Functional Specification

Year:2024 Semester: Spring Team:12 Project: Microphone Interface
Creation Date: 1/17/2024 **Last Modified:** 2/6/2023
Member1: Liam Roach **Email: leroach@purdue.edu**
Member2: Jakub Kowalski **Email: kowals14@purdue.edu**
Member3: Joshua Hom **Email: hom2@purdue.edu**
Member4: Shubo Xie **Email: xie347@purdue.edu**

Assignment Evaluation: See Rubric on Brightspace Assignment

1.0 Functional Description

Web meetings, live streaming, and online video gaming are all things that have become common in the digital age, and all of these applications involve using a microphone in conjunction with a computer to communicate with others over the internet. However, there are a large multitude of different microphones, and furthermore, situations where these microphones might be used, and it's common that some kind of audio processing of a microphone is desired.

Our microphone interface seeks to provide the following functionality:

- Be powered via micro-USB.
- Receive an analog audio input via 3.5mm TRS jack.
- Apply the following user selectable and user controllable DSP effects:
  - Five band equalization
  - Distortion
  - Delay
- Allow the user to navigate between effects via push buttons.
- Allow the user to manipulate the following DSP parameters via rotary encoders:
  - Equalization (for each band):
    - Frequency
    - Gain
    - Width
  - Distortion:
    - Level
  - Delay:
    - Time
    - Feedback (level)
- Display the states of each DSP parameter (with a separate page for each effect) via a graphical user interface.
- Output the modified audio signal via a 3.5mm TRS jack.

The following are stretch functionality that we will attempt to implement given time:

- Digital audio output via USB
- Additional DSP effects:
  - Compression
  - Pitch adjustment

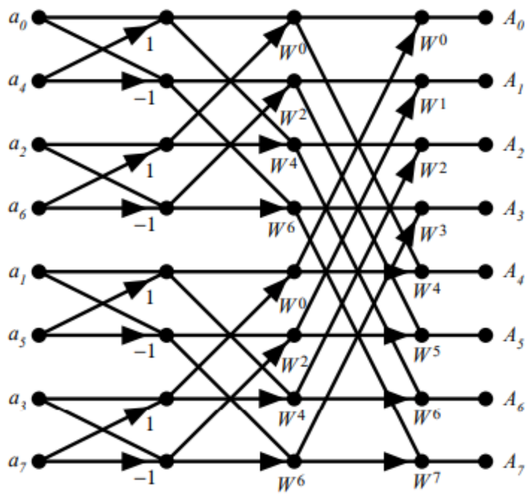
2.0 Theory of Operation

Our device will heavily rely on DSP to implement our audio effects. We can make use of mathematical tools such as the Fourier Transform, Z Transform, and an algorithm such as the FFT to analyze the frequency content of signals and design filters.

To begin, the Fourier Transform is a technique for converting signals between the time and frequency domains. Essentially, it breaks down a signal into a series of weighted sinusoidal signals which helps us analyze the amplitude and phase of different frequencies present in a signal. This is useful in the context of filter design, as it allows us to inspect which frequencies are present in the signal and design filters with cutoffs that eliminate the unwanted frequencies. There are several types of Fourier Transforms which are used depending on the type of signals we are dealing with. In the context of our device, we will need to use the Discrete Fourier Transform (DFT), not to be confused with the Discrete-Time Fourier Transform (DTFT). The DTFT is used with discrete, aperiodic signals which require an infinite amount of sinusoids to synthesize, making the DTFT impossible to calculate using a computer algorithm [c]. The alternative to this is the DFT, which is used on a finite sequence of a discrete, periodic signal. Although the sampled signal of a human voice coming through the microphone isn't periodic, we can capture a section of the signal and treat it as one period of an infinite, periodic signal, which makes the DFT practical to use [c]. The equation for the DFT is defined as:

$$X(k) = \sum_{n=0}^{N-1} x[n] * e^{-j2\pi \frac{k}{N} n}$$

In the context of DSP, a powerful algorithm for calculating the DFT is known as the Fast-Fourier Transform (FFT). The advantage of this algorithm is that it calculates the DFT much faster: O(Nlog(N)) vs O(N^2) [d]. Without delving too deep into the specifics of the algorithm, the FFT essentially breaks down a sequence of data comprised of N points (power of base 2) into smaller DFTs, which are then later combined using "butterfly operation" until we reach the original length of the sequence. This process overall performs fewer multiplications and additions compared to the straightforward computation of the DFT. Here is the representation of the butterfly operation on an 8-point DFT [e]:



Another useful tool is the Z Transform, which is similar to the Laplace Transform, however it was designed for discrete systems. It is commonly used in the context of filter design since it allows us to analyze a system and observe its behavior at different frequencies. The Z Transform equation can be expressed as:

$$X(z) = \sum_{n=0}^{\text{inf}} x[n] * z^{-n}$$

When designing filters, we usually start with some kind of difference equation used for defining the filter system. Once we transform this into the Z domain, we can solve the systems transfer function:

$$H(z) = \frac{Y(z)}{X(z)}$$

From this transfer function, we can then analyze the different poles and zeros present in the system. We can determine a region of convergence for the system, and then deduce whether or not it is stable depending on if there are poles present inside of this region.

3.0 Expected Usage Case

Our microphone interface is expected to be used by wish to apply digital signal processing effects to a microphone. The device accepts and produces an analog audio signal interfaced via 3.5mm TRS, so theoretically it could be used audio input/output. Thus, the device could end up being used in a wide array of situations, but the primary use case is when a user wishes to connect a microphone to a computer and wishes to apply effects to the microphone signal.

4.0 Design Constraints

4.1 Computational Constraints

Our design uses one MCU and performance is a concern as we are performing digital signal processing. Therefore, a powerful MCU is desired. A potential audio buffer for DSP might be 2048 samples at 24 bits per sample, which would require at least 6 kilobytes of memory. Additionally, since DSP is being performed, it is required that the microcontroller have support for floating point instructions. 44kHz is a typical sample rate for audio, so both microcontrollers would need to operate at at least that speed. However, a much higher clock would be desirable as the DSP must be performed in real time, will be using algorithms like the Fast Fourier Transform, and there are multiple simultaneous DSP effects.

4.2 Electronics Constraints

Our project uses one screen to display the graphical user interface, and it also includes several rotary encoders to control the digital effect applying to the input signal. Digital/analog conversion of the audio signal will be handled by a codec connected to the MCU via I2S and I2C. The I2S interface has synchronization problems in high sampling frequency and bit rate so we should make sure the transmitted data bit rate is under I2S's maximum bit rate. The major interfaces will include I2C and I2S. Since our project is powered by the output USB port, it is expected to have power regulation including RLC filter and low-dropout voltage regulator. The constraint of filter value will be decided by the amount of power for our project. The last constraint is to have a clock speed that is fast enough to not undersample the rotary encoders.

4.3 Thermal/Power Constraints

For our project, the digital signal processing will require a substantial amount of computation. This leads us to believe that the processor(s) will be the most significant contributor to the temperature within our device. An article states that consumer grade electronics operate in a range from 0°C to 70°C or 85°C[a]. We have leading ideas to 3d print our packaging. Another article recommends that standard plastic (ABS) should be kept in a continuous temperature range of -20°C to 80°C [b]. From these sources, we should aim to keep our device to a continuous temperature of 70°C .

Another main aspect of our design is to connect devices via USB cables. This means that our device should at least be able to run on the maximum output of a computer USB port which is 5 volts. Since some electronic components will have different analog performance based on the supply voltage, we need to consider adding the power regulation to our circuit. Additionally, the device should not consume so much power such that a plugged in microphone is not able to power on. Beside that, the tradeoff between power consumption and noise susceptibility is also important because higher input voltage will have higher noise susceptibility.

4.4 Mechanical Constraints

Our intention is for this device to be used at room temperature and on a table. Our mechanical constraints for packaging should be large enough to protect the electronics from common hazards and elements such as dust, small spills, and low drops. We should adjust the packaging to be reasonably small. Our initial thought is to 3d print our packaging. This allows us to adjust the material later.

4.5 Economic Constraints

This device should be a relatively low-cost device. Most of the cost funds will be allocated toward the microcontroller that will be doing digital signal processing. Additionally, we have gone forward with the idea that we will use two MCU's for our design. One to act as a host for the input and one as a device for the output. Although we will be needing another MCU, it won't be doing any DSP. This means that it won't need to have high computational abilities thus being cheaper than the other MCU. A product with a similar construct is the FIFINE Gaming Audio Mixer which is being sold for $58.99. Since our baseline product will include a minimal number of effects and custom presets, we should try to keep it under this price point.

4.6 Other Constraints

Because the device will support multiple DSP effects, that all have different parameters, it's necessary that there be a graphical interface that can show all current parameter values of each effect in real time. The device will have a small form factor and is meant to sit on a desk, so the parameters must be displayed large enough so that they can be legible from about half an arm's distance away.

5.0 Sources Cited:

[a]M. Wright, "Selecting and applying mcus in extended- and high-temperature applications," *DigiKey*, 2012. [Online]. Available: https://www.digikey.com/en/articles/selecting-and-applying-mcus-in-extended--and-high-temperature-applications

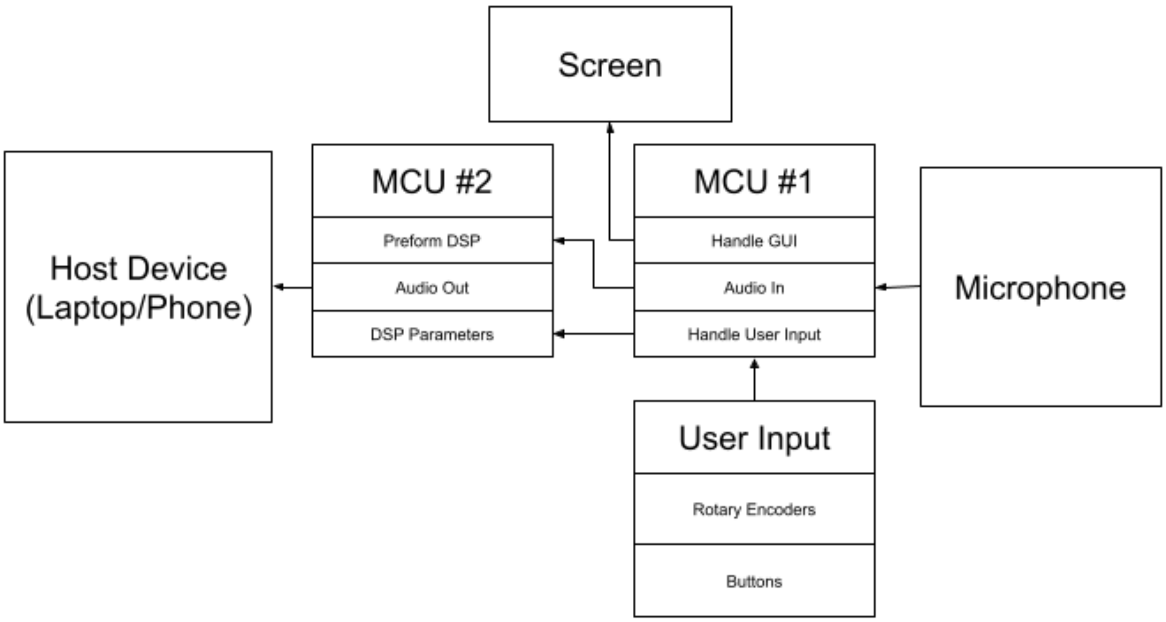
[b]"Acrylonitrile butadiene styrene (ABS) plastic," *Polymershapes*, 2023. [Online]. Available: https://www.polymershapes.com/product/abs/

[c]S. W. Smith, *The scientist and engineer's guide to digital signal processing*. San Diego, Calif.: California Technical Pub, 1997.

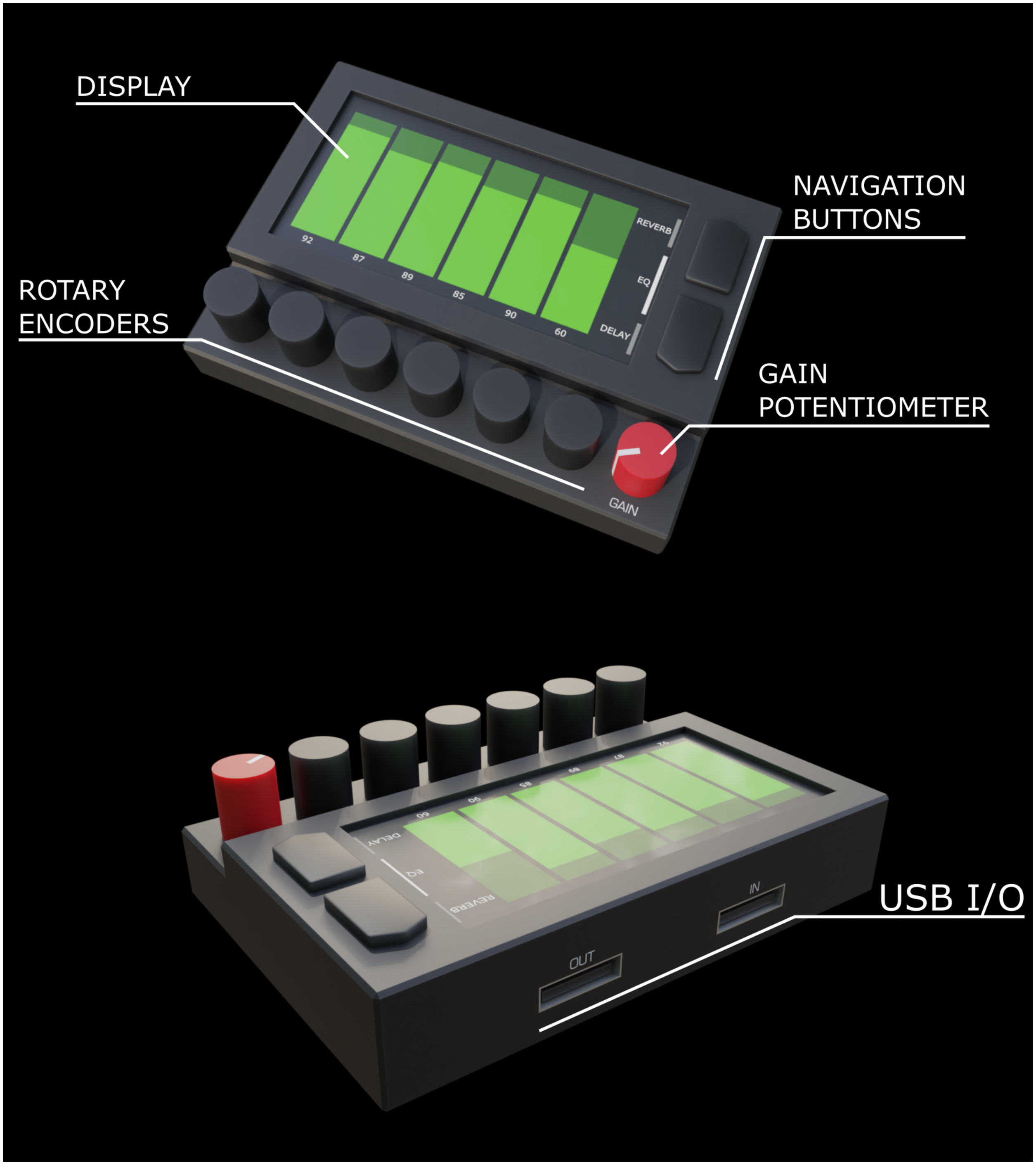
[d]J. W. Cooley and J. W. Tukey, "An Algorithm for the Machine Calculation of Complex Fourier Series," *Mathematics of Computation*, vol. 19, no. 90, p. 297, Apr. 1965, doi: https://doi.org/10.2307/2003354.

[e][1]P. Heckbert, “Fourier Transforms and the Fast Fourier Transform (FFT) Algorithm,” *Computer Graphics*, vol. 3, pp. 15–463, 1995, Available: https://www.cs.cmu.edu/afs/andrew/scs/cs/15-463/2001/pub/www/notes/fourier/fourier.pdf

Appendix 1: Functional Block Diagram



Appendix 2: Sketch of Project Prototype


DISPLAY
NAVIGATION BUTTONS
ROTARY ENCODERS
GAIN POTENTIOMETER
REVERB
EQ
DELAY
92
87
89
85
90
60
GAIN
USB I/O
OUT
IN