# Earthquake Alarm

Cabancalan, Ivan Justine E.
Espejo, Ma. Lourvic C.
Lacaba, Oliver G.
Lalangan, Wyder S.
Lantano, Alyssa Mae S.
Lopez, Gerry Jr. G.

*Department of Electronics Engineering*

*College of Engineering and Architecture*

*Urdaneta City University*

May 26, 2019

# Contents

## 1 Introduction

Earthquake are sudden and cannot be predicted when, where and how it will happen. This phenomenon is due to sudden release of energy in our Earth's crust that creates waves. This releases of energy causes the seismic waves that does ground the shake and make vibrations. Due to this doing, results to energy travelling all the way through the earth passing through either the length of the earth's surface or through the earth's interior.

Earthquakes are usually triggered when rock underground suddenly shifts along a fault. The fault plane is the subterranean surface along which the rocks move and breaks. Along this is the seismograph that is use for the magnitudes or size of the amplitude of the seismic wave that tells motion of the earth.

Earthquake causes destruction to human lives. It usually causes drastic changes, including ground movements of the surface that causes difference in the ground flow. Other than shaking of ground, earthquake may also produce sudden occur under ocean that can cause more phenomenon.

In today's generation, despite all the scientific experiments in the field of predicting shocks and earthquake, the idea of right alarm and monitoring of earthquakes using electronic monitoring tools for detection or the so-called sensors do still remain vision of the future. Although many techniques are implemented and the path of scientific warning based on study of electronic instrumentation, explanation and a consistent time between a possible warning and its eventuality parts an idea for any scientists.

Scientists around the world have used the scenario of earthquake alarm diagnosis and design based on any detection of vibration and wave before the following waves can intensify and cause to large number of destruction and damage. Some of earthquake detection device are already known to detect earthquake waves.

An earthquake unavoidable and unpredictable that often causes harm to lives and property. We cannot fight it but we can stay alert and aware using technology.The project initiative is to create a monitoring system which can detect if there are chances of an Earthquake coming or in the near future. This project is about warning you while the earthquake is happening. Along side using Arduino with Atmega644p micro-controller a buzzer that we will create and Earthquake alarm system that will tell the people about the Earthquake.

## 2 High Level Design

The Gizduino is a microcontroller board based on the ATmega328, ATmega644 and ATmega168. It has 14 digital input/output pins, 6 analog inputs, a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header,and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-toDC adapter or battery to get started. It is an open source computing platform based on a simple input/output (I/O) board and the use of standard programming language; in otherwords, it is a tool for implementing a program you have designed. Gizduino is programmed using the IDE (Integrated Development Environment).

Gizduino is ideal for beginner programmers and hobbyists because of its simplicity compared to other platforms. It is a multiplatform environment; it can run on Windows, Macintosh, and Linux. It is programmable via USB cable, which makes it more accessible and allows communication with the computer.

GizDuino+ series are Arduino IDE compatible and Sanguino inspired boards that features more I/O - 12 additional I/Os, over the standard gizDuino* we are already accustomed with. Using a more feature rich picoPowerAVR chip ATMEGA164P, you are now afforded with an additional hardware UART port (Serial1), and one more SPI channel, all of which are configurable as general purpose digital I/O.

The Gizduino is a microcontroller board based on the ATmega328 and ATmega168. It has 14 digital input/ output pins, 6 analog inputs, a 16 MHz crystal oscilla- tor, a USB connection, a power jack, an ICSP header, and a reset button.

Figure 1: Gizduino

Each Gizduino contains a micro-controller in this project the ATmega644 is the component being used as micro-controller. The ATmega644 is a 40 pin DIP chip that is available with an Arduino bootloader and support for the Arduino code via the Sanguino libraries.

The ATmega644 is a low-power CMOS 8-bit microcontroller based on the AVR enhanced RISC architecture. By executing powerful instructions in a single clock cycle, the ATmega644 achieves throughputs approaching 1 MIPS per MHz allowing the system designer to optimize power consumption versus processing speed.

The ATmega644 provides the following features: 64 Kbytes of In-System Programmable Flash with Read-While-Write capabilities, 2 Kbytes EEPROM, 4 Kbytes SRAM, 32 general purpose I/O lines, 32 general purpose working registers, Real Time Counter (RTC), three flexible Timer/Counters with compare modes and PWM, 2 USARTs, a byte oriented 2-wire Serial Interface, a 8-channel, 10-bit ADC with optional differential input stage with programmable gain, programmable Watchdog Timer with Internal Oscillator, an SPI serial port, IEEE std. 1149.1 compliant JTAG test interface, also used for accessing the On-chip Debug system and programming and six software selectable power saving modes. The Idle mode stops the CPU while allowing the SRAM, Timer/-Counters, SPI port, and interrupt system to continue functioning. The Power-down mode saves the register contents but freezes the Oscillator, disabling all other chip functions until the next interrupt or Hardware Reset. In Power-save mode, the asynchronous timer continues to run, allowing the user to maintain a timer base while the rest of the device is sleeping. The ADC Noise Reduction mode stops the CPU and all I/O modules except Asynchronous Timer and ADC, to minimize switching noise during ADC conversions. In Standby mode, the Crystal/Resonator Oscillator is running while the rest of the device is sleeping. This allows very fast start-up combined with low power consumption. In Extended Standby mode, both the main Oscillator and the Asynchronous Timer continue to run.

The device is manufactured using Atmel's high-density nonvolatile memory technology. The Onchip ISP Flash allows the program memory to be reprogrammed in-system through an SPI serial interface, by a conventional nonvolatile memory programmer, or by an On-chip Boot program running on the AVR core. The boot program can use any interface to download the application program in the application Flash memory. Software in the Boot Flash section will continue to run while the Application Flash section is updated, providing true Read-While-Write operation. By combining an 8-bit RISC CPU with In-System Self-Programmable Flash on a monolithic chip, the Atmel ATmega644 is a powerful microcontroller that provides a highly flexible and cost effective solution to many embedded control applications.

The ATmega644 AVR is supported with a full suite of program and system development tools including: C compilers, macro assemblers, program debugger/simulators, in-circuit emulators, and evaluation kits. Below shows the figure of ATmega644 and its Pinout diagram.

Sensors are sophisticated devices that are frequently used to detect and respond to electrical or optical signals. A Sensor converts the physical parameter (for example: temperature, blood pressure, humidity, speed, and many more) into a signal which can be measured electrically.

In this project we use a specific sensor for vibration detector, the ADXL335 Accelerometer. The ADXL335 is a small, thin, low power, complete 3-axis accelerometer with signal conditioned voltage outputs. The product measures acceleration with a minimum full-scale range of 3 g. It can measure the static acceleration of gravity in tiltsensing applications, as well as dynamic acceleration resulting from motion, shock, or vibration.

The user selects the bandwidth of the accelerometer using the CX, CY, and CZ capacitors at the XOUT, YOUT, and ZOUT pins. Bandwidths can be selected to suit the application, with a


Figure 2: ATmega644

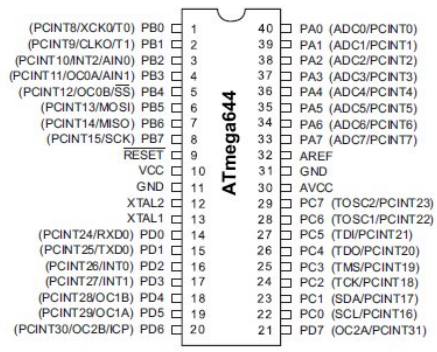


Figure 3: Pinout Diagram of ATmega644

range of 0.5 Hz to 1600 Hz for X and Y axes, and a range of 0.5 Hz to 550 Hz for the Z axis.

The ADXL335 is available in a small, low profile, 4 mm 4 mm 1.45 mm, 16-lead, plastic lead frame chip scale package (LFCSP$_LQ$).
$ADXL335workslikeacriticalsensorindetectingvibrationsandpredictearthquake.Thedevice\backslash EarthquakeDetecto$
$X-,Y-andZ-axes.TheseoutputpinsareconnectedtoADCpinsofArduinoUno.Foranyaccelerationproductionor$

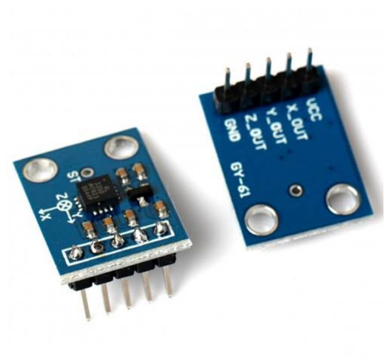


Figure 4: ADXL335 Accelerometer

Hence, this project is built for notify and warning it wouldn't be complete without a component that shows its notification that notifies the user. Here, we use the 16x2 LCD Display for visual warning and the S36G24 Speaker for its audio warning.

LCD (Liquid Crystal Display) screen is an electronic display module and find a wide range of applications. A 16x2 LCD display is very basic module and is very commonly used in various devices and circuits. These modules are preferred over seven segments and other multi segment LEDs. The reasons being: LCDs are economical; easily programmable; have no limitation of displaying special even custom characters (unlike in seven segments), animations and so on.

A 16x2 LCD means it can display 16 characters per line and there are 2 such lines. In this LCD each character is displayed in 5x7 pixel matrix. This LCD has two registers, namely, Command and Data. The command register stores the command instructions given to the LCD. A command is an instruction given to LCD to do a predefined task like initializing it, clearing its screen, setting the cursor position, controlling display etc. The data register stores the data to be displayed on the LCD. The data is the ASCII value of the character to be displayed on the LCD.

A range of weatherproof mylar cone loudspeakers with ferrite magnets. These are specifically designed for outdoor use as the mylar cones are impervious to water or moisture, and their wide frequency response makes them suitable for many applications, including alarm sounders. Available with round or square frame, size ranges 40mm(1") to 100mm (4"). Impedance 8 nominal all models. An example of this is the S36G24 Speaker.

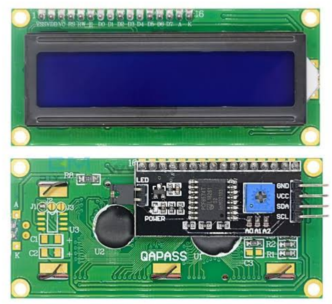


Figure 5: 16x2 LCD display

Figure 6: S36G24 Speaker

Lastly, to signify if the device is ready in use a red led is use a indicator. A light-emitting diode (LED) is a semiconductor light source that emits light when current flows through it. Electrons in the semiconductor recombine with electron holes, releasing energy in the form of photons.

Figure 7: Red LED

## 3 Hardware Design

We have designed an electronic device which will help us to show detection of the first signs of ground shaking during an earthquake. In structures, it is hard to define a certain position where the maximum impactof vibration will be felt.

A device having the components of Gizduino with ATmega 644 with a vibration or shock sensor, ADXL335 Accelerometer. To indicate if the device is working, a Red LED is placed. For visual warning and output notification of the Alarm, a 16x2 Liquid Crystal Display is connected and for its audio warning a S36G24 Speaker is connected.

In oriented structures, it is hard to define a certain position where the maximum impact of vibration will be felt. In order to generalize, we normally consider the joints of the steel based structure to carry the maximum burden of the load and can certainly develop bends which leads to fractures in the nearby corners.

When piezoelectric ceramics are used as sensors, there is usually some noise in the measure voltage. All sensors located with merged in modern radio channel which analyzes the data obtained for their size and approximately the expected power events, as well as the position of the epicenter. In the present earthquake sensor circuit, we have used a high frequency vibration detection monitoring circuit which can be used for the detection of seismic detector unit as the detecting agent as shown.

The Arduino program sketch has been shown in the code to follow as shown in figure below. When a high frequency signal is detected within the next seconds, the output is detected as high and the LED will blink and the BUZZER circuit will be active. The circuit is based on the principle of latching that is used for earthquake alarm.

As we mentioned earlier that we have used Accelerometer for detecting earthquake vibrations along any of the three axes so that whenever vibrations occur accelerometer senses that vibrations and convert them into equivalent ADC value. Then these ADC values are read by Arduino and shown over the 16x2 LCD.

First, do calibrate the Accelerometer by taking the samples of surrounding vibrations whenever Arduino Powers up. Then we need to subtract those sample values from the actual readings to get the real readings. This calibration is needed so that it will not show alerts with respect to its normal surrounding vibrations. After finding real readings, Arduino compares these values with predefined max and min values. If Arduino finds any changes values are more then or less then the predefined values of any axis in both direction (negative and positive) then Arduino trigger the speaker and shows the status of alert over the 16x2 LCD and a LED also turned on as well. We can adjust the sensitivity of Earthquake detector by changing the Predefined values in Arduino code.

'Prevention is better than cure', a statement that goes perfect with the events whose chance of occurrence is highly possible. Earthquake is one of the disaster that comes as bad fate and sweeps out human lives, it is that unpredictable phenomenon that cannot be avoided, but at least we can take steps to lessen the huge effect of its consequences. In doing so, our recent technologies play a vital role.

Objective of this project is to rapidly detect an earthquake, estimate the level of ground shaking expected, and issue a warning before significant ground shaking starts. This is by detecting the first energy to radiate from an earthquake. The motivation for doing this study is that to notify us beforehand an earthquake occurs and by which shows the possible threshold during monitoring mode and indicates if needed to evacuate; such that to lessen the damage or destruction brought by it. Taking these actions before shaking starts can reduce damage and casualties during an earthquake. It can also prevent cascading failures in the aftermath of an event.

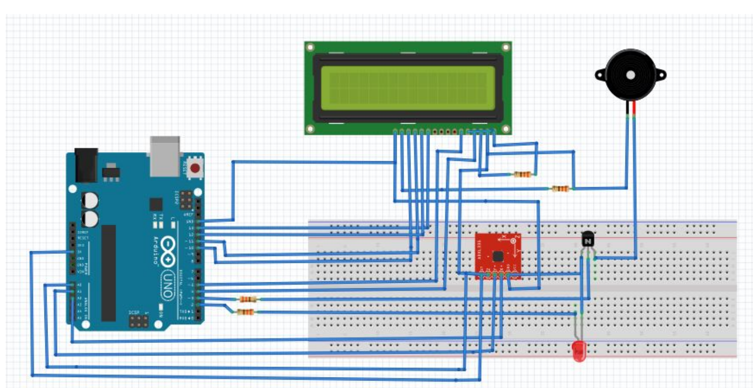

Figure 8: Earthquake Alarm Hardware Design

# 4 Software Design

The earth quake alarm circuit is based on the principle that earth quake create slow and suddenly vibration. In order to detect this vibration geo mechanical detector is used. In this section lies the software design of our device, Earthquake Alarm Detector. Below, shows the figure of our Project Design Flow Chart.

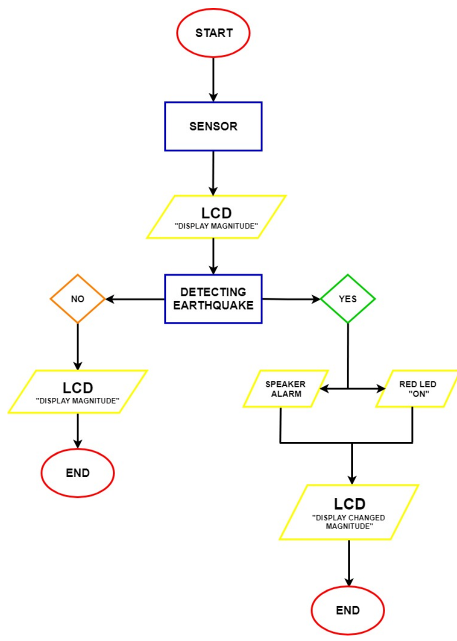


Figure 9: Project Design Flow Chart

For the codes, we use the software Arduino IDE. We program the LCD using "LiquidCrystalDisplay.h" for it to display command. Next, "Buzzer.h" is included in the code for the audio output of the device when alarming. Then, we define the LED as "LED 2" for easier detection in code. For the sensor, we define X, Y and Z as values of which detects vibration of earth and generates analog voltages in 3 axes.

In this Earthquake Detector Alarm, we have made three codes: one for Arduino to detect an earthquake and another for Processing IDE to plot the earthquake vibrations over the graph on Computer and to reset the device when the earthquake is done. Below shows the main code of our device: (For actual codes needed, visit https://github.com/walogi)

```
#include <LiquidCrystal.h>
LiquidCrystal lcd(4,5,13,12,11,10);
#include <Buzzer.h>
Buzzer buzzer(3);
#define led 2
#define x A2
#define y A1
#define z A0

int xsample=0;
int ysample=0;
int zsample=0;
long start;

#define samples 10
#define maxVal 5
#define minVal -5

void setup() {
  lcd.begin(16,2);
  Serial.begin(9600);
  delay(1000);

  lcd.print("  EaRThQuAkE  ");
  lcd.setCursor(0,1);
  lcd.print("     ALARM     ");
  delay(2000);
  lcd.clear();
  lcd.print(">>eLecTrON!X<< ");
  lcd.setCursor(0,1);
  lcd.print("_Micro_Project_");
  delay(2000);
  lcd.clear();
  lcd.print("Calibrating.....");
  lcd.setCursor(0,1);
  lcd.print("Please wait...");
  pinMode(3, OUTPUT);
  pinMode(led, OUTPUT);
  digitalWrite(3, LOW);
  digitalWrite(led, LOW);
for(int i=0;i<samples;i++)
  {
    xsample+=analogRead(x);
    ysample+=analogRead(y);
    zsample+=analogRead(z);
  }
  xsample/=samples;
  ysample/=samples;
  zsample/=samples;
```

```
  delay(3000);
  lcd.clear();
  lcd.print("Calibrated");
  delay(1000);
  lcd.clear();
  lcd.print("Device Ready");
  delay(1000);
  lcd.clear();
  lcd.print(" X      Y      Z    ");
}

void loop() {
    int value1=analogRead(x);
    int value2=analogRead(y);
    int value3=analogRead(z);
    int xValue=xsample-value1;
    int yValue=ysample-value2;
    int zValue=zsample-value3;

    lcd.setCursor(0,1);
    lcd.print(zValue);
    lcd.setCursor(6,1);
    lcd.print(yValue);
    lcd.setCursor(12,1);
    lcd.print(zValue);
    delay(100);

 if(xValue < minVal || xValue > maxVal  || yValue < minVal || yValue > maxVal  || zValue < minVal || zValue > maxVal)
    {
      lcd.setCursor(0,0);
 lcd.print("Earthquake Alert ");
lcd.setCursor(0,1);
lcd.print("                   ");
      if(3 == LOW)

      start=millis();
      buzzer.begin(0);

          buzzer.sound(NOTE_E8, 100);
         buzzer.end(10);


          digitalWrite(led,HIGH);
          delay(10);
          digitalWrite(led,LOW);
          delay(10);

    }
   else

     }
    else
 {
 lcd.clear();
 lcd.print(" X     Y       Z ");
 }

     digitalWrite(3, LOW);
     digitalWrite(led, LOW);

     Serial.print("x=");
     Serial.println(xValue);
     Serial.print("y=");
     Serial.println(yValue);
     Serial.print("z=");
     Serial.println(zValue);

 }
```

Figure 10: Project Code

# 5 Result

To show the results of our Product, we have filmed and upload it to a website which in the link is given below.

Product Video Link:

# 6 Conclusions

The design of the device is a representation of an innovative approach in construction of an earthquake warning and alert system. Reducing the capacitor further enhances the response time of the circuit. The system uses two different subsystems and detects the change in the strain pattern by the subsystems. All sensors used are readily available and cheap thus making it a user friendly and affordable product. Initially spurious readings, plus the trigger is sometimes late.

# 7 Appendix

## 7.1 Cost Details

| Item Description | Quantity | Cost |
|---|---|---|
| Gizduino+644V4 | 1 | Php 555.00 |
| 16x2 LCD Display | 1 | Php 149.75.00 |
| ADXL335 5V | 1 | Php 359.00 |
| BC548 | 3 | Php 9.00 |
| SSW-WSH-1016 | 1 | Php 10.00 |
| S36G24 | 1 | Php 22.00 |
| Female Header | 40 | Php 38.00 |
| Male Header | 40 | Php 16.00 |
| SF-Flux | 1 | Php 28.00 |
| 330 Ohms 1/4W Resistor | 4 | Php 2.50 |
| 1k Ohms Resistor | 2 | Php 2.50 |
| 10k Ohms Resistor | 2 | Php 2.50 |
| 3mm Red Led | 10 | Php 13.00 |
| 3mm Green Led | 10 | Php 13.00 |
| 5mm Green Led | 2 | Php 6.00 |
| 5mm Red Led | 2 | Php 6.00 |
| 2m Connecting Wire | 1 | Php 15.00 |
| Pre-sensitized PCB | 1 | Php 300.00 |
| Food | | Php 500.00 |
| Gas | | Php 100.00 |
| Transportation Fare | | Php 255.00 |
| Print | | Php 287.00 |
| Ferric Chloride | 1 | Php 50.00 |
| Drill Bit | 1 | Php 30.00 |

## 7.2 PCB Design

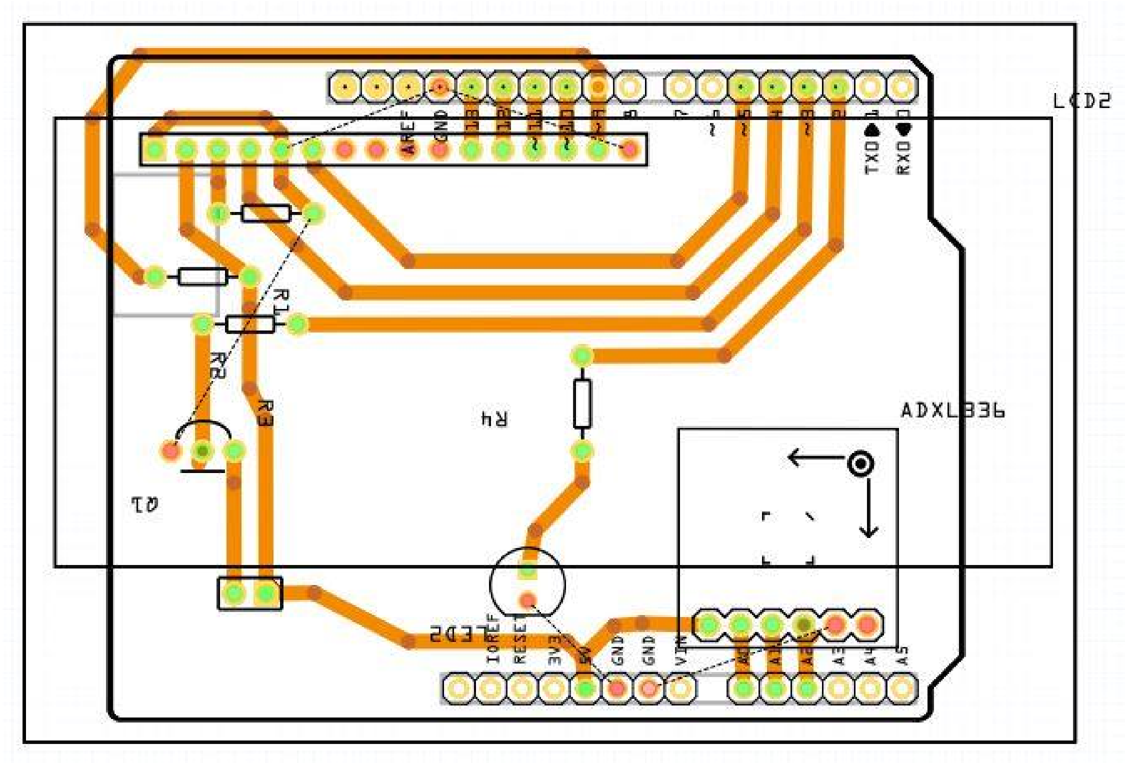


Figure 11: PCB Designl

## 7.3 Full Schematics

Circuit of this Earthquake detector Arduino Shield PCB is described here. In this project, we have used Arduino that reads accelerometer's analog voltage and convert them into the digital values. Arduino also drives the buzzer, LED, 16x2 LCD and calculate and compare values and take appropriate action. Next part is Accelerometer which detects vibration of earth and generates analog voltages in 3 axes (X, Y, and Z). LCD is used for showing X, Y and Z axis's change in values and also showing alert message over it. This LCD is attached to Arduino in 4-bit mode. RS, GND, and EN pins are directly connected to 4, 5 and 13 , GND and 8 pins of Arduino and rest of 4 data pins of LCD namely D4, D5, D6 and D7 are directly connected to digital pin 12, 11, and 10 of Arduino. The speaker is connected to pin 3 of Arduino through an 330 Ohm.

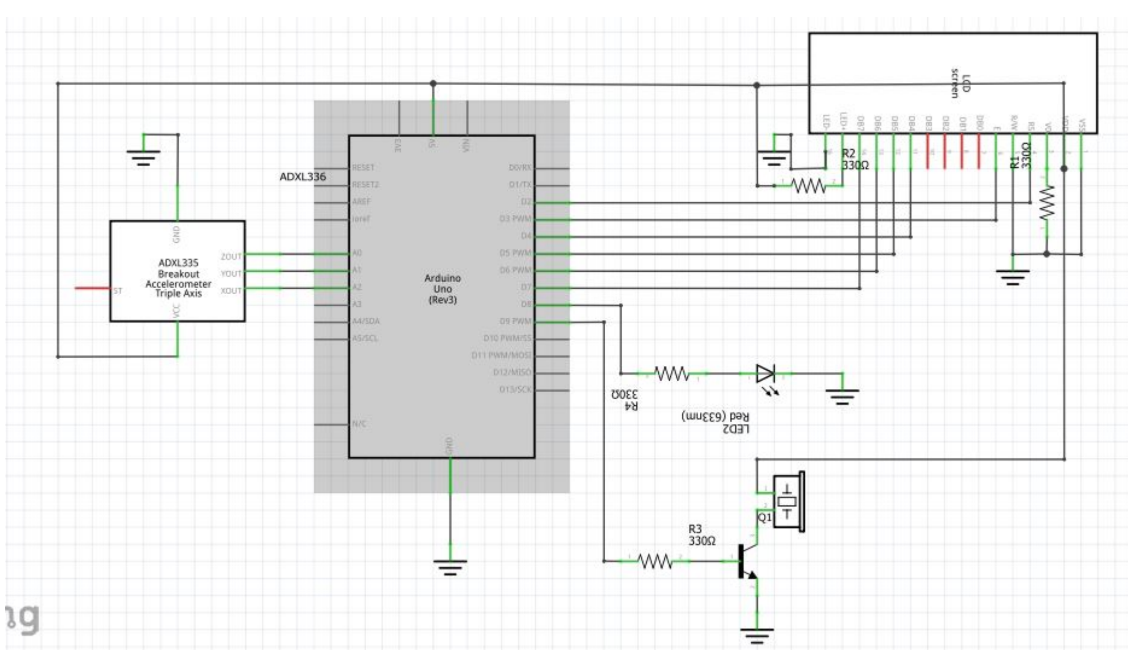


Figure 12: Circuit Diagram of the Proposed Electronic Model

## 7.4 Licenses

1. Arduino IDE is developed and maintained by the Arduino team. The IDE is licensed under GPL.

2. EarthQuake Graph By Circuit Digest

### 7.5 Copyrights

1. https://circuitdigest.com/microcontroller-projects/arduino-earthquake-detector-alarm-circuit

### 7.6 Task Breakdown

- Drawing of Schematic / Circuit Diagram
- Breadboard Troubleshooting
- Designing of PCB
- Printing of PCB Design in Tracing Paper with Laser Printing
- Transfering of PCB through light
- Etching of PCB
- Drilling of holes in PCB
- Placing of components in PCB
- Soldering of Components in PCB
- Making and Testing of Codes in ARDUINO IDE
- Troubleshooting with ARDUINO IDE
- Checking if the device is properly working
- Finalize output device

| Task | Name |
|---|---|
| Paper/Technical Report | Ma. Lourvic C. Espejo / Alyssa Mae S. Lantano |
| PCB Design | Wyder S. Lalangan |
| PCB Etching | Oliver G. Lacaba / Gerry G. Lopez Jr. |
| PCB Drilling | Alyssa Mae S. Lantano |
| PCB Components Soldering | Wyder S. Lalangan |
| Arduino Coder | Ivan Justine E. Cabancalan / Wyder S. Lalangan |
| Troubleshooting | Ivan Justine E. Cabancalan |
| Casing | Ivan Justine E. Cabancalan |
| Runner of Equipment and Materials | Oliver G. Lacaba / Gerry G. Lopez Jr. |
| Polishing of Project | Wyder S. Lalangan |
| Finance Manager | Alyssa Mae S. Lantano |

### 7.7 References

1. https://www.researchgate.net/publication/325619046$_Design_and_Development_of_an_Earthquake_Warning_Syst$ $//aim.gov.in/pdf/Activity_Card_2 - Earthquake_detection.pdf$

2. http://www.aeuso.org/includes/files/articles/Vol7$_Iss26_3582-3594_Earthquake_Alarm_Detector_Microcont.pdf https$ $//www.slideshare.net/MuhammedAnazPK/lifi - audio - transmission - project - report$

3. https://www.draw.io/

4. https://bestengineeringprojects.com/earthquake-detector-indicator-circuit-using-vibration-sensor/

5. https://bestengineeringprojects.com/earth-quake-alarm/

6. https://circuitdigest.com/microcontroller-projects/arduino-earthquake-detector-alarm-circuit

7. https://www.researchgate.net/publication/319291913$_Earthquake_Alarm_Detector_Microcontroller_based_Circuit_for_i$
$//www.yumpu.com/en/document/view/$44321250$/buzzers - and - speakers - e - gizmo/$3

8. https://www.elarms.org/info/AboutElarmS.php

9. https://www.electroschematics.com/3625/seismic-sensor/

10. https://www.e-gizmo.net/

11. https://www.youtube.com/watch?v=enzoLLuu4KAt=12s

### 7.8 Source Code Listings

1. https://circuitdigest.com/microcontroller-projects/arduino-earthquake-detector-alarm-circuit

2. https://bestengineeringprojects.com/earthquake-detector-indicator-circuit-using-vibration-sensor/

3. https://bestengineeringprojects.com/earth-quake-alarm/