

---

# 用 arduino 制作音乐

朱津纬 物理学系

本课题综合利用 arduino 的蜂鸣器、电位器、舵机和串口等功能模块制作音乐。

## 一、模块分述

先依次介绍蜂鸣器、电位器、舵机和串口各模块或功能的实现。

### 1、直接用蜂鸣器制作音乐

可直接用 `tone`(音调,长度)函数让蜂鸣器发出频率不同的声音。将音调转化成频率,就可以对照曲谱写代码了。附录 A 展示了我照葫芦画瓢,套用了 [https://blog.csdn.net/m0\\_37738838/article/details/95062958](https://blog.csdn.net/m0_37738838/article/details/95062958) 模板,将《恍然那时》一曲用代码写出。

### 2、用电位器控制蜂鸣器产生不同音调

还可以用电位器间接控制蜂鸣器发声的频率,大体思路是让程序根据电位器的不同阻值产生不同频率的方波,从而发出不同的音调。代码实现见 <http://Ilovmcu.taobao.com> 的 arduino 学习手册。

### 3、舵机

可用不同频率脉冲使舵机转过不同角度。代码实现见 <http://Ilovmcu.taobao.com> 的 arduino 学习手册。

### 4、串口

可以在程序的串口端输入数据控制程序的运行。

### 5、小结

综合下来,我的想法是制作这样一个程序:在串口端输入数值,根据该数值使舵机和与之相粘连的电位器一起转过相应角度,从而使蜂鸣器改变音调。

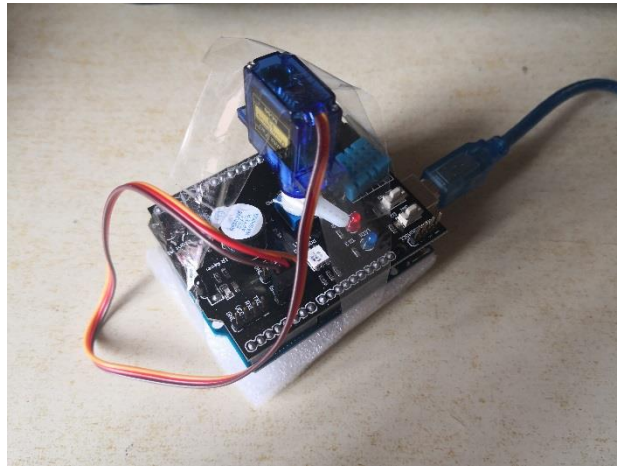
## 二、结果及分析

下图展示了装置图,程序代码在附录 B 里(基本上是整合了 <http://Ilovmcu.taobao.com> 商家提供的舵机、蜂鸣器等例程)。

该程序实现了从串口输入舵机的角度(0-180°),改变蜂鸣器音调的功能。未来需要做的工作是对音调调音(可使用手机上的调音器),建立舵机角度和音调

---

的对应关系。但由于蜂鸣器声音太刺耳，对耳朵杀伤力大，精确调音的工作没有进行下去。



装置连接图

## 附录

### A 《恍然那时》代码

```
#define NTC0 -1
#define NTC1 262
#define NTC2 294
#define NTC3 330
#define NTC4 350
#define NTC5 393
#define NTCs5 416
#define NTC6 441
#define NTC7 495

#define NTCL1 131
#define NTCL2 147
#define NTCL3 165
#define NTCL4 175
#define NTCL5 196
#define NTCL6 221
#define NTCL7 248
```

---

```

#define NTCH1 525
#define NTCH2 589
#define NTCH3 661
#define NTCH4 700
#define NTCH5 786
#define NTCH6 882
#define NTCH7 990
//列出全部 C 调的频率
int tune[]=                //根据简谱列出各频率
{
    NTC0,NTCL6,NTCL6,NTC1,
    NTC2,NTC3,NTC4,NTC3,NTC2,
    NTC3,NTC0,NTC1,NTC3,NTC5,
    NTC5,NTC1,NTC6,NTC5,NTC4,
    NTC3,NTC0,NTCL6,NTCL6,NTC1,
    NTC2,NTC3,NTC4,NTC3,NTCL7,
    NTC2,NTC1,NTC1,NTC0,NTC1,NTC3,NTC5,
    NTC5,NTC1,NTC6,NTC5,NTC3,NTC2,NTC1,
    NTC1,NTC0,NTC1,NTC5,NTC7,
    NTCH1,NTC7,NTC7,NTC5,NTC3,NTC3,NTC5,
    NTC6,NTC5,NTC5,NTC1,NTC5,
    NTC3,NTCs5,NTCH2,NTC7,NTCH1,NTCH1,NTCH2,
    NTCH3,NTCH2,NTC7,NTC0,
    NTC6,NTC5,NTC5,NTC1,NTC5,NTC0,NTC1,
    NTC4,NTC5,NTC6,NTC1,NTC1,NTCL7,
    NTC1,
};
float durt[]=              //根据简谱列出各节拍
{
    2.5,0.5,0.5,0.5,
    1,1,0.75,0.75,0.5,
    2,0.5,0.5,0.5,0.5,
    1,1,0.75,0.75,0.5,
    2,0.5,0.5,0.5,0.5,
    1,1,0.75,0.75,0.5,
    0.5,0.5,1,0.5,0.5,0.5,0.5,
    0.5,0.5,0.5,1,0.5,0.75,0.25,
    2,0.5,0.5,0.5,0.5,
    0.5,0.5,0.5,0.5,1,0.5,0.5,
    0.5,0.5,0.5,0.5,2,
    0.5,0.5,0.5,0.5,1,0.5,0.5,
    2,1,2,1,
    0.5,0.5,0.5,0.5,1,0.5,0.5,
    0.5,0.5,0.5,0.5,1,1,2,

```

---

```

};
int length;
int tonepin=5; //得用 5 号接口
void setup()
{
  pinMode(tonepin,OUTPUT);
  length=sizeof(tune)/sizeof(tune[0]); //计算长度
}
void loop()
{
  for(int x=0;x<length;x++)
  {
    tone(tonepin,tune[x]);
    delay(1000*0.83*durt[x]); //这里用来根据节拍调节延时，500 这个指数可以自己调整。
    noTone(tonepin);
  }
  delay(2000);
}

```

## B 串口控制音调代码

```

#include <Arduino.h>

#define PWM_pin 7
#define Pot A0
#define Buzzer 5

int PotBuffer = 0;
int pulsewidth = 0; //高电平时间
int degree=0;

void setup ()
{
  pinMode(PWM_pin,OUTPUT);
  Serial.begin(9600); //设置通讯的波特率为 9600
  while(Serial.read()>= 0){} //clear serialbuffer
  pinMode(Buzzer,OUTPUT); //蜂鸣器引脚设定为输出
}

void loop()
{
  if(Serial.available()>0){
    delay(100);
    degree=Serial.parseInt(); //read delaying time typed in Serial Montor

```

---

```
    }
    while(Serial.read() >= 0){
        Serial.print(degree);
        pulse(degree);           //设置舵机指向 90 度
        PotBuffer = 4*analogRead(Pot);    //读取 AD 值
        Serial.print(PotBuffer);
        for(int i = 0 ; i < 2000 ; i++)    //循环 100 次
        {
            digitalWrite(Buzzer,HIGH);    //设置输出高电平
            delayMicroseconds(PotBuffer); //延时 PotBuffer 值 us
            digitalWrite(Buzzer,LOW);     //设置输出低电平
            delayMicroseconds(100);       //延时 100us
        }
        Serial.print("ring over");

        delay(200);
    }

void pulse(int angle)           //设置舵机角度为 angle
{
    for(int i = 0 ; i < 100 ; i++)
    {
        pulsewidth=int ((angle*11)+500); //计算高电平时间
        digitalWrite(PWM_pin,HIGH);     //设置高电平
        delayMicroseconds(pulsewidth);   //延时 pulsewidth (us)
        digitalWrite(PWM_pin,LOW);      //设置低电平
        delay(20-pulsewidth/1000);      //延时 20-pulsewidth/1000 (ms)
    }
}
```