基于arduino+LCD1602的实时时钟设计

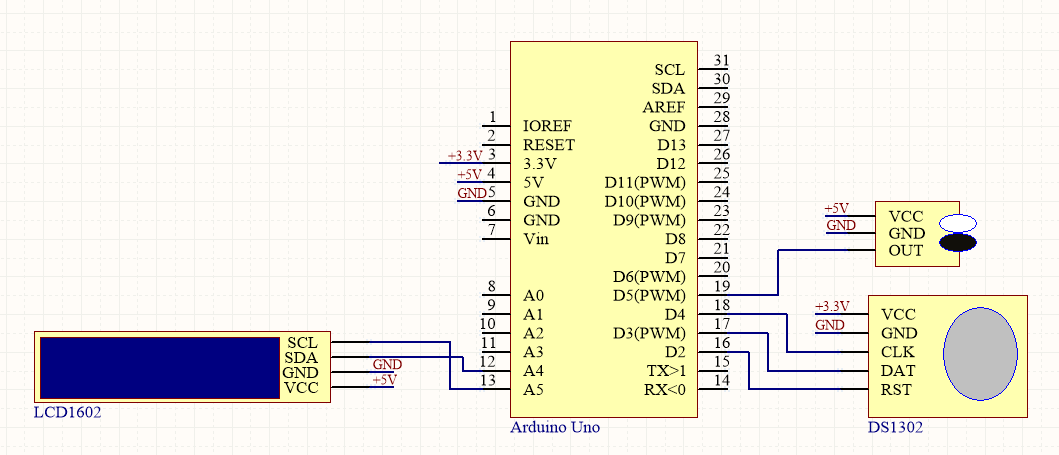
设计者：STCode

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。

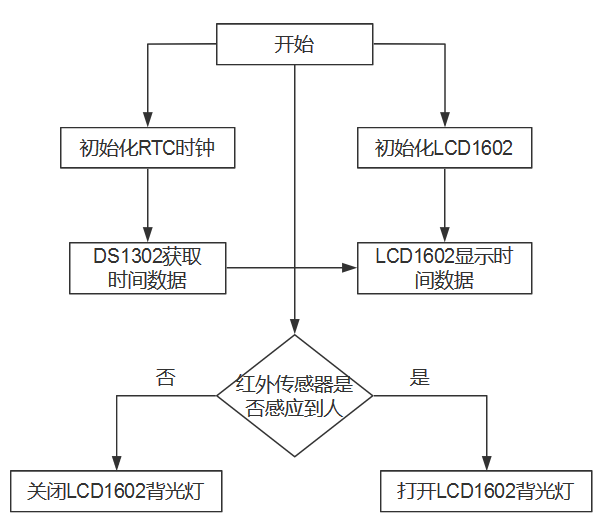
1. 使用主要器件
2. Arduino Uno控制板
3. LCD1602液晶显示屏
4. DS1302 时钟模块
5. 红外传感器模块
6. 杜邦线
7. 电路设计

|  |  |  |
| --- | --- | --- |
| 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 |



1. 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

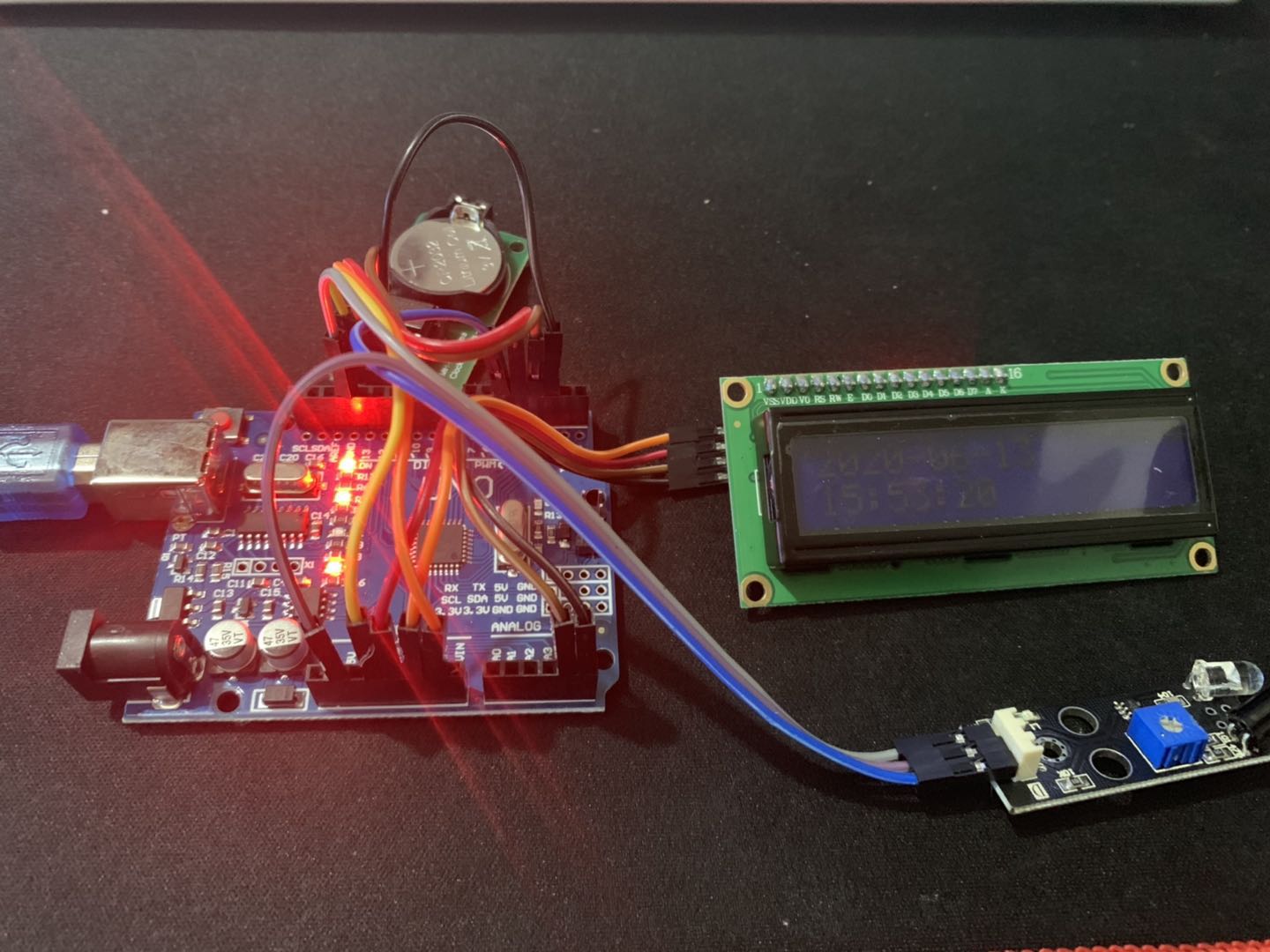


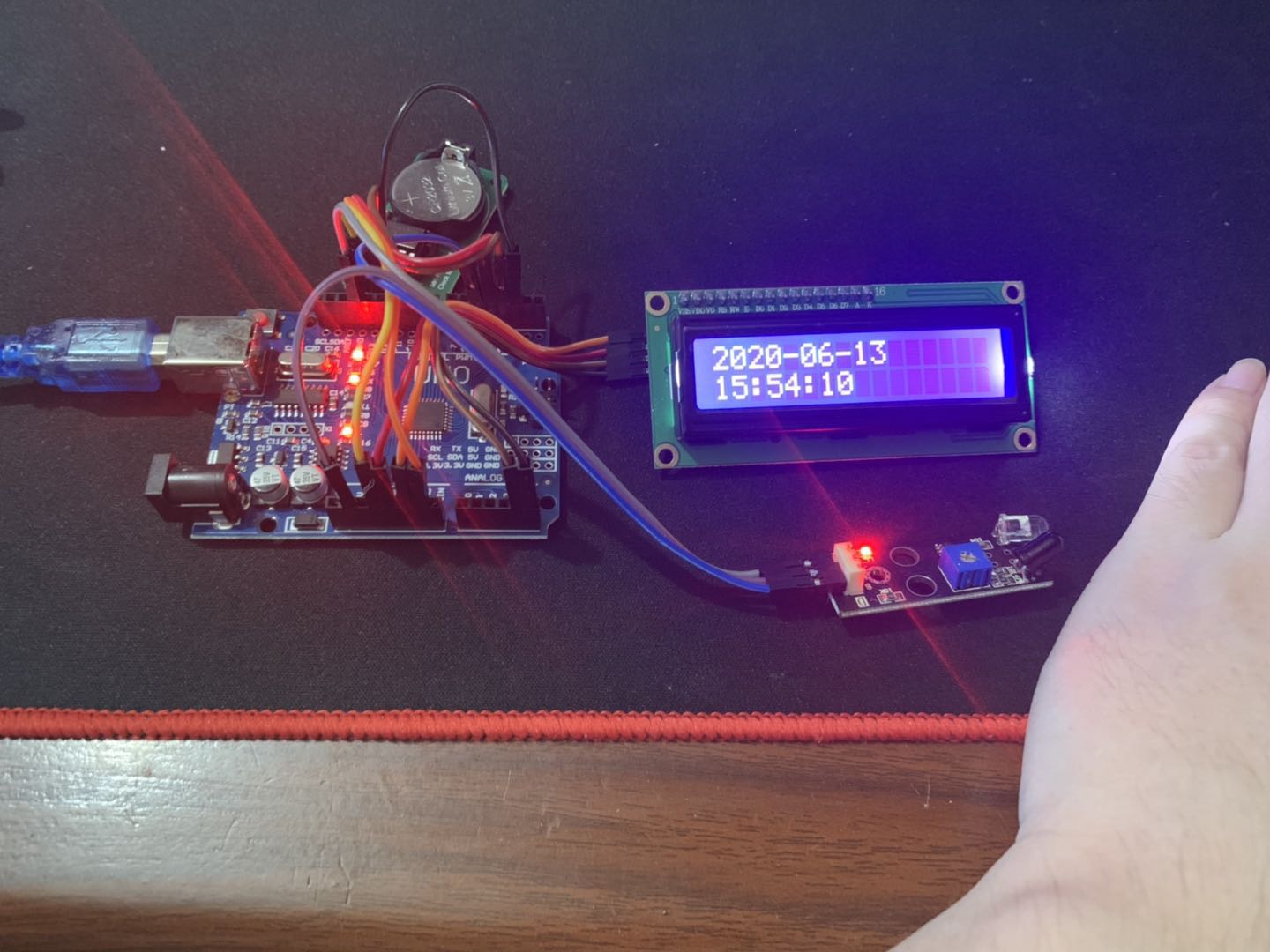
1. 编译和测试

1、编译结果截图



2、测试图片





1. 程序清单
2. #include <DS1302.h>   //DS1302头文件
3. #include <Wire.h>     //Wire头文件
4. #include <LiquidCrystal\_I2C.h> //引用I2C通讯的LCD1602库
5. int sensonpin=5;   //定义红外传感器引脚
6. DS1302 rtc(2, 3, 4); //对应DS1302的RST,DAT,CLK
7. 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
8. char buf1[50];  //定义字符1
9. char buf2[50];  //定义字符2
10. void initRTCTime(void)//初始化RTC时钟
11. {
12. rtc.writeProtect(false);   //关闭写保护
13. rtc.halt(false);           //清除时钟停止标志
14. Time tt(2020, 6, 13, 15, 51, 30, 7); //创建时间对象，参数7为星期数据，周日为1，周一为2以此类推
15. rtc.time(tt);//向DS1302设置时间数据
16. }
17. void printTime()//打印时间数据
18. {
19. Time tim = rtc.time(); //从DS1302获取时间数据
20. snprintf(buf1, sizeof(buf1), "%04d-%02d-%02d",tim.yr, tim.mon, tim.date);   //字符1为年月日数据
21. snprintf(buf2, sizeof(buf2), "%02d:%02d:%02d",tim.hr, tim.min, tim.sec);    //字符2为时分秒数据
22. Serial.println(buf1);   //串口打印数据
23. Serial.println(buf2);
24. }
25. void setup()
26. {
27. Serial.begin(9600);   //设置串口波特率
28. pinMode(sensonpin,INPUT\_PULLUP);
29. lcd.init();           // 初始化LCD
30. lcd.backlight();      //设置LCD背景等亮
31. initRTCTime();
32. //新模块上电需要设置一次当前时间，
33. //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
34. }
35. void loop()
36. {
37. printTime();    //打印时间子函数
38. if(digitalRead(sensonpin)==LOW)
39. {
40. lcd.backlight();
41. lcd.setCursor(0,0);  //设置显示位置
42. lcd.print(buf1);     //输出字符
43. lcd.setCursor(0,1);
44. lcd.print(buf2);
45. }
46. else if(digitalRead(sensonpin)==HIGH)
47. lcd.noBacklight();
48. }