PHYS 406

Spring 2016

Final Report

Stereo Analog Amplifier & Real-time Sound Spatialization

Wei Ren & Decheng Fang

University of Illinois at Urbana-Champaign

CONTENTS

1 Stereo Analog Amplifier 1

1.1 Introduction 1

1.2 Circuit Design 1

1.2.1 Circuit Board 3

1.2.2 Operational Amplifier 3

1.2.3 Power Supply 4

1.2.4 Gain 4

1.2.5 Other Components 5

1.3 Complete Circuit 5

1.4 Testing 6

1.5 Conclusion and Future Improvement 6

2 Real-time Sound Spatialization 8

2.1 Introduction 8

2.2 Design 8

2.2.1 Binaural Hearing 8

2.2.2 Head Related Transfer Function (HRTF) 9

2.2.3 Field-Programmable Gate Array (FPGA) based design 9

2.3 System Architecture 10

2.3.1 Audio ADC/DAC 10

2.3.2 HRTF module 11

2.3.3 NIOS II 12

2.4 Testing 12

2.5 Conclusion and Future Improvement 13

2.6 References 14

1 STEREO ANALOG AMPLIFIER

1.1 INTRODUCTION

With our motivation in perusing better sound quality, we first aimed to build an amplifier that could drive headphones with large impedance to produce high fidelity sound. Just as the name suggests, an amplifier increases or amplifies the amplitude of low-power signals. As we found out that the analog output from computer or portable devices are not capable of fully driving SENNHEISER HD 650[1], which has an impedance of 300 Ohms (normal earbuds usually just have impedance of 32 Ohms, which are perfect for devices with low power outputs like mobile phones and laptops). The sound quality reduced since the headphone is not operating at its designed power. Thus, we decided to implement a stereo amplifier that could have gain value suitable for relatively large impedance headphones like HD 650.

1.2 CIRCUIT DESIGN

Since the existing output from most of our devices are analog, we would like to keep signals analog at amplifier level. We first did some research on headphone amplifier DIY kits online. Objective 2 DIY kit[2] was one particular example that we looked at. The kit includes all the components such as jacks, chips and circuit board. Although they do provide schematics for the amplifier, we do not have much option in modifying the circuit since the main PCB are printed and connection between components is fixed. What we can do with the kit is to do soldering and try to understand the functionality of each component. However, we are not sure how well the circuit is grounded nor the quality of

[1] Link: http://en-us.sennheiser.com/high-quality-headphones-around-ear-audio-surround-hd-650

[2] Link: https://www.jdslabs.com/products/82/objective2-diy-kit/

PCB. Therefore, we decided to build an amplifier from scratch. We think this is also a good way to understand amplifier circuit.

With the help of Professor Steven Errede, we came up with a circuit similar to the amplifier used in microphone circuit in the PHYS 406 lab. The following diagram is our initial design.

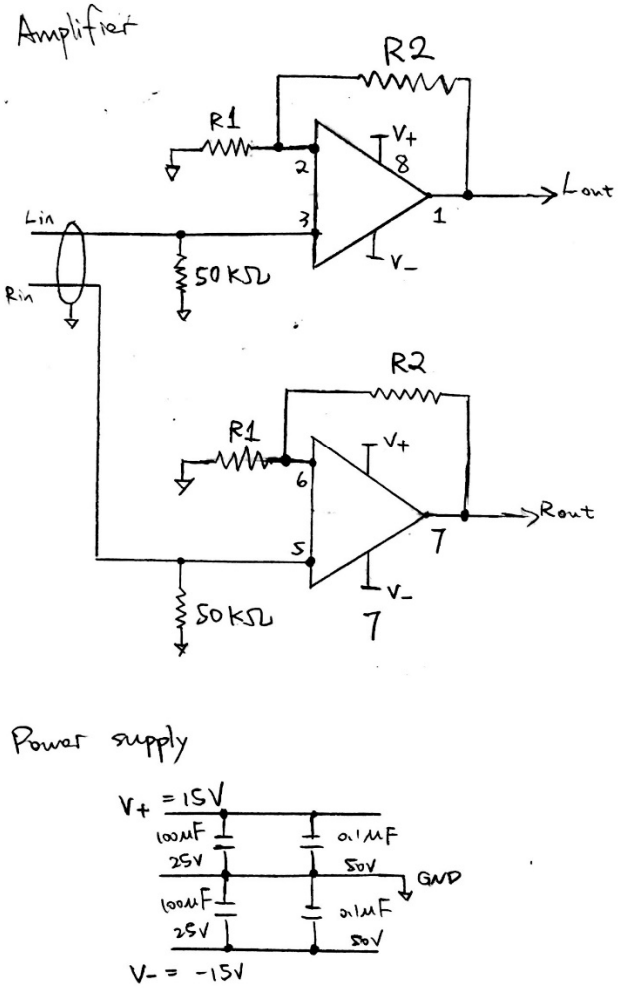

Figure 1 Circuit Diagram for Amplifier Circuit

1.2.1 Circuit Board

We started with a circuit board of 5.5 cm × 4.5 cm. We first need to ground everywhere on the board by taping the copper circuit tape. This is convenient for later soldering and it provides excellent connectivity with all the ground points. We do not have to worry about flaky connection on ground.

1.2.2 Operational Amplifier

Operational amplifier (or op-amp) is the core part of the circuit.

Professor Steven highly recommended us to use OPA 2134[3] from Texas Instrument. OPA 2134 is an integrated chip intended for high performance audio instruments. From the datasheet,[4] Total Harmonic Distortion (THD) plus Noise is under 0.0001% (or -118 dB) in the frequency range 20 Hz to 1 KHz, which means the any extra noise that the chip itself introduces is under 0.0001% of the input signal. Even in the high frequency range

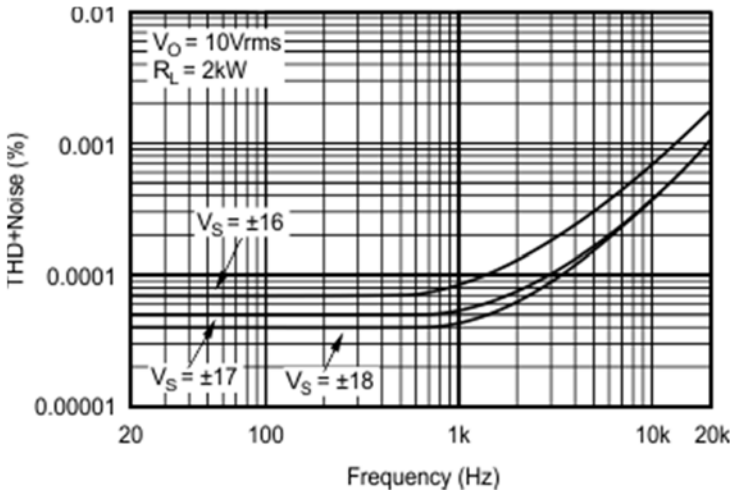

Figure 2 THD + Noise vs Frequency, from OPA 2134 datasheet

up to 20K, THD plus Noise remains below 0.001% (with voltage supply at ±18V). It is obvious that the chip outputs with low distortion and low noise, which is perfect in audio and recording circuit. To provide the stereo sound, it also incorporates two channels on a single chip.

[3] Link: http://www.ti.com/product/OPA2134
[4] Link: http://www.ti.com/product/OPA2134/datasheet

1.2.3 Power Supply

After we discussed with Professor Steven, he suggested that we could balance the output voltage at zero volt so that there is no need for extra capacitors and resisters, which otherwise might introduce noise at output. Since the optimal input voltage supply of OPA 2134 of ±15V, we must supply voltage at +15V and -15V so there is no bias on the output voltage.

1.2.4 Gain

Gain of an amplifier is how many multiples of input signal amplitude can be produced at the output signal. Gain depends on the way that op-amps are connected.

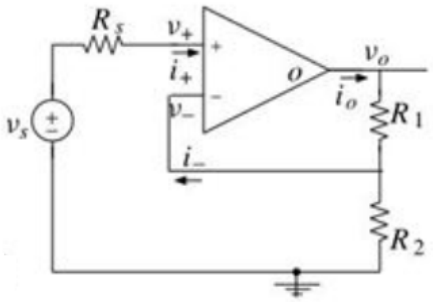

Figure 3 A non-inverting amplifier

In our circuit diagram the op-amp are non-inverting. The above Figure 3 shows a typical non-inverting amplifier circuit. Here is some basics of op-amps: if an op-amp is operating in the linear regime (so that it functions properly), then we have the following relation:

$$v_+ \approx v_- ,$$
$$i_+ \approx i_- \approx 0$$

By KVL and KCL or simply using voltage-dividing rule, we can conclude that:

$$v_- = \frac{R_2}{R_1 + R_2} * v_o$$

Given that $v_+ \approx v_-$,

$$v_0 \approx \left(1 + \frac{R_1}{R_2}\right) * v_s$$

Therefore, the gain G is the ratio between output and input voltage:

$$G \equiv \frac{v_0}{v_s} \approx 1 + \frac{R_1}{R_2}$$

In the case of our amplifier circuit, the theoretical gain is $G \approx \frac{R_1+R_2}{R_1} = 1 + \frac{R_2}{R_1}$. We chose our R1 and R2 to be 1 KΩ and 9 KΩ, respectively. This leads to a gain of 10 in total.

1.2.5 Other Components

Other components include various resisters, capacitors, wires and two audio jacks for input and output. We also added a Motorola audio potentiometer to the input signal in order to control the volume, by adjusting the input voltage.

1.3 Complete Circuit

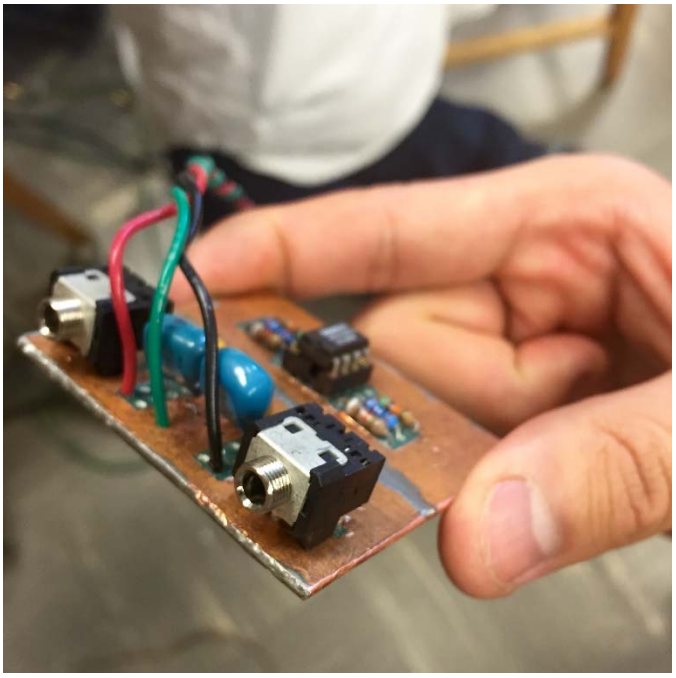

Figure 4 Complete circuit board

This picture shows what we have accomplished before we added the volume control. We first need to test if the circuit works as expected before any further improvement.

We also placed complete circuit into a metal box so that the amplifier is better shielded.

1.4 TESTING

For the testing part, we used oscilloscope and function generator. Output from function generator is 1 KHz sine wave with amplitude 100 mV. We confirmed that by directly connecting oscilloscope input to function generator. We then needed to connect the circuit output to oscilloscope after powering it on. First few try failed because of the flaky connection in cables and connection jack. The oscilloscope just showed some noise with small amplitude. After carefully examining these connections, the output was showing a sine wave with amplitude 950 mV. We concluded that 9.5 is the actual gain of the amplifier. The frequency response of the amplifier was also satisfyingly flat. To the precision of oscilloscope, the output frequency was the same as input frequency from 20 Hz even up to 50 KHz (much higher than human hearing frequency range). There is also no phase change between left and right channels on the oscilloscope waveform.

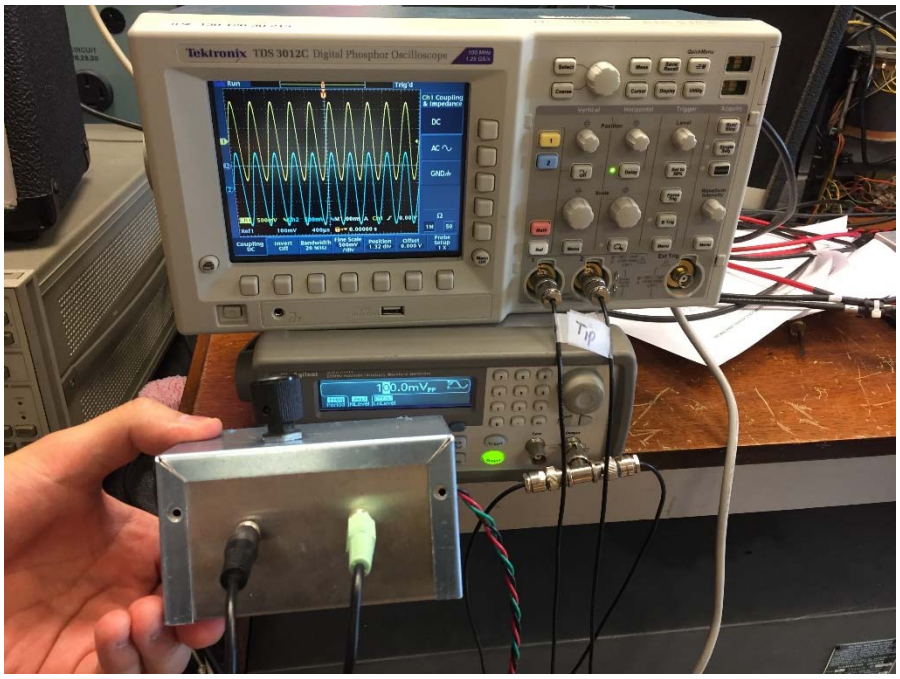

Figure 5 Final Testing with volume control functionality

1.5 CONCLUSION AND FUTURE IMPROVEMENT

As an entry-level headphone amplifier, it was intended to be built with relatively accessible and elementary materials and less complex circuit with enough power to drive

a high impedance headphone like Sennheiser HD650. After we finished the circuit as well as the case, we connected it to the power supply and conducted a hearing test via MacBook Pro 2012 mid. As the noise from the MacBook was not negligible, we connected a Fiio E10k DAC+AMP between our amplifier and the MacBook for cleaner sound. Fiio E1 0K is a very neutral DAC so it did not change the timbre of the sound too much. In addition, just from objective judgements, the amplifier did a very good job at powering HD650, which unveiled the huge potential of the HiFi level headphone, with very open and wide soundstage with precise object positioning. The amplifier also improved the treble response of the headphone, making an otherwise "veiled" sound signature brighter when underpowered. But one thing as a drawback is that the amplifier could only be powered by a 15V power supply, which is nowhere to be found at home, nor on batteries in the market. So in the future, when making an improved version, suitable power supply should be taken into serious consideration.

2 REAL-TIME SOUND SPATIALIZATION

2.1 INTRODUCTION

The other part of the project is to build a spatial sound system that is able to allow listener localize the sound source from different positions. The system outputs original input sound but listener would feel the sound coming from a specified location. We proposed to have locations among various azimuthal angles and elevation. After researching this topic online, we found many software programs, such as Pure Data[5], are able to accomplish the task. It is an open source visual programming language intended for processing sound, video, 2D/3D graphics and MIDI. The limitation of software implementation is that it requires the source to be in the digital format. Sometimes it is difficult to get a compatible digital format (usually *.WAV) or the sound output is purely analog. We hope to impose a real-time effect on the sound source.

2.2 DESIGN

2.2.1 Binaural Hearing

Humans are very good at localizing by combining variety of methods to localize sound source. As we learned from lecture and chapter 5 of the textbook, there are two

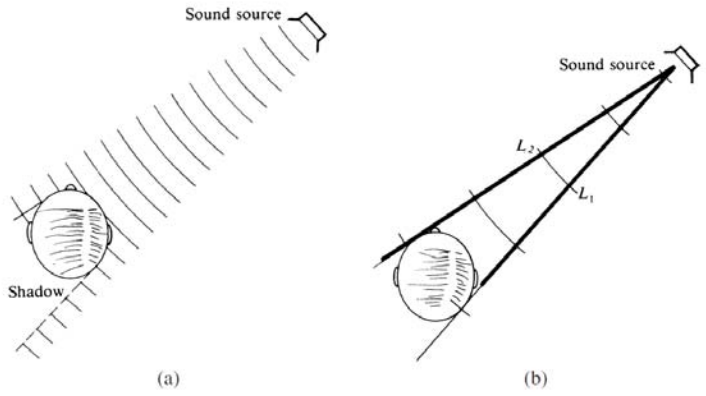

Figure 6 Chapter 5, the Science of Sound

[5] Link: https://puredata.info/

basic mechanism in human binaural hearing: intensity difference at two ears and inter-aural time difference.[i]

The (a) on the left in Figure 6 shows that sound arriving at left and right ears will have different intensity due to a "head shadowing effect" at high frequency (above 4000 Hz), since the relatively small wavelength cannot overcome the spatial difference of the head. At lower frequency (below 1000 Hz), the shadowing effect is small because of longer wavelength. Time difference dominates more. The (b) on the right of Figure 6 shows that arrival time difference is more sensitive to human ears at low frequency. The difficulty remains in between frequency range 1000 to 4000 Hz because two mechanism overlaps and the calculation is not simply liner superposition.

2.2.2 Head Related Transfer Function (HRTF)

There are also other effects determining binaural hearing. Instead of constructing a complex mathematical model, we aim to look for practical measurement data, which might work even better. We found out many software programs utilize the data from Head Related Transfer Function (HRTF) measurement done by Bill Gardner and Keith Martin at MIT Media Lab[ii]. They used a KEMAR dummy head microphone, which can represent human head and ear canal in real shape. They measure impulse response of different combination of azimuthal and elevation angles. With all the impulse response at a particular azimuthal and elevation angle, we apply convolution onto original sound source for left and right channel separately. All the responses then consists a set of transfer function.

2.2.3 Field-Programmable Gate Array (FPGA) based design

Another difficulty is to process the input source in real-time. A field-programmable gate array (or FPGA) is a specialized integrated chip with highly programmable capability. FPGA is mostly used for simulating and verifying prototype hardware design because it is intended to be re-configured and programmed. Since Wei already had some experience with signal processing and FPGA design, we decided to build a purely hardware implementation. Usually many other peripheral I/O devices (like audio CODEC and ADC/DAC) are integrated with the chip, which is another benefit using FPGA board.

Audio CODEC and ADC/DAC would convert analog input into digital signals for us to process without the need to do fast Fourier transform by ourselves in the design. We are using Altera DE2-115 FPGA board from Terasic for this project.

2.3 System Architecture

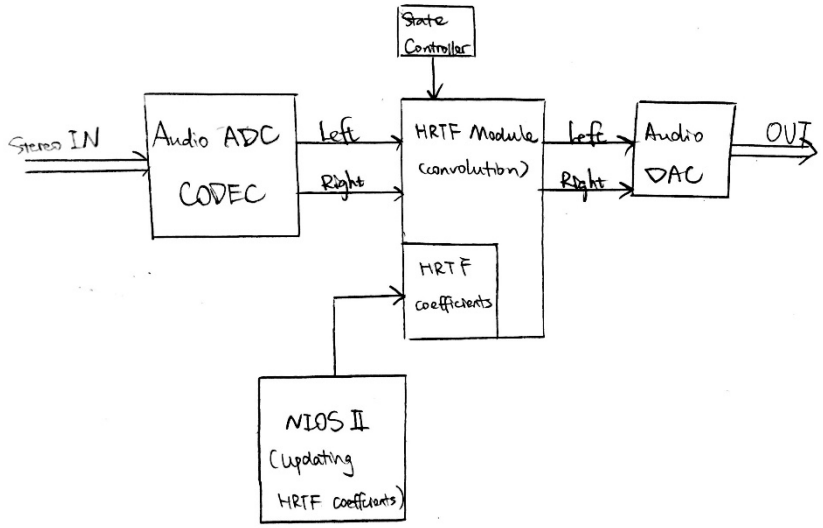

Figure 7 Block diagram

Figure 7 shows the block diagram of the hardware system. It takes analog input and converts to digital signals to perform operation on HRTF. After processing, audio DAC on FPGA board would then output analog line-out. FPGA is configured to run at 50 MHz, which is much higher than the sampling rate 44.1 KHz in normal sound recording. The oversampling ensures that there is no loss in sound quality comparing to original input.

2.3.1 Audio ADC/DAC

The DE2-115 FPGA board already incorporates a 24-bit audio CODEC and line-in/line-out. The manufacturer provides examples code on how to setup the audio ADC/DAC. We need to configure the codec to be able to decode at 44.1 KHz and use line-in instead of microphone as input source. The work here is to set up some internal register

values in WM8731[6] (audio codec used by DE2-115) so that it outputs the digital signals as expected.

2.3.2 HRTF module

This module is the core part of the system.

After DAC converting the analog input into discrete signals, we always apply HRTF coefficients on the latest 128 inputs. 128 registers is connected in series holding previous 128 input values for this purpose. We also have 128 HRTF coefficients (reduced set of complete measurement) at one particular elevation. According to spatial sound tutorial[iii] from UC Davis, output is the convolution of input source and HRTF (or effectively coefficients). The convolution here is to take values from the registers and then multiply them by corresponding HRTF coefficients. The sum of the products is the current output.

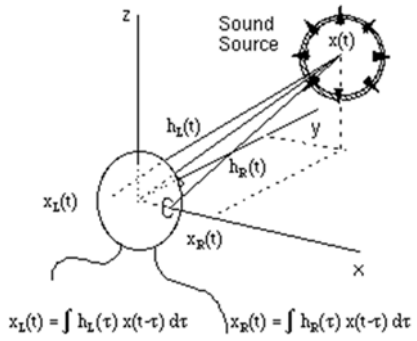

Figure 8 Spatial sound tutorial, UC Davis

We also use a state-machine controlled design to manipulate multiplication and addition easily. After convolution done a signal "done" will be sent to output module as an enable signal.

There is also another dedicated submodule storing HRTF coefficients because we have multiple sets of coefficients for different elevation angles. Storing all of them purely in hardware (register based) takes extremely long time to synthesis (which is the process to convert hardware description language into format that FPGA understands). We created a RAM like module for HRTF coefficients, which will be synthesized into memory

[6] Datasheet: http://www.rockbox.org/wiki/pub/Main/DataSheets/WM8731_8731L.pdf

blocks on FPGA board in a short amount of time. Other modules can still update the values in RAM with correct control signal.

2.3.3 NIOS II

NIOS II is a 32-bit embedded processor for Altera FPGAs. It is a software core IP for designer to include in the project to have the capability to run compiled binary such as C/C++ programs. NIOS II will be synthesized into the design using much more registers and memory blocks. The advantage is to have a general-purpose processor to handle all the lightweight tasks in C or other language. We use NIOS II CPU here for updating the HRTF coefficients because we wrote code in C for easier testing and debugging. Updating the coefficient is not a task constantly running or time-intensive. This software and hardware co-design communicate through several I/O signals. Every time the hardware needs another set of coefficients, it will have a signal asking C program to update for it. C program will then output the control signals and values to the HRTF coefficients submodule for updating.

2.4 TESTING

Most of the testing in this part of the project is to debug. We did not get the full system functioning properly until the last lecture of PHYS 406. We wrote testbench for each module. Every module seems working fine. However, we could only hear sound with glitches combining them as a whole system. Later on, we found out that the clocks for the HRTF module is not fully synchronized since we have logic operations on the clocks. In general, clock gating is not a good design habit but it would suffice for our project if taken serious care. Another problem is noise in the output when playing music with high dynamic range. We need to normalize before output so that it does not generate popping sound.

2.5 CONCLUSION AND FUTURE IMPROVEMENT

Overall, the system works as expected. Line-in takes analog sound source from laptop or smartphone. We can adjust the azimuthal angles by on-board buttons. Two of buttons can increment or decrement the azimuthal angle by 5° since that is the precision of the coefficients used. The other two buttons can increase or decrease the azimuthal angle by 90°. On-board LEDs also display current azimuthal angle in degrees. The angles are kept in the range of 0° to 360°. Four of the switches on FPGA board are used to specify elevation. The switch must only be high exclusively (with other switches kept low).

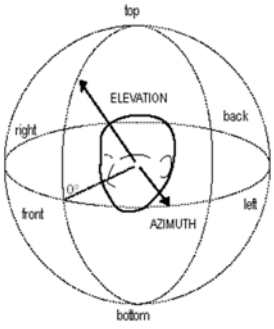

Figure 9 Azimuthal and elevation angle, from roland

We found out that human are more sensitive to azimuthal angle change. Although we do not have the time to do blind test, all listener could localize the sound source or at least differentiate which direction the source is moving in azimuthal direction. However, it is relatively difficult for listener to localize the exact elevation angle. As taught in one of the lectures confirms, it confirms that humans are more adept at horizontal plane than vertical plane when localizing sound sources. We also discovered different perception among listeners. The measurement from MIT Media Lab uses a KEMAR dummy head microphone. People have different shapes of ear canal or head, which all differ from KEMAR dummy head. Theoretically, everyone should have different HRTF coefficients. The drawback of using measurement data is that we are fixed to one particular set of coefficients. Future improvement can work on constructing a model that can adjust the coefficients based on the model and shape of one's head.

The output sound is smooth. We did not hear any delay in the system, even when NIOS II updating coefficients in C. However, the sound quality is not so great. There is still audible noise. It might be caused by not perfectly grounded. We should also look into the filter design in the ADC/DAC.

PHYS 406 projects gives us a better understanding in acoustic physics. We get many insights during the semester. We would like to thank for Professor Steven Errede and TA John Whiteman for their patience and help on the project.

2.6 REFERENCES

[i] Rossing, Thomas D., F. Richard. Moore, and Paul A. Wheeler. The Science of Sound. San Francisco: Addision Wesley, 2002. Print.

ii Gardner, Bill, and Keith Martin. "HRFT Measurements of a KEMAR Dummy-head Microphone." (1994).

[iii] "CIPIC International Laboratory." CIPIC International Laboratory. UC Davis, n.d. Web. 12 May 2016.