Formi: A New Age of Music Listening

ECE 445 Design Document

Kshithij Kamal and Drew Stogner
TA: Jon Hoffman

10/1/19

1 Introduction

1.1 Problem and Solution Overview

With Americans spending on average 4.5 hours per day listening to music, it is no doubt that we love music. [1] In fact, it has been used as a stress reliever for hundreds of years yet each modern speaker, headphone, etc optimizes for a limited range of music dictated by its sound signature.[2][3]

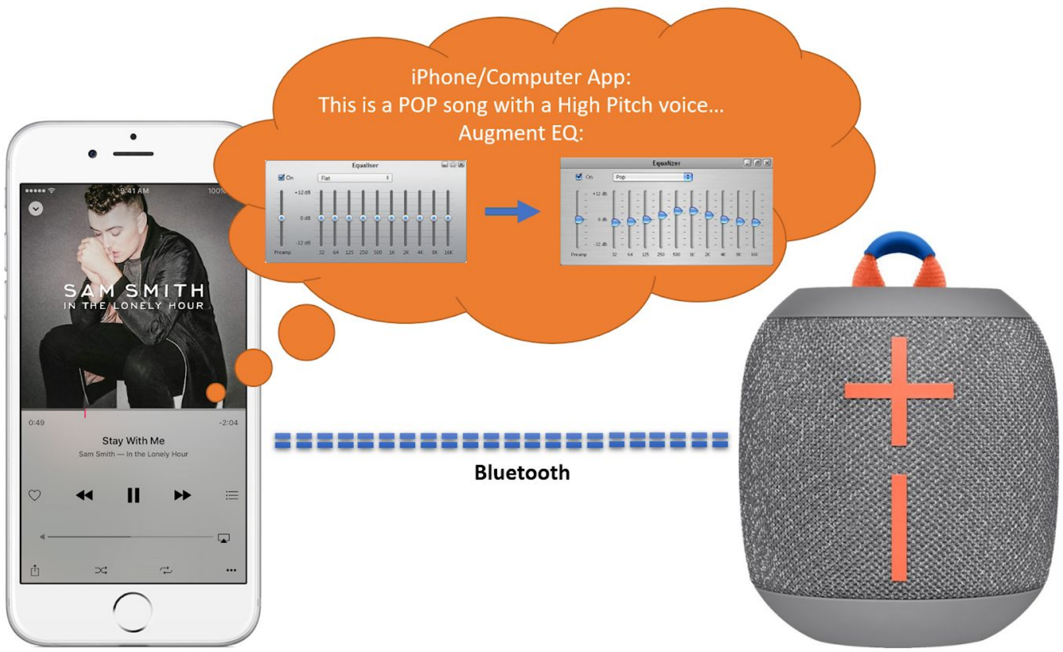
Sound signature emanates from two elements:

1) Programmed equalization in the DACs (Digital to Analog Converters)
2) Circuit Components

As avid music listeners, we recognized this common problem that many musicians and music lovers alike have faced. In order to enjoy a wide and varied range of music genres, people have to buy multiple high-end headphones. This can become quite costly with the price of high end headphones ranging from eighty dollars to over three-hundred dollars. With every headphone having different sound signatures, each one works best for a specific genre or use case and may even cause fatigue when mismatched. This is especially troubling if you love multitudes of music genres, compose or record music. Concurrently, modern pop music is incorporating multiple genres into one. With artists like Lil NasX, Billy Eilish, and Twenty One Pilots all in one genre, the differences in sound can be quite extreme.

Our solution targets all of these elements by optimizing circuit components for adaptability and utilizing a baseline/neutral DAC equalization setting. Building on this, we will create a computer/phone (with Bluetooth connectivity) application that augments equalization based on the music playing. For example, it would analyze *Old Town Road* and make certain EQ parameter decisions based on the musical/signal qualities of the song itself in this case the vocal range. [4] The goal of the hardware portion is to create a Bluetooth based music reproduction system that is uniquely adaptable to a wide range of genres. Concurrently, the goal of the software portion is to create an application that can make EQ decisions dynamically based on song attributes such as genre and characteristic such as a singer's range driven by ML. With a successful POC, our solution has the potential to shape the music listening industry by helping audiophiles, artists, producers,etc. save money(no need for multiple headphones) and enhance the music listening experience for all music lovers. Concurrently, this technology has additional prospects in enhancing tangential experiences such as movies to immersive ones such as AR/VR.

1.2 Visual Aid



1.3 High-Level Requirements

- [1]The hardware device must take musical data through a bluetooth signal and convert it into an analog signal that can be played on the POC speaker.
- The software must be able to dynamically change EQ based on file embedded (basic) genre and [2]ML derived genre/characteristics at 70% accuracy.
- [2]The software must exhibit an equalization change through means of a dB vs frequency augmentation (must be transferred to our device).

Footnotes:

[1] Characteristics can differentiate types of genres. (i.e. high pitch vs low pitch singers)

[2] EQ or Equalization augmentation can be detected as change in decibels over frequency.

2 Design

2.1.1 Hardware Block Diagram

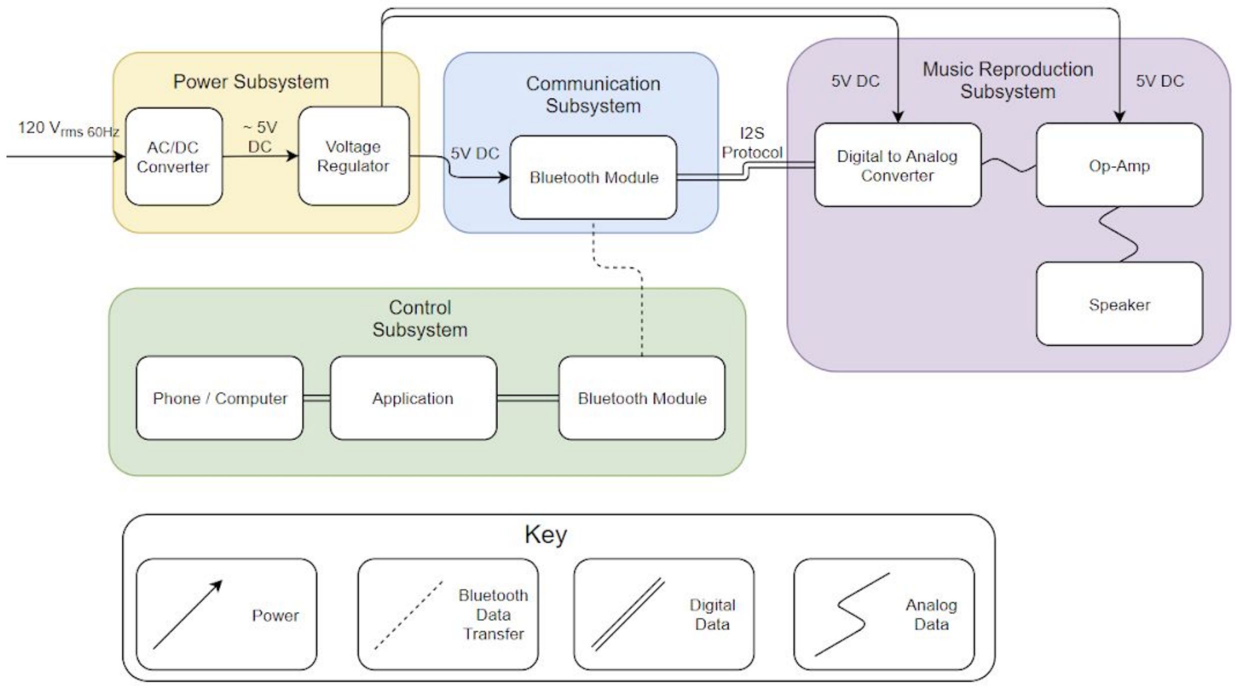


Fig 1. Hardware Block Diagram

Above, we can see that wall power is sent to the AC-DC converter (power subsystem) and, subsequently, the voltage regulator ensuring a safe 5V DC to power subsequent components. After conversion, power is sent to all other PCB components (including Communication and Music Reproduction Subsystem(sans Speaker which will be connected to terminals on the PCB)). The song is chosen, analyzed and augmented with chosen EQ within the control subsystem. Then, it is sent digitally to the Bluetooth module, converted to an analog signal through the DAC, and finally sent to the Op-Amp/Speaker.

2.1.2 Software Block Diagram

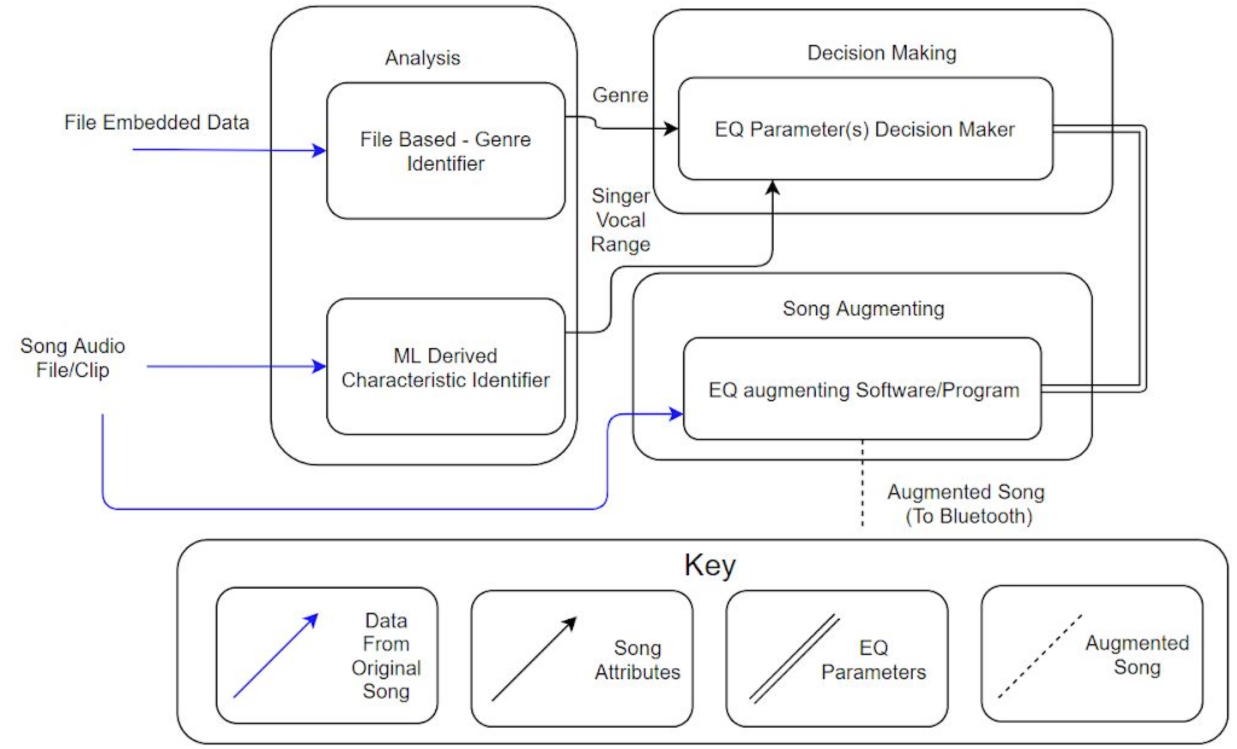


Fig 2. Software Block Diagram

*Above, we can see the high level outline of the software portion of the project. There are two key data elements from the song we want to play that are required: 1)Song/Audio Clip 2)file embedded data. The file embedded data is run through the "File Based - Genre Identifier" to output a genre to the "EQ parameter decision maker." Concurrently, the song's audio is analyzed by the ML algorithm to determine the vocal range of the singer. With both these inputs, the "EQ parameter decision maker" makes decisions on the EQ parameters and outputs it to the "EQ augmenting Software/Program" where the EQ parameters are applied to the original song clip and outputted to the Bluetooth module for playback.

Footnotes:

[1] This block diagram only refers to the end result of the software portion of the project and does not make any inferences on programs or software needed for creating, analyzing the data set or training the ML Algorithm.

2.2 Physical Design

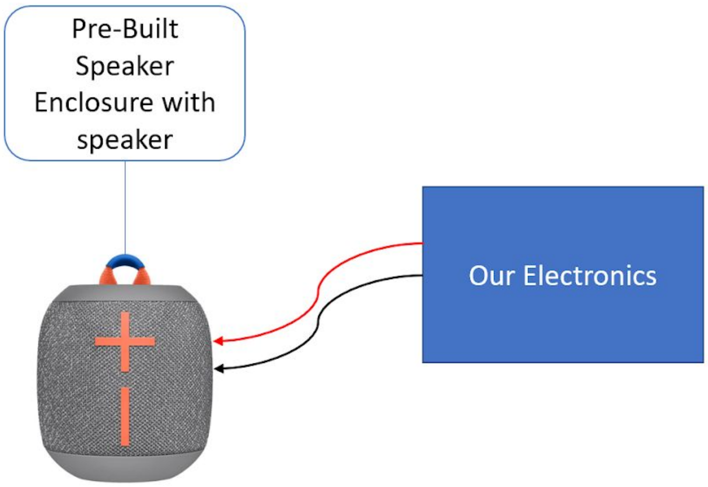


Fig 3. Physical Design Sketch

To alleviate the demands of fine-tuning the physical speaker design, we will use a pre-existing speaker and speaker enclosure setup with pre-tuned acoustics which will connect to our PCB (Printed Circuit Board). Our PCB will be connected to wall power to optimize current powering the speaker and long testing cycles when fine-tuning EQs. Since the control subsystem is a preexisting digital device, i.e. phone or computer, which communicates through Bluetooth to our PCB/electronics, we have not included it in our physical design.

2.3.1 Power Subsystem

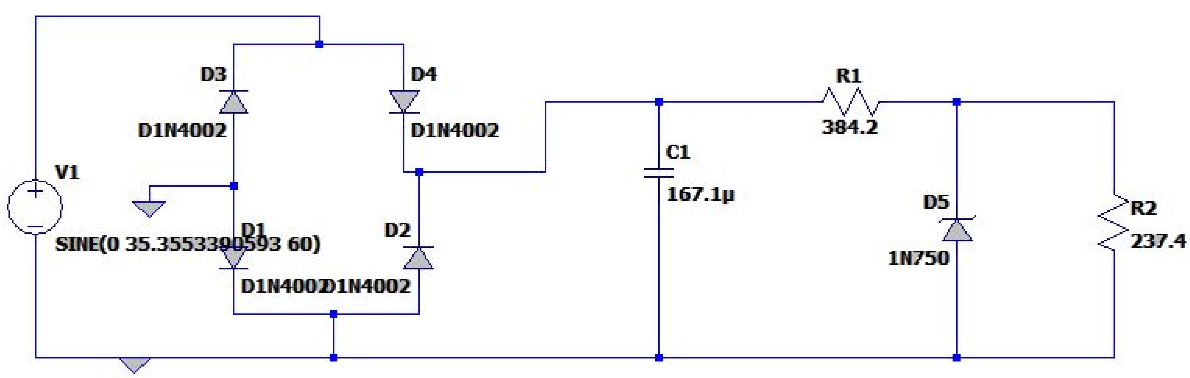


Fig 4. AC-DC converter and Voltage regulator

In figure three is our design for both the AC-DC converter and voltage regulator. What should be noted as well is there is a 5:1 transformer between the input from the wall source and the input to the circuit shown above. This circuit takes a 120 V at 60 Hz and successfully transforms it into anywhere from 4.7 to 4.95 VDC depending on the load resistance. This is mandatory because the DAC, Bluetooth and OP-AMP can be all powered by 3.3 VDC - 5 VDC.

Requirements	Verification
1. Output voltage must be regulated from 4.5 V to 5 V. 2. Output Current cannot exceed 57.5 mA.	1. Observe output voltage on an oscilloscope to ensure voltage remains between 4.5 and 5 V. Change output load from 100 ohms to 1000 ohms in increments of 100 ohms. 2. Observe output current on the oscilloscope to ensure current does not exceed 57.5 mA. Change output load from 1000 ohms to 100 ohms in increments of 100 ohms.

2.3.2 Communications Subsystem

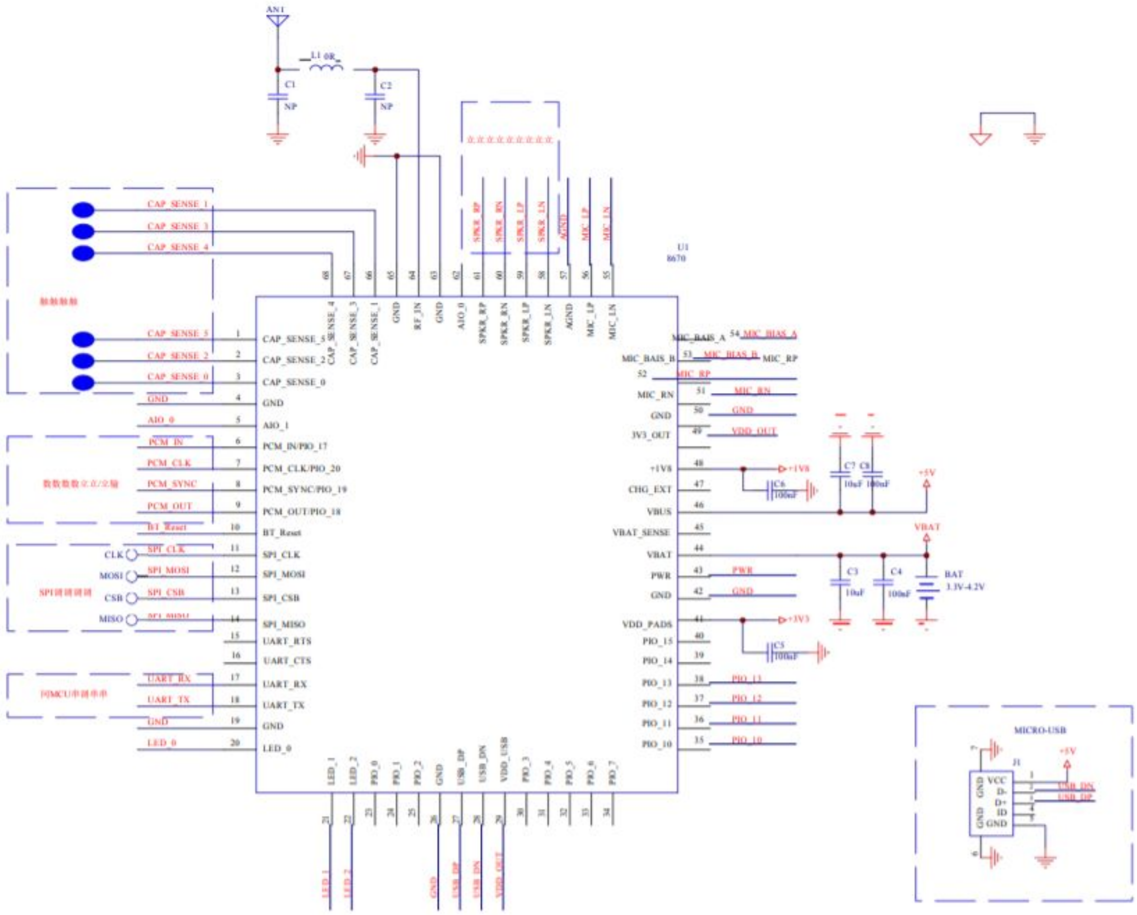


Fig 5. Bluetooth IC

The bluetooth module takes a signal from a computer or phone and then send a digital signal to the DAC. This module allows for wireless signal transferring and as a bonus allows us to bypass the DACs on computers and phones ensuring that the signal is unaltered by any other components other than ours.

Requirements	**Verification**
1. This module has to be able to take a signal from a phone and/or computer and then send that signal from its output.	1. This can be verified using a single tone signal from either a computer or a phone and then using an oscilloscope to measure the output of the bluetooth module allowing us to see if the signal

	was transferred correctly.

2.3.3 Music Reproduction Subsystem

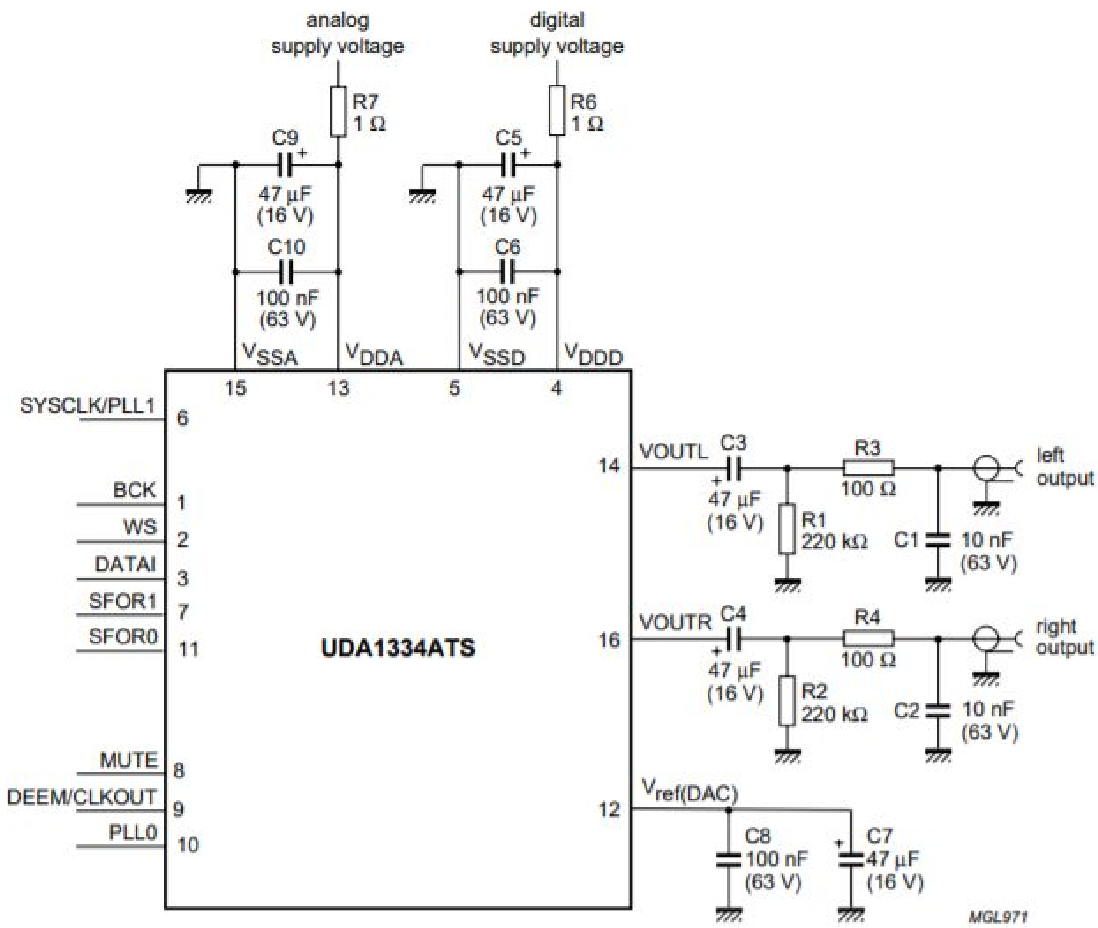


Fig 6. DAC

This component takes into its input the digital signal from the bluetooth module and outputs the analog data to be sent to the amplifier. This is important because in order for music to be played from the speaker it has to be changed from digital data into analog since the speaker will only play the analog signal correctly. [10]

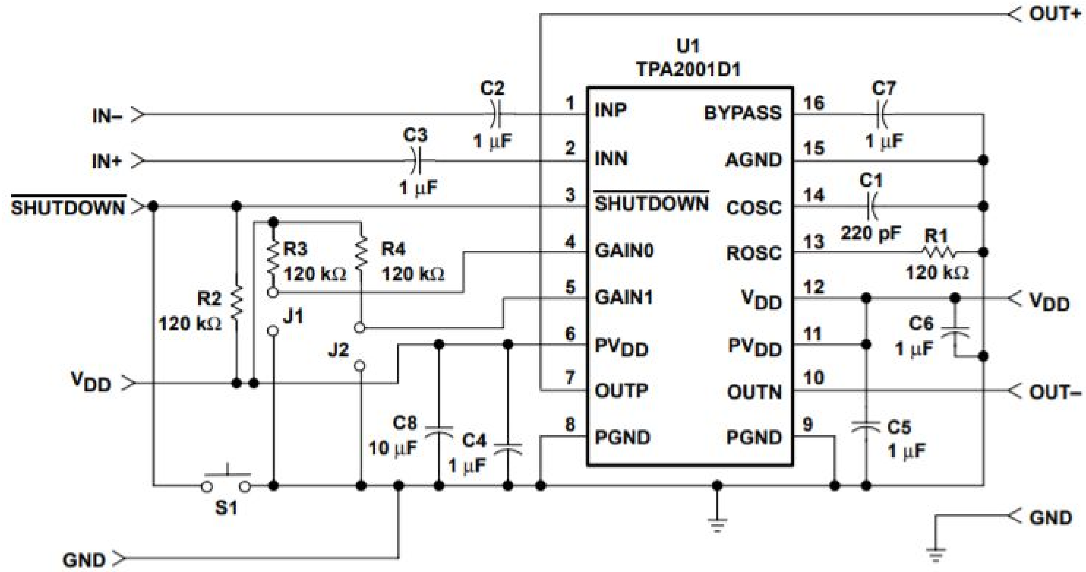


Fig 7. Class D Operational Amplifier

The operation amplifier would take the analog signal produced from the output of the digital to analog converter and then amplify the signal to be sent to the speaker. Without this component, the signal would not be strong enough to produce a recognizable volume from the speaker. The speaker is the final piece. The speaker takes the amplified analog signal from the operational amplifier and then transmit that energy into vibrations that produce sound waves.

Requirements	Verification
1. The DAC has to convert a digital signal into an analog signal correctly. 2. The operational amplifier has to be able to produce 1 watt of power for an 8 ohm load. 3. The speaker has to audibly play sound from an analog signal.	1. Using an oscilloscope on both the input and output of the DAC and sending in a periodic steady state digital signal, we can observe the output and compare to the predicted waveform. 2. Placing an 8-ohm load as the output of the operational amplifier, we can use a wattmeter to measure voltage and current through the load to ensure that 1 watt of power is produced.

	3. Simply sending in a sufficiently powered analog signal should produce an audible result from the speaker.

2.3.4 Control Subsystem

The control subsystem is a software-based subsystem residing as a program on a phone or computer. This subsystem contains the software portion that utilizes metadata to decipher a song's genre in conjunction with machine learning to correctly identify the vocal range of the singer. Using this, this subsystem makes decision on equalization parameters and augments the the signal accordingly to be sent to the Bluetooth module.

As a result of the high complexity of this subsystem, we have created a separate block diagram above (fig. 2). Accordingly, the nature of Machine Learning leaves a lot of unknowns regarding complexity and feasibility which can impede project development/completion. We expand on these problems in the next section (Tolerance Analysis).

Requirements	**Verification**
1. This must correctly identify the vocal range/characteristics of a song. 2. Also, it must change the equalization of the signal depending on the genre. 3. The machine learning aspect of code must be at least 70% accurate. 4. The time latency must be limited to <= 0.5 seconds for the ML decision and any accompanied pre-filtering.	1. This can be tested by displaying the vocal range/characteristic identified and compared to the range prescribed by determining the highest and lowest notes that the singer sang manually. 2. The software will include a graphing function that will take the output of the program and either plot it in a spectrogram of a dB vs frequency graph to illustrate the change made. 3. Using controlled test cases and using a print function in the code will allow us to use a simple true/false result system

	and then computing the effectiveness of the system. 4. Using exemplative test cases, we can capture the cpu time spent for computation for each subprogram. Adding these times up, we can identify the total time spent for the ML decision and any accompanied pre-filtering.

2.4 Tolerance Analysis

The subsystem with the greatest risk for failure is the control subsystem/software portion. A huge portion of the risk lies in the use of ML to make rapid decisions on songs to augment EQ. The two main concerns are 1) The Data Set and 2) Training the Algorithm.

2.4.1 The Data Set

From a layman level, more data points means one can create a more accurate ML algorithm. However, there is no formula to figure out the exact amount of data needed for critical success. It is much more complex and based on multiple factors such as the complexity of the decision it is making, complexity of the learning algorithm, the number of features required to decipher differences,etc. [14] To give an example, an ML algorithm created by Global Fishing Watch to identify illegal shipping activity utilized 60 million data points (grown to 37 billion points over past 5 years) while a Music genre identifier (research project) required a dataset of 40,540 songs. [15] [16] [17]

To create our data set, there are three methods for classifying the audio files:

1) One classifying method is listening and labelling songs as having a certain vocal range. This would be painstakingly difficult to create a large data set in a timely manner. Rather, we could identify artists with a fixed vocal range, and label all of their songs with respective vocal range.

2) The second method is to programmatically filter and,subsequently, analyze the vocal range from a plethora of songs to label it accordingly. This can be done by a unique trick we refer to as

the “anti-karaoke” method which (theoretically) isolates only the vocal elements of a song. Then, we can analyze the resultant audio’s frequency range and classify the song to the identified vocal range. Unfortunately, the anti-karaoke method is time intensive (~20 seconds for 3 minute 30 second song) so it cannot be used in real time hence the need for a robust ML algorithm. However, this latency is fine for dataset creation purposes. Below, you can see an example of the “anti-karaoke” function in action.

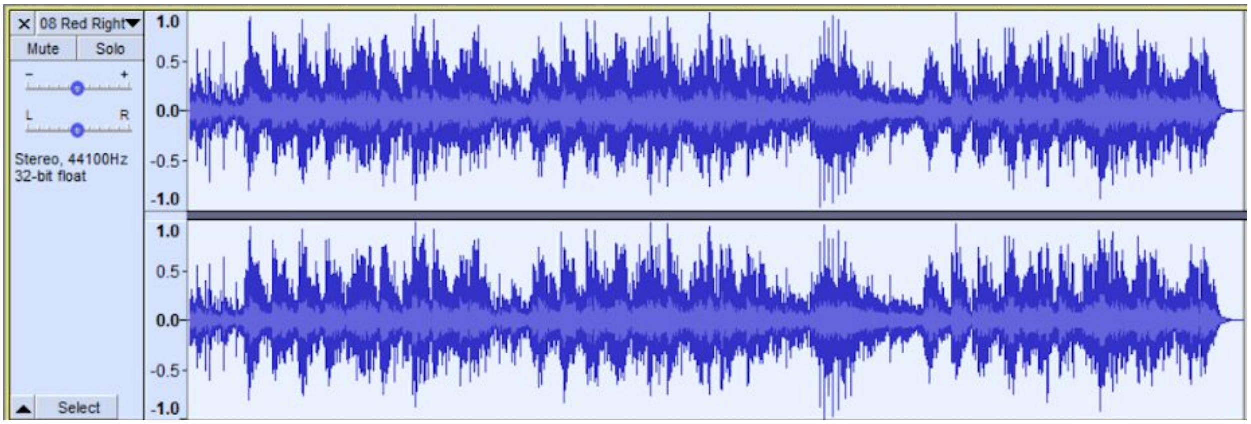


Fig 8. (Before) Anti - Karaoke Method

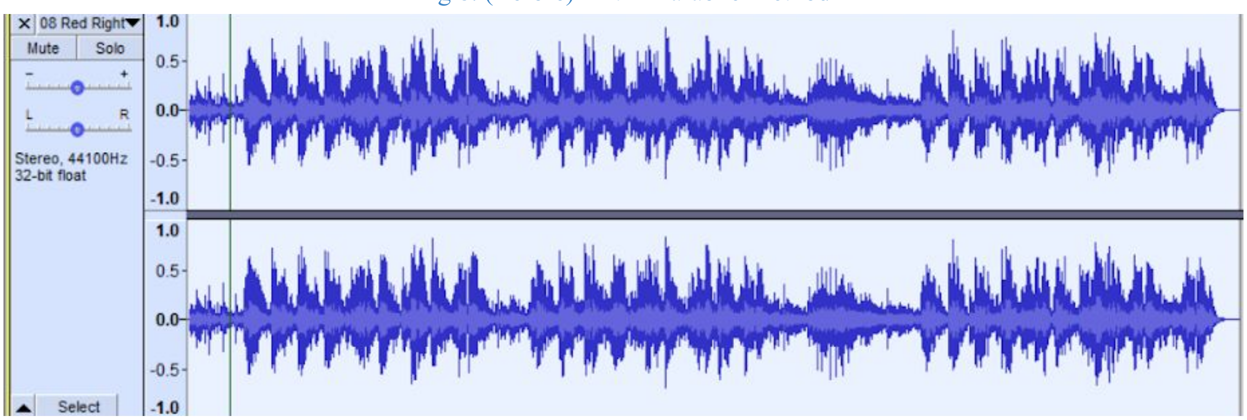


Fig 9. (After) Anti - Karaoke Method

3) The third method is to simply find notated music sheets and finding the highest and lowest vocal notes to label a song with the identified vocal range.

While we plan to use a combination of all methods, our personal library of music combined with the first method will help us populate a very accurate data set. However, the second method is uniquely formidable for its programmatic nature. However, there are some key hurdles we have already identified. Preemptively testing the second method, we tried two programs to isolate the vocals from the rest of the song. One was a built in function in Audacity and another was a pre-built python program [18]. The resultant from Audacity was of higher quality (qualitative assessment on our part) than from the python program. In this case, quality refers to noise levels, obscure noises, in the song as well as amount of instrumentation, and vocals that were left in the song after separation. The most compelling differentiation was that the python program seemed to have taken out a large portion of the vocal range.

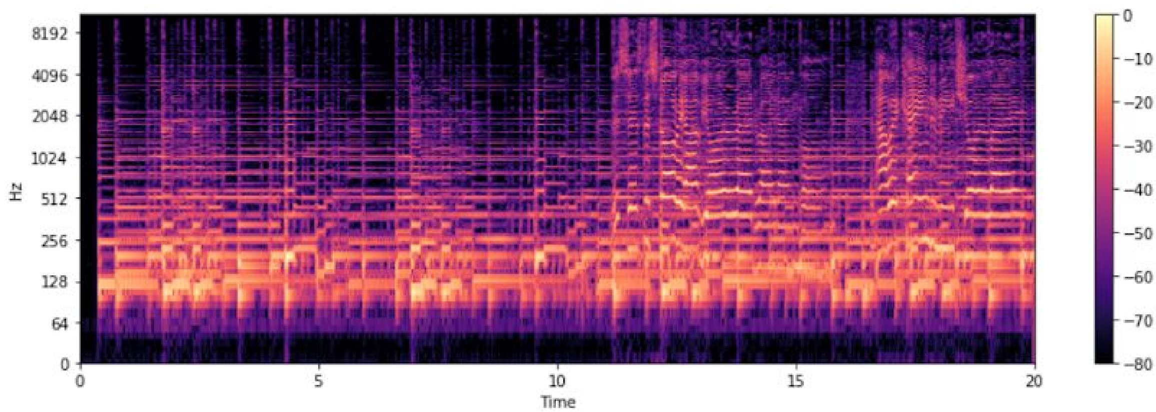


Fig 10. (Audacity "Anti-Karoake") Frequency (Hz) vs Time (seconds) Spectrogram

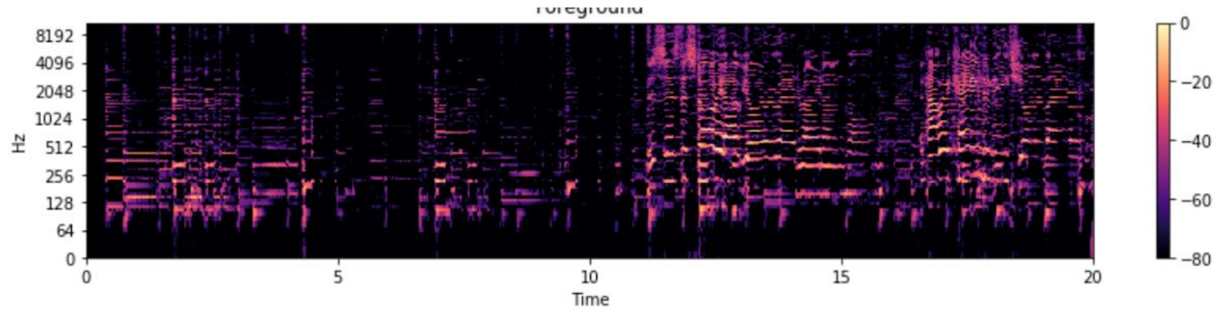


Fig 11. (Python "Anti-Karoake") Frequency (Hz) vs Time (seconds) Spectrogram

Building on this, the "anti-karaoke" programs are not perfect since they leave some instrumentation in their resultants. From song to song, the amount of instruments left can change. This reveals that there can be error even when creating the data set which can affect the ML algorithms ability to learn accurate information from the dataset.

2.4.2 Machine Learning

In order to ensure that at least 70% of our ML decisions are correct we will use the following formula to determine its accuracy:

$$(\text{Correct Decisions})/(\text{Total Decisions}) \times 100\% = \text{Success Rate}$$

A common worry when designing a classifier is overfitting. An overfit classifier would result in high accuracy when songs from the training data set are used for testing but much lower accuracy when tested with non-training data set songs(data the ML has never seen before). To ensure we do not design a classifier that overfits to the trained data, our testing songs will not overlap with songs used to train the algorithm.

Prior to choosing what characteristics our ML algorithm will use as features, we have to examine our data set manually with the following(below) feature extractions and data visualizations. Using these, we can choose from the following(possibly with modifications) based on how different they are for differing labelled data. Subsequently, the more features we choose the more complex the ML Algorithm will be. On the other hand, it may not be able to make accurate predictions with few features. Since the complexity directly relates to the latency of the ML algorithm, this is a potential concern. We can only understand what and how many features to use after diving into the data. However, we can explore another tangential concern. When a song is chosen to be played, the same features we trained the ML algorithm on must be extracted from the song. To understand the latency of each feature extraction and data visualization, we can explore an instance of each:

- Waveform of amplitude vs. time: Plotting waveforms of different songs combined with other data augmentation such as filtering frequencies, we may be able to come across unique traits for different vocal ranges. If the filtering methods are extensive, then we need to determine whether the latency of the method is worth the results. The time to plot this waveform for a 3 minute 30 second song was 0.0648 seconds.

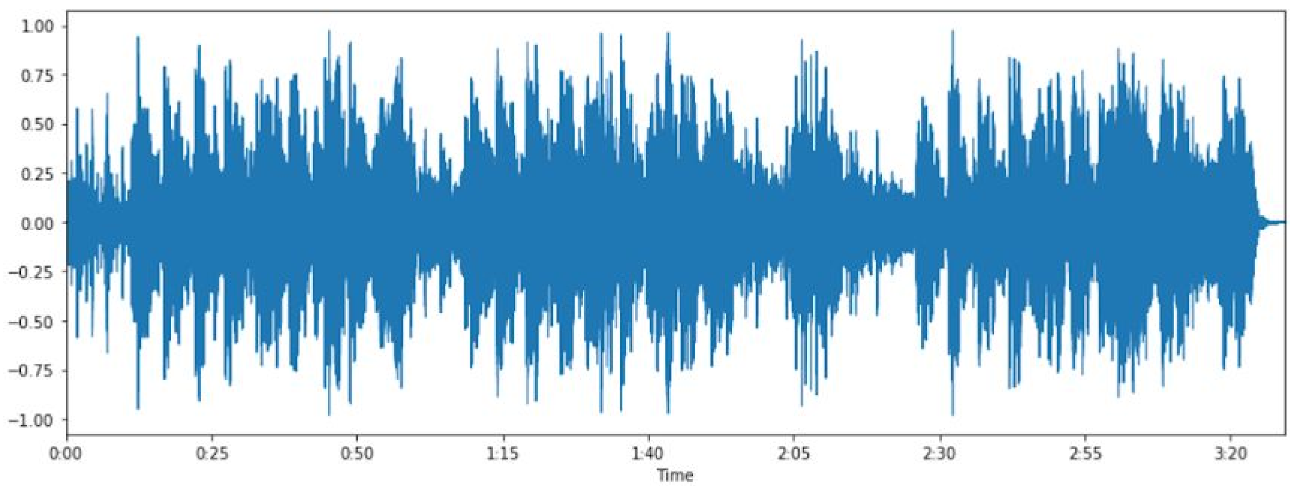


Fig 12. Amplitude vs Time Waveform Example

- Spectrogram of frequency vs. time: Much like the waveform, we can use spectrograms combined with other augmentations to identify unique traits in frequency response for different vocal ranges. The time to plot this spectrogram for a 3 minute 30 second song was 0.730 seconds

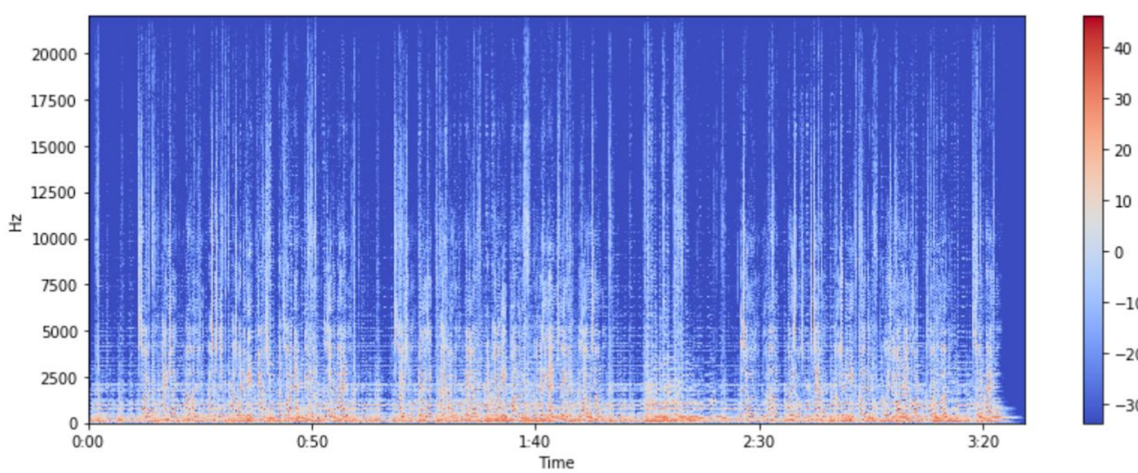


Fig 13. Frequency (Hz) vs Time (seconds) Spectrogram Example

- Zero crossing rate:

The Zero crossing rate is an important trait used extensively in speech recognition and music information retrieval. It calculates the number of times the rate of sign change of the audio signal. Hence, it may be valuable for us to explore. For our 3 minute 30 second song, it identifies 302259 zero crossings in 0.0718 seconds.

```
--- 0.07178044319152832 seconds ---
302259
```

Fig 14. Zero Crossing Rate Example

- Spectral Centroid:

The sectral centroid is the weighted mean of the frequency range of the song. For our 3 minute 30 second song, it took 0.8567 seconds to retrieve the spectral centroid.

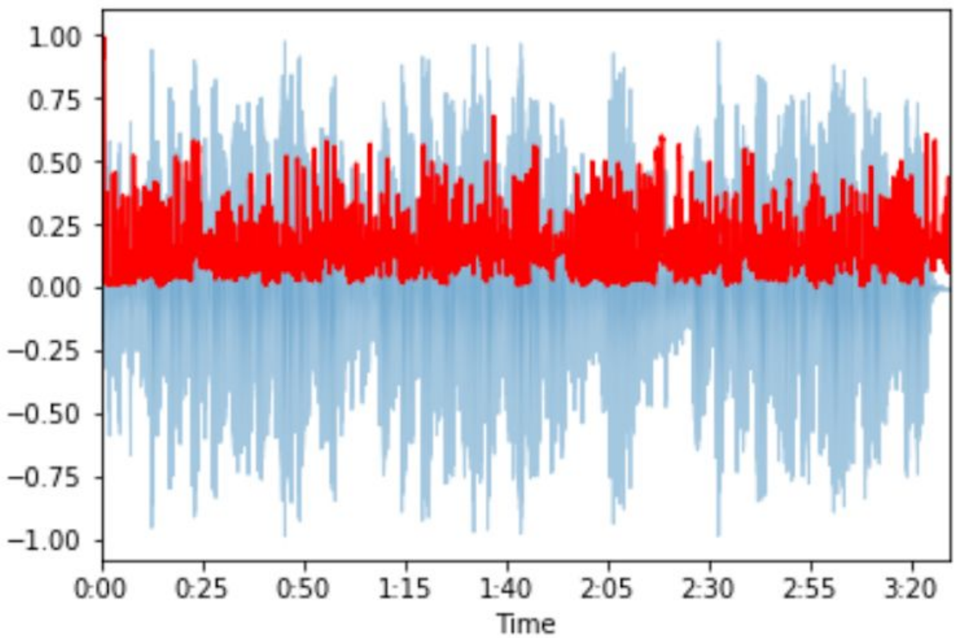


Fig 15. Spectral Centroid Example

- Spectral Rolloff:

The spectral rolloff represents the signal shape below a specific(given) percentage of total signal energy. This can be used to distinguish the rolloff frequency for each frame(plotted below) or over the whole signal. For our 3 minute 30 second song, it identifies 302259 zero crossings in 0.6931 seconds.

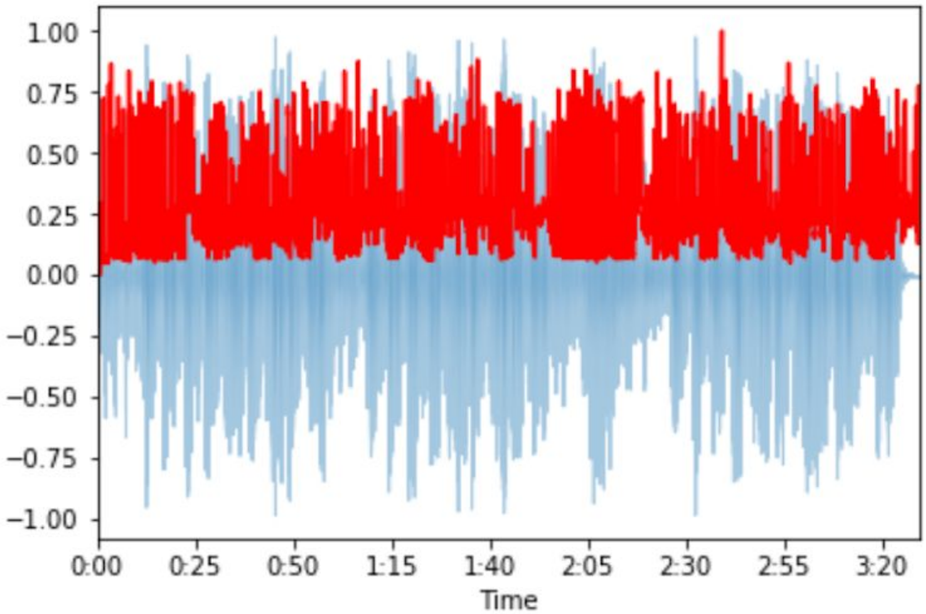


Fig 16. Spectral Rolloff Example

- Mel-frequency Cepstral Coefficients:

Mel-frequency Cepstral Coefficients or MFCCs are coefficients derived from the shape of the spectral envelope that represent a broader set of features that are usually used for characterizing human voice. For our 3 minute 30 second song, it identifies 302259 zero crossings in 0.481 seconds.

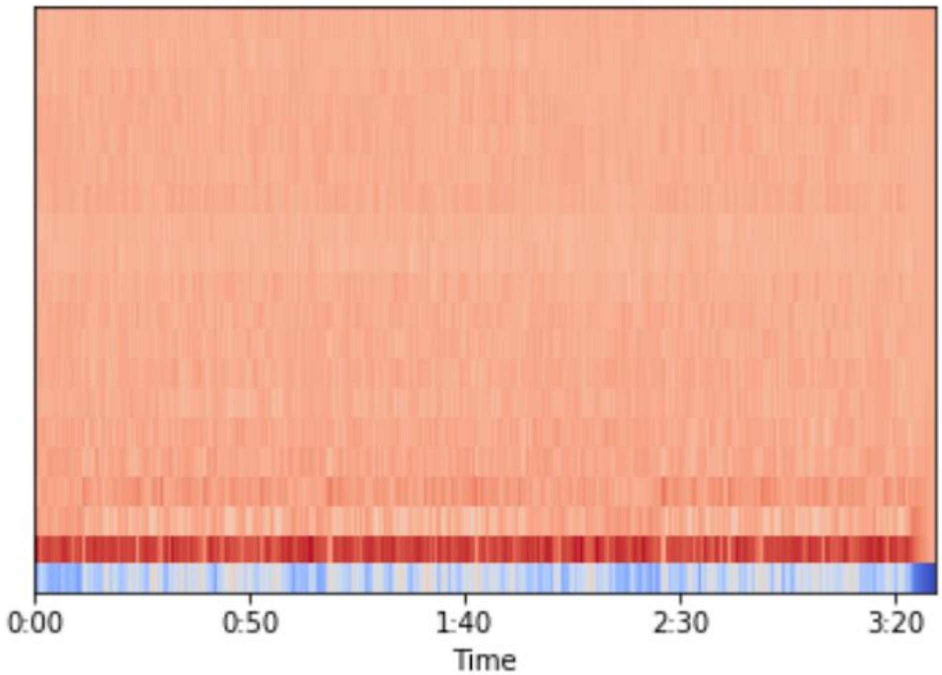


Fig 17. MFCC Example

If we identify certain features as more distinctive in identifying different vocal tones, we can utilize feature scaling to make these features more prominent (example below). However, this feature scaling increases complexity and time latency. For this 3 minute 30 second song, the increase in time was 0.0179 seconds.

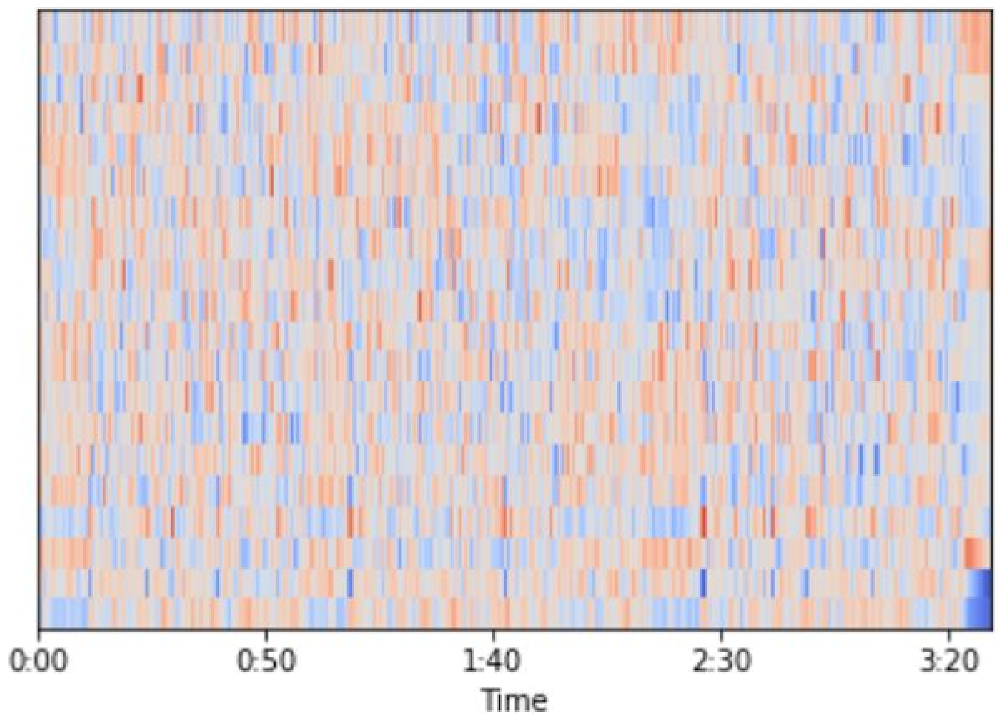


Fig 18. MFCC with feature scaling Example

- Chroma Frequencies:

Chroma Frequencies are specifically used for big data projects related to music. Dividing our frequency spectrum into the 12 semitones of the musical octave, we can use these features for training our ML algorithm. For the 3 minute 30 second song, it took 2.06 seconds.

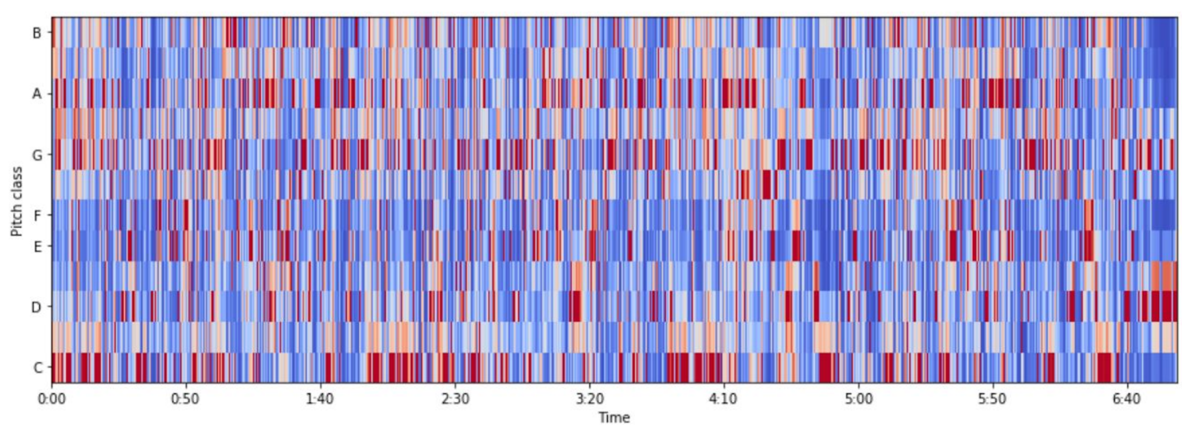
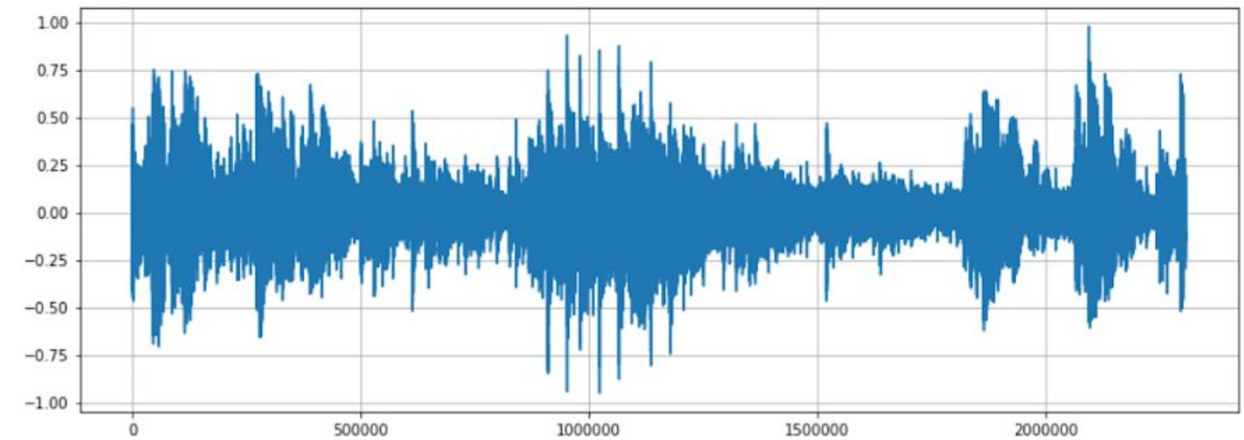


Fig 19.Chroma Frequency Example

Unfortunately, some of the time latencies exceed our time latency requirement for the whole control subsystem portion of the project. While there is the possibility that some of the above features may not prove useful, we need to devise an alternative for features that are useful. To counter these time latencies, we can try to extract the data from short clips of the song rather than the whole song. Below is an example of the waveform and zero cross rating of a 52.5 second clip of the previously used 3 minute 30 second song. With this clip, we take ¼ of the original song clip and identify the zero crossing rate. The time to derive the rate was 0.0239 seconds, roughly ~30% of the time needed to retrieve the zero crossing rate of the whole song.




```
--- 0.023920059204101562 seconds ---
65265
```

Fig 20. ¼ Song Zero Crossing Rate with respective waveform Example

Utilizing this method of analyzing clips rather than the entire song reduces time latency and may be accurate enough to derive relevant features. For the above example, if we multiply our 65256 zero crossings by 4, we get 261060 zero crossings as an estimate for the whole song. This is only a 13.6 % divergence from our true 302259 zero crossings which may be "close enough" for the ML algorithm to make a accurate assessment.

3 Cost and Schedule

3.1 Cost Analysis

Assuming that a competitive salary would be approximately $35/hr and that we will spend 12 hrs/week on this project individually, an approximated labor cost can be calculated as such:

2 People x 2.5 x $35/hr x 12 hrs/week x 16 weeks

= $33,600

The total cost for our labor for this project can be estimated to be $33,600. Comparing this the average value for an electrical engineering graduate at $67,000 and calculating a yearly salary for this project, we would get $108,387. This is greater than the typical starting salary, but given that we are the engineering, marketing and research team, we feel as though this is a fair salary.

Part	Cost
1-W FILTERLESS MONO CLASS-D AUDIO POWER AMPLIFIER	$.96
Adafruit I2S Stereo Decoder - UDA1334A Breakout	$6.95
Bluetooth 5.0 APTX Audio Module - TS8670	$12.95
Speaker - 3" Diameter - 8 Ohm 1 Watt	$1.76
Assorted Components. E.G. Resistors, Inductors, Capacitors, Transformers, Etc.	$30
PCB	$10

The total cost for this project would come to be $33,662.56.

3.2 Schedule

Week	Drew	Kshithij
9/30	Buy components	Buy components
10/7	Assemble and bench test power subsystem, Bluetooth subsystem, the functionality of operation amplifier, DAC and speaker	Find or Create Music Dataset w/ genre and sound clips
10/14	Circuit schematic and PCB Design (Order PCB)	Start Analyzing Dataset to find patterns; Liason with Musicians, Producers, etc.
10/21	Order PCB	Continue Analyzing Dataset to find patterns; Liason with Musicians, Producers, etc.
10/28	Create custom EQs for genres	[1]Build Rudimentary ML

	and test for errors in design (Order new PCB)	derived genre/characteristic detector
11/4	Test for errors in design. (Order new PCB)	Build iPhone/Computer app that can augment EQ
11/11	Add functionality to augment EQ based on file embedded genre	Add functionality to augment EQ based on file embedded genre
11/18	Miscellaneous / Fine-tune EQs	Add functionality to augment EQ based on ML Algorithm
11/25	Miscellaneous / Fine-tune EQs	Miscellaneous / Fine-tune ML Algorithm
12/2	Miscellaneous / Fine-tune EQs	Miscellaneous / Fine-tune ML Algorithm
12/9	Project Demo and Final Report	Project Demo and Final Report

Footnotes:

[1] Characteristics can differentiate types of genres. (i.e. high pitch vs low pitch singers)

4 Ethics and Safety

4.1 Concerns and 4.2 Mitigating Procedures

There are numerous potential safety hazards that we may face when executing this project.

Concern	Mitigating Procedure(s)
There are numerous potential safety hazards that we may face when executing this project. There is some risk involved with using wall power since the voltage coming from the wall is 120Vrms.	- [8] To ensure safe practice, someone else should always be present in the laboratory when utilizing wall power. - Accordingly, having a TA check the circuit prior to plugging in or powering the circuit, should be adhered to.

	- The one-hand rule can be useful in ensuring that the person working on the circuit is never the quickest path to ground.
[ACM 1.2] There is a potential fire hazard if we raise the decibel levels through volume or EQ such that clipping occurs.	- We should place hard upper limits on the decibel levels in the code for the equalization. - Listen for signs of clipping and act appropriately. - As well, in case of emergency, a protocol for the laboratory fire emergency must be followed.
Listening to loud music for long periods can damage eardrums.	- Use ear protection when nuanced listening is not imperative. - Limit long periods of music listening.

While the ethical concerns are limited in the scope of this project, there are some key ones to point out.

Concern	Mitigation
[13] Mismatching EQs with Songs or distorting music can cause fatigue or dizziness	- Do not create EQs purposefully to cause harm - Try not to pair songs with non-conforming EQs - Allow for plenty of breaks if the operator starts to feel fatigued or dizzy.

5 Citations

[1] We Listen to Music For More Than 4 1/2 Hours A Day, Nielsen Says. (2017, November 13). Retrieved September 9, 2019, from https://www.marketingcharts.com/industries/media-and-entertainment-81082.

[2] Releasing Stress Through the Power of Music. (n.d.). Retrieved September 30, 2019, from https://www.unr.edu/counseling/virtual-relaxation-room/releasing-stress-through-the-power-of-music.

[3] Kisla, R. (2017, August 1). Sound Signatures Explained. Retrieved September 30, 2019, from https://www.headphonezone.in/blogs/audiophile-guide/sound-signatures-explained.

[4] Newman, M. (2019, April 9). Country or Not? Lil Nas X's 'Old Town Road' Raises Big Questions About Genre in the Streaming Age. Retrieved September 30, 3n.d., from https://www.billboard.com/articles/business/8506428/lil-nas-x-old-town-road-splits-country-music-industry-opinion-genre-streaming-age.

[5] Hearing loss and music: MedlinePlus Medical Encyclopedia. (n.d.). Retrieved from https://medlineplus.gov/ency/patientinstructions/000495.htm

[6] Ethics Resources and Organizations. (n.d.). Retrieved from https://www.ieee.org/about/ethics/resources.html

[7]ACM Code of Ethics. (n.d.). Retrieved from https://www.acm.org/code-of-ethics

[8]IEEE Safety Recommendations For High Voltage Testing. (n.d.). Retrieved from https://www.spellmanhv.com/en/Technical-Resources/Articles/IEEE-Std-510-1983-IEEE-Recommended-Practices-for-Safety-in-High-Voltage-and-High-Power-Testing

[9] LABORATORY SAFETY TRAINING (n.d.). Retrieved from https://ovcrportal.research.illinois.edu/Training/Overview.aspx?TrainingId=67

[10] Adafruit Industries. (n.d.). Adafruit I2S Stereo Decoder - UDA1334A Breakout. Retrieved September 30, 2019, from https://www.adafruit.com/product/3678?gclid=EAIaIQobChMIj-LPpff35AIVUL7ACh0xwwx3EAAYASAAEgJ_WvD_BwE

[11] TPA2001D1PWR. (n.d.). Retrieved September 30, 2019, from https://www.ti.com/store/ti/en/p/product/?p=TPA2001D1PWR&utm_source=google&utm_medium=cpc&utm_campaign=ecm-tistore-invf-GMC_US-cpc-store-google-wwe&utm_con

tent=GMC&ds_k=PRODUCT_GROUP&DCM=yes&gclid=EAIaIQobChMI3JHK6rz45AIVxsDACh2SdAYrEAQYBSABEgLGKvD_BwE&gclsrc=aw.ds.

[12] Services, E. I. T. S. (n.d.). Salary Averages. Retrieved September 30, 2019, from https://ece.illinois.edu/admissions/why-ece/salary-averages.asp.

[13] Why Do Some Sounds Make People Sick? (2015, November 2). Retrieved from http://mentalfloss.com/article/69578/why-do-some-sounds-make-people-sick.

[14] Brownlee, J. (2019, May 22). How Much Training Data is Required for Machine Learning? Retrieved October 5, 2019, from https://machinelearningmastery.com/much-training-data-required-machine-learning/

[15] Bahuleyan, Hareesh. (2018). Music Genre Classification using Machine Learning Techniques. Retrieved October 5, 2019, from https://www.researchgate.net/publication/324218667_Music_Genre_Classification_using_Machine_Learning_Techniques/

[16] Bladen, S. (2019, July 11). How our vessel tracking map works. Retrieved from https://globalfishingwatch.org/map-and-data/technology/

[17] Haughey, C. J. (2019, June 27). 6 Complete Machine Learning Projects. Retrieved from https://www.springboard.com/blog/machine-learning-projects/

[18] Zafar Rafii and Bryan Pardo. "Music/Voice Separation using the Similarity Matrix," ***13th International Society for Music Information Retrieval*****, Porto, Portugal, October 8-12, 2012.**