

Earthquake Sensor

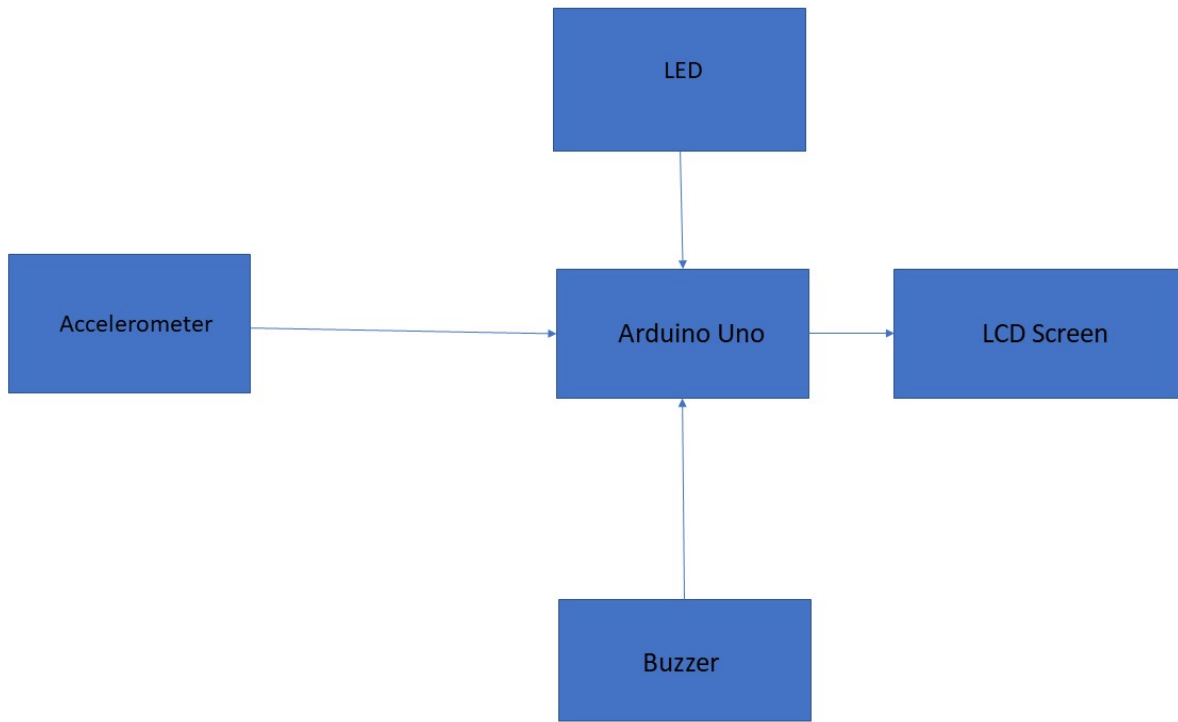
Proiect realizat de Manole Rares

Introducere

Dispozitivul este facut in asa fel incat sa detecteze miscarile seismice. Cand acestea sunt resimtite, el porneste alarma audio-vizuala din dotare. Aceasta se opreste dupa 10 secunde de la ultima unda. Ulterior, este afisata magnitudinea seismului resimtit. Acest dispozitiv este util intrucat ne permite sa intelegem mai bine seismele printr-un mod atractiv, putand sa simulam acest fenomen cand dorim.

Descriere generală

Cand suprafata unde se afla dispozitivul este zguduita, aceasta miscare este preluata de accelerometru. Acesta transmite magnitudinea catre arduino. Cat timp aceasta nu este 0, LED-ul se aprinde intermitent iar buzzer-ul va suna alarma. Dupa trecerea a 10 secunde de la ultimul soc, este afisata magnitudinea maxima a seismului resimtit.



Hardware Design

Componente:

*Arduino Uno

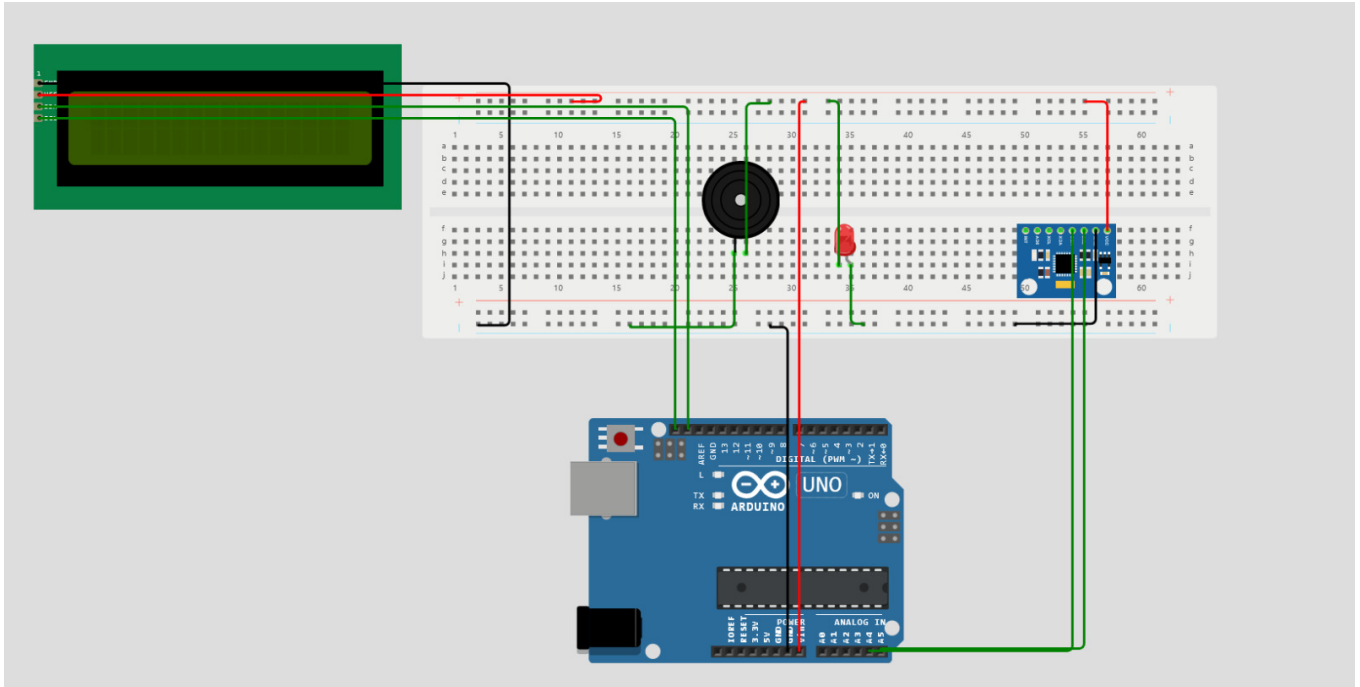
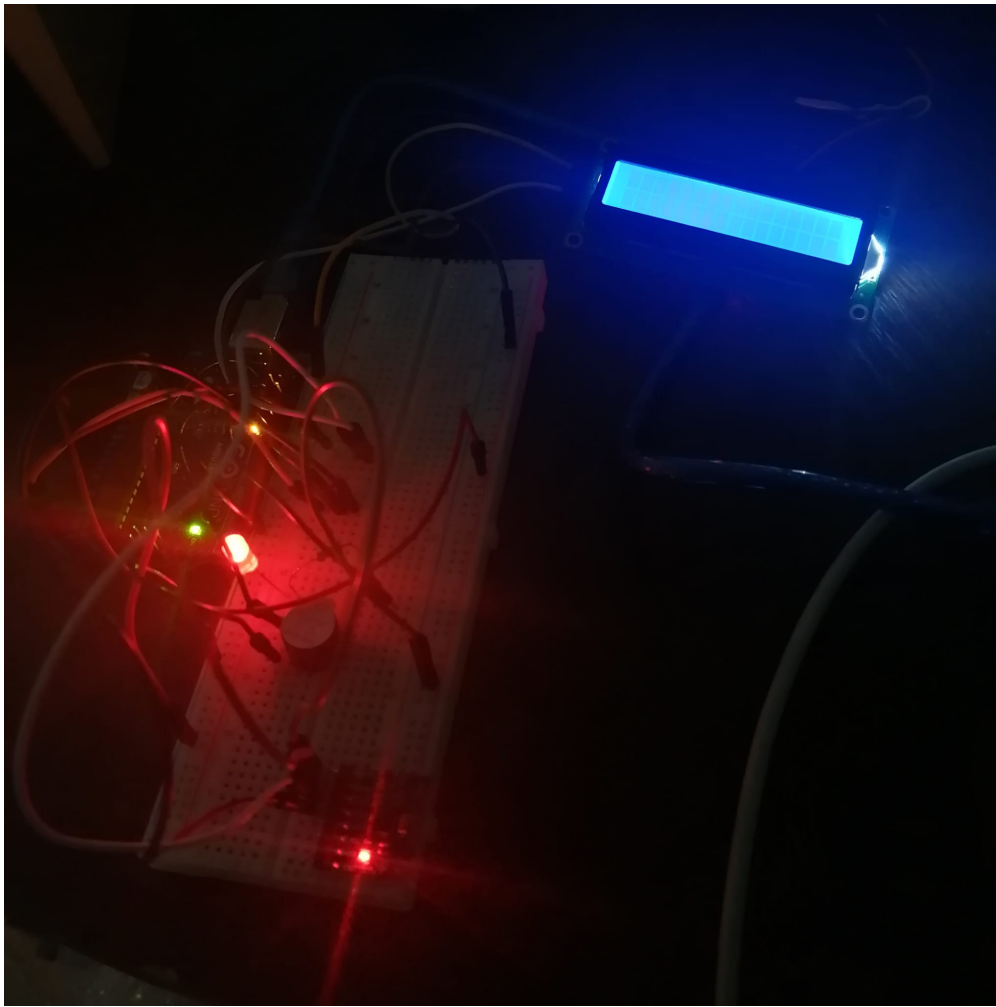
*LED

*Accelerometru

*Buzzer

*LCD Screen

*Breadboard



Software Design

Din biblioteca pentru accelerometru, am folosit un exemplu prestabilit care afisa valorile x,y,z a senzorului. Pe baza acestuia s-a bazat codul, facand modificarile aferente cerintei.

```
//librariile folosite

#include "Wire.h"
#include "MPU6050.h"
#include <LiquidCrystal_I2C.h>

LiquidCrystal_I2C lcd(0x27,20,4);

MPU6050 mpu;
int16_t ax, ay, az;
int16_t gx, gy, gz;

struct MyData {
  byte X;
  byte Y;
  byte Z;
};

//valorile de referinta folosite pentru determinarea magnitudinii

float iX=0;
float iY=0;
float iZ=0;

//pragul magnitudinii de unde incepe alarma

float noiseTreshold=2.5;

bool alarmActive=false;

const int ledPin=13;
const int buzzerPin=12;

MyData data;

float magnitude = 0.0;
float prev_magnitude = 0.0;

unsigned long lastMagnitude = 0;

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

```
mpu.initialize();

pinMode(ledPin,OUTPUT);
pinMode(buzzerPin,OUTPUT);

lcd.begin(16,2);
lcd.backlight();
lcd.setCursor(0, 0);
  lcd.print("Waiting...");
}

void loop()
{
  mpu.getMotion6(&ax, &ay, &az, &gx, &gy, &gz);

  // se calculeaza magnitudinea in baza datelor precedente
  float acc_x = ax - iX;
  float acc_y = ay - iY;
  float acc_z = az - iZ;

  float acc_total = sqrt(acc_x * acc_x + acc_y * acc_y + acc_z * acc_z);

  //avand in vedere "acuratetea" accelerometrului folosit, am calibrat
  formula

  magnitude = log10(acc_total) *4.25 -8 ;

  // updatarea datelor ce vor deveni de referinta pentru calculele viitoare
  prev_magnitude = magnitude;
  iX=ax;
  iY=ay;
  iZ=az;

  data.X = map(ax, -17000, 17000, 0, 255); // axa x
  data.Y = map(ay, -17000, 17000, 0, 255); // axa Y
  data.Z = map(az, -17000, 17000, 0, 255); // axa Z

  lcd.setCursor(0, 1);

//activarea alarmei in urma depasirii treshold-ului

if(magnitude>noiseTreshold){
  digitalWrite(ledPin, HIGH);
  tone(buzzerPin, 1000);

  lcd.print("Earthquake!      ");
  lcd.setCursor(0, 2);
  lcd.print("");
```

```
    lastMagnitude=millis();

    alarmActive=true;
}else{

    // oprirea ei
    digitalWrite(ledPin,LOW);
    noTone(buzzerPin);
    lcd.print("Nothing detected");

    if (alarmActive && (millis() - lastMagnitude >= 10000)) {
        // Reset the alarm state
        alarmActive = false;
    }

}

delay(500);

//printarea in serial monitor valorile axelor X-Y-Z si magnitudinea
resimtita

Serial.print("Axis X = ");
Serial.print(data.X);
Serial.print(" ");
Serial.print("Axis Y = ");
Serial.print(data.Y);
Serial.print(" ");
Serial.print("Axis Z = ");
Serial.print(data.Z);
Serial.print(" ");
Serial.print("Magnitude = ");
Serial.println(magnitude, 2);
}
```


Rezultate Obținute

Concluzii

M-am familiarizat mai mult cu partea de hardware si sa trec peste anumite dificultati din acest punct

de vedere. De asemenea, a fost folositor sa imbin programarea cu partea de matematica/fizica, fiind necesar sa ma interesez cum se calculeaza magnitudinea unui seism in scara Richter (de asemenea sa imi dau seama de amplitudinea unui cutremur: de exemplu, un cutremur cu magnitudinea 7 e de 10 ori mai puternic decat unul cu magnitudinea 6 sau de 100 de ori decat unul cu magnitudinea 5, ca intensitate). Din pacate nu am putut sa fac proiectul cum mi-am dorit initial, intrucat accelerometrul are destul de mult "noise" si nu are o acuratete asa buna. Totusi, sunt multumit de rezultatul final.

Download

O arhivă (sau mai multe dacă este cazul) cu fișierele obținute în urma realizării proiectului: surse, scheme, etc. Un fișier README, un ChangeLog, un script de compilare și copiere automată pe uC crează întotdeauna o impresie bună .

Fișierele se încarcă pe wiki folosind facilitatea **Add Images or other files**. Namespace-ul în care se încarcă fișierele este de tipul **:pm:prj20??:c?** sau **:pm:prj20??:c?:nume_student** (dacă este cazul). **Exemplu:** Dumitru Alin, 331CC → **:pm:prj2009:cc:dumitru_alin**.

Jurnal


Puteți avea și o secțiune de jurnal în care să poată urmări asistentul de proiect progresul proiectului.

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/avaduva/earthquake_sensor 

Last update: **2023/05/29 19:11**