多功能机器人设计与实现

19130104 计科(卓工) 顾佳凯

# 1.概述

## 课题背景

基于人类活动感知力的自主调光机器人，增添数码管滚动显示数字模块，使其功能更多样。

## 问题提出

由于本人arduino的编程能力欠佳，故我选择在前面课程作业的基础上增添新模块，作为单片机课程的课程设计。

## 国内外现状

* 国外

Arduino官网的最新Blog中报道了一位Arduino爱好者Bruton，他制作了由Arduino控制的平衡背包，成功解决了走钢丝的问题。[1]

Arduino爱好者dumont，收到了Arduino Uno迷你限量版项链后，将其放在了一个有趣的可穿戴设备的显著位置，避免了大多数硬件的命运—安全地存放在盒子里或放在架子上展示。[2]

罗马尼亚的一个村庄，搭建了一个使用Arduino Nano作为控制板的系统，该系统可以连续监测七户人家的水位，不再需要人工监测。 [3]

* 国内

当字节跳动的Tiktok走向了世界，不禁感慨何时中国的硬件也能崛起？硬件一直是国内的一个短板。

Arduino在国内起步较晚，在知网发现了一篇较为有趣的Arduino文献，该文献的作者是一名广东省的高中生，他基于arduino设计了智能监控抓拍器，用于捕获偷用电脑的证据。[4]

国内有学者借助立创EDA平台设计了系统硬件电路，使用Arduino IDE软件编写系统控制值程序，实现对空气中酒精浓度的实时检测。该车辆防酒驾系统被验证是可靠的。[5]

国内有研究人员以Arduino控制板为核心，利用软件编程实现了对小车的蓝牙控制。[6]

## 创新点

本课程设计无创新点。虽说创新是人类进步的阶梯，但殊不知创新是建立在你对该学科充分理解的基础上进行的学术活动。我目前的水平对于单片机这门课程最多算是入了个门，也可能门还没入，故不考虑创新点。

# 需求分析

## 功能需求

设计并制作一个桌面机器人，有如下功能：

* 加入光敏电阻器，使得机器人能监控环境光的强弱。
* 加入PIR传感器，使得机器人能具备检测有效范围内的人类活动。
* 当机器人检测到有效范围内有人类活动，则LED灯亮起。
* 当机器人检测到有效范围内有人类活动，并且此刻的光强度过大，则发生报警。

## 非功能性需求

* 数码管滚动显示数字模块，见证时间流逝。

## 可行性分析

* 技术可行性

基于Arduino编程可实现上述需求，由于某些功能模块在平时练习中出现过，故可复用。技术上可行。

* 投资可行性

本课程设计无创新点，故不需要投资。

* 财务可行性

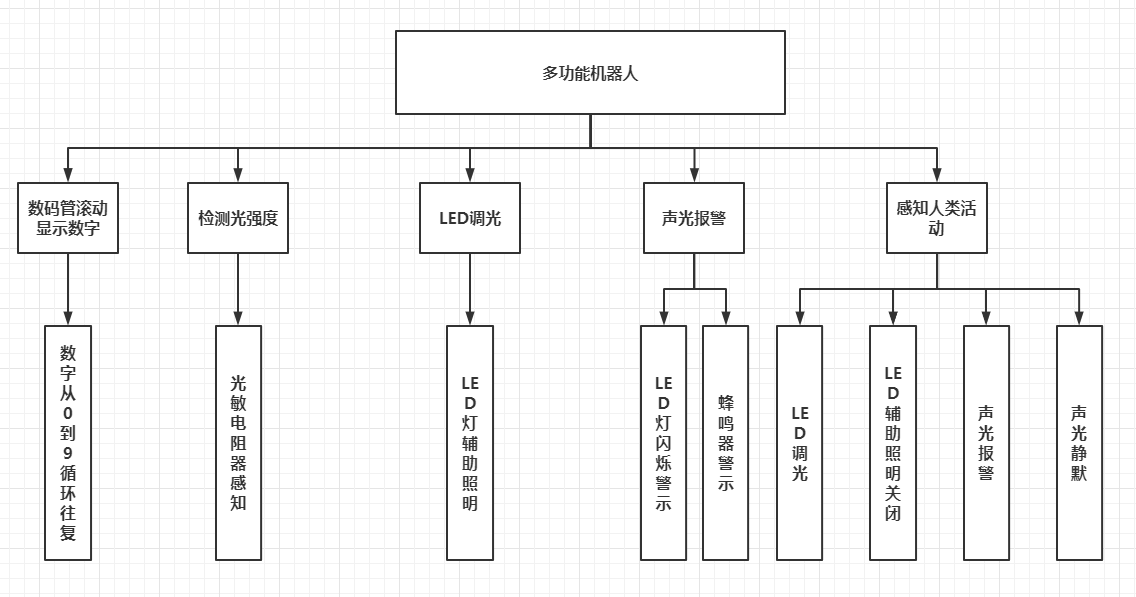
本课程设计仅需仿真平台Tinkpad即可完成，故无需考虑财务问题。

* 组织可行性

本课程设计由我按步就班完成，定能保证在截止日期前交付。

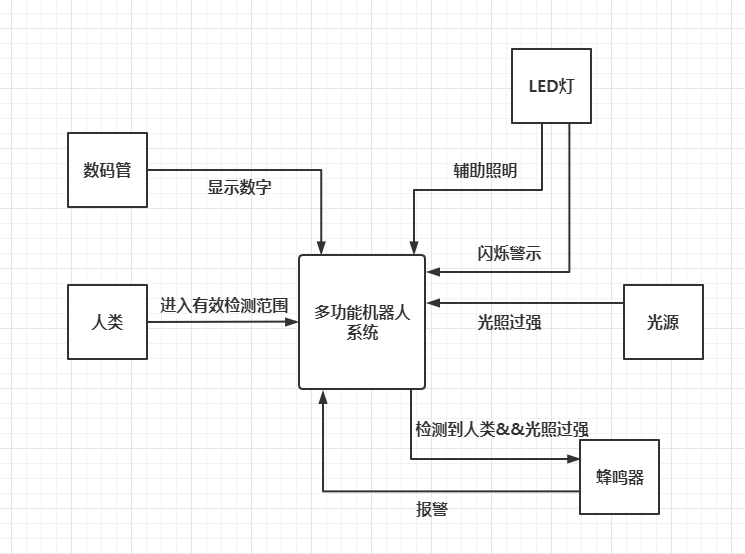
# 3.总体设计

## 3.1系统功能框图

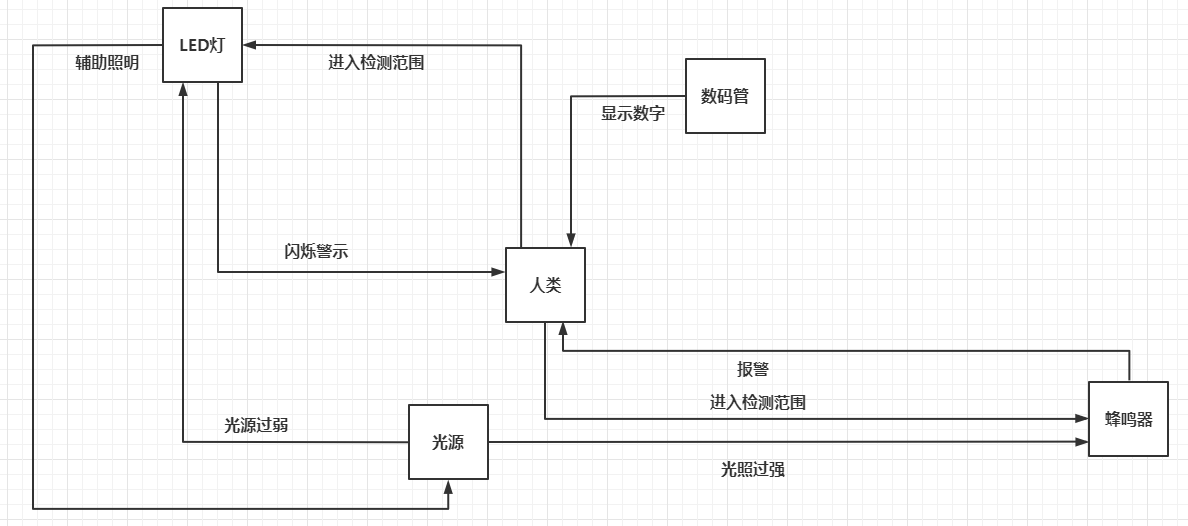


## 3.2.数据流图

* 顶层数据流图

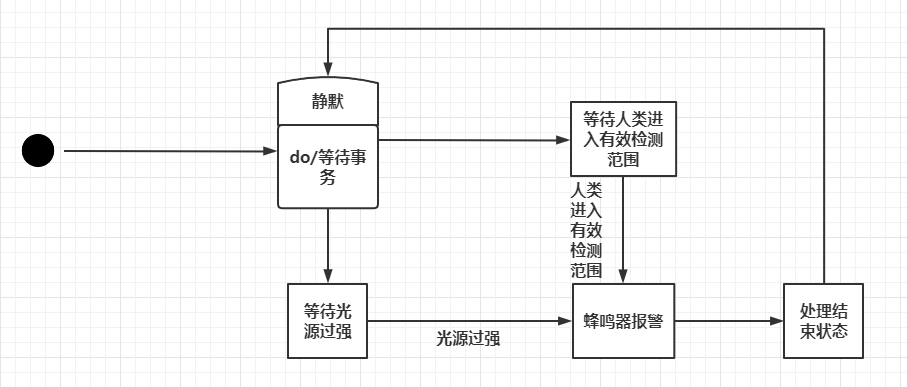


* 0层数据流图

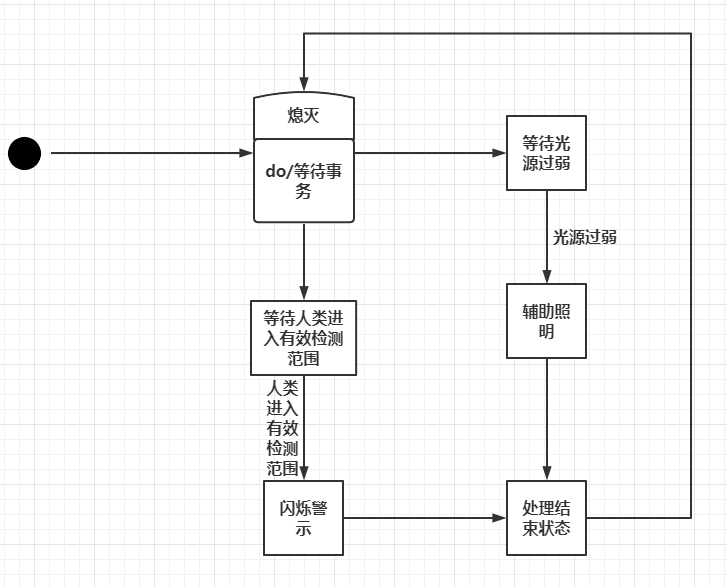


## 3.3.状态图

* 蜂鸣器的状态转换图



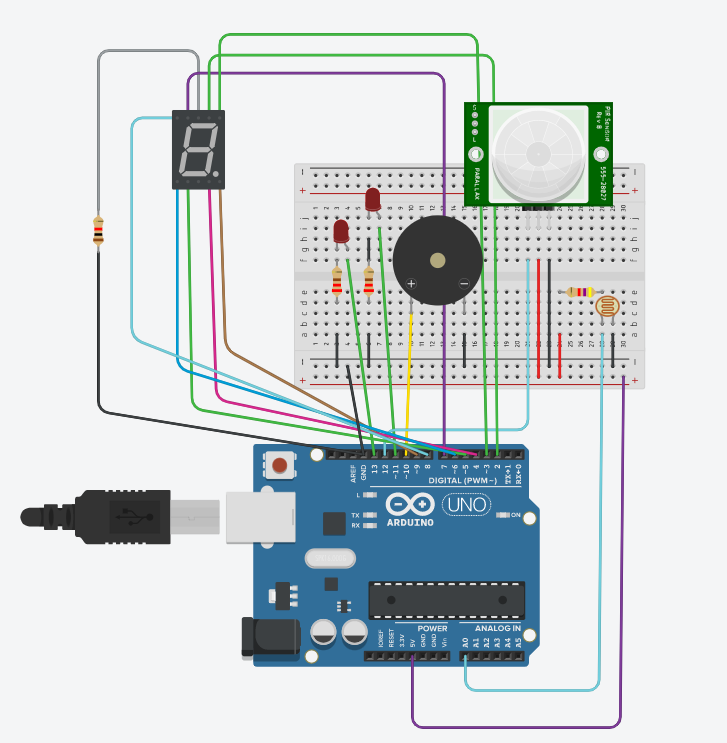
* LED的状态转换图



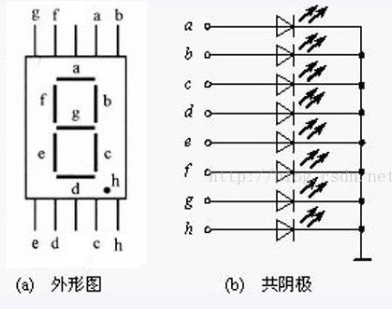
# 4.详细设计

## 4.1.硬件设计

* 完整电路图



* 原理图、元器件清单

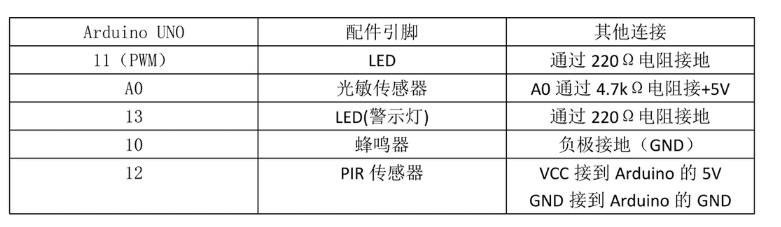


具有人类活动感知功能的自适应调光机器人在之前实验的基础上增

加了PIR传感器。PIR具有三个引脚，信号口与ArduinoUNO的12

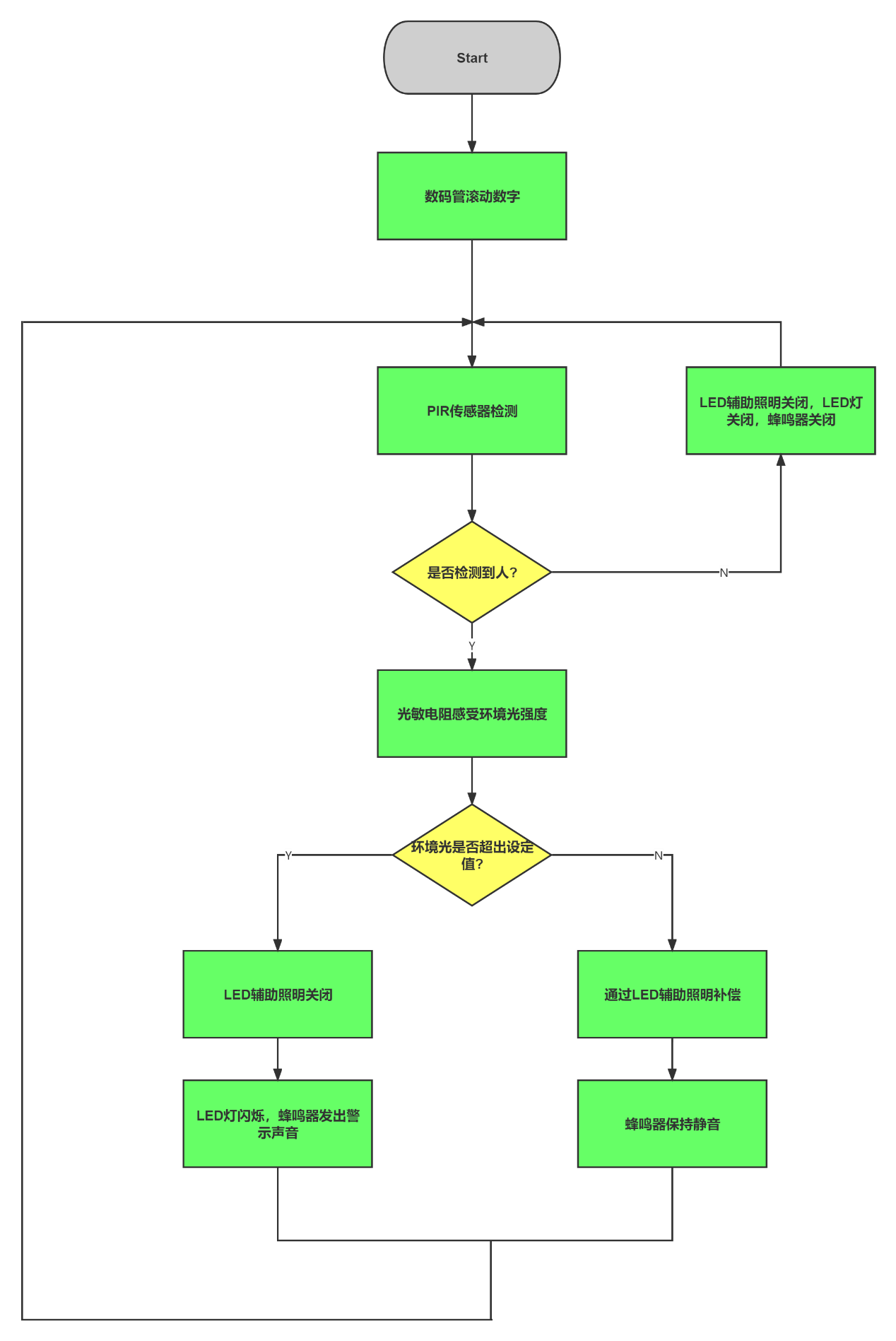
引脚电路连接，VCC和GND分别接入ArduinoUNO的VCC和GND。具

体电路端口分配如下表所示。



## 4.2.软件设计

* 系统程序流程图



* 代码描述

#define LED 13

#define Buzzer 10

#define PIR\_pin 12

//分配给数码管的数字引脚

**const** byte pins[] = {9, 8, 7, 6, 5, 4, 3, 2};

//将Arduino控制数码管的端口初始化

**void** Init\_seg()

{

**for** (**int** i = 0; i < 8; i++)

    {

        pinMode(pins[i], OUTPUT); //设置为输出模式

    }

}

//用字节数组存储0~9数字所对应符号编码

byte DIGI\_DISP[] = {

    // 前缀B 表示二进制数

    B11111100, //=0(自左向右依次对应a,b,c,d,e,f,g,dp）

    B01100000, // 1

    B11011010, // 2

    B11110010, // 3

    B01100110, // 4

    B10110110, // 5

    B10111110, // 6

    B11100000, // 7

    B11111110, // 8

    B11110110, // 9

};

//显示函数

//参数num：用户需要显示的符号，取值范围：0~255

**void** display\_Num(byte num)

{

**for** (**int** cnt = 0; cnt < 8; cnt++)

    {

        //读出每一位，并输出到对应的引脚

        digitalWrite(pins[cnt], bitRead(DIGI\_DISP[num], cnt));

    }

}

//清屏函数

//参数dly:清屏所需时长，单位：毫秒

**void** Clean\_scrn(**int** dly)

{

**for** (**int** cnt = 0; cnt < 8; cnt++)

    {

        //所有位都为低电平，LED都断电

        digitalWrite(pins[cnt], 0);

    }

    delay(dly);

}

**void** setup()

{

    Init\_LDR();

    Init\_Light();

    Init\_Buzzer();

    Init\_PIR();

    Init\_seg(); // Arduino对数码管的端口初始化

}

**void** loop()

{

    // 0~9数字依次显示

**for** (**int** cnt = 0; cnt <= 9; cnt++)

    {

**if** (Val\_PIR())

        {

**if** (Val\_LDR() < 150)

            {

                LED\_Off();

                Alarm();

            }

**else**

            {

                display\_Num(cnt); //显示指定的字符

                delay(1000);      //字符显示1秒钟

                //显示内容切换期间，执行清屏

                Clean\_scrn(30);

                LED\_PWM(Val\_LDR());

                Mute();

            }

        }

**else**

        {

            display\_Num(cnt); //显示指定的字符

            delay(1000);      //字符显示1秒钟

            //显示内容切换期间，执行清屏

            Clean\_scrn(30);

            LED\_Off();

            Mute();

        }

    }

}

**void** Init\_LDR()

{

}

**void** Init\_Buzzer()

{

    pinMode(LED, OUTPUT);

    pinMode(Buzzer, OUTPUT);

}

**void** Alarm()

{

**int** cnt = 0;

**for** (**int** Hz = 200; Hz <= 800; Hz++)

    {

        cnt++;

**if** (cnt == 40)

        {

            cnt = 0;

            digitalWrite(LED, !digitalRead(LED)); // LED状态翻转

        }

        tone(Buzzer, Hz);

        delay(5);

    }

    delay(1000);

**for** (**int** Hz = 800; Hz >= 200; Hz--)

    {

        cnt++;

**if** (cnt == 40)

        {

            cnt = 0;

            digitalWrite(LED, !digitalRead(LED)); // LED状态翻转

        }

        tone(Buzzer, Hz); //向蜂鸣器输出频率

                          //     play\_Joy();

                          //     delay(5);

        delay(5);

    }

}

**void** Mute()

{

    noTone(Buzzer);

    digitalWrite(Buzzer, LOW);

    digitalWrite(LED, LOW);

}

**int** Val\_LDR()

{

**int** value, out;

    value = analogRead(A0);

    out = map(value, 0, 1023, 0, 255);

**return** out;

}

**void** Init\_Light()

{

    pinMode(11, OUTPUT);

}

**void** LED\_On()

{ //开灯

    digitalWrite(11, HIGH);

}

**void** LED\_Off()

{ //关灯

    digitalWrite(11, LOW);

}

**void** LED\_PWM(byte degree)

{

    analogWrite(11, degree);

}

**void** Init\_PIR()

{

    pinMode(PIR\_pin, INPUT);

}

**bool** Val\_PIR()

{

**bool** val;

    val = digitalRead(PIR\_pin);

**return** (val);

}

# 5.系统调试与实现

## 5.1.单元测试

* 数码管

/\*

 \*显示驱动程序设计

 \*功能：

 \*1.数码管相关的硬件初始化

 \*2.数码管常用显示的字符编码库

 \*3.数码管的显示函数

 \*4.数码管清屏函数

 \*/

//分配给数码管的数字引脚

**const** byte pins[]={9,8,7,6,5,4,3,2};

//将Arduino控制数码管的端口初始化

**void** Init\_seg(){

**for**(**int** i=0;i<8;i++){

    pinMode(pins[i],OUTPUT);  //设置为输出模式

  }

}

//用字节数组存储0~9数字所对应符号编码

byte DIGI\_DISP[]={// 前缀B 表示二进制数

  B11111100, //=0(自左向右依次对应a,b,c,d,e,f,g,dp）

  B01100000, //1

  B11011010, //2

  B11110010, //3

  B01100110, //4

  B10110110, //5

  B10111110, //6

  B11100000, //7

  B11111110, //8

  B11110110, //9

};

//显示函数

//参数num：用户需要显示的符号，取值范围：0~255

**void** display\_Num(byte num){

**for** (**int** cnt=0;cnt<8;cnt++){

    //读出每一位，并输出到对应的引脚

    digitalWrite(pins[cnt],bitRead(DIGI\_DISP[num],cnt));

  }

}

//清屏函数

//参数dly:清屏所需时长，单位：毫秒

**void** Clean\_scrn(**int** dly)

{

**for**(**int** cnt=0;cnt<8;cnt++)

  {

    //所有位都为低电平，LED都断电

    digitalWrite(pins[cnt],0);

  }

  delay(dly);

}

/\*

 \*主程序设计

 \*功能：

 \*共阴极数码管滚动显示“0~9”

 \*美化显示效果，每次显示结果清屏

 \*/

**void** setup()

{

 Init\_seg();  //Arduino对数码管的端口初始化

}

**void** loop()

{

  //0~9数字依次显示

**for**(**int** cnt=0;cnt<=9;cnt++){

    display\_Num(cnt);  //显示指定的字符

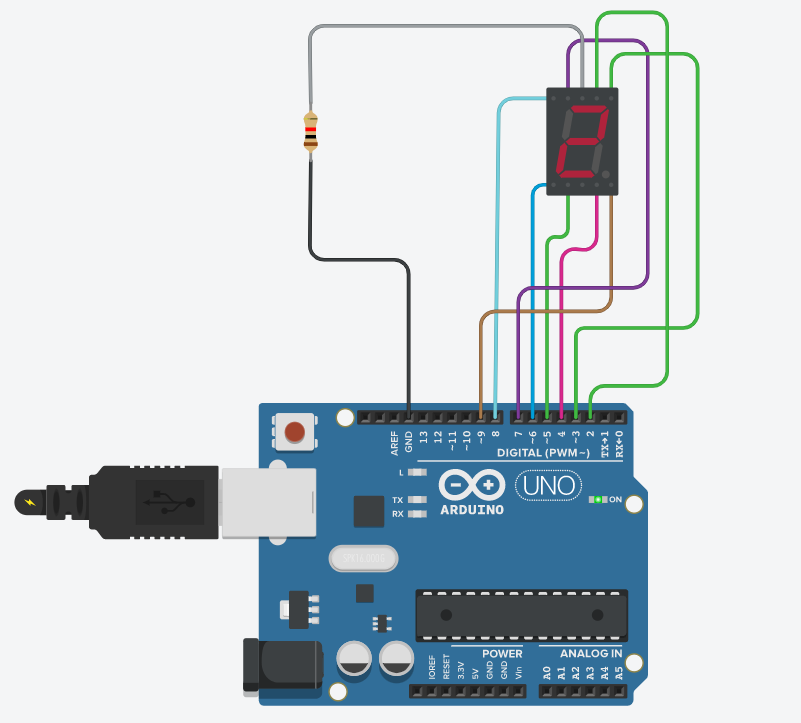
    delay(1000);  //字符显示1秒钟

    //显示内容切换期间，执行清屏

    Clean\_scrn(30);

  }

}



运行结果正常。

* 具有人类活动感知力的自主调光机器人

#define LED    13

#define Buzzer 10

#define PIR\_pin 12

**void** setup()

{

  Init\_LDR();

  Init\_Light();

  Init\_Buzzer();

  Init\_PIR();

}

**void** loop()

{

**if**(Val\_PIR()){

**if**(Val\_LDR()<150){

       LED\_Off();

       Alarm();

    }

**else**{

       LED\_PWM(Val\_LDR());

       Mute();

    }

  }**else**{

    LED\_Off();

    Mute();

  }

}

**void** Init\_LDR(){

}

**void** Init\_Buzzer(){

  pinMode(LED,OUTPUT);

  pinMode(Buzzer,OUTPUT);

}

**void** Alarm(){

**int** cnt=0;

**for**(**int** Hz=200;Hz<=800;Hz++){

    cnt++;

**if**(cnt == 40){

        cnt=0;

        digitalWrite(LED,!digitalRead(LED));    //LED状态翻转

    }

    tone(Buzzer,Hz);

    delay(5);

    }

  delay(1000);

**for**(**int** Hz=800;Hz>=200;Hz--){

    cnt++;

**if**(cnt == 40){

        cnt=0;

        digitalWrite(LED,!digitalRead(LED));    //LED状态翻转

    }

    tone(Buzzer,Hz);    //向蜂鸣器输出频率

//     play\_Joy();

//     delay(5);

    delay(5);

  }

}

**void** Mute(){

  noTone(Buzzer);

  digitalWrite(Buzzer,LOW);

  digitalWrite(LED,LOW);

}

**int** Val\_LDR(){

**int** value,out;

  value = analogRead(A0);

  out = map(value,0,1023,0,255);

**return** out;

}

**void** Init\_Light(){

    pinMode(11,OUTPUT);

}

**void** LED\_On(){   //开灯

    digitalWrite(11,HIGH);

}

**void** LED\_Off(){   //关灯

    digitalWrite(11,LOW);

}

**void** LED\_PWM(byte degree){

    analogWrite(11,degree);

}

**void** Init\_PIR(){

 pinMode(PIR\_pin, INPUT);

}

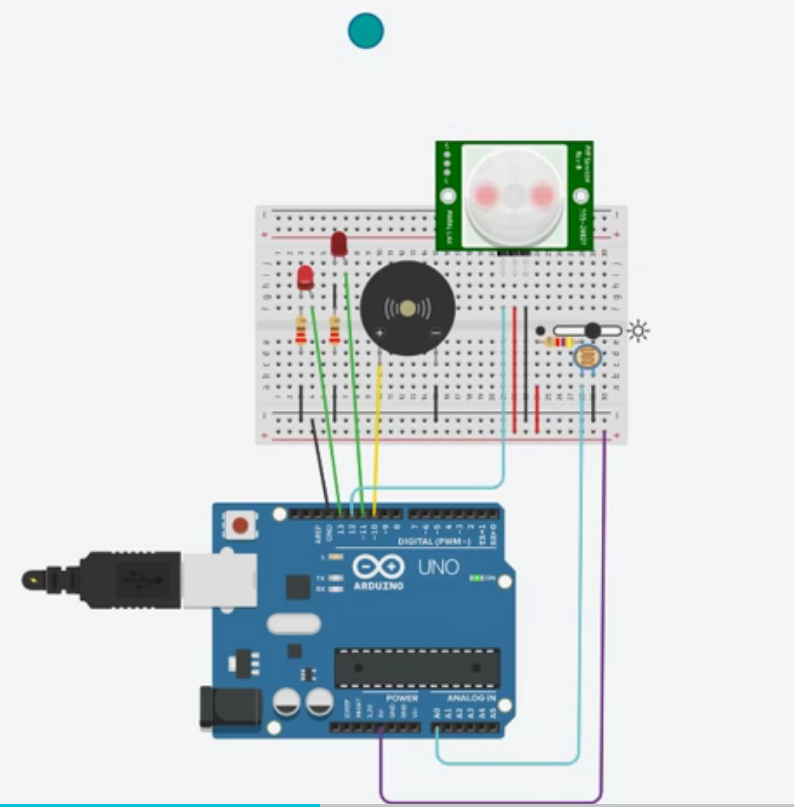
**bool** Val\_PIR(){

**bool** val;

  val = digitalRead(PIR\_pin);

**return**(val);

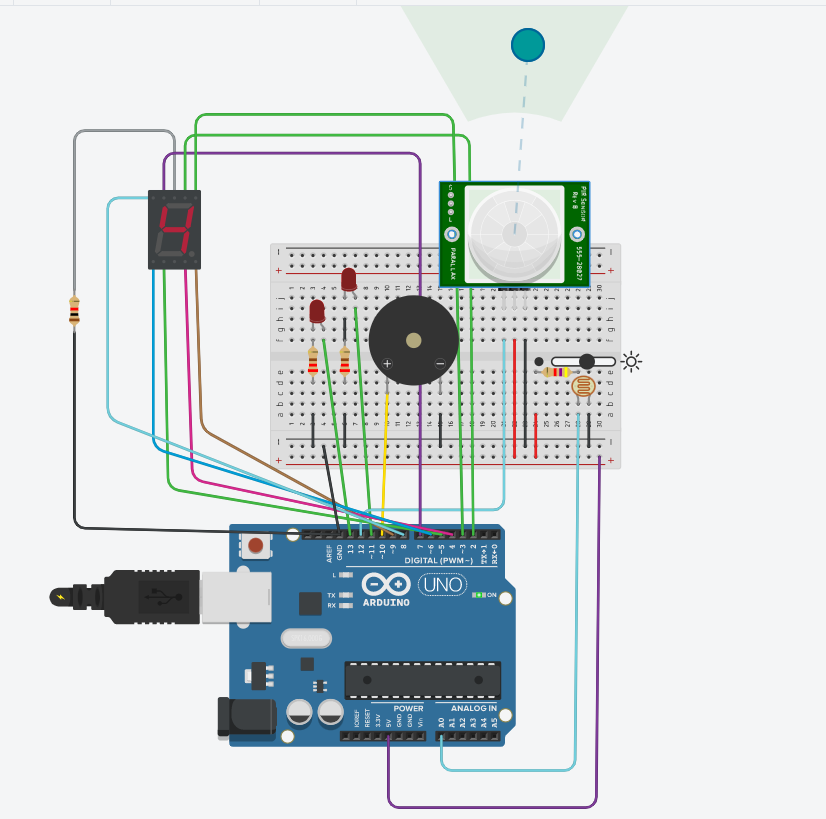
}



运行结果正常。

## 5.2.集成调试

* 多功能机器人



运行结果正常。

# 6.测试与交付

## 6.1.测试方案

* 环境、工具

Tinkpad模拟仿真平台

* 方法

模拟仿真

## 6.2.测试结果

和预期的结果一致。

# 7.总结与展望

## 7.1.当前系统的优缺点

优点：

能实时监测人类活动，并监测环境光的强度。

缺点：

数码管模块没有与具有感光功能的机器人模块完全隔离。程序并没有达到低耦合的效果。

## 7.2.未来展望

学习单片机，要先把单片机这个概念搞懂。什么是单片机？单片机是将一台能工作的计算机包含的如下部分：CPU、RAM、ROM、I/O，集成在一块集成电路芯片上。

对于Arduino Uno而言，其只有一个核心，这意味着在同一时刻，它只能做一件事。

这使得我的单片机课程设计的有了缺憾。我的蓝图是数码管模块单独分离运行，不受具有人类感知能力的可调光机器人模块的影响，但由于Arduino Uno的缘故，无法实现彻底的解耦。

上了大学，我一直在反思学习到底是为了什么？大一、大二那时候我总是被老师牵着鼻子走，这边学一点，那边学一点，到头来学的都是皮毛。

随着学习的深入，我渐渐开始觉醒，我学习不是为了应付考试，不是为了创造自身的价值，我学习仅仅是为了我自己开心！学自己感兴趣的知识，会使得自己更有动力，并愿意深究技术细节。

学校开设的课程，对我而言，仅仅是起一个引路作用，让我明白计算机专业有哪些路子可循。经过一个学期的探索，对于单片机这门课程，我已经有了自己的答案，这不是我未来考虑的方向。

感谢明老师一个学期的陪伴，江湖再见！！！！！

# 参考文献

[1] Can Arduino help with tightrope walking?[EB/OL]//Arduino Blog. (2022-06-16)[2022-06-17]. https://blog.arduino.cc/2022/06/17/can-arduino-help-with-tightrope-walking/.

[2] An UNO Mini Limited Edition necklace is a must-have accessory for Arduino lovers![EB/OL]//Arduino Blog. (2022-06-11)[2022-06-17]. https://blog.arduino.cc/2022/06/11/an-uno-mini-limited-edition-necklace-is-a-must-have-accessory-for-makers/.

[3] This system continuously monitors water levels for seven houses[EB/OL]//Arduino Blog. (2022-06-08)[2022-06-17]. https://blog.arduino.cc/2022/06/08/this-system-continuously-monitors-water-levels-for-seven-houses/.

[4] 陈霖凯. 是谁动了我的电脑——基于Arduino的智能监控抓拍器设计[J]. 发明与创新(高中生), 2022(05): 39-41.

[5] 贺焕利, 王宇峰, 章丽. 基于Arduino的车辆防酒驾系统[J]. 湖北汽车工业学院学报, 2022, 36(01): 63-65.

[6] 李柯. 基于Arduino的智能蓝牙小车设计[J/OL]. 电子测试, 2022, 36(07): 21-23. DOI:10.16520/j.cnki.1000-8519.2022.07.015.