

Decibelmeter

Introducere

Scopul proiectului este realizarea unui decibelmeter, un aparat capabil sa capteze intensitatea sunetelor si sa prezinte utilizatorului informatia atat digital prin intermediul unui ecran, cat si analogic folosind un ledbar.

Descriere generală

Utilizam un modul microfon pentru captarea sunetului si o placuta Arduino Uno R3 pentru procesarea lui, in functie de intensitatea sunetului se va aprinde secvential un ansamblu de leduri si se va transmite valoarea in dB unui ecran LCD.



Hardware Design

Componente utilizate:

- Arduino Uno R3
- Breadboard
- LEDs
- Cabluri
- Rezistente
- Modul microfon
- Display LCD

Schema hardware:



Software Design

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C lcd = LiquidCrystal_I2C(0x27, 16, 2);

const int sampleWindow = 50;
unsigned int sample;

#define SENSOR_PIN A0
#define PIN_QUIET1 3
#define PIN_QUIET2 4
#define PIN_MODERATE1 5
#define PIN_MODERATE2 6
#define PIN_LOUD1 7
#define PIN_LOUD2 8

void setup ()
{
  pinMode (SENSOR_PIN, INPUT);
  pinMode(PIN_QUIET1, OUTPUT);
  pinMode(PIN_QUIET2, OUTPUT);
  pinMode(PIN_MODERATE1, OUTPUT);
  pinMode(PIN_MODERATE2, OUTPUT);
  pinMode(PIN_LOUD1, OUTPUT);
  pinMode(PIN_LOUD2, OUTPUT);

  digitalWrite(PIN_QUIET1, LOW);
  digitalWrite(PIN_QUIET2, LOW);
  digitalWrite(PIN_MODERATE1, LOW);
  digitalWrite(PIN_MODERATE2, LOW);
  digitalWrite(PIN_LOUD1, LOW);
  digitalWrite(PIN_LOUD2, LOW);

  lcd.backlight();
  delay(20);
  lcd.init();
}

void loop ()
{
  unsigned long startMillis= millis();
  float peakToPeak = 0;
```

```
unsigned int signalMax = 0;
unsigned int signalMin = 1024;

while (millis() - startMillis < sampleWindow)
{
    sample = analogRead(SENSOR_PIN);
    if (sample < 1024)
    {
        if (sample > signalMax)
        {
            signalMax = sample;
        }
        else if (sample < signalMin)
        {
            signalMin = sample;
        }
    }
}

peakToPeak = signalMax - signalMin;
int db = map(peakToPeak,0,500,28,85);
lcd.setCursor(0, 0);
lcd.print("Loudness: ");
lcd.print(db);
lcd.print("dB");
if(db <= 30){
    digitalWrite(PIN_QUIET1, LOW);
    digitalWrite(PIN_QUIET2, LOW);
    digitalWrite(PIN_MODERATE1, LOW);
    digitalWrite(PIN_MODERATE2, LOW);
    digitalWrite(PIN_LOUD1, LOW);
    digitalWrite(PIN_LOUD2, LOW);
}

else if (db > 30 && db < 35)
{
    digitalWrite(PIN_QUIET1, HIGH);
    digitalWrite(PIN_QUIET2, LOW);
    digitalWrite(PIN_MODERATE1, LOW);
    digitalWrite(PIN_MODERATE2, LOW);
    digitalWrite(PIN_LOUD1, LOW);
    digitalWrite(PIN_LOUD2, LOW);
}

else if (db >= 35 && db < 40)
{
    digitalWrite(PIN_QUIET1, HIGH);
    digitalWrite(PIN_QUIET2, HIGH);
    digitalWrite(PIN_MODERATE1, LOW);
    digitalWrite(PIN_MODERATE2, LOW);
}
```

```
    digitalWrite(PIN_LOUD1, LOW);
    digitalWrite(PIN_LOUD2, LOW);
}

else if (db >= 40 && db < 45)
{
    digitalWrite(PIN_QUIET1, HIGH);
    digitalWrite(PIN_QUIET2, HIGH);
    digitalWrite(PIN_MODERATE1, HIGH);
    digitalWrite(PIN_MODERATE2, LOW);
    digitalWrite(PIN_LOUD1, LOW);
    digitalWrite(PIN_LOUD2, LOW);
}

else if (db >= 45 && db < 50)
{
    digitalWrite(PIN_QUIET1, HIGH);
    digitalWrite(PIN_QUIET2, HIGH);
    digitalWrite(PIN_MODERATE1, HIGH);
    digitalWrite(PIN_MODERATE2, HIGH);
    digitalWrite(PIN_LOUD1, LOW);
    digitalWrite(PIN_LOUD2, LOW);
}

else if (db >= 50 && db < 60)
{
    digitalWrite(PIN_QUIET1, HIGH);
    digitalWrite(PIN_QUIET2, HIGH);
    digitalWrite(PIN_MODERATE1, HIGH);
    digitalWrite(PIN_MODERATE2, HIGH);
    digitalWrite(PIN_LOUD1, HIGH);
    digitalWrite(PIN_LOUD2, LOW);
}

else if (db >= 65)
{
    digitalWrite(PIN_QUIET1, HIGH);
    digitalWrite(PIN_QUIET2, HIGH);
    digitalWrite(PIN_MODERATE1, HIGH);
    digitalWrite(PIN_MODERATE2, HIGH);
    digitalWrite(PIN_LOUD1, HIGH);
    digitalWrite(PIN_LOUD2, HIGH);
}

    delay(200);
    lcd.clear();
}
```

Dezvoltat folosind Arduino IDE, am utilizat "Wire.h" si "LiquidCrystal_I2C" pentru interactiunea cu ecranul LCD.

Rezultate Obținute

Realizarea unui decibelmtru functional, usor de folosit, capabil sa lucreze in intervale de sunet uzuale (30dB-65dB).

Concluzii

Download

Jurnal

Finished initial doc. (5/6/2023)

Added hardware schematic and source code of the project. (5/26/2023)

Bibliografie/Resurse

Listă cu documente, datasheet-uri, resurse Internet folosite, eventual grupate pe **Resurse Software** și **Resurse Hardware**.

[Export to PDF](#)

From:

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

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2023/tmiu/decibelmeter>



Last update: **2023/05/29 15:50**