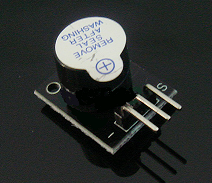
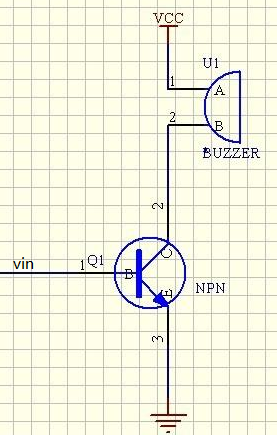
----------------------------------------------

蜂鸣器模块



Arduino buzzer module  
Active speaker  
Compatible with PC, printer, car audio system DIY  
  
**Specifications:**  
  
Voltage: 5V  
Color: Black + silver gray  
Package dimension: 77x42x13mm  
Weight: 5g



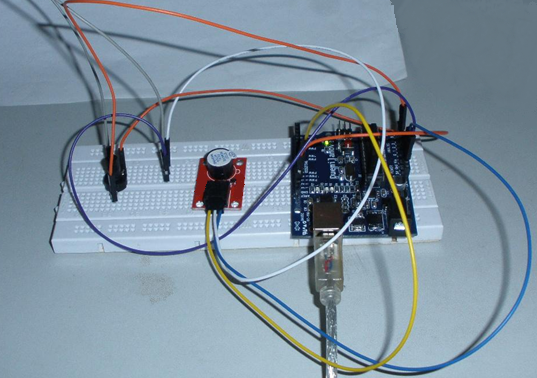
Test Requires

Arduino Controller × 1

USB Cable × 1

Buzzer Module × 1

10K potentiometer（10K） × 1

Connection: 

Below test is use Analog to control buzzer frequency.

Pin10 is to control buzzer.

Pin3 is Analog control, and we use 10K potentiometer.

Function: Adjust potentiometer, we can hear buzzer frequency change.

int speakerPin = 8;//控制喇叭的引脚

int potPin = 4;//控制可调电阻器的引脚

int value = 0;

void setup() {

pinMode(speakerPin, OUTPUT);

}

void loop() {

value = analogRead(potPin);读电阻器引脚的值

digitalWrite(speakerPin, HIGH);

delay(value);调节喇叭响的时间；

digitalWrite(speakerPin, LOW);

delay(value);调节喇叭不响的时间；

}

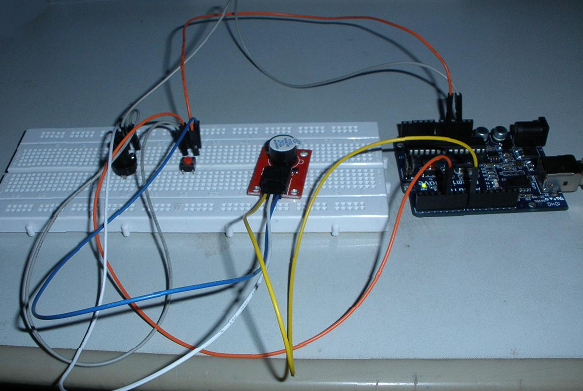
We can use potentiometer to change delay time, then we can change buzzer frequency.

下面我们加了一个按键开关控制蜂鸣器，这样我们就能模拟一个简单的门铃，当按

键被按下时喇叭就可以发出响声了。

Another Test:

We add button to control buzzer, then we can imitate a simple doorbell. When push button, buzzer can ring.

Connect way:

Code:

const int buttonPin = 4; // 按键引脚;

const int speakerPin = 8; //蜂鸣器引脚;

// variables will change:

int buttonState = 0; // 读取按键引脚的一个值

void setup()

{

//设置按键引脚为输入模式，蜂鸣器引脚为输出模式；

pinMode(speakerPin, OUTPUT);

pinMode(buttonPin, INPUT);

}

void loop(){

// 读取按键一个初值，这里在电路中我接了是在默认高电平，所以初值为高；

buttonState = digitalRead(buttonPin);

/\* 如果按键为高，则蜂鸣器不响；

因为我刚开始接在硬件电路中初值为高，所以if条件成立，蜂鸣器不响

\*/

if (buttonState == HIGH) {

digitalWrite(speakerPin,LOW);

}

else {

//这里按键的值为低电平（也是按键被按下时）；蜂鸣器响起

digitalWrite(speakerPin,HIGH);

}

}

Below is use PWM to control buzzer:

Code:

int speakerPin = 8;

byte song\_table[] = { 30, 30, 30, 40, 50, 60, 70, 80, 90, 100,110, 120, 130, 140,

150, 160, 170, 180, 190, 200, 210, 220, 230, 240, 250, 250, 240, 230, 220, 210, 200, 190, 180,

170, 160, 150, 140, 130, 120, 110, 100, 90, 80, 70, 60, 50, 40, 30, 30, 30 };

int MAX = 50;

int count = 0;

void setup() {

pinMode(speakerPin, OUTPUT);

}

void loop() {

analogWrite(speakerPin,song\_table[count]);

count ++;

if (count > MAX) {

count = 0;

}

delay(50);

}