A MATLAB GUI for Engineering Education in the Undergraduate Laboratory

Ernesto Momox* and Luz María Alonso-Valerdi

***Abstract*—Physics and engineering combine various subjects and disciplines into one single umbrella. Technology and computer engineers working in electronic instrumentation aim at making measurements quickly, and accurately using skills and knowledge that they mainly acquired at university by means of hands-on lab experiments. Teaching undergraduate students about (1) computer interfacing, and (2) instrument control techniques for collecting and processing data to automate processes are subjects that seldom appear in their curriculum. A computational software package that allows rapid prototyping is MATLAB. In this paper, therefore, we describe a graphical user interface (GUI) developed for controlling commonly available undergraduate lab electronic test instruments. In addition, we move towards the digital signal processing, providing a teaching lesson for biomedical engineering students.**

***Index Terms*—MATLAB, GUI, electronic instrumentation, impedance, data acquisition, digital signal processing, biosignals, Python, higher education**

I. Introduction

Technology and computer graduates regularly need to use skills beyond their knowledge of basic physics and engineering. Teamwork, scientific writing, applying engineering concepts to interdisciplinary problems, designing and developing products, and managing complex projects are all acquired skills. Due to the unequivocal fact that engineering is plural, graduates use their knowledge and skills in number-crunching, simulations and modelling, physics, games, and of more immediate interest, in electronic instrumentation for data acquisition, automation, and processing; among many other fields. Instrumentation is typically demanded in manufacturing facilities, process plants, research centres, and in academia. Before their major in technology and computer engineering, undergraduate students must take a number of lab assignments that are focused on a variety of analogue and digital circuit design experiments. Thus, learning how to use electronic test instruments such as a digital oscilloscope, and a waveform generator is just one of the first steps in the engineering lab curriculum; some of them will enter today's electronics industry after they graduate. A number of practical applications of an oscilloscope in the undergraduate lab can be readily found in the literature [1–3], even smartphones and iPads can now be used as virtual oscilloscopes [4, 5]. However, many of them would benefit from a wider and deeper knowledge of computational-analysis tools, particularly industry-standard packages introduced in the undergraduate curriculum [6]. In the electronics test instruments control, two packages dominate, namely, MATLAB and LabVIEW. The advantages of implementing these packages as a teaching tool have been discussed by Sharp *et al*. [7, 8] for the former, and the dis advantages of implementing these packages as a teaching tool have been discussed by Tiernan [9] for the latter. In time-consuming processes or in advanced measurements [10], developing a GUI may be the best approach for reliable data acquisition and processing; a responsibility that more often than not rests with information technology engineers.

Several professionals in education have opted for software purchase rather than software development. However, software acquisition from outside vendors is a practice that only temporally catches the students' attention and motivates them for a very short time span [11].

Furthermore, a review undertaken by Reeves and Crippen showed that these proposals do not promote the design and development in engineering students, what should be a priority to arouse the intellectual curiosity [12]. Hence, the development of their own acquisition and processing setups seems to be a convenient strategy to encourage both divergent and convergent thinking in engineering students [13]. Moreover, computational thinking has been pointed out as a key component for effective learning [14, 15].

A number of researchers have reported the development of MATLAB GUIs for different engineering disciplines that are to be used by educators and students in their final years as a teaching-learning tool [16–20]. However, students exceptionally write the GUI.

With these reasons as a motivation, we firstly lectured and trained first year engineering undergraduates in using MATLAB for developing a GUI to interactively and remotely control a digital storage oscilloscope (Tektronix TDS 2022), and an arbitrary waveform generator (Agilent 33521A) in order to automate electrical signals acquisition. Secondly, basic instrumentation by highlighting the non-intuitive effects of impedance matching was an addressed topic. After engaging students with the aforesaid, we presented them a lecture on biosignals and their processing—as a way to illustrate some real-life engineering application. The GUI that was written along with its components is discussed in Section II. Section III briefly describes our lectures for communicating with the instruments and how to route the electrical signal from the generator to the scope. Section IV is intended for students and instructors who want to dive further into digital signal processing.

Manuscript received September 1, 2022; revised November 12, 2022; accepted December 1, 2022.

Ernesto Momox is with Tecnológico de Monterrey, Querétaro, Santiago de Querétaro, 76130, Mexico.

Luz María Alonso-Valerdi is with Tecnológico de Monterrey, Nuevo León, Monterrey, 64849, Mexico.

*Correspondence: momox@tec.mx (E.M.)

doi: 10.18178/ijiet.2023.13.5.1880

II. MATLAB GUI

MATLAB [21] offers different ways to build GUIs; users can create them with the App Designer [22], using GUIDE (although this environment will be removed in a future MATLAB release), and programmatically. In this work, our students wrote their GUI programmatically (we have successfully used this concise technique in M-learning [23]) because its layout and behavior are coded using MATLAB commands and functions rather than in a drag-and-drop environment where objects must be linked to code anyway (callbacks).

The GUI has two tabs: *Function Generator* and *Oscilloscope*. In the first tab, the student can select a waveform type and enter its corresponding signal parameters; a pictorial representation of such parameters is also displayed to facilitate understanding their role. In the second tab, the student can read the electrical signal from the oscilloscope and to put some interpretation on its features (e.g., amplitude, period, phase, etc). Familiar options such as zoom in, zoom out, cursor, etc., are also available. See Fig. 1 and Fig. 2, respectively.

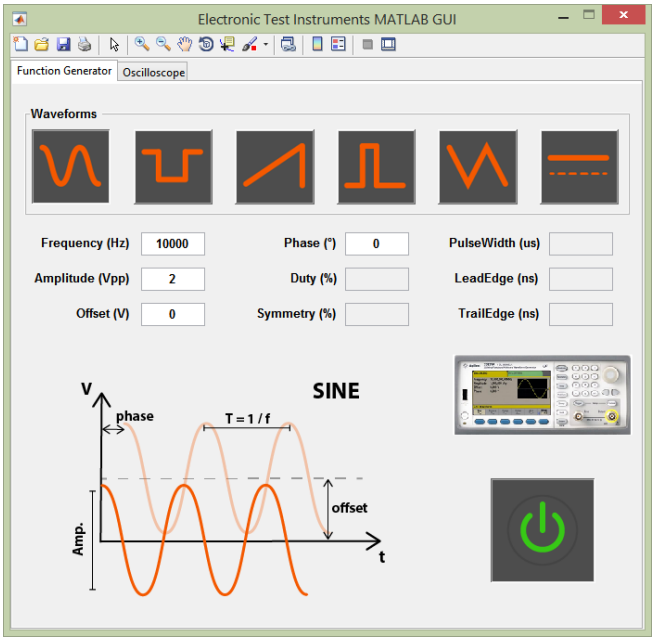

Fig. 1. MATLAB GUI for the undergraduate engineering lab. Tab for controlling the arbitrary waveform generator (Agilent 33521A).

A. Function Generator

This tab has a button group that contains six push-buttons which correspond to different waveforms produced by the signal generator (students tried their best at giving the GUI an intuitive appearance and functionality). The buttons in this group are mutually exclusive and when one is selected, its corresponding signal parameters are enabled. For instance, when the *sine* waveform is clicked, the frequency, amplitude, offset, and phase parameters are the only ones the user can modify (see Fig. 1). When the user selects another waveform, the diagram that illustrates the role of each parameter changes accordingly. The On/Off button sends parameters and commands to the generator. It is worth mentioning that the default appearance of the buttons was modified, images were embedded in the push-buttons by overwriting one of their properties, *CData*. With this change, buttons have a self-explanatory functionality, and the GUI, ease of use.

B. Oscilloscope

This tab has a button (*auto set*) that when pushed sends commands for the scope to display a visible and stable signal. The *auto set* button is very convenient because students or users (without training) often try to obtain a signal by randomly twisting knobs and pressing buttons on the scope's panel. There is another button for importing waveform data to the PC (data acquisition), usually 2500 data points. In this tab, the volts-per-division (V/div) as well as the seconds-per-division (s/div) that are returned by the scope are displayed here; these values are relevant for interpreting the data. The screen of the oscilloscopes has a graticule with ten horizontal divisions (time), and eight vertical divisions (amplitude), see Fig. 2.

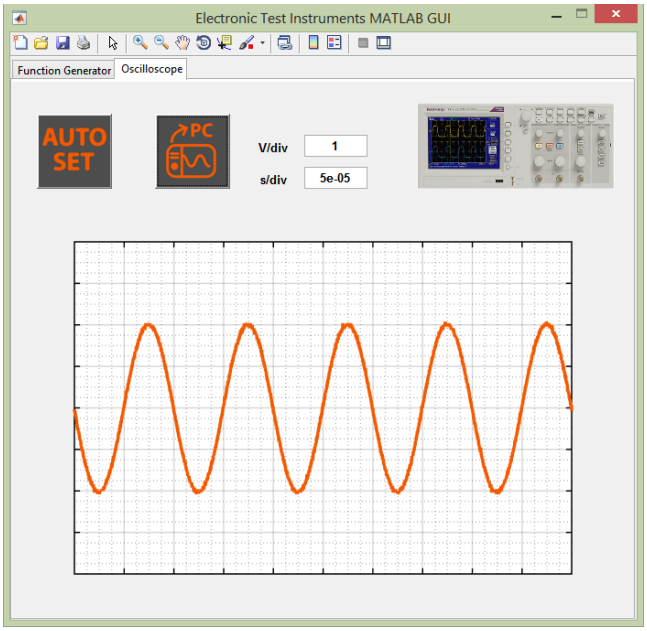

Fig. 2. MATLAB GUI for the undergraduate engineering lab. Tab for the waveform acquisition from oscilloscope (Tektronix TDS 2022).

III. Interfacing Test Instruments to a PC

MATLAB can control and communicate with test and measurement instruments via instrument drivers and text-based commands over commonly used communication protocols such as GPIB, VISA, TCP/IP, and UDP. In this paper, students used MATLAB R2015a and the Instrument Control Toolbox Version 3.7. On the other hand, in order to use electronic test instruments effectively, it is necessary for students to understand basic measurement theory [24] (e.g., electrical quantities, power, direct and alternating current, waveforms, decibels, measurement error) and how it relates to practical experiments [25]. A discussion of this lies beyond the scope of this manuscript. This section, however, will focus on the connection between instruments and a PC, and electronic equipment (waveform generator—oscilloscope). They programmed, constructing a *visa object*, the waveform generator via the local area network (LAN) whilst communication with the oscilloscope was established through a GPIB-USB board constructing a *gpib object*. A pictorial representation of the lab equipment can be seen in Fig. 3. Universities generally have this standard laboratory equipment available for students and tests.

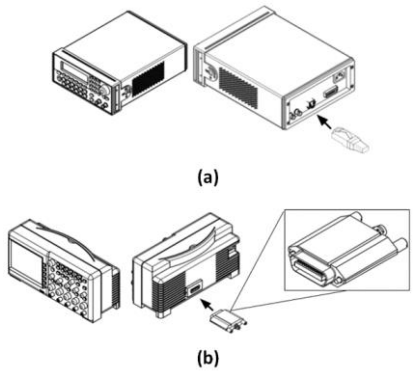

Fig. 3. (a) Agilent 33521A 30 MHz arbitrary waveform generator (communication via LAN). (b) Tektronix TDS 2022 200 MHz oscilloscope (communication via GPIB-USB).

A. Instrument Drivers

In order to establish a connection between the electronic test instruments and a PC running MATLAB, a couple of drivers must be downloaded and installed. For the communication via LAN, the IP address of the waveform generator can be obtained from the front panel using the following simple sequence: System >> I/O Config >> LAN Settings >> IP Address, see Fig. 4. As for the communication via GPIB, the GPIB address of the oscilloscope must be at hand, to retrieve its address, in MATLAB, go to Instrument Control >> GPIB >> Scan, see Fig. 5.

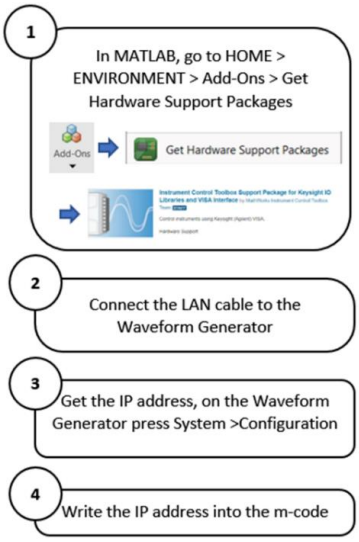

Fig. 4. Steps for obtaining the drivers to remotely control the waveform generator.

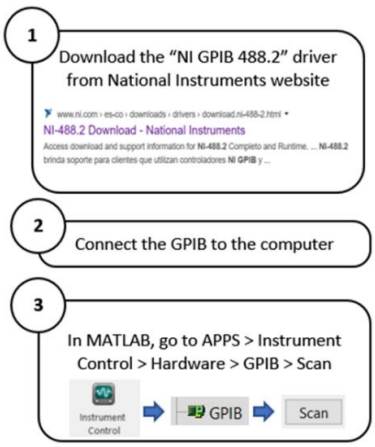

Fig. 5. Steps for obtaining the drivers to remotely control the oscilloscope.

B. Connection of the Waveform Generator to the Scope

Although it is assumed that the reader and most technology and computer undergraduates are familiar with basic electrical principles, the fundamental concepts of instrumentation such as the so-called loading effect and impedance mismatch were taught during lab sessions and are briefly reviewed here. These effects are of importance when two circuits are connected [25, 26]. In our case, the source is the waveform generator and the oscilloscope plays the role of the load. Thus, to make the loading effect small, the oscilloscope has high-impedance inputs (1 MΩ) whereas the output impedance specified by the waveform generator manufacturer is 50 Ω. This difference causes an impedance mismatch which may lead to erroneous measurements if not properly accounted for. For instance, if the waveform generator is connected to the oscilloscope via a 50 Ω coaxial cable (which is standard practice in the lab), as shown in Fig. 6, the amplitude of the electrical signal will double unless a 50 Ω matching termination is used as pictorially presented in Fig. 7. A number of engineering applications may benefit from this effect nevertheless. This non-intuitive behavior can be readily understood from the expressions for the transmission and reflection coefficients [27–30], see Eq. 1 and Eq. 2, respectively.

$$\rho_t = \frac{2z_t}{z_t + z_i} \tag{1}$$

$$\rho_r = \frac{z_t - z_i}{z_t + z_i} \tag{2}$$

When the electrical signal encounters a medium with different impedance, then part of the signal will be reflected back into the original medium (z_i) and part of the signal will continue to propagate into the new medium (z_t). Now, the setup presented in Fig. 6 can be analyzed. In this case, $z_i = 50\ \Omega$ (impedance of the waveform generator), and $z_t = 1\ \text{M}\Omega$ (impedance of the oscilloscope, which for algebraic practical purposes, we will assume as infinity); this yields $\rho_t = 2$ and $\rho_r = 1$. The coefficients are defined as $\rho_t = V_t/V_i$, and $\rho_r = V_r/V_i$, where V_t, V_i, and V_r represent the transmitted, incident, and reflected voltage signal, respectively. Then, with this line of reasoning, we can see that $V_r = V_i$ (a signal with similar features as the incident one is reflected back into the waveform generator), and $V_t = 2V_i$ which indicates that the signal the oscilloscope displays is twice as high as produced by the waveform generator. This scenario can be corroborated from Fig. 1. The waveform generator was set up with a peak-to-peak voltage $V_{pp} = 2\text{V}$ (see the field *Amplitude*), however, Fig. 2 shows a signal with $V_{pp} = 4\text{V}$, as explained above.

As for Fig. 7, when the termination pad is plugged in between the function generator and the oscilloscope, z_t is a parallel impedance (1 MΩ || 50 Ω) equal to 50 Ω; which means that $\rho_t = 1$ (the input signal is fully transmitted, $V_t = V_i$) and $\rho_r = 0$ (no part of the signal is reflected back, $V_r = 0V_i$). Fig. 7 is, in fact, the proper interconnection for setting and analyzing a required waveform via the oscilloscope.

In our university and during hands-on instrumentation

activities in the lab, we have observed that impedance matching is misleading, and undergraduates are puzzled until they are explicitly instructed about the importance of this phenomenon and about the GUI development for accurate data and signal acquisition.

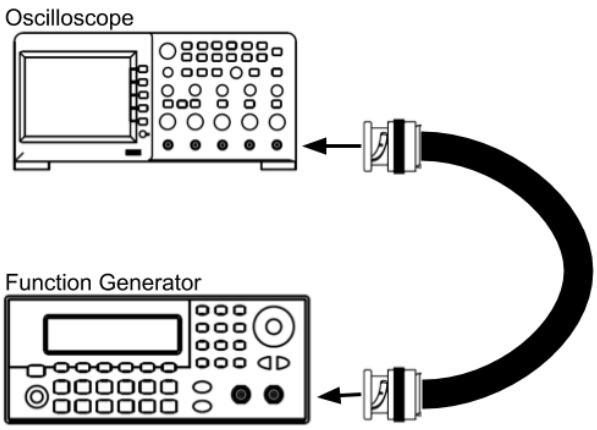

Fig. 6. Impedance mismatch between the waveform generator (50 Ω) and the oscilloscope (1 MΩ); the amplitude of the measured signal will be twice as high as that of the actual signal generated by the source.

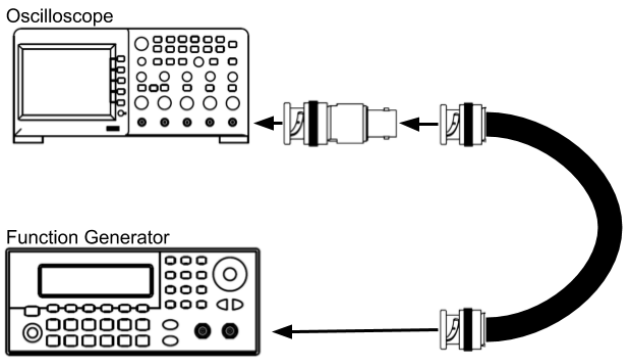

Fig. 7. The waveform generator is impedance matched to the oscilloscope using a termination pad (dummy load); this does not give rise to reflections.

IV. Digital Signal Processing

Having acquired signals, what follows is their processing. Digital signal processing (DSP) refers to the mathematical manipulation of signals to obtain the greatest amount of information about the mechanisms that generate them. At present, one of the most prolific scientific areas is the DSP [31]. The growth of this discipline has been such that it has been expanded to other sciences such as Neurosciences, Psychology, Therapies and Rehabilitation, Diagnostic Medicine, Mechatronics, Control and Automation, among others [32].

Unfortunately, teaching techniques for effective DSP learning have lagged behind. Frequently, students fail to apply fundamental computational methods of signal and system modelling with applications to selected case studies such as the implementation of brain-computer interfacing systems, which involves acquisition, processing, translating and use of data in real-time mode [33]. For this reason, there is an urgent need to develop tools that allow instructors to efficiently illustrate the DSP process. This process can be undertaken in three fundamental stages: (1) data acquisition, (2) time analysis, and (3) frequency analysis [34].

In the present paper, the development of a GUI for basic instrumentation and data acquisition has been the main topic. However, as our future work in engineering education, the GUI can be extended to account for time and frequency analysis. Regarding time-domain analysis, five basic features are being taken into account. These are: (1) average techniques, (2) zero crossing, (3) peak detection, (4) digital filtering, and (5) power estimation. To compare all of these parameters, correlation and convolution should be included, as well. On the other hand, frequency analysis is often based on Fourier principles [35]. Therefore, Fourier series and Fourier transform must be computed to analyze and synthesize frequency content of the signals. It is worth mentioning that the bridge between two domains is the Parseval theorem, which establishes that the average energy in time is equal to the average energy in frequency. Demonstration and application of this theorem is similarly of our interest towards the development of teaching tools for DSP.

To propose progressive and attractive teaching methods in DSP, applications to selected case studies such as robotics or speech recognition, among others, are not only very motivating, but they are also challenging and illustrative [33]. Therefore, an introductory DSP teaching lesson for biomedical purposes was delivered and is briefly described in this paper. First of all, a biosignal is a signal coming from any living entity. This signal should be measurable, trackable and analyzable. A biosignal can be either electrical or not electrical. Biosignals have attributes (or features) that provide valuable information to investigate the physical (or even mental) condition of the living entity. The most monitored systems are [36]: (1) dermal, (2) respiratory, (3) cardiovascular, (4) peripheral and central nervous system, and (5) muscular.

The following five biosignals (Figs. 8 to 12) can provide information concerning the functionality of the previously mentioned systems. First, electrodermal signal (Fig. 8) is an indicator of physiological and/or psychological arousal, and the area under the curve (AUC) is a common attribute to be estimated in time domain. Second, breathing signal (Fig. 9) reflects the level of effort to oxygenate the blood and quantity of energy demanded by human body. A usual attribute of this signal is the number of cycles per minute. Third, electrooculography signal (Fig. 10) refers to the degree and amplitude of eye motion that reflects level of attention. Waveforms are usually what is estimated and what defines blinks, eye rotations, and saccades. Fourth, electrocardiography signal is an indicator of the level of functional rhythmicity of the cardiovascular system. Several attributes including heart rate variability, number of cycles per minute, waveforms, and spectral content are frequently of interest. In Fig. 11, the estimation of number of cardiac cycles based on Fourier transform is illustrated. Finally, electroencephalography signal is dynamic and non-linear system that reflects the electrical activity of the central nervous system. A wide variety of features both in time and frequency domains has been proposed so far. As a case in point, the individual alpha frequency that refers to a unique neural pattern that depends on demographic and clinical conditions is shown in Fig. 12. The code to plot all of these biosignals, and to estimate their features has been implemented as a Python notebook in Google Colab. This is available for teaching purposes in [36].

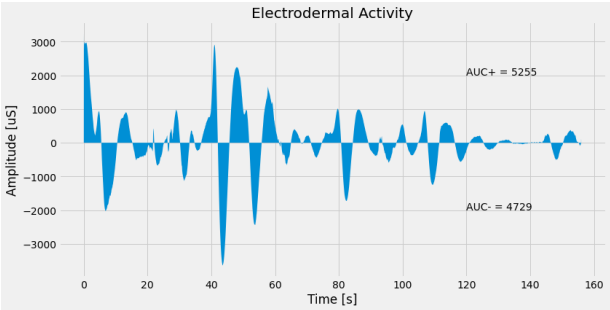

Fig. 8. Biosignal: electrodermal signal. Feature: area under the curve. Sampling frequency: 64 Hz.

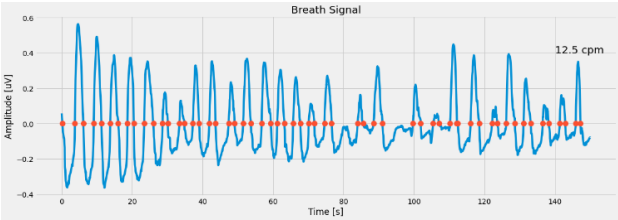

Fig. 9. Biosignal: breathing signal. Feature: number of cycles per minute. Sampling frequency: 200 Hz.

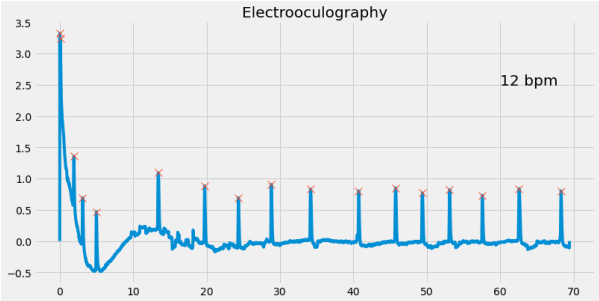

Fig. 10. Biosignal: electrooculography signal. Feature: waveforms and number of blinks per minute. Sampling frequency: 100 Hz.

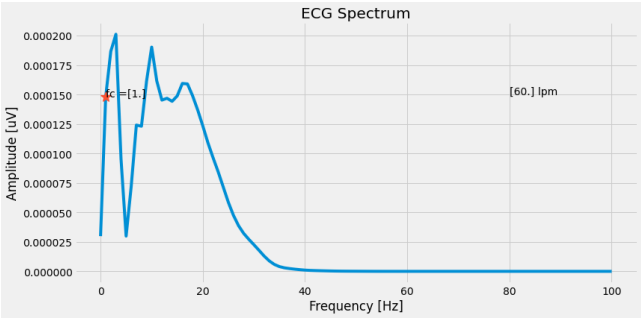

Fig. 11. Biosignal: electrocardiography signal. Feature: power spectral density. Sampling frequency: 200 Hz.

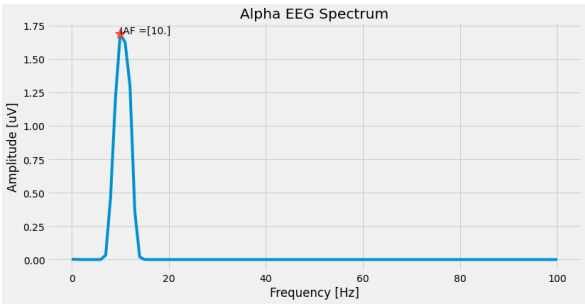

Fig. 12. Biosignal: electroencephalography signal. Feature: peak amplitude of the power spectral density. Sampling frequency: 200 Hz.

In this first approach, the DSP teaching lesson is provided in Python language because it is a feasible programming tool, that does not require a license. However, we are moving towards offering the complete teaching scheme proposed in this paper in MATLAB language. Indeed, we intend to include MATLAB-based open source toolboxes for specific purposes such as EEGLAB [37].

V. Conclusion

Technology and computer engineering is plural because of its broad spectrum of applications in different disciplines and fields. Today's industry demands graduates competent in instrumentation and in computational software packages that are widely used in the private and public sectors as well as in academia.

This paper presented the development of a simple GUI using MATLAB and highlighted the importance of understanding the underlying physical concepts of instrumentation for connecting a function generator to an oscilloscope accounting for proper impedance matching. Understanding this concept is of importance for graduates (working in instrumentation) to avoid committing the heresy of assuming that the electrical signal simply doubles its amplitude when the generator is plugged into the oscilloscope without a proper termination. The GUI is better suited to non-engineering majors and users whom might be puzzled by the many knobs and buttons in the front panel of the electronic test instruments. The basic engineering concepts reviewed here may pave the way for the automation of time-consuming data acquisition processes and represent a competitive advantage for the graduates in the workplace.

On the other hand, this kind of hands-on lab activity merges the disciplines of mathematics, science, engineering, and computing—a lecture's feature that is well received by learners. Furthermore, DSP conceptualization and exemplification in practical scenarios (biomedical) were also briefly introduced as a way to engage students, and to provide evidence of our future work in engineering education.

This paper can be used as a pre-lab lecture on remote data acquisition so that students get acquainted with some of the instrumentation fundamentals and by instructors to tinker their lab sessions for putting modern technology and its applications into the hands of learners as early as possible in their education.

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

All authors provided critical feedback and helped shape the research, analysis, and manuscript. E Momox conceived the study and was in charge of the overall direction and planning. He also conducted the MATLAB and lab sessions. L. M. Alonso-Valerdi trained the undergraduates in DSP.

Funding

This work was supported by Novus (https://novus.tec.mx/en) and Writing Lab (https://writinglab-tec.com/) initiatives within the Institute for the Future of Education that is dedicated to the development of the culture of research in Educational Innovation and to the enhancement of the academic production of the faculty members at Tecnológico de Monterrey, Mexico.

ACKNOWLEDGMENT

A peer-reviewed journal or conference would not survive without the generous time and insightful comments of the reviewers, whose efforts often go unrecognized. For this reason, the authors would like to publicly acknowledge the passion that reviewers bring to deliberations. The authors would also like to extend their gratitude to all study participants for their time and energy spent on this project.

REFERENCES

[1] P. Buyl, "A digital oscilloscope setup for the measurement of a transistor's characteristic curves," *American Journal of Physics*, vol. 78, no. 12, pp. 1425–1429, 2010.

[2] Y. Kraftmakher, "Digital storage oscilloscopes in the undergraduate laboratory," *European Journal of Physics*, vol. 33, no. 6, p. 1565, 2012.

[3] M. F. Masters and R. E. Miers, "Use of a digital oscilloscope as a spectrum analyzer in the undergraduate laboratory," *American Journal of Physics*, vol. 65, no. 3, pp. 254–255, 1997.

[4] K. Forinash and R. F. Wisman, "Smartphones as portable oscilloscopes for physics labs," *The Physics Teacher*, vol. 50, no. 4, pp. 242–243, 2012.

[5] R. Ramos and C. Devers, "The iPad as a virtual oscilloscope for measuring time constants in RC and LR circuits," *Physics Education*, vol. 55, no. 2, p. 023003, Jan 2020.

[6] L. D. Feisel and A. J. Rosa, "The role of the laboratory in undergraduate engineering education," *Journal of Engineering Education*, vol. 94, no. 1, pp. 121–130, 2005.

[7] J. S. Sharp, P. M. Glover, and W. Moseley, "Computer based learning in an undergraduate physics laboratory: interfacing and instrument control using Matlab," *European Journal of Physics*, vol. 28, no. 3, p. S1, 2007.

[8] Z. Lei, H. Zhou, W. Hu, and G.-P. Liu, "Teaching and comprehensive learning with remote laboratories and MATLAB for an undergraduate system identification course," *IEEE Transactions on Education*, vol. 65, no. 3, pp. 402–408, 2022.

[9] P. Tiernan, "Enhancing the learning experience of undergraduate technology students with LabVIEW software," *Computers & Education*, vol. 55, no. 4, pp. 1579–1588, 2010.

[10] A. Platzker, A. Palevsky, S. Nash, W. Struble, and Y. Tajima, "Characterization of GaAs devices by a versatile pulsed IV measurement system," *IEEE International Digest on Microwave Symposium*, IEEE, 1990, pp. 1137–1140.

[11] V. J. Bhute, P. Inguva, U. Shah, and C. Brechtelsbauer, "Transforming traditional teaching laboratories for effective remote delivery—a review," *Education for Chemical Engineers*, vol. 35, pp. 96–104, 2021.

[12] S. M. Reeves and K. J. Crippen, "Virtual laboratories in undergraduate science and engineering courses: A systematic review, 2009–2019," *Journal of Science Education and Technology*, vol. 30, no. 1, pp. 16–30, 2021.

[13] L. J. Hirshfield and M. D. Koretsky, "Cultivating creative thinking in engineering student teams: Can a computer-mediated virtual laboratory help?" *Journal of Computer Assisted Learning*, vol. 37, no. 2, pp. 587–601, 2021.

[14] J. A. Lyon and A. J. Magana, "The use of engineering model-building activities to elicit computational thinking: A design-based research study," *Journal of Engineering Education*, vol. 110, no. 1, pp. 184–206, 2021.

[15] B. Bettin, M. Jarvie-Eggart, K. S. Steelman, and C. Wallace, "Preparing first-year engineering students to think about code: A guided inquiry approach," *IEEE Transactions on Education*, vol. 65, no. 3, pp. 309–319, 2022.

[16] M. S. Nogueira, J. E. Gunther, K. Komolibus, and S. Andersson-Engels, "User-friendly graphical user interface for simulating tissue optical properties and fluence rates: Improving students learning in tissue optics," *Optical Interactions with Tissue and Cells XXXI*, vol. 11238, SPIE, 2020, pp. 45–53.

[17] S. Y. Wong and S. Y. Lim, "Electromagnetic education: Development of an interactive GUI for demonstrating wave polarization," *Computer Applications in Engineering Education*, vol. 28, no. 5, pp. 1190–1219, 2020.

[18] P. H. O. Silva, L. G. Nardo, S. A. M. Martins, E. G. Nepomuceno, and M. Perc, "Graphical interface as a teaching aid for nonlinear dynamical systems," *European Journal of Physics*, vol. 39, no. 6, 065105, 2018.

[19] F. Cambronero-López, A. Gomez-Varela, and C. Bao-Varela, "Designing an ultrafast laser virtual laboratory using MATLAB GUIDE," *European Journal of Physics*, vol. 38, no. 3, 034006, 2017.

[20] M. Victoria, O. M. Querin, C. Díaz, and P. Martí, "liteITD a MAT—LAB graphical user interface (GUI) program for topology design of continuum structures," *Advances in Engineering Software*, vol. 100, pp. 126–147, 2016.

[21] M. R2015a, "The MathWorks, inc." Natick, Massachusetts, United States

[22] C. S. Lent, *Learning to Program with MATLAB: Building GUI Tools*, John Wiley & Sons, 2022.

[23] E. Momox and C. Maio, "Computer-based learning in an undergraduate physics course: Interfacing a mobile phone and matlab to study oscillatory motion," *American Journal of Physics*, vol. 88, no. 7, pp. 535–541, 2020.

[24] B. M. Oliver and M. C. John, *Electronic Measurements and Instrumentation*, McGraw-Hill Book Company, 1971.

[25] R. A. Witte, *Electronic Test Instruments: Theory and Applications*, Prentice-Hall, Inc., 1993.

[26] R. L. Boylestad, *Introductory Circuit Analysis*, Pearson Education, 2010.

[27] M. N. O. Sadiku, *Elements of Electromagnetics*, 7th ed. Oxford University Press, 2018.

[28] J. M. Serra, M. C. Brito, J. M. Alves, and A. Vallera, "A wave lab inside a coaxial cable," *European Journal of Physics*, vol. 25, no. 5, p. 581, 2004.

[29] R. O. Cleveland and J. A. McAteer, "The physics of shock wave lithotripsy," *Smith's Textbook on Endourology*, vol. 1, pp. 529–558, 2007.

[30] P. Pope and A. Winter, "An undergraduate experiment on acoustic impedance," *European Journal of Physics*, vol. 6, no. 1, p. 13, 1985.

[31] R. Ortner, E. Grünbacher, and C. Guger, "State of the art in sensors, signals and signal processing," *Gtecjapan. Com*, pp. 1–18, 2013.

[32] V. Stanford, "Biosignals offer potential for direct interfaces and health monitoring," *IEEE Pervasive Computing*, vol. 3, no. 1, pp. 99–103, 2004.

[33] A. Prochazka, O. Vysata, and V. Marik, "Integrating the role of computational intelligence and digital signal processing in education: Emerging technologies and mathematical tools," *IEEE Signal Processing Magazine*, vol. 38, no. 3, pp. 154–162, 2021.

[34] R. A. Ramírez-Mendoza, J. d. J. Lozoya-Santos, R. Zavala-Yoé, L. M. Alonso-Valerdi, R. Morales-Menendez, B. Carrión, P. P. Cruz, and H. G. Gonzalez-Hernandez, *Biometry: Technology, Trends and Applications*, CRC Press, 2022.

[35] Y. Kraftmakher, "Some applications of Fourier's great discovery for beginners," *European Journal of Physics*, vol. 33, no. 5, p. 1249, 2012.

[36] L. M. Alonso-Valerdi, "Digital biosignal processing, Google Colab," 2022.

[37] R. Martínez-Cancino, A. Delorme, D. Truong, F. Artoni, K. Kreutz-Delgado, S. Sivagnanam, K. Yoshimoto, A. Majumdar, and S. Makeig, "The open EEGLAB portal interface: High-performance computing with EEGLAB," *NeuroImage*, vol. 224, 116778, 2021.

Copyright © 2023 by the authors. This is an open access article distributed under the Creative Commons Attribution License which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited (CC BY 4.0).