# 电子设计课程实验报告

指导教师：\_\_\_\_\_\_\_\_\_

设计人员（学号）：\_\_\_\_\_\_\_\_

组号：\_\_\_\_\_\_\_\_

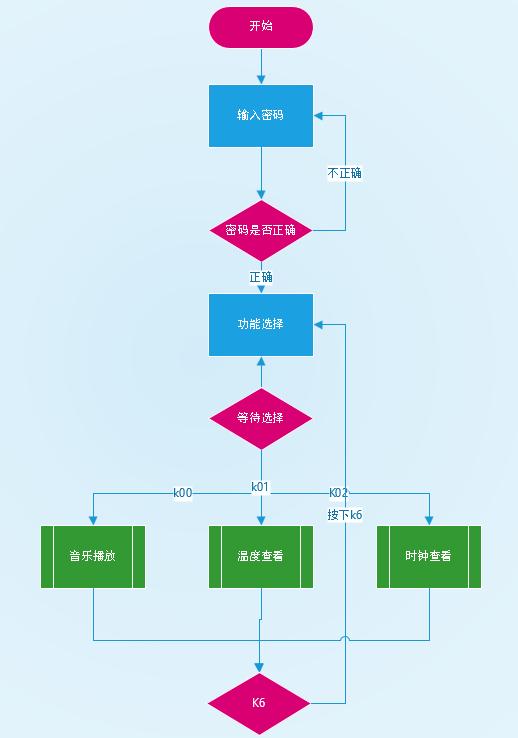
班级：\_\_\_\_\_\_\_\_

日期：\_\_\_\_\_\_\_\_

## 设计任务书

电子密码锁+音乐播放+温度检测+时钟显示

## 设计框图及整机概述



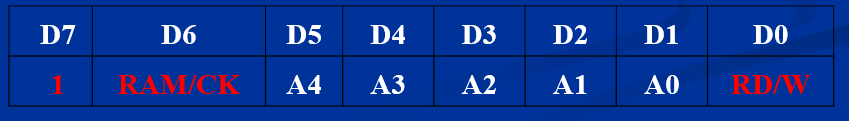
开机从4\*4键盘输入密码后数码管循环显示3个功能，按键k00, k01, k02分别进入音乐播放器， 温度显示， 时钟显示。按键k6退出当前程序进入菜单选择界面

## 各单元电路的设计方案及原理说明

首先设计DS1302时钟部分，DS1302是通过SPI串行总线跟单片机通信的，当进行一次读写操作时最少得读写两个字节，第一个字节是控制字节，就是一个命令，告诉DS1302是读还是写操作，是对RAM还是对CLOK寄存器操作。第二个字节就是要读或写的数据了。

单字节读写：只有在SCLK为低电平时，才能将CE置为高电平。所以在进行操作之前先将SCLK置低电平，然后将CE置为高电平，接着开始在IO上面放入要传送的电平信号，然后跳变SCLK。数据在SCLK上升沿时，DS1302读写数据，在SCLK下降沿时，DS1302放置数据到IO上。DS1302 的寄存器及片内RAM

控制寄存器用于存放DS1302的控制命令字，DS1302的RST引脚回到高电平后写入的第一个字就为控制命令。它用于对DS1302读写过程进行控制，它的格式如下：



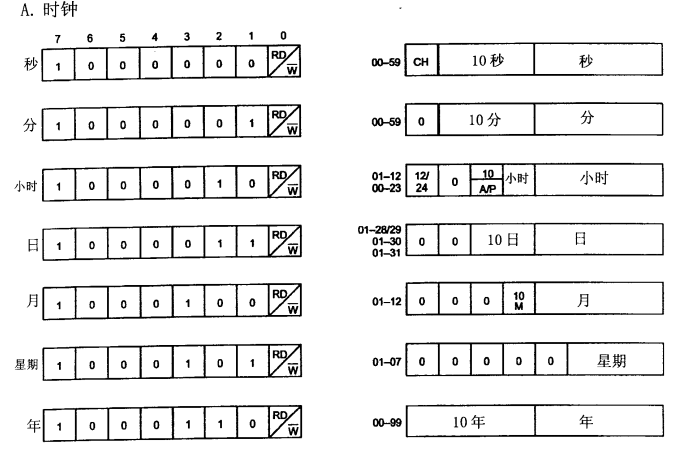
D7：固定为1

D6：RAM/CK位，=1片内RAM，=0日历、时钟寄存器选择位。

D5~D1：地址位，用于选择进行读写的日历、时钟寄存器或片内RAM。对日历、时钟寄存器或片内RAM的选择见表。

D0： 读写选择，=0写，=1读

日历 时钟寄存器



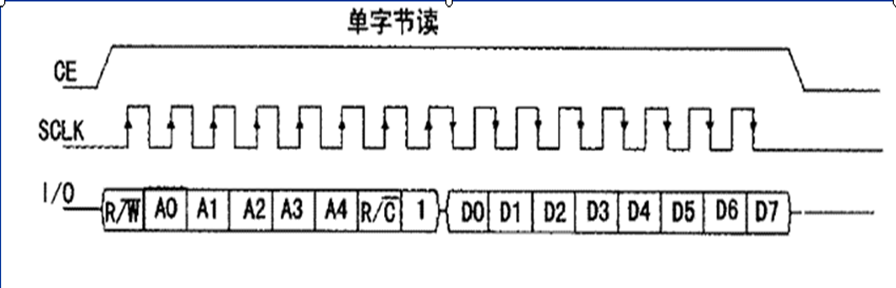
写保护寄存器

IMG_258

数据输出输入

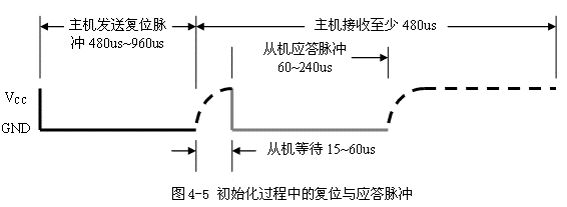
在控制指令字输入后的下一个SCLK时钟的上升沿时，数据被写入DS1302，数据输入从低位即位0开始。同样，在紧跟8位的控制指令字后的下一个SCLK脉冲的下降沿读出DS1302的数据，读出数据时从低位0位到高位7。

单字节读和写



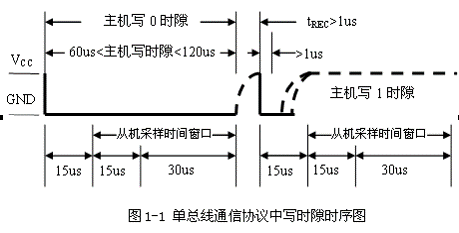
其次便是DS19B02温度传感器部分，DS18B20是一种单总线数字温度传感器，测试温度范围-55℃-125℃，具有体积小，硬件开销低，抗干扰能力强，精度高的特点。单总线，意味着没有时钟线，只有一根通信线。单总线读写数据是靠控制起始时间和采样时间来完成，所以时序要求很严格，这也是DS18B20驱动编程的难点。

1.单总线通信初始化



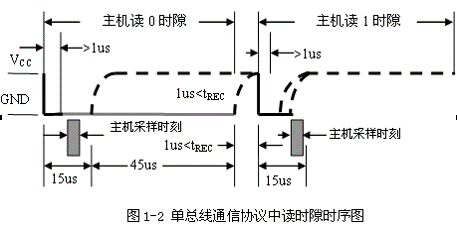
初始化时序包括：主机发出的复位脉冲和从机发出的应答脉冲。主机通过拉低单总线480-960μs产生复位脉冲；然后由主机释放总线，并进入接收模式。主机释放总线时，会产生一由低电平跳变为高电平的上升沿，单总线器件检测到该上升沿后，延时15～60μs，接着单总线器件通过拉低总线60～240μsμ来产生应答脉冲。主机接收到从机的以应答脉冲后，说明有单总线器件在线，到此初始化完成。然后主机就可以开始对从机进行ROM命令和功能命令操作。

2.位写入时序



写时隙：当主机把数据线从逻辑高电平拉到逻辑低电平的时候，写时间隙开始。有两种写时间隙：写1的时间隙和写0时间隙。所有写时间隙必须最少持续60us,包括两个写周期间至少1us的恢复时间。DQ引脚上的电平变低后，DS18B20在一个15us到60us的时间窗口内对DQ引脚采样。如果DQ引脚是高电平，就是写1，如果DQ引脚是低电平，就是写0。主机要生成一个写1时间隙，必须把数据线拉到低电平然后释放，在写时间隙开始后的15us内允许数据线拉到高电平。主机要生成一个写0时间隙，必须把数据线拉到低电平并保持60us。

3.位读取时序



当主机把总线从高电平拉低，并保持至少1us后释放总线；并在15us内读取从DS18B20输出的数据。

然后是音乐播放模块，

想用蜂鸣器模拟出音乐，就需要先把乐谱转换为十六进制数，简单看来音乐就是高低不一，长短不一声的音间隔不同时间的排列组合，所以乐谱改编成十六进制就是三个要素：音符即DO,RE,MI,FA,SO,LA,SI这七个不同音符，音高即高中低三种音，节拍即音符之间的间隔时长.

每个音符使用1个字节，字节的高4位代表音符的高低，低4位代表音符的节拍，下表为节拍码的对照。但如果1拍为0.4秒，1/4拍是 0.1秒，只要设定延迟时间就可求得节拍的时间。假设1/4节拍为1DELAY，则1拍应为4DELAY，以此类推。所以只要求得1/4拍的 DELAY时间，其余的节拍就是它的倍数，如下表为1/4和1/8节拍的时间设定。

重置的时候要把音乐的数组重置，同时定时器蜂鸣器要复位，共用了定时器0，要重新设置计时方式。

最后是登陆模块，程序启动后等待4\*4键盘输入密码，每按下一位按键便保存一位，从左至右依次循环，同时不断比较当前已经输入的密码是否与设定密码一致，一致则进入菜单选择页面，循环显示三个功能，同时等待1\*4按键响应。

## 调试过程，结果分析及调试体会

与其他模块一起调试的时候总是初始化了时间，因此只有第一次烧入时需要设置初始时间，之后便需要将初始化时间的代码注释掉。

如果温度是一个负温度，要将读到的数据减一再取反

在推出模块时有些界面不能及时推出（比如播放音乐以及显示时钟时），调试发现是由于没有在定时器中的显示数码管函数中加入退出判断，致使显示的时候不能响应推出至菜单选择页面

#### 对本次课程设计的意见及建议

课程总体体验很好，如果能多点扩展比如外置存储器可能会更好。