Comparison of MATLAB Simulink application with PLC application of real-time classical PID controllers in laboratory

Bassam Mohsin Atiyah[1], Saber Hameed Yadgar[2], Murad Ghazy Khalaf Alabdullah[3]

[1] Department of Electrical Techniques, Kirkuk Technical Institute, Northern Technical University, Iraq
[2,3] Department of Electronic Techniques, Kirkuk Technical Institute, Northern Technical University, Iraq

ABSTRACT

Process control is a common area of study for different engineering disciplines. Controllers working with the proportional–integral–derivative (PID) method are generally used in process control. Parameters like level, flow, position, pressure, and temperature are controlled in-process control. PID algorithm can be formed with different microprocessor-based devices. Process control is implemented by using analog input and outputs terminals of PLCs (Programmable logic controller) which is recently the most commonly used in industries. Also, a real-time PID method can be implemented by using a computer-based MATLAB program in the laboratory.

The liquid level and flow control in the laboratory has been done by using PID algorithms and PLC device and MATLAB real-time interface by using an experimental set. Necessary connections and configurations have been prepared to perform process control. (SIMATIC Manager Step7) is used for programming in PLC. Besides, the MATLAB simulation program is configured to implement real-time control to be compatible with the MF624 data acquisition card. Liquid level and fluid flow control experiments were performed separately for the same PID algorithms formed on two different devices, then liquid level detection and liquid flow control results have been collected and investigated. Based on these results, a total of four experiments were compared with both devices considering the ease and difficulties of using PLC and MATLAB.

Keywords:	PID, PLC, Level control, Flow control

Corresponding Author:

Murad Ghazy Khalaf Alabdullah
Department of Electronic Techniques, Northern Technical University
Kirkuk Technical Institute, Kirkuk, Iraq
Email: murad.kerkukly@gmail.com

1. Introduction

Automatic control systems are systems that aim to keep the system to the desired values without requiring human intervention. In a good control system, as the input sizes change, the output sizes are expected to reach the set values as fast as possible and the changes caused by disturbing quantities are expected to be self-corrected by the system. A controller is added to the control system that does not include these features to provide the desired features. Control of systems (industrial automation systems) is widely used in PLCs instead of conventional relay control since it provides advantages such as reduced production cost, easy installation, increased reliability, and quality.

Logical Controller is realized with microprocessor-based or real-time MATLAB computer-controlled systems. PLCs support advanced industrial communication protocols and can exchange information over long

distances. This allows centralized control and monitoring of industrial automation systems over long distances that are difficult to reach [1], [14].

MATLAB Simulink can be used to analyze the systems. According to automatic control systems, analysis and control of the systems can be carried out to ensure that data is received in real-time, i.e. from a device such as data transfer card or a serial or parallel port, and at the same time, the data is transferred and processed into the MATLAB Simulink environment.

A liquid level modeled in MATLAB Simulink environment and the general structure of the flow mechanism, as well as the Simulink, real-time workshop, real-time windows target toolboxes used in the system and the usage patterns of these structures are introduced [20]. The most widely used feedback controller system today is the Proportional-Integral-Derivative (PID). In the automatic control industry, most control loops are based on PID controllers, which are of economic importance for industrial processes [15]. PID controllers are indispensable elements of industrial systems. A PID controller uses (error value), which is the difference between the desired output value and the measured output value, as input [16]. The controller works to achieve the desired set value by minimizing the error by adjusting the system input. To achieve an error-free control, the controlled signal is fed back and compared to the reference signal. A drive signal proportional to the difference between the input and output signals are calculated by the PID controller and applied to the system to correct the error [2], [14].

This study aims to determine the positive and negative aspects of both systems by using PLC software in liquid level and flow control systems or PID controller created in real-time MATLAB Simulink environment and then determine which method is suitable for which system. The below table shows the parameters used in this study.

Table 1. Key parameters

Abbreviation	Explanation	Abbreviation	Explanation
Kp	Proportional gain	τ	The time constant in seconds
Ki	Integral gain	τd	The time delay in seconds
Kd	Derivative gain	g	the gravitational acceleration
L	Liquid level	a	the cross-sectional area of the exit hole at the bottom of the tank
Qin	Flow input	A	The cross-sectional area of the tank
$Qout$	Flow output	PID	(Proportional - integral - derivative) controller
Vp	Motor voltage	Ti, Td	Integral time, Derivation time
R	Resistance	PLC	Programmable Logic controller

For general PID controller design, the following steps are taken to obtain the desired response in PID controller design [3], [5]:

1) There is an open-loop response and requirements are determined (rise time, settling time, maximum overrun, etc.).
2) A proportional controller is added to shorten the rise time.
3) An integral controller is added to remove the steady-state error.
4) A derivative controller is added to reduce the over-run.
5) *Kp, Ki,* and *Kd* are adjusted until the desired response is achieved.

We can find out which controller controls which characteristic by using Table 2.

Table 2. The result of adding *Kp, Ki* and *Kd* controllers on a closed-loop system

Abbreviation	Rise-time	Over-run	Settlement-time	Steady-state error
Kp	reduced	increased	changed slightly	reduced
Ki	reduced	increased	increased	disappeared
Kd	changed slightly	reduced	reduced	changed slightly

2. Methods and models

2.1. Ziegler-Nichols method

One of the most common methods is the Ziegler-Nichols method. There are two different methods of Ziegler–Nichols; the first is the oscillation method which can be applied if the controlled system can enter full oscillation. The second Ziegler-Nichols method is called (the reaction curve) method [4]. In this method, PID coefficients are found by the reaction curve. The open-loop function of the system, the unit step function is examined by investigating the value shown in the graph in Figure 1. This answer is equivalent to the response of a first-order time delay function to the unit step function as in (1).

$$G(s) = \frac{K.e^{-\tau_d.s}}{1+\tau.s} \tag{1}$$

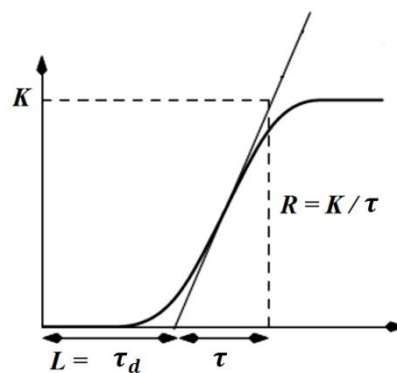Figure 1. Ziegler-Nichols reaction curve [16].

These values are obtained by using Table 3 formulas to obtain ideal PID coefficients. Where τ is the time constant (*s*) and τ_d is the time delay (*s*).

Table 3. Ziegler-Nichols reaction curve method

Abbreviation	Kp	Ti	Td
P	$\dfrac{\tau}{K.\tau_d}$	∞	0
PI	$0.9\dfrac{\tau}{K.\tau_d}$	$3.\tau_d$	0
PID	$1.2\dfrac{\tau}{K.\tau_d}$	$2.\tau_d$	$0.5\,\tau_d$

2.2. Cohen-Coon method

This method is very similar to the Ziegler-Nichols reaction curve. Figure 1 shows the difference in the calculation of PID parameters. Table 4 is used for these calculations.

Table 4. Cohen-Coon method coefficients

Abbreviation	Kp	Ti	Td
P	$\frac{\tau}{K.\tau_d}\left(1+\frac{\tau_d}{3.\tau}\right)$	∞	0
PI	$\frac{\tau}{K.\tau_d}\left(0.9+\frac{\tau_d}{3.\tau}\right)$	$\frac{\tau_d\,[\,30\,\tau+3\,\tau_d]}{[\,9\tau+20\,\tau_d]}$	0
PID	$\frac{\tau}{K.\tau_d}\left(\frac{4}{3}+\frac{\tau_d}{3.\tau}\right)$	$\frac{\tau_d\,[\,32\,\tau+6\,\tau_d]}{[\,13\tau+8\,\tau_d]}$	$\frac{[\,4.\,\tau_d\,.\,\tau]}{[\,11\tau+2\,\tau_d]}$

3. System model

The system used in this study consists of a pump and two liquid tanks with a vent. There is a liquid flow sensor at the outlet of the pump and a level detection sensor in the first tank. All tanks have the same diameters and valves will change these outputs. In this lab set, the pump supplies the (Tank 1) and the output of the (Tank 1) supplies the (Tank 2). The outlet of the tank 2 is reconnected to the pump, thereby ensuring a continuous flow of liquid.

3.1. Tank model of fluid systems

In general, industrial liquid processes consist of pipes and tanks. In all fluid flow systems, the flow frequently causes turbulence. Flow systems are generally represented by nonlinear differential equations. The definition of the resistance and capacitance of the liquid level is explained by simple formulas and dynamic properties. The rate of liquid level change over time is given for each tank.

The fluid level (L) can be calculated with the provided input-output difference of the flow $Qin\,(t) – Qout\,(t)$ and the constant cross-sectional area A of the tank

$$\frac{Lt}{dt} = \frac{1}{A}\left(Q_{in}(t) - Q_{out}(t)\right) \tag{2}$$
$$Q_{in}(t) = K_p V_p \tag{3}$$
$$Q_{out} = a\sqrt{2gL} \tag{4}$$

The output ratio for each tank is given by (4), where (g) means gravity acceleration and (a) cross-sectional area of the outlet opening at the bottom of the tank. The liquid level dynamic equation of the tank (5) was obtained using the expression as in (2).

$$\frac{L(t)}{dt} = \frac{K_p}{A}V_p(t) - \frac{a}{A}\sqrt{2gL(t)} \tag{5}$$

In fluid systems, elements such as capillary tubes, valves, and porous plugs, which make flow difficult and restrict, become resistance elements [3]. The connection between the (Q) flows through these resistors is shown in (6). Fluid resistance and electrical resistance are completely similar. If the internal variable flow as input and three variable levels as output are taken, the fluid resistance transfer function is as shown in the equation (7).

$$L = R\,Q \tag{6}$$

$$R = \frac{dL}{dQ} \cong \frac{8\eta\,h}{\pi r^4 \rho g} \tag{7}$$

$$\frac{L(s)}{Q(s)} = R \tag{8}$$

Meanwhile, the *dt (In-Out)* is equal to the extra amount contained in the water tank over the small-time interval, and the Q-H relationship is shown in the differential equation for the fixed resistance as follows. Tank capacitance; The amount of liquid (m^3) required changing to one unit in the tank at the liquid level (m), the capacitance of the tank is proportional to the cross-sectional area, if the cross-sectional area is constant, and the capacitance is constant at any level [19].

$$C = \frac{stored\ fluid\ change}{liquid\ level\ change}$$

$$dL = (q_{i-}q_o)dt \tag{9}$$

$$q_o = \frac{L(s)}{Q(s)} \tag{10}$$

$$RC \frac{dl}{dt} + 1 = R\,q_i \tag{11}$$

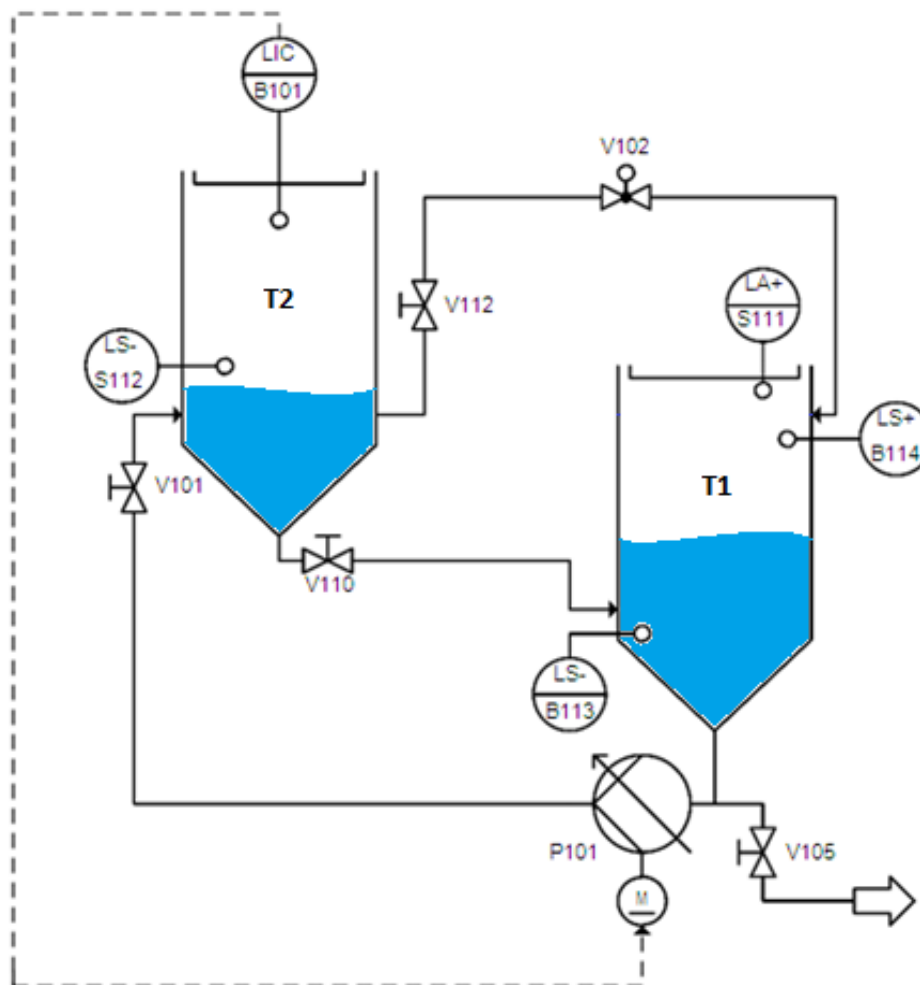$$\frac{L(s)}{Qi(s)} = \frac{R}{RCs+1} \tag{12}$$

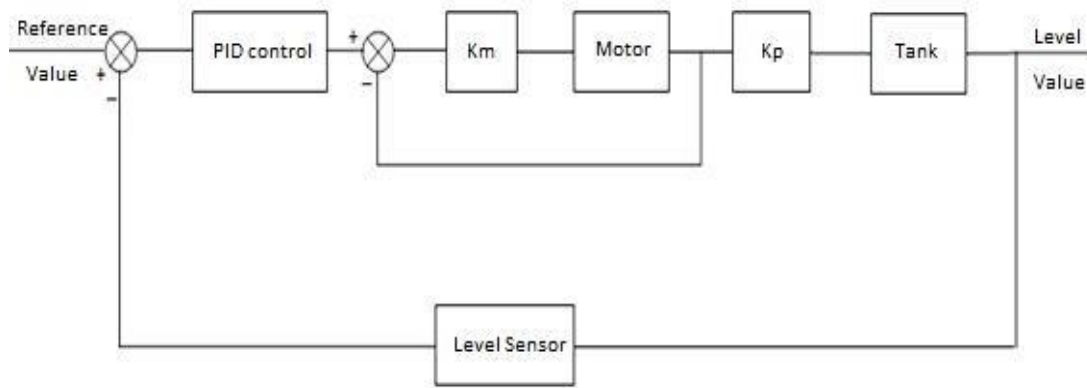Figure 2. Liquid level control system diagram

Figure 3. Liquid level control block diagram

3.2. System control with PLC

PLC is designed for industrial applications, it is an electronic device that controls a system with the function written according to digital principles using input and output cards and provides general control with the functions of timing, counting, storage, and arithmetic operation [18]. In automatic control systems, in addition to speed, control, safety, and product quality, the reconstruction of the control circuits for the manufacture of a new product, the assembly, and removal of the connections by PLC programming alone was a huge advantage. This has brought the use of PLC-based control systems as an indispensable controller in industrial automation circuits. The ladder is the most common programming language. However, in complex applications and programs that require intensive mathematical and system-related block software, STL becomes more prominent [24].

3.3. Realization of PID control with MATLAB real-time simulınk

The MF 624 multifunctional *I/O* card is designed for the need to connect real-world signals, the MF 624 multi-function *I/O* board is equipped with cable extender, which is suitable for desktop use, with 37-point D-type female *X1* and additionally with 37-point D-type female *X2* connectors [23]. To run a simulation or to create a real-time application, a new model is created by opening the Simulink library in the Matlab Command Window. In the model, the input and output of the system are realized with analog input and output blocks. PID block "Simulink Extra" library is added to the model from the browser [17]. Within the reference value, the signal generator is added. Analog input and output blocks have been added to the Simulink model and parameters have been set to enable the input and output of the controlled system to the computer environment [21]. As shown in Figure 4. Closed-loop connections have been established by establishing inter-block connections. The PID controller parameters were applied to the PID block by using the "Ziegler Nichols" method and optimized by selecting the best settling time value for several trials.
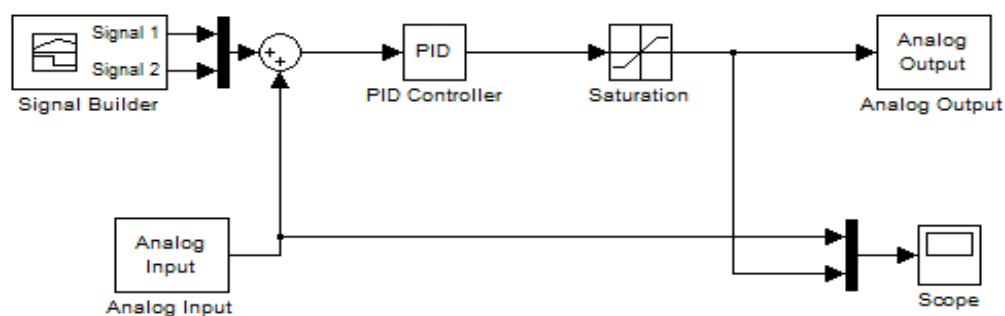

Figure 4. Closed-loop PID control model

4. Results

4.1. PID flow control results with MATLAB and PLC

The PID control algorithm was applied to the model of the flow control system which was found by utilizing the system recognition tool. The parameters of the PID controller "Kp, Ki and Kd were determined using the Ziegler-Nichols method. It has been optimized by making a few trials with the shortest settling time and minimum overrun of these values. The sampling time for the flow system was selected as 0.01s, upon applying the same values to both systems $Kp = 3,4$ $Ki = 3,7$ and $Kd = 0,55$, the results were compared.

The real-time results obtained by the PID algorithm of the flow control are shown in Figure 5 and Table 5 for the MATLAB part, the rising time of the desired value, the settling time of 1% and the steady-state error are shown. For the PLC part, the same values were determined in Table 6 and Figure 6.
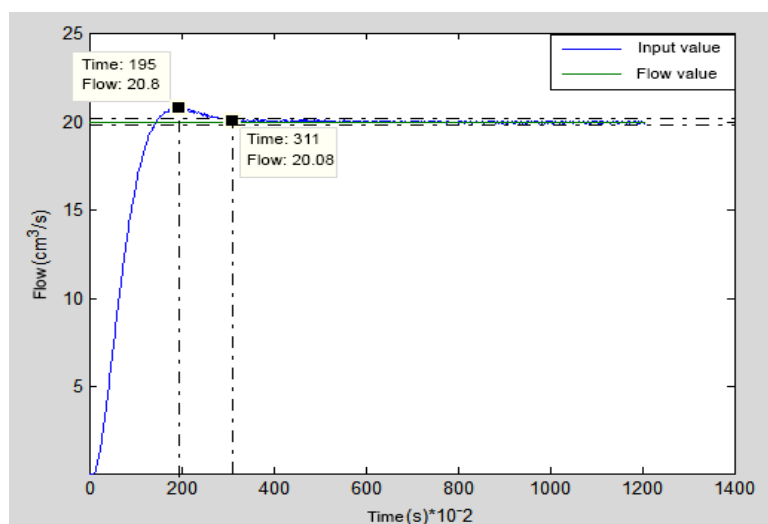

Figure 5. Matlab flow PID control

Table 5. Matlab flow PID control output

	Rise time	Settling time	Steady-state error
Time (s)	1,95	3,11	0,006
Flow (cm^3)	20,8	20,08	

Table 6. Flow PID control output

	Rise time	Settling time	Steady-state error
Time (s)	3,01	4,2	0,004
Flow (cm^3)	20,93	19,8	

This PID control experiment performed with both methods was obtained at different flow values and the results were obtained and plotted on top of each other for comparison of the two methods are shown in Figure 7 when these results are examined.
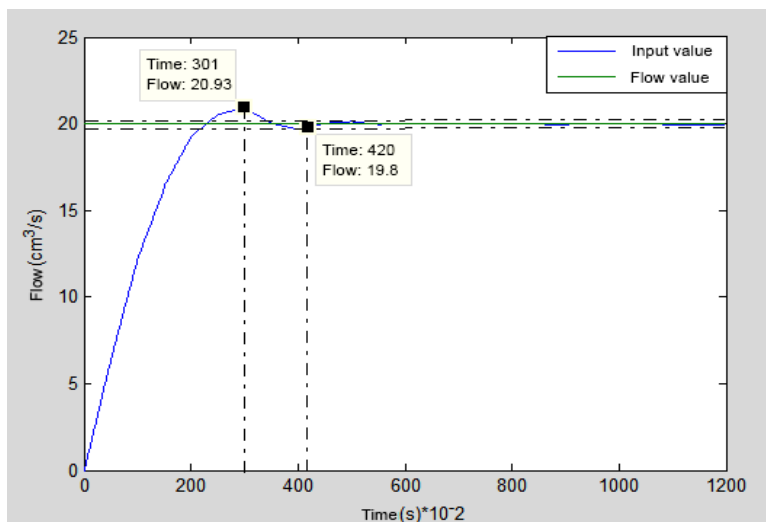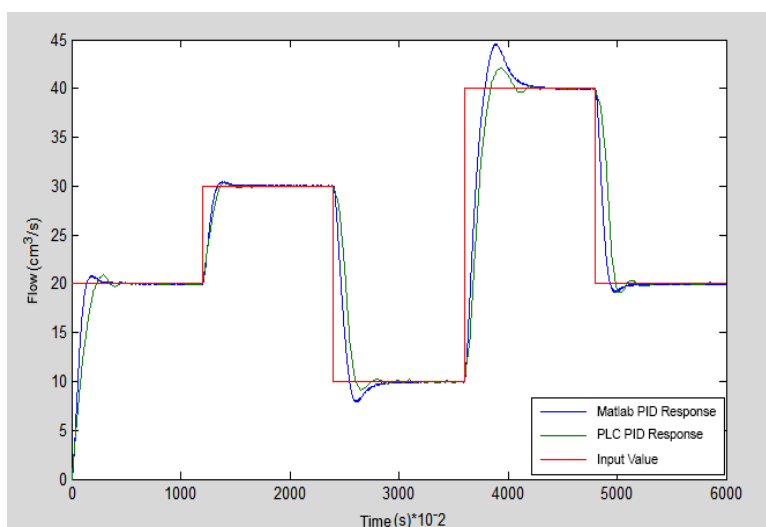
Figure 6. PLC Flow PID control

Figure 7. Matlab and PLC output of PID Flow control according to different reference values.

As shown for flow control, different values were selected and applied, these values 20, 30, 10, 40 and 20 were selected. The purpose of selecting different flow values is to see and examine the response of the PID method to flow control at different values. By keeping the selected Gain, integral time and derivative time constant, Extreme values show different excesses, but these excesses do not reach 15%. In this experiment conducted over Matlab and PLC, Excess and % 1 settling values are given in Table 7.

Table 7. Matlab and PLC outputs of PID flow control at different values

	$20\ cm^3/s$	$30\ cm^3/s$	$10\ cm^3/s$	$40\ cm^3/s$	$20\ cm^3/s$
MATLAB	20.82	30.44	7.91	44.55	19.18
Over	% 4	% 1.5	% 10.05	% 15.1	% 4.1
MATLAB Settling Time (s)	3.07	3.05	4.88	6.02	3.1
PLC	20.92	30.49	9.07	42.14	19.12
Over	% 4.6	% 4.9	% 4.46	% 7.1	% 4.4
PLC Settling Time (s)	4.2	4.6	5.2	6.02	4.26

According to these results, the flow PID control output response with MATLAB is faster than the PLC, also it shows different overruns at the highest and lowest flow rates.

On the other hand, in the same experiment performed with PLC, the output responses have no difference with MATLAB, but they were a little bit slower.

4.2. PID level control results with MATLAB and PLC

In the level control system, as in the case of flow control, the model of the system was obtained from the system recognition tool. The control system was controlled using the PID algorithm. The parameters of the PID controller "Kp, Ki and Kd were determined using the Ziegler-Nichols method [22]. It has been optimized by making a few trials with the shortest settling time and minimum overrun of these values. The sampling time for the level system was selected as 0.1 s, after applying the same values "$Kp = 16$ $Ki = 1.67$ and $Kd =$ 0.55 to the system in both ways, the results were compared. In the level control system, the PID algorithm is used to keep the level at the intended value. The PID algorithm was implemented via a computer-based Matlab Simulink program and PLC. Figure 8 for Matlab and Figure 9 for PLC displays the results of these two experiments done in both ways.
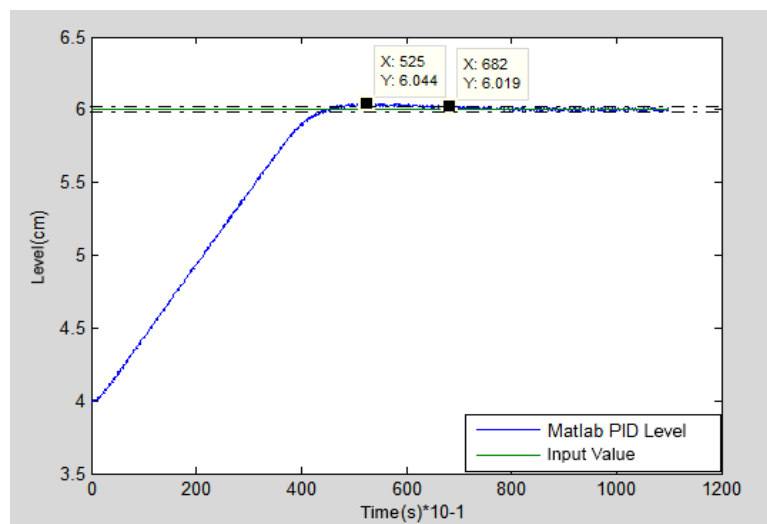

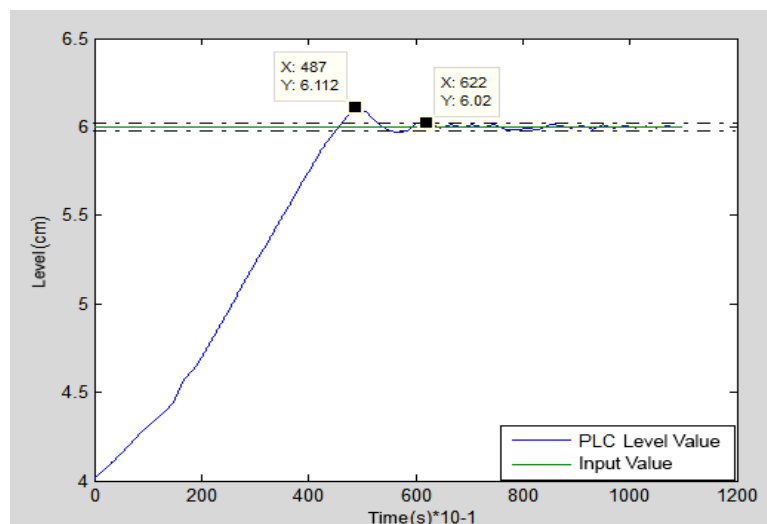Figure 8. Matlab level PID control

Figure 9. PLC level PID control

Different reference values were applied for the liquid level to be checked at 4,6,2,8 and 4 liters levels and the results of the system were obtained through MATLAB application and PLC application and both results were plotted on top of each other to see the different levels of the response of the PID System control. Figure 10 shows the results made by both methods.
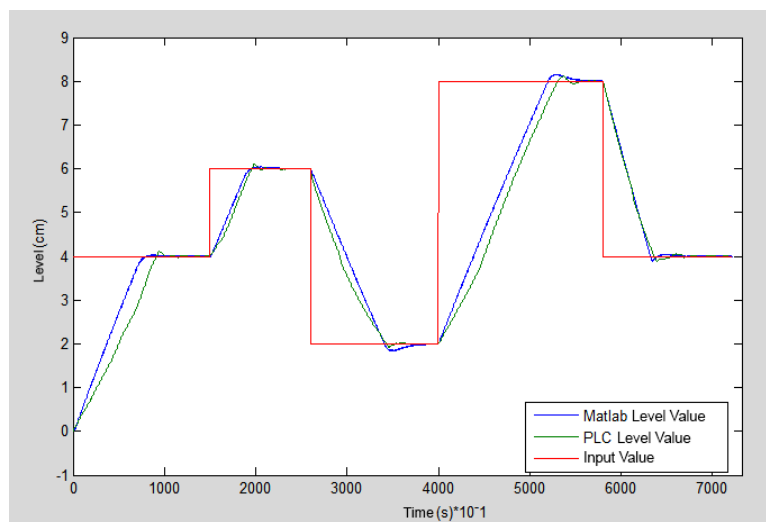

Figure 10. Matlab and PLC output of PID level control according to different reference values.

Table 8. Matlab and PLC outputs of PID level control at different values

level control	4 cm	6 cm	2 cm	8 cm	4 cm
MATLAB over	4.06 % 1.5	6.044 % 2.2	1.836 % 4.1	8.156 % 2.51	3.88 % 3
MATLAB settling time (s)	93.5	69.4	95.7	140	78.6
PLC over	4.109 % 2.8	6.111 % 5.5	1.915 % 2.1	8.134 % 2.2	3.89 % 2.9
PLC settling time (s)	99.3	62.2	89.3	145	84.5

5. Conclusions

In this study, flow and liquid level control from process control are discussed. In these two experiments, the PID algorithm was written in both (lab MATLAB Simulink real-time workshop) and (PLC S7300 SIMATIC manager), then the necessary analog input and output connections were established. Based on the results obtained, the advantages of PID control over MATLAB Simulink and PLC in real-time experiments in the laboratory were determined. The models of the system were determined using the MATLAB system recognition tool. Ziegler-Nichols and Cohen-Coon methods were used for the determination of PID coefficients, but high overrun and late settling time was observed when the coefficients found by these methods were used. PID coefficients have been optimized as a result of various attempts to eliminate them. System overrun is reduced and a stable response is obtained. According to the results obtained by the control of MATLAB PID on the MF624 data acquisition card in a computer-based real-time application, it is found to be more effective in fast systems. Looking at the results of the flow control system, PID control results applied with Matlab Simulink fit less time. On the other hand, PID control with PLC and PID control with MATLAB do not show much difference in the level control system which reacts more slowly.

References

[1]   Aström K.J. and Hagglund T, PID Controllers: Theory, Design, and Tuning. 2nd edition, NC: Instrument Society of America (3), 59-117, 1995.

[2]   Graham C. Goodwin, Stefan F. Graebe, Mario E. Salgado, Control System Design, (6), 157- 172, 2000.

[3]   Real-time Workshop User's Guide, the Math Works Inc., September 2000.

[4]   Jing-Chung Shen Huann-Keng Chiang, PID Tunning Rules Of Second-Order Systems. Department of Automation Engineering National Huwei Institute of Technology Huwei, Yunlin, Taiwan, 2002.

[5]   Gwo-Ruey Yu and Rey-chue Hwang, Optimal PID Speed Control of Brushless DC Motors Using LQR Approach, IEEE international conference on systems, Electrical Engineering Department I-Shou University Kaohsiung County, Taiwan, 2004.

[6]   A. Calvallo, A. Di Nardo, M. Di Natale, and C. Natale, "An optimal fuzzy approach to automated reservoir management", *IFAC'papers*, July, Prague, 2005.

[7]   D. Sbarbaro and R. Ortega, "Averaging level control of multiple tanks: A Passivity Based Approach", *CDC-ECC'05 papers*, December, pp. 7384- 7389, Seville, 2005.

[8]   Hug Jack, "Automating Manufacturing Systems with PLC", V4.7 (25), 622- 627, 2005.

[9]   M.Azzouzi, and D. Popescu, "Optimal control of liquid level", *IA'2006 papers*, September, vol. 2, pp. 479-488, IASI, 2006.

[10] M.zzouzi and D. Popescu, "Liquid level control using optimization and identification software"*CONTI'2006 papers*, Jun, pp. 81-84, Timisoara, 2006.

[11] Berger H, "Automating with STEP 7 in STL and SCL: Programmable Controllers SIMATIC S7-300/400", Wiley-VCH, (3), 187-191, 2007.

[12] M. J. Lengare, R. H. Chile, L. M. Waghmare and Bhavesh Parmar, Auto-Tuning of PID Controller for MIMO Processes. *International Journal of Electrical, Computer, Energetic, Electronic and Communication Engineering*, Vol. 2, No.9, 2008.

[13] Bijay Kumar, Rohtash Dhiman, Optimization of PID Controller for liquid level Tank system using Intelligent Techniques", *Canadian Journal on Electrical and Electronics Engineering* Vol. 2, No. 11, 2011.

[14] Bernard Friedland, Control System Design: An Introduction to State-Space Methods, Dover Publications, ISBN: 978-0486442785, March 2005.

[15] Radu Duma, Mirela Trusca, Petru Dobra, "Tuning and Implementation of PID Controllers using Rapid Control Prototyping", *Control Engineering and Applied Informatics*, Vol.13, No.4, pp. 64-73, 2011.

[16] Rames C. Panda, Introduction to PID Controllers: Theory, Tuning and Application to Frontier Areas, ISBN: 978-953307927-1, DOI: 10.5772/2422, February 2012.

[17] Marjan Golob, Bozidar Bratina, "Web-Based Monitoring and Control of Industrial Processes Used for Control Education", *The International Federation of Automatic Control*, Sheffield, UK, August 28-30, 2013.

[18] Swetla Barwal, Hari Mohan Rai, Seema Das, Rinku Sanghwan, "PID And Fuzzy-PID Controller For Various Electrical Machine Applications: A Comparative Study", *International Journal of Advanced and Innovative Research* (2278-7844), Vol. 2, No. 11, 2013.

[19] H.I Jaafar, S.Y.S Hussien, N.A Selamat, M.S.M Aras, M.Z.A Rashid, "Development of PID Controller for Controlling Desired Level of Coupled Tank System", *International Journal of Innovative Technology and Exploring Engineering (IJITEE)*, ISSN: 2278-3075, Vol. 3, No. 9, February 2014.

[20] Sk Hasan Hafizul Haque, H. M. Imran Hassan, S. M. Abul Hossain, "Comparison of Control System Using PLC & PID", *ASEE 2014 Zone I Conference*, University of Bridgeport, Bridgeport, CT, USA, April 3-5, 2014.

[21] S.Y.S Hussien, H.I Jaafar, N.A Selamat, F.S Daud, A.F.Z Abidin, "PID Control Tuning VIA Particle Swarm Optimization for Coupled Tank System", *International Journal of Soft Computing and Engineering (IJSCE)*, ISSN: 2231-2307, Vol. 4, No. 2, May 2014.

[22] Amruta S.Jondhale, Varsha J.Gaikwad, Satish R. Jondhale, "Level Control of Tank System using PID Controller-A Review", *International Journal for Scientific Research & Development*, ISSN (online): 2321-0613, Vol. 3, No. 10, 2015.

[23] Pooja Panchal, Prof. Alpesh Patel, "Interfacing of PLC with NI-Lab-view using Modbus Protocol", *International Journal of Advance Engineering and Research Development*, ISSN: 2348 – 4470, Vol. 6, No. 1, pp. 19-26, Jan. 2015.

[24] Haoqiang Ji, "PLC Programming for a Water Level Control System: Design and System Implementation" University of Victoria, 2017.