

به نام خدا  
دانشگاه صنعتی امیرکبیر (پلی تکنیک تهران)  
دانشکده مهندسی کامپیوتر



## آزمایشگاه ریزپردازنده

### گزارش کار پروژه نهایی

### گیتار برقی سه سیم

استاد درس: مهندس معصوم زاده

مهدی رحمانی / ۹۷۳۱۷۰۱  
محمد امین رضائی / ۹۷۳۱۰۲۴

## گیتار الکتریکی ۳ سیم

### هدف پروژه



بعد از آنکه پیانوی الکتریکی را یاد گرفتید، اکنون نوبت این رسیده که یک گیتار الکتریکی پیاده سازی کنید! از طریق کیبورد ورودی بگیرید. دست راست با سه دکمه M K O نواختن سیم را تعیین می کند و دست چپ با دکمه های چپ تر کیبورد، سیم را میگیرد. چالش سطحی، پیاده سازی نت های گیتار روی صفحه کلید است. چالش اصلی، نحوه اجرای چند نت همزمان است. قبلاً گفتیم این ممکن نیست، چون فقط یک تایمر استفاده می شود، ولی راه هایی وجود دارد.

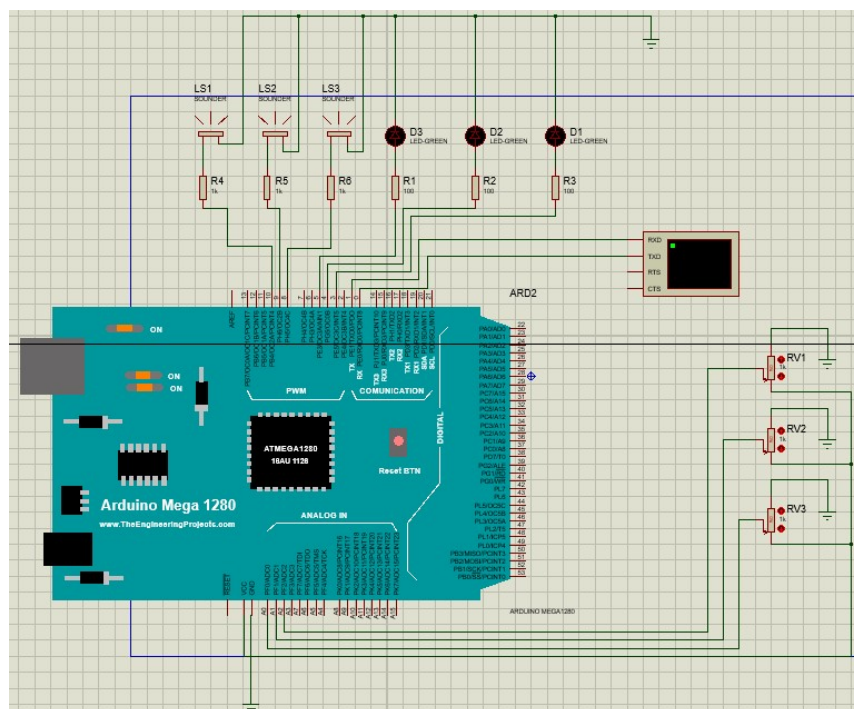
امتیاز: یکسری پتانسیومتر که نقش کوک کردن سیم ها را دارد. هرچه بیشتر چرخانده شود، نت های سیم بم تر و زیر تر می شود.  
**قطعات مورد نیاز:**

- برد Arduino Mega2560
- پتانسیومتر
- پیزوالکتریک اسپیکر
- ال ای دی به مقدار کافی

در این پروژه ما با کمک یک برد آردوینو و ۳ بلندگو و ۳ پتانسیومتر و ۳ ال ای دی یک گیتار الکتریکی میسازیم.

برای اجرای چند نت همزمان از ۳ اسپیکر استفاده میکنیم و چون برد آردوینو چند تایمر دارد این امکان وجود دارد که همزمان چند نت را پخش کنیم.

ابتدا شماتیک مدار را میبینیم:



مشاهده میشود که ما برای دادن فرامین از یک ترمینال مجازی استفاده میکنیم. مشاهده میشود که پتانسیومتر ها به پایه های آنالوگ متصل میباشند و ال ای دی ها و اسپیکر ها به پایه های دیجیتال متصل میباشند.

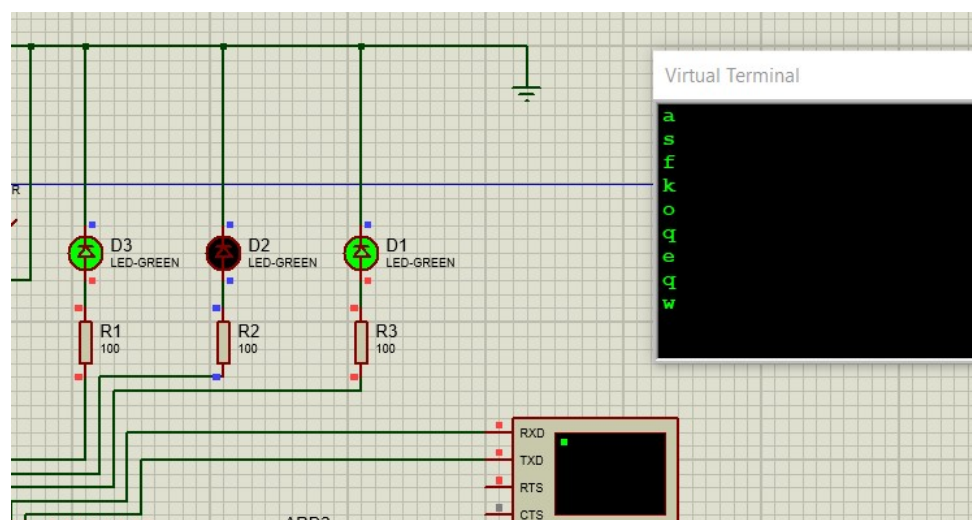
در این مدار ما با پتانسیومتر ها هر یک از سه سیم گیتار را کوک میکنیم. هر ال ای دی نیز نشان دهنده ی فعال بودن یک سیم میباشد.

برای در دست گرفتن سیم های اصلی از سه کلید: M K O استفاده میکنیم و از ۶ کلید از کلید های سمت چپ این ۳ کلید برای نواختن نوت ها استفاده میکنیم.

کلید های کنترلی در شکل زیر مشخص شده اند:



تصویر شبیه سازی و مدار در حال کار:



حال به بررسی کد میپردازیم:

```
#include <Tone.h>

#define FREQ_DIV 4

Tone tone0;
Tone tone1;
Tone tone2;

const int leds[] = {3, 4, 5};
bool isPlaying[3] = {0};
int notes[3][6] = {
    {NOTE_G3, NOTE_GS3, NOTE_A4, NOTE_AS4, NOTE_B4, NOTE_C4},
    {NOTE_D3, NOTE_DS3, NOTE_E3, NOTE_F3, NOTE_FS3, NOTE_G3},
    {NOTE_G2, NOTE_GS2, NOTE_A3, NOTE_AS3, NOTE_B3, NOTE_C3}
};
int frets[3] = {0};
float scale;

void setup() {
    Serial.begin(9600);

    pinMode(leds[2], OUTPUT);
    pinMode(leds[1], OUTPUT);
    pinMode(leds[0], OUTPUT);

    tone0.begin(8);
    tone1.begin(9);
    tone2.begin(10);
}
```

با کمک کتابخانه `tone` پروژه پیاده سازی شده است. ابتدا یک متغیر تعریق میکنیم و آنرا برابر ۴ میگذاریم تا بتوانیم در نهایت در پخش صوت با آن بهتر صدا را پخش کنیم. سپس ۳ `tone` را برای هر بلندگو میسازیم. سپس پین های ال ای دی ها را مشخص میکنیم و یک آرایه سه تایی که معرف هر سیم است از جنس بولین میسازیم. در ادامه نوت ها را اضافه میکنیم. سپس برای هر یک از نوت های خاص یک سیم یک آرایه سه تایی میسازیم. در ادامه و در بخش `setup` پین های خروجی را مشخص میکنیم.

```

void loop() {
  if (Serial.available() > 0) {
    char key = Serial.read();
    Serial.println(key);

    switch (key) {
      case 'z':
        frets[2] = 0;
        break;
      case 'x':
        frets[2] = 1;
        break;
      case 'c':
        frets[2] = 2;
        break;
      case 'v':
        frets[2] = 3;
        break;
      case 'b':
        frets[2] = 4;
        break;
      case 'n':
        frets[2] = 5;
        break;
      case 'a':
        frets[1] = 0;
        break;
      case 's':
        frets[1] = 1;
        break;
      case 'd':
        frets[1] = 2;
        break;
      case 'f':

```

در بخش حلقه کلید فشرده شده را با کمک ترمینال مجازی از کاربر گرفته و چک میکنیم تا ببینیم چیست و واکنش مناسب را نشان دهیم. اگر کلید های فرعی بود(همان ۱۸ کلید برای نوت های هر ۳ سیم) مقدار مناسب را از یک تا شش به آن اختصاص میدهیم.

```

case 'm':
    if (isPlaying[2]) {
        isPlaying[2] = false;
        digitalWrite(leds[2], LOW);
        tone2.stop();
    } else {
        isPlaying[2] = true;
        digitalWrite(leds[2], HIGH);
    }
    break;
case 'k':
    if (isPlaying[1]) {
        isPlaying[1] = false;
        digitalWrite(leds[1], LOW);
        tone1.stop();
    } else {
        isPlaying[1] = true;
        digitalWrite(leds[1], HIGH);
    }
    break;
case 'o':
    if (isPlaying[0]) {
        isPlaying[0] = false;
        digitalWrite(leds[0], LOW);
        tone0.stop();
    } else {
        isPlaying[0] = true;
        digitalWrite(leds[0], HIGH);
    }
    break;
};

```

در این بخش اگر ورودی جزو سه کلید اصلی برای فعال شدن سیم ها بود؛ چک میکنیم اگر سیم فعال بود؛ پس باید ال ای دی خاموش و tone متوقف شود و سیم غیر فعال شود و همینطور اگر سیم غیر فعال بود؛ باید ال ای دی مربوطه روشن شود و سیم فعال شود.

```

    if (isPlaying[2]) {
        scale = analogRead(A2) / 512.0;
        tone2.play((int) (notes[2][frets[2]] * FREQ_DIV * scale));
    }
    if (isPlaying[1]) {
        scale = analogRead(A1) / 512.0;
        tone1.play((int) (notes[1][frets[1]] * FREQ_DIV * scale));
    }
    if (isPlaying[0]) {
        scale = analogRead(A0) / 512.0;
        tone0.play((int) (notes[0][frets[0]] * FREQ_DIV * scale));
    }
}
}

```

در آخر هم با توجه به فعال بودن یا نبودن سیم؛ ابتدا کوک بودن و مقیاس آن مشخص میشود. میدانیم که آنالوگ ۱۰۲۴ مقدار به ما میدهد؛ پس مقدار میانی آن برای پتانسیومتر ۵۱۲ میباشد که ۵۰ درصد میباشد.

برای پخش صوت هم با کمک دستور `play` نوت مورد نظر را در مقیاس و مقدار ثابت ۴ که همان `FREQ_DIV` میباشد ضرب میکنیم تا صدای متناسب با کوک پخش شود.

یاعلی

پایان