

Design and Realization of the Guitar Tuner Using MyRIO

Patrik Šarga*, Daniel Demečko

Technical University of Košice, Faculty of Mechanical Engineering, Letná 9, 042 00 Košice, Slovak republic

*Corresponding author: patrik.sarga@tuke.sk

Abstract The digital tuner converts the continuous analog audio signal recorded by the microphone into the sampled digital discrete signal. The digital signal is converted to the frequency using the functions and the nearest tone is determined [1]. Many digital tuners, after sampling the analogue signal, filter out noise and accept sound from a certain level of volume. The result is displayed on the output device, which can be a LED indicator, a display, or a monitor. As a digital tuner can also serve a computer, tablet, or mobile. Our goal was to design and create a digital tuner with myRIO and use the LabVIEW programming environment.

Keywords: digital tuner, myRIO, LabVIEW

Cite This Article: Patrik Šarga, and Daniel Demečko, “Design and Realization of the Guitar Tuner Using MyRIO.” *Journal of Automation and Control*, vol. 5, no. 2 (2017): 41-45. doi: 10.12691/automation-5-2-2.

1. Principle of Functioning of Digital Tuner

In The principle of functioning of digital tuner consists of recording the sound and its digital conversion. Based on the digital signal, the tuner determines the measured frequency and its deviation from the required frequency of the tuned string [1]. This deviation along with the measured frequency is then displayed using the UI. Figure 1 illustrates this process.

Our solution for a digital guitar tuner is based on the usage of myRIO. A microphone is attached to the device to record the frequency of individual strings of the guitar. For the functionality of this device, we need to design a program in LabVIEW.

1.1. Microphone Selection

The most suitable type of microphone for sound measurement is a condenser microphone with a narrow directional characteristic that does not record noise and tones emitted by another source. Its frequency characteristic should exhibit minimum deviations within the frequency range

of the guitar strings, from 82Hz to 330Hz. The microphone will be connected by a 3.5mm jack connector. This parameter complies with the HAMA 57153 microphone with a frequency range from 30 Hz to 16 000 Hz, impedance of 1.4 kΩ and sensitivity of 62 ± 3 dB [2,3].

1.2. Conversion of the Analog Signal to Digital

A device that converts the analog signal to digital is called the A/D (analog-to-digital) converter (Figure 2). The A/D converter works at the sampling rate f_s (e.g., 44.1kHz) and converts the analog signal to a known bit resolution (e.g., 16bit). These parameters indicate the resulting quality of the record. The sampling frequency indicates the sampling rate, or how many times per second, the instantaneous value of the analogue signal will be recorded. By digitizing we can never imitate the analog signal perfectly, it is only possible to get closer to it as much as possible. Digitalization of a simple audio signal is more accurate than digitalization of the complex signals [1].

For our solution, it will not be necessary to use an external A/D converter because myRIO analog inputs are connected to a FPGA chip of a microcomputer that is working as an A/D converter and will be programmed in LabVIEW [4,5,6].

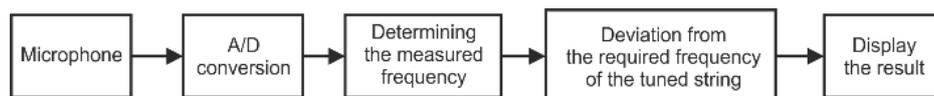


Figure 1. Principle of functioning of digital tuner

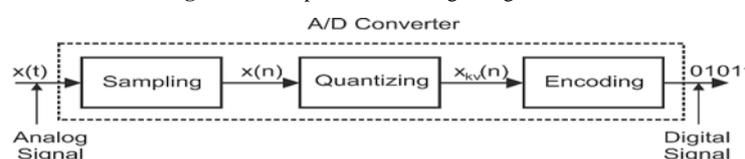


Figure 2. Scheme of A/D converter [7]

1.3. myRIO – Embedded Device

The National Instruments myRIO-1900 is a portable reconfigurable I/O (RIO) device that students can use to design control, robotics, and mechatronics systems. (Figure 3). The NI myRIO-1900 provides analog input (AI), analog output (AO), digital input and output (DIO), audio, and power output in a compact embedded device. The NI myRIO-1900 connects to a host computer over USB and wireless 802.11b.g.n. [5].

A Description of Figure 3:

1. NI myRIO 1900,
2. myRIO Expansion Port (MXP) Breakouts,
3. Power Input Cable,
4. USB Device Cable,
5. USB Host Cable,
6. LEDs,
7. Mini System Port (MSP) Screw-Terminal Connector,
8. Audio In/Out Cables,
9. Button.

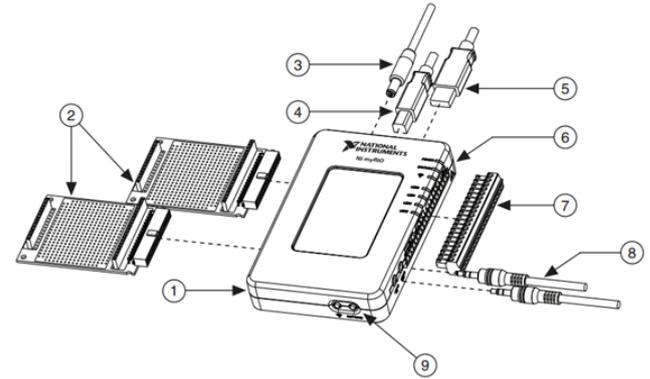


Figure 3. Description of NI myRIO-1900 [5]

The Figure 4 shows the arrangement and functions of NI myRIO-1900 components.

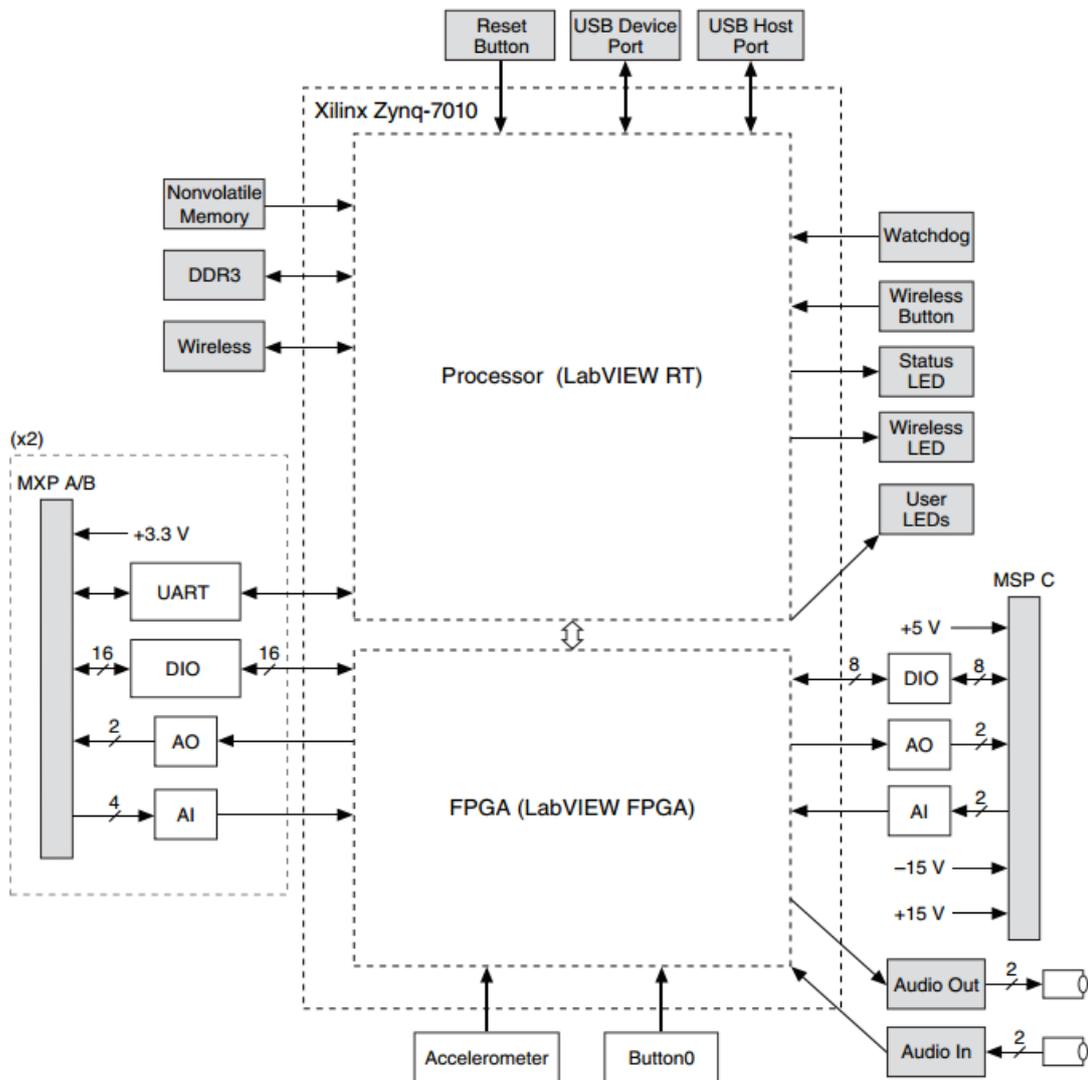


Figure 4. NI myRIO-1900 Hardware Block Diagram [5]

MyRIO will be programmed by the LabVIEW programming environment. The advantage of programming in LabVIEW is that after simple modification of the

software created for myRIO, it can also be used on PLCs or microcontrollers. This advantage reduces the time needed to put the application prototype into operation. [5]

2. Realization of a Digital Tuner with myRIO

For myRIO operation, power supply from 7-12 VDC is required. The adapter provided with the device was used during the installation. The microphone was connected to the myRIO via a 3.5 mm standard JACK connector. Analog input to device A1 was used. PC with LabVIEW is connected to myRIO through USB Type B connector. The connection of the individual components is shown in Figure 5.



Figure 5. Block scheme

2.1. Creating a Software in LabVIEW

The benefits of creating of software in LabVIEW include the short time required for design, high reliability and optimization of the resulting software. Programs designed in LabVIEW support asynchronous programming and parallel run of the program on multiple processor

threads. However, the programmer does not have to deal with them, the development environment suggests and optimizes them when creating a program. The program can be distributed and assembled for a variety of devices [4].

The resulting software records the analog audio signal captured by the microphone and converts it into digital form. Based on the frequency, it determines the closest desired guitar string tone frequency and its deviation from the measured value. This deviation is displayed in the graphical interface. The measurement is repeated in the endless loop of the program.

In Figure 6 shows a flowchart showing the principle of the tuning operation.

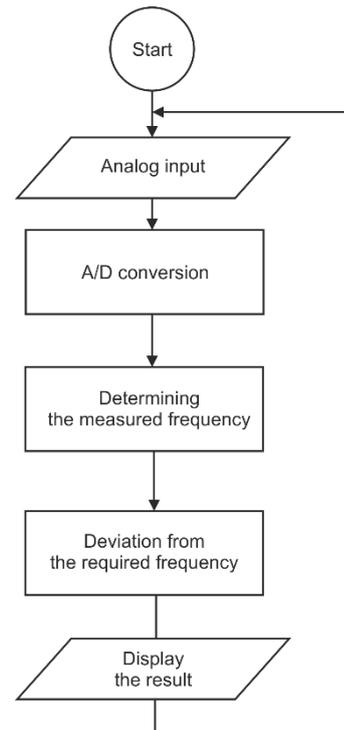


Figure 6. Flowchart

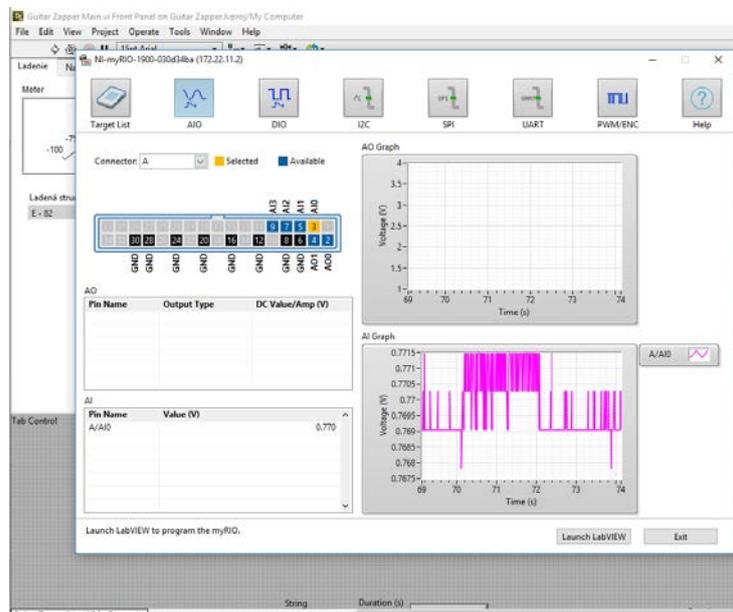


Figure 7. LabVIEW environment for myRIO

Figure 7 shows the LabVIEW environment used to activate and verify myRIO functionality after hardware connection.

2.2. User Interface and Programming Environment

The resulting software consists of three tabs: Tuning, Settings, and Graph. In Figure 8 shows the main window for tuning. The dial on the meter shows a deviation from the desired frequency of the tuned string. The range of the meter is within ± 5 Hz of the desired frequency, one main scale step is 1Hz. The center of the meter is a zero deviation from the desired tuned frequency. When zero deviation is reached, we consider the string to be tuned, which also indicates the background color that changes to green. In addition to the meter, the current numerical value of the frequency and the desired frequency of the tuned string are displayed along with its music label for easier understanding.

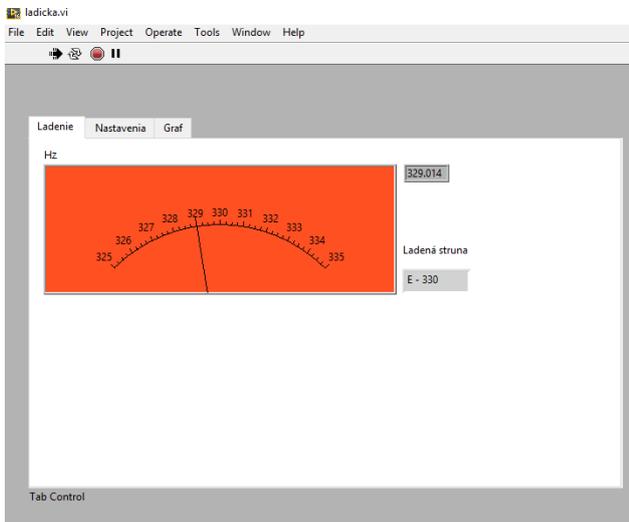


Figure 8. Tuning tab

On the Settings tab, the user has the option to change the sampling rate.

After making some changes in the software, the programmer can set the tuner to allow for multiple musical instruments to be tuned. These changes must be made in the Settings tab.

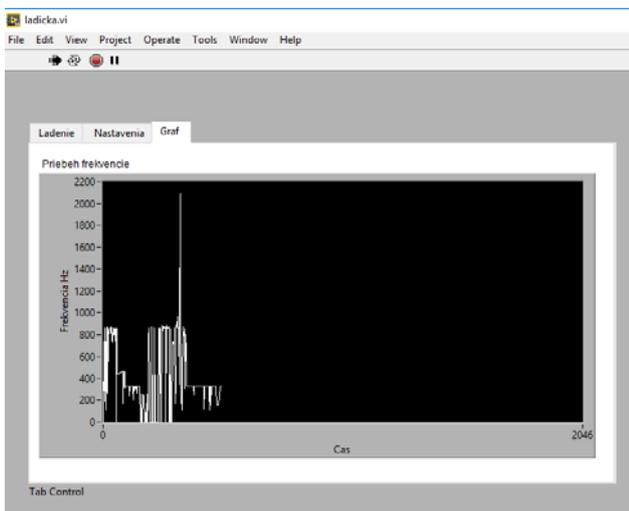


Figure 9. The Graph tab

The Graph tab is informative, showing the change of frequency real-time throughout tuning (Figure 9).

In Figure 10 and Figure 11 show part of the block diagram of the final software for a digital guitar tuner.

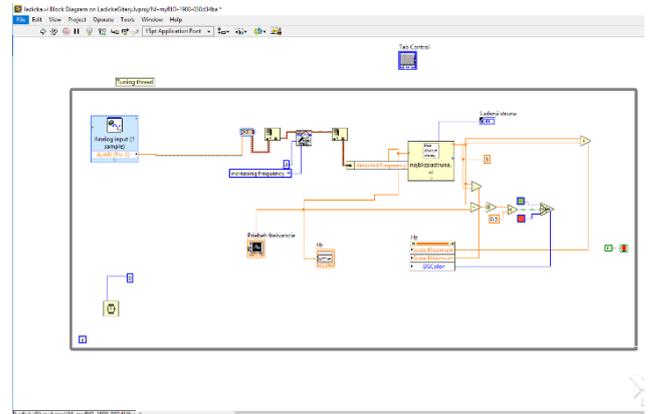


Figure 10. Block diagram in LabVIEW

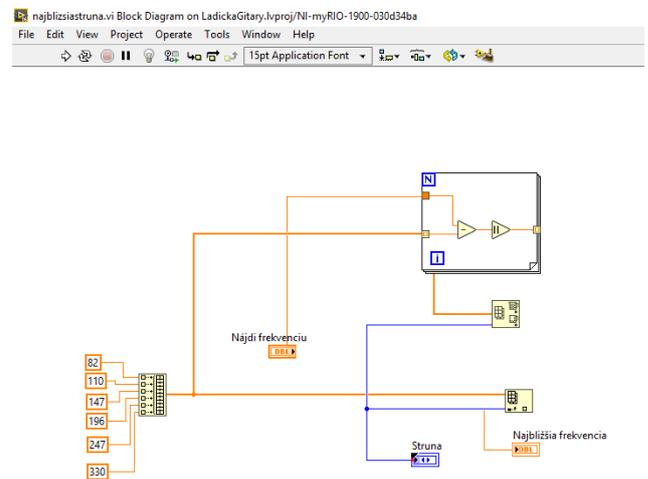


Figure 11. Block diagram for detecting of a tuned string

After completing the software, we have verified it. We verified the tuner with the sound of the desired frequency generated in the smartphone application. This verification was followed by a practical guitar tuning test that confirmed the functionality of our solution.

3. Conclusions

The aim was to demonstrate myRIO's capabilities in conjunction with the LabVIEW development environment. This is demonstrated by the example of creating a digital tuner.

The software converts the analog signal into a digital form. Subsequently, the digital signal adjusts to user-friendly information displayed in the graphical interface of the program. Designed digital tuner can be expanded for tuning several types of musical instruments.

The solution could also be used for other types of problems. By modifying the input device and the software, it would be possible to process other physical quantities whose analysis could bring a high utility value, for example, in the modal analysis of technical constructions.

Acknowledgements

This work was supported by grant projects VEGA No. 1/0393/14.

References

- [1] Brüel & Kjør. - Measuring sound. [online]. 1984.
<http://www.bksv.com/doc/br0047.pdf>.
- [2] BRANDSTEIN, Michael - Microphone arrays: signal processing techniques and applications. Berlin: Springer, 2001, ISBN 3-540-41953-5.
- [3] Brüel & Kjør. - Microphone Handbook vol.1. [online]. 1996,
<http://www.bksv.com/doc/be1447.pdf>.
- [4] LARSEN R. – LabVIEW for Engineers, Boston: Prentice Hall, 2011, ISBN 978-0-13-609429-6.
- [5] NATIONAL INSTRUMENTS – NI myRIO-1900 User guide and specifications [online].
<http://www.ni.com/pdf/manuals/376047a.pdf>.
- [6] NATIONAL INSTRUMENTS – NI myRIO Project Essentials Guide [online].<http://www.ni.com/myrio/project-guide>
- [7] TAN, Li. - Digital Signal Processing - Fundamentals and Applications. [online].
<http://app.knovel.com/hotlink/toc/id:kpDSPFA003/digital-signal-processing-2/digital-signal-processing-2>.