

ECE 445: Noise-to-Color Visualizer

Han Young Kim

Hyun Soo Kim

Team 41

TA : Kexin Hui

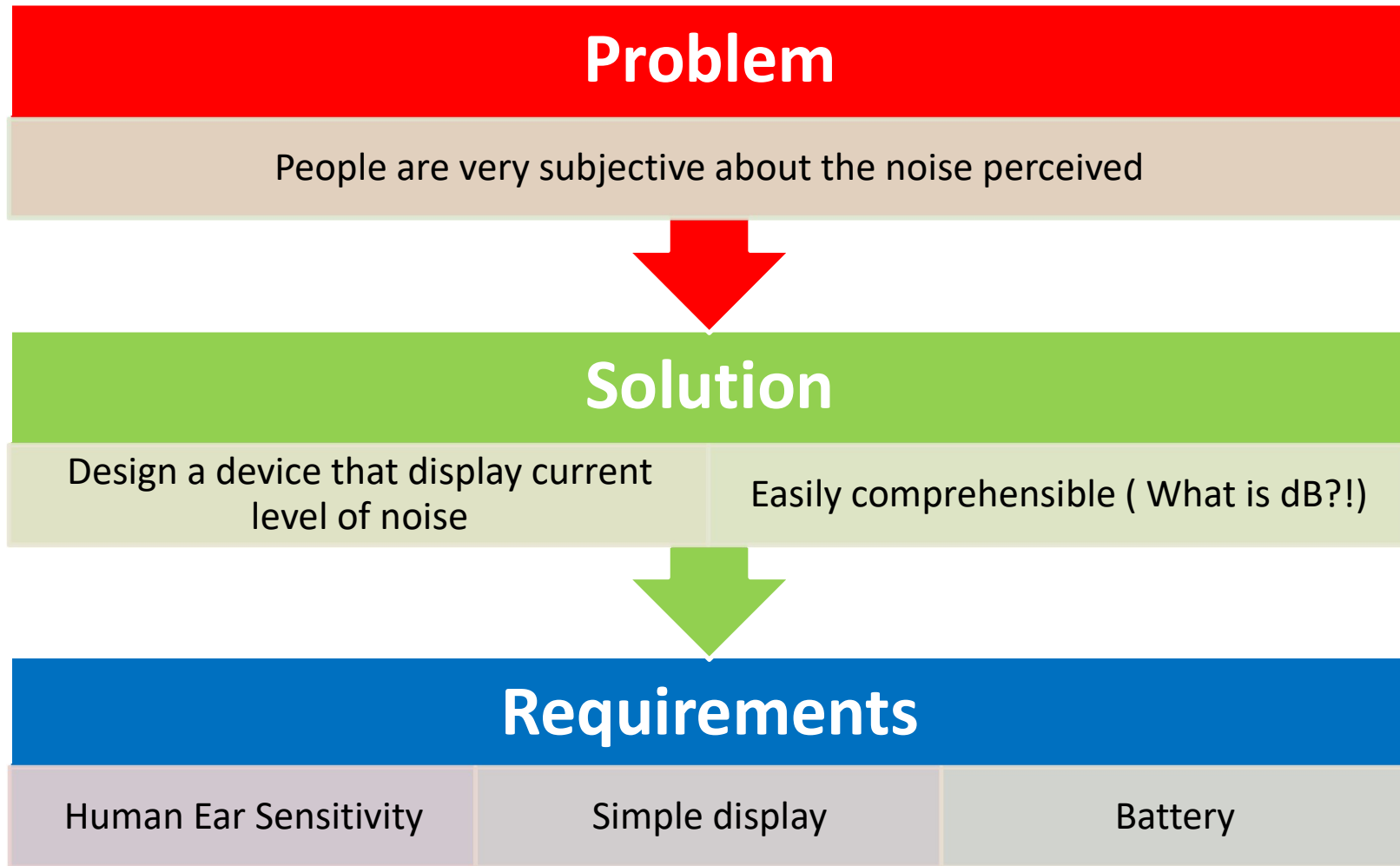


ECE ILLINOIS

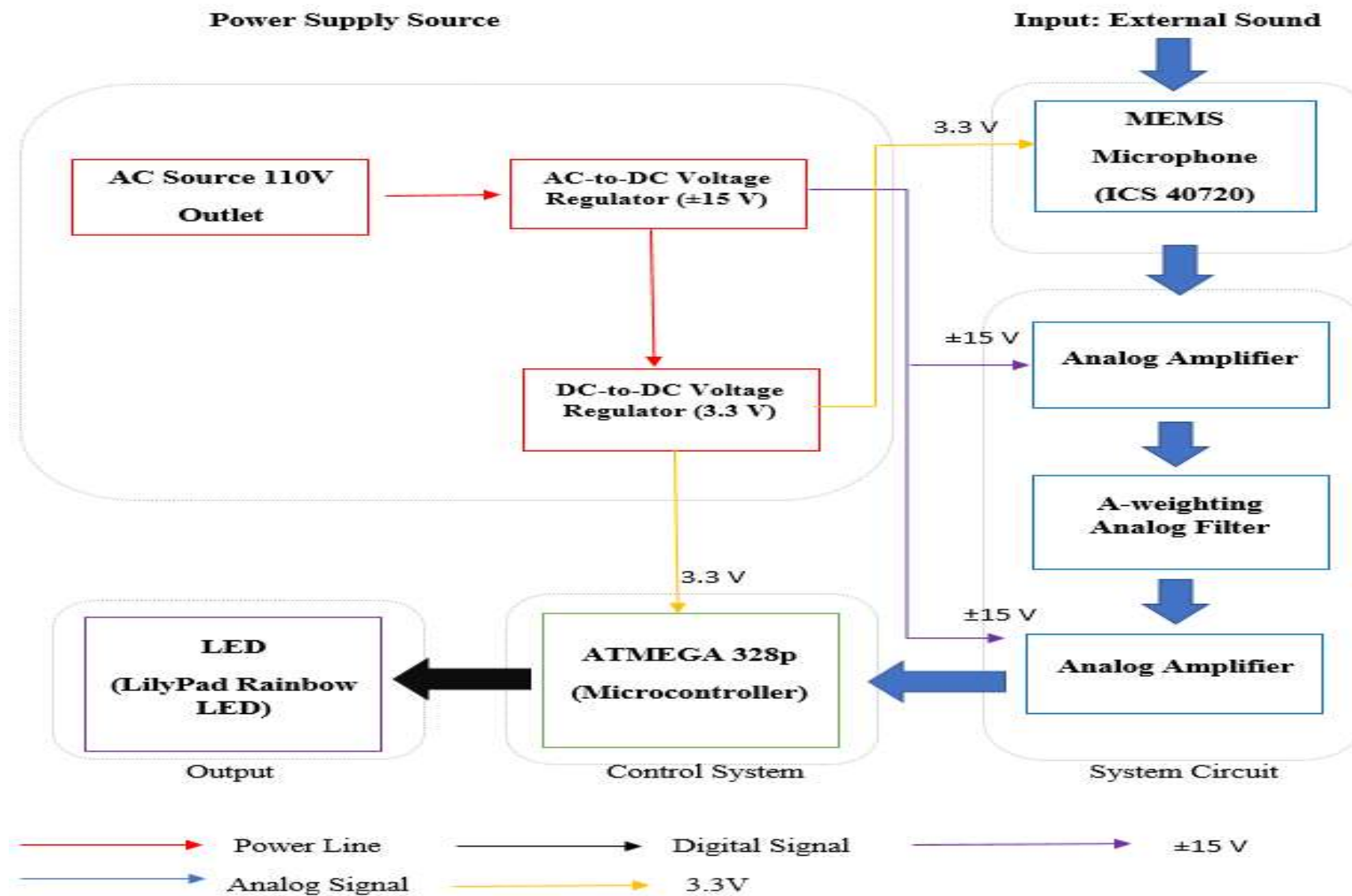


I ILLINOIS

Introduction and Objectives



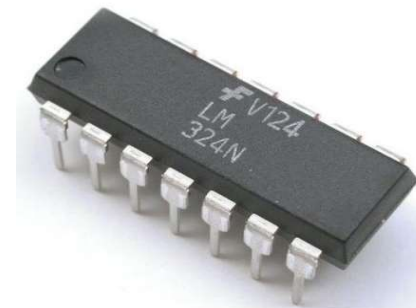
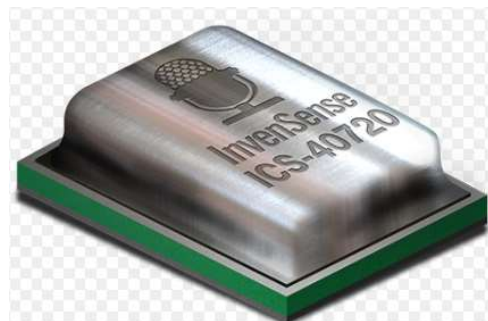
Block Diagram



Power (1): Overview and Goal

- Objective : Build circuit that supplies corresponding voltages to the elements

3.3 V	12 V ~ 15 V
Atmega328P-PU	LM324N
MEMS Mic	



Power (2) : Strategy

110V-to-12~15V

- SPAC265-3W



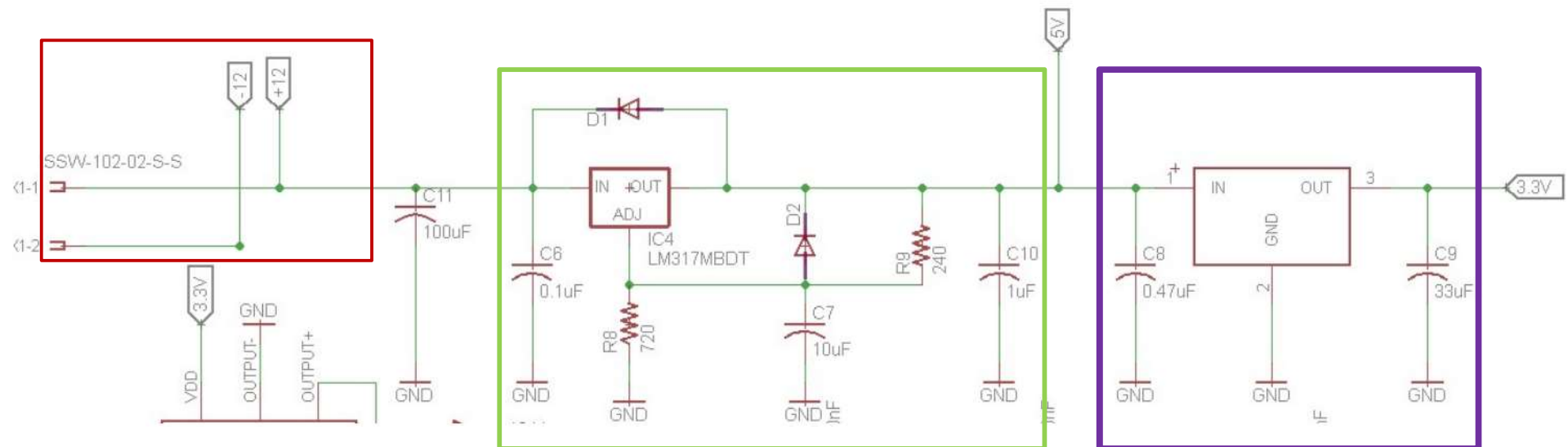
15V-to-5V

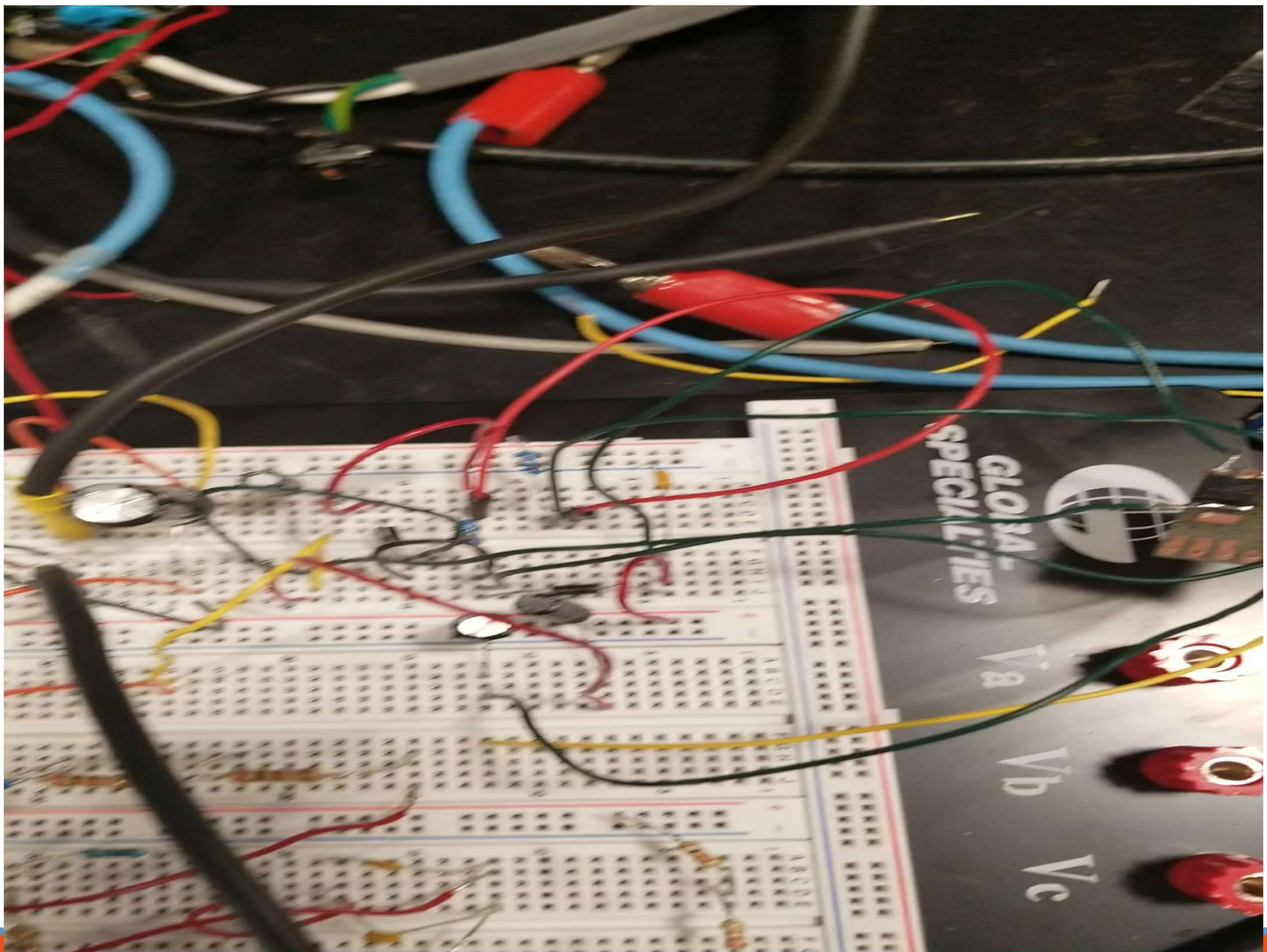
- LM317



5V-to-3.3V

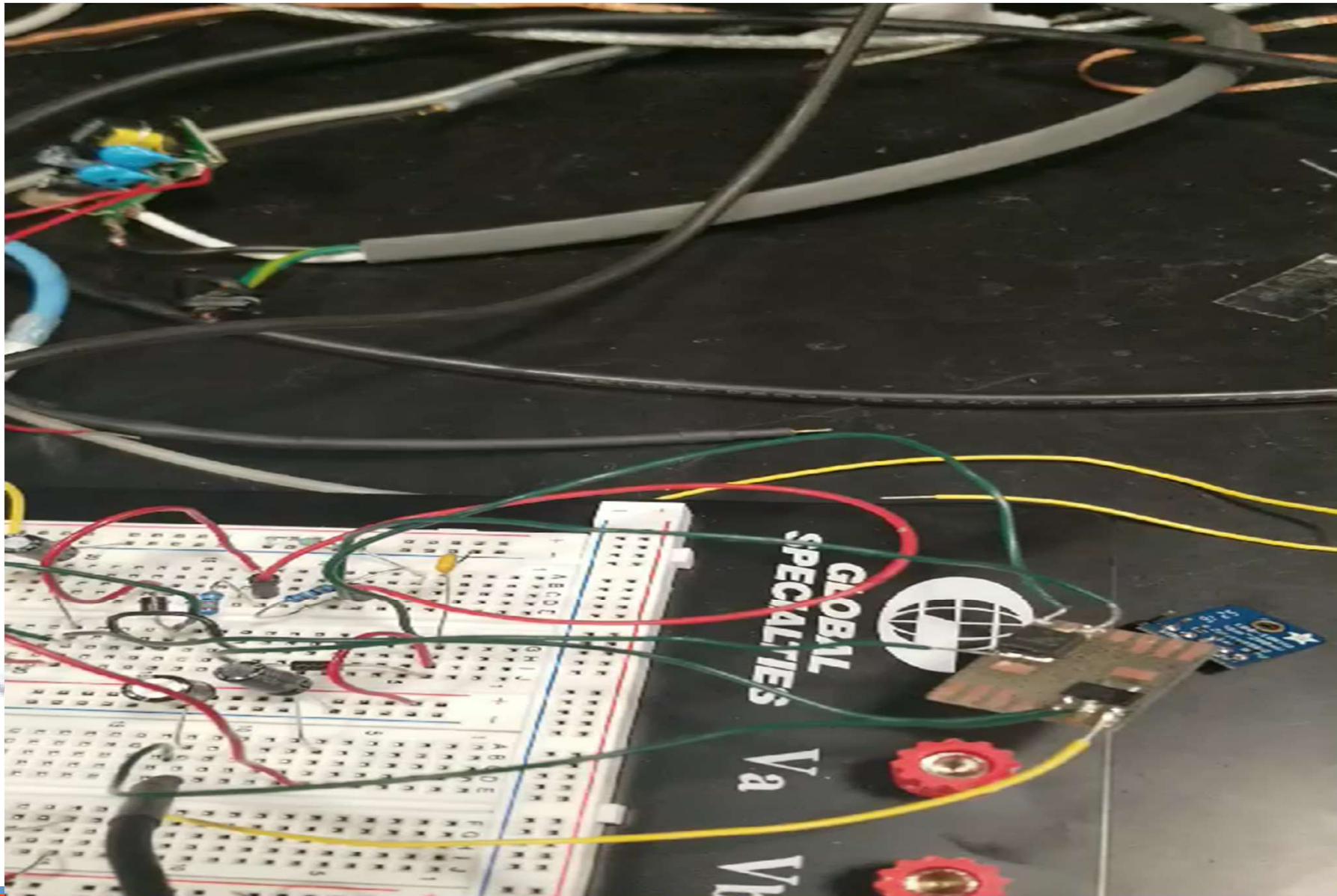
- LM3940







Power (3) : Details in design



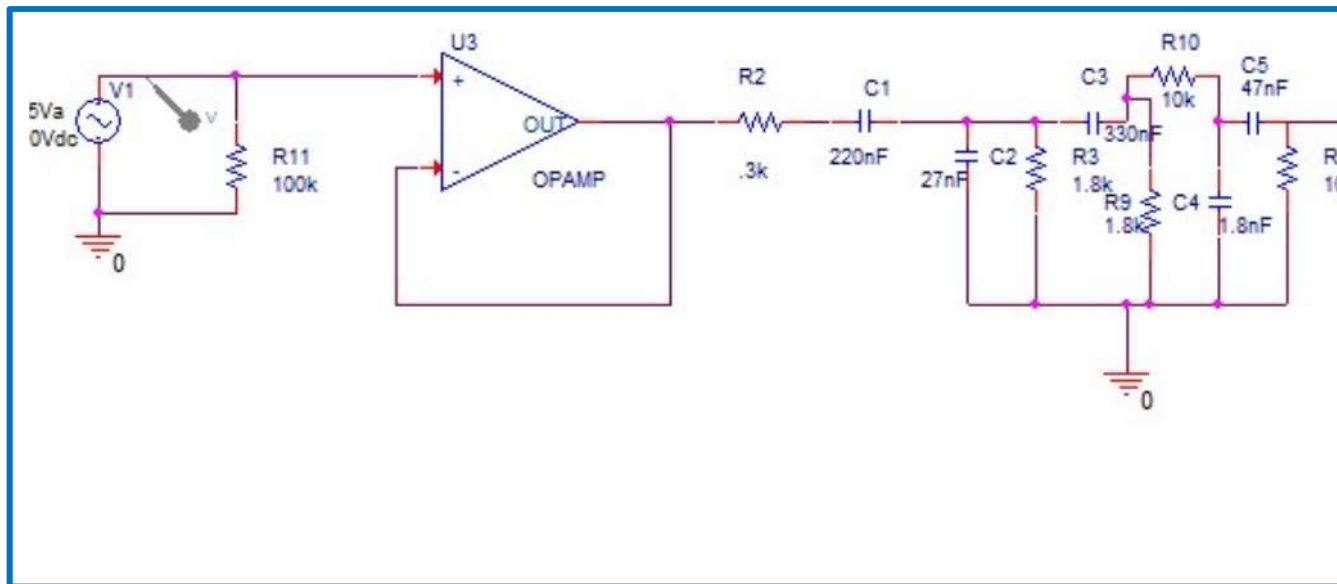
System Circuit (2): Overview

- Objective : Convert to human perception level of the noise level from the machine calibrated range.
- Design A Weighting Filter and Amplifier

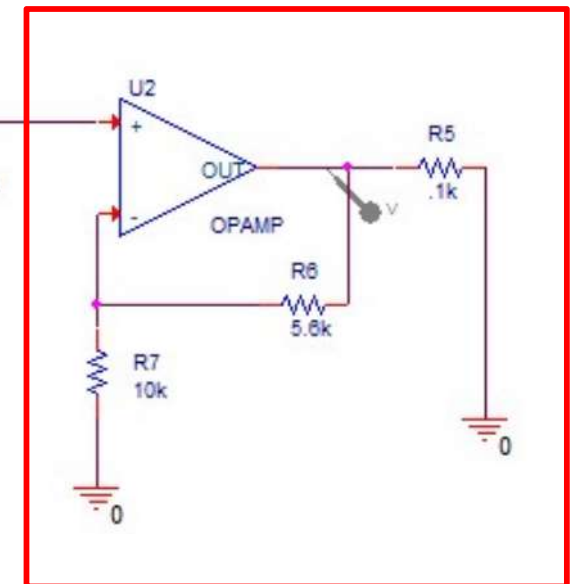


System Circuit (2): Schematic Breakdown

- Objective : Convert to human perception level of the noise level from the machine calibrated range.
- Design A Weighting Filter and Amplifier

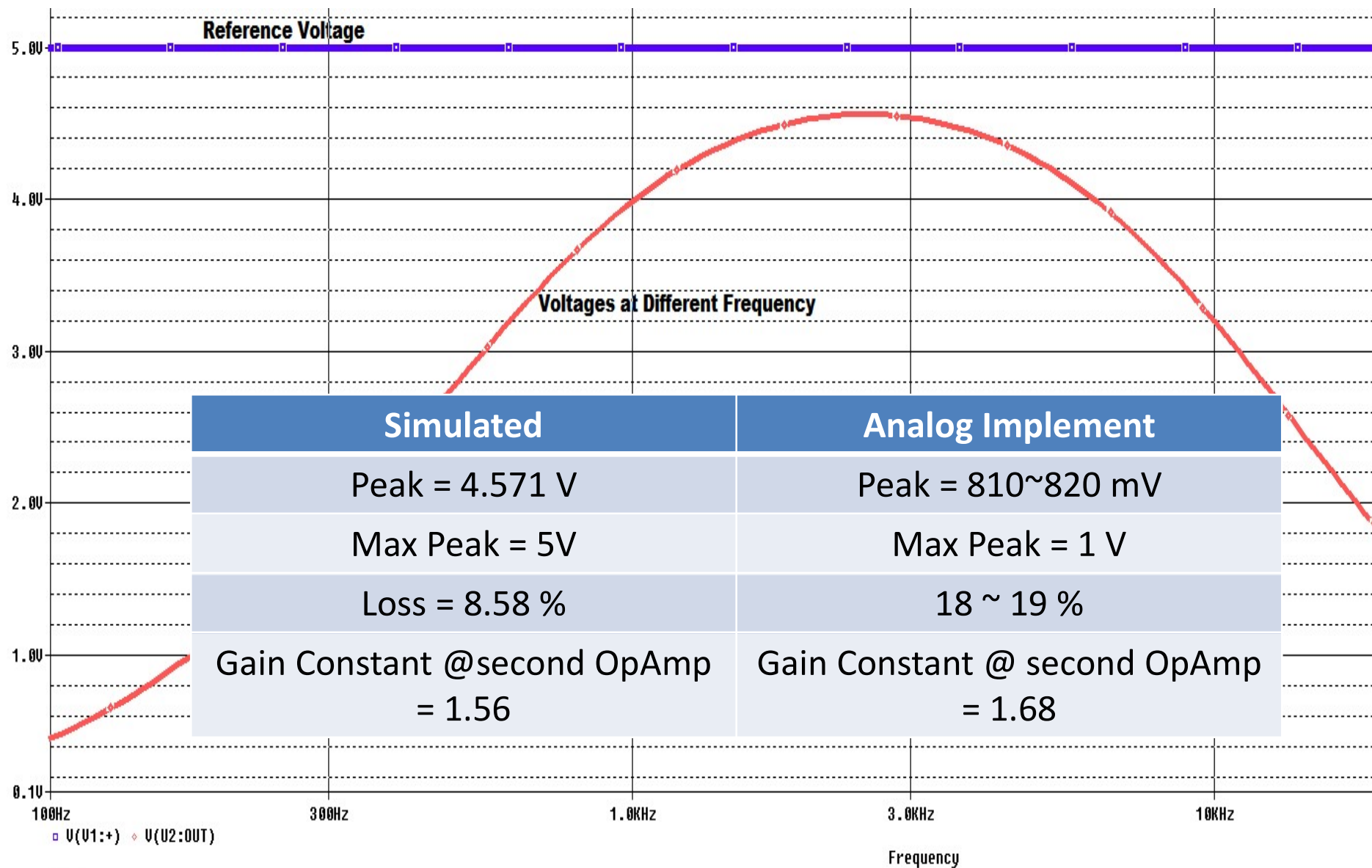


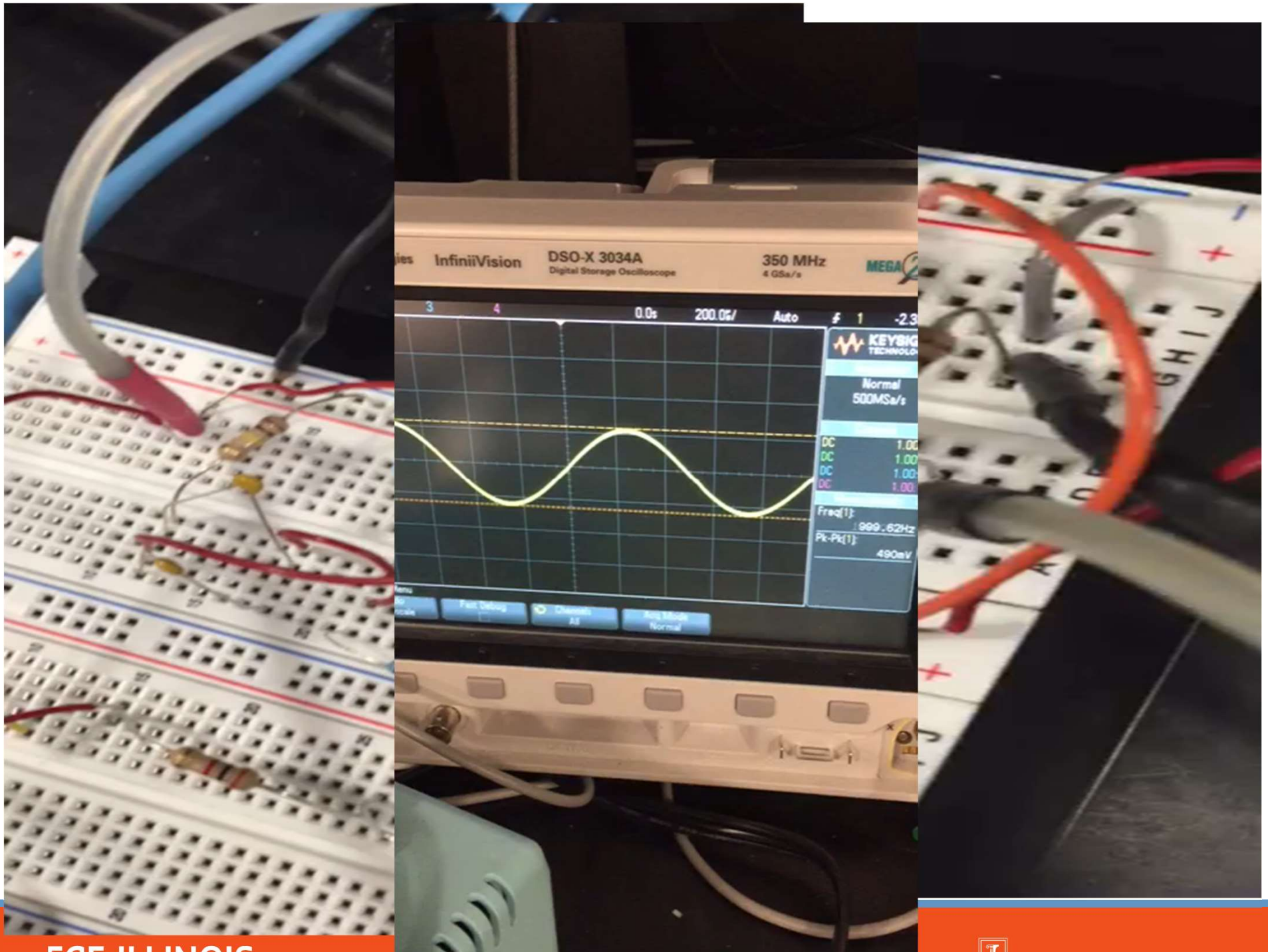
A Weighting Filter



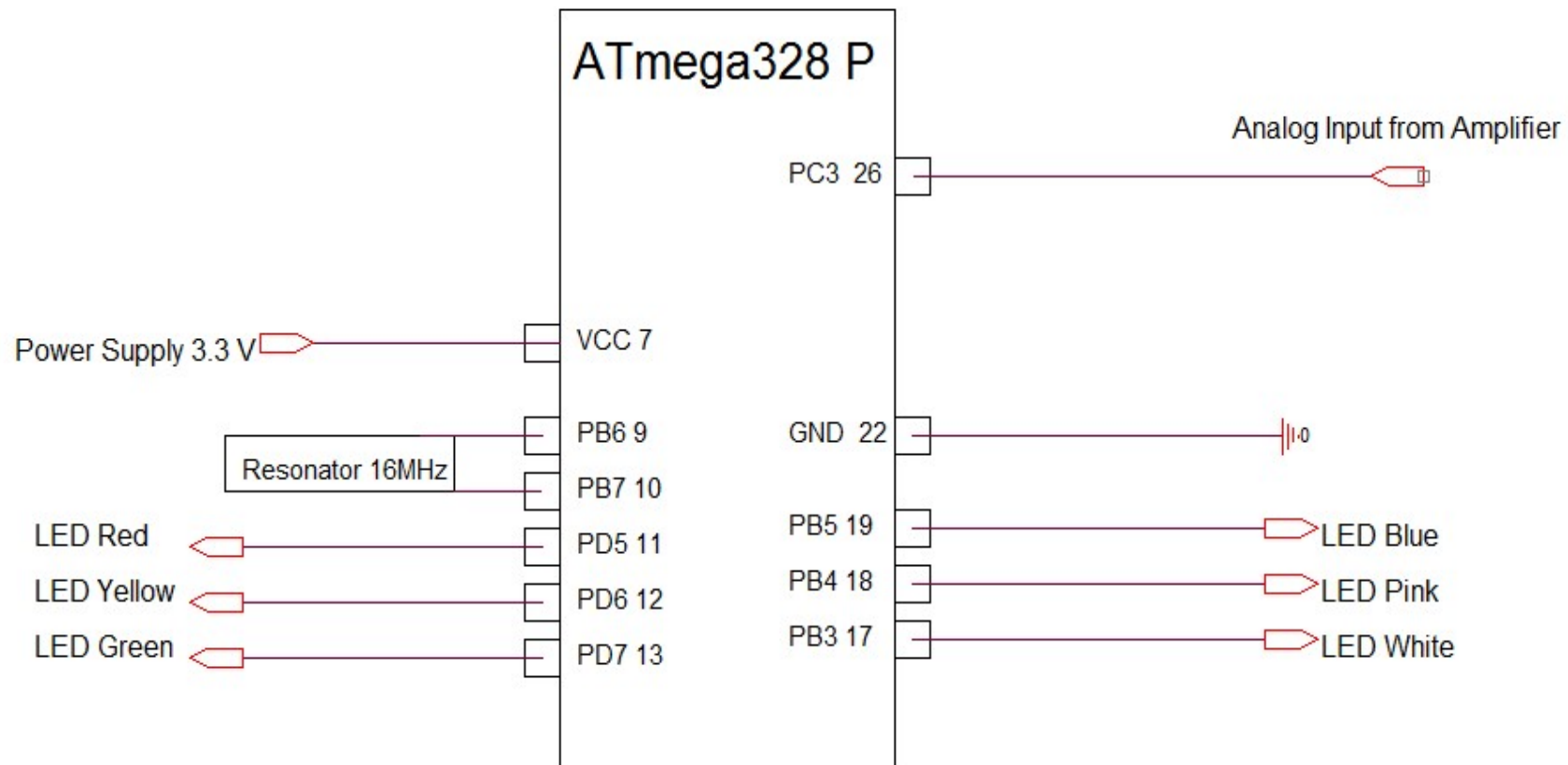
Amplifier

System Circuit (2): Strategy



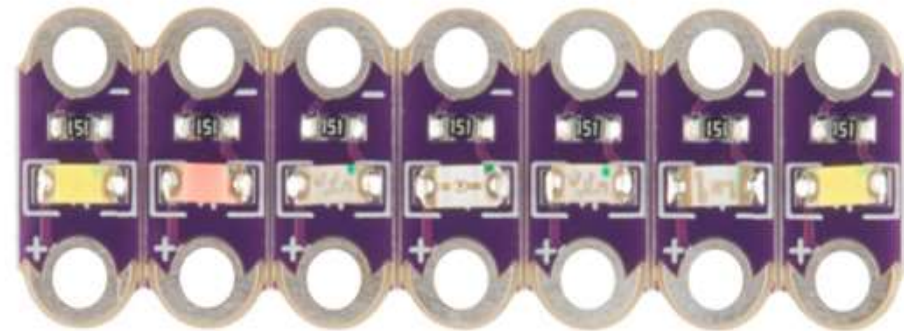


Pin Description of MCU (ATmega328P-PU)



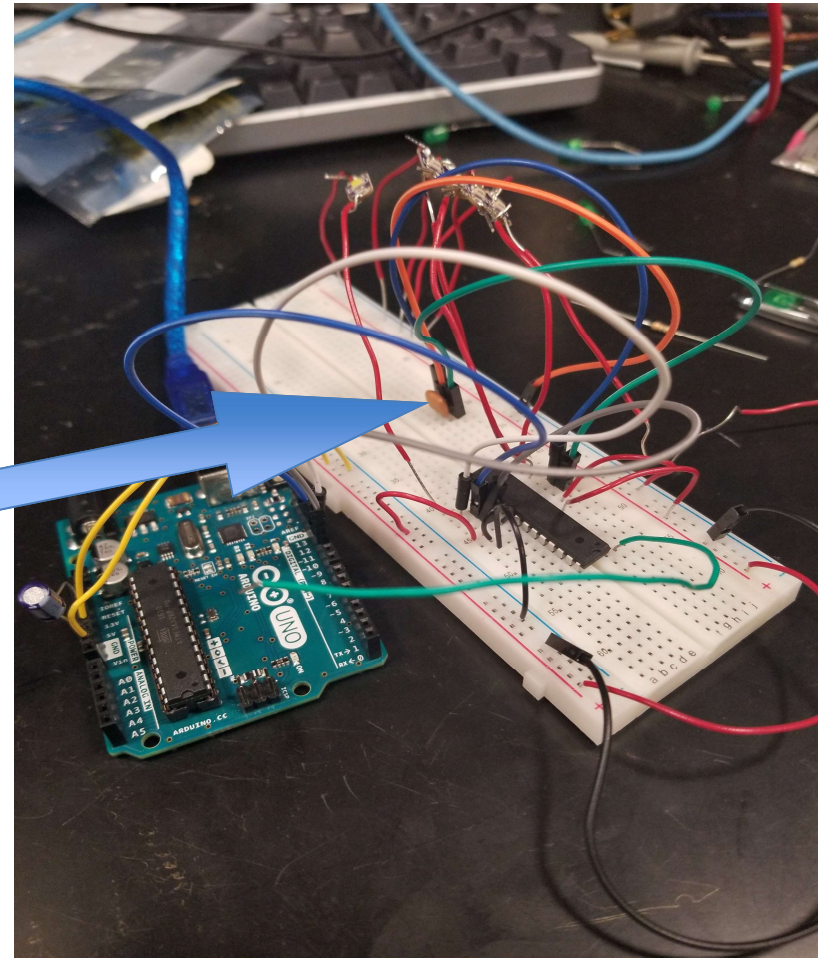
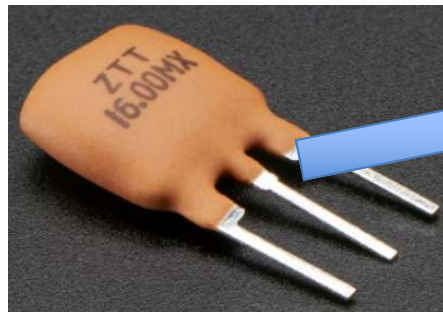
Microcontroller Unit (ATmega328P-PU)

- Why ATmega328P-PU?
 - ➔ It's because Atmega328P-PU supports Analog-to-Digital Converter. We need ADC to obtain discrete numeric input to calculate the dB scale.
 - ➔ Compact-ness of MCU



How to program the Code

- 1) Implement a circuit
- 2) 16MHz Resonator
- 3) Arduino ISP mode
- 4) Program the code



Program Code

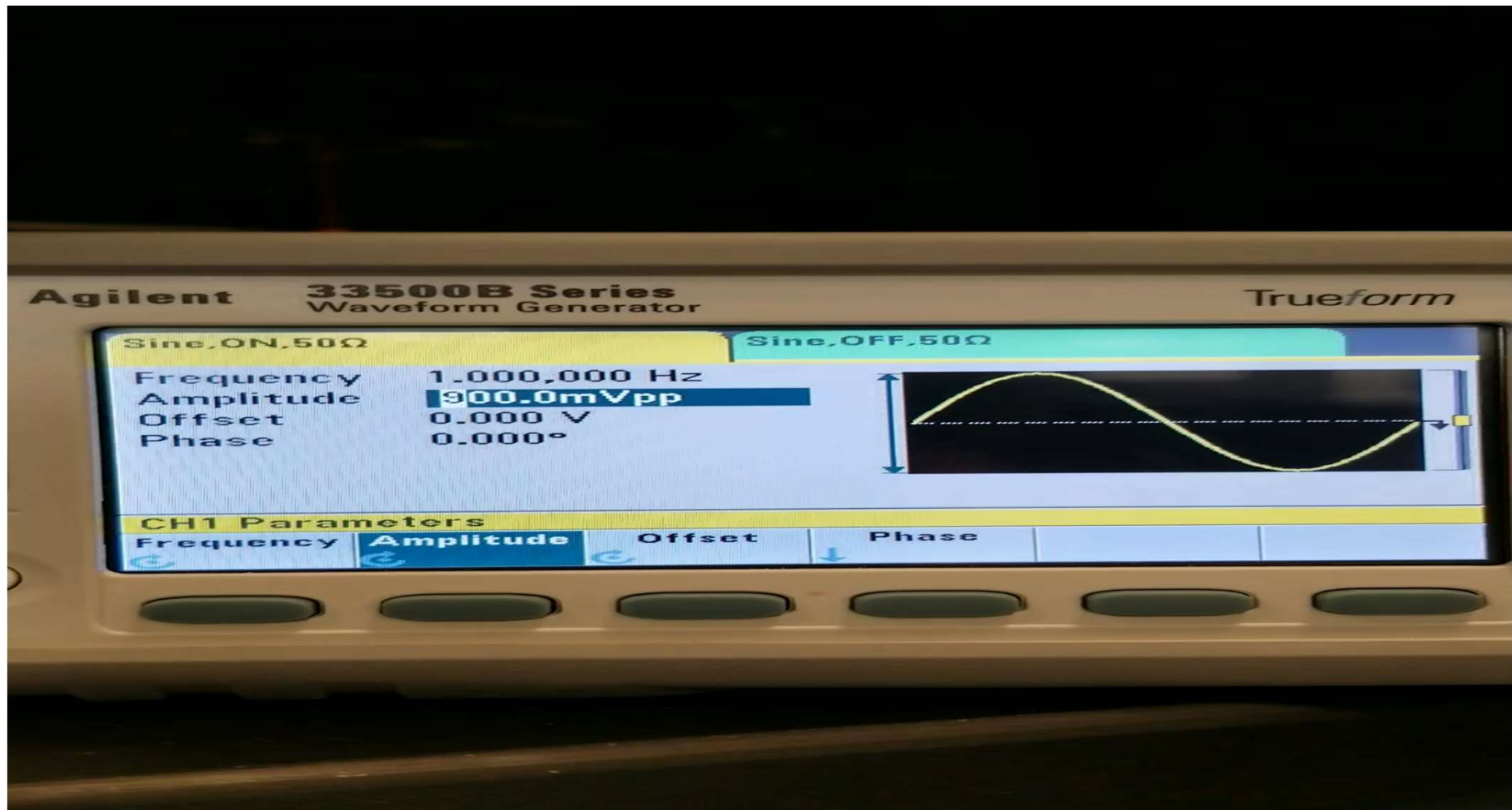
```
int val =0;
float voltage =0;
float db = 0;
float v0 = 0.775;
float a = 0;
const int analogInPin = A3;
int sensorvalue =0;

void setup() {
  // put your setup code here, to run once:
  for(int thisPin = 5; thisPin < 11; thisPin++)
  {
    pinMode(thisPin, OUTPUT);
  }
  Serial.begin(9600);
}

void findLED(){
  sensorvalue = analogRead(analogInPin);
  voltage = sensorvalue * (5.0/1023.0) ;
  a = voltage/v0;
  db = 20*log(a);

  {
    if((0.0<=db) && (db<6.0))
    {
      digitalWrite(5,HIGH);
    }
    else if((6.0<=db) && (db<12.0))
    {
      digitalWrite(6,HIGH);
    }
    else if((12.0<=db) && (db<18.0))
    {
      digitalWrite(7,HIGH);
    }
  }
}
```

MCU: Simulation and Verification



Result

➔ Unfortunately, the system failed at...

1. PCB board integration

- Size of the circuit elements
- Usage of Negative Power Voltage

2. Our input source, MEMS Microphone not working.

- Misjudgment of How small it could be
- Too-close soldering that its internal circuit may get affected

3. Microphone Amplifier

- Some microphones may not work without amplifier

Conclusion and Future Plan

- ➔ Managed to test every block elements of the entire System
 - ➔ Usage of Oscilloscope, Breadboard, and Block-PCB from the store.
- ➔ Improvements
 - ➔ Use Bigger Microphone to collect sounds due to its availability as well as easier implementation using USB 3.0 Port
 - ➔ Use different Microcontroller Chip because Atmega328P-PU has too many pins that it's not space-wise efficient
- ➔ Future Work: Design a far-distance noise detection relative to near-space.

Thank you! Questions?