Global Journal of Pure and Applied Mathematics.
ISSN 0973-1768 Volume 11, Number 3 (2015), pp. 1171-1178

http://www.ripublication.com

Acoustic Guitar Tuner and Identification of Chords using LabVIEW

[1]Swagat Sourav, [1]S. Balamurugan, [1]Marimuthu R,[1]Sudha Ramasamy, [2]A. Bagubali

[1]School of Electrical Engineering, VIT University, Vellore-632014
[2]School of Electronics Engineering, VIT University, Vellore-632014
Email: rmarimuthu@vit.ac.in; sbalamurugan@vit.ac.in ; sudha.r@vit.ac.in; abagubali@vit.ac.in

ABSTRACT

Music is an art form whose medium is sound and silence. A musical instrument is a device created or adapted for the purpose of making musical sounds. In principle, any object that produces sound can serve as a musical instrument—it is through purpose that the object becomes a musical instrument. Every musical instrument needs to be tuned before it could be used to play. Any song which is to be played in the guitar consists of chords. A song cannot be played without the knowledge of the chords involved in it. In this paper we suggest two algorithms using the NI LabVIEW 2010 aimed at tuning an acoustic guitar and identifying the chord being played in the guitar. This project is more efficient and portable than other methods of Guitar tuners since here we need only a laptop and LabVIEW installed in it and no other hardware kit is required, besides the cost involved in the project is also zero.

Keywords–Acoustic Guitar, Tuning, Pitch, Note, Chord, Octave

I. INTRODUCTION

A Guitar is a plucked string instrument, usually played with a pick or fingers. There are many types of guitars- Bass guitar, acoustic guitar, classical guitar, electric guitar etc. In this paper we shall deal with an Acoustic Guitar. An acoustic guitar has 6 strings. They are named: E, A, D, G, B and E Each string has its unique fundamental frequency [2]. Srtings E, A, D, G, B and E has frequency of 82.4, 110, 146, 196, 246.9, 329.6 Hz respectively. Tuning a guitar refers to matching the frequency of each string with the fundamental frequency. Above reference is for standard tuning. When speaking of a tuning such as standard tuning, EADGBE refers to the pitches of the strings from lowest pitch (low E) to the highest (high E).Pitch is a perceptual property that allows the ordering of the sounds on a frequency related scale. Pitches are compared as "higher" and "lower" in the sense associated with musical melodies, which require "sound whose frequency is clear and stable enough to be heard as not noise".

Note is a sign used in musical notation to represent the relative duration and pitch of a sound and the pitched sound itself may also refer to as a note. There are different notes such as: A, B, C, D, E, F, and G. In a guitar fret-board each fret gives a different frequency and hence the note. In music, harmony is the use of simultaneous pitches or chords [4]. Chord in music is any harmonic set of two or more notes that is heard as if sounding simultaneously. In a guitar when three or more strings are strummed together then it constitutes a chord. Chords can be of many types: open chords, power chords, barred chords etc. They can be further classified into major chords, minor chords, power chords etc. Each chord has a unique notes combination, as mentioned earlier that chord is a harmonic set of notes [2]. Hence this distinguishes one chord from the other. So for the knowledge of the chord we got to have the knowledge of the set of notes involved. Hence the identification of a chord is an indirect way of identification of the set of notes. Chords are also named as: A, B, C, D, E, F, and G.

Octave is the interval between one musical pitch and another with half or double its frequency. The octave relationship is a natural phenomenon that has been referred to as the "basic miracle of music", the use of which is "common in most musical systems". Between any two notes, except B - C and E - F, we also have a sharp and/or flat note [3, 5].

If we list the notes, again, and include the sharps and flats, we get:

A - A#/Bb - B - C - C#/Db - D - D#/Eb - E - F - F#/Gb - G - G#/Ab - A

One important thing to notice is that X#/Xb is one note that has two names. For example, A# is the exact same note as Bb. Sometimes, one name will be used, and sometimes the other name will be used. Another thing to notice is that after G#/Ab, we arrive at A again. This second A vibrates exactly twice as fast as the first A, and therefore, the ear tends to hear it as another version of the same note. The second A is called the octave of the first A. If we continue after the second A, we get A#/Bb an octave higher than the first, B an octave higher, C an octave higher etc., etc., until we get to A again. This A is two octaves higher than the first A. If we keep going, the whole pattern just repeats over and over until we can't get any higher on the instrument. So if one note has a frequency of 440 Hz, the note an octave above it is at

880 Hz, and the note an octave below is at 220 Hz. The ratio of frequencies of two notes an octave apart is therefore 2:1. Further octaves of a note occur at 2n times the frequency of that note (where n is an integer), such as 2, 4, 8, 16, etc. and the reciprocal of that series.

II. GUITAR TUNER PROGRAM

A. Methodology

As mentioned earlier that every musical instrument needs to be tuned before using it to play. Hence we first design the Guitar tuner program. The requirements for the program design is LabVIEW 2010 and a PC with an inbuilt microphone to capture the real time sound of the guitar strings.

1. Plucking the guitar string

As mentioned there are 6 strings, we shall pluck one string at a time , once its tune we will move to next string and so on. The guitar should be placed in an ideal position so that the microphone of the computer system is able to sense the frequency of the sound clearly [2].

2. Signal input to the VI in LabVIEW

This acquired signal is real time which is to be given as an input to the guitar tuner program which can be used to process.

3. Signal Processing

This signal will be processed which has been described in the program design section . The LabVIEW tool blocks are used effectively to process the signal and giving us the desired results.

4. Output

Finally our output will be shown which will be in the form of LEDs giving information to the user as the string is: Tuned (frequency matched), Flat (frequency to be increased) or Sharp (frequency to be decreased) [4].

B. Program design and Execution

Our LabVIEW VI will be using the concept of SUB-VIs which means that inside one Main VI there will be smaller modules. In other words, it means executing a series of Vis in a single Main VI as shown in figure 1 (a). These SUB-VIs are actually the VIs of the six different strings of the guitar all connected by Main VI. We have used case structure to implement the six cases or strings. The user is given the freedom to choose the string he wants to tune using a Knob in the Main VI. Let’s say the user chooses to tune the E (low) string as shown in Figure 1 (b). The acquired signal is passed through a band-pass filter, since we will be dealing with only a small range of frequencies we won’t be concerned with the frequencies outside the range.

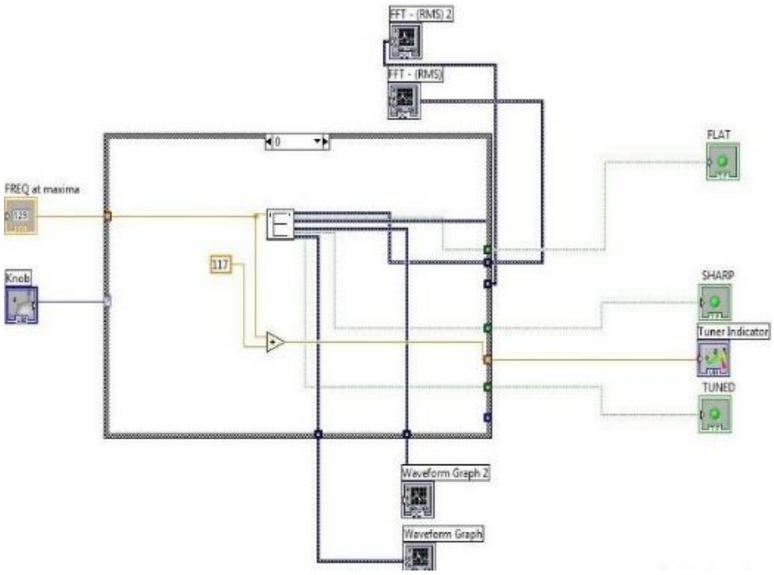

Figure 1(a): Main VI

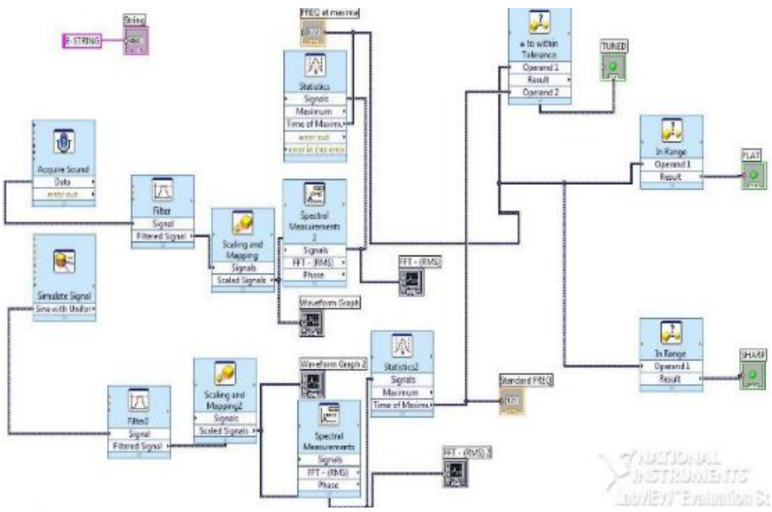

Fig. 1(b) E-string sub VI

The signal then enters into a scaling & mapping block which enables the user to recognize any variation in the graph of the signal when there is a change. Now as the signal is in time domain and we need to convert into frequency domain since we will be doing all our recognition in terms of frequency, so we use an FFT block (Fast Fourier Transform) for the conversion [1]. Besides FFT block also shows a peak whenever there is a change in the signal. Then the signal enters into the Statistics block which identifies at what frequency the peak occurs. Finally the signal enters into a Range block. As it was earlier mentioned that each string has a unique fundamental frequency at which the particular string is tuned, hence here for E string the Range block will compare the frequency at the peak (whose information is provided by the statistics block) with 82.4 Hz. If the frequency is in the range of +/- 0.5 range then Tuned LED is lit. If its below that then Flat LED is lit and if its above that then Sharp LED is lit. There is also a pointer to assist the user how much more he has to adjust the string knob of the guitar in order to get it tuned with respect to the program. Similarly, the user can change to next string VI which he wants to tune and can successfully tune all the six strings of the guitar. The output of the program is shown in figure 2.

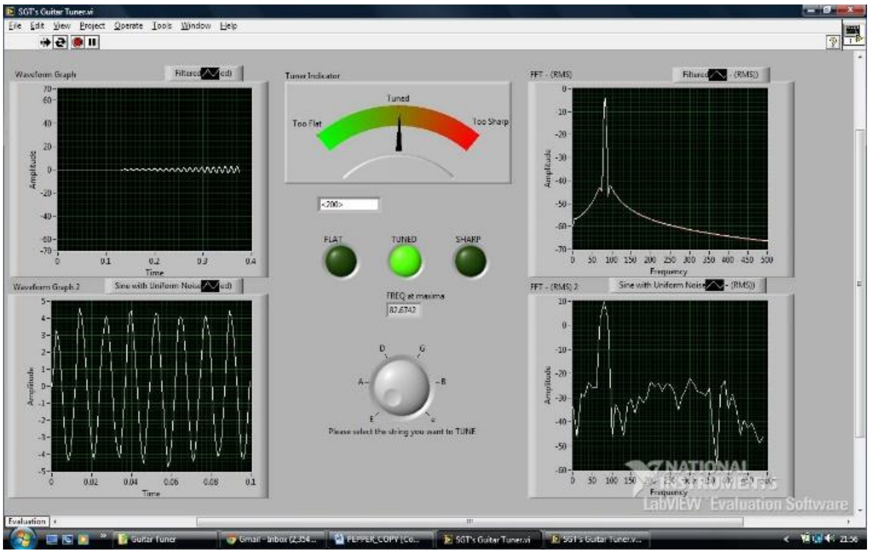

Figure 2: Output showing a tuned E (low)

III. CHORD IDENTIFIER PROGRAM

The methodology of Chord Identifier program is same as that of the Guitar Tuner program, only difference is that instead of one string, a chord is played. As mentioned earlier in order to identify the chord the knowledge of the notes involved in the chord is to be known. Hence our objective is to form a logic on the basis of the notes combination. The signal processing is the same as in the guitar tuner program. Here it is done inside a For loop with a ten-time iteration. The acquired frequencies are stored in an array. These frequencies are basically the in individual frequencies of the notes involved in a chord. Each of these elements are passed in a stacked sequence whose working is based on a case structure. Here each element is compared with the cases to check that to which note it is associated with. In these cases the element of the array is simply compared with a range of frequency that is associated with different notes as mentioned earlier. In this way the notes are identified. Now as we know that different chords will have different notes combination, hence while each chord is played, different notes will be identified. Then using gates we design a logic which is used to assign a unique notes-combination for a particular chord. Table 1 shows the logic used to design the program for different chords.

Table 1: The CHORD-LOGIC table

CHORD	LOGIC
A	(A **AND** E) **OR** C#
B	D #
C	C **AND** E
D	A **AND** D
E	(E **AND** B) **OR** G#
F	(C **AND** A) **OR** F
G	B **AND** (D **OR** G)

Now different chords are naturally played with a unique finger-positioning on the guitar fret-board and hence we get different sounds for different notes and hence different frequencies. It is also worth mentioning that there are variations in the finger positions of different chords. We have considered the most frequently or the most popular finger-positions of playing the chords. Figure 3 is the snapshot of the complete Chord Identifier program in LabVIEW. The output that is the identified chord is shown through seven LEDs which are marked as A,B,C,D,E,F,G on the basis of the acquired knowledge of the notes combination or in other words the logic assigned to each chord as shown above in the table. Figure 4 shows the output of the program-

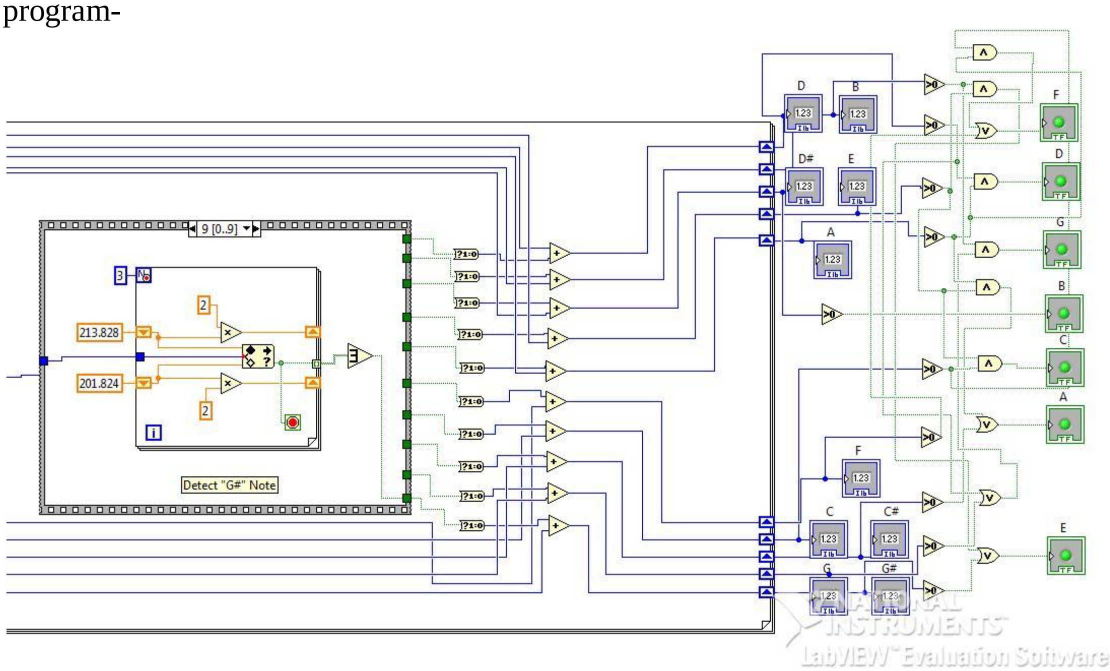

Figure 3: Block Diagram of Chord Identifier

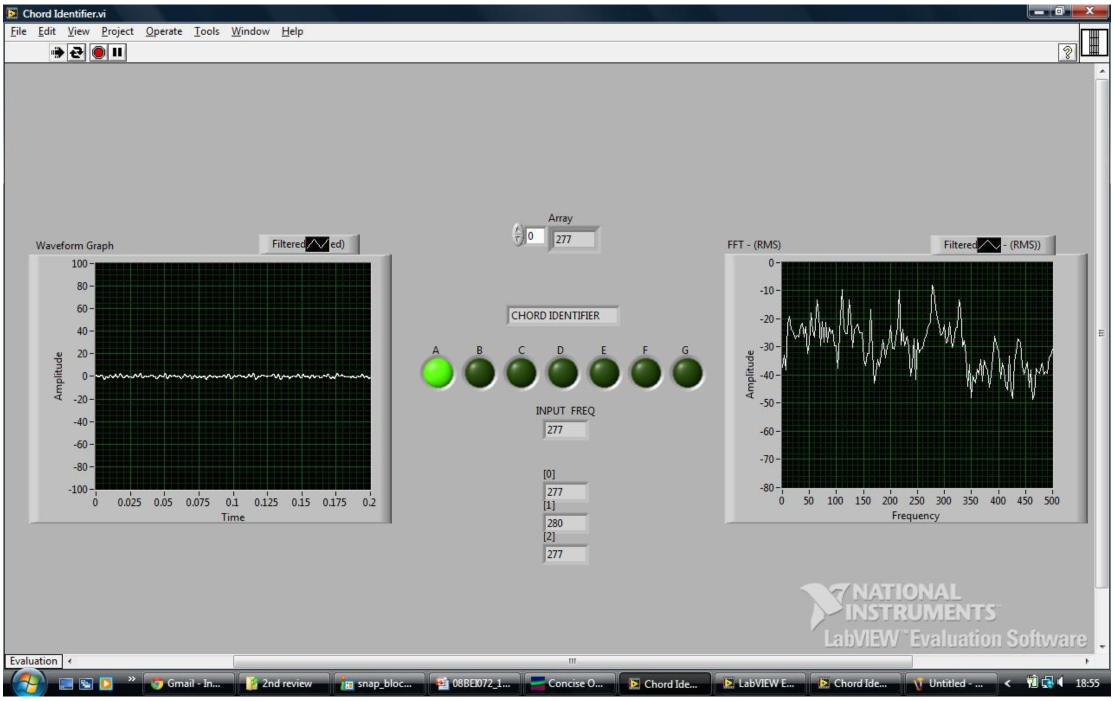

Figure 4: Output showing A chord

IV. ADVANTAGES OF GUITAR TUNER IN SOFTWARE

Well it’s clear that to tune our guitar we just need a PC with an in-built microphone and LabVIEW installed in it and nothing else. So from a Guitarist’s point of view it is naturally more portable than having a separate hardware kit of an electric guitar tuner. He can of course carry a laptop and doesn’t have to take the burden of carrying a separate kit for tuner. And finally it’s cheaper as not a single penny had to be invested in order to make the programs.

V. CONCLUSIONS & FURTHER RECOMMENDATIONS

Thus we have successfully designed the Acoustic Guitar Tuner and Chord Identifier using NI LabVIEW 2010. We discussed the factors involved in tuning an acoustic guitar .The important factor being the fundamental frequency of notes and the method involved in finding them [3]. We have made use of only a personal computer (PC) with in-built microphone, LabVIEW and Pluto acoustic guitar. No other resources are used to accomplish our objective. Even though we have designed our program for an acoustic guitar, the same logic can also be used in Bass guitar. Now a guitar can be played either by fingers or by using a Pick. It is recommended that a Pick is used for playing the guitar while running the programs because a pick will generate more volume of the sound of the string or chord than fingers. If the volume is more, then the microphone can sense the frequencies of all the notes distinctly and we won’t get any undesired results. Another important recommendation is that the microphone should be kept at a suitable distance and orientation. Otherwise the frequencies of a few notes might not reach the microphone and the chord won't be detected even if it is played. Also the chord has to be played by a person having the knowledge of playing guitar since while playing the instrument it is very natural that a string might unknowingly be muted. And as a result the chord won't be detected.

And finally the most important recommendation is that the program is run in a silent environment. The microphone won't be able to distinguish the sound of the guitar from the external disturbances. Hence a noisy environment can lead to improper functionality of the program.

REFERENCES

[1] Fuzzy Automatic Guitar Tuner, KouroshRahnamai, Brian Cox, Kevin Gorman, Department of Electrical Engineering, Western New England College, Springfield, MA 01119, USA

[2] Automatic Acoustic Guitar Tuner, Alfredo Bocanegra, Massachusetts Institute of Technology, June 2005

[3] A Digital Guitar Tuner, Mary Lourde R., Anjali KuppayilSaji, Department of Electrical & Electronics Engineering, BITS-Pilani, Dubai, UAE

[4] Gorrell, L. & R. S, "Automatic Guitar Tuner", Department of Electrical and Computer Engineering, University of Illinois at Urbana-Champaign, December 2002.

[5] Motorized Guitar Tuner, Michael Weißensteiner, Robert Viehauser., University of Technology, Graz, Austria, Telematics