HARDWARE IN THE LOOP SIMULATION FOR INDUSTRIAL PROCESS CONTROL

FAKIR, FELIPE [1]
E-MAIL:FELIPE.FAKIR@EMERSON.COM

BRUNINI, DANILO [1]
E-MAIL:DANILO.BRUNINI@EMERSON.COM

GODOY, EDUARDO P.[2]
E-MAIL:EPGODOY@SOROCABA.UNESP.COM

[1] EMERSON PROCESS MANAGEMENT – SOROCABA

[2] GROUP OF AUTOMATION AND INTEGRATION SYSTEMS (GASI), UNESP – SÃO PAULO STATE UNIVERSITY - SOROCABA

Abstract— Real-time simulation for control loop design and operators training systems (OTS) are increasingly being used in the process industry due to some advantages as safety, repeatability, comprehensiveness, and reduced cost and time of development. One technique to perform real-time simulations is HIL – Hardware-In-the-Loop, which consists in the simulation of an industrial process part operating in real-time with real equipment belonging to the control system. HIL addresses the use of a dedicated hardware for real-time model simulation and communication interface with control systems. This whitepaper highlights the use and discusses the benefits of the HIL technique for industrial process control. The process dynamic model, a coupled tank industrial process, was designed in the Mimic simulation software and downloaded in to a real-time hardware, VIM, for process simulation purpose. This process model is integrated to the DeltaV distributed control system (DCS), designed to perform process control. By using the HIL solution, it is possible to enable OTS of industrial control systems, besides simulating the behavior of industrial processes with different controller tuning, reducing risks, tuning and startup times of automation systems.

Keywords— DeltaV, Mimic, hardware in the loop, process control, simulation and modeling.

1 Introduction

Studying characteristics of systems through simulation has gained importance over the years, and this practice ensures significant benefits to automation and control companies, such as reducing costs with prototyping, possibility to testing systems under different conditions with high repeatability; and as a result optimizing processes and product development (DEMERS et al., 2007). Simulation for control loop design and operators training systems in the process industry usually demands real-time responses to provide better reliability (SAHIN et al. 2012; RATH, 2013). In real-time simulation, the temporal behavior of the simulation must be as similar as possible to the controlled process. According to Kopetz (1997), the definition of real-time requires that all computational efforts of the calculations that are required to be performed in a restricted time interval, must be obligatorily performed in this time interval. Kweon et al. (1999) complements that not only the calculations of the models as well as the exchange of information between the systems involved, i.e., the communication between them should be included in the same time interval.

There are different techniques and purposes for conducting process control simulations. The three most known simulation techniques are the Rapid Control Prototyping (RCP), the Hardware-In-the-Loop (HIL) and the Software-In-the-Loop (SIL) (ISERMANN, 2008). RCP and HIL are considered real-time simulations as there is a requisite for a real-time operational system and dedicate hardware for models to provide deterministic temporal behavior. According to Bélanger & Dufour (2012), in RCP applications, a controller is implemented using a real-time simulator and is connected to a real physical plant. The advantages of RCP over a controller prototype are that developing the controller in a real-time simulator is faster and more flexible. Considering the HIL technique, usually a physical controller is connected to a virtual plant executed on a real-time simulator. The main advantages of HIL is allowing test of controllers in conditions unavailable on real plants. SIL usually does not require real-time and deterministic responses as both controller and plant can be simulated on the same simulator. The main advantage of SIL over RCP and HIL is that no I/O or hardware needs to be used.

Figure 1 shows a real-time simulation structure, WinMod (2013). In details a system for real-time process simulations and operator training environment is presented. The operators interface is accomplished through the HMI (Human Machine Interface). The simulated system can be operated also new control loops strategy can be designed through this equipment. In order to reach the same or similar actual real process performances, the control system (PLC or DCS) must be the same control system installed in the real process. It's also necessary a hardware dedicated to operate in real-time devoted to mathematical models simulation. As a result, it performs the hardware in the loop function executing simulations of process dynamics, actuators, sensors, and can also simulate data and information exchange with control systems as discussed in Popovic &

Mosterman (2012). The simulated models developed in the computing environment are running on dedicated hardware and work in real time and in synchronism with the actual equipment of the system under development, making it possible to perform tests in different scenarios and under conditions close to real.

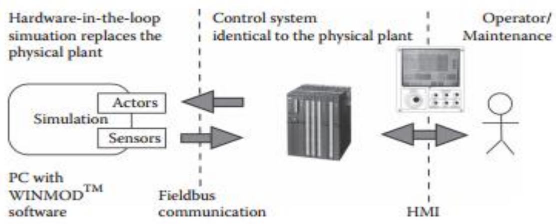

Figure 1: Real-time simulation system: Hardware-In-the-Loop configuration

According to Faty et al. (2006) studies, HIL simulation has become indispensable for the aerospace, automotive, marine and defense industries and is justified by its many advantages, including cost effectiveness, rapid prototyping, repeatability, safety, comprehensiveness, automated testing, etc.

• Cost effectiveness: A HIL simulation often requires less hardware than physical prototypes, thereby costing less.

• Rapid prototyping: Often the HIL requires less hardware than fully physical prototypes. HIL simulators can also be considerably faster to build.

• Repeatability: Systems that operate in variable environments can often be tested in controlled lab, which may significantly increase repeatability of tests and simulations.

• Safety: HIL simulation can provide destructive tests events, for example, vehicle accidents and trapping missile without incurring high cost and possible destruction of the test platform. It can also be used to train human operators in critical processes where human error can lead to disaster, for example, flight simulators for airplane pilots and operation of safety critical systems such as a nuclear power plant.

• Comprehensiveness: The HIL simulation promotes the possibility of control the entire around environment and operating conditions of the process.

• Automated testing: Tests can be performing in automated environments without human intervention.

The context that fits HIL simulation shows advantages of using real-time models emulation in a highly controlled environment with low cost implementation. The main objective of this work is to show that not only automotive, aerospace industries; etc can use hardware in the loop simulations for a system validation, design and training.

Next sections provide valuable information of how this simulation tecnnique can be applied for process industries, as chemical, petrochemical, mining, etc. This work presents how to create a process model for a simple process unit, and simulate it in a real-time hardware integrated with a control system and an IHM for operator's interface. The relevancy of this work is also to present a industrial HIL simulation architecture and provide initial information for future work in HIL simulation applied for process industries.

Section 1 shows a briefly review of state of art in HIL simulation, section 2 presents a system architecture for a HIL simulation system for process industries, in the section 3 a coupled tanks is modeled, section 4 shows the design of the process dynamics and control strategy implementation, section 5 presents control testing performances and 6 conclusions.

2 Proposed HIL Architecture

Figure 2 presents a HIL system designed to simulate industrial process dynamics. In this proposed architecture, each system component has a specific role described next (Mimic, 2013; DeltaV, 2013).

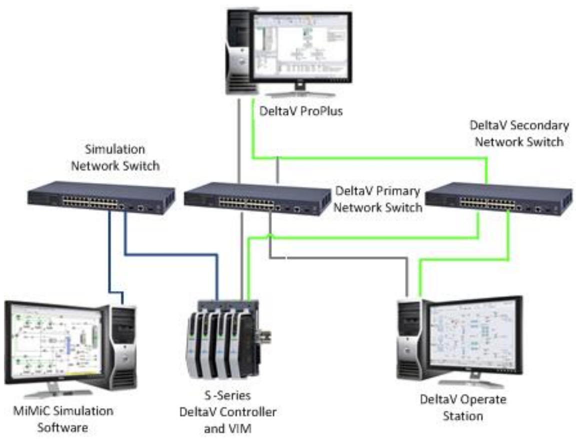

Figure 2: HIL for process simulation and operators training system (OTS)

• Modeling station: This station contains the Mimic simulation software used for industrial processes modeling. With this software is possible through functional blocks simulate many processes dynamics.

• Real-time simulation hardware: The process models contained in the database are downloaded in a real-time hardware called VIM. VIM plays the role of the process to be controlled.

• Control strategy: The DeltaV Proplus station is used for the design of control strategies using DeltaV software.

• Industrial controller: The DeltaV controller performs the role of supervision and control of the industrial process; it uses the data contained in the Proplus station.

• Supervisory station: It's the operator station, where testing and operation of the simulated process are performed. This has an interface with historian, operation graphics, process details and diagnostics.

3 Process Modeling

Process modeling is a very important step in control strategies design and control loop tuning. The process is modeled as technical specification and can be used for familiarization and training of operators who will operate the industrial process. An important step in process modeling is to survey the characteristics of the process being controlled. It is extremely important that the generated model is faithfully, but sometimes it’s acceptable to have a few differences between simulation and actual operating conditions of the process unit.

The proposed process for modeling, characterization and control is presented in Figure 3. The problem exposed was to develop a PID controller to control the level h_2 of the tank $_2$ by manipulating the inlet valve of tank $_1$ through its flow Q_E.

In order to reach this, it was considered a level transmitter LT, 4-20 mA with a calibrated range of 0-2 m. The control valve is actuated via an analog output 4-20 mA and is transformed through an electro positioner I / P that converts 4-20 mA to 3-15 psi, positioning the control valve in the desired set point. The control valve flow characteristics are from 0 to 0.002 m3/s and the process time constant is 10.0 s.

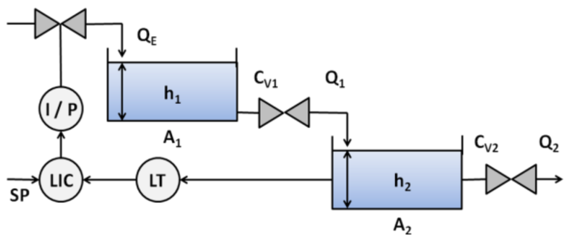

Figure 3: Coupled tanks

The variables and values of the concerning systems are presented below:

- A_1 / A_2 – Tanks base area, $A_1 = A_2 = 1$ m^2.
- K_1 / K_2 – Output valves constants characteristics considering turbulent regime, $K_1 = 0.025$ and $K_2 = 0.018$.
- h_1 / h_2 – Water level in the tanks.
- Q_E / Q_1 / Q_2 – Water flow in the tanks.
- LT – Level Transmitter.
- LIC – Level Controller.
- I / P converter – Current to pressure converter.

The differential equations relating the process flows were generated. The relationship between the inlet flow Q_E and outlet flow of the tank $_1$ Q_1 can be expressed by equation (1).

$$A_1 \times \frac{\partial h_1}{\partial t} = Q_E - Q_1 \quad (1)$$

Where Q_1 is the reason h_1 by r_1, expressed in equation (2).

$$Q_1 = \frac{h_1}{r_1} \quad (2)$$

The results of the substitution of the equation (2) in (1) results in the differential equation (3) which governs the behavior of the level h_1 in time function.

$$A_1 \times \frac{\partial h_1}{\partial t} = Q_E - \frac{h_1}{r_1} \quad (3)$$

Analogously it is possible to write the relation between the inlet flow Q_1 and the outlet flow Q_2 of the tank $_2$ as shown in equation (4).

$$A_2 \times \frac{\partial h_2}{\partial t} = \frac{h_1}{r_1} - \frac{h_2}{r_2} \quad (4)$$

For the differential’s equations resolution, Laplace transform technique was used. Applying the Laplace transform on equations (3) and (4) the equations (5) and (6) are found and are respectively displayed in the frequency domain.

$$\frac{H_1}{Q_E}(s) = \frac{r_1}{(A_1 \times r_1 \times s) + 1} \quad (5)$$

$$\frac{H_2}{H_1}(s) = \frac{r_2}{(A_2 \times r_1 \times r_2 \times s) + r_1} \quad (6)$$

In order to find the relationship between H_2 and Q_E equation (5) and (6) are multiplied resulting in the transfer function of the plant to be controlled.

The expression (7) and (8) show the transfer function of the process.

$$G_p(s) = \frac{H_1}{Q_e}(s) \times \frac{H_2}{H_1}(s) \quad (7)$$

$$G_p(s) = \frac{r_1 \times r_2}{((A_1 \times r_1 \times s) + 1) \times ((A_2 \times r_1 \times r_2 \times s) + r_1)} \quad (8)$$

The r_x calculation is presented in Figure (9), it’s a nonlinear equation. Assuming 1 m the desired control operational value, it’s possible to simplify the calculations doing the linearization at this same value in equation (9). Substituting all process parameter values in equation (8) the transfer function G_p are found and presented in (10).

$$r_x = 2 \times \sqrt{\frac{h_x}{K_x}} \quad (9)$$

$$G_p(s) = \frac{8889}{(s+0,01253)\times(s+0,01795)} \tag{10}$$

The proposed model for the control valve can be expressed by equation (11). As defined at the beginning of this section a first order model with the proposed characteristics represents de process dynamics of this actuator.

$$G_v(s) = \frac{0.000167}{(10\times s)+1} \tag{11}$$

The G_{ip}, I / P converter, was modeled as a proportional gain equal to 0.75 which is the conversion rate of the electrical signal 4 - 20 mA to a pneumatic signal 3-15 psi connected to the actuator.

For the sensors dynamic it's also considered a proportional gain, which the conversion rate of 0-2 m to 4-20 mA, this signal goes to the DCS analog input, this signal needs to be convert inside the DCS by 4-20 mA to 0-2 m, so the treatment of the sensor dynamic was taken out of the simulation. If there is a time constant assigned to the sensors dynamics, this should be considered in the process model.

4 Modeling and Controller Implementation

With the process models exposed in the frequency domain, the design strategy presented by Figure 4 was adopted.

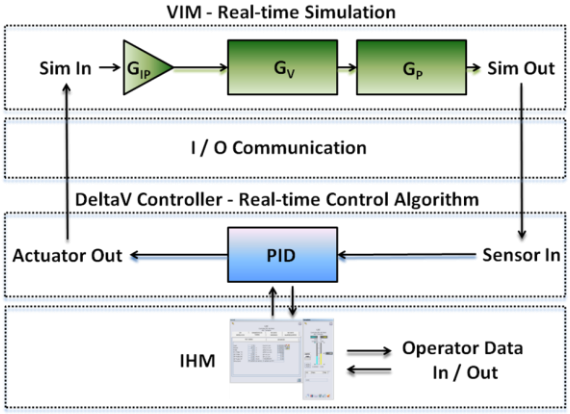

Figure 4: Hardware in the loop design strategy

In the Proplus station database, a PID control module was designed with LIC tag and the control loop created with this module was downloaded to the DeltaV controller.

The objective of the control module is to control the process modeled in Mimic through a PID control strategy. The control module is presented in Figure 5, the PID block has internal tuning parameters that can be modified to perform the control loop tuning, also provides set point parameter for automatic mode operation and the parameter OUT for manual mode operation, in this mode, the PID control algorithm is not considered so it's possible to act directly in the valve actuator.

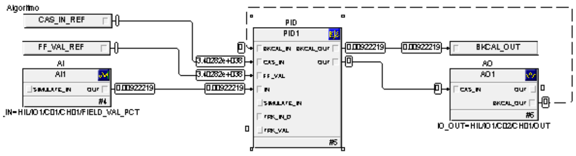

Figure 5: DeltaV – PID control module

In order to implement the equation (10) the process dynamics was uncoupled to make the implementation easier. The calculation method of partial fraction expansion was applied as the identity shown in equation (12).

$$G_p(s) \cong \frac{A}{(s+0,01253)} + \frac{B}{(s+0,01795)} \tag{12}$$

The results were A = 1640036,9 e B = -1640037,9. Substituting A and B in the equation (12) we found the uncoupled model (13).

$$G_p(s) = \frac{1640036.9}{(s+0,01253)} + \frac{-1640036.9}{(s+0,01795)} \tag{13}$$

To implement the model G_p two first-order filter function blocks were used, it was necessary to manipulate the equation (13) for direct parameter input in the modeling software.

After this operation equation (14) was found, and was implemented directly in a control module in Mimic simulation software.

$$G_p(s) = \frac{130888818,8}{(79,8\times s+1)} + \frac{-9166958,22}{(55,71\times s+1)} \tag{14}$$

The valve model has also been implemented using a first order filter, according to equation (11) it was also inserted in the dynamic model the I / P converter. The results of these implementations are presented in Figure 6.

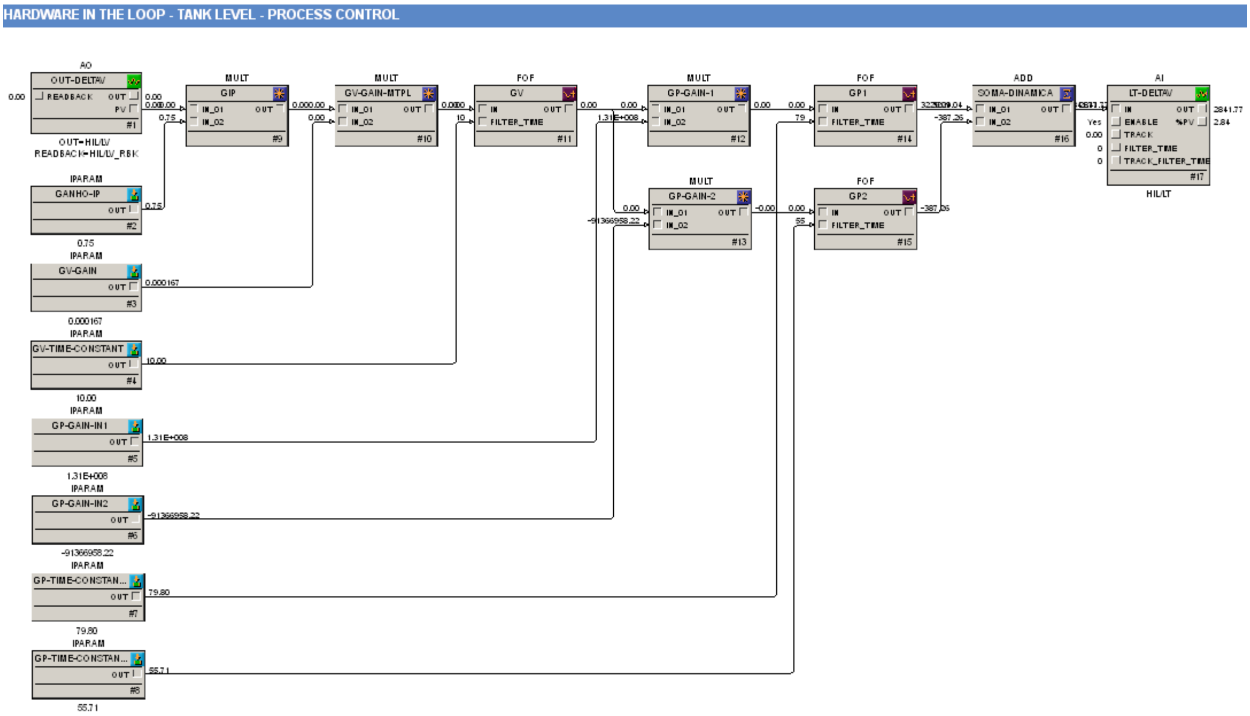

Figure 6: Simulated process in Mimic software

5 Control loop testing performance

With the process model running in real time on the VIM hardware and the PID control module running in the DeltaV controller, the PID block was set to Manual mode and a step change was applied in the manipulated variable, the valve was manually opened and closed to check the model behavior. Figure 7 presents the data collected from the process model in the DeltaV historian.

For the PID controller tuning tests, DeltaV auto tuning tool were used. Using this tool, it was possible to find the initial values of the PID tuning parameters. Subsequently a fine tune of the control loop was adjusted manually.

Figure 8 shows the disturbance made by the automatic control loop tuning using the Lambda PI rule (Lambda PI, 2014) cycle methodology in the auto tuning tool.

The methodology used for control loop fine tuning was divided into four stages called tuning tests.

Table 1 shows the values of PID tuning parameters set for each step of tuning test. First, the values found by DeltaV auto tuning were used; this experiment was called *test 1* then the others tests performed were to manually adjust the control loop tuning parameters for study and analysis of the controlled system behavior.

Table 1: PID tuning parameters

PID tuning parameters	Test 1	Test 2	Test 3	Test 4
Gain	3.89	1.75	1.75	2.2
Reset	149.5 s	127 s	87 s	87 s
Rate	0 s	0 s	0 s	0 s

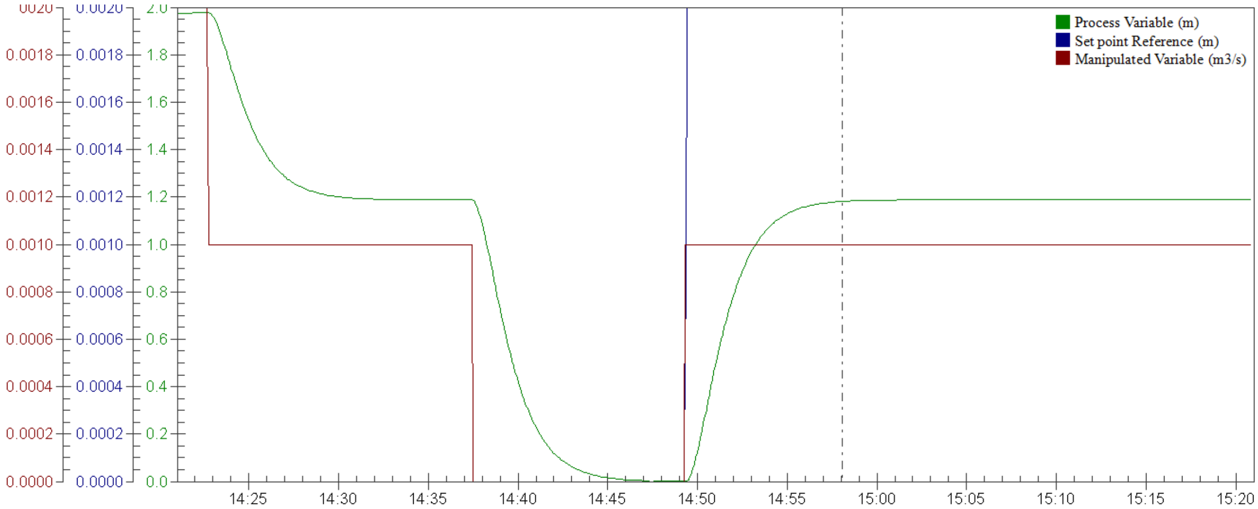

Figure 7: Open loop process response

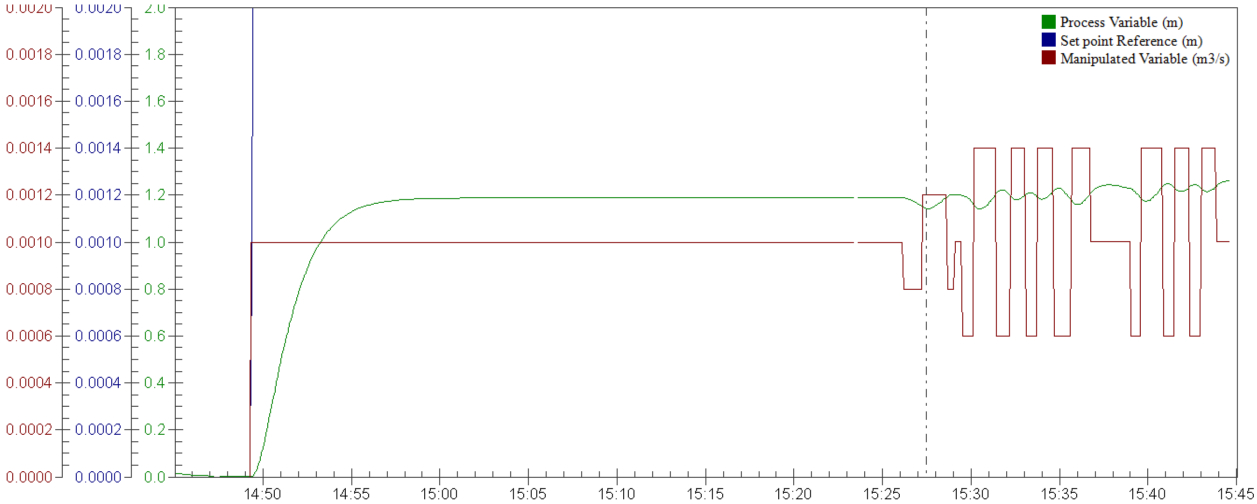

Figure 8: Process disturbances for tuning

Figure 9 shows the graphical interface built for data input into the process, two control screens were created. A change in the set point value when the PID is in auto mode or a change in the OUT parameter in manual mode, can act on the process directly, positioning the control valve at a specific flow. The second window is called detail page where you can perform tuning parameters change of the PID controller.

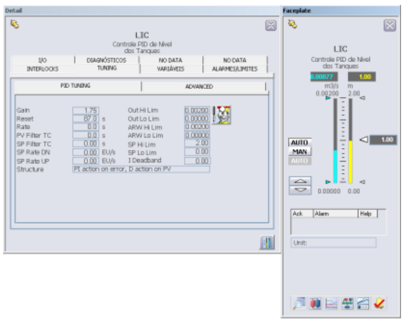

Figure 9: IHM - Process operator interface for the PID Loop

The process response curves for each test are presented in Figure 10. It presents four disturbances performed by set point changes, each one represents a specific test, the process responds for the disturbances is in the historical chart respectively.

Using graphical analysis, Table 2 was created and presents the results of some control parameters of interest. These results were obtained through the response process by the disturbances set point.

Table 2: Process response due to set point changes

Project parameters	Test 1	Test 2	Test 3	Test 4
Overshoot	1 %	0 %	1 %	0.5 %
Settlement Time	195 s	180 s	320 s	150 s
Rise time	57 s	97 s	101 s	85 s

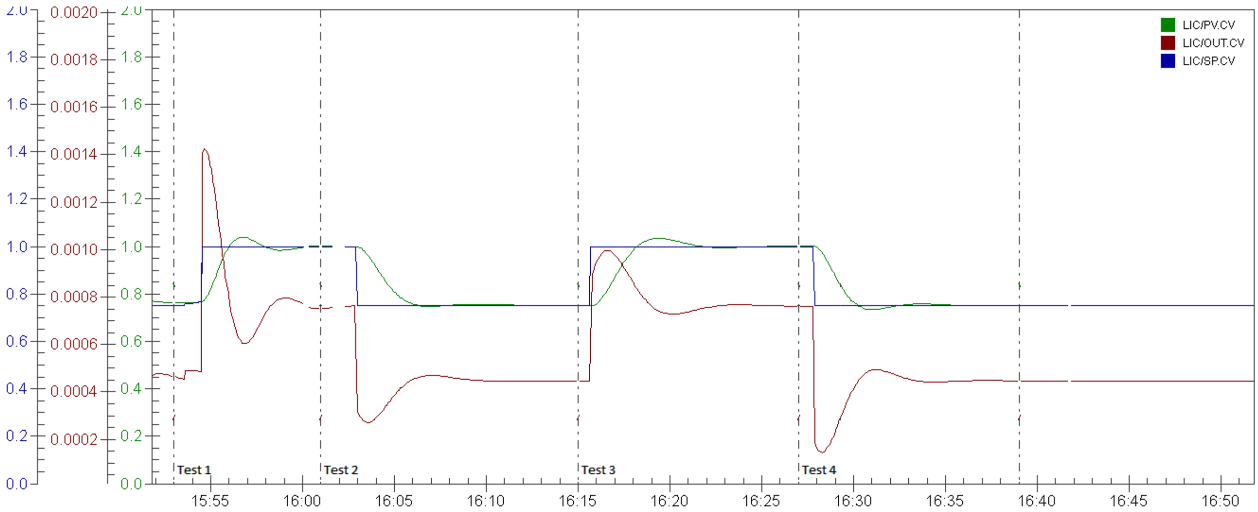

Figure 10: Control loop testing with different PID tuning parameters

The best response of a controlled process cannot always be assessed as being the lowest rise time, or with the smallest overshoot and, finally, the answer to the shorter time settlement.

It should take into account the requirements of the process, there are processes which does not require rapid responses but stabilization, or processes that cannot be oscillatory, there should be no overshoot, and thus can be used the tuning parameters found in test 2.

Other processes require quick responses no matter with settling time and oscillations, and may use the parameters tuned by test 1.

Sometimes the response of the system requires a tradeoff between two factors should have a relatively fast response accepting a minimum of overshoot can be used to tune the parameters found by the test 4.

6 Conclusion

The use of HIL simulations in design of control systems and operator training environment is effective since the designer knows well the behavior or the dynamics of the process to be controlled. The HIL architecture, as the one proposed in this paper, allows several studies to be conducted, for example the design and implementation of new control strategies. In addition, and maybe more important, is the possibility of building operator training centers fast and safely, reducing training costs and time to operate an industrial plant. The advantage of using a real-time simulator for OTS is that the user can acquire a feeling for the controller and plant that correctly and reliably represents the real system, without the delays and limitations commonly found in training environments based on recorded scenarios.

The concepts and results of this paper can be expanded to other types of processes models, predicting its real behavior for different types of industrial controller tuning, reducing risks, cost and time with tuning and startup of automation projects.

References

Kopetz, H *Real-Time Systems: Design Principles for Distributed Embedded Applications*. Netherlands: Kluwer Academic Publishers, 1997.

Kweon, S. K.; K. G. Shin; Q. Zheng *Statistical Real-Time Communication over Ethernet for Manufacturing Automation Systems*. In Proceedings of the 5th IEEE Real-Time Technology and Applications Symposium, pp. 192–202. Vancouver, Canada, 1999.

Winmod *Real Time Simulation Center for Automation*. Accessed November 27th 2013. http://www.winmod.de.

Popovic, J, Mosterman *Simulation Technologies: Principles, Methodologies and Applications*. Taylor & Francis Group, LLC, USA, 2013.

Mimic *Mimic Simulation Software*. Accessed November 27th 2013. http://www.mynah.com/products/mimic.

DeltaV *The DeltaV Digital Automation System* Accessed November 27th 2013. http://www2.emersonprocess.com/en-US/brands/DeltaV/Pages/index.aspx.

Lambda PI *Lambda Tuning-the Universal Method for PID Controllers in Process Control* Accessed July 27th 2014. http://www.controlglobal.com/assets/13WPpdf/131022-coughran-controllers.pdf

Rath, G. Simulation for Operator Training in Production Machinery. In Real-Time Simulation

Technologies: Principles, Methodologies, and Applications, Chapter 18, ed. Katalin Popovici, Pieter J. Mosterman. pp. 439–460, CRC Press, 2012.

Şahin, S., Y. İşler, and C. Guzelis. A Real Time Simulation Platform for Controller Design, Test and Redesign. In: Real-Time Simulation Technologies: Principles, Methodologies, and Applications, Chapter 19, ed. Katalin Popovici, Pieter J. Mosterman. pp. 461–500, CRC Press, 2012.

Bélanger, J. and Dufour, C. Modern Methodology of Electric System Design Using Rapid-Control Prototyping and Hardware-in-the-Loop. In: Real-Time Simulation Technologies: Principles, Methodologies, and Applications, Chapter 9, ed. Katalin Popovici and Pieter J. Mosterman, pp. 219–242, CRC Press, 2012.

Bonivento, C.; Cacciari, M.; Paoli, A. and Sartini, M. (2011). Rapid prototyping of automated manufacturing systems by software-in-the-loop simulation. Chinese Control and Decision Conference (CCDC), pp. 1-6.

Resmerita, S.; Derler, P.; Pree, W. and Butts, K. (2013). Validator Tool Suite – Filling the Gap between Conventional Software-in-the-Loop and Hardware-in-the-Loop Simulation Environments. In: Real-Time Simulation Technologies: Principles, Methodologies, and Applications, Chapter 8, ed. Katalin Popovici and Pieter J. Mosterman, pp. 199–218, CRC Press, 2012.

Chioran, D. and Machado, J. (2011). Design of a Mechatronic System for Application of Hardware-in-the-loop Simulation Technique. Proceedings of International Conference On Innovations, Recent Trends And Challenges In Mechatronics, Mechanical Engineering And New High-Tech Products Development – MECAHITECH, vol. 3, pp. 1-9.

Demers, S.; Gopalakrishnan, P. and Kant, L. (2007). A Generic Solution to Software-in-the-Loop. IEEE Military Communications Conference - MILCOM,pp. 1-6.

Isermann, R. Mechatronic systems—Innovative products with embedded control, Control Engineering Practice, Vol. 16, No. 1, pp. 14-29, 2008.

Hosam K. Faty, Zoran S. Filipi, Jonathan Hagena and Jeffrey L. Stain. Review of Hardware-in the-Loop Simulation and Its Prospects in the Automotive Area, Modeling and Simulation for Military applications, Vol. 6228 6280E-1, 2006.