Noise-to-Color Visualizer (NCV) Device

Team 41

Han Young Kim

Hyun Soo Kim

ECE 445 Final Report – Fall 2017

December 13th 2017

TA: Kexin Hui

Abstract

The purpose of this project is to help people to realize and understand current level of noise they are receiving or generating. 80 dB does not sound plausible enough for people to realize it's a level that would hurt auditory organs.

The project involves the use of microcontroller along with LEDs, MEMS microphone, A-Weighting Filter as well as amplifiers. With LEDs, which we would map each color LED with the certain range of noise level, it's easier and noticeable at a single glance. Say, lighting up red LED when the noise level is above certain level would alert people to keep the noise down and controlled. By coming up with a device that displays the current level of noise, we hope to prevent people from being exposed to noise that may damage ears.

Table of Contents

1. Introduction

1.1. Objective

We are exposed to various intensity of sound daily. Constructions, sound bar, café and streets are all filled with noise generated by many different sources such as cars, animals, and human conversation. However, according to American Academy of Otolaryngology, one in 10 Americans has a hearing loss that affects his or her ability to understand normal speech [1]. Moreover, as the audio technologies advance, the demands for High-Fidelity headphones are skyrocketing, which also affects our hearing as the headphones and earphones greatly stimulate eardrums. Today, 1 in 5 teens has some form of hearing loss, which many experts believe is due to the increased use of headphones, as presented in American Osteopathic Association [2]. Clearly, the exposure to excessive noise may damage hearing, and we have to avoid such situation as much as we can.

Our project is to design a device that visualizes the noise level by classifying the noise levels into six different color: Red, Yellow, Green, Blue, Pink, and White. We will scale red being the loudest and pink being the quietest. By interpreting the decibel into a simple color, we can clearly observe how noisy the surrounding is. Moreover, with this device, people cannot be selfish and subjective about the noise level because the device would indicate the level of sound at a glance.

1.2. Background

Often, we are very subjective about the noise level around us. Some people desire listening to music with booming sound while some just want the music to be controlled at certain level. Or, when we are involved in conversation, we often ignore that the fact we are making noise that disturbs the people around us. There are cases like even in crowded space, some people say it’s manageable while some say the place is very noisy. Considering these facts, we believe it’s just too difficult to be neutral at judging the level of noise. Plus, numeric display of noise level does not actually make sense to children or those who are not familiar in the area. For them and in general, it would be much easier to visualize the noise level with a color so that when we tell our friends about the noise level at certain place, it would give them clear image of how quiet/noisy the place is. Additionally, we would be able to avoid unwanted exposure for loud and disturbing level of noise for several hours.

2. Design

The project has four main modules to represent the system: A-weighting analog filter, power supply, Microcontroller Unit, input and output elements. The input and output element of the system consist of MEMS microphone and LilyPad Rainbow LEDs. MEMS microphone takes the input noise as analog signal, and LilyPad Raibow LEDS indicates the noise level of the input signal. A-weighing analog filter is a circuit that converts the noise to human-friendly noise level. For the power supply section, we used AC source 110V outlet to serve power to each components of the system. The MCU converts analog signal of the noise to digital signal, and we can program microcontroller to derive the output that we expected.

2-1. Physical Design

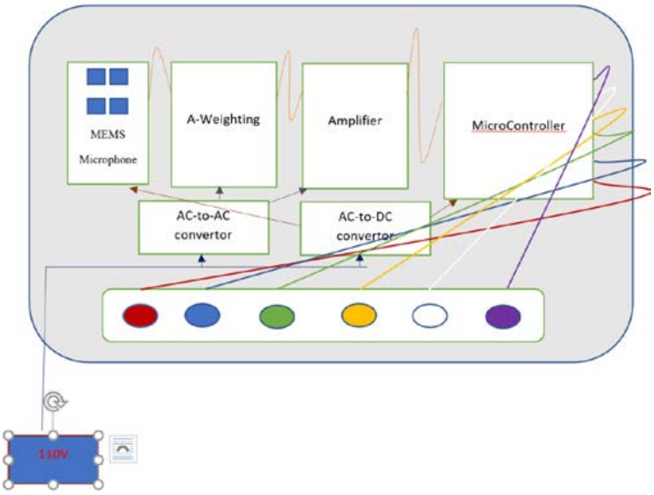

Figure 1. Physical Design

The physical feature of this device is the LEDs placed on the bottom so that it's easily noticeable to the people watching device. We also considered its portability so that it could be hung around the wall or the window-side, indicating how quiet the place is currently.

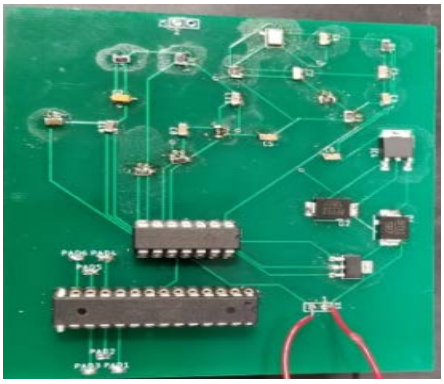

Figure 2. PCB Outlook

2-2. Device Block Diagram

We would divide the device into sections: Power Supply, Input-to-Output (I/O) system Circuit, and control system.

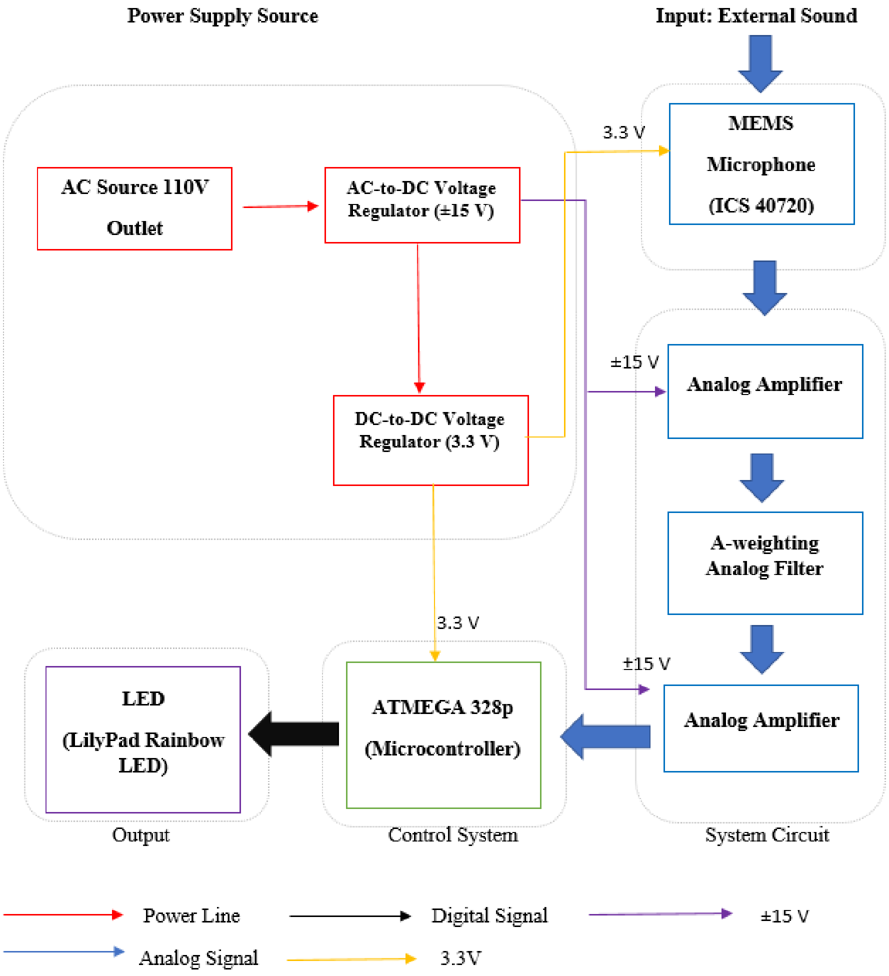

Figure 3. Block Diagram

2-3. Power Supply

To supply power into the device, we will use 110 V outlet from the wall or any adaptors. First, using SPAC265-3W module, AC voltage is then converted to 12~15 V. We will then use linear voltage regulators by implementing AC-to-DC converter to supply voltage into the microcontroller, which operates with 3.3 V. MEMS microphones also requires 3.3 V.

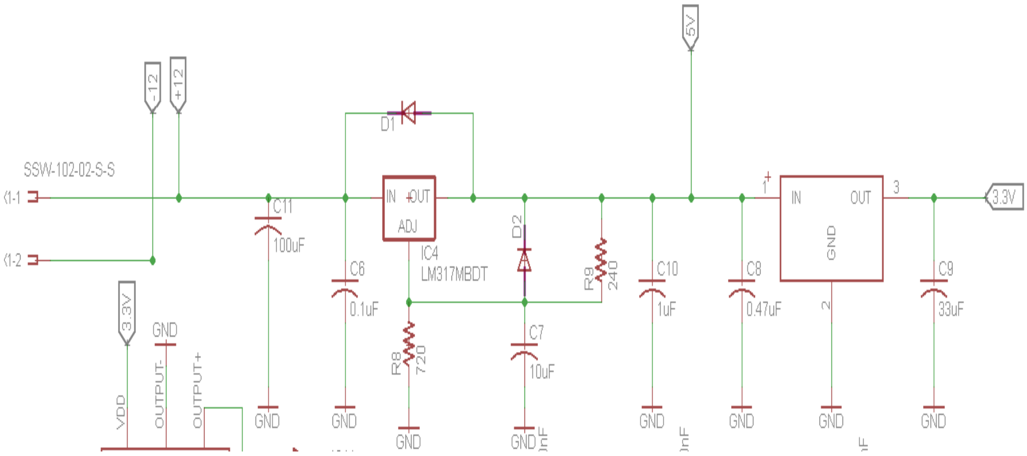

Figure 4. Power Circuit Schematic in EAGLE

2.3.1. Calculation for DC-to-DC converter (12~15V-to-5V)

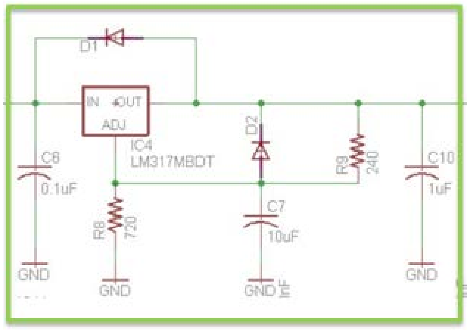

Figure 5. 12~15V-to-5V Converting circuit

With the circuit above, LM317 has $V_{REF} = 1.25\ V$, which would be used for the equation below.

$$V_O = V_{REF}\,(1 + R2 / R1) + (I_{ADJ} \times R2) \qquad (2.1)$$

To get 5V, we need the ratio of $\frac{R2}{R1} = 3$ while $I_{ADJ} * R2$ can be neglected because it is in micro-scale. Thus, R2 here is R8, which is 720 Ω. R1 here is R9, which is 240 Ω, and the ratio is 3, allowing 5V.

2.3.2. Calculation for DC-to-DC converter (5V-to-3.3V)

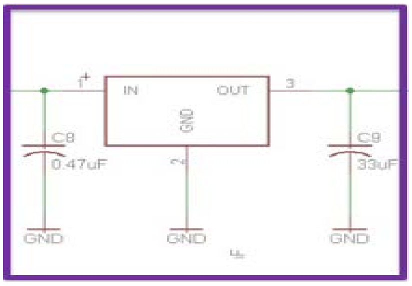

Figure 6. 5V-to-3.3V Conversion

LM3940 converts the input 5V within ±10% range (4.5 ~ 5.5 V). To stabilize the output voltage, capacitors were used. The output, 3.3V, is then supplied to Microcontroller and MEMS microphone.

Moreover, to make the soldering process easier than the through-hole implementation.

2.4 I/O System Circuit

2.4.1 MEMS(Microelectromechanical Systems) Microphones (Model: ICS 40720) [6]

Operating Voltage : 3.3V

Input: Collected sound and noise from surroundings

Output: Analog signal of the collected sound and noise

MEMS microphone will collect the noise level of the surrounding. To be accurate, omni-directional MEMS microphones is used to collect noise samples from different directions in respective to the user. Another fact about microphone is that it is already Band Pass filter because it detects the sound between the range 75 Hz to 20 kHz, according to the specification [6].

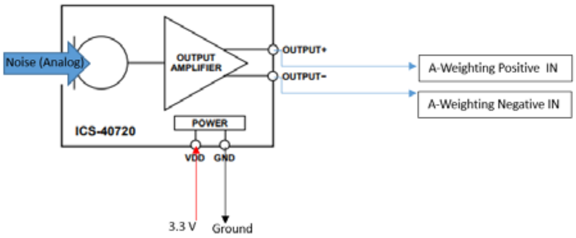

Figure 7. Outlook of MEMS Microphone [6]

Table 1. MEMS Microphone Pin Description

PIN Name	Decription
VDD	Supply Voltage
GND	Ground
Output+	Positive Analog Output Voltage
Output-	Negative Analog Output Voltage

2.4.2. A-Weighting Filter

Operating Voltage : ± 15 V for Op-Amp (**LM324N**) <DC>

Input: Collected analog noise signal from MEMS microphone

Output: Filtered analog signal

Table 2. A-Weighting Filter & Amplifier

Product Used	Purpose	Adjusted (if applicable)
LM324 (First Op-Amp)	A-Weighting Filter	Unity Gain (None)
LM324 (Second Op-Amp)	Amplifier (non-inverting)	

By definition, A-Weighting Filter took model human ear perception because human ears are unable to sense certain high frequency sound levels as supported with the graph below.

When measuring the sound level through machine, this is not very accurate description of what people perceive because machines and human ears have different sensitivity. Human ear responds more to frequencies between 500 Hz and 1 kHz and less sensitive at certain range of 1 kHz to around 8 kHz [3]. This means that at the peak of the graph shown above, there is a matching point that intersects 0 dB. Therefore, weighting filters are used to help converting instrument-measured sound level to relative human-hearing loudness. Typically, A Weighting Filter is most commonly used filter for measuring the sound level because it effectively cuts off lower and higher frequencies that average person cannot hear, which resembles human ear.

As implementing the schematics for the A-Weighted filter, we set the Op-Amps within the filter to be operated at voltage up to ±32V, using the model LM324 [4], with Unity gain for U3 for fully compensated internal frequency.

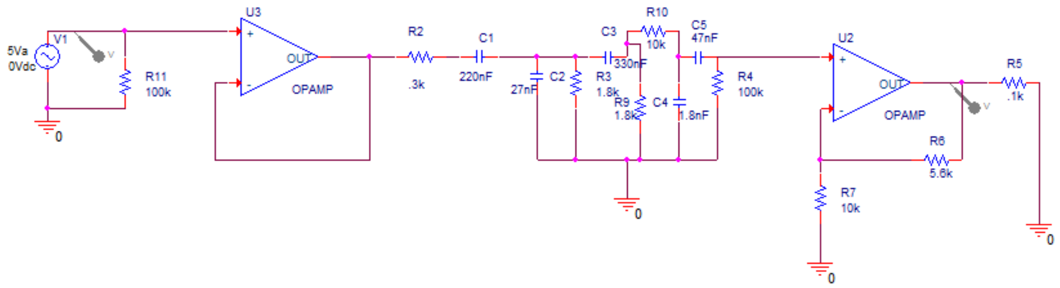

Figure 8. Schematics for A-Weighting Filter [5]

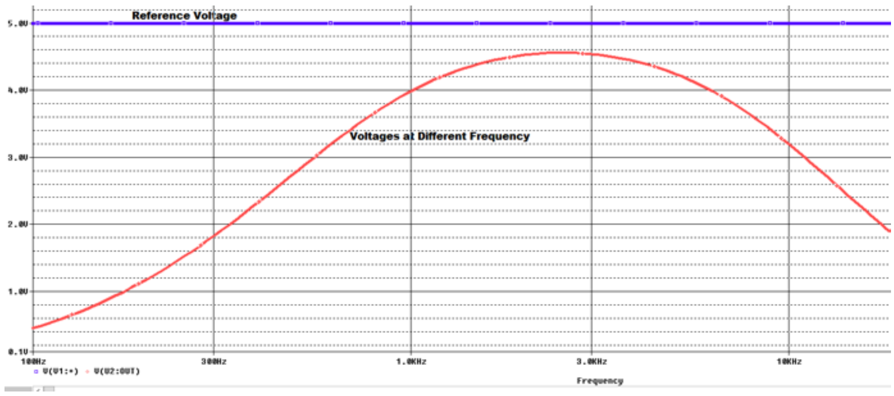

Figure 9. Voltage Response from 100 Hz to 20kHz

From the graph above, the frequencies below 100 Hz are not observed. This phenomenon can be explained by the frequencies under 100 Hz demonstrate slow curvature in the area of speech processing, forming irregular behavior and thus making them difficult to be observed. The most important frequencies for speech fall between the 250 and 6000 Hz [7], so this would mean that the noise generated by human falls between that range.

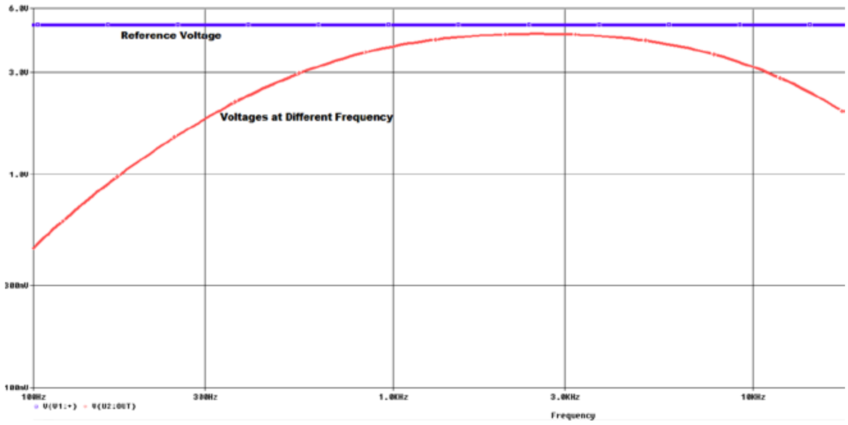

Figure 10. Log Scaled version of Voltage Response when generated with AC ±5V source

2.4.3. Amplifier

Operating voltage Range: ±15 V for Op-Amp <DC>

Input: A-Weighting Filter output analog signal

Output: Amplified signal of the input

Problem: Since the input signal amplitude is varying depending on the loudness, the specific value of gain coefficient is yet determined.

For amplifier, we used Non-Inverting technique using Op-Amps, which has gain constant of

$$A_v = (1 + \frac{R6}{R7}) \qquad (2.2)$$

Originally, R6 = 5.6 kΩ and R7 = 10kΩ. The gain constant is therefore 1.56. However, this is the case when there is no loss of signal in the circuit. Since hardware circuits generally possess loss in the amplitude of signal as the signal passes through different elements, we would like to implement circuit for an amplifier to restore back to the original amplitude of signal. The gain of the amplifier depends on how much voltage level is achieved by the A Weighting filter. For example, when the original signal has amplitude of 5V and post-A Weighting filter amplitude was observed as 4.571 V.

Table 3. Difference between Simulated and Analog Implementation

Simulated	Analog Implement
Peak = 4.571 V	Peak = 810~820 mV
Max Peak = 5V	Max Peak = 1 V
Loss = 8.58%	18~19%
Gain Constant @ Second Op-Amp = 1.56	Gain Constant @ Second Op-Amp = 1.68

Since not dealing with the stable source of AC signal, it is a dilemma to use time-invariant gain constant. Thus, since the noise level of the surroundings may vary depending on how loud and quiet the inputting noises are, the gain coefficient of this amplifier must be flexible or derive relationship between voltage level and the amount of gain logically deduced after sufficient number of calculations and measurements with different amplitudes of the noise signals.

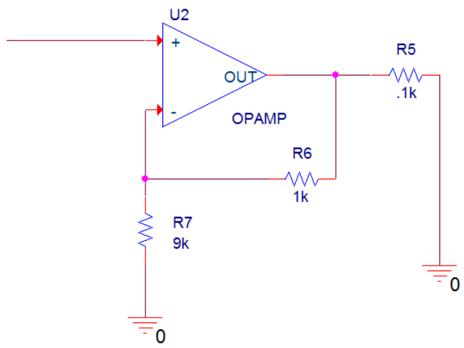

Figure 11. Original option of Amplifier Circuit with gain coefficient of 1.11

First, the noise should be collected using the MEMS microphone from various environments and surroundings and then draw correlation between the maximum input signal (voltage) amplitude and amount of gain needed for the peak voltage level to reach exact 0 dB in log scale.

Second, the reference voltage was determined by the maximum voltage that MEMS microphone could generate, the loudest sound.

For the Op-Amp chip we will be using is the LM324 shown below. This chip would also be used in the A-Weighting filter.

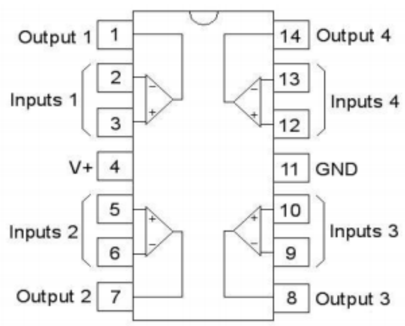

Figure 12. Pin Description of LM324

2.4.4 LED – LilyPad Rainbow LEDs

Operating voltage Range : 3.3V <DC>

Input: Digital Signal from Microcontroller

Output: LED light

As the microcontroller makes instruction, the command signal would pass to LED panel to light up certain color. There are total six colors of LEDs, each with different colors, and depending on the sound level, one of the LEDs would light up. Lilypad Rainbow LED has 6 colors (Red, Blue, Green, Yellow, pink, and White) and furthermore can be customize the color as customer's sate. Since the device is time-varying system, the LED light keep changing according to the analog input signal at every time.

2.5. Control System

Microcontroller Model: ATMEGA 328P

Operating Voltage Range: 1.8 – 5.5 V.   Allow ± 5% tolerance range

Input: Analog Signal from amplifier circuit.

Output: Digital Signal that control LED light up.

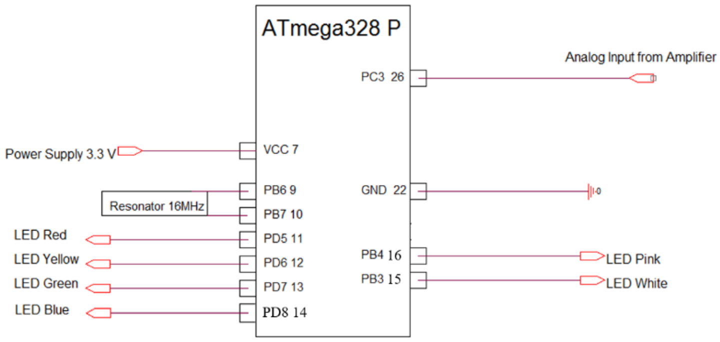

Figure 13. Pin Description of ATmega328P

The ideal microcontroller for the project should be able to support Analog-to-Digital Converter (ADC) embedded and the functionality of FFT as well as programmable. Considering these qualifications, ATEMGA 328P is most desirable to operate the system.

To program the ATMEGA 328P, it should be connected to the Arduino UNO to access to the Arduino IDE program where the programming sketch(code) can be written and uploaded to ATMEGA 328P. To connect ATMEGA 328P to the Arduino UNO, the pin 10, 11, 12, 13 of Arduino UNO should be wired to pin 1, 17, 18, 19 of ATMEGA 328P respectively as shown **Figure 14.** In addition, a 10μF capacitor is desired to use for the stability of the reset signal and it is connected to Reset and GND pin of the Arduino UNO. Lastly, 16MHz resonator should be connected at pin 9 and 10 of ATMEGA 328P to enhance the clock signal to match with the clock speed of the Arduino UNO.

After implementing the circuit, the code can be programmed into the ATMEGA 328P by using Arduino IDE program. In the Arduino IDE program, the mode of the program should be changed to Arduino ISP.

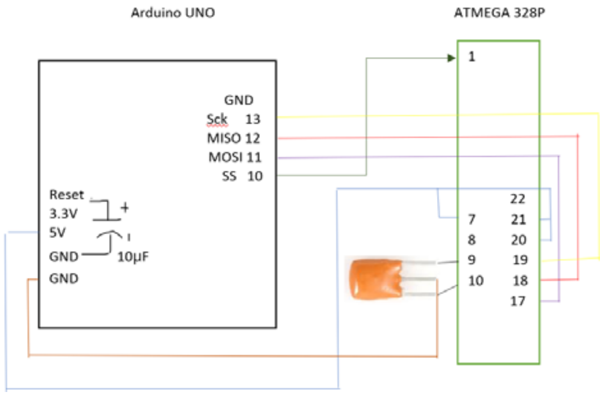

Figure 14. Connection between ATmega328P and Arduino

For the programming code, the analog signal of the noise should be converted to the decibel value to indicate how much noise is. As microcontroller receive the analog signal from amplifier circuit, it converts the voltages into decibel scale using the logarithm.

$$P_o(Watts) = 2 * 10^{-6} \ \ Pascals \tag{2.3}$$

$$L(dB) = 10\log\left(\frac{P}{P_o}\right)^2 = 20\log\left(\frac{P}{P_o}\right) \tag{2.4}$$

The equation for the power is

$$P = VI = \frac{V^2}{R} \tag{2.5}$$

Note that since the signal went through the analog circuits, the resistance is same for P_o and P can be cancelled off. Thus, the calculation of noise level is now just

$$L \ \ (dB) = 10\log\left(\frac{V}{V_o}\right)^2 = 20\log\left(\frac{V}{V_o}\right) \tag{2.6}$$

Here, V_o is 0.775 conventionally and V is the varying voltage amplitude of signal.

3. Design Verification

3.1 Power Circuit

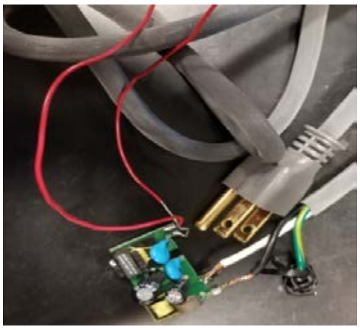

Figure 15. SPAC265-3W Module and ROHS power cord

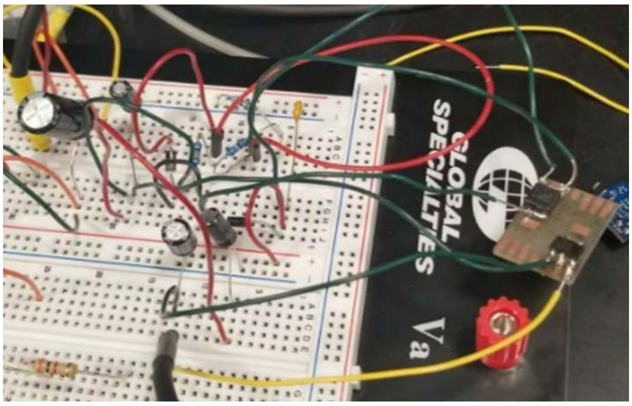

Figure 16. Overall Design of power circuit

Table 4. Result for Power Circuit Test

Component	Expected Range	Obtained	Error
SPAC265-3W	12 ~ 15 V <DC>	13.118 V	N/A
LM317	4.9 ~ 5.1 V	4.694	4.2 ~ 7.9 %
LM3940	3.20 ~ 3.40	3.38	2.4 %

To verify the requirement of the power supply, we first built a power circuit to measure the voltage as shown **Figure 16.** By using the SPAC265-3W, the AC 110V outlet is converted to 13.118 DC V as shown **Table 4**, and the power circuit is initially supplied with 13.118V. With the power circuit containing capacitors and diodes, it can prevent energy loss within the circuit that system stably convey the energy to gain the 5V and 3.3V respectively, and protect the circuit from the short circuit. By using resistor values to 240 Ω and 720 Ω, we can derive the 4.7 DC voltage when we measure the point after the LM317. Lastly, we get 3.38 DC voltage by using the LM3940.

3.2 A-Weighting Filter

We took three steps of verifying the A Weighting Filter and amplifiers.

First, we took PSPICE simulation before actual implementation of an analog circuit because the simulations are always assuming the ideal case that all components are working properly with stable source. As we simulated the **Figure 8** above, the result is shown below.

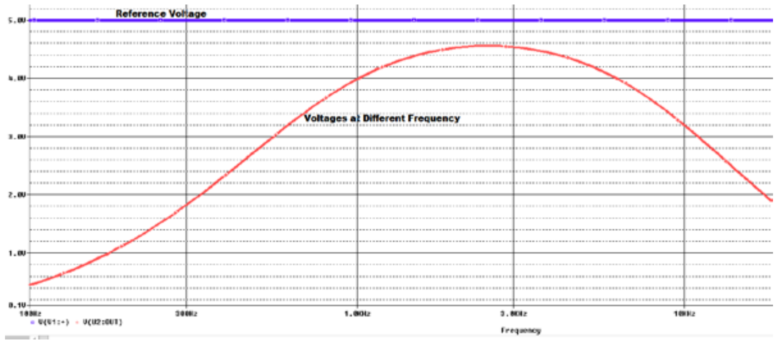

Figure 17. PSPICE Allegro simulation of A-Weighting Filter in Frequency Domain

The second step is to verify whether the filter is working along with the amplifiers. To test out the functionality of the filter as well as the amplifiers, we used an arbitrary and stable signal from the function generator with frequency of 1 kHz, which is within the range of human perceptible frequency.

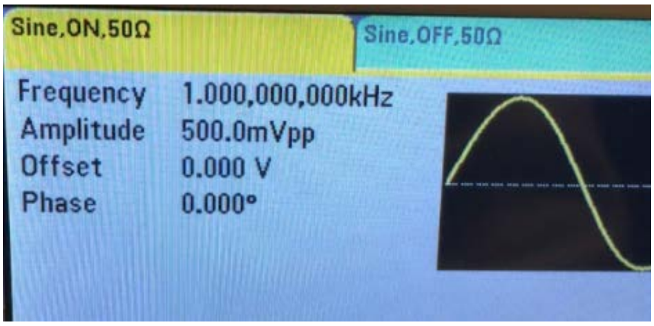

Figure 18. Test Signal of 1V (500 mV peak-to-peak)

The first amplifier, which is within the A-Weighting Filter, has unity gain to ensure its stability.

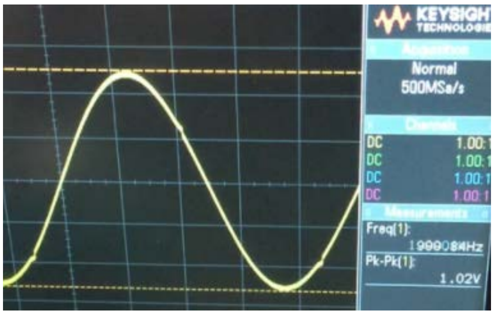

Figure 19. Output of 1.02 V after Unity Gain

We then noticed that there has been some loss as the signal passes through each components of circuit. This showed the need for further increase in the Gain constant than the expected.

Therefore, to adjust and recover the portion of lost signal amplitude as well as to boost up the signal itself to it reaches close, which would be resolved by implementing additional amplifier circuit with higher gain.

3-3. Amplifier

Unfortunately, due to the loss shown above, we designed additional amplifier that not only boost up overall amplitude to get close to the 0 dB but also recover some of the lost signal, with the Gain constant with calculated resistance limit.

Figure 20. Recovered and adjusted (810 mV) Signal after

Taking the output of A-Weighting Filter, which is 480 mV, the amplifier boosted up and adjusted to the signal close to the original amplitude while keeping the tolerable resistance ranging from 6.8 kΩ to 7.1 kΩ because above that range it would surpass the amplitude of original input. With a gain 1.69 while keeping the tolerable resistance range, 810 mV was obtained and therefore generated a signal stable enough as the output.

3-4. ATMEGA 328P and Lilypad Rainbow LED

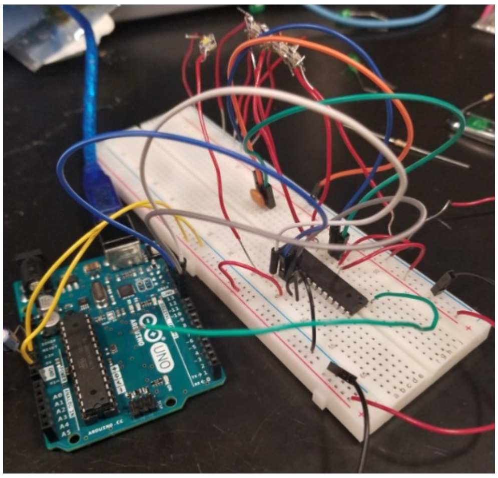

Figure 21. Connection between ATMEGA328p and Arduino UNO

Because of the malfunction of MEMS microphone, we used the function generator to create random analog signal and the signal goes into the pin 26 of ATMEGA 328P as shown **Figure 13**. We set each LED light to respond to the specific range of decibel values by programming the code in the ATMEGA 328P.

Table 5. The range of the decibel for each LED

LED Color	Decibel Range(db)
Pink	0 to 4
Red	4 to 8
Yellow	8 to 12
Blue	12 to 16
Green	16 to 20
White	20 to 24

First, we set the input signal is sin function to indicate that the system of the device is time-varying. Then, we increased input voltage from 0V to 7.5V. As we expected, the LED lights kept changing because input signal is sin function and the number of LED that light up increased as input voltage increased, but the only one of LEDs was turned on at that time.

Table 6. Result of the LED light according to the input signal

Input Voltage(V)	LED light
0(inf db)	Pink
1 (2.21db)	Pink
1.5 (5.74db)	Pink, Red
3 (11.76db)	Pink, Red, Yellow
4.5 (15.27db)	Pink, Red Yellow, Blue
6 (17.78db)	Pink, Red Yellow, Blue, Green
7.5 (19.72db)	Pink, Red Yellow, Blue, Green

For the programming code, we used decibel equation and if-else statement to turn on the respective LED lights according to the input signal.

```
void findLED(){
    sensorvalue = analogRead(analogInPin);
    voltage = sensorvalue * (5.0/1023.0) ;
  a = voltage/v0;
  db = 20*log(a);
  {
  if((voltage == 0) || ((0.0<db) && (db<4.0)))
 {
    digitalWrite(5,HIGH);
 }
 else if((4.0<=db) && (db<8.0))
 {
    digitalWrite(6,HIGH);
 }
 else if((8.0<=db) && (db<12.0))
```

Code 1. The part of the code for color selection

4 Cost and Schedule

4.1. Cost Analysis

Table 7. Cost Analysis

Name	Description	Manufacturer	Quantity	Cost
ICS-40720	MEMS microphone	TDK Invensense	5	19.9dollars ($ 3.98 each)
ATMEGA 328p	Microcontroller	ATMEL	1	$ 4.81
Lily Pad Rainbow LED	LED	SparkFun Electronics	1	$ 4.95
SMDJ12CA-H	Diode	Mouser	2	$6.84 ($3.42 each)
SPAC265-3W	Regulator	STMicro-electronics	1	$6.52
LM317MDT-TR	Regulator	Texas Instrument	2	$0.46 ($0.23 each)
LM3940	Regulator	Texas Instrument	2	$2.56 ($1.28)
Amplifier (LM324)	Op-Amp	Bonanza - bpelectronic's booth	1	$ 1
Arduino UNO	Microcontroller ISP	Arduino	1	$24.90
Labor Cost	Han Young Kim	$2.5 * 30 \frac{dollars}{hr} * 10 \frac{hr}{week} * 16\ \ weeks$		$ 12000
	Hyun Soo Kim	$2.5 * 30 \frac{dollars}{hr} * 10 \frac{hr}{week} * 16\ \ weeks$		$ 12000
Total				$ 24071.94

5. Conclusion

5.1 Accomplishment

Our LED representation of varying voltage level was successful as we tested with function generator. This would allow when successfully integrated with the power and system circuit, the LEDs would light up as the input sound would generate differing voltage intensity. The user can easily see the current intensity (amplitude) of the sound captured from MEMS microphone, which means that the user can see how loud the surrounding noise with the color that would light up. Moreover, the stability of power was ensured so that the optimal goal of safety was accomplished. With the power cord, the concern for battery is resolved.

5.2 Uncertainties

The biggest and most challenging task for our project was the A Weighting Filter, which is used to convert machine calibrated noise level into human-ear-friendly level. As conducting the experiments with the filter, there were large loss of the signal amplitude. This required higher gain constant at the Amplifier and this may have damaged the accurate calculation toward human ear. Manipulating the value at the amplifier was quite cautious because it may be over-calculated with small increase in the gain constant.

Another uncertainty lies on the instability of the input signal. As the microphone collects the time-varying noise signal, the behavior of signal should not be constant and stable because noise varies every single moment. Thus, the filter seemed to fluctuate much depending on the stability of the input signal. While it generated stable output with a tolerable loss, the signals coming from MEMS microphone may be fluctuating and likely to demonstrate unexpected input.

5.3 Ethical Consideration

There were a few safety concerned hazards that may occur during our project. The power supply from wall outlet was the most concerned module since there could be large surge of voltage drop, which would break the internal circuit of our device. To address this problem, we had to research deeper to resolve this issue. Thus, the chips we used contain short-circuit prevented fuse. Furthermore, we used diodes to protect the circuit elements. With the cautious choice and safety-first philosophy, we've considered IEEE code #1, 5, 6, and 9 [8]. To devise optimal hardware design, the conversation among the team was placed as a priority and all members' opinions were valued to come up with the components of the final product listed above, as IEEE code #2, 6, 8, 9, and 10 [8]. We have assisted each other on implementing and testing the output as well as providing feedbacks to each other. As our project continued, there were aspects where the supervising staff pointed out and we took those advice and feedback as valuable assets for continuing our project. To accomplish sincere and innovative approach, we had disclosed what we had designed and let the public know what we were concerning with as IEEE code 3, 4, and 7 [8] the most.

5.4 Future Work

In sound engineering and acoustics, the most crucial part is the input source quality. Thus, the area of improvement of our project is to use better input device that has higher sensitivity. The higher the sensitivity, the more accurate and specific the collected sound would be. When these input noise are to be calibrated through our design, it would be much accurate to calculate actual level of the noise. Moreover, since MEMS microphones are not readily available for public use, it would be more efficient to use the normal microphone that could collect broader noises.

6 References

[1] "Noise and Hearing Protection," in entnet.org, 2017. [Online]. Available: http://www.entnet.org/content/noise-and-hearing-protection. Accessed: Sep. 17, 2017

[2] "Hearing Loss and Headphones – Is Anyone Listening?" in osteopathic.org, N.A. [Online]. Available : http://www.osteopathic.org/osteopathic-health/about-your-health/health-conditions-library/general-health/Pages/headphone-safety.aspx. Accessed: Sep. 17, 2017

[3] "Frequency Weightings – A-Weighted, C-Weighted or Z-Weighted?" N.A. [Online]. Available: https://www.noisemeters.com/help/faq/frequency-weighting.asp. Accessed: Sep. 17, 2017

[4] "LMx24-N, LM2902-N Low-Power, Quad-Operational Amplifiers" Jan 2015. [Online]. Available: http://www.ti.com/lit/ds/symlink/lm224-n.pdf. Accessed: Oct 1, 2017

[5] "A-Weighting Filter For Audio Measurements" Aug 2013. [Online]. Available: http://sound.whsites.net/project17.htm. Accessed: Sep 30, 2017

[6] "ICS-40720 Ultra-low Noise Microphone with Differential Output" N.A. [Online]. Available: https://www.invensense.com/wp-content/uploads/2015/02/DS-000045-v1.3-ICS-40720.pdf. Accessed: Sep 30, 2017

[7] "How good Your Hearing Video reveals frequencies that you can hear or can't hear" May, 2014. [Online] Available: http://www.dailymail.co.uk/sciencetech/article-2643864/How-good-YOUR-hearing-Video-reveals-frequencies-hear.html. Accessed: Oct. 4, 2017

[8] "IEEE Code of Ethics", in ieee.org 2017. [Online]. Available: http://www.ieee.org/about/corporate/governance/p7-8.html. Accessed: Sep 19, 2017

Appendix A.

1. MEMS

Product Used: ICS-40720 – TDK Invensense

Table 8. MEMS Microphone RV Table

Requirements	Verification	Verified (Y/N)
1. Must be able to take input sound and convert to analog scale.	1. Build a basic circuit and use oscilloscope to test the functionality	1. N

2. A Weighting Filter

Product used: LM324N – Texas Instrument

Table 9. A-Weighting Filter RV Table

Requirements	Verification	Verified (Y/N)
1. Must minimize the rippling and unstable analog input for optimal performance. 2. Must have reference data that would be the checkpoint for accuracy. 3. Stabilize the incoming waveform	1. Used capacitors for each usage of resistors with different capacitance as needed. 2. a). Use PSPICE simulation with arbitrary AC voltage input. b). Build an analog circuit on breadboard before PCB implementation. c). Use oscilloscope to check the functionality of the filter. Should see multiple peaks along with narrowed waveform. 3. Implement Unity Gain mode amplifier for fully compensated internal frequency	1. Y 2. a). Y b). Y c). Y 3. Y

3. Amplifier

Product used: LM324N – Texas Instrument

Table 10. Amplifier RV Table

Requirements	Verification	Verified (Y/N)
1. Amplifier should produce a gain that would return closest to 0 dB in log scale from 1 kHz to 8 kHz with realistically implementable value of resistance. 2. Must provide a DC voltage not exceeding 32V to LM324.	1. Connect with the endpoint of A-Weighting with gain of 1.56 using non-inverting op-Amp amplifier. This is done by using 10kΩ and 6.8kΩ. 2. Connect DC 12~15V from the power supply and check with the oscilloscope.	1. Y 2. Y

4. Power Supply

Product Used: SPAC265-3W(RHOS), LM317-MDT(Texas Instrument), LM3940(Texas Instrument)

Table 11. Power Supply RV Table

Requirements	Verification	Verified (Y/N)
AC - to – DC converter 1. Use 110 AC V from outlet and change to 12 DC V to use in Voltage regulator. 2. Implement Voltage regulator circuit to convert 12 DC V to get 5V with LM317 3. 3.3 V DC voltage: LEDs, Microcontroller, MEMS Microphone	1. use SPAC265-3W to convert 110 AC V to 12 DC V. 2. Modify LM317 circuit part (resistor values to 240 Ω and 720 Ω) connected to V_{adj} to derive out 5V. 3. Use LM3940 to convert 5V to 3.3V	1. Y 2. Y 3. Y 4. Y

4. Must consider potential electrical surge or short as well as stability.	4. a). Use diodes (D1N4001) to protect the system from input short circuit and output short circuit as shown D1 and D2. b). Implement capacitors for reducing the spiking and ripple voltages for stable power source.	

5. LED pad

Product Used: Lily Pad Rainbow LED (SparkFun Electronics)

Table 12. LED pad RV Table

Requirements	Verification	Verified (Y/N)
1. LEDs should be lit up based on the code that is implemented in the microcontroller. 2. No more than 1 LED should be lit up simultaneously.	1. Set the digital output of Microcontroller to tentatively to test whether LEDs are turned on as Microcontroller outputs 1 and turned off as output of 0. 2. Test each LEDs to verify only one LED is lit up by the programming code	1. Y 2. Y

6. Microcontroller

Product Used: ATmega 328p PU (ATMEL), Arduino UNO(Arduino), 16 MHz Resonator

Table 13. Microcontroller RV Table

Requirements	Verification	Verified (Y/N)
1. Implement the circuit to program the code in the Microcontroller	1. a). Use Arduino ISP (In System Programming) to upload a code or sketch. b). Build a circuit that connects Arduino Uno board and ATmega328P.	1. Y 2. Y 3. Y

	c). Change the setting of Arduino IDE to be Arduino ISP mode, indicating that Arduino would be used to program the connected chip d). Connect 16 MHz Resonator at Pin 9 and Pin 10 (XTL) at ATmega328P to enhance the clock signal to match Clock Speed of Arduino Board	
2. Implement the Code to turn on the LED When the specific range of the decibel is calculated with the input Voltage.	2. Use if-else programming code and decibel calculation	
3. The output of the microcontroller should be digital command of turn on or turn off (1 and 0) for each digital pin connected to each of the LEDs.	3. Only one of the digital outputs would be True(1) and the others be False(0) since there could not be overlapping sound level.	

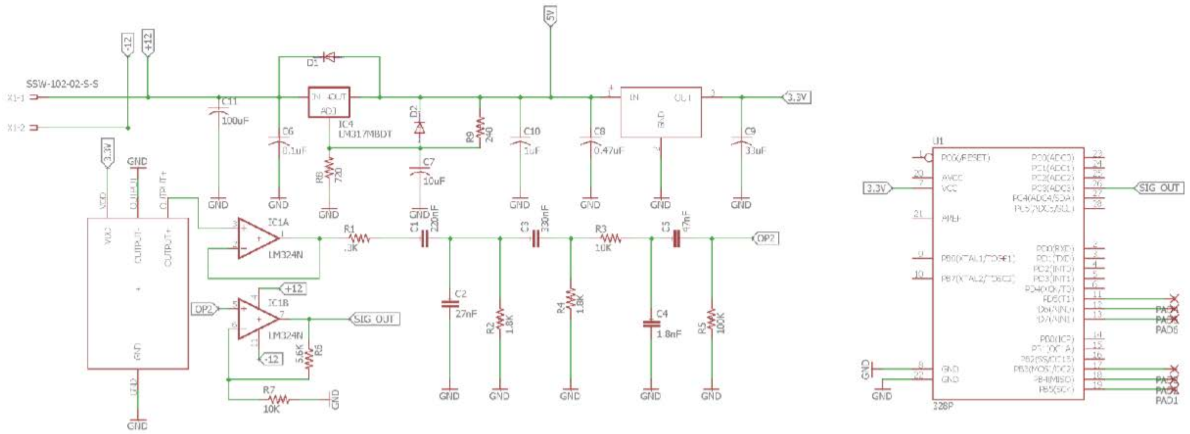

Figure 22. Overall Schematics