Modeling and implementation of a proportional hydraulics system to control the position and pressure of a plastic extruder

[1,a] Rivera F. Jonathan, [1,b] Muñoz C. Viviana, [1,c] Sánchez O. Wilson,

[1] Departamento de Eléctrica y Electrónica, Universidad de las Fuerzas Armadas ESPE, ID: 60104598, Sangolquí, Ecuador

[a]jarivera10@espe.edu.ec, [b]vbmunoz1@espe.edu.ec, [c]wesanchez@espe.edu.ec

Abstract*— Model and implement a hydraulic system with fuzzy position control of a double acting cylinder and PID pressure control of a hydraulic motor on a plastic extruder. The control system will be implemented with the Festo TP 702 by means of a controller, display, graphical interface and link to Fuzzy control and MATLABTM PID_Tuner to obtain graphs. Modeling provides the process transfer function, proportional control constants and position and pressure control graphs, which are then validated between the WinCC Advanced interface graphs of the physical process implementation with the Festo S7-1200 PLC, generating a graphical comparison of the control of position and pressure variables. This results in an accuracy error of 1% between the result of the modeling and the physical implementation, which is acceptable in an industrial process. The HMI, as well as the SCADA system was performed in WinCC Basic and WinCC Advanced respectively, generating a graphical interface both local and remote in approximately real time. The mathematical modeling of an industrial process guarantees the reliability and optimal behavior of the plastic extrusion process, minimizing production costs and raw material waste.

Index Terms—**Position and pressure control, proportional control, modeling, MATLABTM, hydraulic valves.**

I. INTRODUCTION

The fuzzy controller is a technique that makes decisions at the expert's discretion. [1], the exit criterion is explicitly based on the discrete number of possible decisions for the formation of membership functions with their respective entrances and exits. [2]

The fuzzy logic controller has been studied by the authors [3] [4], determine that the control technique for the position of pneumatic actuators is simple robust and suitable for non-linear systems, has two signals setpoint and sensor, which between its difference gives the error, the performance is measured by the accuracy and robustness of the fuzzy logic controller. [4]

The use of the fixed gain PID controller is the most popular used in the industry, as it is simple, versatile and successful in linear processes. [5]

The design of a multivariable PID controller is studied by the authors [6], that use a Biggest Log Modulus Tunning algorithm, which seeks to contribute by proposing a generic model to estimate the desired behavior of the model, compared to the mathematical foundations and its algorithm. [6]

The following article presents the results obtained in the precision and stability of the position and pressure systems by means of the diffuse and PID controllers respectively, analyzed by means of their simulation and implementation. Mathematical modeling tools such as MATLABTM and SIMSCAPETM are used for this purpose, contributing to the development of control systems through a graphic environment that facilitates the analysis, design and simulation of electrical, mechanical and hydraulic control systems. [7]

The position hydraulic system uses a fuzzy controller, which imposes fuzzy control rules using classical logic with reasoning towards imprecise concepts. [8]

The hydraulic pressure system uses a PID Tuner controller, which automatically linearizes the plant and delivers the initial compensator tracking response. [9]

Once the model and mathematical simulation have been carried out, the physical implementation of the system is carried out, using a PLC S7-1500 master module that commands two PLC S7-1200 slaves that correspond to the position and pressure control respectively in a Profinet industrial network.

The programming is done in TIA Portal, for the diffuse control the data obtained from the SIMSCAPETM environment are exported, while for the PID controller, the hydraulic circuit is done in SIMSCAPETM adjusting the data of the hydraulic pump, hydraulic motor, pipe diameters, valves, type of fluid, used in the implementation of the system.

The plastic extrusion process is controlled by a local control with the implementation of an industrial HMI, which allows the operation of the system, while displaying the graphs of position and pressure. The remote control is carried out by means of a SCADA system obtained by programming the TIA Portal software in WinCC Advanced, which allows control, supervision and data acquisition to develop a complete integrated system. [10]

II. Mathematical Modeling

A. *Position control system*

The mathematical model of the position control of a double-acting cylinder is made up of a mechanism that consists of using "yes-then" sentences which are called "rules". These rules are useful, since they refer to variables and adjectives that describe the variables. The order in which the rules are used has no relevance. [11]

In the design of the mathematical model the Mamdani type inference method is used, this controller is formed by fuser, fuzzy rules and defuser, fig. 1. [12]

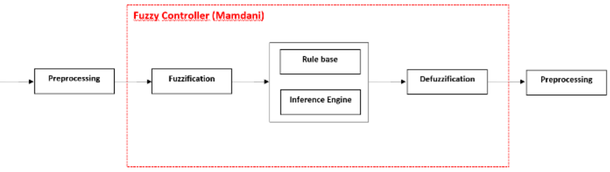

Fig. 1 Diagram of fuzzy controller type Mamdani [13]

This driver is developed using the "Fuzzy Logic Toolbox" plug-in in the MATLABTM SIMSCAPETM environment. [14] To access the plugin, type in the command window "fuzzy", from there you can develop a fuzzy controller that can be Mamdani or Sugeno, add input and output variables, and then create a set of rules. [15]

The system's Mamdani type fuzzy controller has one input and two outputs, fig. 2.

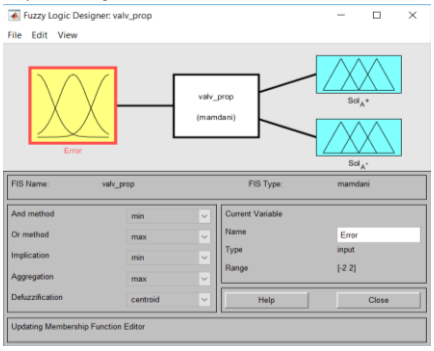

Fig. 2 Design of the Fuzzy Mamdani position control controller, MATLABTM.

The input has the linguistic variable "Error", the same one that comes from the difference between the setpoint and the sensor data. The error is positive if the value of the distance is below the value of the setpoint, and negative if the value of the distance is above that value, table I. The error is positive if the value of the distance is below the value of the setpoint, and negative if the value of the distance is above that value, table I.

TABLE I

DESCRIPTION OF TRIANGULAR MEMBERSHIP FUNCTIONS OF THE VARIABLE "ERROR"

Function	Description	Parameters
Left +	The piston is widely used	[-3 -2 -1]
Left -	Piston is slightly extended	[-1.6 -1 -0.4]
Cero	The piston is at the desired value	[-0.5 0 0.5]
Right -	The piston is slightly contracted	[0.4 1 1.6]
Right +	The piston is very contracted	[1 2 3]

For the output there are two linguistic variables, "Sol A+" and "Sol A-", which allow the progressive opening and closing of the solenoid until the value entered in the setpoint is reached. These variables are the controller's responses to the input variable "error", tables II and III.

TABLE II.

DESCRIPTION OF THE FUNCTIONS OF THE TRIANGULAR TYPE MEMBERSHIP OF THE VARIABLE "SOL A+".

Function	Description	Parameters
Maintain	Do not open the solenoid	[-5 0 2]
Open -	Open the solenoid to a value of 4	[2 4 6]
Open +	Open the solenoid to a value of 10	[5 10 15]

TABLE III.

DESCRIPTION OF THE FUNCTIONS OF THE TRIANGULAR TYPE MEMBERSHIP OF THE VARIABLE "SOL A-".

Function	Description	Parameters
Maintain	Do not open the solenoid	[-5 0 2]
Open -	Open the solenoid to a value of 4	[2 4 6]
Open +	Open the solenoid to a value of 10	[5 10 15]

The rules collect information from the system because, if there is any variation in the entry, those rules must know how to behave, each rule should result in an exit membership function. The entire linguistic process is summarized in 5 fuzzy rules, which are formed in the FLD rule editor, fig. 3.

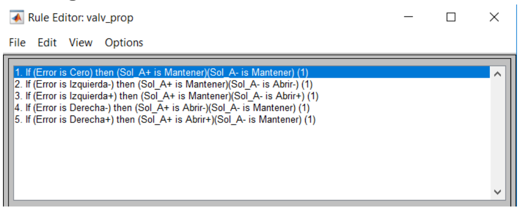

Fig. 3 Position System Rule Editor, MATLABTM.

- If the error is "0", then Sol A+ and Sol A- is "maintain".
- If the error is "left -", then Sol A+ is "maintain" and Sol A- is "open -".
- If the error is "left +", then Sol A+ is " maintain " and Sol A- is "open +".
- If the error is "right -", then Sol A+ is "open -" and Sol A- is "maintain".
- If the error is "right +", then Sol A+ is "open +" and Sol A- is "maintain".

These rules with their parameters are exported to the MATLABTM workspace, for later use in programming the S7-1200 PLC for implementation.

SIMSCAPETM is a language for modeling a physical environment of oriented objects, based on MATLABTM that allows the user to create models of systems, components and physical tools using a casual modeling approach. [16] From the point of view of modeling, "oriented object" means that a model can be constructed almost real or similar to a physical system, taking the components and their characteristics to connect them in a model. [16]

The use of SIMSCAPETM is suitable to carry out the simulation of the system with the elements of the "Fluid - Hydraulics" library and the configuration with the approximately real measured parameters of the elements used in the physical implementation.

Therefore, the "Fuzzy Logic Controller" block is entered, the FIS name parameter is configured, in which the name of the variable that was previously exported "valv_prop" is entered in order to extract the information from the fuzzy control created.

Then the fuzzy controller is designed and assigned two inputs "Inport", "1" and "2" that generate the system setpoint, i.e. the setpoint input and the values of the position sensor. Followed by the "Add" block, this block creates the "error" mentioned above.

The "Saturation" block generates an output signal which is the input value limited to the upper and lower saturation values. [17] The "Fuzzy Logic Controller" block is used for control, in which the fuzzy control rules are assigned. Then the "Demux" block is used, which extracts the components of an input vector signal and emits separate signals. Finally, two "Outport" outputs, which creates an output port for the subsystem. In this case the outputs are "Sol A+" and "Sol A -", fig. 4.

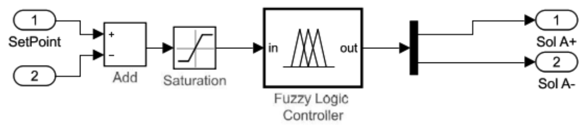

Fig. 4. Diffuse position control subsystem, SIMSCAPETM.

We proceed to create a subsystem with the elements that make up the control part, i.e. the elements of fig. 4. Therefore, we obtain a block with the name "Subsystem" in which the fuzzy control subsystem is immersed.

A closed-loop control model is generated, where the setpoint values are entered using the "Uniform Random Number" block that generates random numbers uniformly. [18] In the control block there is the subsystem that has programmed the fuzzy control that commands the position system, then the hydraulic system block, formed by a double effect piston "Cylinder", a 4-way valve that allows the bi-directional flow of hydraulic fluid, a hydraulic pump and a position sensor in which the precision and stability of the system controlled by fuzzy logic can be visualized. Finally, it is connected to the "Scope" with the two setpoint signals and position sensor, fig. 5. Subsequently, the subsystem data will be used for the physical implementation.

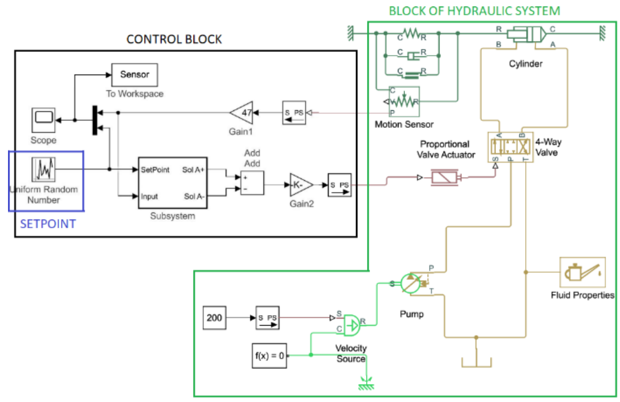

Fig. 5. Diagram of fuzzy position control system, SIMSCAPETM.

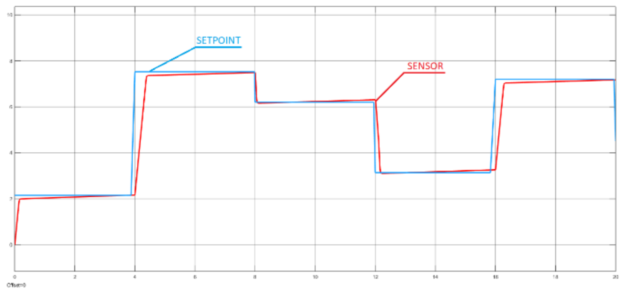

Fig. 6. Diffuse controller response to random setpoint (setpoint), SIMSCAPETM.

Fig. 6 shows the two responses, the random setpoint values, and the diffuse type-controlled system response through the values of the position sensor, which approximates according to the rules declared in Fig. 3.

B. *Pressure control system*

The mathematical model of the hydraulic motor pressure control is controlled by a PID that calculates the error between the measured value and the desired value. The interaction of parameters P, D and I set the desired process value. [19], fig. 7.

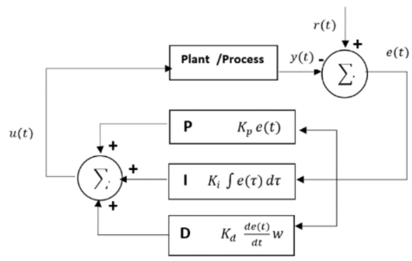

Fig. 7. Representative PID control diagram. [20]

The mathematical model is designed using the MATLAB PID Tuner plug-in, and three steps are considered to generate the model:

- Importing the input test data from the output.
- Identification of the plant model.
- The use of the model generated within the Control

System Toolbox environment. [21]

In order to import data, it is necessary to implement the hydraulic circuit of pressure control, and obtain the measured values of input-output of the system generated in the Data logging. [22], the same ones that are imported into the MATLABTM workspace.

In the PID tuner add-on you select "Identify New Plant", after this since the values are arbitrary in the implementation you select "Arbitrary I/O Data" where you enter the input (setpoint) and output (sensor) data, the same that are separately in the workspace.

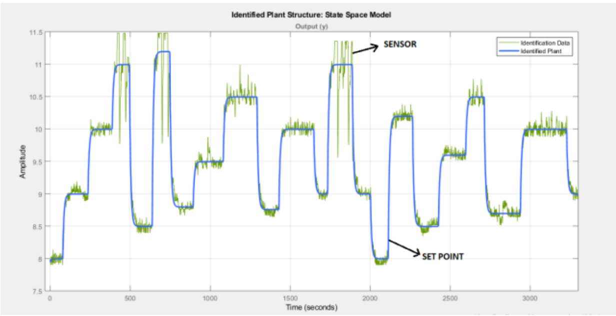

Fig. 8. Identified plant structure (physical implementation data), MATLABTM.

Fig. 8 shows the values exported by PLC S7-1500 data logging, the physical plant behaviour, the setpoint data and the system response by means of the pressure sensor data, which will be taken to MATLABTM by means of the "model of chosen order" command. This structure facilitates the ability to vary the order of the transfer function obtained as a result of the PID tuner, fig. 9.

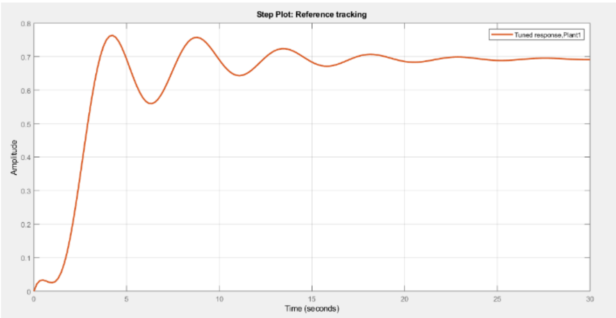

Fig. 9. Transfer function of the identified plant, MATLABTM.

The transfer function obtained from the third degree represents the relationship that describes the dynamics of the system under consideration. [23] In order to obtain the PID constants from the transfer function, the Relay method is applied, it is one of several ways to obtain the parameters (Kd, Tp, Ti). This method is also known as the Åström and Hägglund method. [24], which determines the end point using a relay test signal as the system input. [25]

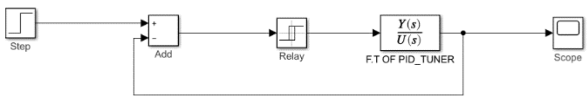

Fig. 10. Application model of the relay method, SIMSCAPETM.

Fig. 10 shows the model for estimating the end point or relay method, the only design parameter that can be changed is the amplitude of the relay, which is inherently small. To obtain the PID parameters are calculated using the Ziegler-Nichols adjustment rules [26], table IV.

TABLE IV.
ZIEGLER-NICHOLS PID CONTROLLER PARAMETERS [25]

Controller	Kp	Ti	Td
P	$0.5 * K_u$	-	-
PI	$0.45 * K_u$	$T_u/1.2$	-
PID	$0.6 * K_u$	$T_u/2$	$T_u/8$

The Ziegler-Nichols equations used to obtain the parameters of a PID control are:

$$k_u = \frac{4*d}{\pi*a} \quad (1)$$

$$k_p = 0.5 * k_u \quad (2)$$

$$T_i = \frac{T_u}{2} \quad (3)$$

$$T_d = \frac{T_u}{8} \quad (4)$$

Where:

d is relay amplitude

a is the amplitude of the graph

Tu you are the period of the graph

k_p proportional constant

T_i integration time

T_d derivation time

In the graph obtained from the model represented in fig. 10, the values Tu and a are obtained, fig. 11, with these values the parameters of the PID controller of the Ziegler-Nichols rules are found and the previous equations are solved.

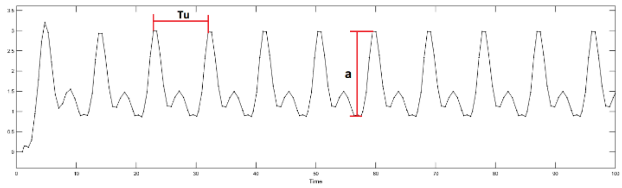

Fig. 11. System response with relay method, SIMSCAPETM.

The parameters in the graph: Tu= 9.15 and a= 2.095. The result is the PID behavior graph, fig. 12, with the calculated parameters of the Åström and Hägglund method together with the Ziegler-Nichols rule.

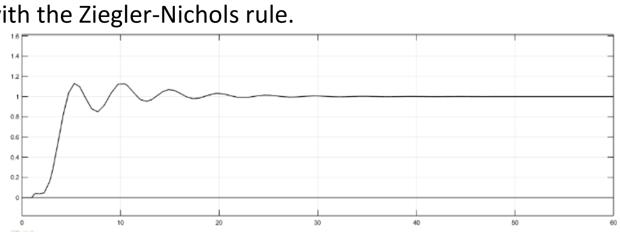

Fig. 12. PID transient response with calculated parameters, SIMSCAPETM.

The values of the calculated parameters correspond to Kp= 3.6465, Ti= 0.4575, Td= 0.1143. By obtaining a suitable response with the calculated parameters, the simulation of the physical system can be performed in the SIMSCAPETM environment of MATLABTM.

The closed-loop control model is created, consisting of the entry of setpoint values using the "Uniform Random Number" block that generates random numbers uniformly. In the control block there is the PID controller that has been programmed the derivative integral proportional control with their respective previously calculated parameters, Kp, Ti and Td. Next, the hydraulic system block, made up of a hydraulic motor, a proportional pressure limiting valve that allows the passage of hydraulic fluid to increase or decrease the pressure to the system, a hydraulic pump and a pressure sensor in which the precision and stability of the system controlled by PID integral proportional derivative control can be visualized. Finally, it is connected to the "Scope" with the two setpoint and pressure sensor signals, fig. 13.

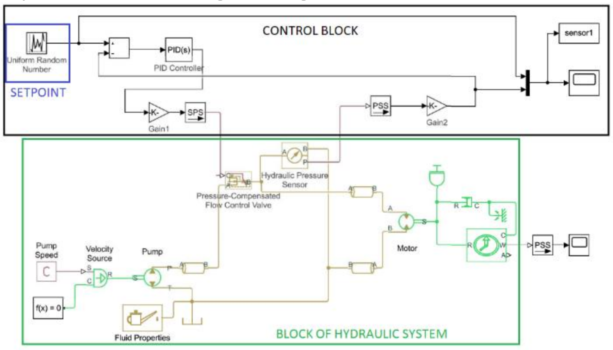

Fig. 13. Diagram of the PID pressure control system, SIMSCAPETM.

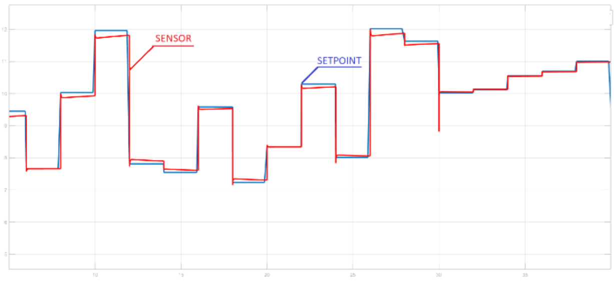

Fig. 14. PID controller response to random setpoints (setpoint).

Fig. 14 shows the two responses, the random setpoint values, and the PID controlled system response through the pressure sensor values, which approximates according to the calculated parameters Kp, Ti, and Td.

III. Implementation

The physical implementation of the hydraulic system of position and pressure is summarized in fig. 15 and 16 respectively.

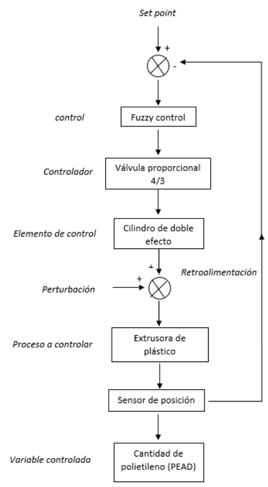

Fig. 15. Block diagram of the position control system.

The position control system commands a double acting piston by means of a 4/3 valve that is controlled by fuzzy logic. The objective of the system is to enter data by means of a keyboard (setpoint) and by means of fuzzy control to deliver a response that achieves the values entered by keyboard.

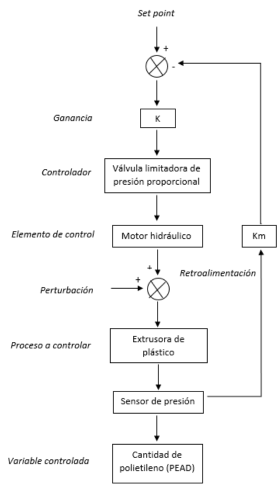

Fig. 16. Block diagram of the pressure control system.

The pressure control system commands a hydraulic motor by means of a proportional pressure limiting valve that is controlled by means of PID (Derivative Integral Proportional Control), the objective of the system is to enter data by means of a keyboard (setpoint) and by means of the PID control to deliver a response that achieves the values entered by keyboard.

The hydraulic process is programmed in TIA Portal, assigning a controller for the acquisition and storage of data, later the hardware detection is carried out for the PLC´s that carry out the control of the system and the HMI for the local control. Like the configuration in the PC system, SIMATIC HMI Application and WinCC RT Advanced for the SCADA system, table V.

TABLE V.
ELEMENTS OF THE DISTRIBUTED SYSTEM

N°	Name	Description

1	Maestro 1500	S7-1500 [CPU 1511-1 PN]
2	PLC_POSICIÓN	S7-1200 [CPU 1214C DC/DC/DC/]
3	PLC_PRESIÓN	S7-1200 [CPU 1214C DC/DC/DC/]
4	HMI_1	[TP700 Comfort]
5	PC-System_1	[SIMATIC PC station]

The elements of table V have a PROFINET industrial network scheme for communication between them, fig. 17. This allows for proper process control and data acquisition of system operation, as well as remote control via the SCADA system. [10]

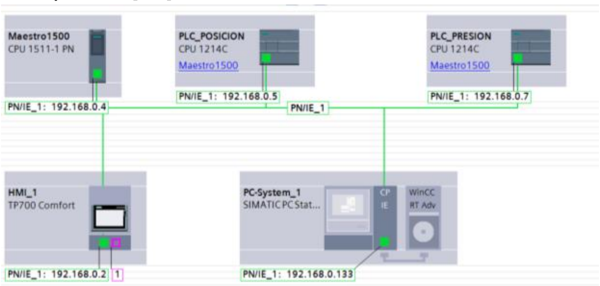

Fig. 17. PROFINET industrial network diagram of devices and equipment, TIA Portal.

PROFINET industrial communication allows control in the MAESTRO-ESCLAVOS mode, the PLC S7 1500 acts as the master. [27], internal programming has four programming blocks and a database block, fig. 18. Where the "Main_presión" block contains the configuration of the control variables of the PLC_PRESION, the "Main_Hyd" block contains the configuration of the control variables of the PLC_POSICION, the "Medicion_Hyd" block contains the configuration to start storing the setpoint and sensor variables of the PLC_POSICION, the "Medicion_Presion" block contains the configuration to start storing the setpoint and sensor variables of the PLC_PRESION and the "Registers" block stores the above.

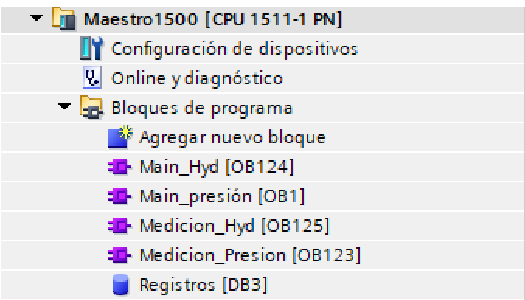

Fig. 18. Master Programming Blocks S7-1500, TIA Portal.

Variables are used for local (HMI) and remote control (SCADA). The operator of the position and pressure control processes communicates with the Master S7-1500 and through the PROFINET interface communicates with the PLC_POSICION and PLC_PRESION so that both can control the plastic extrusion process.

If you are using *Word,* use either the Microsoft Equation Editor or the *MathType* add-on (http://www.mathtype.com) for equations in your paper (Insert | Object | Create New | Microsoft Equation *or* MathType Equation). "Float over text" should *not* be selected.

A. Position control system

The programming of the position control is done in PLC_POSICION S7-1200 which is assigned as slave, table V, the programming is executed in TIA Portal, the position process already explained above contains a subsystem, the same one that carries with it the fuzzy control programming, to take the SIMSCAPETM data to the PLC controller programming software Simulink PLC Coder is used, belongs to MATLABTM, generates structured texts and Ladder type diagrams under IEC 61131-3. This product is compatible with Siemens TIA Portal. [18] The "PLC Coder" is configured from the model parameter configuration window, where "PLC Code Generation" is selected and for this case Siemens TIA Portal is selected. Once the configuration is done, the file is coded, i.e. the data of the subsystem previously generated in SIMSCAPETM is extracted, generated by the "PLC Code" option and then "Generate Code for Subsystem", when this code is generated, it is created in a text format according to IEC 61131-3 standards, for which the file must be saved as .scl.

Once the subsystem code has been generated, TIA Portal generates the programming blocks where the fuzzy controller codes will go. To do this, add the fuzzy controller file generated in MATLABTM, click on "add new external file", then on "generate blocks from source", read the file and generate the programming blocks type FB and FC functions, since the subsystem created by SIMSCAPETM needs its own functions to operate, fig. 19.

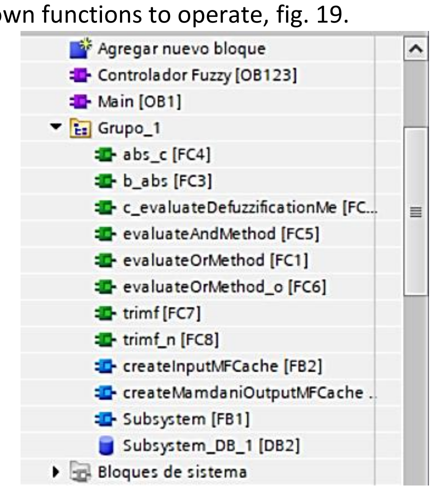

Fig. 19. Blocks generated from SIMSCAPETM export, TIA Portal.

Finally, the "Subsystem DB_1" block is added to the ordinary program blocks and the inputs and outputs of the block are configured, fig. 20.

Fig. 20 shows the "Subsystem DB_1" block, which has been programmed in MATLABTM; this block is activated by means of the "Crtl fuzzy" contact which allows the controller to be switched on. At the input of the block, the values of the system setpoint are entered locally or remotely using the "Setpoint piston" variable and the values of the system position sensor using the "Distance" variable. While in the output the response of the fuzzy controller is obtained, with the variables "Piston Output" and "Piston Input", which allow the passage and return of the hydraulic fluid respectively in the physical diagram.

The position controller programming contains analog inputs and outputs that are sent and received from the master, which are then used to perform the SCADA system and HMI display configuration.

B. Pressure control system

The implementation of this system is done in the TIA Portal software, where the PLC_PRESION is configured for pressure control by means of a PID controller. The programming contains two blocks, the first block "Main [OB1]" is programmed to control the on/off of the system, the acquisition of external signals and the sending of the PID control signal. The second block "Cyclic interrupt [OB30]" is programmed only the block "PID_Compact_1", fig. 21, is a PID controller for process control. [28], which carries out programming with the previously calculated parameters, fig. 22.

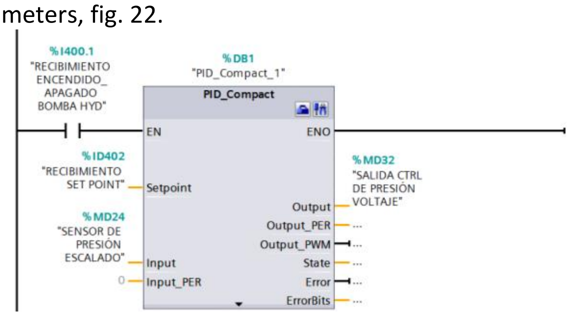

Fig. 20. Pressure control block PID_Compact_1, TIA Portal.

Fig. 21 shows the block PID_Compact_1, it is activated by the contact "RECIBIMIENTO ENCENDIDO_APAGADO BOMBA HYD" which allows the controller to activate and turn on the hydraulic pump at the same time. The block consists of two inputs and one output, the first input corresponds to Setpoint through the variable "RECIBIMIENTO SET POINT" which is the value entered locally or remotely, the second input is Input through the variable "SENSOR DE PRESIÓN ESCALADO", where the values of the system pressure sensor are entered and as output it delivers the PID control response signal in percentage by means of the variable "SALIDA CTRL DE PRESIÓN VOLTAJE", these values are scaled and parameterized in the "Main [OB1]" block.

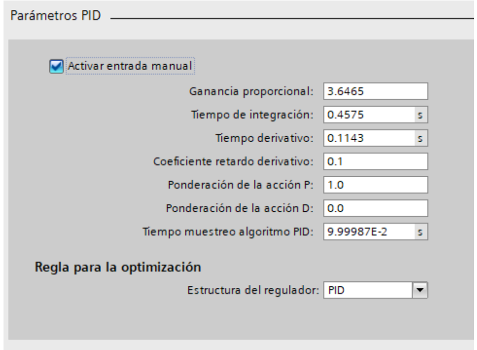

Fig. 21. Manually entered PID parameters, TIA Portal.

The fine optimization of the system is carried out within the configuration of the PID_Compact, giving values of the constants, fig. 22.

C. Control local and remote

The pressure and position control system is controlled by local and remote control. The remote control is done by programming on the HMI (Human Machine Interface) industrial display. [29] The inputs and outputs of the system are programmed by means of the HMI display, i.e. the switching on and off of the group and the hydraulic pump, setpoint inputs, as well as the visualisation of the process by means of graphs of process curves, fig. 23 and 24.

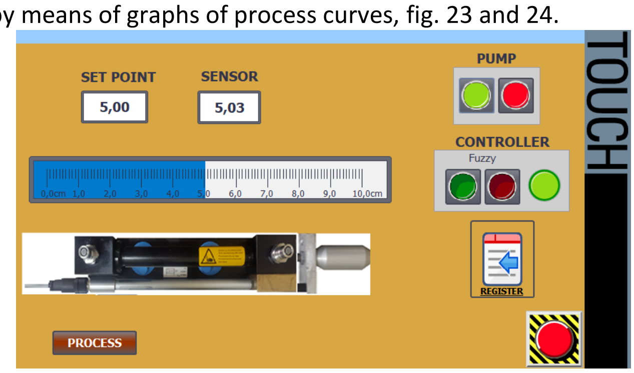

Fig. 22.Local control display (HMI) position, TIA Portal.

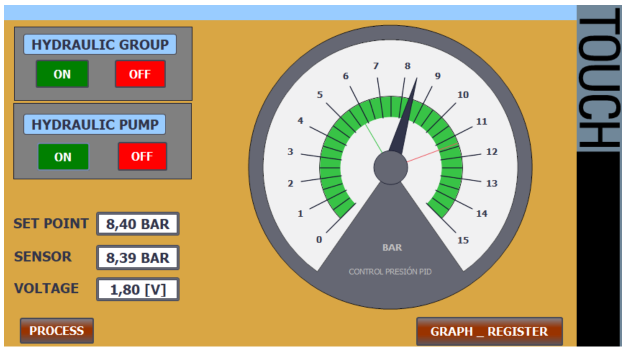

Fig. 23. Local control display (HMI) pressure, TIA Portal.

The remote control is carried out with the WinCC RT Advanced PC system, which is used as a SCADA system. [30], which will allow the control of position and pressure variables by means of curve viewers, input-output fields and the acquisition of variable data, fig. 25 and 26, by means of the Data logging block, from which CSV or TXT type files can be exported to facilitate the analysis of curves.

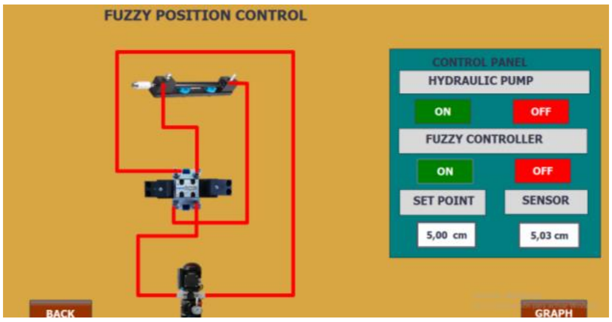

Fig. 24. Remote Control Display (SCADA) Position, TIA Portal

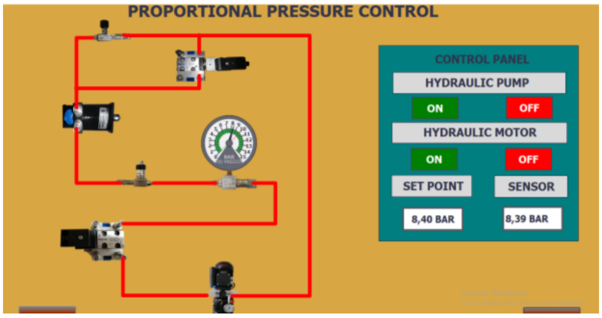

Fig. 25. Remote Control Display (SCADA) Pressure, TIA Portal

IV. Analysis of Results

Two tuning curves are obtained, fig. 27 that corresponds the first curve to the SIMSCAPETM simulated system "Fuzzy_Simulation" and the second tuning curve that is obtained from the system implemented and programmed in TIA Portal V14 "Fuzzy_Implementation".

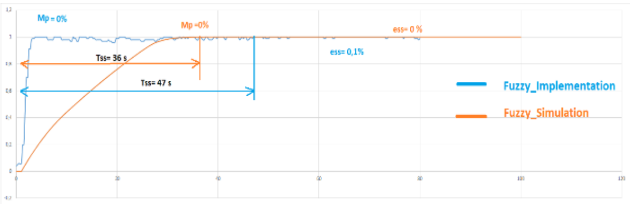

Fig. 26 Tuning curves of fuzzy application models simulation and implementation.

Fig. 27 shows in "Fuzzy_Simulation" and "Fuzzy_Implementation" an Mp of 0%, therefore, its variation range is null. The stabilization time Tss corresponding to Fuzzy_Simulation is 36 s, while the Tss of Fuzzy_Implementation is 47 s, the stability of the system is in an acceptable range in response to a change of state. The error in steady state ess in Fuzzy_Simulation is 0%, and in "Fuzzy_Implementation" is 0.1% which shows the excellent accuracy of the system, table VI.

TABLE VI.
ERRORS IN THE AUTOMATIC AND MANUAL PID TUNING CURVES

	Fuzzy_Simulation	Fuzzy_Implementation
Tss [s]	36	47
ess [%]	0	0,1
Mp [%]	0	0

The results obtained in the simulation of the process in response to random numbers uniformly in the process of position control are satisfactory, fig. 28, a precision error of 0.70% is obtained that is admissible in an industrial process.

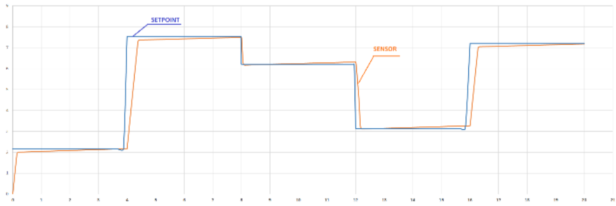

Fig. 27. Simulation of the fuzzy control of the position variable.

The results obtained in the implementation of the process in response to numbers entered locally or remotely through the setpoint in the process of position control, fig. 29, was obtained as a result of a precision error of 1.09% that like the simulation error are acceptable values in a process of industrial proportional control, noting that in this system were made several disturbances to the system, noting that the system responds in an acceptable manner to disturbances that may exist in the various industrial processes.

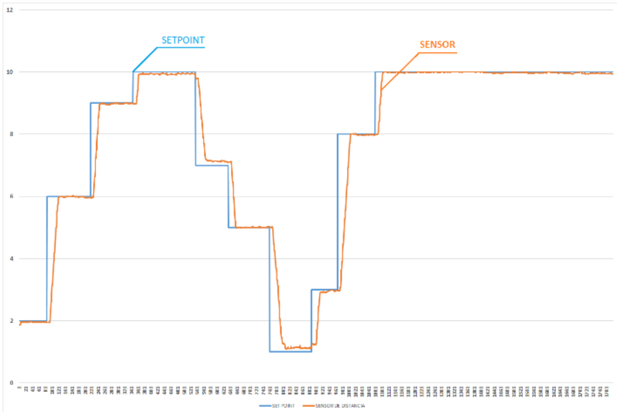

Fig. 28. Implementation of the fuzzy control of the position variable.

Two tuning curves are obtained, fig. 30 that corresponds the first curve to the system with the values of the constants Kp, Td, Ti calculated by the relay method (Åström and Hägglund) "PLC_Manual" and the second tuning curve that is obtained from the fine optimization of the PID_Compact of the TIA Portal V14 "PLC_Automatic".

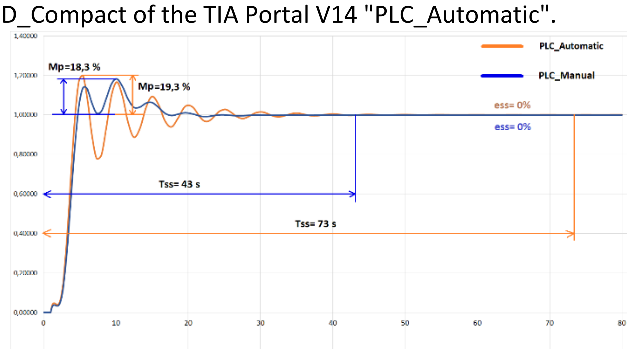

Fig. 29. Tuning curves for automatic and manual PID application models.

Fig. 30 shows an Mp over pulse of 18.3% "PLC_Manual", while the next curve of "PLC_Automatic" an Mp of 19.3%, the over pulse in both cases vary in 1%, the calculated data and those generated automatically have a percentage in moderate range. The stabilization time Tss corresponding to PLC_Manual is 43s, while the Tss of PLC_Automatic is 73s, that is, the stability of the system is in an acceptable range in response to a change of state. The error in stable state is in both cases 0%, which shows the excellent accuracy of the system, table VII.

TABLE VII.
ERRORS IN THE AUTOMATIC AND MANUAL PID TUNING CURVES

	PLC_Automatic	PLC_Manual
Tss [s]	73	43
ess [%]	0	0
Mp [%]	19.3	18.3

Unlike the PID controller, the fuzzy controller does not have over pulses, the PID controller has a maximum stabilization time of 73s, while the fuzzy system is 47s, that is 35% more and therefore the system is much more stable. Regarding the error in stable state with a difference of 0.1% the PID controller is more accurate. This data is taken in response to a step.

The results obtained in simulating the process in response to uniformly random numbers in the pressure control process are satisfactory, fig. 31, a precision error of 0.76% is obtained that is admissible in an industrial process.

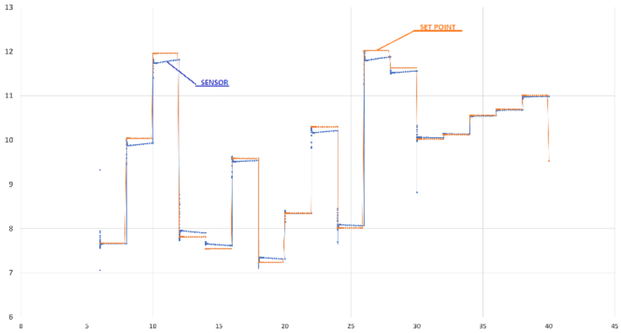

Fig. 30. Simulation of the PID control of the pressure variable.

The results obtained in the implementation of the process in response to numbers entered locally or remotely by means of the setpoint in the pressure control process, fig. 32, were obtained as a result of an accuracy error of 0.79% which, like the simulation error, are acceptable values in an industrial proportional control process, noting that in this system several disturbances were made to the system, noting that the system responds in an acceptable manner to disturbances that may exist in the various industrial processes.

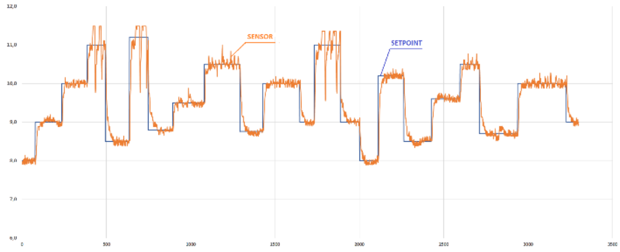

Fig. 31. Implementation of PID control of the pressure variable.

Given the percentages of accuracy in the position and pressure controllers, it can be seen that the PID controller has greater accuracy, with a difference of 0.26%, than in industrial processes is minimal.

V. Conclusion

The modeling of the position and pressure control systems performed in MATLABTM SIMSCAPETM has sufficient accuracy with a maximum error of 0.76%, indicating that industrial systems can be implemented.

The implementation of the modeled systems is done with the selected elements that allow a proper operation with a maximum error difference of 0.39% of the modeling, which is admissible due to disturbances and factors external to the system.

Data acquisition, visualization and control of the process is carried out by means of the SCADA remote control implemented in TIA portal ensures the reliability and efficiency of the system.

The communication of the control elements in the PROFINET industrial network allow operation and control in a single control unit, for further study can be carried out for different control units with wireless communication.

REFERENCES

[1] S. Mu, S. Goto, S. Shibata y T. Yamamoto, «Intelligent position control for pneumatic servo system based on predictive fuzzy control.,» *Computers and Electrical Engineering*, vol. 75, pp. 112-122, 2019.

[2] F. A., F. R. y S. E., «Control difuso una alternativa para aplicaciones de alta precisión,» *Centro de Instrumentos - UNAM*, 2015.

[3] A. Jebelli, M. C. Yagoub, R. b. HJ., A. Rahim y H. Kazemi., «Design and construction of an underwater robot based fuzzy logic controller,» *IREME International Review of Mechanical Engineering*, vol. 7, nº 1, 2013.

[4] S. Amine, O. Mokhiamar y M. Hammoud., «Fuzzy Logic Position Control of Pneumatic Actuators,» *IREME International Review of Mechanical Engineering*, vol. 12, nº 3, 2018.

[5] J. Rodríguez, *Estudio comparativo de técnicas de control PID difuso aplicadas al mejoramiento de desempeño de lazos de control continuo*, D. d. E. y. E. Facultad de Ingeniería, Ed., Bogotá: Universidad Nacional de Colombia, 2018.

[6] A. Orjuela, O. Ramos y D. Amaya., «PID Control for distilled product and bottom concentration in a binary distillation column,» *IREME International Review of Mechanical Engineering,* vol. 11, nº 4, 2017.

[7] C. Lapusan, R. Balan, O. Hancu y A. Plesa., «Development of a multi-room building thermodynamic model using SIMSCAPE Library,» *Energy Procedia,* vol. 85, pp. 320-328, 2016.

[8] W. Mendes, F. Meneghetti, R. Dutta y D. Heeren., «Fuzzy control system for variable rate irrigation using remote sensing,» *Expert System with Applocations,* vol. 124, pp. 13-24, 2019.

[9] C. Lorenzini, A. S. Bazanella, L. F. A. Pereira y G. R. G. d. Silva., «The generalized forced oscilation method for tuning PID controllers,» *ISA Transactions,* vol. 87, pp. 68-87, 2019.

[10] W. Sánchez, Ó. Proaño, P. Salazar, E. Salazar, W. Loza y J. E. Sánchez., «Distributed System SCADA using Multiple Industrial Communication Protocols on Allen Bradley Equipment,» vol. 12, nº 14, 04 2019.

[11] O. P. Verma y H. Gupta, «Fuzzy Logic Based Water Bath Temperature Control System,» *IJARCSSE,* vol. 2, nº 4, pp. 334-335, 04 2012.

[12] R. León, «Diseño, programación e implementación de un sistema de control difuso para el proceso de producción de hidrogeno a partir de vapor sobrecalentador de agua,» Escuela Politécnica Nacional, Quito, 2014.

[13] S. Ramin, I. Wan y I. Wan, «Control system design for a field survey agricultural mobile robot,» *International Conference on Agricultural and Food Engineering for Life (Cafei 2012),* 2012.

[14] Matlab, «MathWorks,» 2019. [En línea]. Available: https://www.mathworks.com/products/fuzzy-logic.html?s_tid=srchtitle.

[15] D.Guzmán y V.M.Cataño, «La lógica difusa en ingeniería: Principios, aplicaciones y futuro.,» *Ciencia y Tecnología,* nº 87-107, pp. 88-106, 2006.

[16] D. Canete, J.F.Saz-Orozco, P. d. Moreno-Boza y D. Duran-Venegas, «Object-oriented modelinh and simulation of the closed loop cardiovascular system by using SIMSCAPE,» *Computers in Biology and Medicine,* vol. 43, nº 4, 2013.

[17] Matlab, «MathWorks,» 2019. [En línea]. Available: https://www.mathworks.com/help/simulink/slref/saturation.html?searchHighlight=saturation&s_tid=doc_srchtitle.

[18] MATLAB, «MathWorks,» 2019. [En línea]. Available: http://www.mathworks.com/.

[19] MATLAB, «MathWorks,» 2019. [En línea]. Available: https://la.mathworks.com/discovery/pid-control.html?fbclid=IwAR0fRvjFz1z98RclWJTBEV0AtbUxWdmlA1z_IHML82yhgCWxKtT648msKmg.

[20] S. W. Sung, J. Lee y I.-B. Lee., «Proportional-Integral-Derivative Control,» de *Process Identification and PID Control,* Singapore, John Wiley & Sons, 2009, pp. 111-147.

[21] W. Sánchez, Á. Mullo y P. Velasco., *Autosintonía PID de controladores industriales,* Caracas: Universidad Central de Venezuela, 2019.

[22] S. I. O. Support, Process Data Acquisition and Monitoring, 2017.

[23] Bishop, R. C. y D. -. R. J., Modelos matemáticos de los sistemas, Sistemas de control moderno, Madrir: PEARSON EDUCATION S.A., 2007, pp. 52-62.

[24] R. Alanen, Auto-tuning Control Based on Åström's y Hägglund's theory, University of Oulu, 1993.

[25] S. Levy, S. Korotkin, K. Hadad, A. Ellenbogdn, M. Arad y Y. & Kadmon, «PID Autotuning using feedback,» *IEEE 27th Convention Electrical and Electronics Engineers in Israel,* pp. 1-3, 2012.

[26] K. Ogata, de *Ingeniería de control Moderna*, Madrid, Pearson Education S.A., 2003, pp. 683-690.

[27] F. A. Fernandes, G. S. Sestillo, A. L. Dias, D. Brandão y P. Ferrari., «Influence of network parameters on the recovery time of a ring topology PROFINET network,» *IFAC Conference Paper Archive,* vol. 49, nº 30, pp. 278-283, 2016.

[28] S. I. O. Support, «PID Control with PID_ Compact,» 2014, pp. 2-50.

[29] M. Rasmussen y K. Laumann, «Potential use of HMI evaluation methods in HRA,» *PROCEDIA MANUFACTURING,* pp. 1359-1363, 2015.

[30] S. McGrady, «Introduction,» de *Designing Scada Application Software*, Canadá, Elsiever, 2013, pp. 1-10.