#71 Digital Theremin with LED

Team - Phong Che (pche3), Juan Chapa (jachapa2), Chawakorn Wichulada (wichula2)

ECE 445 Project Proposal - Spring 2017

TA: Jose Sanchez Vicarte

Mock Design Review

1) Block Diagram

![Block Diagram of Overall Circuit](image)

Figure 1: Block diagram of Overall circuit

Above is the schematic of our design for the instrument that we came up with. Each rectangle represents a general component to the circuit. The block diagram will identify each component in the circuit that will be analog and those that will be digital signal through the legend provided using a color scheme for the wires.

2) One circuit schematic
Above is a circuit diagram for the finished product from ECE 395. Here are the changes that will be made. The zxsensor1 will be replaced with a different sensor that converts distance into analog voltage. Between the new sensor and the microcontroller will be an ADC that converts the voltage into a 12-bit digital signal [4]. A second component like I just described will also be added to one of the I/O pins of the microcontroller. Also, where it says output, there will be a waveform generator chip followed by a bandpass filter followed by the output. Attached to a second pin will be a second waveform generator followed by a second bandpass filter. At the end of both bandpass filters, they will both be attached to the same output which then goes to the volume pedal and then the speaker.
3) One calculation

We will have two sensors with a total frequency range of 65.41 to 523.25 Hz (C2 to C5 or three octaves). The frequency range of each sensor will be different. One will play higher frequencies and the other will play lower frequencies. The higher frequency sensor will play frequencies from 130.81 to 523.25 Hz (C3 to C5 or two octaves). The lower frequency sensor will play frequencies from 65.41 to 261.63 Hz (C2 to C4 or two octaves). One octave has 12 different notes (seen below), so to play every single note requires representing at least 24 distinct pieces of information. However, the goal is to transition from note to note as smoothly as possible, which means adding semitones. The goal then becomes to represent as many notes as possible. This is essentially done by adding enough semitones that the human ear can’t hear a difference between the different states. This can be done by first choosing the number of bits used to represent distinct distances and therefore distinct frequencies. MIDI uses 12 bit information since anything below 12 bits would make it difficult to create smooth transitions between notes as they rise and fall. This is what we plan on using. With 12 bits of information, we can represent 4095 different values, which means 4095 different notes can be played, which means that if each sensor plays 2 octaves, there are 4095/24 - 2 ~ 167 distinct semitones that can be played between notes (the -2 subtracts the two real tones). On the frequency end, 523.25 - 130.81 = 392.44 Hz range and 261.63 - 65.41 = 196.22 Hz. The reason for this is simply due to the way human
perceive hearing. As you can see, the frequency range for the higher frequencies is twice that of the lower frequencies, which is characteristic of the logarithmic scale that models human hearing. That being said a possible way to model the increase of notes with code would be with an inverse log equation such as \( f = \left(\frac{1}{20}\right) \times 10^{x/r} \) where \( x \) is the distance, \( f \) is the frequency and \( r \) is some reference point (such as \( r = 7.828 \)) to properly attenuate the pitch. This means that the extra amount of semitones will be more important for the higher frequencies. If I used fewer bits, for example 8, I could only represent \( 256/24 - 2 \sim 9 \) semitones between notes, which is a drastic difference.

Figure 2: Representation of an octave [2]
Figure 3: Flowchart of our algorithm
4) **One plot (simulation or experiment)**

Goal:

![Figure 4: Signal of a square wave.](image)

**Experiment Result:**
Figure 5: Signal achieve through the design

The top figure is a square wave of 261.63 Hz (or middle C). This is an example of what a square wave should look like after leaving the microcontroller and even after leaving the waveform generator if it is in the square wave mode. The bottom figure is an example of what the waveform actually looked like after exiting the microcontroller. Early on in 395, pure square waves were only able to be generated before the sensor Testing will be done with oscilloscope

5) One Block Description

Sensors (High Pitch):
The sensor will provide a signal that will represent a high pitch note. It will accurately measure the distance between the hand and the sensor with a range of 0 to 80 cm [3]. To identify the distance it will utilize infrared sensors to detect. Once the distance have been detected an analog signal of will be sent out to represent the distance and be will be process by the programmable hardware to form the sound.

6) Requirements and verifications for one module from the block diagram

LED array display:
Requirement:
1. Must be able to light up the correct LED represented by the note
2. All LED must be visible from at least three meter away.

Verification:
1. Requirement 1 verification.
   a. Send a specific note signal to the display to test each of the 12 notes.
   b. Check each LED is wire correctly like the layout we design for the notes.
2. Requirement 2 verification.
   a. Measure 3 meters distance from LED
   b. Check that when the LED is on, it is clearly visible to a human eye at least 3 meter distance away.

7) Ethics and Safety statement

Safety is a very important factor in our implementation of the project. Throughout the development of our project, one safety risk that we will pay close attention to is the monitoring of our power source(s). It is important that as we develop our project we keep in check what voltages are going through each component to make sure that it does not overload the capacity that could result in possibly frying it or starting a fire. Beyond the development phase, we will apply a safety measure to the circuit with a fuse to ensure that the current flowing through product is the correct amount to prevent the component from being overloaded.

One possible ethical issue that could arise is that we might not be aware of our product not being able to produce all the notes/pitches that we claim. According to IEEE Code of Ethics, it is important that we are honest and realistic in stating claims based on the data that is available to us [1]. Therefore, to ensure that our design address that issue, we will be using a music tuner to test each pitch. We will check that all the range of sound that we claim can be achieved by our instrument.

We as a group envision to follow the IEEE Code of Ethics. We will accept all responsibilities in decisions that we make to ensure the well-being and safety of the public [1]. We want our product to be as safe as possible to any user by applying safety measure listed above. Furthermore, we will be seeking feedback from many different people such as friends, families, instructors, and teaching instructors. As an engineer, we will accept all the criticism of the product, acknowledge the errors, and correct them [1]. Our goal is to meet the expectation that others have for our product and to make it as flawless as possible. Although our product will initially be far from perfect, but with every mistake corrected we will be one step closer to reaching that goal.
8) References


