

# Arduino 课程设计

---

智能多功能交通灯

## 答辩报告

报告人

---

%%%%%%%%

**01** PART ONE  
项目思路方向

**02** PART TWO  
分部实现与集成  
组装实现功能

**03** PART THREE  
问题与优化

**04** PART FOUR  
进一步探索

**CONTENT**



01

# PART ONE

## 项目思路

# 项目思路

着重于行人道交通灯，不注重于道路交通灯



满足题目要求

01



增加语音提示

02



信号灯结束时  
铃声警告

03



声波雷达  
警告闯红灯

04

## 确定项目目标

• • •



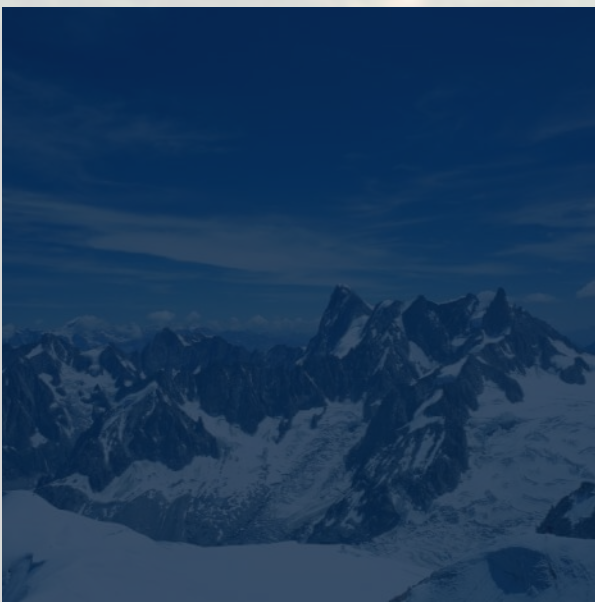
### 满足基本信号灯的功能

- 1) 绿灯常亮6秒，绿灯闪烁3秒，黄灯常亮3秒，红灯常亮5秒，红灯闪烁3秒，回到绿灯常亮状态；
- 2) 要求用数码管显示时间的倒计时；
- 3) 如果有按键按下，表示有紧急车辆通过（救火车等），亮红灯，蜂鸣器响，如果另外一个按键按下则恢复正常。

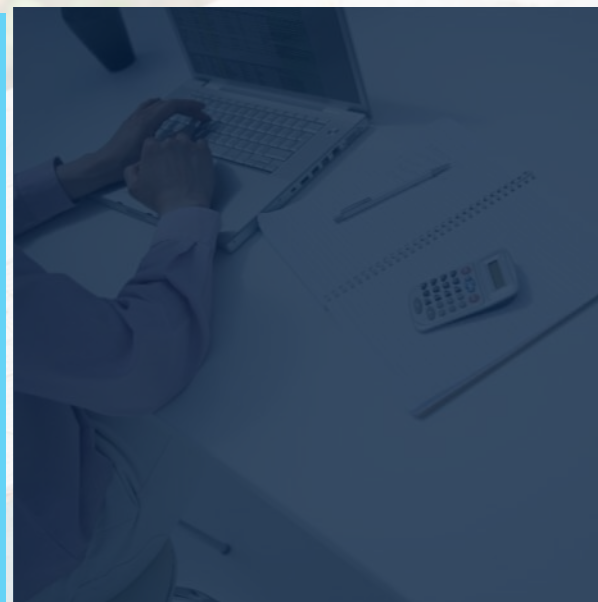


## 体现人文关怀

...



使盲人人  
可以不再  
受红绿灯  
困扰



**使交通灯能通过声音使盲人也可以使用**

注重提醒行人交通安全，  
减少交通事故发生

在绿灯快要结  
束时发出提示  
声音提醒行人  
加快步伐快速  
通过

ANNUAL REPORT  
BY SHENXIAN\_PPT

注重于交通安全  
YOUR LOGO

#### CONTENT

- ① SINGLE SPARK CAN START A PRAIRIE FIRE.
- ② A FRIEND IN NEED IS A FRIEND INDEED.
- ③ EACH MAN IS THE ARCHITECT OF HIS OWN FATE.

通过超声波判断是否  
闯红灯，并通过警示  
灯发出警告

ADD YOUR TITLE HERE  
YOUR LOGO

#### CONTENT

- ① SINGLE SPARK CAN START A PRAIRIE FIRE.
- ② A FRIEND IN NEED IS A FRIEND INDEED.
- ③ EACH MAN IS THE ARCHITECT OF HIS OWN FATE.

ANNUAL REPORT  
BY SHENXIAN\_PPT



02

# PART TWO

分部实现与集成组装



# 题目要求目的的实现

...

## 1: 红路灯亮灭控制

常亮灯: 与后面提示音混合使用for循环

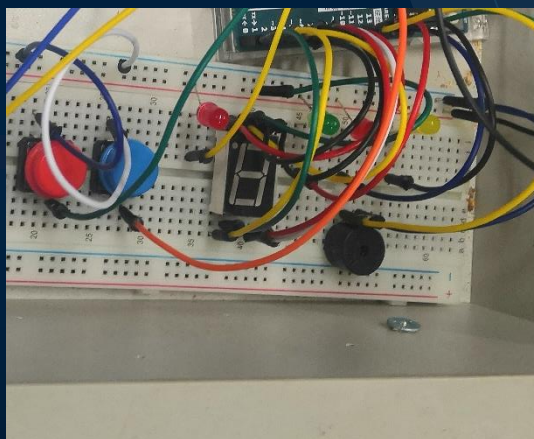
闪烁灯: 通过计算使for循环中的循环变量与delay延迟时间的乘积等于闪烁时间控制

## 2: 显示管倒计时的控制

通过和提示音和常亮进行for循环嵌套

## 3: 按键控制

在loop循环开头放置if判断语句通过digitalRead函数读取按键信号并进行判断和做出响应



# 信号提示语句

...

由于语音信号不易录入，使用音乐截取代替，在红灯部分使用倒序播放以示区别

使盲人在无法看到信号灯使一样可以通过路口

使用for循环嵌套语音输出倒计时和红路灯常亮

```
for (int x = 0; x < 60; x++) //循环60次，统计60次音符播放时间为6秒
{ digitalWrite(g, 0);使红灯常亮
  int a = 5 - (x / 10);
  deal(dofly_DuanMa[a]);通过函数实现倒计时数值与for循环和时间的关系
  tone(tonePin, tune[x]); //依次播放tune数组元素，即每个音符
  delay(100 * duration[x]); //每个音符持续的时间，即节拍duration，
  noTone(tonePin);//停止当前音符，进入下一音符
```

路灯将结束时提示铃声

...

使用蜂鸣代替提示铃声

在闪烁循环中插入蜂鸣函数使其延迟时间和灯光延迟时间之和与for循环变量的乘积等于闪烁时长

```
for (int i = 0; i < 6; i++)  
{ digitalWrite(g, 1);  
  delay(125);  
  digitalWrite(tonePin, 1);  
  delay(125);  
  digitalWrite(g, 0);  
  delay(125);  
  digitalWrite(tonePin, 0);  
  delay(125);  
}
```

**使行人在信号灯快要结束时  
有意识地加快步伐预防交通事故**

## 声波测距提示不要闯红灯

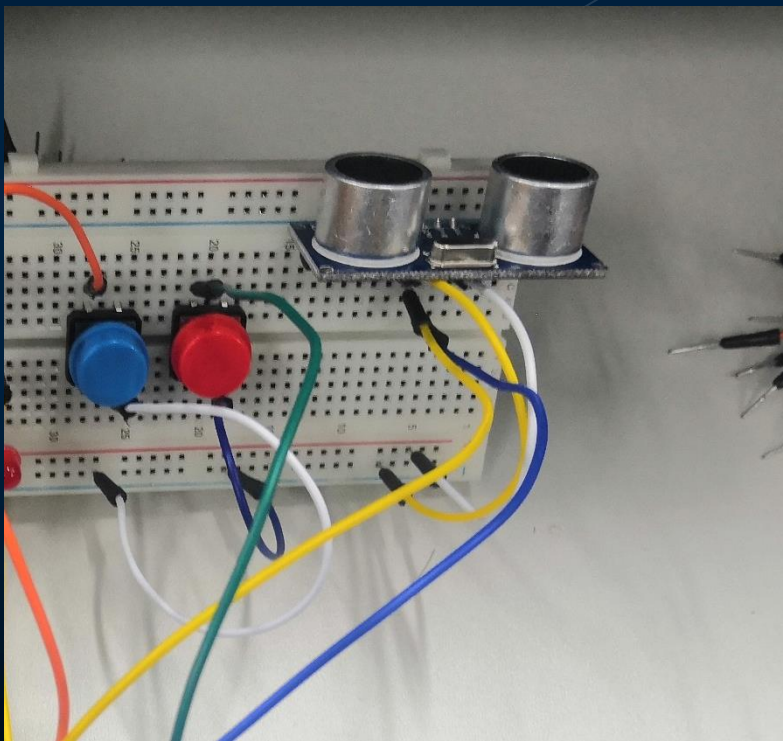
...

初始声波发出器接受器并通过函数换算成厘米单位

```
digitalWrite(trigPin, LOW);  
delay(2);  
digitalWrite(trigPin, HIGH);  
delay(10);  
digitalWrite(trigPin, LOW);  
duration1 = pulseIn(echoPin, HIGH);  
distanceCm = duration1 * 0.034 / 2;
```

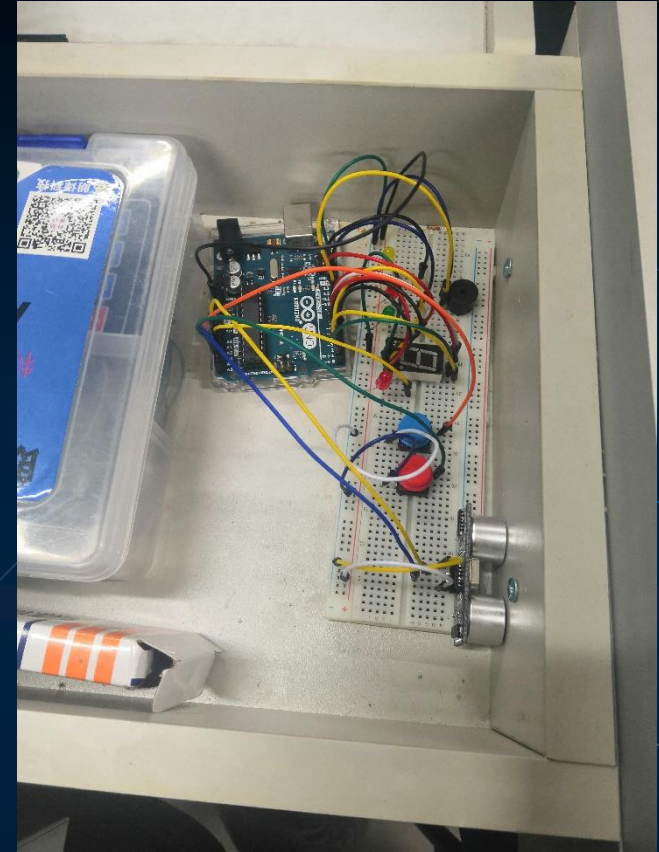
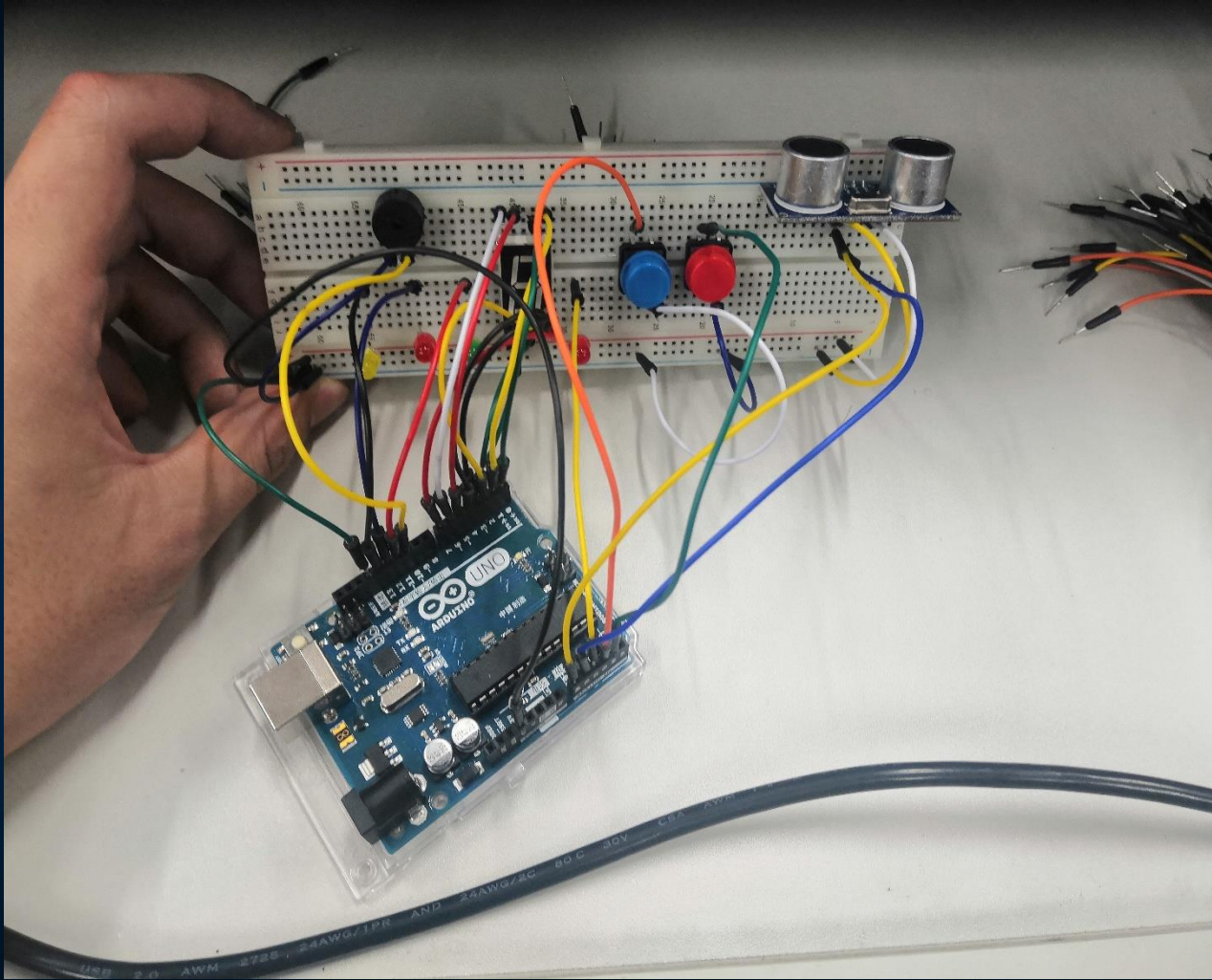
```
if (distanceCm <= 50)当测距小于50厘米时实现该程序  
{  
    digitalWrite(LED, LOW);  
}
```

在交通路口信号灯处放置超声波，在红灯时当声波器有物体接近路口就发出警示





# 硬件连接展示





03

# PART THREE

问 题 与 优 化

# 问题与优化

...

## PROBLEM1:

歌曲音调的延迟导致对时间控制不够精确

```
//曲子的节拍，即音符持续时间
float duration[] =
{
    1, 1, 1 + 1,
    0.5, 1, 1 + 0.5, 1,
    1, 1, 0.5, 0.5, 1,
    0.5, 1, 1 + 0.5, 1,
    0.5, 0.5, 0.5, 0.5, 1 + 1,
    0.5, 1, 1 + 0.5, 1,
    1 + 1, 0.5, 0.5, 1,
    1 + 1 + 1 + 1,
    0.5, 0.5, 0.5 + 0.25, 0.25, 0.5 + 0.25, 0.25, 0.5 + 0.25, 0.25,
    0.5, 1, 0.5, 1, 1,
    0.5, 0.5, 0.5 + 0.25, 0.25, 0.5 + 0.25, 0.25, 0.5 + 0.25, 0.25,
};
```

用户对于声音节拍和韵律的要求比时间要求低，所以采用更改播放频率和声音节拍的方法大幅度地降低声音延迟对时间的影响

```
//曲子的节拍，即音符持续时间
float duration[] =
{
    1, 1, 1 ,
    1, 1, 1 , 1,
    1, 1, 1, 1, 1,
    1, 1, 1 , 1,
    1, 1, 1, 1, 1 ,
    1, 1, 1 , 1,
    1 , 1, 1, 1,
    1 ,
    1, 1, 1, 1, 1, 1, 1, 1,
    1, 1, 1, 1, 1,
    1, 1, 1, 1, 1, 1, 1,
};
```

将歌曲播放频率从400更改为100

更改后时间误差降低到非常小

## PROBLEM2:按键终止程序

按键终结程序无法在任意时刻进行终结

在程序开始时加入一个按键判断

在loop循环中与if嵌套

在if里判断执行

```
if (digitalRead (buttonApin) != LOW)
{
    超声波部分
    灯光部分
    提示音部分
    .....
}
else
{
    终止部分【红灯亮蜂鸣器响其他灯灭】
}
if(digitalRead (buttonBpin) == LOW)
{
    超声波部分
    灯光部分
    提示音部分
    .....
```



优化:

```
for(int m=0:digitalRead(buttonApin) == LOW;m++)
{digitalWrite(g, 1);
 digitalWrite(r, 0);
 digitalWrite(y, 1);
 if(digitalRead(buttonBpin) == LOW)
break;
}
```

```
void loop() {
  for(int m=0:digitalRead(buttonApin) == LOW;m++)
  {digitalWrite(g, 1);
   digitalWrite(r, 0);
   digitalWrite(y, 1);
   if(digitalRead(buttonBpin) == LOW)break;
  }
  //-----超声波部分-----
  digitalWrite(trigPin, LOW);
  delay(2);
  digitalWrite(trigPin, HIGH);
  delay(10);
  digitalWrite(trigPin, LOW);
  duration1 = pulseIn(echoPin, HIGH);
  distanceCm= duration1*0.034/2;
  //-----在绿灯亮起的6秒钟内，提醒盲人声音同时亮起【此处使用音乐代替】，显示管进行倒计时
  for (int x = 0; x < 60; x++) //循环音符的次数
  { digitalWrite(g, 0);
    int a = 5- (x / 10);
    deal(dofly_DuanMa[a]);
    tone(tonePin, tune[x]); //依次播放tune数组元素，即每个音符
    delay(100 * duration[x]); //每个音符持续的时间，即节拍duration，400是调整时间的越大，曲子速度越慢，越小曲子速度越快
  }
}
```



04

# PART FOUR

进 一 步 扩 展 探 索

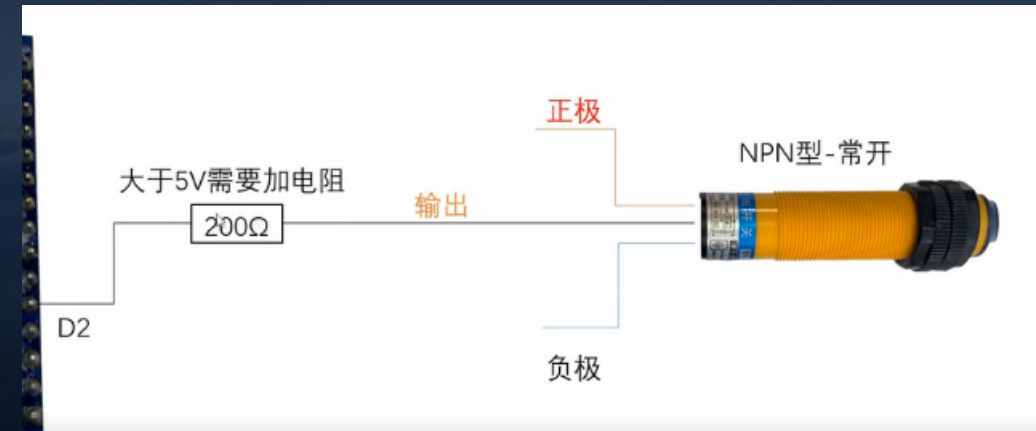
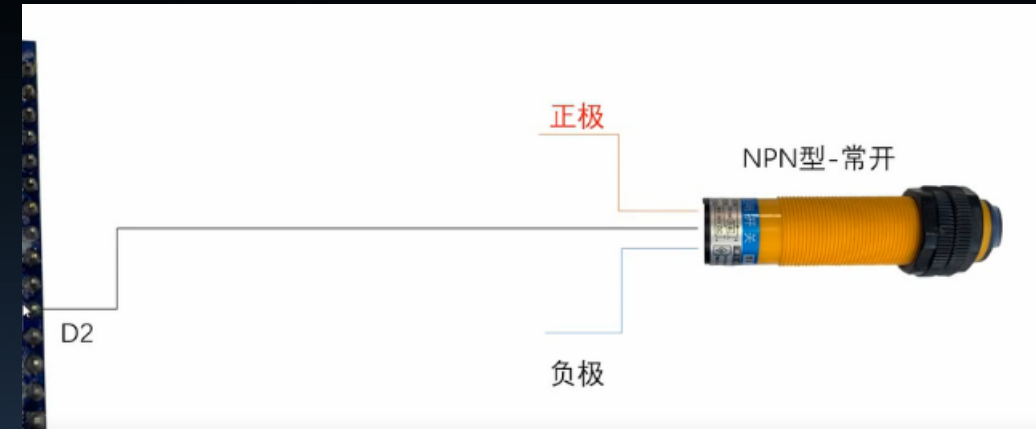


给红外线  
接一个  
12v的信号

将声波测距传感器换成光电  
传感器，使测试更精确

```
Int sin=2;
Int led=13;
```

```
Void setup(){
pinMode(sin,INPUT);//使sin接口为接入接口，接受来自红外线的信号
digitalwrite(Sin, 1); //初始化为高电平
pinMode(led,OUTPUT);// 使led接口为输出接口，输出信号
digitalWrite(led,0);//初始化为关
void loop();//主循环
If(digitalread(sin)==low)//判断输入接口的信号
{digitalWrite(led,HIGH);
}
```



串联分压

# 扩展...想法



## 或许交通灯还可以这样做

Idea1: 在交通灯上装上温度湿度感受器，采集温度湿度信息

Idea2: 交通拥挤时通过对挤压车流的数量和时段判断调整红绿灯时间

Idea3: 【人工智能：通过统计在最经常过的时段，调增交通管制；】对车流量进行统计，将数据提供给市政部门进行城市交通改造





The end

Thanks for listening

The end