET TIME:%2d:%2d:%2d\r\n",S\_Hr,S\_Min,S\_Rcount);  
  }  
  if(KEY9==1) //减  
   {  
sprintf((char\*)ST,"%2d:%2d:%2d:",S\_Hr,S\_Min,S\_Rcount);  
  LCD1602\_Show\_Str(1,0,"             ");  
  LCD1602\_Show\_Str(1,0,ST);  
  if(count==0)  
  {  
  S\_Rcount-=1;  
        if(S\_Rcount<=0)  
      {   
          S\_Rcount=0;  
        } }   
  if(count==1)  
  {  
   S\_Min-=1;  
        if(S\_Min<=0)  
      {   
          S\_Min=0;  
      } }   
  if(count==2)  
  {  
   S\_Hr-=1;  
        if(S\_Hr<=0)  
      {   
          S\_Hr=0;  
   } } }  
if(KEY11==1)//时分秒切换

{  
   count++;  
   if(count==3)  
{  
count=0;  
}}  
if(KEY10==1)  
{  
         music=1;  
}  
if(KEY12==1)//切歌  
{  
         music=2;  
}  
if(KEY13==1)  
{  
         music=3;  
}