Audio Effect Pedal

Milestone: *Preliminary Design Review*

Douglas Brock, Jonathon Miller, Stephen Piechowicz, Eric Razanousky

Introduction & Objectives

Music has played a large and impactful part in the lives of many people. It can be found and heard around the world, having all different genres to satisfy the ones who listen to it. There are a vast amount of different instruments played and owned by musicians, and guitar is the most popular in many locales of the western world. For the ones who play this instrument frequently as a hobby or profession, having the ability to manipulate and alter the music they've created is an important factor to improve and experiment their skills. Professional musicians and a certain few who invest in expensive equipment/instruments will have access to many of these widespread features to edit their sound to their liking. Unfortunately, for the ones who mostly play for fun or can't overcome the financial barrier of these specialized instruments, they are hindered from performing audio effects on their sound. Having an inexpensive device that executes one of these basic features would be beneficial to this demographic of musicians.

To meet the needs and concerns stated, we will design a guitar pedal that balances sound fidelity and low cost to create a budget pedal; this will allow people to enjoy desired sound control without stressful expense. This pedal will take a standard ¼ - inch audio jack for guitars and copy the original signal to modify to an audio effect, and then recombine with the original signal. The device will also have a way for the user to control the intensity of the effect, and a display to show the user what they have the intensity set to. The device will also passively offer decent sound quality for a relatively low price. We believe there would be a market for this among companies that would like to see a modular design that could be used and modified to offer many different effects. Hobbyists and startup musicians seeking an expanded inventory of sound and value per dollar spent would find this to be a desirable all-in-one solution, balancing cost and quality of sound.

Methods and Technical Approach

Specifications: The high level specifications for the Audio Effect Pedal are:

- **Responsive:** The audio output of the device should be in real time. The input should have minimal delay of 5 milliseconds.
- **Quality:** The sound of the effect should have a signal-to-noise ratio of 50dB.

- **Informative:** The effect display will accurately show the signal information with 5 LED lights. Each light represents 20% signal strength.
- **Simple:** The interface should be easily understood, and the setup/installation process should be unambiguous. The user should be able to edit the code to control the magnitude of the effect.

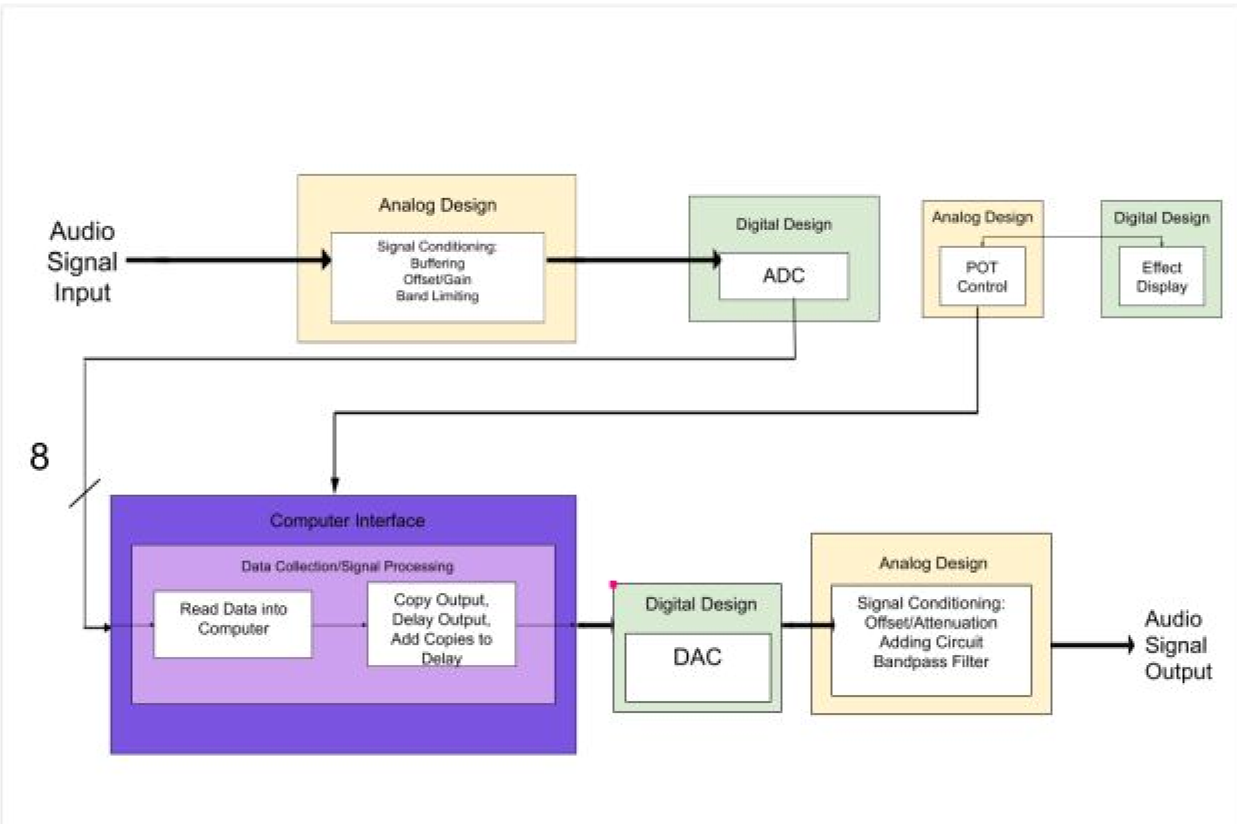


Figure 1: Block Diagram of the Audio Effect Pedal

Figure 1 shows a block diagram of the Audio Effect Pedal. The different shaded backgrounds are indicative of which of the four essential components to the project. Analog, Digital, Interface to a microcontroller, and data collection with signal processing. An overview of each of these components is described below.

Digital Design: The digital components of this project involve making a potentiometer control for the microprocessor, and an LED display for the data of the signal intensity. The LED display will be connected to ladder circuits that will control which LEDs are lit up, allowing the user to see the level at which they have their effect set to. The separate LED display will show the maximum intensity of the effect, and the percentage of the maximum that is currently being utilized. The ADC and DAC will be controlled by code. A GPIO pin will send a pulse to the ADC and DAC signaling to collect data and to process data. After the ADC has a sample ready, the uCPU will collect the data. With 1.5kHz as the upper bound of frequency, the ADC will need to sample the input at 3.5kHz. Both the ADC and DAC will be 8 bit. The DAC will output signal on a pulse from the uCPU.

Analog Design: For the analog design, the input stage impedance will need to be matched to the guitar's impedance. The signal will need to be offset to accommodate the ADC taking positive values only. The signal may need to be boosted as some guitars output 100mV - 400mV. Though the normal guitar output frequencies are 80hz to 1.2Hhz, it would be better to provide a bit of a wider range to preserve the original signal so we will pass the frequency 60Hz to 1.5kHz. The DAC's output will have to be filtered as well. The filter for the DAC will be the same as the filter for input circuit.

Computer Interface: We will be doing some research into microcontrollers/microcomputers (μCPU) that are available. Instead of using a device like an Arduino or Raspberry Pi, we will try to find a specialized μCPU that is cheaper and uses less power to run a specialized task. Arduino and Raspberry Pi units generally have processing overhead contrary to our goals. The control μCPU will accept a parallel input for an 8-bit ADC from the signal, and the control mechanism will run through the μCPUs ADC. There will also be an output for the signal that is done processing.

Data Collection with Signal Processing Elements: A circular buffer will be used to store the (approx.) 5600 most recent 8-bit data points from the ADC. The buffer will then send data through digital filters which will use digital signal processing to modify the signal. A delay algorithm will be run to modify the time at which the μCPU will send the digital signal to a DAC. This signal is connected to an adding amplifier to combine the modified copied signal with the original, which has been maintained from the start. Along with this, we are processing a signal from the pot control through an ADC which will provide a single constant number on how the audio is processed and modified. The pot control will be processed by sampling its signal at a rate between 50 and 300 Hz, and putting the most recent data point in a known location so that it can be used as a reference for controlling the audio signal.

Detailed Design:

The following section provides schematics of the digital, analog, and interface to the microcontroller components of the audio effect pedal. This will include the flowchart explaining the data collection/signal processing elements.

Computer Interface:

Figure 2 contains a schematic of the NUCLEO-F411RE microcontroller, including pin locations, and signal definitions for devices connecting to the µCPU: an 8-bit parallel to the ADC, inputs to the built-in ADCs from pot controls, signal to the ADC alongside the 8-bit parallel port, and utilization of the GPIO ports. GPIO pins will be designated for the 7-segment display.

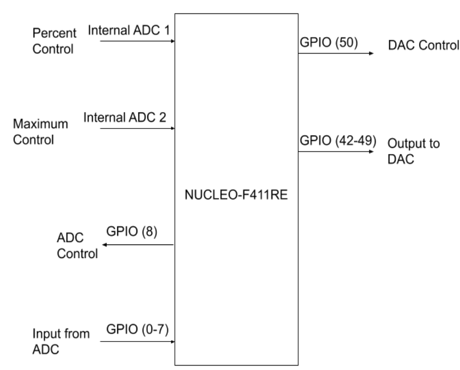


Figure 2: Interface to uCPU ports for different inputs

Data Collection/Signal Processing

Figure 3 contains a flow chart identifying key steps in the data collection/signal processing program on the µCPU: reading and storing the gathered 8-bit samples from the GPIO ports into the circular buffer, input of the circular buffer into time delay functions, and a control algorithm (modeled in Figure 4) for the time delay functions which is modified by inputs from the built in ADCs.

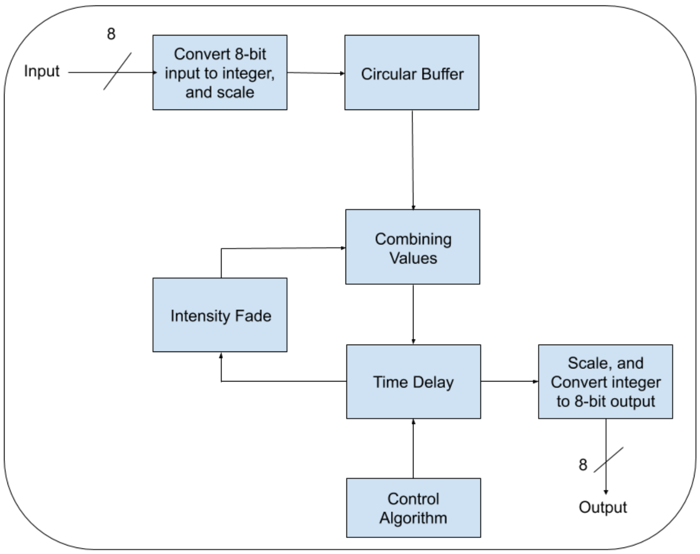


Figure 3: Main algorithm for Data Collection/ Signal Processing

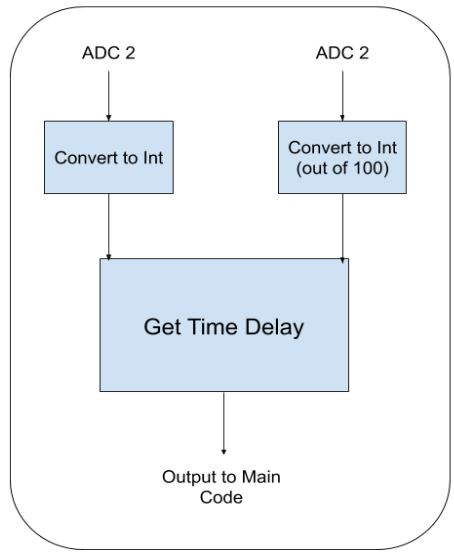


Figure 4: Control Algorithm

Analog Design

From the input, the signal will be buffered by an op amp.The buffering op amp will also provide the gain for the signal. The signal will then go through a capacitor and voltage divider circuit. The voltage divider circuit will put the signal into the middle of the ADC operating range will be right around 2.5V. Figure 5 shows this circuit. The gain of the op amp is determined is determined by $1 + \frac{R_{g2}}{R_{g1}}$. The voltage divider circuit will offset the circuit to 2.5 volts, where $V_{out} = V_{ss}\frac{R_4}{R_4+R_3}$. The next part of the input circuit will be the

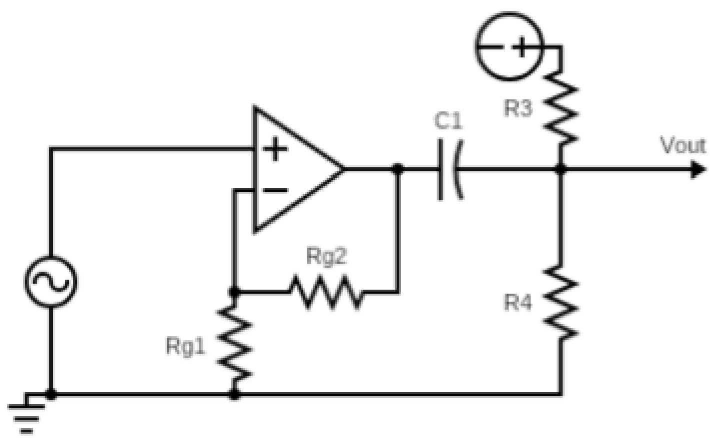


Figure 5: Buffering Op amp and Voltage divider circuit

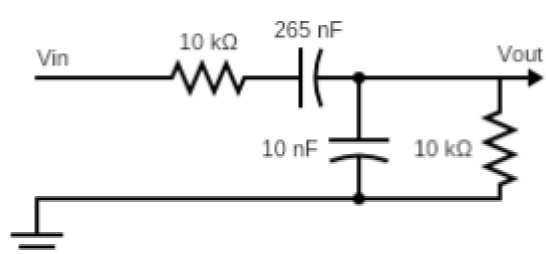


Figure 6: The Band Pass Filter

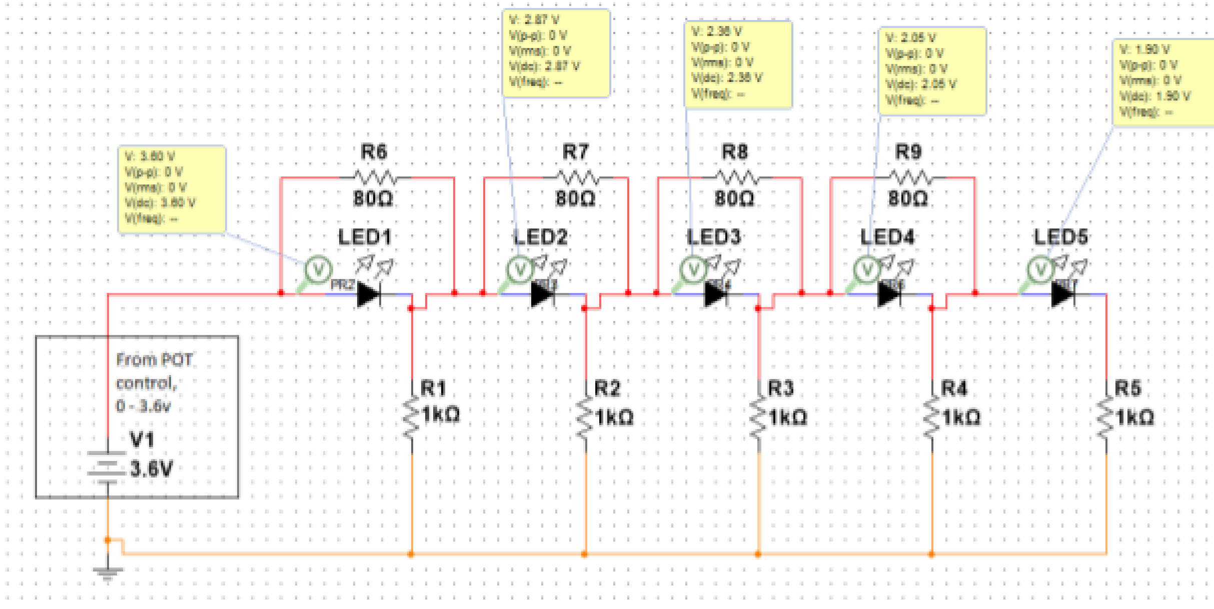


Figure 7: LED display circuit for current strength of effect

filter. The band pass filter is shown in Figure 6. The filter may need to become a higher order filter. The DAC will have the same filter for its output to smooth the output signal. The next part of the analog design is the LED indicator circuit. The comparator circuit is shown below in Figure 7. The LED indicator circuit is a comparator circuit where only after a certain threshold voltage does an LED turn on.

The only conditioning for the potentiometer that needs to be considered is its output voltage. The three main reasons to condition a signal before sending the signal through an ADC can be avoided by matching the max the potentiometer can output to the maximum input voltage of the ADC. The first reason to condition a signal is to offset the signal so it is always positive; the potentiometer never has a negative voltage. The second reason to condition the signal is to provide a gain to the signal to get the signal to use the full range of the ADC; by setting the maximum output of the potentiometer to the maximum input of the ADC this conditioning is not needed. The final reason to condition the signal is to filter out high frequency components; the potentiometer is DC.

Digital Design

The digital Design revolves around the ADC and DAC. The control for the ADC is shown in Figure 8. The ADC's pins to start taking a sample are turned on by the uCPU. After the sample is complete, the ADC's sample ready pin will activate and the sample will be fed to the uCPU where it will be stored. The DAC goes through a similar process. The DAC's pin to start processing data gets activated. When the sample is processed, an analog signal is the output. The digital design for the potentiometer is pretty simple. The DAC's flowgraph is shown in Figure 9. The potentiometer will use an internal ADC.

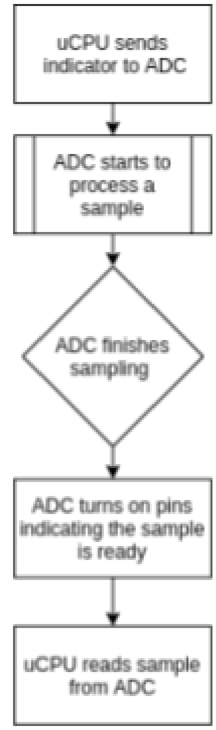


Figure 8: Flow graph of ADC

Results and Discussion:

Currently there are no results available for the different project aspects for the Audio Effect Pedal.

Work Breakdown:

Douglas Brock is currently focusing on writing up the concepts for the code for the circular buffer for storing and reading the data from the signal. He will continue to mold the plans for the code as he goes forward, and continue establishing necessary steps in the code. More detailed plans will be available soon to have an idea on future steps, and the final form of the code. Douglas will also provide updates to his components in reports and weekly updates

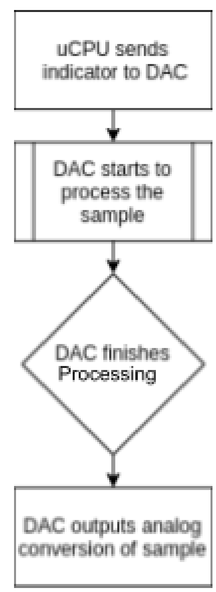
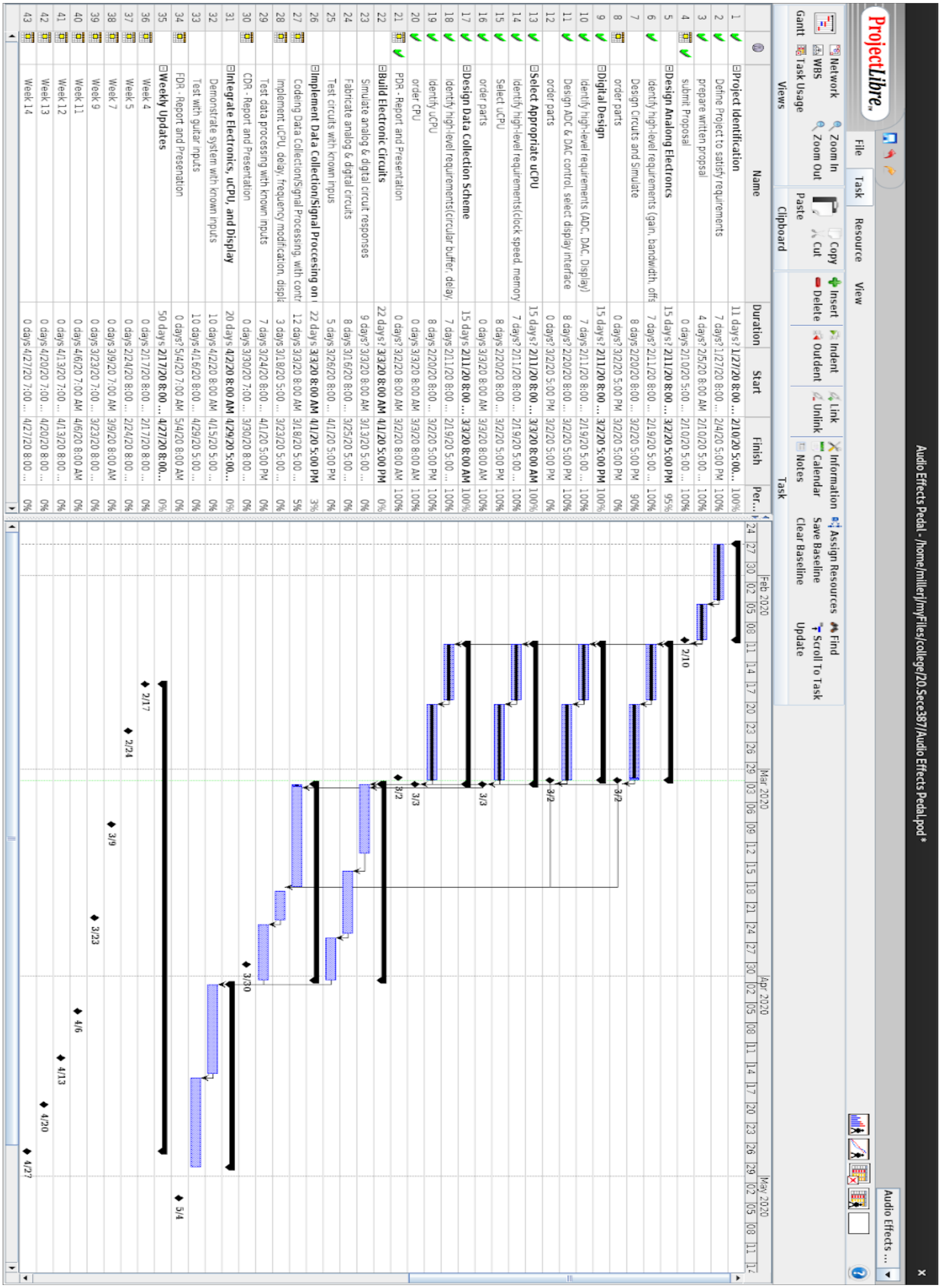


Figure 9: DAC flowgraph

Stephen Piechowicz will continue modeling, building, and testing a pot control for the signal controlling microprocessor, as well as simulating and building the LED chain and ladder circuit for allowing the user to monitor how they control the signal, as well as a secondary control for a maximum effect on the pedal, which will be displayed in two 7-segment displays. Stephen will also provide updates to his components in reports and weekly updates

Jonathon Miller will continue research into the input circuit, as well as running more simulations to make sure that the attenuation, and offset are where they need to be. He will also continue with researching an ADC to purchase while consulting Douglas for information about the specifications of the program and the microcontroller the ADC will interface with. When an ADC is decided on, he will order it, and start to run tests with it. Soldering of PCB. Jonathon will also provide updates to his components in reports and weekly updates

Eric Razanousky will be researching and simulating an adding circuit that will combine the original signal with the processed one from the μCPU. He will also simulate a circuit for reverting the signals back to their original value ranges so that the signal can be output to an amplifier, which is not being designed by our group. He will also do the conditioning for the potentiometer. Once the simulations have provided conclusive results, he will order the parts and start building.He will also find the DC voltage for the potentiometer so that there is no further need to condition the potentiometer. Eric will also provide updates to his components in reports and weekly updates



Audio Effect Pedal