创客教育期末作品报告

06/20/2020

顾子安

1. **项目名称：**基于人体红外热释电的家用照明控制装置
2. **灵感来源：**

我从小和我的爷爷奶奶一起生活。爷爷因为年纪大了有些老年痴呆，总是特别健忘。晚上每次他从厨房出来，几乎次次都会忘记把灯关掉，然后被奶奶教育训斥。为了解决这一问题，同时节约用电，我将利用Arduino Nano（采用Atmel Atmega328PAU单片机）作为控制器，结合人体红外热释电传感器、光敏电阻、继电器等，制作一个简易的智能家用照明控制装置。

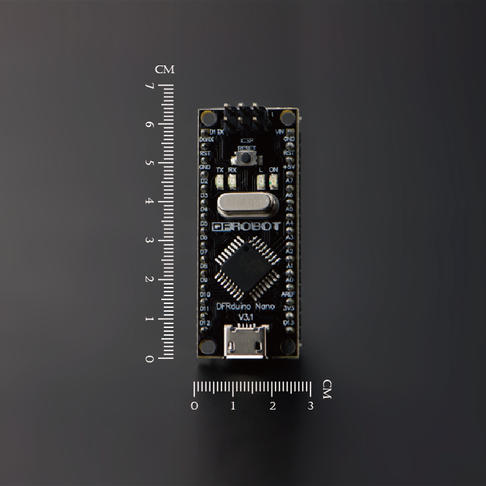
1. **项目功能：**
2. 在初始环境光线不充足的情况下（通过光敏电阻设置阈值，下同），整个装置将始终处于激活状态，绿色指示灯亮起。即当人为开启室内照明系统后，在10min以内房间内有人活动或有热源不断发出时，继电器才会导通，若10min以内一直没有检测到活动热源，则继电器将自动断开，同时红色led亮起作为指示；直到有人再次进入，继电器恢复导通状态，红色led熄灭。
3. 在环境光线自然充足（并非由开灯引起）的情况下，整个装置将处于关闭状态，绿色指示灯熄灭继电器长期处于闭合状态。
4. 所有功能需要在不改变照明电路原有线路的情况下实现，并尽可能简化改装。
5. **技术实现**
6. **硬件列表**

**表1：硬件列表**

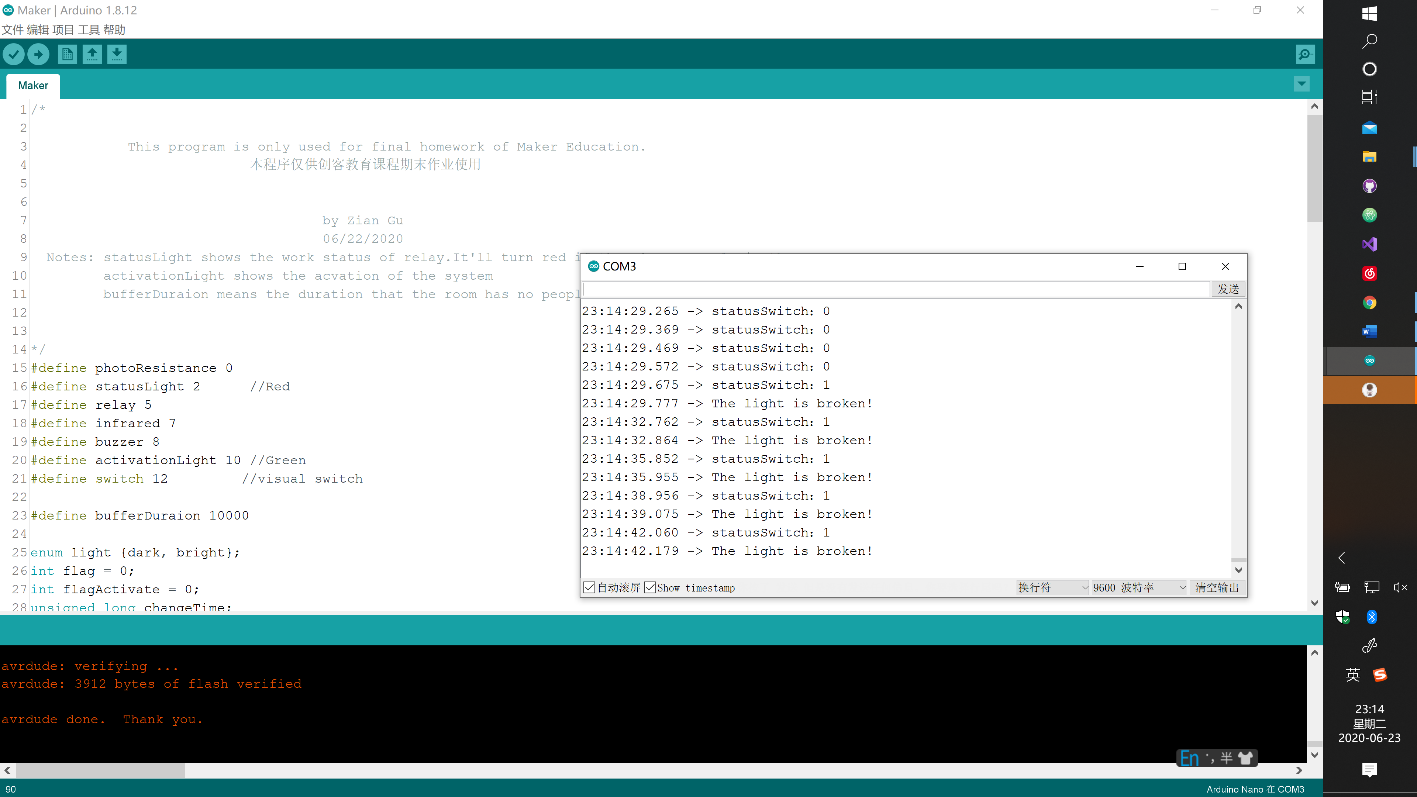
|  |  |  |
| --- | --- | --- |
| **物品** | **数量** | **成本** |
| Arduino Nano（基于Atmel Atmega328PAU单片机） | 1 | 12元/块 |
| HC-SR501数字人体红外热释电运动传感器 | 1 | 3元/块 |
| 光敏电阻 | 1 | 0.1元/只 |
| 固态继电器（DC5V控制AC220V） | 1 | 回收利用 |
| 杜邦线 | 若干 | 回收利用 |
| 电压检测模块 | 1 | 空缺 |
| 锂电池 | 1 | 回收利用 |
| 锂电池充电模块 | 1 | 空缺 |

1. **实现过程**

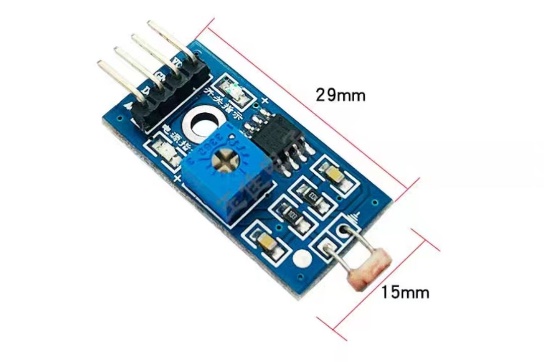
为了缩小装置的体积，本项目选用Arduino Nano（基于Atmel Atmega328PAU单片机）作为控制器**(图1)**。在Arduino IDE环境**(图2)**下使用C语言进行编程调试。使用光敏电阻**（图3）**读取环境光亮度，以此作为系统的激活依据；使用HC-SR501数字人体红外热释电运动传感器**（图4）**来判断房间内是否有人活动或者有热源释放；使用固态继电器（为减小体积，其实可以选用HF3FA电磁弹簧继电器替代）来控制光源；使用电压检测模块来检测。



**图1：Arduino Nano v3.0**



**图2：Arduino IDE平台**

****

**图3：光敏电阻 图4：人体红外热释电传感器**

1. **激活逻辑**

|  |  |  |
| --- | --- | --- |
| **环境亮度（envBrightness）** | **照明开关状态（statusSwitch）** | **Action**  **动作** |
| Bright | ON | Activated |
| Bright | OFF | Do nothing |
| Dark | ON | The light is broken（The buzzer starts working to give warnning） |
| Dark | OFF | Do nothing |

1. **代码实现**

/\*

This program is only used for the final homework of Maker Education.

本程序仅供创客教育课程期末作业使用

by Zian Gu

06/22/2020

Notes: statusLight shows the work status of relay.It'll turn red if there's no people in the room

activationLight shows the acvation of the system

bufferDuraion means the duration that the room has no people or heat source

\*/

#define photoResistance 0

#define statusLight 2 //Red

#define relay 5

#define infrared 7

#define buzzer 8

#define activationLight 10 //Green

#define switch 12 //visual switch

#define bufferDuraion 10000

enum light {dark, bright};

int flag = 0;

int flagActivate = 0;

unsigned long changeTime;

void activate();//activate the system

int envBrightness();//return 0 if dark, return 1 if bright

void setup()

{

Serial.begin(9600); // open serial port, set the baud rate to 9600 bps

pinMode(relay, OUTPUT);

pinMode(buzzer, OUTPUT);

pinMode(statusLight, OUTPUT);

pinMode(activationLight, OUTPUT);

pinMode(switch, INPUT);

pinMode(infrared, INPUT);

digitalWrite(relay, HIGH);

digitalWrite(statusLight, LOW);

digitalWrite(activationLight, LOW);

}

void loop()

{

int statusSwitch = digitalRead(switch);

Serial.print("statusSwitch：");

Serial.println(statusSwitch);

delay(100);

if (envBrightness() == bright && statusSwitch == LOW || envBrightness() == dark && statusSwitch == LOW)

{

digitalWrite(relay, HIGH);

digitalWrite(statusLight, LOW);

digitalWrite(activationLight, LOW);

}

else if (envBrightness() == bright && statusSwitch == HIGH)

{

delay(1000);

activate();

digitalWrite(activationLight, HIGH);

}

else

{

digitalWrite(statusLight, LOW);

digitalWrite(activationLight, LOW);

Serial.println("The light is broken!");

tone(buzzer,1000);

delay(1000);

noTone(buzzer);

delay(2000);

}

}

//activate the system

void activate()

{

int val\_infrared = digitalRead(infrared);

Serial.print("val\_infrared:");

Serial.println(val\_infrared);

delay(100);

if ((val\_infrared == LOW) && ((millis() - changeTime) >= bufferDuraion) && (!flagActivate))

{

tone(buzzer, 2000);

delay(1000);

noTone(buzzer);

digitalWrite(relay, LOW);

digitalWrite(statusLight, HIGH);

// changeTime = millis();

flagActivate = 1;

}

else if (val\_infrared == HIGH)

{

digitalWrite(relay, HIGH);

digitalWrite(statusLight, LOW);

flagActivate = 0;

changeTime = millis();

delay(1000);

}

flag = 1;

}

//return 0 if dark, return 1 if bright

int envBrightness()

{

int status;

int val\_photoResistance = analogRead(photoResistance);

if (val\_photoResistance > 1000)

{

status = dark;

return status;

}

else

{

status = bright;

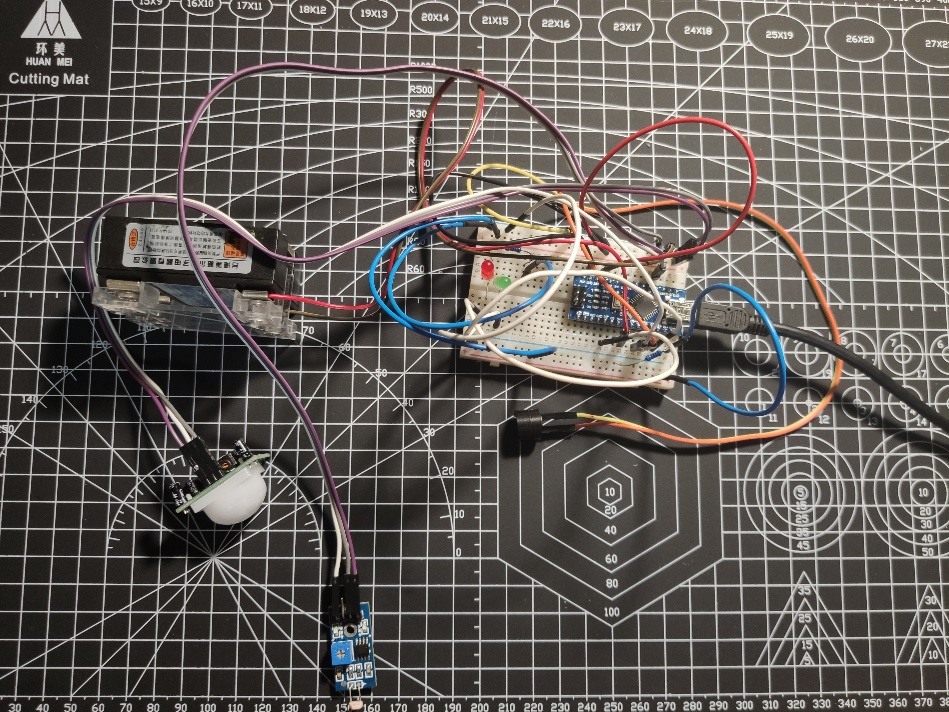
return status;

}

}

1. **项目展示**

为演示项目成品的基本功能，在此仅使用开发版给出简单展示（demo）。时间等参数的设置也以便于展示为首要考虑因素。之后作为成品可重新设计PCB板并进行印刷，并增加电压检测模块。项目demo的图片如**图5**所示，视频演示请见附件。



**图5：项目成品（开发版）**

1. **项目总结**

本项目的开发版本经过实验，可以正常工作，鲁棒性好，可以满足需求，达到展示效果。若要将本项目实用化，需要重新设计PCB板并进行印刷，同时增加电压检测模块识别墙壁开关是否打开，增加锂电池及锂电池充电模块保障装置工作用电。开发版本为演示项目效果仅简单使用数字开关模拟墙开功能，并直接用充电宝供电。照明用电端的导通与否直接由继电器自带的的工作状态指示灯表示。