

Design Document

Sound Effect Devices for Musicians: Synthesizer

Presented by:

Francisco Alegria, Blake Beyer, Tim Day, Eric Fischer, & Travis Gillham

Delivered to:

Dr. Degang Chen, & Dr. Randy Geiger

Frontal materials

Table of Contents

Introductory materials	5
Acknowledgement	5
Problem statement	5
Operating environment	5
Intended user(s) and intended use(s)	5
Assumptions and limitations	5
Expected end-product and other deliverables	6
Specifications and Analysis	7
Proposed Design	7
Oscillator	8
Noise	9
Mixer	10
Filter	10
Output Amplifier	11
Design Analysis.....	11
Oscillator	11
Filter	12
Noise	13
Output Amplifier	13
Mixer	14
User Interface	14
Testing and Instrumentation	14
Interface Specifications.....	14
Hardware / software:.....	15
Modeling, Simulation, & Functional Testing:.....	16
Oscillator	16
Filter	16
Noise	16
Output Amplifier	17
Mixer	17

User Interface	18
Non-Functional Testing.....	18
Performance	18
Security	18
Usability	18
Compatibility.....	18
Results.....	18
Oscillator	18
Filter	19
Noise	21
Output Amplifier	22
Mixer	23
User Interface	23
Issues and Challenges	24
Closure Materials	24
Resources.....	26

List of figures

Figure 1 - Block Diagram

Figure 2 - Schematic for Oscillator circuit

Figure 3 – Schematics for Oscillator outputs & Triangle to Sine Converter

Figure 4 - Schematic for Noise circuit

Figure 5 - Schematic for Mixer circuit

Figure 6 - Schematic for Low pass filter circuit

Figure 7 - Schematic for High pass filter circuit

Figure 8 - Schematic for Output amplifier circuit

Figure 9 - Low pass filter low-end cutoff frequency plot

Figure 10 - Low pass filter high-end cutoff frequency plot

Figure 11 – Filter Data

Figure 12 – Time Distribution

Figure 14 – Power Density Function

Figure 15 - Main display for software user interface, iPad version.

List of equations

Equation 1 - Low pass filter transfer function

Equation 2 - High pass filter transfer function

Equation 3 - Voltage from thermal excitation

Equation 4 - Power from BJT brake down

Equation 5 - Gain formula for audio amplifier

List of tables

Table 1 - Thermal noise source variables

Table 2 - Table for testing equipment

Table 3 - Concepts for Noise module

Table 4 – Concepts for Mixer module

Table 5 - Oscillator results table

Introductory materials

Acknowledgement

The Sound Effect Devices for Musicians team would like to kindly thank both Dr. Chen and Dr. Geiger for advising us as well as supporting us with knowledge through the duration of this project.

Problem statement

In today's world, musicians can be put in a difficult spot when trying to find new equipment that will help them excel in the music world while staying under a reasonable budget. Musical devices, such as synthesizers, can cost anywhere from \$500 - \$10,000, with this most musician do not have the cash to afford upgrading their equipment to compete in the competitive industry of music.

Our goal is to create a synthesizer for musicians is easy to use, modular, and affordable. Our target is to create the synthesizer for under \$250. Musicians thrive on having the latest and greatest devices to create music, so we plan to create our synthesizer with our own unique spin on it by using a tablet with a compatible app that has a user-friendly interface as the keyboard instead of the traditional keyboard layout. This device we are creating will not only look like a state-of-the-art synthesizer but will also sound like one too as we plan to give the musicians ample freedom to make music their way with our device being modular.

Operating environment

The operating environment for the synthesizer, in a perfect world, would be in a controlled environment where the users can make sure that the product is always in the best shape. Naturally, that will not be the case always. Operating environments are really determined by the users. If the user would like to use the product in a controlled environment that is fine and if they would like to take, it to live shows that is fine as well. As you can imagine, that means the product needs to be able to withstand normal everyday use both inside and outside a controlled environment. It will need to withstand rough conditions, even if it might never be exposed to these conditions. Simply putting it, our product will be built to last and perform at its highest capabilities.

Intended user(s) and intended use(s)

Intended users of the synthesizer include everyone that has an interest in making music or making noise for entertainment. All musicians, from amateurs to professionals, are the target audience for this product. We aim at making this not only a highly sophisticated tool, but also one that is easy and fun to use.

The intended use is up to the users. Our hope is this tool is being used to help professionals create music but also people that are just having fun. This product in the general realm is intended for musicians who want to make high-quality music on a lower end budget.

Assumptions and limitations

Assumptions:

- Users can read and understand English - to understand brief introduction
- Users will have access to a wall outlet
- All components will work inside the product

- Product will be taken care of by owners
- Users will have access to a Windows or Mac computer with *Max* software, and an iPad with the *Mira* application if they want to use the external UI features.

Limitations:

- People without musical knowledge will have a learning curve
- Group members have conflicting schedules so finding an ample amount of time each week to work is difficult

Expected end-product and other deliverables

The final deliverable for this project will aim at handing over a sophisticated synthesizer that is not only easy to use but is also a very powerful tool for musicians to have. The synthesizer will cost under \$250 to be made, but that will not affect the functionality of the synthesizer. The tool will be fully functional and include eight modules that give musicians the capability to make a wide range of music. With this project, we do have specifications that it must meet, as we want it to be user friendly with the user interface, we also are setting a requirement that we want to make sure we can both hook this up to a speaker and plug in headphones into those speakers.

The final deliverable for this project will also include a brief instruction, specifying modules that are included in the synthesizer and how they work. The instruction manual will target users that already have a musical background, so they can quickly look through the manual and get into making music as quickly as possible.

Outside of these main goals, we do have some stretch goals. These include adding options that will allow users to connect our synthesizer to other equipment such as a keyboard to operate. The reasoning for this is for some musicians they may prefer the physical layout of a keyboard compared to a touch screen and so giving them this capability would really help users feel more comfortable with using it.

We estimate completion and delivery of this project to be May of 2019, the end of the spring 2019 semester.

Specifications and Analysis

Proposed Design

We have decided to create a synthesis device that implements the basic sound generation functionality of conventional synthesizers, as well as some additional effect processing features and external control features. For the basic functionality we need eight separate modules; Voltage-controlled audio frequency oscillators, voltage-controlled low frequency oscillators, a multichannel mixer, voltage-controlled filters, envelope generators, voltage-controlled amplifiers, a user interface that generates control voltages, and a power delivery module. Figure 1 shows a block diagram showing how the basic modules will be connected inside of the device. With these 8 modules, the user can design new sounds by setting the available parameters and then playing those sounds with the keyboard. All the modules have their inputs and outputs available as control voltage sources and destinations on the hardware user interface. With these inputs and outputs, the user can reconfigure the internal signal paths. The basic modules have controls on the hardware user interface to set all the sound influencing parameters. The controls all go through the keyboard module's microcontroller, which acts as the digital brain for the entire device. By having the controls go through the microcontroller, the user has the ability save and recall presets. The microcontroller also allows for a wired communication path via USB with an external PC machine. The PC machine can run a mux patch that communicates with an iPad through Wi-Fi. With the iPad, the user can have a wireless user interface that mimics the hardware user interface with additional touch and motion controls.

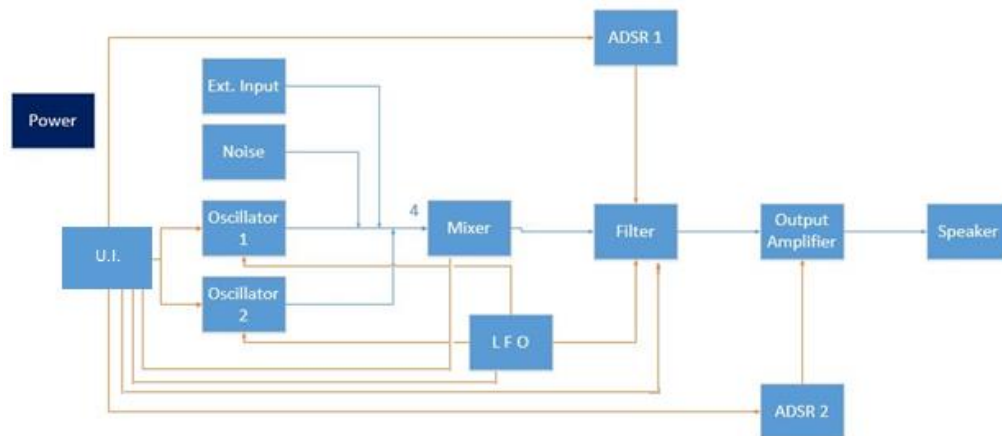


Figure 1

Oscillator

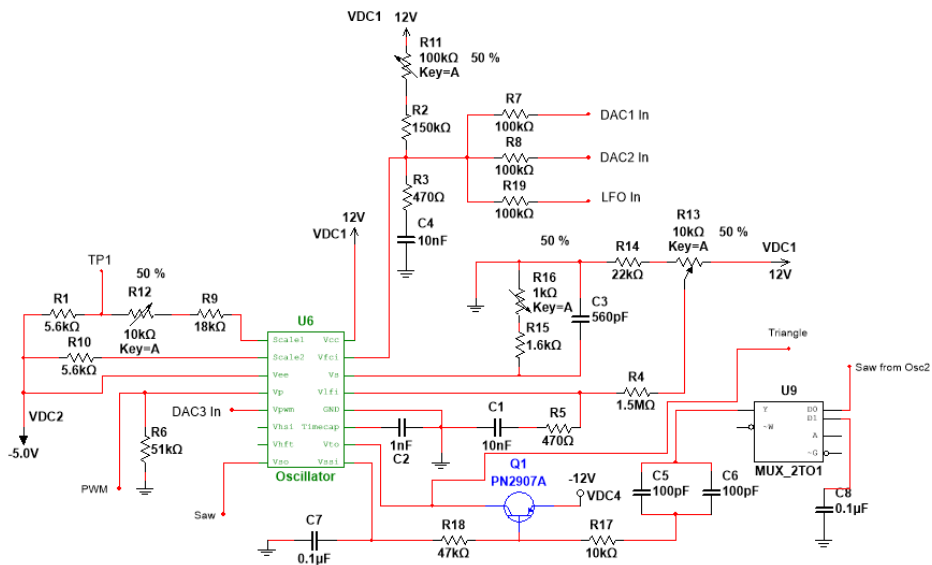
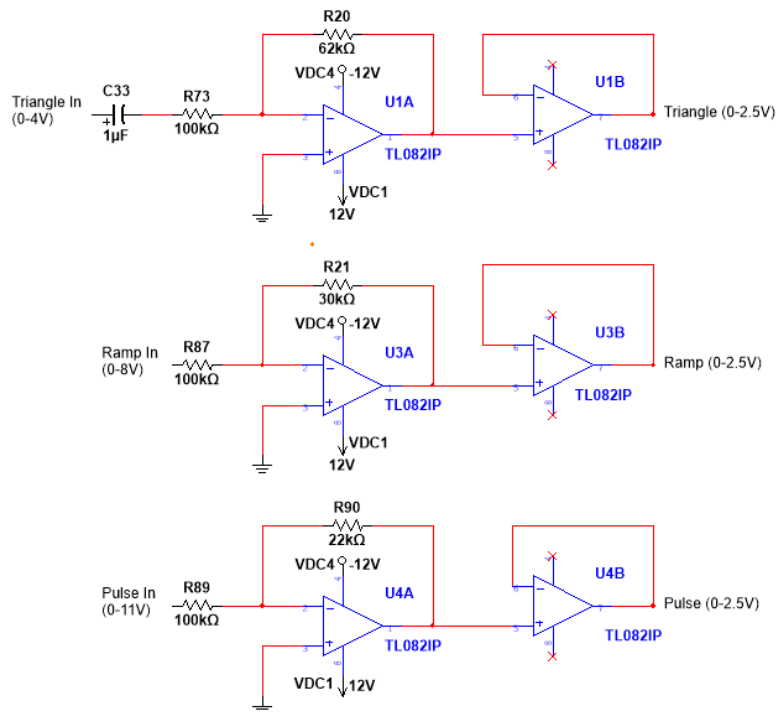


Figure 2



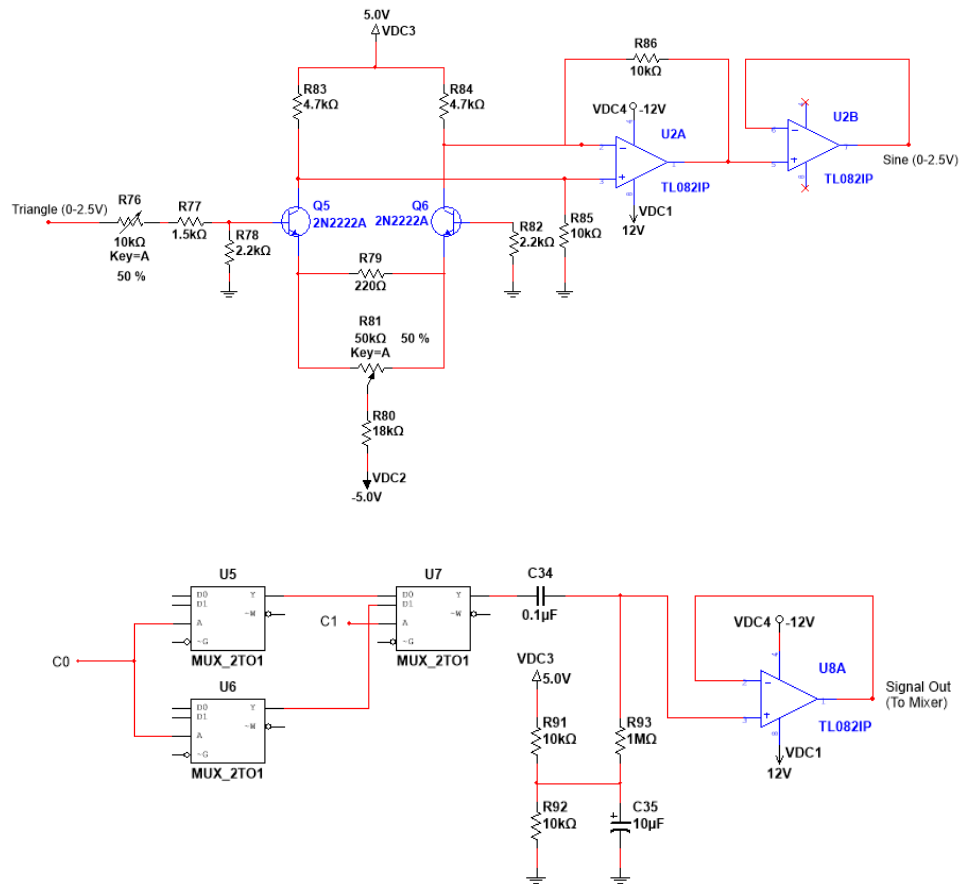


Figure 3

Noise

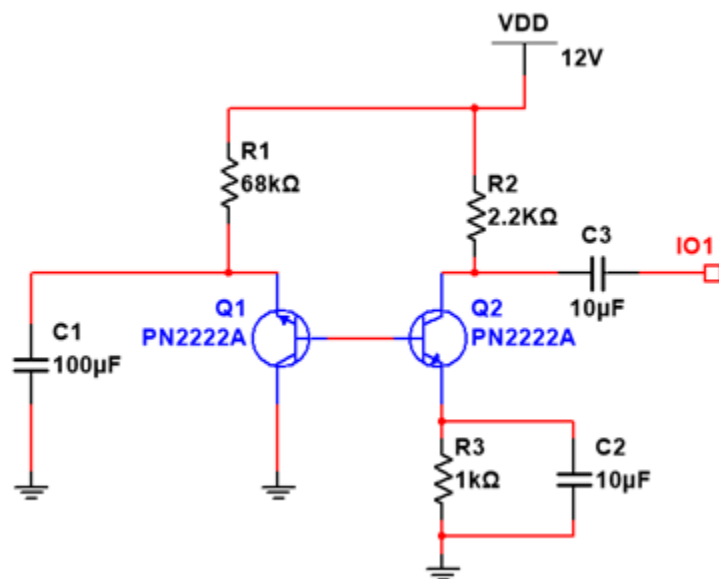


Figure 4

Mixer

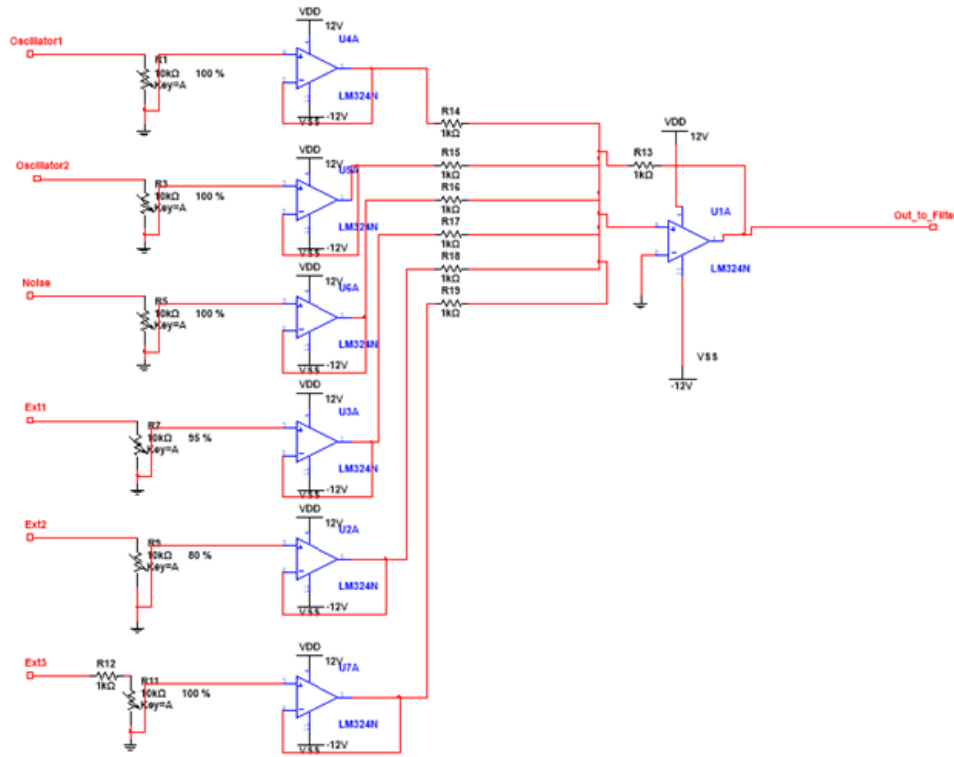


Figure 5

Filter

Below is a circuit schematic of the second order low pass filter. The input comes from the Mixer. The output goes to the Output Amplifier.

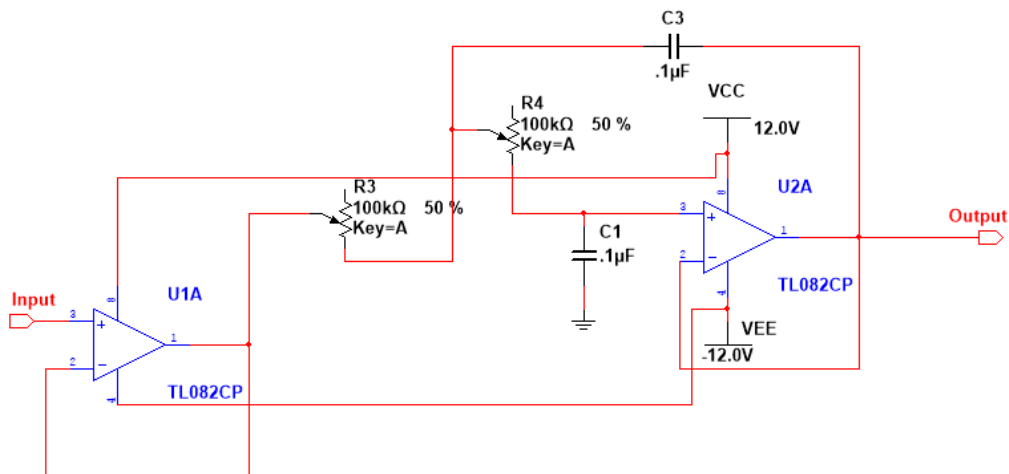


Figure 6

Below is a circuit schematic of the second order high pass filter. The input comes from the Mixer. The output goes to the Output Amplifier.

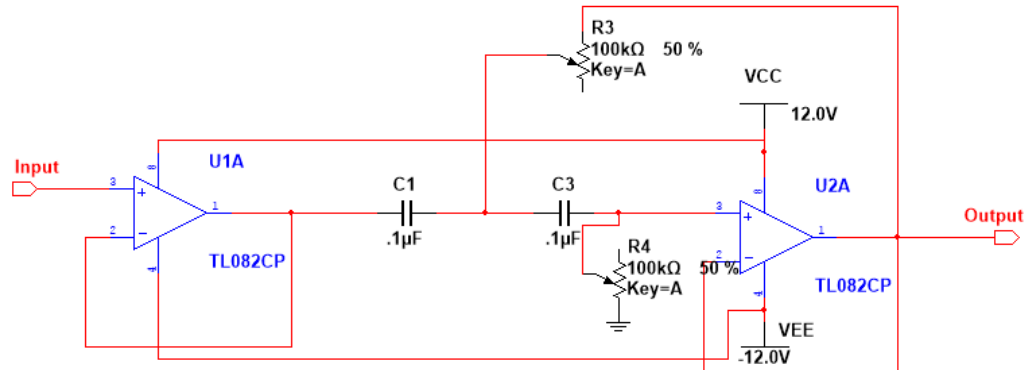


Figure 7

Output Amplifier

Below is the circuit we are using for the output amplifier.

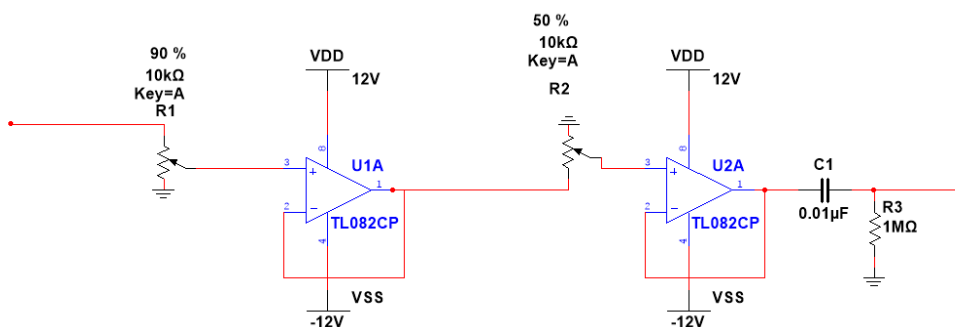


Figure 8

Design Analysis

This segment will be broken down by each module and will explain the decisions that have been made to reach the final product.

Oscillator

The oscillators are designed around the CEM 3340 voltage-controlled oscillator. A primary reason this chip was selected was that it very easily handles the 1 volt per octave standard. The chip also provides four output waveforms, triangle, ramp, pulse, and square. Given a 12-volt supply, the output amplitudes for the triangle, ramp, and pulse/square are 4V, 8V, and 10.5V respectively. The triangle and ramp outputs are fed into amplifier stages to center them on 0 volts. This is done because audio signals are centered on 0 volts and the CEM outputs waveforms that are above 0V. All output waveforms are then

attenuated to line-level, about 2.5 V_{pp}. These amplifiers will be TL082s. It has low total harmonic distortion. The CEM also has a temperature compensator circuit and has been designed in accordance with the datasheet. Sync will also be implemented in accordance with the datasheet.

We also desire a sine generator. We used a variant of the triangle to sine converter found in Microelectronic Circuits by Sedra and Smith. The sine wave is created by passing the triangle wave through a differential pair of BJTs with 1 side grounded. The BJTs are biased in such a way that they go into non-linear mode towards the peaks of the triangle. This creates a wave shape that is not exactly sine but very similar. This method provides a lower THD than other methods such as diode wave shaping circuits. However, this method does not guarantee a perfectly symmetrical sine wave. This will need to be fine-tuned using trim pots.

The oscillator module includes two voice oscillators, which are identical. A digital switch will determine the output waveform that is passed on to the mixer.

Filter

The design for the low pass and high pass filters are second order Sallen Key filters. The reason to do the low pass and high pass filter this way is that this will allow us to reach our specification of 12/24 dB per octave. The reason for a Sallen Key design is to achieve maximum flatness on the top of the frequency response, as well as Sallen Key filters are commonly used to design filter circuits for sound/music applications. This design was chosen for the high pass and low pass filters because adding these circuits together in parallel and series will create a band pass and notch filter without having four separate circuits. We want to have the cutoff frequency to be adjustable between ~20 Hz and ~20kHz, which is the audio range. These values are the frequencies of each octave from the first to the tenth octave. Having the cutoff frequency adjustable through this range allows the user to determine what frequencies they want to let through to create the sound they desire.

The low pass filter was designed by having two sets of resistor-capacitor in parallel connected to the positive pin on the op amp with a unity gain going from the negative pin to the output. The resistors will be digital potentiometers to make the cutoff frequency adjustable through our digital user interface. The op amp chosen is a TL082. The reason being is this op amp's application is audio, in which it is very low harmonic distortion. Which is what we desire in order to achieve perfect harmonics, or as close to perfect as we can get. The function for a low pass filter should have a zero in the numerator that is at infinity, meaning the numerator should be a constant. Q in the transfer function equation is the pole quality factor or resonance, which determines the sharpness of the features in the frequency response. For example, high Q means the cutoff is very sharp and is not smooth and low Q returns smooth cutoff. For our application, a Q of ~ 1 or slightly less is ideal. The high pass filter was designed in a similar way. The transfer function for high pass should have a numerator that is equal to the gain*s² which is just s² because the gain is a unity gain. The transfer function for the low pass is equation 1, and the high pass is equation 2 when the resistor values are 91k Ohms to achieve a cutoff frequency of 18 Hz, both are shown below.

$\frac{12075.83625166}{s^2 + 219.78021978022s + 12075.83625166}$	$\frac{s^2}{s^2 + 219.78021978022s + 12075.83625166}$
<i>Equation 1</i>	<i>Equation 2</i>

Noise

To create a noise generator the first approach was to see what noise should be amplified. There are two major types of noise that we chose to investigate thermal, and avalanche. The method that thermal noise can be collected is amplifying it off a resistor. The thermal or Johnson noise comes from the random motion of charge from thermal excitation [1]. This noise source can be described by this equation:

$$V_{th}^2 = 4kTR\Delta f$$

Equation 3

k	Boltzmann Constant
T	Absolute Temperature
4kT	$1.61 \cdot 10^{-20} \text{V}\cdot\text{C}$ at room temperature

Table 1

We turned against this idea because of the amplification and the number of DC biases needed to get the desired white noise.

Instead, we selected to use avalanche noise. Avalanche noise originates from the breakdown of the reversed bias junction of a BJT. This noise originates from carriers in the junction gaining energy from the high electric field then colliding with the crystal lattice [1]. This equation for this noise is characterized by the equation below:

$$S_{av} = \frac{(2qI)}{(2\pi f\tau)^2}$$

Equation 4

Where τ represents the time constant of the resistor and capacitor across the BJT. This equation was used to determine the power from the resistors and the capacitors.

To get the brake down voltage a 2222 npn transistor 12 volts was placed above the emitter to get the brake down according to the datasheet the brake down voltage from the emitter to base is 7 volts. It was found that this breakdown did occur. From here, a common emitter amplifier and an operational amplifier along with DC biases provided the output waveform of the noise with an RMS value in between 500mV and 1V. From here, it was then continuously tested to get the desired output.

Output Amplifier

The output amplifier circuit consists of two portions, the first portion is for volume control and the second is for ADSR. Volume control is just limiting what can go to the external speakers which results in a louder or softer sound depending on what the user sets the volume at. ADSR, which stands for "Attack-Decay-Sustain-Release" is a more complex portion compared to the volume control. This section adjusts how the signal initial starts when a key is pressed depending on how sliders are set on the user interface. With this, the output amplifier is a simple circuit designed with an TL082 Wide Bandwidth Dual JFET Input Operational Amplifier and two digital potentiometers of DS1803 Addressable Dual Digital Potentiometer and the AD5144ABRUZ100 Quad Channel, 128-/256-Position, I2C/SPI, Nonvolatile Digital Potentiometer. The reasons the components were chose is since our project works heavily with audio waveforms. With the TL082 this op-amp is known for the practicality with audio related projects. The reason for the two digital potentiometers is that for the ADSR portion of the circuit it only needs a

linear taper digital potentiometer. For the volume control portion of the circuit, it requires a logarithmic taper digital potentiometer. After a signal, goes through these two portions of it will then have to filter to have the DC blocked and then it will be sent to the external speakers.

Mixer

The idea of the mixer is to use a summing amplifier to combine the two oscillators, noise, and three external inputs. All the inputs come it at 2.5Vpp. Once they are summed, the output will not be higher than 2.5Vpp. This means that the values need to be attenuated prior to the summing amplifier. This is where the user will have control and be able to select the percent of each of the input that they expect to receive in the output. An algorithm has been written that will set the values for the digital potentiometers that will adjust the percentage of the value the user wants of an input. The algorithm takes in which inputs are not set to zero and adjusts the percentage to each of the others accordingly making it where the output cannot exceed 2.5Vpp.

User Interface

The user interface module will be used to control all of the synthesis modules as one complete system. In order to control the analog synthesis modules via a digital interface we will need to implement digitally controlled resistances, analog voltages, and analog signal paths. For this we will use digital potentiometers, analog multiplexers/demultiplexers, digitally controlled analog switches, a Wi-Fi transceiver to make the UI wireless, and a microcontroller to set the digital devices based on input controls received from the user. Our approach will be to create a GUI that can be used through the TouchOSC application for mobile devices; iPads, iPhones, and Android phones. The interface will use the Open Sound Control protocol to communicate via the Wi-Fi transceiver with the onboard microcontroller. The microcontroller will take the data received from our interface and process it to set the internal control values. The microcontroller will also send data back to the interface to provide the user with feedback. This type of interface will allow modularity to be fully internal and digitally controlled. A wireless interface will also allow the user to play the device from anywhere in the room, within the Wi-Fi range. The UI application also allows the user to instantly switch between multiple interfaces that are preloaded, more modularity and user customization. Using Wi-Fi instead of Bluetooth as the communication method will also help reduce latency between the user input and the synthesis device to improve response time.

Testing and Instrumentation

Interface Specifications

The interface that we will be using to confirm that the device works is by using the lab equipment at Coover Hall at Iowa State University. Below is a list of the equipment that will be used in the lab.

Equipment	Purpose
Function Generator	This device will allow us to test the voltage source for a sine, pulse, and ramp. The major benefit of this device is being able to select the amplitude and frequency of the wave.

Oscilloscope	The device measures the waves throughout the circuit. It also provides a good measurement system for voltage peak-to-peak, amplitude, and RMS of an AC wave.
Multimeter	This device measures the resistance, voltage, and current throughout the circuit.
DC source	This will provide a consistent voltage source that we will use for our Vdd and Vss of the circuit.
Lab View	This will allow us to automate the testing of the device if needed.

Table 2

The user will interface with the device by using an iPad or a smart phone and communicate with it wirelessly. The user interface will consist of buttons and sliders depending on the module it is controlling. For example, the control for the oscillator will be selecting what the output waveform is while the control for the mixer will be a slider that will adjust the percent output of the value that is being adjusted. The wireless connection will modify the digital potentiometers. When testing this in lab the outputs will be monitored on a multimeter and oscillator.

Hardware / software:

To model out the schematics of each module the program that we decided to use is National Instruments Multisim. Multisim is a SPICE simulator and circuit design software. This allows the user to be able to run analog, digital, and power electronic simulations and be able to capture the data. This program is useful for our needs because it is a clear way to represent a simulation of our modules since it has the characterization of many parts. An easy to use software allows us to make a circuit diagram for our schematics so that we can present them to each other and our advisors.

One of the biggest benefits of NI Multisim is that it contains the footprints for the components which makes using the PCB designer National Instruments Ultiboard. This program allows us to design the PCB for each module. This program is renowned for being an excellent way to do PCB design and routing which will help us optimize the dimensions of the boards that we are fabricating.

Prior to creating the PCB's, we will solder the device onto a perf board. This will allow us to confirm that the device will work with the components soldered down. The testing of this device will be confirmed using Coover's laboratory equipment.

To have our own customized PCB design we are using OSHpark. This company has a great user interface when submitting a PCB design to be printed on a PCB. They have a cost that we support of 5- 10 dollars per square inch. Having them fabricate our PCB's is something that we need in our final product.

Once the PCB has been fabricated it will be tested using Coover's lab equipment. This will show final proof that the module will work in our final product.

We will also be using the Arduino IDE to program our microcontroller. This software will simplify a lot of the chunky code with the use of the integrated libraries and widely available open source documentation.

To develop the user interface, we will be using programs developed by Hexler.net. We will use the touchosc-editor software for graphic design of the UI. Then we will use TouchOSC mobile application to

display and run the interface on any device that has our UI loaded and is on the same Wi-Fi network. Both of these programs will allow the user to use any operating system with our device.

Modeling, Simulation, & Functional Testing:

Oscillator

Since we are using a unique chip, the oscillators cannot be simulated on Multisim. The only tests that can be done are in lab. The circuit will be powered as shown in the schematic. To test voltage to frequency accuracy, we will apply control voltages to the exponential input and measure the frequency of the triangle wave output. There are multiple trim pots in the circuit that will be used to fine-tune the circuit. Once an acceptable accuracy has been achieved (actual frequency within 5 cents of desired frequency), the trim pots will be replaced with trim resistors.

The triangle to sine converter can be simulated on Multisim. The input source is an ideal triangle wave and a scope is placed on the output of the op-amp. The two waveforms will be compared. In simulation, there are trim pots on the input and on the emitter of the BJTs. Once acceptable THD (<5%) and symmetry is obtained, the trim pots will be replaced by resistors. This circuit will then be implemented on a breadboard using the resistances found in simulation. Trim resistors will be added to achieve the acceptable THD and symmetry. The schematic will then be updated if these trim resistors are added.

Filter

The first step was using an online Sallen Key filter design tool to see what a second order low pass [7] and high pass filter [8] would look like. This was to attempt to see what controls the cutoff frequency and what controls the resonance by choosing different capacitor and resistor values. After determining how cutoff frequency changes, a schematic was made for both high and low pass filter in Multisim which is shown in Figure 7 and Figure 8. These were simulated to check if cutoff frequencies are in the audio range 20Hz – 20kHz. After simulating the circuits in Multisim and confirming they work, the circuits were built on breadboards. This will allow for hard testing to be done using Lab View or Signal Express and function generator and oscilloscope to see if there are any discrepancies between simulation results and hard testing results. The next step is soldering the circuit on to a perf board. This is just in case our PCB designs don't work together or if they don't come out how we expected them to.

Noise

Since the noise is an occurrence that happens when there is a breakdown between the emitter and the base of the BJT this was not able to be tested on Multisim. The concept of the circuit design came from equation 4 that shows the intensity of the noise is relative to the resistor and capacitors that are used within the system. To make sure the module worked effectively there are a few concepts that need to be validated which can be seen in the table below.

Test Case	Justification
Voltage Output is 1Vpp	This will be tested using an oscilloscope. The concept of what the peak to peak voltage is of noise is mostly determined from the V_{rms} of the noise. Since the noise should be randomly distributed the oscillators output for the noise should be the average that it sees.
There is zero mean(random)	There will be a time domain view that can be seen from the noises output on if it follows any visible pattern or does the output is

	consistently jumping around and does not maintain a repeatable value. This can be verified on the oscillator.
Audible Output	This is one of the most important segments of the testing and that is does the noise produce a sound in the audible hearing range that sounds what is expected. The output of the noise will be connected to the speakers in Coover's lab and this will provide a way to hear the output of the noise module.
Consistent at all frequencies	This will be tested by saving an output from the oscillator and then putting the code into MATLAB and do a Fourier transform on the noise to see the spectrum response.

Table 3

Output Amplifier

Testing for the output amplifier circuit has been completed on a simulator using NI Multisim and in the lab. When the circuit is in NI Multisim you can use a sine wave to be the input of the circuit with a 2.5 Vpp max and record what is on the output. What the output results should look like is the following. Coming out of the volume control portion it should be attenuating the input signal a level that is dependent on what the digital potentiometer is set at. Coming out to of the ADSR portion will be a when the signal is started it will affect how fast the signal reaches its max (what comes out of the volume control portion) and if it sustains there and for how long. This section is difficult to simulate as the digital potentiometer will be programmed to change its values at certain intervals so trying to simulate that for over 200 tap position with the timing is almost impossible. Following this once the circuit has been simulated it would next be put onto a perf board where it would be solder onto.

Mixer

The mixer was first tested on a simulation to confirm that it would work. The code was tested on Matlab to provide confirmation that it would work. Then it was transferred to the Arduino. Below is a table of what is expected from the tests.

Test Case	Justification
User can input percent change	The code needs to be able to take in inputs and be able to adjust the digital potentiometers to the correct resistance. Each digital-potentiometer will be measured along with its percent change from the expected amount.
Voltage output matches percent change	The voltage going through the mixer needs to be what is expected from the percent change. If there is any variance from what was expected it will be recorded to see if it affects the desired output.
Power decrease	For every input that is not set to zero the input voltage needs to be divided by this amount. This will confirm the output will never exceed 2.5Vpp.
No signal distortion	When going through the digital potentiometers and the LM324 amplifier there should be no distortion from the input. If there is a distortion it will be measured, and the schematic will must be adjusted. This process will

	repeat from the first step to this step for any change that was seen needed in the schematic.
--	-----------------------------------------------------------------------------------------------

Table 4

User Interface

Since the user interface module interacts with every other module, it will be a bit difficult to test in the initial phases. The components that need to be tested are all the software buttons, labels, knobs, X-Y pads, and sliders. Every control should send an OSC message to the micro controller via Wi-Fi with the correct label and value. This can be started without all of the other individual modules. Next, the microcontroller has to interpret those signals and set the correct parameters on the hardware digital controls that are in all of the modules. Once that is confirmed, testing for the preset save function is needed. This can be done by saving a preset, turning the device off then back on, and checking that all of the parameters are correctly readjusted after power on. Another key part that needs testing is the algorithm that will handle the monophonic and polyphonic note routing. This can also be started without the other modules, but the complete system will be needed for final functionality confirmation.

Non-Functional Testing

Performance

The integrity of the device will be provided once all the working modules are connected. This will have the complete performance of the device by integrating the user interface on an iPad to adjusting the digital potentiometers on the device to modify the output sound. This performance of this will be evaluated on if each module provides the output that is desired from it. The individual testing of each of them will show that it will work when combined in the final system.

Security

There are no measures to secure that the user's connection over Wi-Fi is secured from outside sources or if outside sources modify the device. The connection over Wi-Fi uses a predefined application on the iPad and ensuring security of its data is not tested by this product.

Usability

The user interface of the device will have a screen that modifies each one of the modules on the device. This usability will be tested by a sample group to make sure that the usability of the user interface is accepted by the group.

Compatibility

Since this application is available on many devices. A test will be conducted to make sure that the same behavior is recognized on an iPad, an iPhone, and an Android Phone. The test will be to modify the output for each module and see if the behaviors are identical.

Results

Oscillator

Below are the results of the oscillator. It was specifically tuned at C1 and C6.

CV and Note	Theoretical Frequency (Hz)	Actual Frequency (Hz)	Expected Frequency (Hz)
0 V (C1)	32.70	32.533 (tuned)	32.533
1 V (C2)	65.41	65.155	65.066
2 V (C3)	130.81	129.65	130.132
3 V (C4)	261.63	260.17	260.264
4 V (C5)	523.25	522.70	520.528
5 V (C6)	1046.50	1045.0 (tuned)	1041.056
6 V (C7)	2093.00	2080.6	2082.112

Table 5 Frequencies obtained and the expected frequency at each CV. All frequencies are in Hertz.

Accuracy of the exponential converter started to decrease beyond this range. This is due to the internal comparator not being able to switch fast enough. A high frequency tracking feedback loop will therefore need to be implemented. It will take the error signal and feed it back to the control voltage input to correct the error.

Filter

Below is the simulation result from Multisim of the Low pass circuit for the low-end cutoff frequency. The expected low cutoff frequency was 20 Hz. As seen below, the simulated low cutoff is about 25 Hz. This is a good enough tolerance for our application, as 20-25 Hz is not a very significant difference.

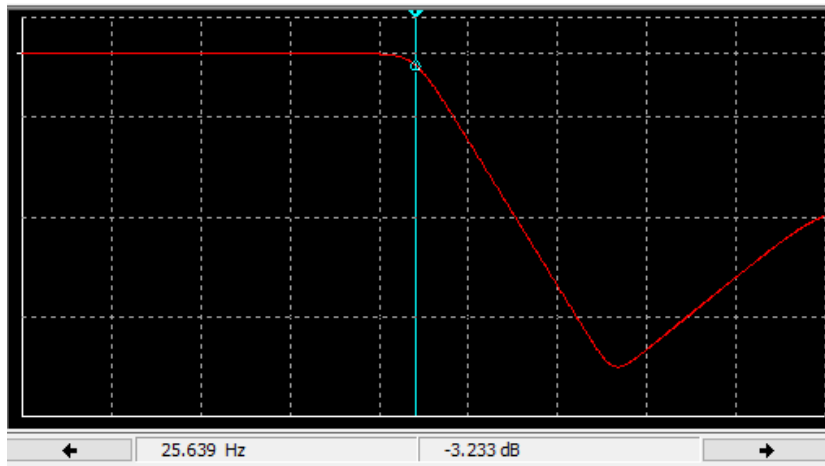


Figure 9

Below is the simulation result from Multisim of the Low pass circuit for the high-end cutoff frequency. The expected high cutoff frequency was 20k Hz. As seen below, the simulated high cutoff is slightly over 20k Hz, which is perfectly okay for our application because humans start not hearing frequencies around 18k-19.5k Hz. Which is why we initially made a specification goal of obtaining a high cutoff frequency of 18k Hz.

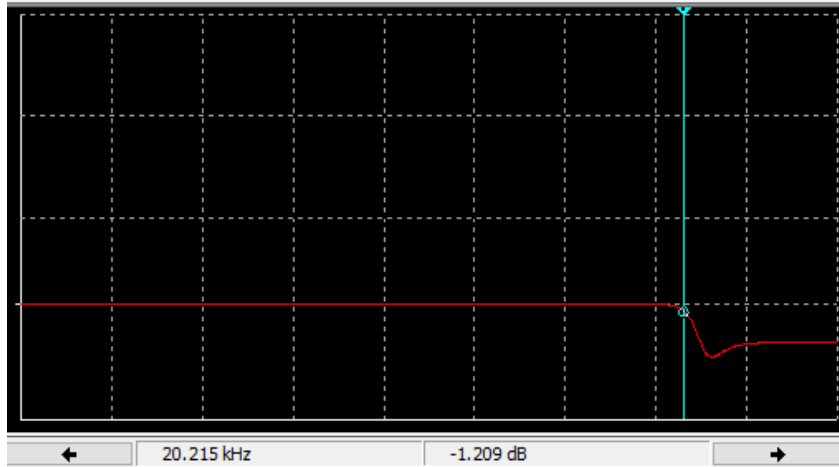


Figure 10

The testing of the breadboards was done by sweeping the function generator through the audio frequency range. The peak-to-peak voltage was recorded with the corresponding step in frequency. This was then converted to logarithmic and then plotted which showed where the -3dB cutoff was. This process is shown below in Figure 11. The y-axis of this plot is the dB, so going across on -3 where that crosses with the data points is the cutoff frequency, as shown with the red line and highlighted data. The resistor values were chosen based on how accurate the cutoff was after the sweep with what we initially expected. For example, if we expected the corner frequency to be 440 Hz and after the sweep, we achieved a corner of 350 Hz . We would need to adjust the resistance to get closer to the 440 Hz . This process was done for both high pass and low pass filter on the low-end (20 Hz), mid (440 Hz), mid (3500 Hz), and for the high end (20 kHz).

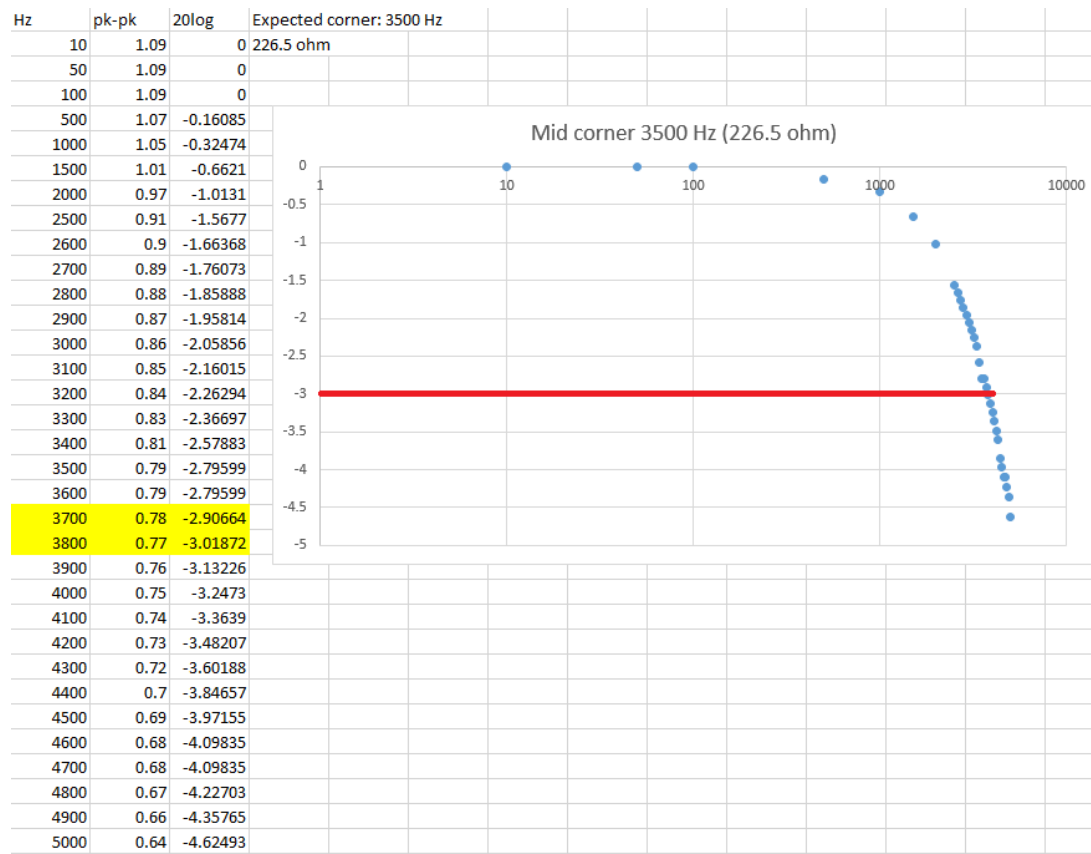


Figure 11

Noise

The interesting part about this design was trying to amplify the noise instead of suppressing it; which was a typical occurrence in circuit design. The first challenge was solved by increasing the values for the capacitors from nano-farads to micro-farads. Not only did this create a better DC bias it also made the circuit work as expected by creating an audible signal. The next challenge that was faced was trying to maximize the gain through the BJT without distorting the noise to get the output to be 2.5 Vpp which means the voltage rms needs to be 0.884V. Once, all the proper components where selected the output provided a sound that was desired

Below is the time domain response:

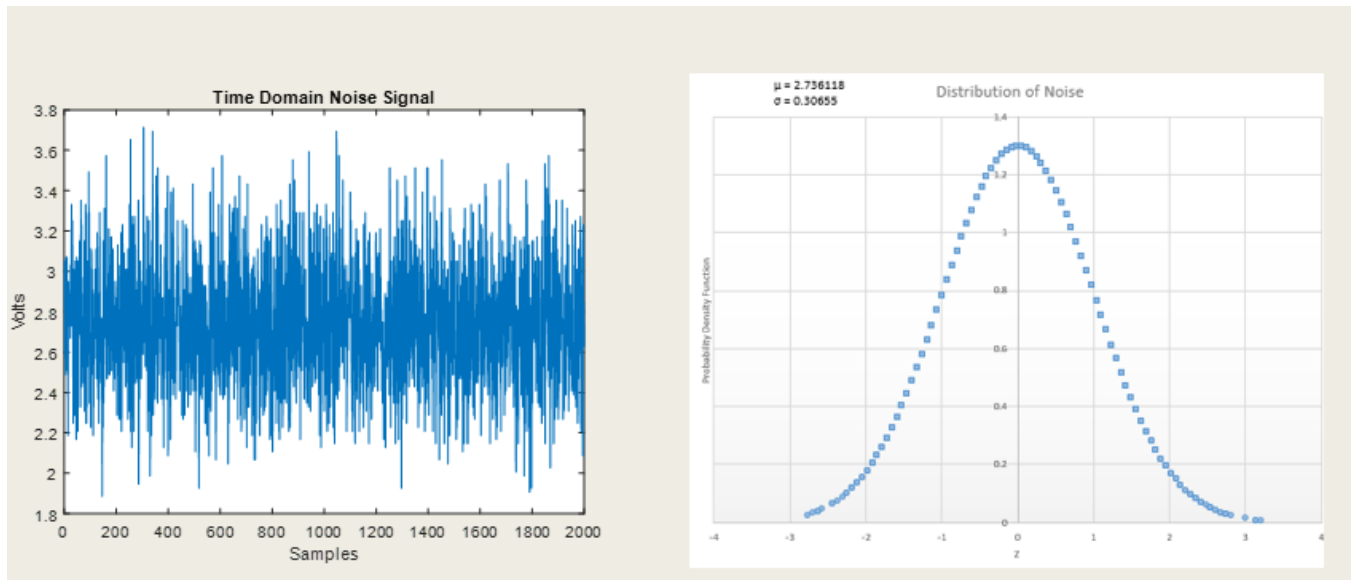


Figure 12

What is seen by this is a perfectly random distribution of the voltage magnitude.

Testing the frequency response, the power spectral density is shown below.

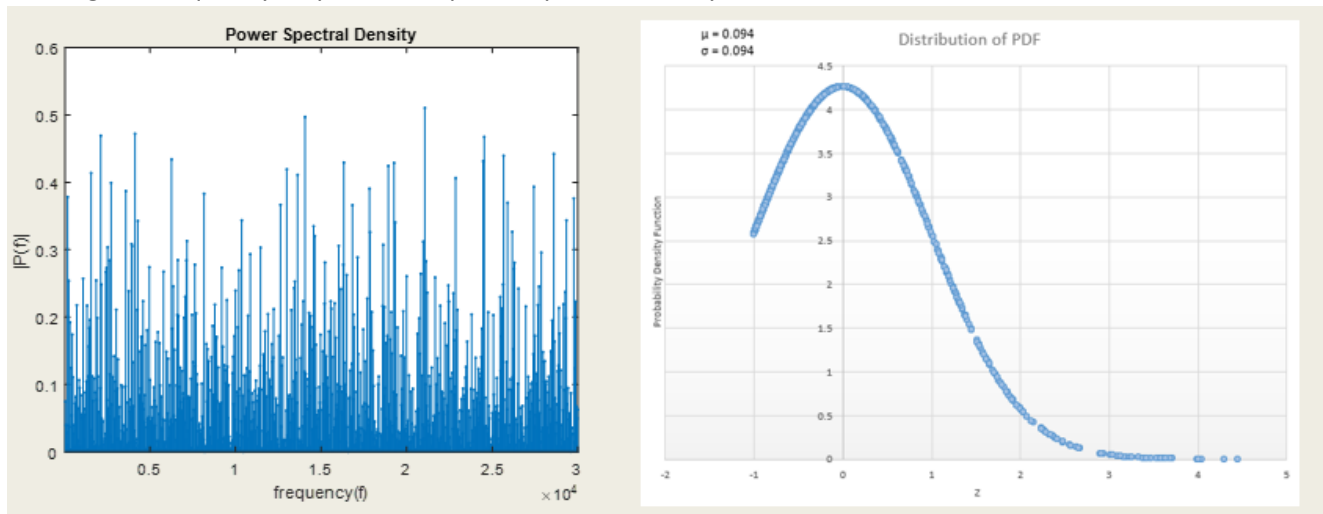


Figure 13

This shows perfect white noise because the mean & standard deviation are equal, and the distribution is completely random. Thus, showing a functional white noise generator.

Output Amplifier

Physical testing of the circuit has not fully wrapped up on this section of the project. But the algorithm for this section has been completed. The algorithm for this section goes as the following. It has two functions, one for the volume control and one for the ADSR section. In the main if the volume slider is adjusted on the user interface it will trigger a function that takes the slider value and sends it to the function to change the digital potentiometer to the desired value. For the ADSR section, this function is

triggered whenever a key is pressed on the user interface. From here it will trigger the ADSR function with the value of the slider for the four parts A-D-S-R. These values will then be in the function used to calculate the rate that the resistance value is change which in turn will affect how the signal is modified and how it sounds.

Mixer

To have a functioning mixer the output should not exceed the line level voltage of 2.5Vpp. The way that this circuit was tested was by using all six of the inputs and sending in different signals and varying their percentage. What was tested and measured was the voltage from the output and did that correlate with the expected value from the percent the user selected. The percent attenuation is driven by a digital potentiometer which was controlled by an Arduino during testing.

The error bounds where measured on this device to see if it would negatively affect the desired output. It was seen that this devise error bounds would not distort the signal or be off where the user would not be able to select a value.

This image below shows the combination of a square wave being generated from one oscillator and a sine wave being created from another oscillator. This test result below showed full integration between the hardware and software as the user can adjust the percent desired on the controls of the use

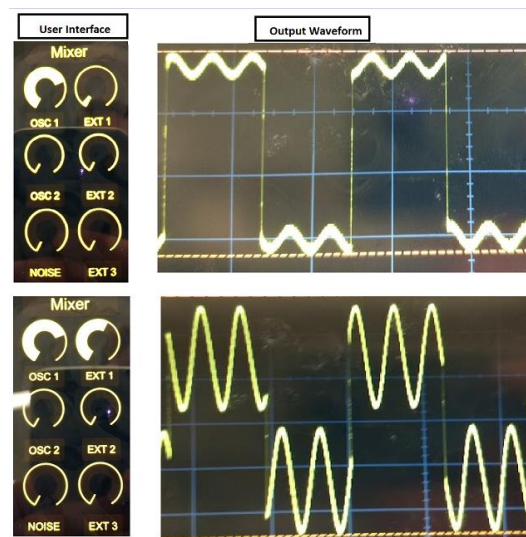


Figure 14

User Interface

The only results that have been obtained are for the implementation of Wi-Fi connectivity, and OSC data transmission. We are able to implement software faders and buttons. In our test, the press of a button turns on an LED controlled by the microcontroller. Once the other modules advance in the design phase, the user interface will be programmed to control those modules vs just turn on LEDs. The figure below shows the current prototype for the main interface design running on the iPad version of TouchOSC.

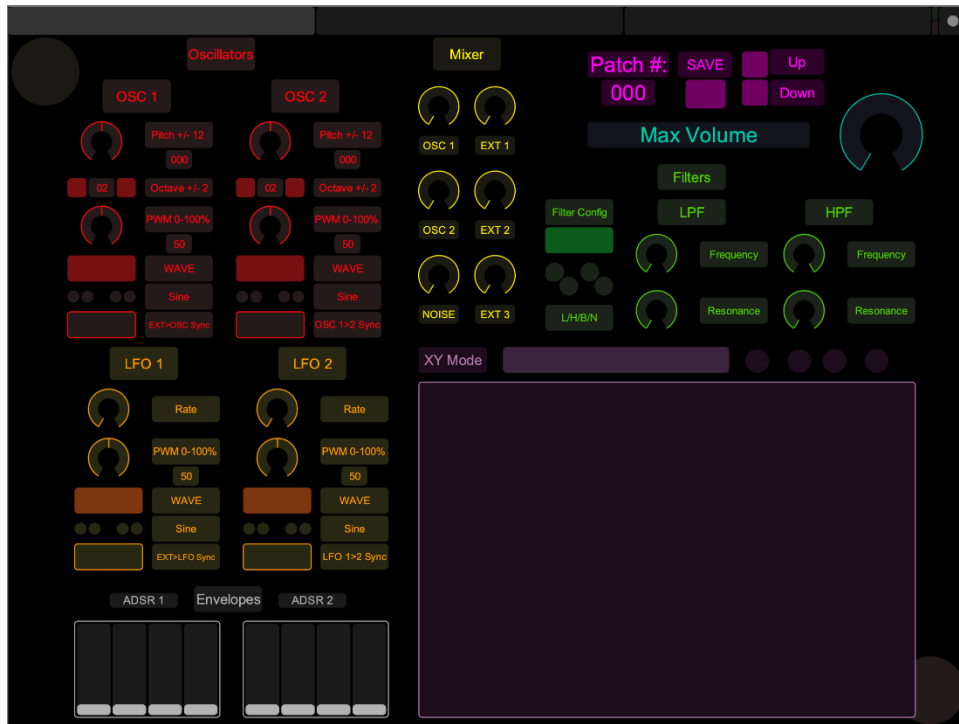


Figure 15

Issues and Challenges

Overall the group did run into some issues and challenges. In the output amplifier portion of the project one of the issues we faced was the understanding of picking which reference to use for ground. When we discussed this with both Dr. Geiger and Dr. Chen we were able to get a firm understanding and grasp on the idea. Following that we were then able to quickly figure out the component that we needed for the project. For the filter portion, at first, we went with a fourth order design. After talking to Dr. Geiger and the rest of the group, we realized we can achieve the same desired specs by designing a second order filter, which is much easier to design and drastically reduces possibilities of errors.

Closure Materials

The objective for this project is to create a musical synthesizer utilizing a digital user interface. The report lays out the design, design conditions and specifications, and test procedures for each module. We split the project up by each team member taking a module to create. Every module has a specific set of design constraints and specifications that need to be met in order to function properly. We made our designs to meet these specifications and the explanation of all of them from design approach to results is mentioned within the report.

We have a few tasks left in the schematic design phase, but once we complete these tasks, we can fully start the testing phase. The testing phase is quite comprehensive, so we have allotted enough time to catch minor errors before sending out our schematics to be printed on PCBs. For many modules, there are several different ways the circuit can be designed to reach the same specifications. We all have

designed each module in a way to help reduce cost as much as possible. We also have designed each module so that it is very easily transferrable to a PCB while minimizing room for error when OSHpark makes the PCBs. Thus, when we start creating the final design all the corner cases should be taken care of. Therefore, the assembly will create a functioning musical synthesizer. Which will be validated by testing each module.

Resources

- [1] B. M. Wilamowski, J. D. Irwin, "Noise in Semiconductor Devices," in *Fundamentals of Industrial Electronics*, 2nd Edition, CRC Press, Boca Raton, Florida.
- [2] C. R. Nave, "Just Noticeable Difference in Pitch," *hyperphysics.phy-astr.gsu.edu*, Available: <http://hyperphysics.phy-astr.gsu.edu/hbase/Music/cents.html#c4>. [Accessed Oct. 12, 2018].
- [3] Curtis Electromusic Specialties, "Voltage Controlled Oscillator," CEM 3340/3345 datasheet, 1980.
- [4] Ecee.colorado.edu. (2018). *ECEN 1400, Intro to Digital & Analog Electronics, Spring 2014 - Create LM386 Component in Multisim*. [online] Available at: <http://ecee.colorado.edu/~mathys/ecen1400/Software/CreateLM386.html> [Accessed 9 Oct. 2018].
- [5] Ni.com. (2018). *What is Multisim™? - National Instruments*. [online] Available at: <http://www.ni.com/en-us/shop/electronic-test-instrumentation/application-software-for-electronic-test-and-instrumentation-category/what-is-multisim.html> [Accessed 09 Oct. 2018].
- [6] Oshpark.com. (2018). *OSH Park* ~. [online] Available at: <https://oshpark.com/> [Accessed 01 Oct. 2018].
- [7] "Sallen-Key High-pass Filter Design Tool," *Okawa Electric Design*. [Online]. Available: <http://sim.okawa-denshi.jp/en/OPstool.php>. [Accessed: 25-Nov-2018].
- [8] "Sallen-Key Low-pass Filter Design Tool," *Okawa Electric Design*. [Online]. Available: <http://sim.okawa-denshi.jp/en/OPstool.php>. [Accessed: 25-Nov-2018].
- [8] Sine.ni.com. (2018). *Ultiboard™ Software - National Instruments*. [online] Available at: <http://sine.ni.com/nips/cds/view/p/lang/en/nid/203428> [Accessed 05 Oct. 2018].
- [9] Tuttle.merc.iastate.edu. (2018). *EE 230 : Topics*. [online] Available at: <http://tuttle.merc.iastate.edu/ee230/topics.htm> [Accessed 07 Oct. 2018].
- [10] Ti.com. (2018). [online] Available at: <http://www.ti.com/lit/ds/symlink/tl082.pdf> [Accessed 20th Oct. 2018].
- [11] Integrated Silicon Solution, Inc., "2.95W MONO FILTER-LESS CLASS-D AUDIO POWER AMPLIFIER," IS31AP2005 datasheet, Aug. 2015 [Revised Sept. 2015].