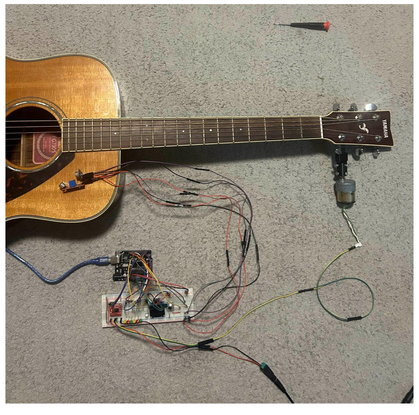
Automatic Guitar Tuner

Alex Shin, Jinwoo Kim

Introduction

The goal of this project was to create an embedded system that automatically tunes any guitar. Our system employs two sound sensors to capture the audio signals generated by plucking the guitar strings. We also have a button along with six LEDs for the user to click through and select the string that they want to tune. Upon selecting a string to tune, the user can attach the motor to the guitar peg of that string, pluck the string, and the motor will automatically rotate to tune the guitar. When the string is in tune, an indicator LED will turn green to indicate that the selected guitar string is in tune, and will be off when the string is out-of-tune. By automating the tuning process, our project seeks to offer guitarists a convenient and efficient solution for achieving optimal tuning accuracy without the need for manual adjustment.



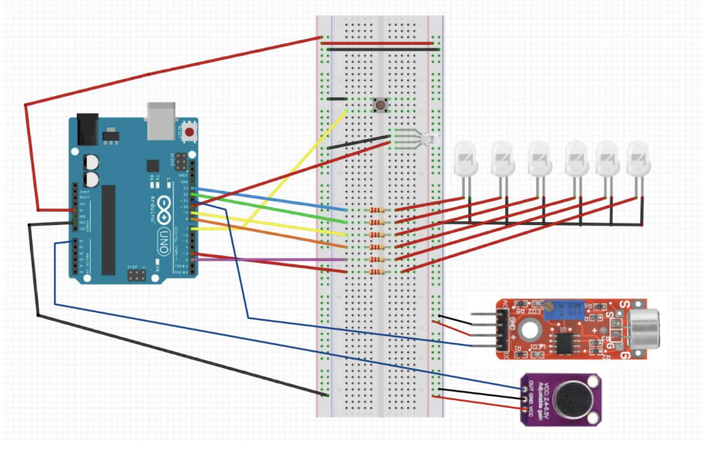
Note Selection

In order for the user to select the string to tune, we opted to use 6 LEDs, which the user can cycle through using a button. The selected string is then handled by our software, which references the tuned frequency for that string and uses audio signal detection to tune the note accordingly with the help of the geared DC motor. Although we originally attempted to use a rotary encoder with an I2C display, this posed problems with the memory usage of our Arduino Uno. The schematic diagram on the right shows how the LEDs and the button are attached to the Arduino.

Schematic and Materials

Materials Used

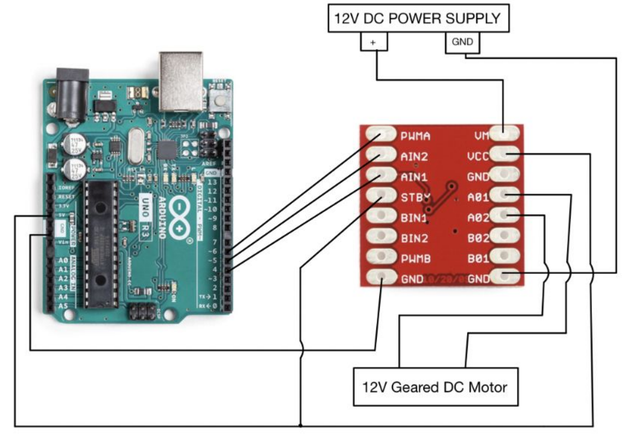
- Arduino Uno Microcontroller
- RGB Led
- 6 LEDs
- Button (we used a rotary encoder push switch but any button will do)
- Sound Sensors
  - KY-037
  - GY MAX4466
- Geared DC Motor
  - Torque @ Max. Efficiency (g-cm) = 850
  - 30:1 Gear Ratio
- 12V DC Power Supply
- 7 Resistors (330Ω)
- Tuning Peg Attachment/Gripper (3D printed)
- Jumper Wires



Audio Signal Detection

We used a KY-037 High Sensitivity Sound module to detect the presence of sound, and a GY MAX4466 Sound Module for a precise analog reading of the sound. By sampling the analog signal and applying a Fast Fourier Transform, we were able to retrieve the frequency of the note being played by the guitar. However, the strings of our guitar did not produce a pure tone, and the sound module displayed some bias in the results. In order to account for this error, we played each string of our tuned guitar and sampled the frequencies we got from our sound sensor, taking the average frequency of ten samples for each string. Most of the frequency recordings we recorded were very similar to the expected values. As an example, for the D-string we received an average frequency of 148.85Hz (the actual frequency of D3 is 146.83Hz). Thus, when tuning the D-string, we set the target frequency to our sampled value.

Geared DC Motor



We designed our own guitar peg gripper, 3D printed it, and attached it to our geared DC motor. By checking whether the detected frequency is too flat or high, we adjusted the motor accordingly in order to turn the pegs in the correct direction. Although we originally planned on using a stepper motor, we realized that the torque was not strong enough to turn the pegs, and decided to employ a DC motor with higher torque instead.