

# Bluetooth transmitter with LED matrix for speakers

## Introducere

Lucian Flutur 1222A

Acest proiect are ca scop realizarea unui transmițător bluetooth cu o matrice LED controlată de un microfon.

## Descriere generală

Placa Arduino este folosită pentru conectarea matricei LED și a microfonului pentru a crea un circuit de analiză de spectru. Acestea sunt conectate sub o tensiune de 5V pe breadboard. Tot prin breadboard este conectat și transmițătorul Bluetooth 5.0 și un fan pentru răcirea circuitelor. Tensiunea de alimentare de 5V este luată de pe placa Arduino iar placa Arduino este alimentată prin mufa USB 2.0. semnalul audio din transmițător este luat mai departe pe mufa RJ45 și se conectează la orice boxă cu mufa RCA.

## Hardware Design

1 x Arduino MEGA 2560 1 x Elected microhone 1 x LED matrix 8×8 1 x Breadboard 1 x Bluetooth 5.0 Transmitter 1 x RJ45 1 x RCA 1 x Fan 5V 0.23A

## Software Design

```
#include "LedControl.h" LedControl lc = LedControl(12, 10, 11, 1);
```

```
const int maxScale = 11; const int sensorPin = A4; const int sampleWindow = 50; 50ms = 20Hz
unsigned int sample; unsigned long startMillis; unsigned long timeCycle; unsigned int signalMax = 0;
unsigned int signalMin = 1024; unsigned char index = 0; unsigned int peakToPeak[8]; unsigned int
displayPeak[8]; unsigned int temp[8]={0,0,0,0,0,0,0,0}; unsigned int signalMaxBuff[8]; unsigned int
signalMinBuff[8]; void setup() { Led matrix
```

```
lc.shutdown(0, false); // bật hiện thị
lc.setIntensity(0, 1); // chỉnh độ sáng
lc.clearDisplay(0); // tắt tất cả led
```

```
Serial.begin(9600);
```

```
}
```

```
void loop() {
```

```
startMillis = millis();
//peakToPeak = 0;
```

```
signalMax = 0;
signalMin = 1024;
```

```
// Get data in 50ms
```

```
while (millis() - startMillis < sampleWindow) {
    sample = analogRead(sensorPin);
```

```
    if (sample < 1024) {
        if (sample > signalMax) {
            signalMax = sample;
        }
        if (sample < signalMin) {
            signalMin = sample;
        }
    }
}
```

```
// 20Hz - 64Hz - 125Hz - 250Hz - 500Hz - 1kHz (timeCycle = 1/F)(ms)
timeCycle = millis() - startMillis;
if (timeCycle == 1 || timeCycle == 2 || timeCycle == 4 || timeCycle == 8
    || timeCycle == 16 || timeCycle == 32 || timeCycle == 40 || timeCycle
== 50) {
    signalMaxBuff[index] = signalMax;
    signalMinBuff[index] = signalMin;
    index = (index + 1) % 8;
    delay(1);
    //Serial.println(timeCycle);
}
}
```

```
// Delete pointer to array
index = 0;
```

```
// Calculation after get samples
for (int i = 0; i < 8; i++) { // i = row (led matrix)
  // sound level
  peakToPeak[i] = signalMaxBuff[i] - signalMinBuff[i];

  // Map 1v p-p level to the max scale of the display
  displayPeak[i] = map(peakToPeak[i], 0, 1023, 0, maxScale);

  // Show to led matrix
  displayLed(displayPeak[i], i);

  // Led drop down
  if (displayPeak[i] >= temp[i]) {
    temp[i] = displayPeak[i];
  }
  else {
    temp[i]--;
  }

  lc.setLed(0, i, temp[i], true);
  delayMicroseconds(250);
}

}
```

```
void displayLed(int displayPeak, int row) {
```

```
switch (displayPeak) {
  case 0 : lc.setRow(0, row, 0x80); break;
  case 1 : lc.setRow(0, row, 0xC0); break;
  case 2 : lc.setRow(0, row, 0xE0); break;
  case 3 : lc.setRow(0, row, 0xF0); break;
  case 4 : lc.setRow(0, row, 0xF8); break;
  case 5 : lc.setRow(0, row, 0xFC); break;
  case 6 : lc.setRow(0, row, 0xFE); break;
  case 7 : lc.setRow(0, row, 0xFF); break;
}

}
```

## Rezultate Obținute

Rezultatul obținut în urma realizării proiectului este un transmițător audio Bluetooth funcțional care poate fi folosit cu orice boxă cu input RCA și voltaj de referință -10dBv.

## Concluzii

Să faci un proiect Arduino când este criză de semiconductori global, nu este distractiv.

From:

<http://ocw.cs.pub.ro/courses/> - **CS Open CourseWare**

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2022/cstan/22>



Last update: **2022/06/02 09:16**