Arduino Based Automatic Guitar Tuning System

Matej Žerjav
University of Ljubljana
Faculty of Mechanical Engineering
Askerceva 6
1000 Ljubljana, Slovenia
Email: zerjav.matej@gmail.com

Primož Podržaj
University of Ljubljana
Faculty of Mechanical Engineering
Askerceva 6
1000 Ljubljana, Slovenia
Email: primoz.podrzaj@fs.uni-lj.si

***Abstract*—Acoustic guitar is one of the best known and most commonly used musical instruments. It emits sound waves because the guitar player makes the strings oscillate. In order to function properly, i.e. emit the sound of the appropriate frequency it is very important the tension in each string is adjusted regularly. This process is called tuning. It is usually done manually with a help of universal tuner. In this paper an automatic guitar tuning system is presented. It is based on the Arduino platform. It uses a microphone to record the sound, calculates its frequency and then turns the tuning pegs on guitar head with stepper motor to get the correct frequency.**

***Keywords*—IEEE, IEEEtran, journal, LaTeX, paper, template.**

I. Introduction

Many engineers working in the field of Mechatronics say, that they gained the most knowledge, when they had to realize some real life application. Although the underlying theory from both Mechanics and Electronics, which give the name Mechatronics, when combined, is of course important for the understanding of key concepts, real life applications give a chance to students to memorize these concepts easier. For this reason it is important to widen the set of mechatronic applications as much as possible, so that everyone can see its applicability and also to make it possible to find an application that a specific person is most familiar with.

In recent years Arduino platform has become one of the most popular (if not the most popular) platforms for real time applications, because it is very inexpensive and quite easy to use. Applications range from drones [1] and robot dogs [2] to motor control [3] and portable solar power source [4]. In this paper an application from the field of acoustics will be presented. It is well known that guitar as all the other stringed instruments has to be tuned in order to produce the sound of the correct frequency. Usually this process is done manually. In this paper an Arduino based automatic guitar tuning system will be presented.

II. Theoretical background

A. String oscillations and pressure waves in gas

It is well known that in physics sound can be described as the oscillations in gas pressure Δp measured from some equilibrium value [5]:

$$\Delta p = \Delta p_{max} \sin(kx - \omega t) \tag{1}$$

The equation represents longitudinal waves in space with the angular wave number k and the angular frequency ω. How musical instruments achieve to produce this waves depends on the type of the instrument. In the case of stringed instruments, this oscillations of pressure are result of the oscillation of various strings. In order to get the relation between the string motion and air pressure we first need the relation between the parameters of the string and the speed of waves on a string. It is given by the following equation [5]:

$$v = \sqrt{\frac{T}{\mu}} \tag{2}$$

where T is the tension in the string and μ the mass per unit length of the string. If the string is fixed at both ends (see the first plot in Fig. 1) the phenomenons of reflection and interference have to be taken into account. This results in the occurrence of standing waves. Different modes of oscillations are possible (see bottom three plots in Fig. 1). The points of zero amplitude are called nodes and the points of maximum amplitude are called antinodes. The section of the standing

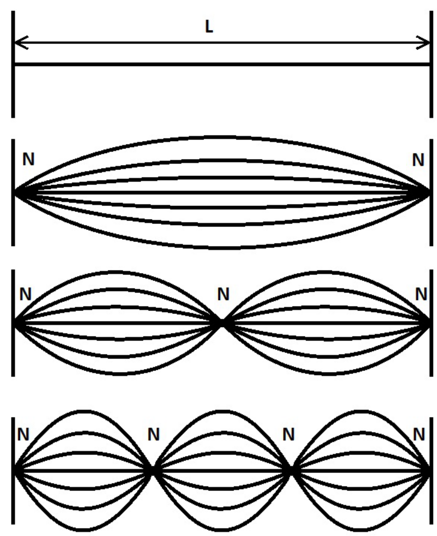

Fig. 1. Schematic representation of a guitar string oscillating with a fundamental frequency (second plot) and first and second harmonic (third and fourth plot)

ISBN: 978-1-941968-42-0 ©2017 SDIWC

wave from one node to the next node is called a loop. The frequency of oscillation is given by the following equation [5]:

$$f_n = \frac{n}{2L}\sqrt{\frac{T}{\mu}} \quad n = 1, 2, 3, \ldots \tag{3}$$

where L is the length of the string and n the number of loops. The lowest frequency (associated with the second plot in Fig. 1), when $n = 1$ is called the fundamental frequency and is therefore given by the following equation:

$$f_1 = \frac{1}{2L}\sqrt{\frac{T}{\mu}} \tag{4}$$

Frequencies for $n \geq 2$ are called harmonics.

B. Acoustic guitar

An acoustic guitar (see Fig. 2) is an instrument with six strings. Its main part is the body to which a neck is attached.

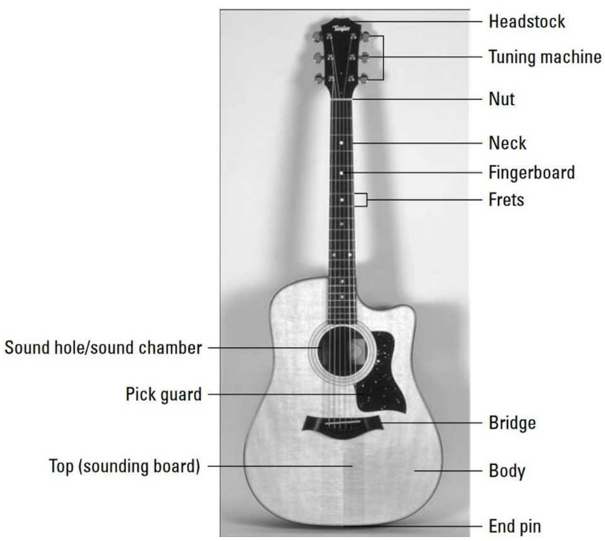

Fig. 2. Acoustic guitar and its basic components [6]

The strings are attached to the bridge on the body and to the head at the end of the neck. The strings differ in mass per length (parameter μ in Eq. 4. The parameter L can be modified during playing by the position of the fingering point. The plucking point determines the presence (the amplitudes) of the higher harmonics. When a string is plucked close to the bridge a tone that is softer in volume is produced. It is also brighter and sharper, and the sound is richer in high frequency components [7]. The other extreme case is when plucking is done near or over the fingerboard, closer to the midpoint of the string. In this case, the tone is louder, mellower and less rich in high frequency components. The fundamental frequencies for each of the strings are given in Table I. The exact frequency can be obtained by a proper adjustment of the string tension (parameter T in Eq. 4. For this purpose each acoustic guitar has the so called tuning machine located on the head of the guitar. The appropriate tension can be obtained by

TABLE I
THE FUNDAMENTAL FREQUENCIES OF GUITAR STRINGS [8]

String	Note	Octave	Frequency [Hz]
6	E	2	82,41
5	A	2	110,00
4	D	3	146,83
3	G	3	196,00
2	H	3	246,94
1	E	4	329,63

the adjustment of the tuning pegs. This process is called the (guitar) tuning.

As humans are unable to exactly determine the frequency of the audible sound, some kind of device to assist in this process. The most common one is the universal tuner shown in Fig. 3. The user interface makes it possible to specify

Fig. 3. Fire&Stone / Coxx CLIP-UT Tuner

the string we want to tune. The tuner then measures the (fundamental) frequency of the sound emitted by the guitar and gives information about the appropriate movement of the specific tuning peg. After several iterations the string is tuned within a specific range of the frequency given in Table. I. Of course this process has to be repeated for each string.

III. EXPERIMENTAL SETUP

The main building block of an automated guitar tuning system is a controller in which control algorithm programmed. For our application, we have decided for Arduino Zero (see Fig. 4). It is namely very inexpensive, commonly used and

Fig. 4. Arduino Zero

user friendly system. The development environment that can be installed free of charge includes many examples. These can then be easily combined to perform more complex control related tasks. In our case the main tasks are the acquisition of the fundamental frequency in real time and the drive of a stepper motor, which turns the tuning peg on the guitar

ISBN: 978-1-941968-42-0 ©2017 SDIWC

head. The scheme of the whole system is given in Fig. 5. In the lower left corner of the electret microphone is shown.

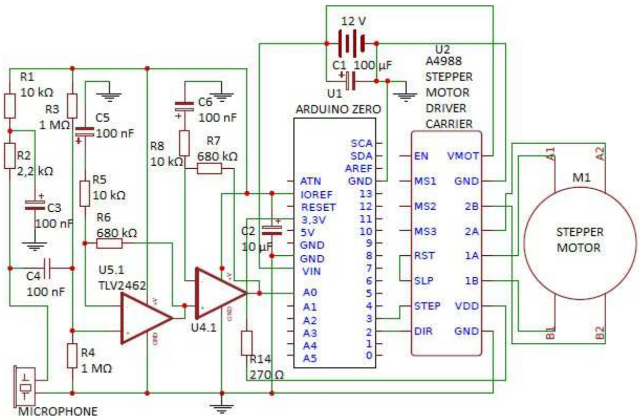

Fig. 5. The electrical scheme of the whole system

Its output signal is then amplified twice before it is fed into the Arduino Zero. This is achieved by the TLV2462 Dual Operational Amplifier shown in Fig. 6. The relation between

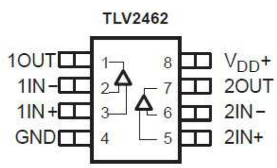

Fig. 6. TLV2462 Dual Operational Amplifier

output voltage V_{out} and two input voltages (V_1 and V_2) is given by the following formula [9], [10]:

$$V_{out} = G_V(V_2 - V_1) \quad (5)$$

where G_V is the voltage gain. It can be controlled by the values of the resistors in input and feedback paths [11]. After the resulting signal is obtained by the Arduino, it is very easy to get the frequency. The code needed to get is namely very simple (see Fig. 7). When the actual fundamental frequency

```
int frequency = meter.getFrequency();
  if (frequency > 0){
    Serial.print(frequency);
    Serial.print(" Hz    ");
```

Fig. 7. The part of code used to get (and write) the fundamental frequency.

is obtained, we just need to compare it with the desired fundamental frequency for the specific string. If the actual fundamental frequency is within a certain predefined range around the desired one nothing needs to be done. If it is either too low or too high, the tuning peg must be turned in the correct direction. In order to be able to turn the tuning pegs on the guitar head a special part had to be made (see Fig. 8).

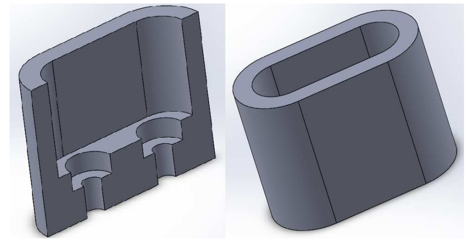

Fig. 8. A part of the system used to turn the tuning pegs

As it must fit the selected stepper motor on one side and the tuning peg on the other, it is difficult to find it. For this reason it was the only part that had to be designed specifically for this application. This was done in SolidWorks and produced using a 3D printer. The part attached to the stepper motor is shown in Fig 9. The stepper motor used was a NEMA 17 stepper motor

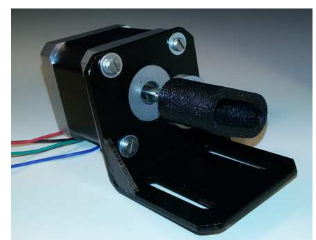

Fig. 9. Stepper motor

(type 42HS4013A4 with a 1.8° step angle). In order to drive it a A4988 Stepper Motor Driver was used. The part of the code used to drive the stepper motor is shown in Fig. 10. The

```
digitalWrite(dirPin,LOW);
for(int x=0; x<15 ; x++ ) {
  digitalWrite(stepPin,HIGH);
  delayMicroseconds(500);
  digitalWrite(stepPin,LOW);
  delayMicroseconds(500);
    }
  delay(2000);
```

Fig. 10. The part of the code used to drive the stepper motor

last delay in the code was selected to be 2 seconds in order to hear and sense the change in frequency of the emitted sound. Some delay is needed in any case, because the system needs some time to determine the frequency correctly.

The stepper motor itself than has to be fixed relative to the guitar head. A special fixture was made for that purpose. The setup is shown in Fig 11. The only part shown in this figure that was not mentioned before is the battery at the top of the figure. The guitar player now only has to put the very light stepper motor into the position and specify which string he wants to tune. After that the string is plucked and the system autonomously tunes the string in.

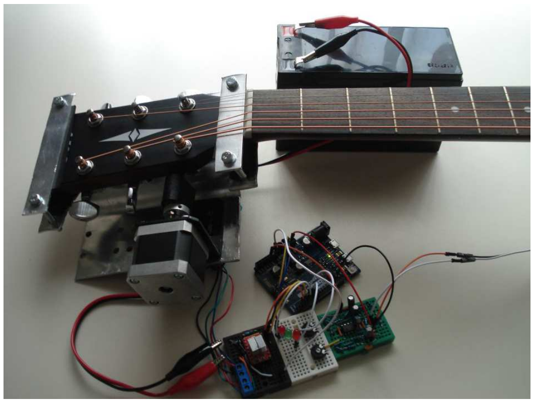

Fig. 11. System setup

IV. Results

In order to verify the precision of the presented automated guitar tuning system it was used in connection with the universal tuner shown in Fig. 3. The only purpose of the tuner is to check if the correct fundamental frequency is obtained. The whole system used for verification is shown in Fig. 12. When tuning is done manually the universal tuner screen is

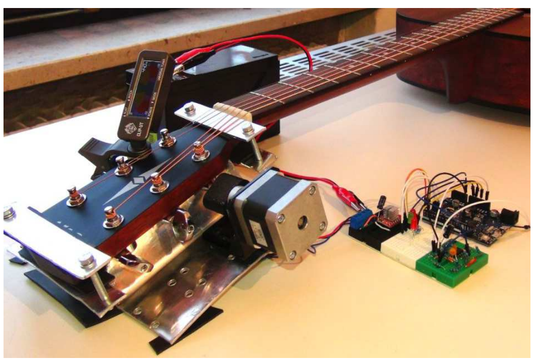

Fig. 12. The photo of the whole system used for verification

observed. Fig. 13 shows the display for two special cases. The

Fig. 13. Universal tuner display (upper photo when frequency is too low; lower photo when frequency is appropriate

upper photo in Fig. 13 shows the display when the frequency is too low. The guitar player must increase the tension in the string. This is done by tuning peg rotation. This step is repeated iteratively until the universal tuner signals that and appropriate frequency has been achieved (lower photo in Fig. 13).

When tuning is done automatically, the guitar player first has to select which string (actually which note to be more exact), he wants to tune. Then the stepper motor is put into the correct position (see Fig. 12) and the string is plucked. The system is then able to autonomously turn the tuning peg into the correct position. There are only two requirements that have to be fulfilled:

- The starting tension in the string must not be too far off the appropriate value. If the string is much too loose the system has difficulties to when trying to determine the fundamental frequency of oscillations.
- Even if the tension of the string is not too far off the appropriate value, it may be needed to pluck the string several times. Especially if we do it very gently. The amplitude of oscillations namely steadily decreases. In order that the system can function, the amplitude of oscillation must be above a certain threshold. It should be noted that how many times the string should be plucked also depends on the selected time delay in the last line of the code given in Fig. 10.

Several guitar players were asked to try the system. After it was explained to them, how the system functions, all of them were able to tune the guitar easily using the presented system.

V. Conclusion

The main purpose of this paper is to present another possible mechatronic application for Arduino based platforms. Guitar tuning is of course done by guitar players, who of course are more artists in nature and often not too interested in science and engineering. The presented automatic guitar tuning system offers them an interesting first step into the field of engineering and electronics, which they might make.

In current setup the system of course lacks in applicability. As we used the components that were used in other projects, many of them are not optimised. Battery for example much too large. If an optimisation were made (especially in the case of battery and in the case of the fixing of the stepper motor, the system would also be of practical value as some of the guitar player that we had invited to test the system have noted.

References

[1] D. McGriffy, *Make: Drones: Teach an Arduino to Fly*. Maker Media, Inc., 2016.
[2] A. J. Boloor, *Arduino by Example*. Packt Publishing Ltd, 2015.
[3] J. Boxall, *Arduino workshop: A Hands-On introduction with 65 projects*. No Starch Press, 2013.
[4] J. Purdum and D. Kidder, *Arduino Projects for Amateur Radio*. McGraw Hill, 2015.
[5] R. A. Serway and J. W.Jewett, *Physics for scientists and engineers with modern physics*, 9th ed. Brooks/Cole, 2014.
[6] M. Phillips and J. Chappell, *Guitar for Dummies, 2nd edition*. Indianapolis, Indiana: Wiley Publishing, 2006.
[7] C. Traube and J. O. Smith, "Estimating the plucking point on a guitar string," in *Proceedings of the COST G-6 Conference on Digital Audio Effects, Verona, Italy*, 2000.
[8] W. M. Hartmann, *Principles of Musical Acoustics*. New York: Springer, 2013.
[9] P. Horowitz and W. Hill, *The art of electronics*, 3rd ed. Cambridge University Press, 2015.
[10] S. T. Karris, *Electronic Devices and Amplifier Circuits with MATLAB Computing*, 2nd ed. Orchard Publications, 2008.
[11] R. L. Boylestad and L. Nashelsky, *Electronic Devices and Circuit Theory*. Pearson Education, 2013.