Educational Linear Systems Emulation Platform

1st Juliane Ritzel Farret
Federal University of Santa Maria
Santa Maria, RS, Brazil
juliane.farret@acad.ufsm.br

2nd Émerson Isaias da Silva
Federal University of Santa Maria
Santa Maria, RS, Brazil
emersonceesp@gmail.com

3rd Lucas Vizzotto Bellinaso
Federal University of Santa Maria
Santa Maria, RS, Brazil
lucas@gepoc.ufsm.br

4th Rodrigo Varella Tambara
Federal University of Santa Maria
Santa Maria, RS, Brazil
rodrigo.tambara@ufsm.br

Abstract—In the field of engineering and exact sciences, there is a constant demand for experimental verification of control strategies, and modulation schemes, among other aspects. For this reason, undergraduate laboratory classes should have modern approaches to control systems education using simulations and experiments to facilitate the learning process. This paper presents the development of a low-cost hardware-in-the-loop platform for linear system emulation. The platform is an AD-DA system that simulates mathematical models in discrete time. The project consists of software and hardware elements that allow the user to send to the microcontroller the parameters by a PC application. The app sends the sampling frequency and the transfer function to the STM32F103 microcontroller board that runs the difference equations in real time. Experimental results done with first and second-order systems, varying the sampling frequency demonstrated the quality of the didactic platform in control system laboratory classes.

Keywords—Linear systems, Differential Equations, Control systems, Hardware-in-the-Loop, Real Time Simulation, Microcontroller.

I. INTRODUCTION

Control and signal processing systems are increasingly being used in many applications, including vehicle propulsion, flight mechanics, aircraft control, and power distribution. The design of these systems requires significant knowledge in several areas of electrical and computer engineering. This includes understanding electronics, control theory, theory of linear and nonlinear systems, and electromagnetics. Furthermore, knowledge of the behavior of electrical machines, modeling techniques, and microprocessors for implementing control algorithms must be established to create efficient designs [1].

For many students, a lab or teaching platform can provide an initial experience in synthesis, requiring them to use knowledge from across the curriculum. In addition, developing knowledge of simulation tools provides support for creative circuit synthesis and evaluation. A simulator is a powerful way to test new ideas without the difficulties associated with making circuits [2]. Likewise, the study of linear systems requires hands-on demonstration for students to intuitively learn concepts such as system dynamics and stability. This suggests that teaching tools used in the laboratory are of great value as a component of an electronics and computer engineering curriculum.

However, lab experiments, while the best strategy, present problems of time, money, and safety. The use of microprocessors and microcontrollers to implement real-time simulation platforms are present in the literature as alternative options when there are impediments in tests on real systems, due to costs, risks or the need for agility in the tests. However, these solutions are directed to particular case studies where it is desired to simulate only one component of a system, such as inductive machines [3].

Several initiatives emerged to encourage and improve the linear and control systems education. Some of them develop entire teaching modules, with embedded control strategie to be studied, and dedicated hardware and software [4]. Other initiatives develop only simulation programs for power electronics application [5]. Nevertheless, when a general-purpose emulation system is needed, commercial solutions can be extremely expensive, which can be an impediment if the study does not justify the costs of such equipment.

This paper presents a low-cost generic hardware emulator for implementing systems up to order two, for teaching and learning in the area of engineering and exact sciences. An architecture for vector and matrix operations is proposed with a didactic focus on the emulation of linear systems based on the representation in state space, and that is simple enough to be implemented using microcontrollers and less sophisticated platforms so that the cost is smaller than commercial solutions.

II. DEVELOPMENT

The project can be separated into software and hardware that represent mainly the embedded and power part. The software part contains the operating modes that are the main modules of the system. The hardware is mainly divided between the analog input and output circuits and their relation with the microcontroller (Fig.1).

A. STM32F103 Board

The STM32F103C8T6 development board is based on the STM32F103xx mid-density performance line family from STMicroelectronics, with ARM Cortex-M3 core. The board is used to receive the parameters and the computer system through the emulation of a common serial port over the USB
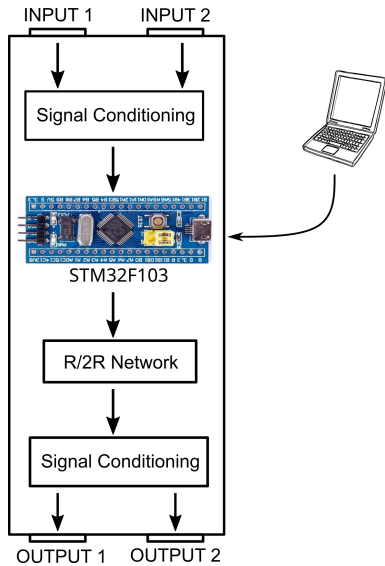
Fig. 1. System overview

interface. STM32F103 was used in this Project because of the Project necessities, such as the processing power acting with timers in real-time, USB communication, and Flash memory recording.

B. Signal Conditioning Board

Firstly, due to the lack of DA converters on the board chosen for processing the emulated systems, it was necessary to design an R/2R network to convert digital information into analog output. Ladder resistor networks provide a simple and inexpensive way to do a digital to analog conversion (DAC). The R/2R ladder was chosen for this work because of its inherent accuracy and easy construction process. In the project presented, the voltage varies from 0 to 3.3V for two digital outputs of 10 bits each. The schematic can be seen in Fig. 2.

For the design of input and output signal circuits, the operational amplifier TL064 was used with the resistor values calculated from the laws of Kirchhoff currents (Fig. 3).
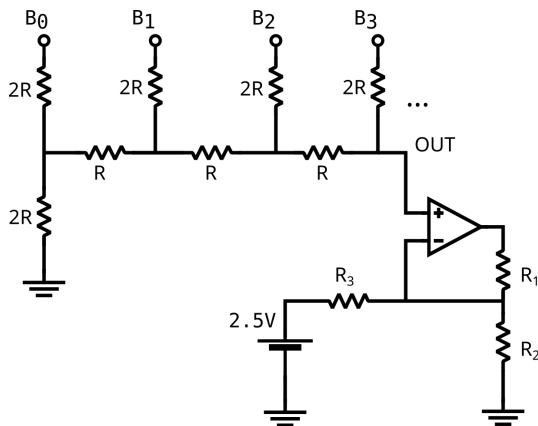

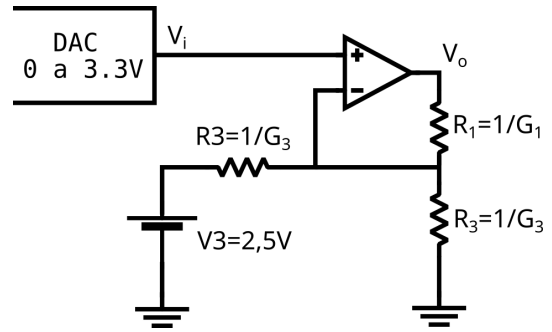Fig. 2. R/2R network DAC converter with digital outputs

Fig. 3. Digital Outputs Circuit

In order to make it possible to use the circuits together and integrate them into the microcontroller, it was realized the need to manufacture a printed circuit board with a shield for coupling. The board was designed using the *Kicad* software, using tracks on both sides of the board due to the volume of components and their connection arrangement. As a means to implement the prototype as close to a final product as possible, SMD components were chosen for assembly due to their more compact size. The PCB used for the tests can be visualized in Fig. 4.
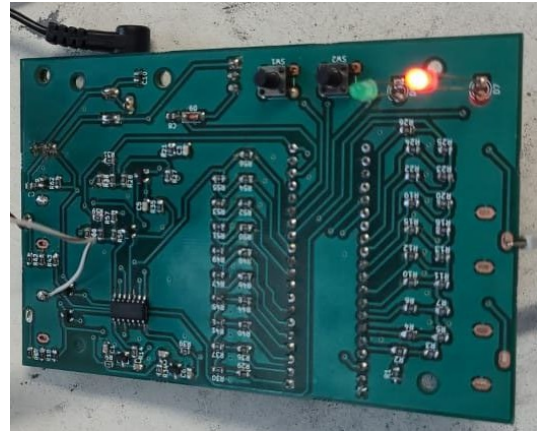

Fig. 4. PCB after soldering the components

C. Power circuit

A symmetrical source of $\pm 12V$ was designed from an external source of 24V to power the operational amplifiers, the TL064 integrated circuit, used in the design of the analog channels of the emulator and the output of $+5V$ for the microcontroller (Fig. 5). For this, voltage regulators LM7912 and LM7805 were used, as well as capacitors and a Zener diode for protection.

D. User Interface

With the idea of processing the separation of the system on the computer to better send the parameters to the microcontroller, a Python application was created to simulate the systems, process the parameters, and interface with the user (Fig. 6).
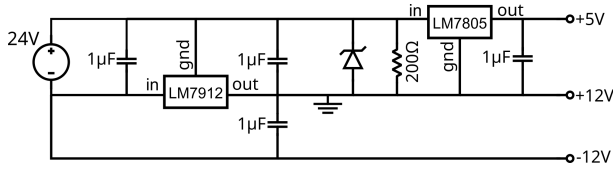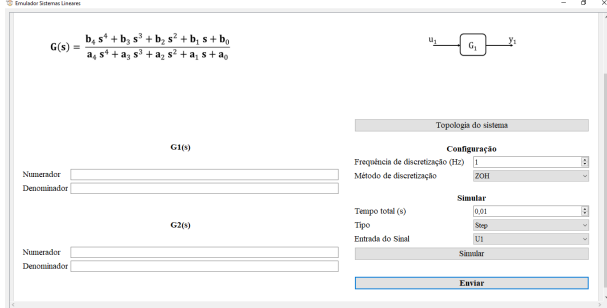
Fig. 5. ±12V Symmetrical Source

Fig. 6. User Interface App

Fig. 7. Inputs and outputs of the emulator system

These systems are sent in the box configuration mode, where the USB port is only read to receive the parameters and when the system is registered in memory (Fig. 7). In addition to the two main modes of operation of the box, there is also the configuration of the timer that occurs when switching between the main states, from configuration to emulation, by pressing both buttons simultaneously for 3 seconds.
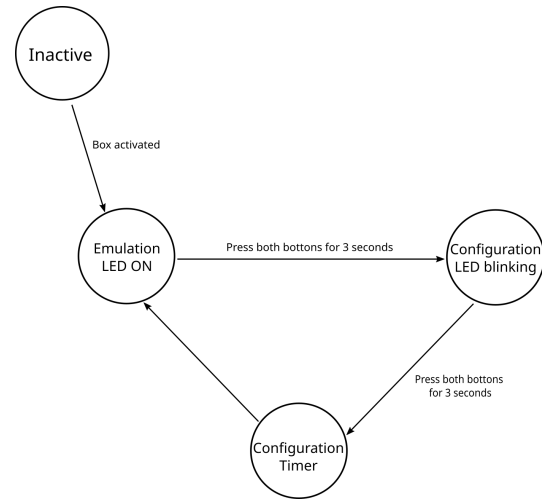

Fig. 8. Finite state machine operation modes

In order to allow users to interact with the system independently, buttons and LEDs were placed. The buttons add a step response to outputs 1 and 2 respectively when pressed separately or change the operating mode when pressed together for more than 3 seconds. The red LED indicates whether the board is powered and the green LED indicates the system's operating modes, which can be configured (blinking at 1Hz) and emulation (fixed).

In addition to the buttons and LEDs on the surface, BNC plugs for the input and output channels and a Jack connector for powering the external 24V power supply plugged into the socket are also included. The micro-USB connector for communication with the computer is specific to the micro-controller. The dimensions of the box were chosen to be as compact as possible and served as the basis for the design of the final printed circuit board.

State 1 (emulation): Start running the system in state space. It only runs the linear system but reads the buttons.

State 2 (configuration): memory system, receives user by user and registers in memory. flashing LED. (User defines which FT will simulate and the timer count value).

One of the functions of the Python application is to calculate the preescaler, and the timer reset, and then send these parameters to the microcontroller via USB. The number of inputs, outputs, system order, and matrix parameters is sent along with the preescaler and timer reset. The sample rate and the emulated transfer function are defined by the user through interaction with the Python application.

E. Operation Mode

The general functioning of the system consists of the operating modes of the box which are the emulation and configuration modes. The box's initial state is inactive, and when it is turned on and taken by the external source, the emulation mode is started, where the values from flash memory are read and the timer interrupt starts to run. The state machine that operates these modes is shown in Fig. 8.

The treatment of the system in the computer serves so that the systems are arranged in order to simulate the space of states in discrete time. This can be done using two algorithms. First when receiving a continuous-time transfer function and the second when receiving a discrete-time transfer function.

After processing these systems into the state space, the parameters are set in the string defined to be sent to the microcontroller, where it will be written to the flash memory at the addresses defined by the USB protocol. The equations below demonstrate the nomenclature used in the pointers used in the communication protocol to send these parameters. Equation (1) defines the inputs that can be up to two and second-order systems and (2) defines the two possible outputs.

$$\mathbf{x}[k+1] = \mathbf{G}x[k] + \mathbf{H}\mathbf{u}[k] \tag{1}$$

$$\mathbf{y}[k] = \mathbf{C}x[k] + \mathbf{D}\mathbf{u}[k] \tag{2}$$

State 3 (configuration timer): Retrieve string from memory (read from flash memory and start running on timer interrupt), convert to (sscanf function), then assign the read values to the variables of the controller (int variables).

III. Tests and Results

To generate the results, a test environment was created with the external source connected to the board, the signal generator at the system input, the oscilloscope for verification, and the computer for sending the parameters. The systems model was simulated with Python software to compare the simulation response with the oscilloscope results.

As these are slow systems with a low sampling rate were tested at frequencies of 100 Hz, 1 kHz, and 10 kHz with step response enter signal. Working completely well, the results of these initial tests were three control signals.

A. First Order System

First, a system with one input was tested. The chosen system can be seen below in (3). Fig. 9 demonstrates the system simulations at all frequencies chosen for the tests. Given that the results are the same and the graphs are superimposed. The same can be observed in the simulations figures 11 and 14.
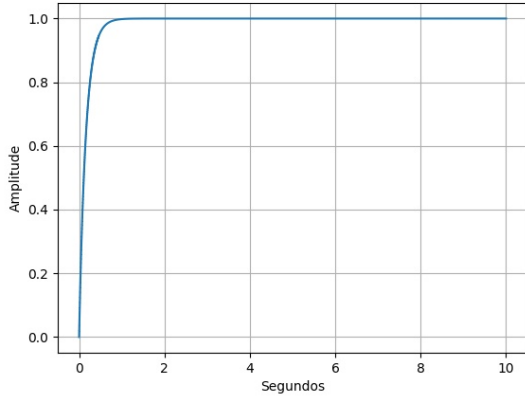
$$G_1 = \frac{1}{\frac{s}{2\pi} + 1} \tag{3}$$

Fig. 10. Experimental result first order system; $f_{sw} = 10\ kHz$; $f_{sign} = 4\ rad/s$

In addition, the variation of parameters of second-order systems may vary the type of response, compared to first-order systems in which changing parameters it only changes the response speed. The examples studied in this work present two conjugated complex poles and no zero, and the test was performed with a step input.

$$G_2 = \frac{1}{s^2 + s + 1} \tag{4}$$

The system has an input, an output, and a response of the underdamped type, where it has a pole at the origin coming from the unit step input and two complex poles, coming from the system.

Fig. 9. Second order G_3 system simulation with $f_{sw} = 100\ Hz, and\ 10\ kHz$

Fig. 11. Second order G_2 system simulation with $f_{sw} = 100\ Hz, and\ 10\ kHz$

As an initial result in Fig. 10, one can see the expected waveform, even for higher frequencies, thus proving the correct functioning of the proposed system. Also, the oscilloscope scales were different in the two signals, both in the input step and in the system response. Which justifies the different sizes of the graphs in some cases.

B. Second Order System

Next, second-order systems were tested, as they are capable of exhibiting a greater amount of possible responses when compared to first-order systems.
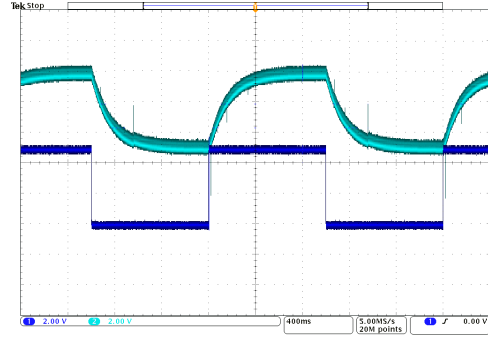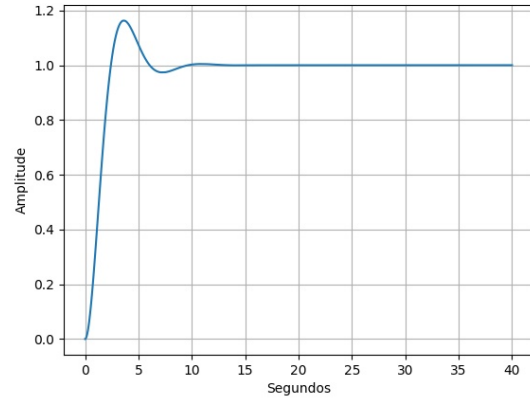
In the results, shown in Fig. 12 and 13, one can see the correct generation of the waveform of the underdamped system with constant frequency. The results were satisfactory and corresponded to the proposed system. Whereas this type of response with moving the poles along a constant radial line produces the constant damping factor. The answers are similar except for speed. The further the poles are, the faster the response.

The last system has an input, an output, and a response of the undamped type, where it has a pole at the origin coming
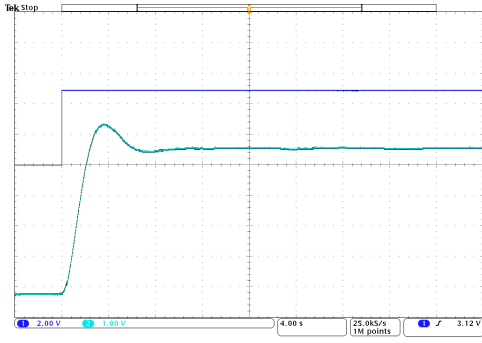
Fig. 12. Experimental result second-order system; $f_{sw} = 100\ Hz$; $f_{sign} = 40\ rad/s$

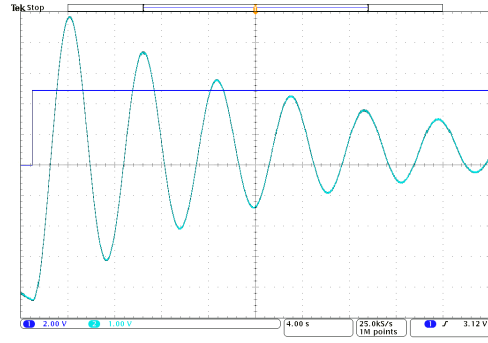Fig. 15. Experimental result second-order system; $f_{sw} = 100\ Hz$; $f_{sign} = 40\ rad/s$
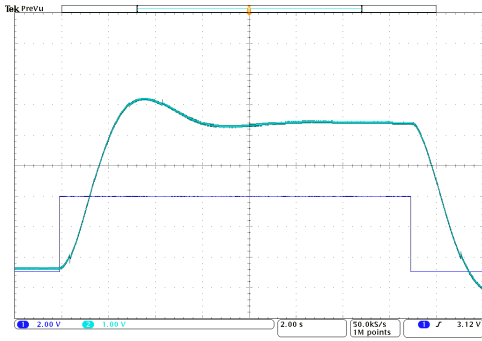

Fig. 13. Experimental result second-order system; $f_{sw} = 10\ kHz$; $f_{sign} = 40\ rad/s$
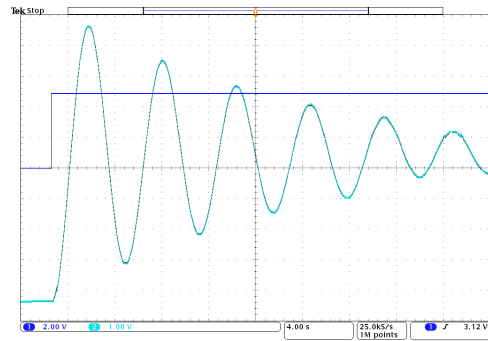

Fig. 16. Experimental result second-order system; $f_{sw} = 10\ kHz$; $f_{sign} = 40\ rad/s$

from the unit step input and two imaginary poles coming from the system.
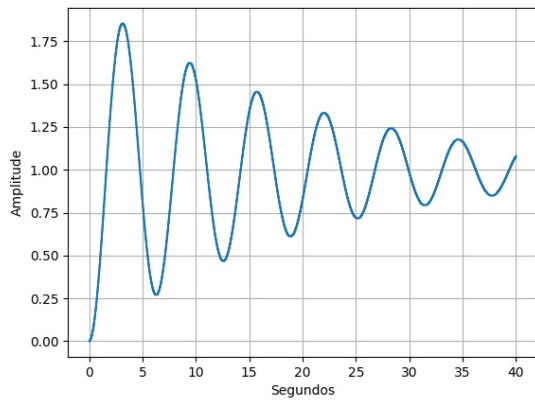
$$G_3 = \frac{1}{s^2 + 0.1s + 1} \tag{5}$$

Fig. 14. Second order G_3 system simulation with $f_{sw} = 100\ Hz, and\ 10\ kHz$

The test performed with the last system also shows good results, justified by the sinusoid generated by the imaginary poles whose frequency is equal to the location of the imaginary axis. Those results can be seen in Fig. 15 and 16.

Generally, when the sampling frequency is increased, there are greater chances of problems with numerical precision, which is the representation of poles and zeros of the system. However, this does not happen in the results presented in the work. Both the results with frequencies of 10Hz and 10kHz are the same, demonstrating that even with higher frequencies it was able to emulate the same systems.

IV. CONCLUSION

The project presented the process of design, simulation, and assembly of a low-cost didactic platform for emulation of linear systems such as the integration of electronic components, the PCB confection, and the power system from the symmetric source. As well as the implementation of the system in microcontroller with auxiliary software.

After the execution of this project, all the objectives were obtained despite the small noise caused probably by the R/2R resistive ladder network, the output signal meets the requirements and this equipment is an appropriate tool for control system laboratory classes.

This educational platform is able to be used, by means of the user's interface, both to explaining about control system theories and any other class about linear systems. Making it possible for the professor to different realistic approaches for these subjects, with low-cost equipment.

For future work, the use of floating point microcontrollers and an integrated D/A converter can be explored. Just as it

is possible to simulate systems with more than two inputs and two outputs, and the emulation of systems with order higher than two with the implementation of a new firmware development strategy.

REFERENCES

[1] J. M. Williams et al., "Versatile hardware and software tools for educating students in power electronics," in IEEE Transactions on Education, vol. 47, no. 4, pp. 436-445, Nov. 2004, doi: 10.1109/TE.2004.825552.

[2] D. A. Torrey, "A project-oriented power electronics laboratory," in IEEE Transactions on Power Electronics, vol. 9, no. 3, pp. 250-255, May 1994, doi: 10.1109/63.311256.

[3] N. R. Tavana and V. Dinavahi, "Real-Time Nonlinear Magnetic Equivalent Circuit Model of Induction Machine on FPGA for Hardware-in-the-Loop Simulation," in IEEE Transactions on Energy Conversion, vol. 31, no. 2, pp. 520-530, June 2016, doi: 10.1109/TEC.2015.2514099.

[4] A. B. C. De Farias, R. S. Rodrigues, A. Murilo, R. V. Lopes and S. Avila, "Low-Cost Hardware-in-the-Loop Platform for Embedded Control Strategies Simulation," in IEEE Access, vol. 7, pp. 111499-111512, 2019, doi: 10.1109/ACCESS.2019.2934420.

[5] L. P. D. Silva et al., "Low-cost didactic module for single-phase inverter teaching," 2017 Brazilian Power Electronics Conference (COBEP), 2017, pp. 1-6, doi: 10.1109/COBEP.2017.8257232.

[6] L. Koleff et al., "Development of a Modular Open Source Power Electronics Didactic Platform," 2019 IEEE 15th Brazilian Power Electronics Conference and 5th IEEE Southern Power Electronics Conference (COBEP/SPEC), 2019, pp. 1-6, doi: 10.1109/COBEP/SPEC44138.2019.9065327.

[7] A. V. S. Andrade and R. M. Stephan, "Didactic System for Control of Electrical Machines in Education and Research Laboratories," 2019 IEEE 15th Brazilian Power Electronics Conference and 5th IEEE Southern Power Electronics Conference (COBEP/SPEC), 2019, pp. 1-6, doi: 10.1109/COBEP/SPEC44138.2019.9065322.

[8] J. M. Jimenez-Martinez, F. Soto, Esther de Jodar, J. A. Villarejo and J. Roca-Dorda, "A new approach for teaching power electronics converter experiments," in IEEE Transactions on Education, vol. 48, no. 3, pp. 513-519, Aug. 2005, doi: 10.1109/TE.2005.852598.

[9] R. S. Balog, Z. Sorchini, J. W. Kimball, P. L. Chapman and P. T. Krein, "Modern laboratory-based education for power electronics and electric machines," in IEEE Transactions on Power Systems, vol. 20, no. 2, pp. 538-547, May 2005, doi: 10.1109/TPWRS.2005.846237.

[10] M. Bacic, "On hardware-in-the-loop simulation," Proceedings of the 44th IEEE Conference on Decision and Control, 2005, pp. 3194-3198, doi: 10.1109/CDC.2005.1582653.

[11] M. D. Omar Faruque et al., "Real-Time Simulation Technologies for Power Systems Design, Testing, and Analysis," in IEEE Power and Energy Technology Systems Journal, vol. 2, no. 2, pp. 63-73, June 2015, doi: 10.1109/JPETS.2015.2427370.