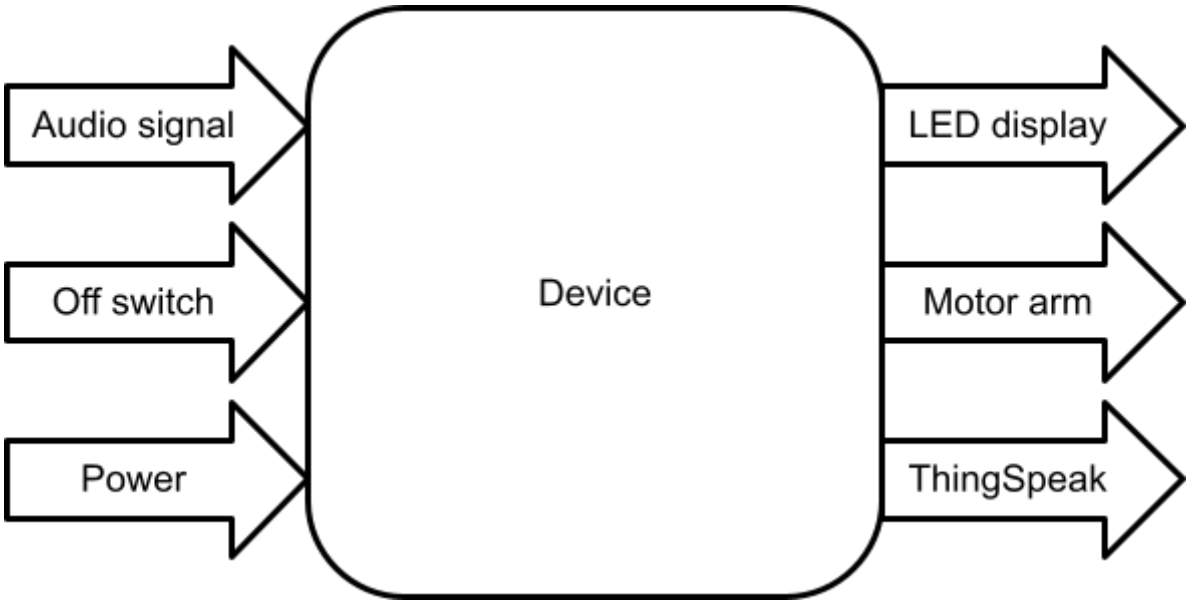
ECEG 201 Application Datasheet

Guitar Tuner

Izzy Philosophe

1. Features

- Tuning capabilities for an electric guitar
- Fundamental frequency detection and reporting
- Dynamic display
    - RGB LED ring
    - Motor arm indicator
- Off switch
- 3.5mm audio jack for audio input



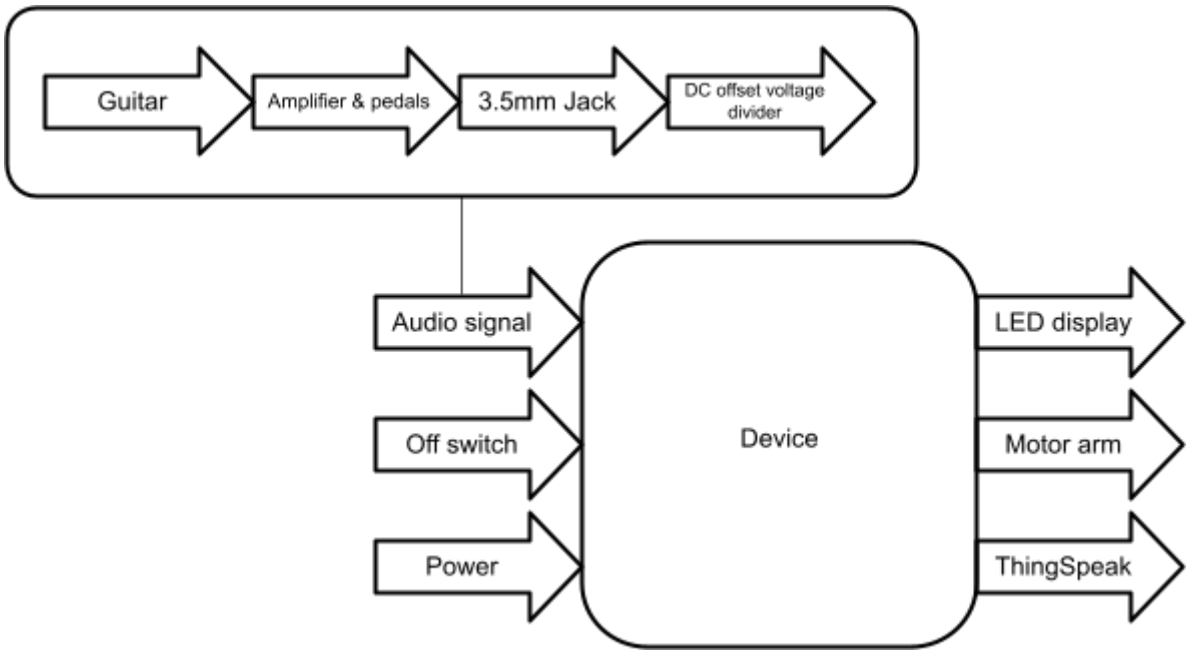
2. Applications

- Music production studios
- Band practices
- Home music setups
- Any setup with power and an audio signal

3. Description

This application for the DAMNED project is a guitar tuner. It takes an audio signal from a guitar as an input, processes the signal, and detects the fundamental frequency of whatever note is being played. It then determines which open-string note the played note is closest to and displays how sharp or flat the played note is compared to this target note. The tuner also displays on the bottom which of the 6 strings it thinks is the target note.
The Tuner also includes an off switch that upon startup the program waits for it to be turned off. Then when in use, if the switch is turned on, the program stops and sends the time it took to finish tuning to a thingspeak channel. The program also periodically sends the detected frequency to the same thingspeak channel, every 20 frequency samples.

4. Inputs and Outputs



- Inputs:
    - 3.5mm audio jack for audio signal input [1]
        - SJ1-352XN 3.5 MM AUDIO JACK
        - Takes in the audio signal produced from the guitar and amplifier and outputs it to a DC offset voltage divider, then feeds that into the Feather
    - Off switch
        - A potentiometer that outputs a voltage between 0V (GND) and 3.3V (3V3 VCC)
    - Power [2]
        - The system is powered by a 12V wall wart power supply that is then converted to a steady 5V, 0.5-1.0A output by the device's internal power supply unit
- Outputs:
    - LED display [3]
        - NeoPixel Ring - 24 x 5050 RGB LED with Integrated Drivers
    - Motor arm [4]
        - 28BYJ-48 stepper motor
    - ThingSpeak [5]
        - Uses Adafruit AirLift FeatherWing – ESP32 WiFi Co-Processor
        - Connects to WiFi and sends output data to a ThingSpeak channel that has 2 fields; one for frequency data and one for total tuning time data

5. Maximum Ratings

- 3.5mm audio jack [1]:

parameter	conditions/description	min	typ	max	units
rated input voltage			12		Vdc
rated input current				1	A
contact resistance	between terminal and mating plug between terminal in a closed circuit*			50 30	mΩ mΩ
insulation resistance	at 500 Vdc	100			MΩ
voltage withstand	at 50/60 Hz for 1 minute			500	Vac
insertion/withdrawal force		0.3		3	kg
terminal strength	any direction for 10 seconds			500	g
operating temperature		-25		85	°C
life			5,000		cycles
flammability rating	UL94V-0				
RoHS	yes				

Notes: *When measured at a current of less than 100 mA / 1 kHz

- Power Supply [6]:

over operating free-air temperature range (unless otherwise noted) (1)(2)(3)

	MIN	MAX	UNIT
VIN, RON to GND	–0.3	25	V
EN, FB, SS to GND	–0.3	7	V
Junction Temperature		150	°C
Peak Reflow Case Temperature (30 sec)		245	°C
Storage Temperature, T_{stg}	–65	150	°C

(1) Stresses beyond those listed under *Absolute Maximum Ratings* may cause permanent damage to the device. These are stress ratings only, which do not imply functional operation of the device at these or any other conditions beyond those indicated under *Recommended Operating Conditions*. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.
(2) If Military/Aerospace specified devices are required, please contact the Texas Instruments Sales Office/ Distributors for availability and specifications.
(3) For soldering specifications, refer to the following document: SNOA549

- Feather and code [7]:

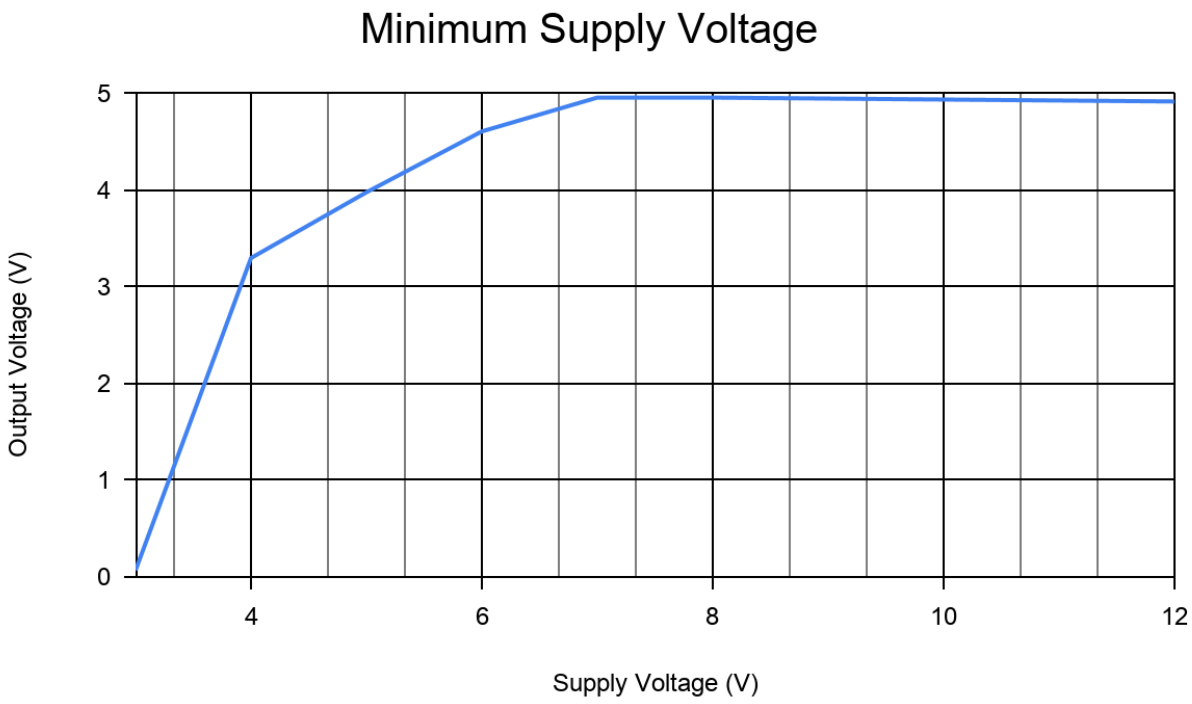
	MIN	MAX	UNIT
A0, A1 to GND	0.0	3.3	V
Detectable frequency	71.326	380.836	Hz

6. Electrical Characteristics

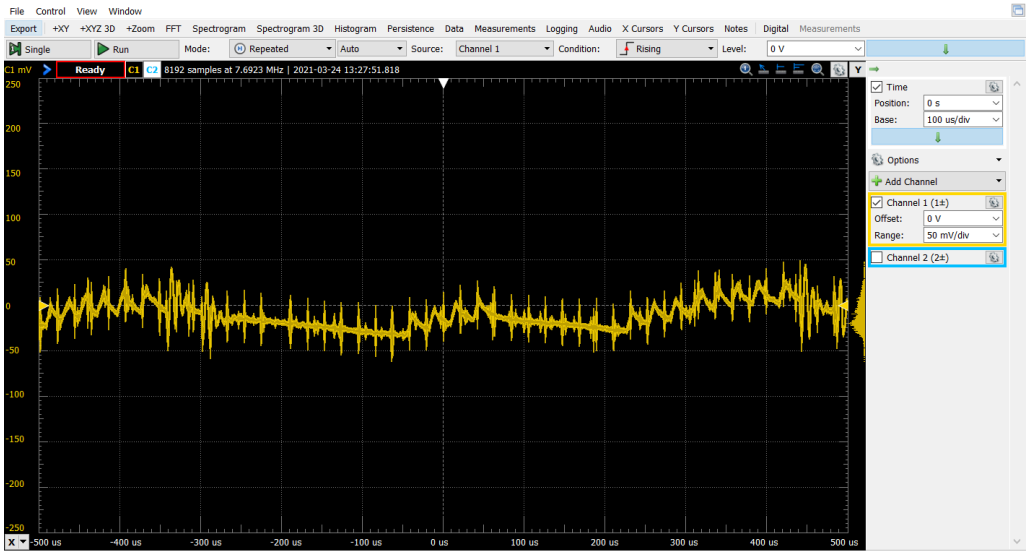
6.1 Measured Characteristics

[This is where you list or provide graphs of characteristics you measured.]

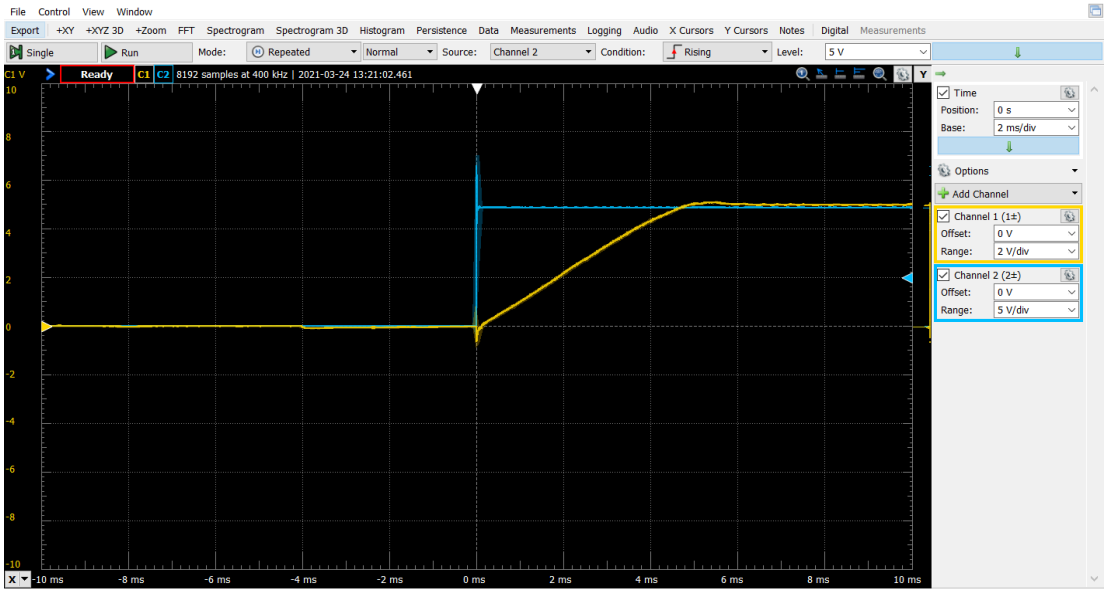
- Minimum Supply Voltage that will still give an output of 5V [4]



- Power Supply Ripple [8]



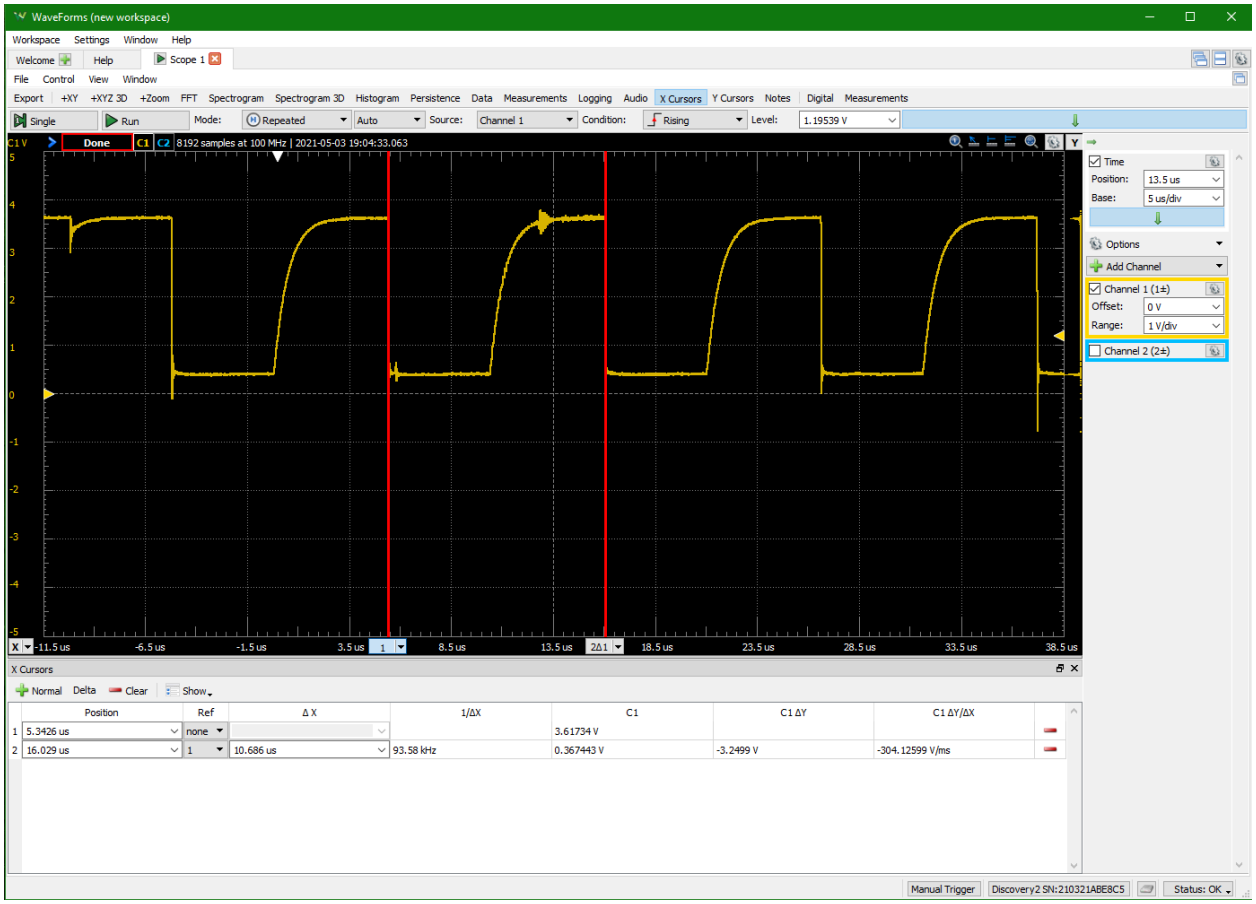
- Power Supply Turn-On Time [8]



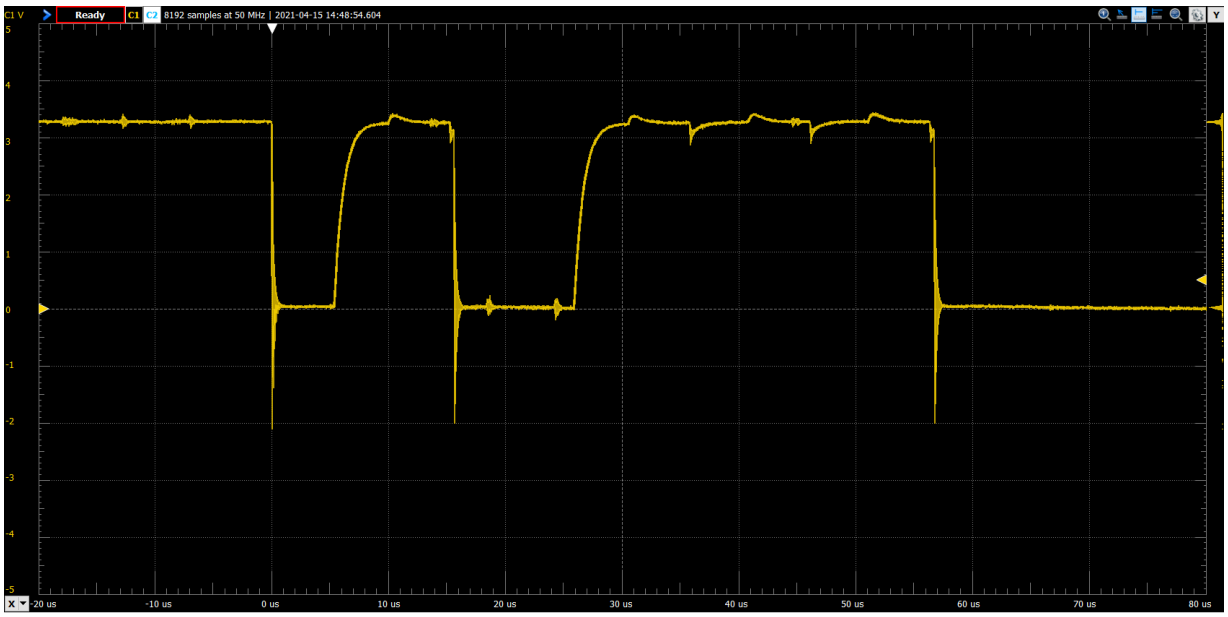
- Stepper Motor Terminal Output Measurements in response to commands [8]

Command	Term. 1 (V)	Term. 2 (V)	Term. 3 (V)	Term. 4 (V)
h2A h00 h10 h00 h00	0	4.98	1.03	1.07
h32 h00 h10 h00 h00	.82	.81	0	4.98
h2E h00 h10 h00 h00	4.98	0	1.28	1.01
h36 h00 h10 h00 h00	.91	.90	4.98	0

- SCL Output Capture [9]



- SDA Output Capture [9]



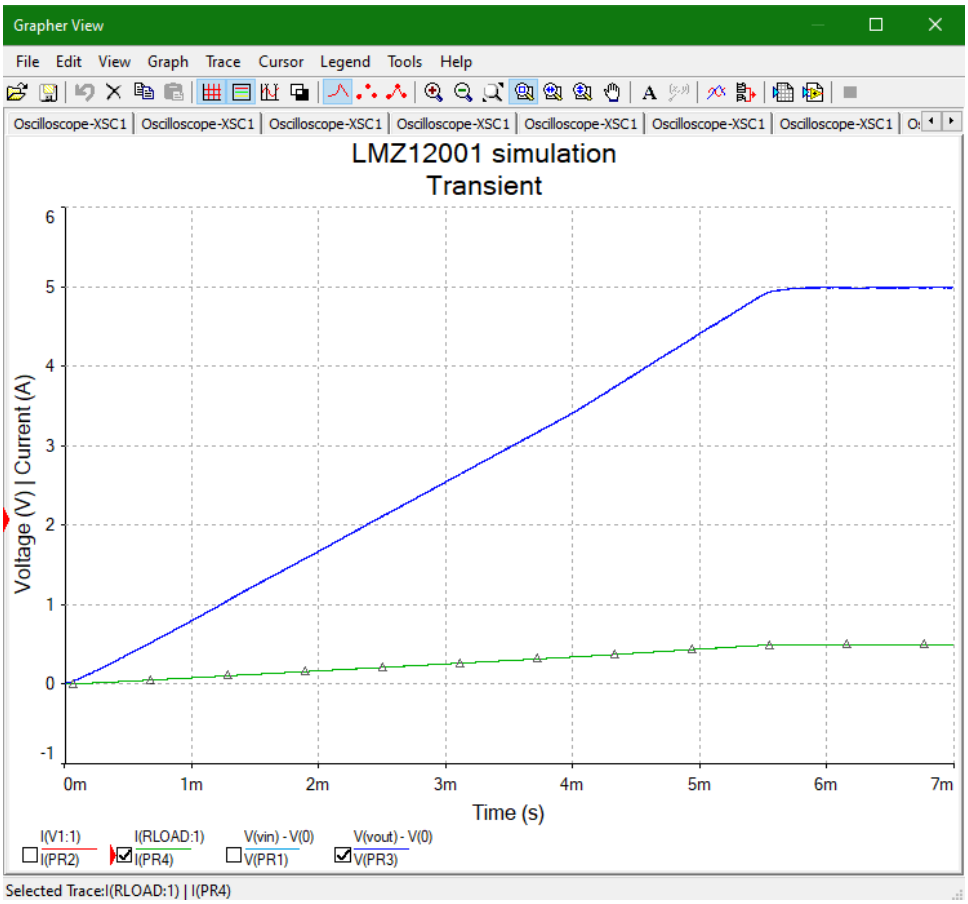
6.2 Modeled Characteristics

- Voltage & Current ins and outs and power efficiency for the power supply [2]

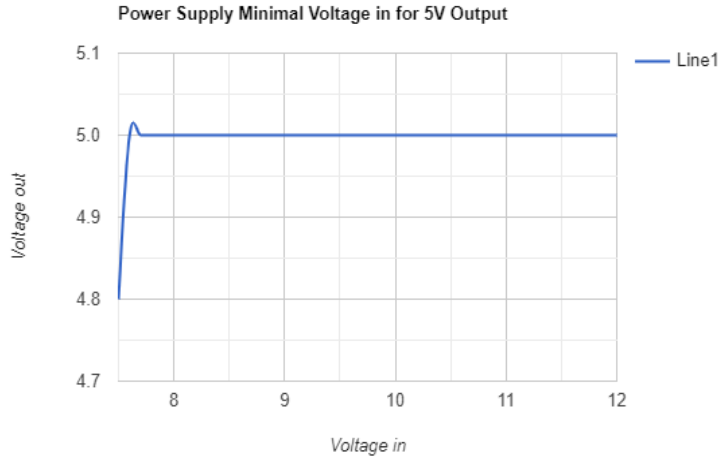
R-LOAD (ohms)	V-IN (V)	I-IN (mA)	V-OUT (V)	I-OUT (mA)	P-IN (W)	P-OUT (W)	P-EFF (%)
1	12	-273	1.6	1600	3.276	2.56	78.14407814
3	12	-657	4.52	1510	7.884	6.8252	86.5702689
10	12	-235	5	500	2.82	2.5	88.65248227
30	12	-80	5	167	0.96	0.835	86.97916667
100	12	-26.5	5	50	0.318	0.25	78.6163522

R-LOAD (ohms)	V-IN (V)	I-IN (mA)	V-OUT (V)	I-OUT (mA)	P-IN (W)	P-OUT (W)	P-EFF (%)
10	9	-285	4.83	483	2.565	2.33289	90.95087719
10	9.3	-276	4.88	488	2.5668	2.38144	92.77855696
10	9.4	-285	4.95	495	2.679	2.45025	91.46136618
10	9.5	-288	5	500	2.736	2.5	91.37426901
10	12	-235	5	500	2.82	2.5	88.65248227

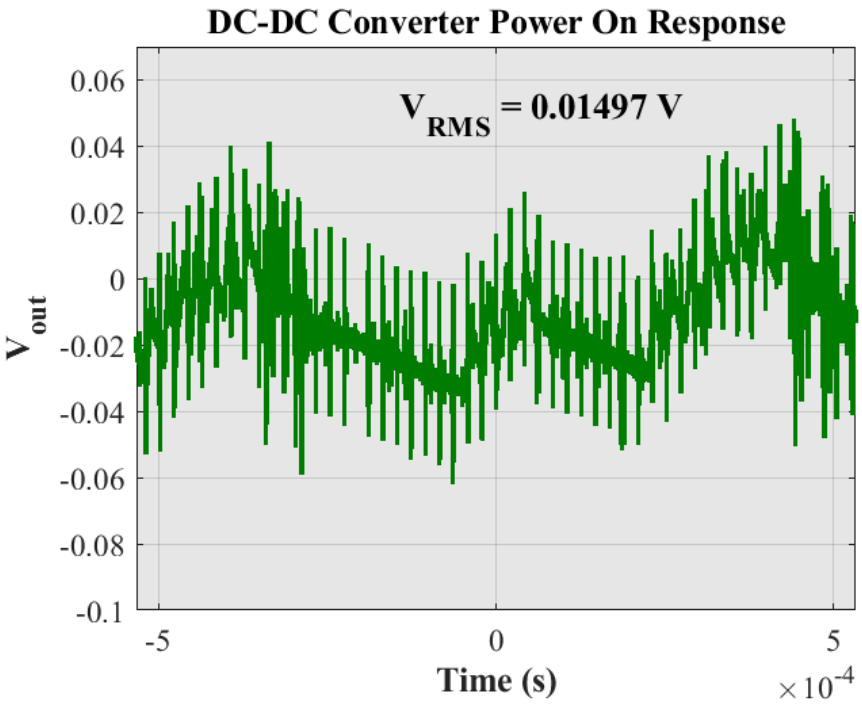
- Power Supply Turn-On Time [2]



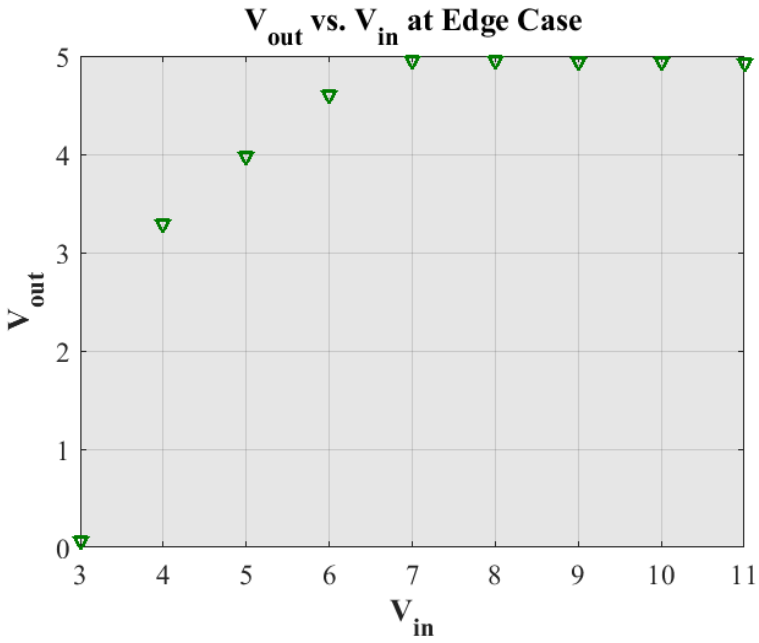
- Power Supply Minimal Voltage in for 5V Output [8]



- Power Supply Ripple [8]

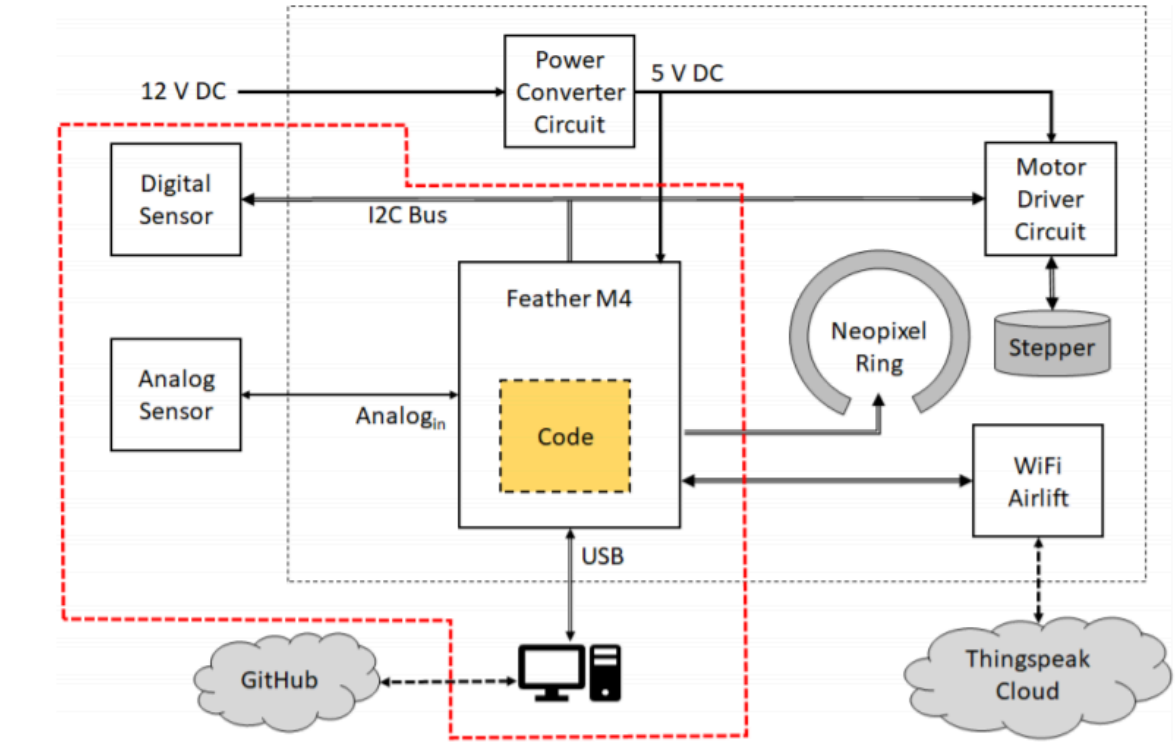


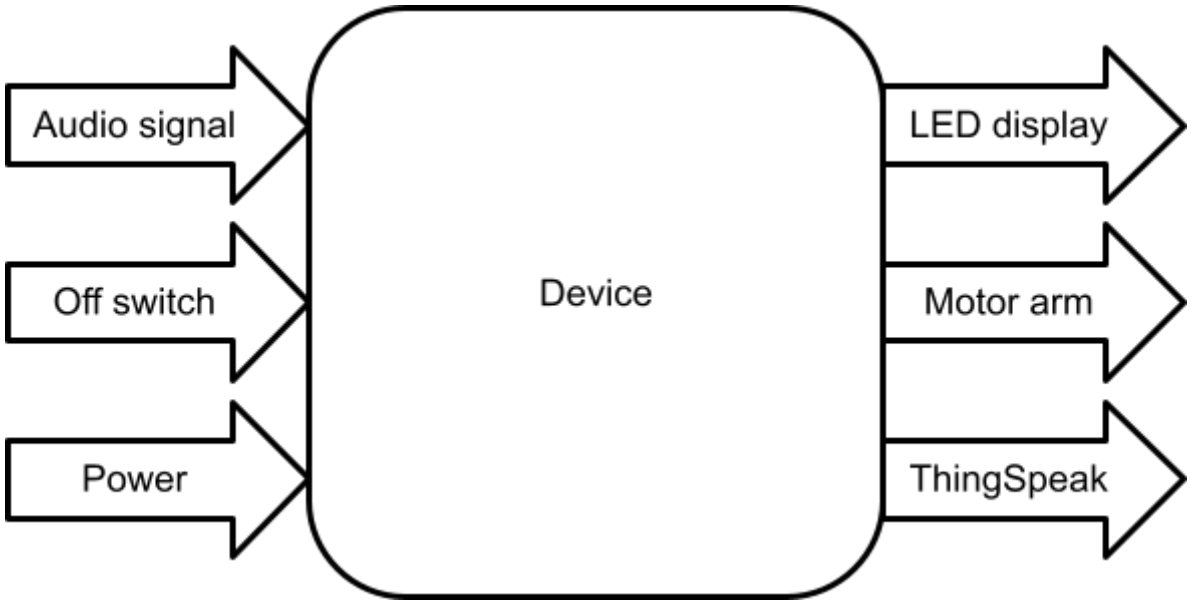
- Voltage in vs Voltage out at edge case [8]



7. Detailed Description

7.1 Functional Block Diagram





The tuner application uses two analog sensors that go into the $Analog_{in}$ ports of the Feather M4. This data is then processed in the code and then outputted to the Neopixel Ring and Stepper Motor through the I^2C Bus, and also sent through the WiFi Airlift module over the Internet to the ThingSpeak cloud channel.

7.2 Feature Description

- DAMNED platform features:
    - Feather M4 microcontroller [7]
        - Digital and analog inputs and outputs
        - 3.3V supply
        - USB to connect to a computer and upload/download code files
    - I^2C connection
        - The Feather provides an I^2C connection for digital sensors and the motor controller
    - Neopixel RGB LED ring [3]
        - A ring of 24 individually addressable RGB LEDs used as part of the main display
    - 28BYJ-48 Stepper motor [4]
        - A stepper motor that drives the dial arm as part of the main display
    - Adafruit AirLift FeatherWing – ESP32 WiFi Co-Processor [5]
        - WiFi module for the Feather M4
        - Allows the Feather to connect to the internet independently from a computer in order to access APIs and online data.

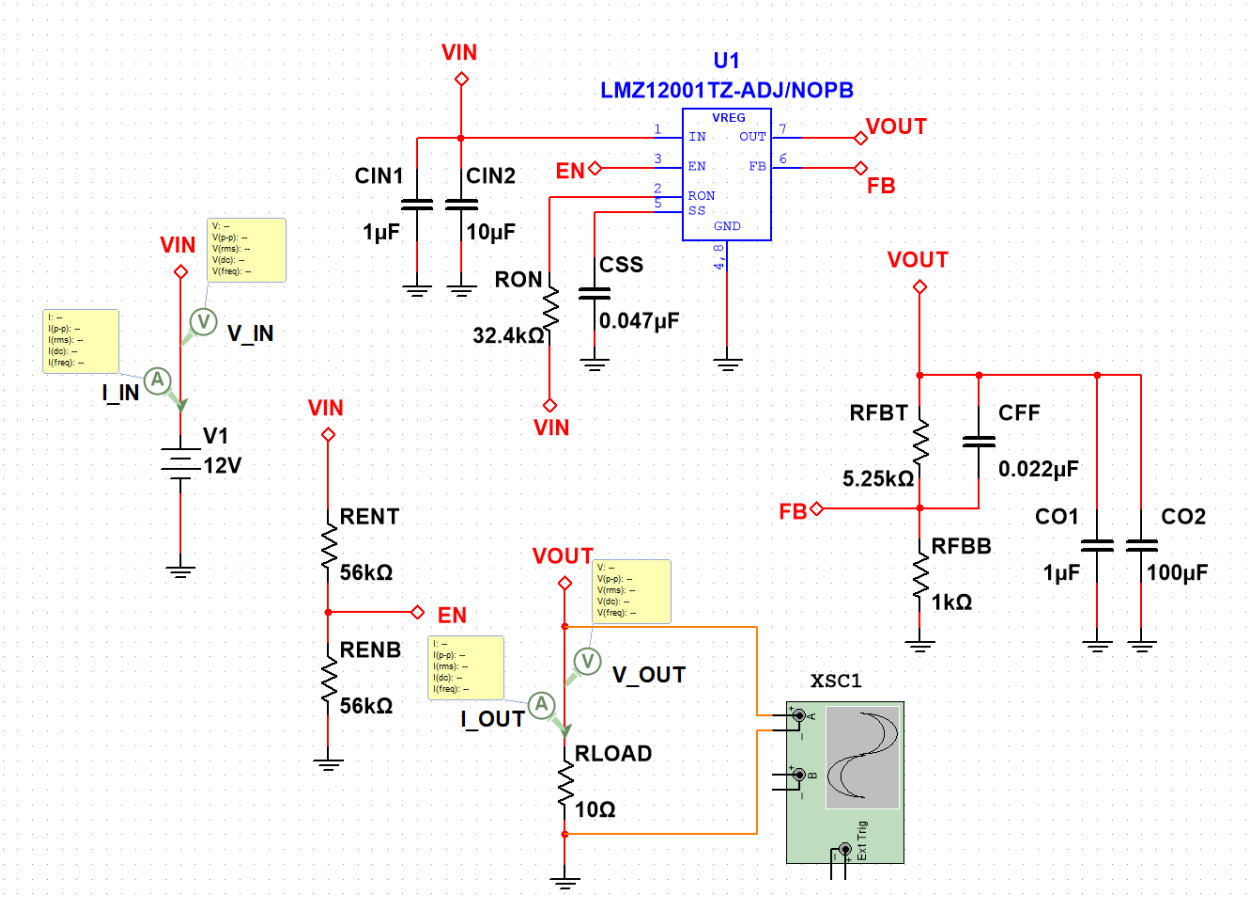
- Application features:
    - Tuning capabilities for an electric guitar
        - This application is able to detect the pitch played by a guitar, determine the string it's being played with, and tell you how sharp or flat you are from the open string.
    - Fundamental frequency detection and reporting [10]
        - The device records a small sample of the audio signal and uses Adafruit's ulab API to process the signal and determine the fundamental frequency of the inputted audio signal, then report back that frequency it detected.
    - Dynamic display
        - RGB LED ring [3]
            - The LEDs are used to display the tuning meter boundaries, the detected target note, and the overall status of the (waiting to start, active, or finished execution)
        - Motor arm indicator [4]

            - The motor arm is used to display on the tuning meter how sharp or flat the perceived note is compared to the target note.
    - Off switch
        - The device uses an off switch to tell it when the user is done tuning their guitar, and when activated the program stops execution
    - 3.5mm audio jack for audio input [1]
        - The device uses a 3.5mm audio jack as the input for the audio signal. This can be used in junction with a guitar and amp, or alternatively you could use this to stream any audio signal from any source to the device.
    - ThingSpeak usage
        - The device pushes two forms of data to a thingspeak channel. First, it pushes the detected frequency to field 1 every 20 samples. Second, once the off switch is activated, the device determines how long it took for the user to finish tuning their guitar and sends this information to field 2

7.2.1 Power System [2] [6]

The power supply unit built into the device takes in an unstable 9.5-20V supply, ideally 12V DC, and converts it, using a LMZ12001, to a steady 5V, max 1A, min 0.5A supply to safely and consistently power the Feather M4 and motor controller. With this setup, we are able to power the device with a simple 12V barrel jack power supply instead of needing something like a USB connection to power everything.

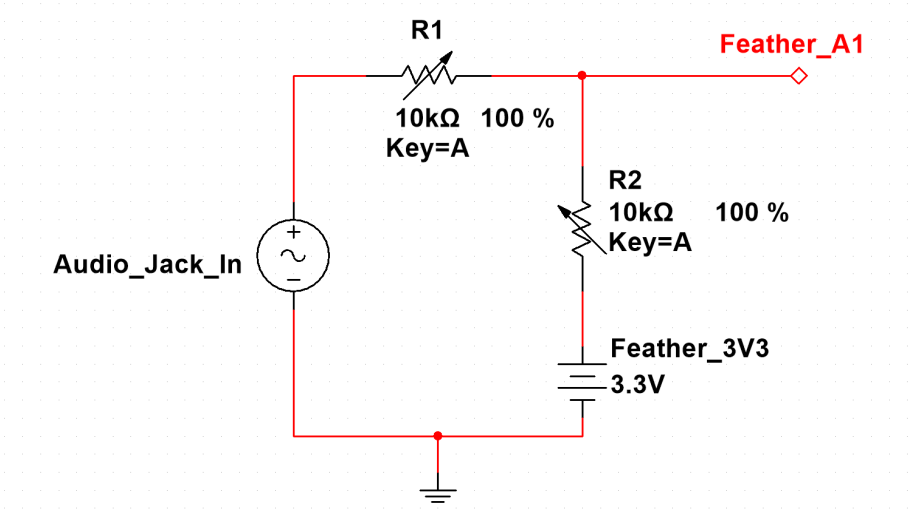
Here is an overview diagram of the power supply unit:



7.2.2 Sensors

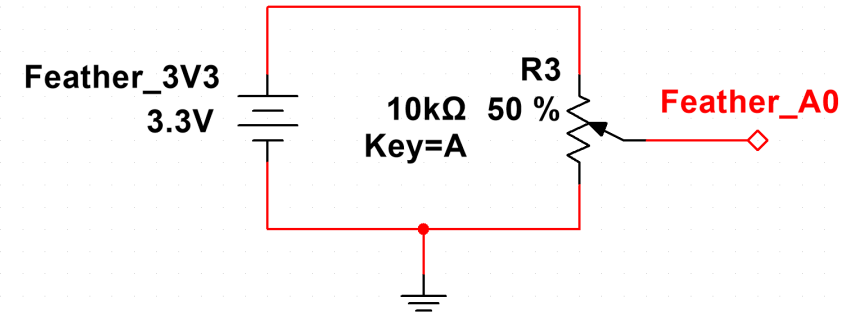
- Audio signal input [1] [7]:
    - The audio signal is generated from the guitar and goes through an amplifier and optionally some effect pedals, then from the amp to the 3.5mm audio jack connected to the device. The audio signal is at this point in the form of an AC voltage signal centered around 0V, but the Feather's analog pins can only read values between 0V and 3.3V, meaning it can't register negative voltages. So the output of the 3.5mm audio jack is then fed through a voltage divider circuit in order to give the audio signal a DC offset and have it fluctuate between 0V and 3.3V so that the Feather can properly read the signal.

      Here is a diagram of the voltage divider circuit:



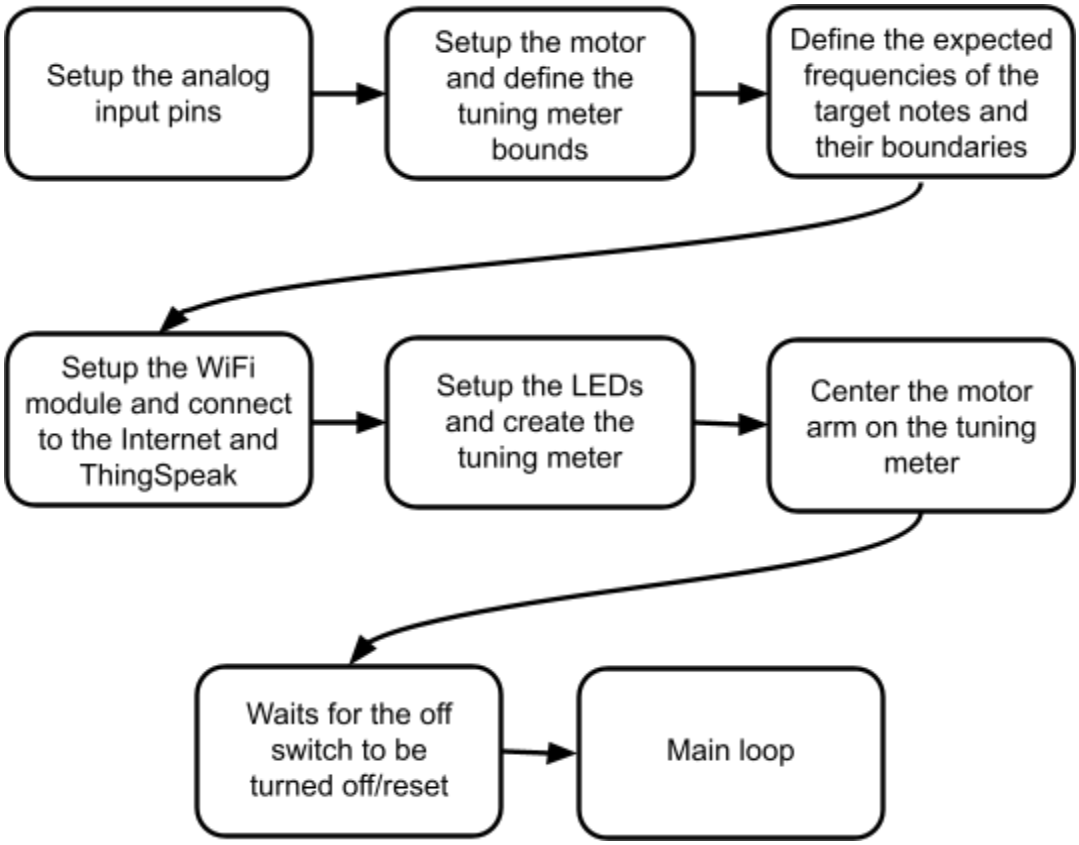
- Off switch
    - The off switch is made by a simple 10kΩ potentiometer that acts as a voltage divider between the Feather's 3V3 and GND to produce an output voltage between 0V and 3.3V.

      Here is a diagram of the off switch circuit:

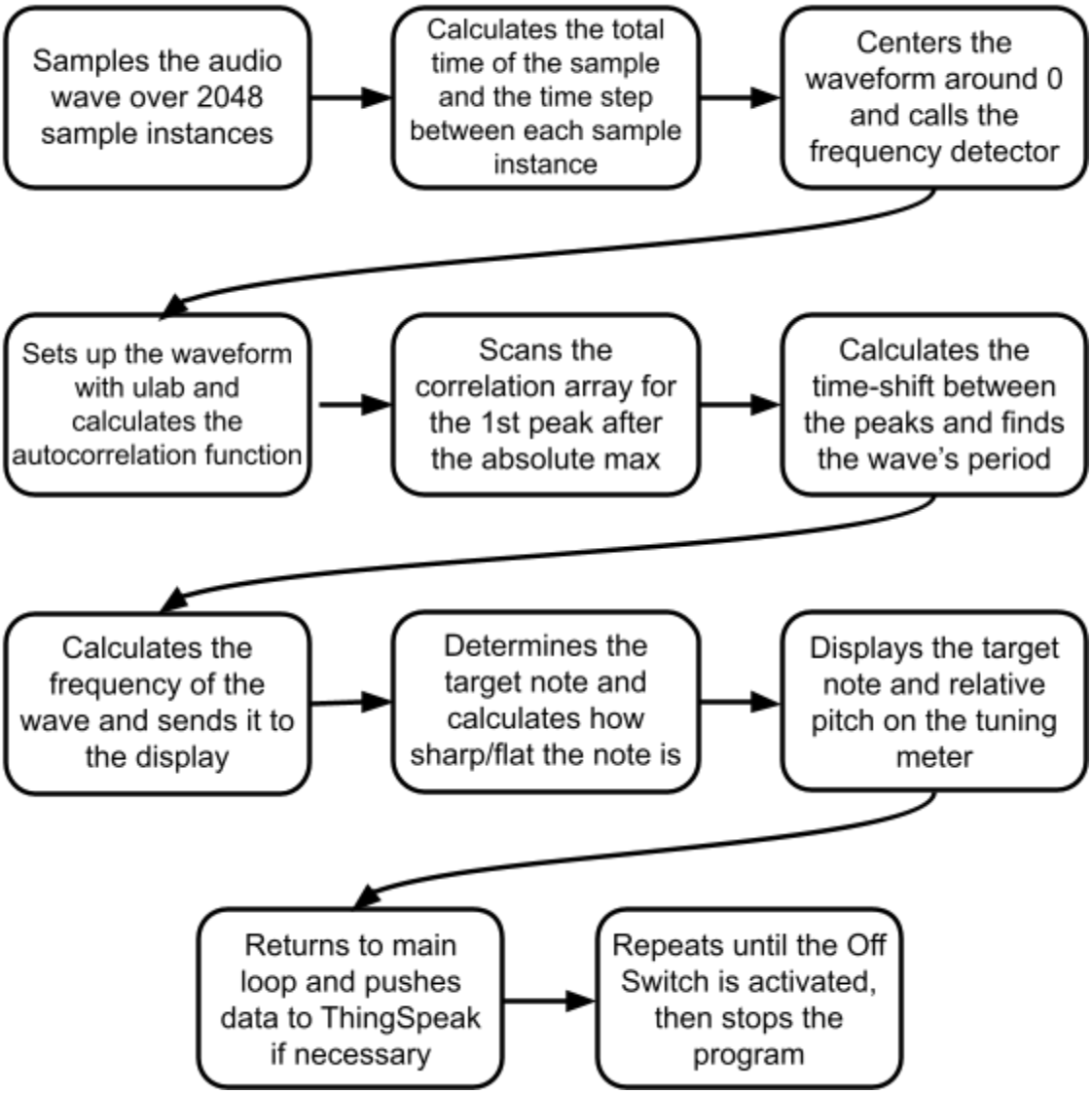


7.2.3 Software and Processor

Initialization:



Main loop [10] [11] [12] [13] [14]:



Needed files and libraries:

- neopixelFunctions.py
    - neopixel
- espFunctions.py
    - Digitalio
    - Busio
    - Adafruit_requests
    - Adafruit_esp32spi
- motorFunctions.py
    - Adafruit_motorkit
    - Adafruit_motor

- time
- board
- analogio
- math
- Ulab

Github repository with all needed .py files here

7.2.4 Web Connectivity [5]

The device connects to the internet via the AirLift FeatherWing ESP32 WiFi module connected to the feather. Once connected, it then connects to a thingspeak server where it pushes and dumps some data. During execution of the program the device will occasionally send the detected frequency through the WiFi module to the field 1 of the thingspeak server. When the program is finished executing, it then computes how long it took overall for the user to tune their guitar, and then sends this data to field 2 of the thingspeak server.
In the code you are able to set the SSID and Password of your local WiFi network, as well as the Thingspeak channel ID and API key

7.2.5 Display

7.2.5.1 Lights [3]

The Neopixel LED ring contains 24 individually addressable RGB LEDs that you can change the color of by inputting a set of three values between 0 and 255, each representing the strength of the corresponding color (R, G, B).
You can also adjust the brightness of the LEDS either through adjusting the individual LEDs overall color level or by changing the value of a variable to change the entire ring's brightness.

7.2.5.2 Dial [4]

The stepper motor is controlled by a motor driver controller integrated in the PCB. The driver controlled via an I^2C communication protocol connecting to the onboard Feather M4
You are able to control the position of the motor in a number of ways. For example, either by inputting a specific degree you want it to go to, or telling it to move a certain number of 'steps' in either direction.
The motor is calibrated at the beginning of the program by making a full counterclockwise rotation and stopping itself on the peg located on the bottom left of the dial. That spot is considered '0' or home by the motor.

8. Application and Implementation

8.1 Applications

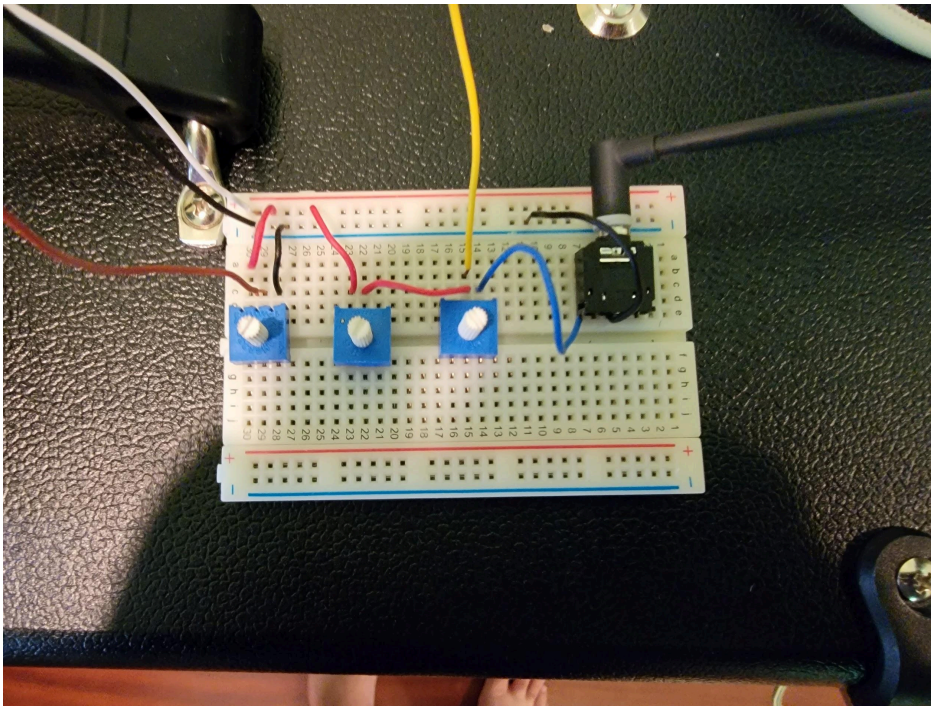
There are many applications for the DAMNED platform. You can use it to detect the weather with temperature and humidity, you can keep track of your stocks, even use it as a clock. As previously stated, my application is a guitar tuner. As a good example of how you could use this application, I have it set up in my bedroom with my guitar setup:

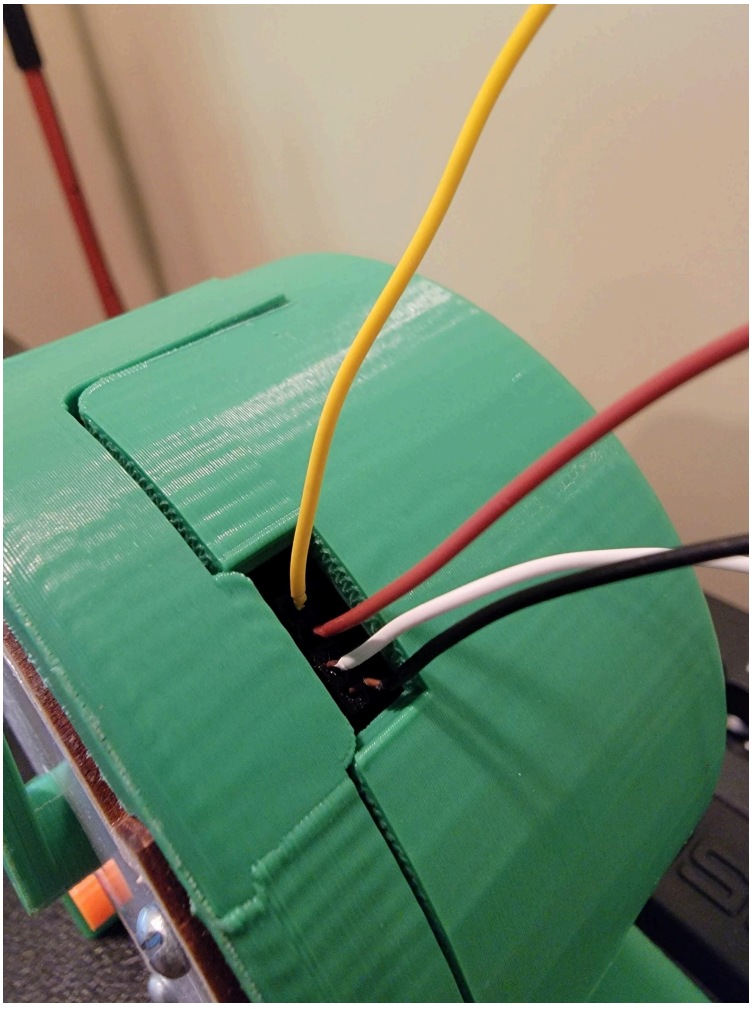
8.2 Setup

The first thing you need to do to set up the device is to go into the code file and configure your WiFi network and ThingSpeak channel settings by modifying a few parameters shown here:

```
226 # Sets up the esp and connects to the thingspeak channel
227 ssid = "[redacted]"
228 password = "[redacted]"
229 channel_ID = [redacted]
230 thingspeak_api_key = '[redacted]'
```

Next you want to set up the device with the headphone jack and voltage divider (diagram above) and then plug in both power and the audio signal as shown here:

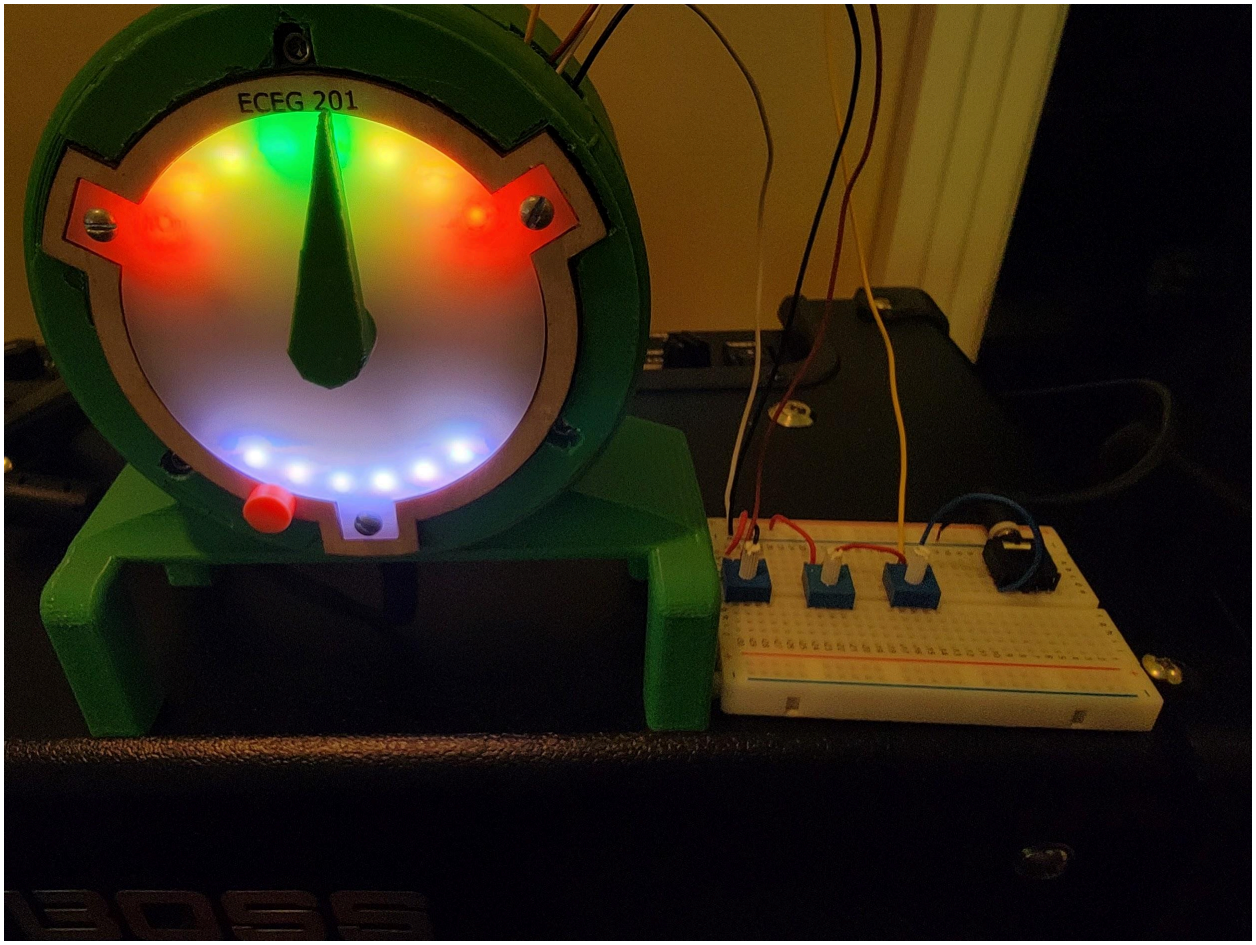




Once you plug those four wires into their designated pins as shown, setup is complete and you can start the device up.

8.3 Operation

On startup, when the bottom LEDS light up red, that indicates that you need to reset the Off switch before the program can continue. Turn the switch all the way counterclockwise. The LEDS should turn white, indicating that the device is functional and active.

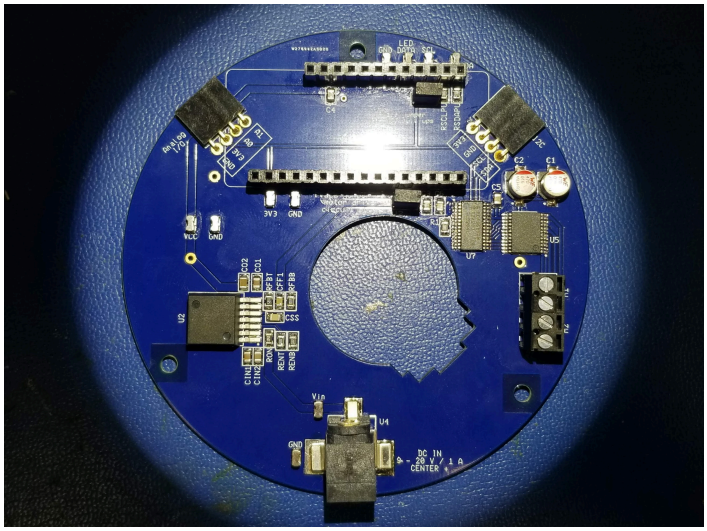


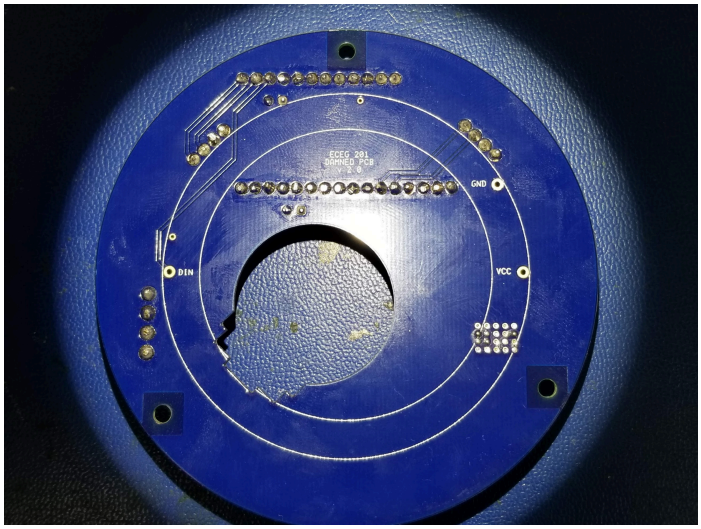
At this point, the device is detecting the frequency, so you can begin tuning your guitar as normal. Just pluck away at your strings and watch the tuning meter tell you how sharp/flat you are to tune your guitar. The device will indicate which string it thinks you're tuning by lighting one of the bottom six LEDs green corresponding to the six guitar strings (Leftmost is the low E2, rightmost is the high E4).

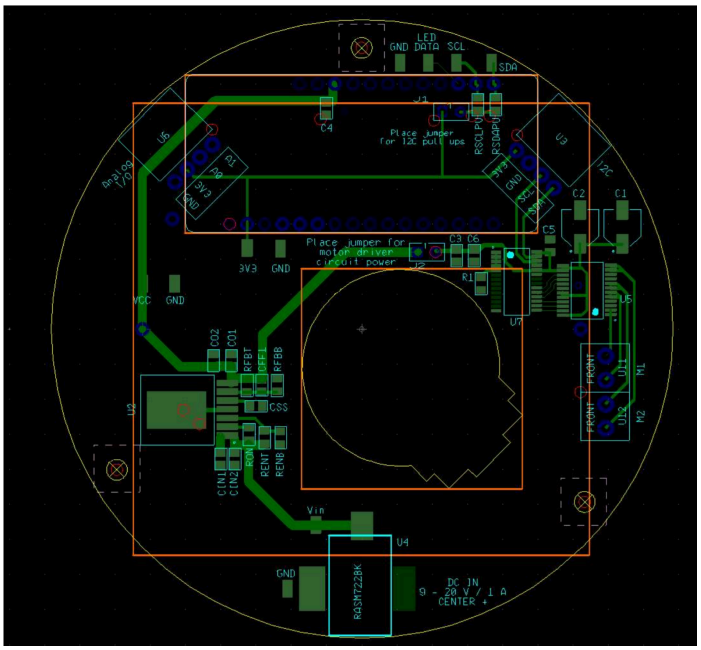
When you have successfully tuned all your strings, activate the Off switch by turning the potentiometer all the way clockwise. The device will then light up the bottom LEDs all green to indicate that the program has ended and the necessary data has been sent to the thingspeak channel.

9. Mechanical Layout and Dimensions

9.1 Printed Circuit Board [15]

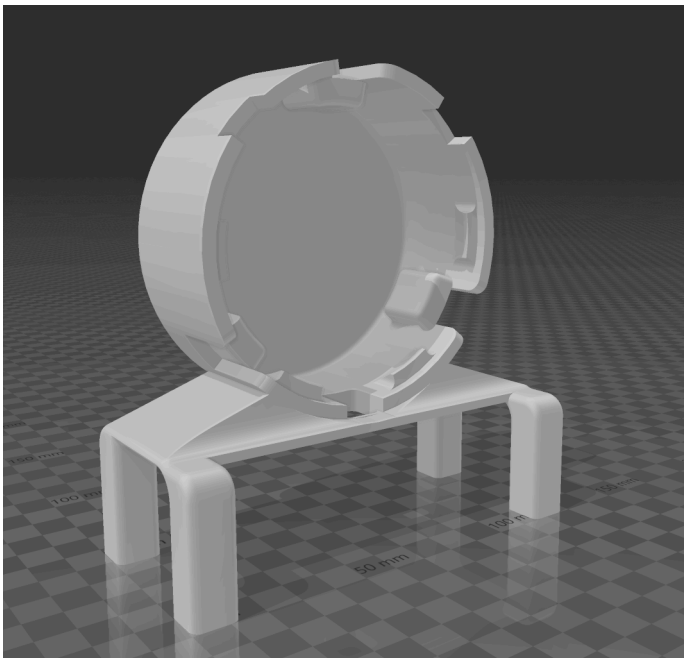


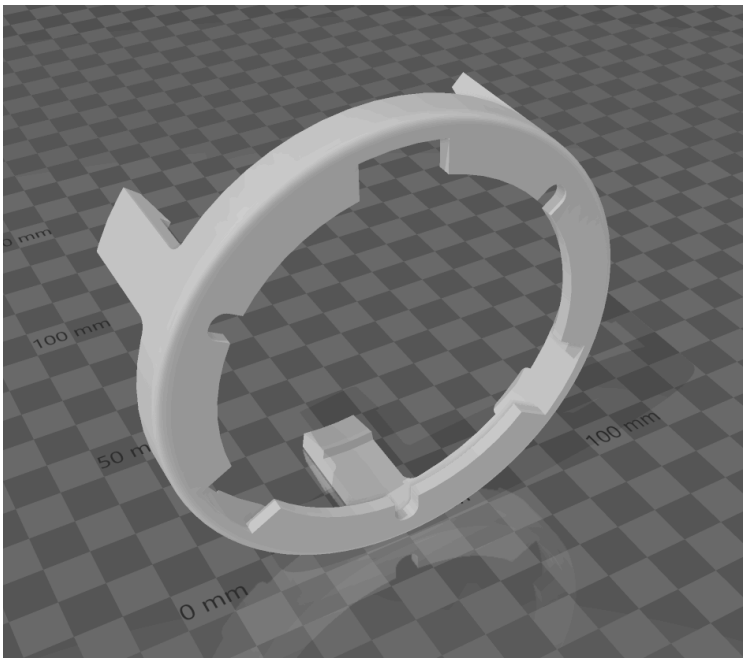


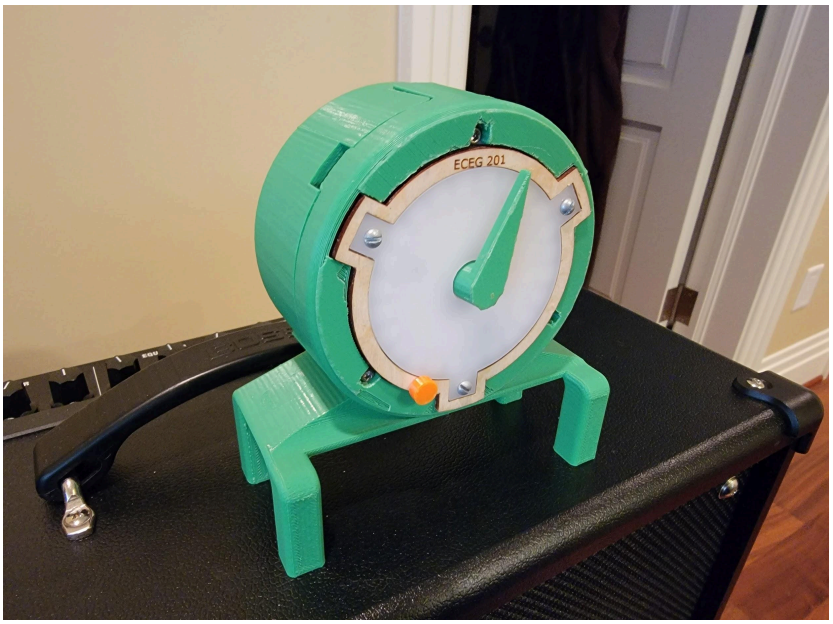


Board Diameter	89.662 mm
Motor Cutout Diameter	28.448 mm

9.2 Case







Shell Front - Diameter	99 mm
Shell Back - Length	128 mm
Shell Back - Height	134 mm
Shell Back - Width	58 mm

10. Citations

[Wherever you cited other documents make sure you include in-text citations with a numbered list here in IEEE format]

Collaborations:

- Ben Buentello, for some figures and operating conditions data I was missing

Sources:

- [1] CUI Devices, “SJ1-352XN Series Datasheet - Jacks | Audio Connectors | CUI Devices,” SJ1-3525N **datasheet**, 10/10/2019. [Online]. Available: https://www.cuidevices.com/product/resource/sj1-352xn.pdf [Accessed: Jun-2021].
- [2] I. Philosophe, “ECEG 201 Design Assignment 1,” Feb-2021. [Online]. Available: https://docs.google.com/document/d/13-4IAQp1PRizNdEJrT1j0krIFEEoZBBVqcdLM9EGg-I/edit [Accessed: Jun-2021].
- [3] A. Industries, “NeoPixel Ring - 24 x 5050 RGB LED with Integrated Drivers,” *adafruit industries blog RSS*. [Online]. Available: https://www.adafruit.com/product/1586. [Accessed: Jun-2021].
- [4] “28BYJ-48,” Jan-2021. [Online]. Available: https://moodle.bucknell.edu/pluginfile.php/1467299/mod_resource/content/1/28BYJ-48%20stepper%20motor.pdf. [Accessed: Jun-2021].
- [5] A. Industries, “Adafruit AirLift FeatherWing – ESP32 WiFi Co-Processor,” *adafruit industries blog RSS*. [Online]. Available: https://www.adafruit.com/product/4264. [Accessed: Jun-2021].
- [6] Texas Instruments, “LMZ12001 1-A SIMPLE SWITCHER® Power Module With 20-V Maximum Input Voltage,” LMZ12001TZX-ADJ/NOPB **datasheet**, Jan-2010. [Online]. Available: https://www.cuidevices.com/product/resource/sj1-352xn.pdf [Accessed: Jun-2021].
- [7] L. Ada, A. Barela, D. Nosonowitz, “Feather Specification,” *Adafruit Learning System*, 14-May-2017. [Online]. Available: https://learn.adafruit.com/adafruit-feather/feather-specification [Accessed: Jun-2021].
- [8] B. Buentello, “Plugged-In Desk Assistant: Datasheet,” Mar-2021. [Online]. Available: https://docs.google.com/document/d/1RVpQAqL6FgfDwfWcqPTDrKmULAHghrHWnKN5HPkxDWU/edit# [Accessed: Jun-2021].
- [9] I. Philosophe, “ECEG 201 Design Assignment 9,” May-2021. [Online]. Available: https://docs.google.com/document/d/1V5KuxAIC0hODWiTivp1UE8US7ZI2pDJnbhqdKRay7BM/edit [Accessed: Jun-2021].
- [10] “ulab – Manipulate numeric data similar to numpy,” *Adafruit Learning System*. [Online]. Available: https://circuitpython.readthedocs.io/en/latest/shared-bindings/ulab/. [Accessed: Jun-2021].
- [11] J. Epler, “ulab: Crunch Numbers fast in CircuitPython,” *Adafruit Learning System*. [Online]. Available:

https://learn.adafruit.com/ulab-crunch-numbers-fast-with-circuitpython?view=all#fir-filter-example-pulse-rate-estimation. [Accessed: Jun-2021].

- [12] “Pitch detection algorithm,” *Wikipedia*, 10-Jan-2021. [Online]. Available: https://en.wikipedia.org/wiki/Pitch_detection_algorithm. [Accessed: Jun-2021].
- [13]“PITCH DETECTION METHODS REVIEW,” *Introduction*. [Online]. Available: https://ccrma.stanford.edu/~pdelac/154/m154paper.htm. [Accessed: Jun-2021].
- [14] T. P. Szynalski, “Online Tone Generator,” *Online Tone Generator - generate pure tones of any frequency*. [Online]. Available: https://www.szynalski.com/tone-generator/. [Accessed: Jun-2021].
- [15] I. Philosophe, “ECEG 201 Design Assignment 4,” Mar-2021. [Online]. Available: https://docs.google.com/document/d/1V-x1GGaQwQBtg8n5-L3OaO0D1HH-6QhhTvXEtuiPcic/edit [Accessed: Jun-2021].