

---

# 基于 arduino+LCD1602 的实时时钟设计

设计者：STCode

## 1、 实时时钟功能介绍

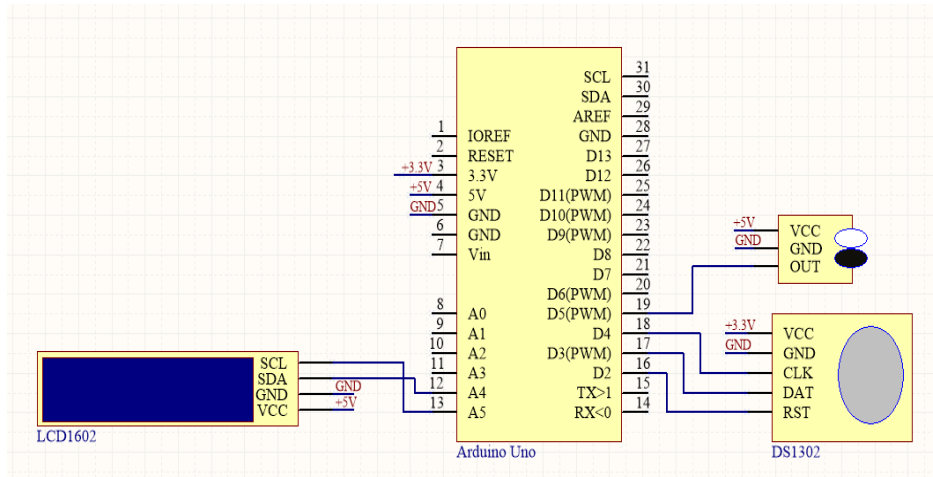
利用 DS1302 时钟模块，获取时间信息，并且通过 IIC 通讯的 LCD1602 显示出来，可以显示出当前的年月日时分秒数据，并且结合红外传感器，当红外感应到人时，开启背光灯，否则，关闭背光灯。

## 2、 使用主要器件

- 1. Arduino Uno 控制板
- 2. LCD1602 液晶显示屏
- 3. DS1302 时钟模块
- 4. 红外传感器模块
- 5. 杜邦线

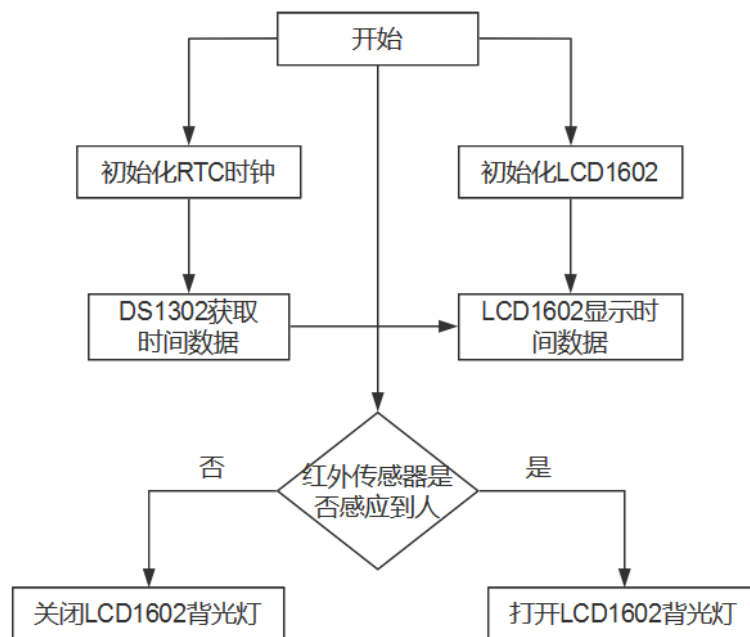
## 3、 电路设计

序号	Arduino Uno	DS1302
1	+3.3V	VCC
2	GND	GND
3	D2	RST
4	D3	DAT
5	D4	SCL
	Arduino Uno	LCD1602
6	+5V	VCC
7	GND	GND
8	A5	SCL
9	A4	SDA
	Arduino Uno	红外传感器
10	+5V	V
11	GND	GND
12	D5	S



#### 4、 程序设计

程序部分主要分为时间获取程序、LCD1602 显示程序、红外传感器感应程序。程序流程如下



#### 5、 编译和测试

##### 1、 编译结果截图

```
sketch_jun14a | Arduino 1.8.5
文件 编辑 项目 工具 帮助

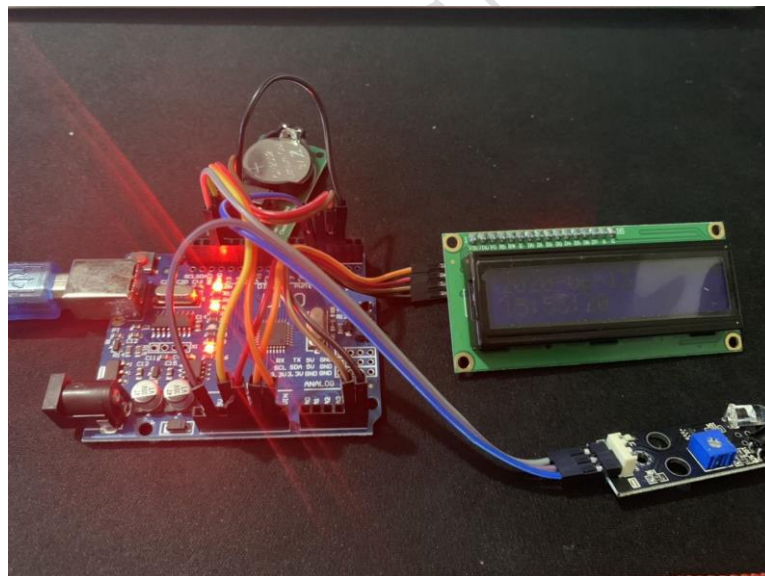
sketch_jun14a $
39 printTime(); //打印时间子函数
40 if(digitalRead(sensopin)==LOW)
41 {
42     lcd.backlight();
43     lcd.setCursor(0,0); //设置显示位置
44     lcd.print(buf1); //输出字符
45     lcd.setCursor(0,1);
46     lcd.print(buf2);
47 }
48 else if(digitalRead(sensopin)==HIGH)
49     lcd.noBacklight();
50
51 }
```

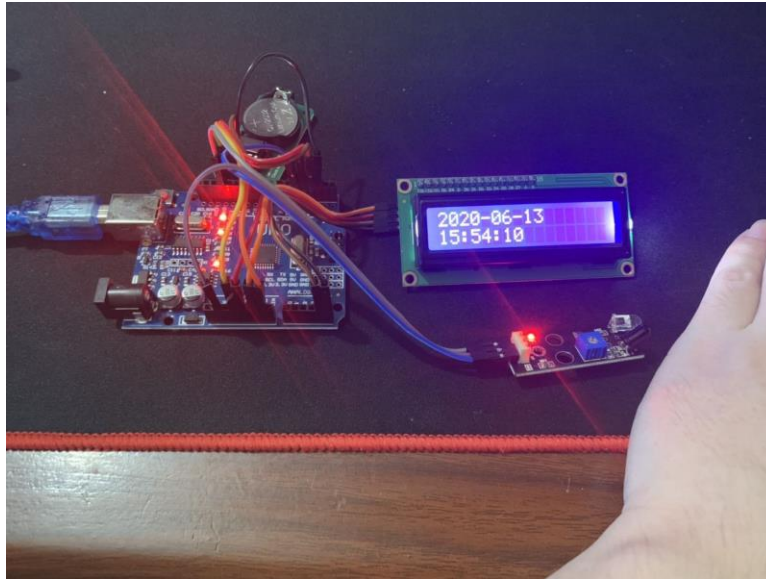
编译完成。

项目使用了 6576 字节，占用了 (20%) 程序存储空间。最大为 32256 字节。  
全局变量使用了532字节，(25%)的动态内存，余留1516字节局部变量。最大为2048字节。

43 Arduino/Genuino Uno @ COM0

## 2、测试图片





## 6、程序清单

```
7、 #include <DS1302.h>    //DS1302 头文件
8、 #include <Wire.h>      //Wire 头文件
9、 #include <LiquidCrystal_I2C.h> //引用 I2C 通讯的 LCD1602 库
10、 int sensorpin=5;    //定义红外传感器引脚
11、 DS1302 rtc(2, 3, 4); //对应 DS1302 的 RST,DAT,CLK
12、 LiquidCrystal_I2C lcd(0x27,16,2); //设置 LCD1602 设备地址,一般为 0x3F、
    0x20、0x27, 可以查看硬件手册
13、 char buf1[50];    //定义字符 1
14、 char buf2[50];    //定义字符 2
15、 void initRTCTime(void) //初始化 RTC 时钟
16、 {
17、     rtc.writeProtect(false);    //关闭写保护
18、     rtc.halt(false);            //清除时钟停止标志
19、     Time tt(2020, 6, 13, 15, 51, 30, 7); //创建时间对象,参数 7 为星期数
    数据,周日为 1,周一为 2 以此类推
20、     rtc.time(tt); //向 DS1302 设置时间数据
21、 }
22、
23、 void printTime() //打印时间数据
24、 {
25、     Time tim = rtc.time(); //从 DS1302 获取时间数据
26、     snprintf(buf1, sizeof(buf1), "%04d-%02d-%02d", tim.yr, tim.mon,
    tim.date);    //字符 1 为年月日数据
27、     snprintf(buf2, sizeof(buf2), "%02d:%02d:%02d", tim.hr, tim.min,
    tim.sec);    //字符 2 为时分秒数据
28、     Serial.println(buf1);    //串口打印数据
29、     Serial.println(buf2);
30、 }
```

```
31、 void setup()
32、 {
33、     Serial.begin(9600);    //设置串口波特率
34、     pinMode(sensonpin,INPUT_PULLUP);
35、     lcd.init();           // 初始化 LCD
36、     lcd.backlight();       //设置 LCD 背景等亮
37、     initRTCTime();
38、     //新模块上电需要设置一次当前时间，
39、     //下载完成后需屏蔽此函数再次下载，否则每次上电都会初始化时间数据
40、
41、 }
42、
43、 void loop()
44、 {
45、     printTime();    //打印时间子函数
46、     if(digitalRead(sensonpin)==LOW)
47、     {
48、         lcd.backlight();
49、         lcd.setCursor(0,0); //设置显示位置
50、         lcd.print(buf1);    //输出字符
51、         lcd.setCursor(0,1);
52、         lcd.print(buf2);
53、     }
54、     else if(digitalRead(sensonpin)==HIGH)
55、         lcd.noBacklight();
56、
57、 }
```