Handheld Automatic Guitar Tuner

A. Lee Swindlehurst, *Advisor*, Victor Darakjian, *Leader/Signal Processing*, Adalberto Sicairos *Hardware Implementation*, Dexter Gianto Suherman, *Microcontroller Programming*, Ha Young An *Application Development*

***Abstract*— The goals of this project are multifaceted. They include:**

1. **Use of a piezoelectric vibration sensor to collect vibration data from a guitar.**
2. **Implementing a fast fourier transform algorithm to find a fundamental frequency of vibration (the frequency of the string).**
3. **Comparing the frequency of the string to what it should be.**
4. **Turning a stepper motor which will turn the tuning peg of a guitar so that the string plays in the correct frequency.**
5. **Develop an app which will guide the user and allow them to customize their tuning.**

The methods involved in completing these goals were, learning how to use a microcontroller (Arduino) and its accompanied Integrated Development Environment (IDE), learning how to control and wire up a stepper motor, learning how to use batteries to power our microcontroller and motor, learning how to use a piezoelectric vibration sensor, learning how to set up a bluetooth connection, and using an oscilloscope and power supply to test our product.

Overall the progress we have made consists of obtaining vibration signals from the guitar, running it through the Fast Fourier Transform code to get the frequency of the string, wiring up and powering the motor, using different voltages to power the microcontroller and stepper motor, and developing the code for a bluetooth connection. We have been successful in these things, however, our vibration signals from the guitar are not always reliable, so will need to explore options for different vibration sensors, or different locations of placing sensors on the guitar.

I. Introduction

THE Handheld Automatic Guitar Tuner project encompasses a wide spectrum of engineering design, ranging from configuring electronic circuits for Arduino microcontroller to setting up remote control connection with a mobile device. The specific need of this project is to produce an Arduino-controlled guitar tuner that can rotate the tuning pegs on guitar strings to reach target frequencies with mathematical precision. The device will be physically held on the tuning peg of a particular string. Once that string is plucked by the user, the device will use signal processing techniques to measure the frequency of the string through vibrations physically traveling through the guitar and into the device. The device will also need to filter out the unwanted noise to obtain the frequency of the string. Then, by comparing the frequency of the string to what it should be, the device will turn a motor that will automatically turn the tuning peg of the string until it reaches the correct frequency. Since there are many different types of guitar tunings, mobile application will be developed to allow the user to select a tuning scheme of their choice and communicate the user's desires to the device.

Overall, the methods used for the project are stepper motor control with a separate motor control board, and implementation of a piezoelectric sensor to gather vibration data to run through a fast fourier transform algorithm, in order to obtain a frequency of the string vibration. Currently the vibration sensor seems to be outputting many different signals, so in the future, we will need to implement some filters to limit the signals that we analyze.

II. Materials Used

Overall, the materials used for the Handheld Automatic Guitar Tuner can be categorized into hardware and software components.

Electric guitar, acoustic guitar, Arduino Mega 2560, Piezo Film Vibration sensor, 12V Hybrid Stepper Motor, Motor Control Shield, Bluetooth Module, and an Operational-Amplifier are equipment used for the hardware.

Arduino Integrated Development Environment, MATLAB programming language, and JAVA programming language are used for the software.

III. High-level hardware and software system

The hardware components consists of: guitars (electric and acoustic) for testing, Arduino Mega 2560 as the microcontroller for stepper motor and the retriever of our sensor data, Piezoelectric Vibration sensor is used to detect the physical vibration frequency traveling through the tuning peg, 12V hybrid stepper motor which will turn the tuning peg of the guitar in the correct direction, Motor Control Shield is an add-on to the Arduino microcontroller which provides the necessary voltage to power the stepper motor, a Bluetooth module to provide a two way communication between the microcontroller and the mobile application, and an Operational-Amplifier to

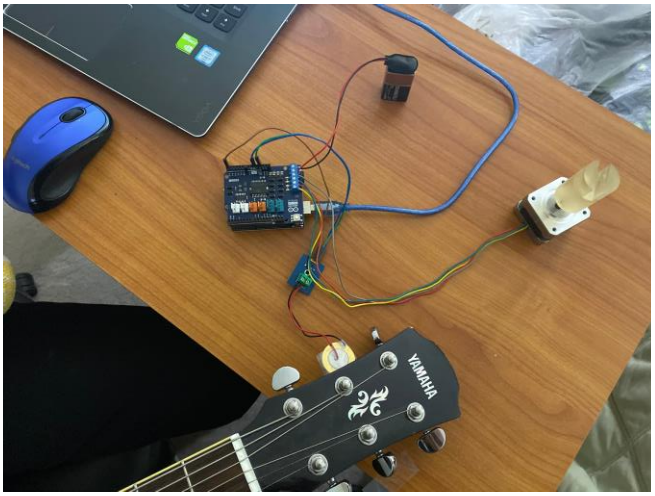

Fig. 1. High-level hardware system used for the Automatic Handheld Guitar Project. An Arduino Mega 2560 as the microcontroller, the Piezo Film Vibration Sensor and an acoustic guitar is shown.

amplify signals from the piezo sensor[2]. Part of the hardware equipment are displayed in Figure 1.

The software component consists of Arduino Integrated Development Environment (IDE) to program the microcontroller to send the data from the vibration sensor to the motor and control the motor in real time. In addition, JAVA programming language in eclipse to code the Android mobile application, which will provide a graphical user interface (GUI) and features within the application to allow users to remotely adjust the stepper motor to their desired frequency.

Altogether, the Android mobile application that will utilize the Bluetooth feature to communicate with the Arduino microcontroller. The mobile application will gather and analyze the data sent from the microcontroller and it will provide users with a feature to choose their desirable tuning scheme for that particular string.

IV. METHODS

The methods involved in working on the project can be divided into hardware and software methods.

For hardware, we used a variable power source to test the different voltages which our motor can produce sufficient torque to turn a guitar tuning peg. We also tested what voltage to power our amplifier for the vibration sensor was optimal. It could run on 3.3 or 5V. It appears the 5 V option would be more reliable.

For software, the methods utilized were mostly found in the Arduino IDE. Here we could run our code for the motor control and Fast Fourier Transform to see how our parts are working. Additionally we heavily relied on the Serial Plotter within the IDE to visually see what frequencies our sensor was picking up. We found that the 110 HZ frequency from the A-string of the guitar was particularly reliable, while the rest were difficult to obtain the desired results[1]. Understanding the concept of Fast Fourier Transform was also a challenge to overcome, and eventually we could implement a Fast Fourier Transform by using the ‘arduinoFFT’ library in our code [3].

V. RESULTS AND PERFORMANCES

Overall, the team managed to get a working prototype with all the main components of the project, as seen in Figure 1. First, the team was able to get a decent reading from the piezoelectric sensor, and after sending that signal through a fast Fourier transform code, the frequency of the guitar string was

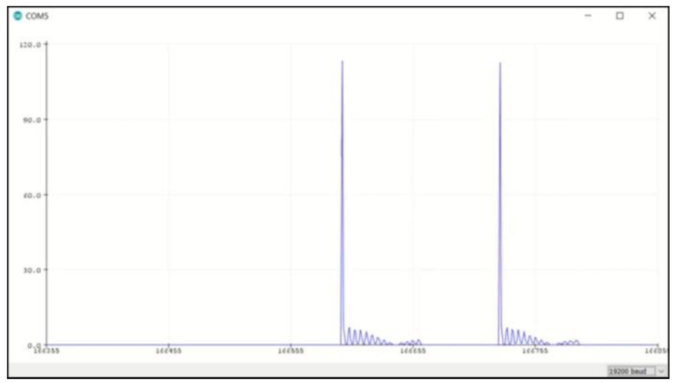

Fig. 2. High-level hardware system used for the Automatic Handheld Guitar Project. An Arduino Mega 2560 as the microcontroller, the Piezo Film Vibration Sensor and an acoustic guitar is shown.

measured. From Figure 2, we see the piezoelectric sensor output readings when the A-string was plucked. As expected, approximately 110 Hertz was displayed as the fundamental frequency [4]. Secondly, the team has successfully connected the DC stepper motor. Although the motor is meant to run at 12V, at this voltage the motor did not provide sufficient torque needed in order to rotate the guitar peg. After careful experimentation at voltages higher than 12V, we found that the voltage needed to get sufficient torque was 17V. We will be developing a power distribution system so that the necessary voltages can reach different parts of the device. Currently the plan is to use two 9V batteries in series and uses a voltage dividers to power the motor with 17 V, the Motor control Shield with 5V, and the Arduino with 5V.

VI. SUMMARY

In summary, the team managed to read vibration frequency input from picked guitar strings and rotate the stepper motor in response. While some of the readings are not very reliable, we will be experimenting with different ways of placing the sensors and using filters to get more reliable data. Further, we found the necessary voltages and devised a plan for power distribution, which has not been executed using 9V batteries yet. In the next quarter, the team will primarily focus on fine-tuning the input frequencies so that the stepper motor can turn the pegs until the input matches desired frequencies for each string. At the same time, the team will develop the mobile app further until remote control is fully implemented. One way to extend on the mobile app will be to implement alternative tuning modes in which the target frequencies differ from those in standard guitar tunings. The user will be able to select a mode and the handheld tuner will target the desired frequencies for that mode.

VII. Acknowledgement and References

A. [1] K. Rahnamai, B. Cox and K. Gorman, "Fuzzy Automatic Guitar Tuner," NAFIPS 2007 - 2007 Annual Meeting of the North American Fuzzy Information Processing Society, San Diego, CA, 2007, pp. 195-199. doi: 10.1109/NAFIPS.2007.383836

B. [2] R. Keim, “How to Design Charge Amplifiers for Piezoelectric Sensors,” All About Circuits, 23-Oct-2018. [Online]. Available: https://www.allaboutcircuits.com/technical-articles/how-to-design-charge-amplifiers-piezoelectric-sensors/. [Accessed: 14-Oct-2019].

C. [3] Heckbert, P. (2019). Fourier Transforms and the Fast Fourier Transform (FFT) Algorithm. [online] Cs.cmu.edu. Available at: http://www.cs.cmu.edu/afs/andrew/scs/cs/15463/2001/pub/www/notes/fourier/fourier.pdf [Accessed 14 Nov. 2019].

[4] S. Hokin, “The Guitar”, 2019. [Online]. Available: http://www.bsharp.org/physics/guitar. [Accessed: 08-Nov.-2019].

The Automatic Handheld Guitar Tuner Project members consist of: Victor Darakjian and Adalberto Sicairos, both of whom study Electrical Engineering (EE), and Young Ha An and Dexter Gianto Suherman, both of whom study Computer Science and Engineering (CSE).

The Handheld Automatic Guitar Tuner Project members would like to acknowledge with gratitude to the support provided by their advisor, Dr. A. Lee Swindlehurst, and class conductor, Dr. Stuart Kleinfelder, for securing and providing funds from the UCI engineering department for this project.

The majority of our hardware equipment came from Arrow Electronics, except for the Bluetooth Module which was ordered from SparkFun Electronics.

Appendix I - Standards

For the Handheld Automatic Guitar Tuner project, we were required to constantly adapt our means and approach to our problems. We find that our initial ideas from the beginning of the quarter continue to improve whenever we reach any dead ends. For instance, the initial idea for the Android Mobile Application, which would remotely control the guitar tuner, was to connect via Wi-Fi. With more research, however, it became clear that Bluetooth connection would be more feasible to implement. Bluetooth connection is secure as it is a point to point connection and you can only connect one device. Moreover, Bluetooth is more versatile and consume less power in comparison to Wi-Fi.

Furthermore, for microcontrollers, there was a consideration to either use Arduino or Raspberry Pi. However, an Arduino Mega 2560 microcontroller is chosen as it is a great tool for developing interactive objects, taking inputs from a variety of switches or sensors, and controlling a variety of lights, motors, and other outputs. The Arduino microcontroller has a convenient and user-friendly time interfacing with analog sensors and motors, while the Raspberry Pi has a complicated method in getting sensor readings.

The Bluetooth module is compliant to the standards in terms of quality and costs. The team picked Bluetooth module which provides features that conforms to the project needs. The Bluetooth module ensures the best quality for the given feature, while keeping it at the lowest cost possible. Furthermore, the Arduino Mega 2560 microcontroller is compliant to the standards in terms of compatibility and features. The Arduino microcontroller is more versatile and compatible with other hardware components than the alternative considered, and the microcontroller provides a friendly feature which allows the gathering and analyzing the data a simplified process.

Appendix II - Challenges

The team project encountered various challenges in the Automatic Handheld Guitar Tuner Project. However, as the team progresses, new solutions, strategies, and methods are discovered to overcome those challenges.

The most prominent challenges the team faces was that funding for the project came late, and therefore, the hardware components only came nearly at the end of the quarter. Therefore, while waiting for the hardware components to arrive, the team started on the software components, which is the Android Mobile Application and experimenting on the Low Pass Filter and Fast Fourier Transform on MATLAB.

The Automatic Handheld Guitar Tuner project must ensure that users will not be exposed to any potential hazards that may jeopardize their safety. We must prevent short circuiting arduino’s battery to something of low resistance, like a wire, that could potentially cause the battery to explode. Since the project deals with electronic circuits, the project has to deal with eliminating the risk of electric shock to the users. We will remedy this risk by designing a chassis to encompass all electronics, so that the user only has contact with insulating plastic.

The project must also assemble the motors such that users will not be injured while using it. Proper and specific instructions are given to users to properly operate the equipment and minimize the risk of potential hazards. One example of injury due to the motor would be the motor uncontrollably tightening a string until it snaps and lashes the user. This would be a major health hazard, and therefore we will ensure that the motor will never continuously tighten for an extended period of time. We will ensure that motor movements are in short calculated intervals. In addition, to prevent an injury to us students, we are developing a test-bench with a cover over the string, so that a snapped string will not lash one of our group members.

Appendix III - Security

The most important security issue we have on the hardware side of this project is that the motor could overtighten a string causing it so snap. One way we can overcome this problem is to program the motor to never rotate past a certain frequency. We will safely over tighten each string and record the frequency at that tension. Then we will program our arduino to stop turning the motor if that frequency is ever reached, since our device will be receiving sensor data continuously, this solution should work. Additionally, we plan to power our handheld device with 2 9V batteries in series. To ensure there are no problems with shock to the user, we will be designing a chassis to contain all our electronics and designing a robust power distribution system so that the circuit board does not overheat, or become under-powered. Currently we are experimenting with voltage dividers to see if we can power each of our device components reliably.

For the software side of the project, the main security issue we are facing is a person trying to pair their device to our project by brute forcing a key. To fix this, we will make a key that is 8 bytes in length. Having a key this large means that even if someone were to try and brute force the password it would take an average of 2^31 attempts. This will ensure user-intended functionality at all times, by limiting the ability for a hacker to pair with the device.