

# Virtual Laboratory in Labview for control and supervision applications with Arduino

L. C. Meneses Silva<sup>1</sup>, Walter S. Díaz<sup>2</sup>, Bryan A. Pardo<sup>3</sup>, David Herrera<sup>4</sup>

*Faculty of Mechatronics Engineering, Universidad Piloto de Colombia, 110231, Bogotá, Colombia.*

## Abstract

In the academic field, specifically in mechatronics engineering, the teaching of automation techniques requires laboratory equipment (control and automation plants), which, due to their cost, generate problems of availability when performing practices. For this reason, plant virtualization is considered, being a viable and economical alternative for the development of laboratory practices. The LabView software from National Instruments is used to develop a virtual plant laboratory, in which the mathematical model of a system is implemented through transfer functions, discretization and recurrence relation (difference equation). Each input and output of the model is associated with a physical port of an Arduino board, allowing interaction with other platforms. The plant to be simulated is a cascade combination of two tanks, fed by a pump. Two interfaces are used, one to represent the simulated plant and another the control and monitoring system. From the latter, the pump ignition signal is sent, as well as the applied voltage, these are communicated by the Arduino card so that the program with the simulated plant performs the signal processing and the response of the system is obtained in real time.

**Keywords:** Arduino, LabView, linear mathematical model, virtualization.

## INTRODUCTION

In the mechatronic, electronics and automation engineering academic environments, it is necessary to have laboratory equipment, specifically, instrumentation, control and automation plants for the use of control practices and robotics systems [1,2,3,4] which allow to develop and strengthen the disciplinary competences of engineering students [5], that is the reason why, the use of mathematical models of dynamic systems in embedded systems (hardware), a technique known as Hardware-in-the-Loop [6,7] or in computer software based on the technique Software-in-the-Loop, constitutes a viable and economical alternative for the development of practices, achieving a very close experience compared to laboratory equipment, allowing an efficiency improvement due to the simplification of the virtual environments use [8,9]. Traditionally, the costs associated with the physical use of a laboratory for practices in control and automation areas are high, just to mention a few [10,11,12]. For this reason, since the end of last century, different low cost alternatives have been proposed for the implementation of both physical and

virtual laboratories which take advantage of different technological concepts and platforms to offer the student the option of an economical and portable environment, which acquirement of different skills and competences without the need for complex assemblies, and even using long-distance learning methods [13,14,15,16]. Trends such as e-learning have impacted the way courses are taught in many fields of knowledge, and engineering have not been unaware of these changes, allowing not only to pass on theoretical knowledge from a distance, but implementing practice forms and laboratories through virtual media. It is already possible to find numerous examples of e-learning applied to the teaching of control and automation [17,18,19,20,21].

The use of dynamic models in computer systems (Hardware-in-the-Loop) is of vital importance when it comes to offering a reliable and realistic environment for a virtual laboratory practice, as it is possible to replace a physical system using the equations that govern its dynamics, without losing fidelity on the real behavior at the time of its use [22]. Tools such as Arduino and LabView have been used successfully to design and implement virtual laboratories for control and automation [23,24].

Within the framework of the project "Design and use of a system for the learning of automation through the simulation of industrial processes and interaction with industrial automation platforms supported by programmable logic controllers" carried out at the Universidad Piloto de Colombia, a tool for the interaction of mathematical models with physical systems has been developed, its methodology for use is explained in this article. The result is an application based on LabView software of National Instruments where the mathematical model of two cascaded tanks fed by a pump is used (Figure 3). Once the model is discretized and expressed in difference equations, the input and output signals of the model are associated with physical digital ports of an Arduino UNO card for its subsequent interaction with other electronic control systems [25]. Figure 1 shows the general structure of the project.



**Figure 1:** General structure of the project.

**MATERIALS AND METHODS**

The application's development for mathematical virtualization models with the capacity for interaction with physical systems was done in the following stages:

- **Stage 1.** Implementation of mathematical models in LabView.
- **Stage 2.** Physical interaction with the Arduino Platform.
- **Stage 3.** Control and monitoring system.

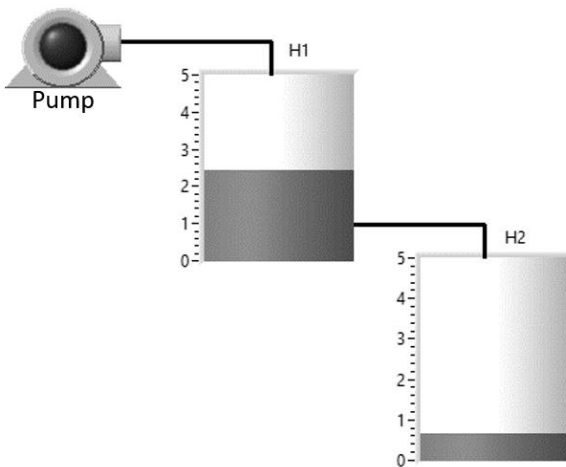
**Stage 1. Implementation of mathematical models in LabView.**

The mathematical models are formulated in differential equations, linearized using Taylor series, subsequently discretized with the Z-transform with a suitable sampling time to avoid the Aliasing phenomenon, and finally, expressed in difference equations which are implemented in the LabView software (Figure 2). The period of discretization used in the Z-transform is the reference for the execution time of the code implemented in LabView, in this way the simulation of the models will be consistent with the response of the simulated models in Matlab.



**Figure 2:** Treatment of the mathematical model.

For the validation of the interaction system, a system of interacting tanks fed by a pump is taken as a model plant (Figure 3) with assumed physical parameters. The differential equations of the model are proposed as linear first order structures invariant in time and as a relationship between the flow and the level for the cascaded tanks (Equation 1) and (Equation 2), as well as a relation of the supply voltage and the pump output flow (Equation 3) [26].



**Figure 3:** Plant Model and implemented in LabView.

$$\frac{H_x(s)}{Q_i(s)} = \frac{A}{C_x s + k_{px}}$$

**Equation 1:** Transfer function of a tank.

$$Q_{ox}(s) = k_p * H_x(s)$$

**Equation 2:** Tank output flow

$$\frac{Q_B(s)}{V(s)} = \frac{A}{\tau s + 1}$$

**Equation 3:** Transference function of a pump

Where,

- $H_x, Q_{ox}$ : Level and output flow of each tank.
- $Q_i$ : Entry flow to tank 1.
- $C_x, k_{px}$ : Capacitance and resistance in the discharge of the tank.
- $A, Q_B$ : Gains and flow of the pump.

It is relevant to highlight that the objective of the project is to use mathematical models in the platform, therefore, the coefficients of these are assumptions to obtain a rapid response allowing the validation of results (Table 1).

The method used for the discretization of the Z-transform model is the trapezoidal rule or Tustin's method [27]. Although numerical integration techniques exist for discretization of more elaborate systems [27,28,29], Tustin's approximation presents an acceptable level of approximation to the response of the discretized continuous system [30], the resulting equations are easy to understand and use on Arduino type platforms [31], facilitating its use in projects of educational nature such as the one presented here. Once the mentioned method is applied, the equations (Equation 4), (Equation 5) and (Equation 6) are obtained.

$$\frac{Q_1(s)}{V(s)} = \frac{ATz + AT}{(2\tau + T)z + (T - 2\tau)}$$

**Equation 4:** Discretized function of the pump.

$$\frac{H_1(s)}{Q_1(s)} = \frac{Tz + T}{(2C_1 + Tk'_1)z + (-2C_1 + Tk'_1)}$$

**Equation 5.** Discretized transfer function of tank No. 1

$$\frac{H_2(s)}{Q_2(s)} = \frac{Tz + T}{(2C_2 + Tk'_2)z + (-2C_2 + Tk'_2)}$$

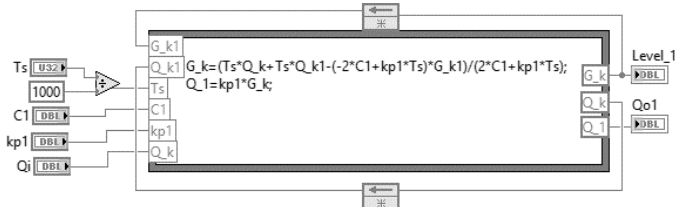
**Equation 6:** Discrete transfer function of tank No. 2

Making use of the above, difference equations are obtained, as an example, the result obtained for tank No. 1 (Equation 7) is presented.

$$H_1(kT) = \frac{T}{(2C_1 + Tk'_1)} Q_1(kT) + \frac{T}{(2C_1 + Tk'_1)} Q_1((k-1)T) - \frac{2C_1 + Tk'_1}{(2C_1 + Tk'_1)} H_1((k-1)T)$$

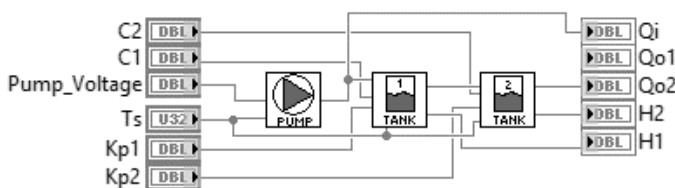
**Equation 7:** Difference equation Model for tank No. 1.

The LabView implementation for each of the difference equations is done using the Node Formula structure (Figure 4) [32], which allows the insertion of mathematical functions in the C++ language within the LabView graphic programming environment, allowing status feedback passed with the Feedback Node tool. Each difference equation is used in an extension file \*.vi designed with the respective input and output variables of the equation in addition to the parameters of the modeled component.



**Figure 4:** Implementation of the model in difference equations tank No. 1

The interaction of the different difference equations is done through the variables of input and output defined for each \*.vi file, in accordance with the modeling variables of each component. In Figure 5, we observe the use of the interaction of the different \*.vi files with the respective physical parameters of each component (gains, capacitance, discharge resistance and sampling time), in addition to the inputs and outputs of each difference equation.

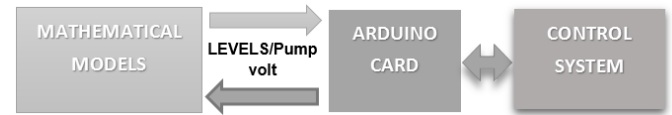


**Figure 5:** Interaction of difference equations used in LabView.

**Stage 2. Physical Interaction with the Arduino Platform**

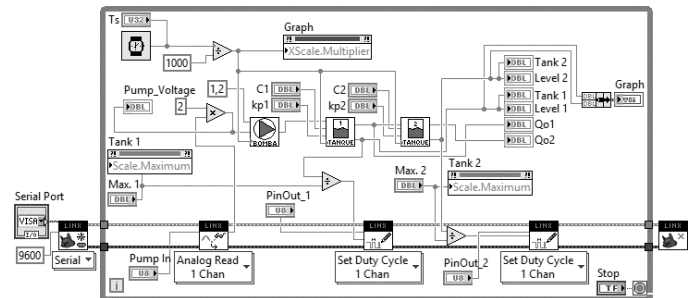
LabView allows the control of the Arduino platform, from its programming to the acquisition and sending of data through its ports, as long as it has the LINX library of MakerHub [33,34]. For this library to operate, the previous installation of National Instruments NI-VISA libraries is required, which allows the control of computer ports and the interaction with external hardware of the company.

Once the mathematical models have been implemented, the levels of the tanks (H1, H2) and the control voltage of the pump are directed to the output and input ports of the Arduino platform, which are then available to interact with the control system. (Figure 6).



**Figure 6:** Variables modeladas con interacción física.

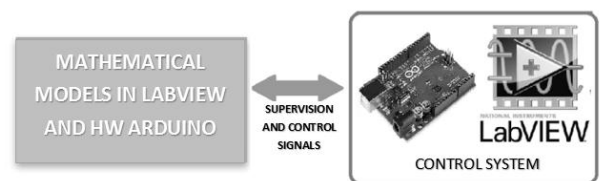
Figure 7 shows the graphic programming for the control of the Arduino card by adding the files of the Linx MakerHub library to those already used (Figure 5). In order to get the control ports to develop steadily, it is necessary that the writing and reading programming of the ports is done within a While structure and the opening and closing is outside of itself.



**Figure 7:** Programming the interaction of the mathematical model with the Linx MakerHub library.

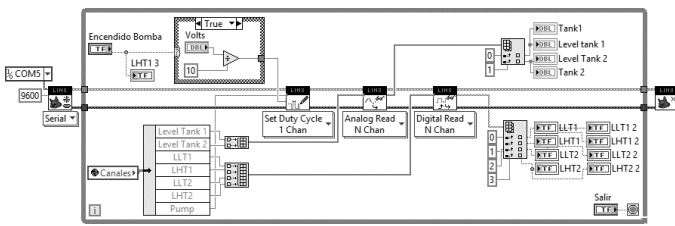
**Stage 3. Control and Monitoring System.**

The development of the project exposed in stages 1 and 2 fulfills the objective of achieving virtualization of mathematical models with the ability to interact with physical systems. To validate the process, a control and monitoring system is implemented using the same LabView-Arduino structure from which the pump control signal is sent and the status of the level signals of the tanks is monitored (Figure 8).



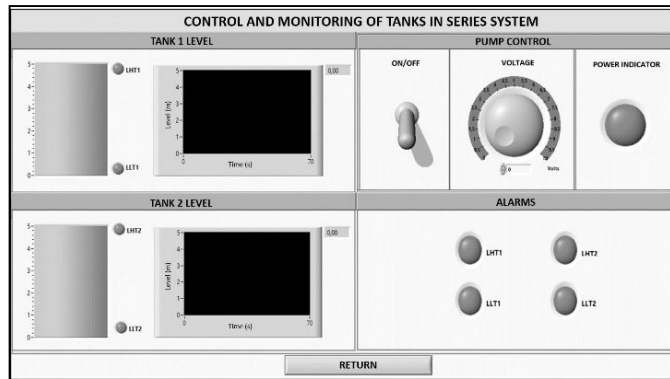
**Figure 8:** Control and Monitoring System.

The Arduino platform does not have analog output ports which allow the delivery of the signals associated with the tank levels, therefore, they are used with the PWM output ports at the standard frequency of 490hz, adding an external low-pass filter with a cutting frequency of 4.2hz, in this way a clean response is obtained and without appreciable delays. The frequency of the PWM signal can be modified with the pre-scaling of the corresponding timers. The control system program can be seen in figure 9.



**Figure 9:** Graphic programming of the Control and Monitoring System.

In the development of the visualization interface of the control and monitoring system (Figure 10), it is possible to visualize tank levels, control the pump turning it on and off and vary the supply voltage in the range of 0- 10Vdc additionally, with the measurement of the level variables, alarm levels are defined for high and low.



**Figure 10:** Control and Monitoring System Interface.

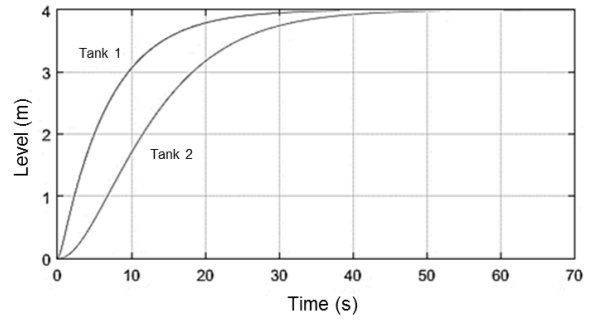
**RESULTS AND DISCUSSION**

In order to validate of the system, the constants of the model, are assumed, specifically the gains, ( $A_x$ ), capacitances ( $C_x$ ) and tank discharge resistors ( $k_{px}$ ), with the values shown in Table 1.

**Table 1:** Dynamic system simulation parameters.

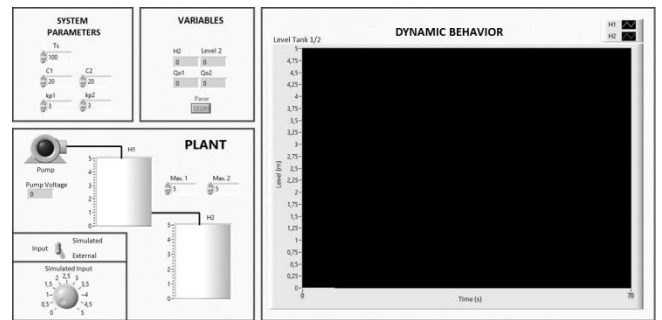
Component	$C_x$	$k_{px}$	$A_x$	$\tau$
Pump	n/a	n/a	1.2	0.4
Tank 1	20	3	1	6.67
Tank 2	20	3	1	6.67

The mathematical model is simulated in Matlab with a step of voltage input in the 10Vdc pump, obtaining the response of Figure 11



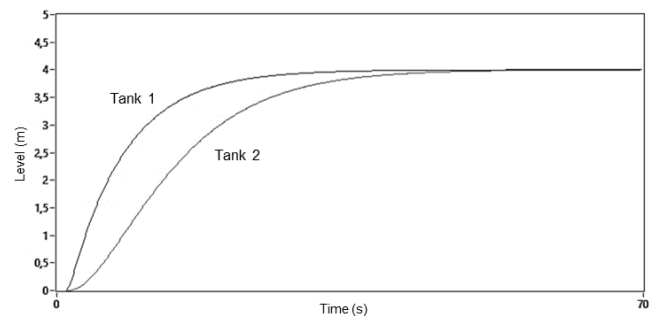
**Figure 11:** Simulation of level outputs in continuous time in the Matlab software.

The simulation constants of the model are entered in the interface of the virtual plant (Figure 12) and the program is executed while the pump and the input signal from the control and monitoring system are on. The virtual plant interface allows to vary the parameters of the system to simulate different configurations of the used system.

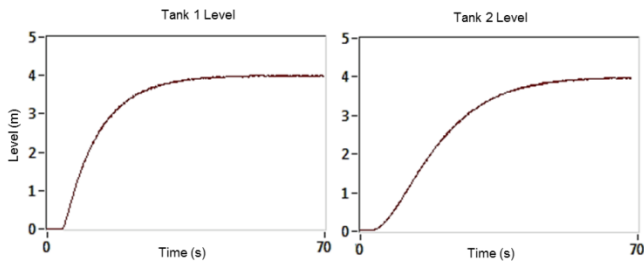


**Figure 12:** Virtual plant interface

The control and monitoring system sends the activation signal of the pump and the voltage of 10V step type. The application of the virtual plant receives the signal and generates the response shown in the virtual plant interface (Figure 13), in addition to sending the level values of the tanks through the PWM ports which are captured by the control and monitoring system (Figure 14). It can be seen that the answers are identical, in this way, validating the development used.



**Figure 13:** Response of the model in the virtual plant interface



**Figure 14:** Response obtained from the control and Monitoring System

## CONCLUSIONS

The results of the project reflect that LabView-Arduino is a viable and reliable tool for the implementation of virtual plants for laboratory practices from linear mathematical models, offering the possibility to implement a large number of virtual applications for the development of practical with external control hardware, for example, microcontrollers, programmable logic controllers (PLC's), FPGA, and even with the same Arduino platform, with the additional requirement of developing electronic hardware for signal conditioning, especially for interaction with electronic systems with different voltage and current levels to the Arduino platform.

In addition to the possibilities of implementing multiple simulated systems, portability and duplication by magnetic means, it favors independent academic work, offering an alternative to strengthen disciplinary competences from independent work, expanding the possibilities of learning by offering the possibility of counting with a "laboratory at home".

Finally, although most of the mathematical models of dynamic systems can be linearized, it is necessary to explore the way in which non-linear mathematical models can be used, which would allow the development of more complex applications, an upturn which can be carried out with the same system.

## REFERENCES

- [1] K. Astrom and A. -. Ostberg, 1986, "A teaching laboratory for process control," in *IEEE Control Systems Magazine*, vol. 6, no. 5, pp. 37-42. doi: 10.1109/MCS.1986.1105142
- [2] J. M. Andújar Márquez and T. J. Mateo Sanguino, 2010, "Diseño de Laboratorios Virtuales y/o Remotos. Un Caso Práctico," *Revista Iberoamericana de Automática e Informática Industrial*, vol. 7, no. 1, pp. 64-72.
- [3] A. S. Diwakar, S. B. Noronha and S. Agashe, 2015, "Virtual and Remote triggered Industrial Automation labs: Collaboration case study," *Proceedings of 2015 12th International Conference on Remote Engineering and Virtual Instrumentation (REV)*, Bangkok, pp. 127-130. doi: 10.1109/REV.2015.7
- [4] F. De la Cruz, M. Díaz-Granados, S. Zerpa and D. Giménez, 2010, "Laboratorio Remoto de Automatización Industrial," *Revista Iberoamericana de Automática e Informática Industrial*, vol. 7, no. 1, pp. 101-106.
- [5] G. Narayanan and A. Deshpande, 2016, "Learning Automation Made Easy through Virtual Labs," 2016 International Conference on Learning and Teaching in Computing and Engineering (LaTICE), Mumbai, pp. 60-65. doi: 10.1109/LaTICE.2016.17
- [6] W. Grega, 1999, "Hardware-in-the-loop simulation and its application in control education," in *Frontiers in Education Conference, 1999. FIE '99. 29th Annual*, San Juan, Puerto Rico.
- [7] M. Bacic, 2005, "On hardware-in-the-loop simulation," *Proceedings of the 44th IEEE Conference on Decision and Control*, Seville, Spain, pp. 3194-3198. doi: 10.1109/CDC.2005.1582653
- [8] S. Demers, P. Gopalakrishnan and L. Kant, 2007, "A Generic Solution to Software-in-the-Loop," *MILCOM 2007 - IEEE Military Communications Conference*, Orlando, FL, USA, pp. 1-6. doi: 10.1109/MILCOM.2007.4455268
- [9] A. Bayha, G. Franziska and B. Schätz, 2012, "Model-based software in-the-loop-test of autonomous systems," *Proceedings of the 2012 Symposium on Theory of Modeling and Simulation*, p. 30:1-30:6.
- [10] Manuel Macías, Israel Méndez, Ernesto Guridi, Abraham Ortiz, 2007, "TELELAB, REMOTE LABORATORY FOR AUTOMATION AND CONTROL", *IFAC Proceedings Volumes*, Volume 40, Issue 19, Pages 94-100, ISSN 1474-6670, ISBN 9783902661326, <https://doi.org/10.3182/20071002-MX-4-3906.00016>.
- [11] Chyi-Shyong Lee, Juing-Huei Su, Cheng-Chang Hsieh, Kuo-En Lin, Jia-Hao Chang, Gu-Hong Lin, 2008, "A Hands-on Laboratory for Introductory Automatic Control Courses", *IFAC Proceedings Volumes*, Volume 41, Issue 2, Pages 9737-9742, ISSN 1474-6670, ISBN 9783902661005, <https://doi.org/10.3182/20080706-5-KR-1001.01647>.
- [12] L. F. R. Riveros, A. C. Llanos, V. Hugo Bernal Tristancho and J. F. B. Navarro, 2016, "Educational module for laboratory practice in the areas of instrumentation, automation, control and industrial communications. ECCI university," 2016 *Technologies Applied to Electronics Teaching (TAEE)*, Seville, pp. 1-8. <https://doi: 10.1109/TAEE.2016.7528240>
- [13] Miguel de la Fuente Ruz, Vicente Feliu Batlle, "Low Cost Laboratory Equipment for Distance Education of Automatic Control", 1996, *IFAC Proceedings Volumes*, Volume 29, Issue 1, Pages 2913-2918, ISSN 1474-6670, [https://doi.org/10.1016/S1474-6670\(17\)58120-3](https://doi.org/10.1016/S1474-6670(17)58120-3).

- [14] Reck, R.M., 2016, Sreenivas, R.S. "Developing an Affordable and Portable Control Systems Laboratory Kit with a Raspberry Pi". *Electronics* 2016, 5, 36.
- [15] Bahadır Çatalbaş, İsmail Uyanık, 2017, "A Low-cost Laboratory Experiment Setup for Frequency Domain Analysis for a Feedback Control Systems Course", *IFAC-PapersOnLine*, Volume 50, Issue 1, Pages 15704-15709, ISSN 2405-8963, <https://doi.org/10.1016/j.ifacol.2017.08.2410>.
- [16] Uyanık I, Catalbas B., 2018, "A low- cost feedback control systems laboratory setup via Arduino–Simulink interface". *Comput Appl Eng Educ.*, 26:1–9. <https://doi.org/10.1002/cae.21917>
- [17] Košťál, Peter - Mudriková, Andrea, 2011, "Use of e-learning and virtual laboratory to automation teaching". - *Annals of Faculty Engineering Hunedoara - International Journal of Engineering*, ISSN 1584-2673, Tome IX, Fascicule 4 (extra), str. 183-186. In: *Acta Technica Corviniensis - Bulletin of Engineering*. - ISSN 2067-3809. - Tom IV (2011), s. 117-120
- [18] M. Tariq Banday, Musavir Ahmed, Tariq R. Jan, 2014, "Applications of e-Learning in Engineering Education: A Case Study", *Procedia - Social and Behavioral Sciences*, Volume 123, Pages 406-413, ISSN 1877-0428, <https://doi.org/10.1016/j.sbspro.2014.01.1439>.
- [19] A. S. Diwakar, S. B. Noronha and S. Agashe, 2015, "Virtual and Remote triggered Industrial Automation labs: Collaboration case study," *Proceedings of 2015 12th International Conference on Remote Engineering and Virtual Instrumentation (REV)*, Bangkok, pp. 127-130. doi: 10.1109/REV.2015.7087276
- [20] J.R. Caldas Pinto, J.M.G. Sã da Costa, 2013, "Virtual and Remote Laboratories for Industrial Automation E-learning", *IFAC Proceedings Volumes*, Volume 46, Issue 17, Pages 286-290, ISSN 1474-6670, ISBN 9783902823434, <https://doi.org/10.3182/20130828-3-UK-2039.00063>.
- [21] G. Narayanan and A. Deshpande, 2016, "Learning Automation Made Easy through Virtual Labs," 2016 International Conference on Learning and Teaching in Computing and Engineering (LaTICE), Mumbai, pp. 60-65. doi: 10.1109/LaTiCE.2016.17
- [22] R. Dormido et al., 2008, "Development of a Web-Based Control Laboratory for Automation Technicians: The Three-Tank System," in *IEEE Transactions on Education*, vol. 51, no. 1, pp. 35-44, doi: 10.1109/TE.2007.893356
- [23] H. Muhammad Asraf, K.A. Nur Dalila, A.W. Muhammad Hakim, R.H. Muhammad Faizzuan Hon, "Development of Experimental Simulator via Arduino-based PID Temperature Control System using LabVIEW", *Journal of Telecommunication, Electronic and Computer Engineering*, Vol. 9 No. 1-5, ISSN: 2180-1843
- [24] Bai, Jianghua & La Rosa, Andres.,2017, *Essentials of Building Virtual Instruments with LabVIEW and Arduino for Lab Automation Applications*. International Journal of Science and Research (IJSR). 604. 10.21275/ART20173325.
- [25] R. W. Fransiska, E. M. Septia, W. K. Vessabhu, W. Frans and W. Abednego and Hendro, 2013, "Electrical power measurement using Arduino Uno microcontroller and LabVIEW," in 3rd International Conference on Instrumentation, Communications, Information Technology and Biomedical Engineering (ICICI-BME), Bandung-
- [26] W. J. Palm III, 2012, *System Dynamics*, 2 ed., McGraw Hill.
- [27] Endre suli, 2014, "Numerical Solution of Ordinary Differential Equations", *Mathematical Institute, University of Oxford*.
- [28] Griffiths, David F. and Higham, Desmond, 2010, "Numerical Methods for Ordinary Differential Equations : Initial Value Problems". Springer, Undergraduate Mathematics Series . Springer, ISBN 978-0-85729-147-9.
- [29] Elmqvist, Hilding; Åström, Karl Johan; Mattsson, Sven Erik, 1998, "Evolution of Continuous-Time Modeling and Simulation". Paper presented at 12th European Simulation Multiconference, ESM'98, Manchester, United Kingdom.
- [30] L. B. Song and L. Xu and X, 2014, "A comparative study on Tustin rule-based discretization methods for fractional order differentiator," 2014 4th IEEE International Conference on Information Science and Technology, pp. 515-518.
- [31] Vikrant Chaudhary, 2014, "Active Vibration Control Using Modal Control and Experimental Implementation on Arduino Microcontroller", MS Thesis, University of Cincinnati.
- [32] National Instruments, 2017, "Tutorial: MathScript and Formula Nodes". [Online]. Available: <http://www.ni.com/white-paper/7572/en/>.
- [33] Schwartz, Marco & Manickum, Oliver, 2015, "Programming Arduino with LabVIEW" Birmingham-Mumbai: PACKT Publishing.
- [34] L. Solaque Guzmán, C. A. Cristancho and C. A. Gil, 2014, "Diseño e implementación de una plataforma experimental de dos grados de libertad controlada por dos técnicas: pid y lógica difusa," *Ciencia e Ingeniería Neogranadina*, vol. 24, no. 1, pp. 99 - 115.



## **BIOGRAPHICAL SKETCH**

### **M. Sc. Luis Carlos Meneses Silva**

Popayán, Colombia. Industrial Automation Engineer from the Universidad del Cauca, Popayán, Colombia. He obtained his Master's degree in Maintenance Engineering in 2010, at the Polytechnic University of Valencia in the city of Valencia, Spain.

He worked as a professor at the University of San Buenaventura for 3 years. Later, he joined the Universidad Piloto de Colombia as a professor in which he has been for the past 6 years

He is currently a professor in the area of automation and industrial maintenance at the same university, and a researcher to the InnovaTIC group where he conducts research on virtual learning systems for automation.

### **Walter Stick Díaz Arias**

Mechatronics engineering student at the Universidad Piloto de Colombia, Bogotá, Colombia.

Currently working as an undergraduate researcher at the Automation and Domotics workshop of the Universidad Piloto de Colombia where he studies virtualization of mathematical models.

### **Bryan Alexis Pardo Fajardo**

Soacha, Colombia. Mechatronics Engineering student at the Universidad Piloto de Colombia, de Bogotá, Colombia.

Currently working as an undergraduate researcher at the Automation and Domotics workshop of the Universidad Piloto de Colombia where he studies virtualization of mathematical models.

### **M.Sc. David Herrera Alfonso**

Mechatronics Engineer from the National University of Colombia, Bogotá, Colombia. He obtained his Master's degree in Automation in 2011 at the same university, where he is also a candidate for a PhD in Engineering in the specialty of Mechanical Engineering and Mechatronics.

He joined the Pilot University of Colombia as a teacher in which he currently works for the last 3 years.

He is a current professor in the area of control in the same university and a researcher to the InnovaTIC group where he conducts research on mobile robotics, control and automation.