POLITECNICO DI TORINO

Master's Degree in Mechatronic Engineering

Master's Degree Thesis

Controlling of a Pneumatic Positioning System by means of a PLC

Supervisors

Prof. LUIGI MAZZA

Ing. GIUSEPPE PEPE

Candidate

DANIEL SERNA LARA

2020/2021

Abstract

Servo pneumatic systems are widely used to obtain a clean, accurate, robust position control in many industrial drives. This technology has the potential to replace electromechanical and hydraulic drives in many applications. The high power-to-weight ratio and low cost of pneumatic actuators make them attractive for a pneumatic cylinder's precision positioning. Still, it is not easy to achieve fast and accurate position control.

This work presents the position control of a single-rod, double-acting cylinder implemented in a PLC with a Camozzi serial node responsible for managing inputs and outputs. A didactic bench provided by the Department of Mechanical and Aerospace Engineering (DIMEAS) is used because it is equipped with different components that allow performing the tests using various schemes, types of valves, and control algorithms.

After describing the didactic test bench components, their main features, and the correct configuration, a full chapter describes the PLC Programming through the software Tia Portal. Finally, a comparison is made between the test results using different pneumatic valves and various techniques to control a pneumatic actuator position. Details of each control method and the results obtained are given, and finally, a global comparison is made with several critical considerations.

The experiments realized on the didactic bench exposed during the thesis will help students for academic exercises during fluid automation and industrial automation courses.

Keywords: Pneumatic systems, Controlling positioning systems, PLC, ladder logic, feedback control.

Acknowledgements

Primero que todo, Sofía, Nubia y Didier, gracias por ese amor y apoyo incondicional, gracias por saber dejar volar la cometa cada vez más alto, gracias por no dejarme caer. Una vez más, orgulloso de ustedes puedo decir que hemos cumplido otro sueño juntos, gracias, gracias y mil gracias.

Quiero agradecer también a las personas que con el pasar de los días se convirtieron en familia, en especial, Hannah, Balles, Chalie, gracias por permitirme compartir esta travesía con ustedes y hacerme sentir como si nunca hubiera salido de casa.

I also want to thank all the people Turin gave me, Ale, Giba, Marco, Beto, Manolo and my all Colombia Calcio Turin teammates, meus irmãos da Ro7 e a todos os brasileiros que conheci e me tornaram parte de sua linda família. Thanks for every single moment. None of this would have been possible without your support. I want to thank all of my friends from Javeriana Cali and Politecnico di Torino, who have lived, suffered and laughed by my side throughout these seven years.

I would also like to show my appreciation and respect for professor Luigi Mazza and Engineer Giuseppe Pepe, the mentors that guided me in this thesis work.

Unos renglones no alcanzan para expresar lo agradecido que estoy con todos los que me ayudaron a cumplir este sueño e hicieron parte de este increíble proceso, donde no solo obtuve el título de *"Dottore Magistrale"* sino que también aprendí a ser persona, a valorar cada momento y a hacer las cosas con amor.

Daniel Fernando Serna Lara

Table of Contents

List of Tables

List of Figures

Acronyms

PWM

Pulse Width Modulation

PLC

Programmable Logic Computer

SB

Signal Board

SM

Signal Module

Chapter 1

Introduction

1.1 Pneumatic Systems

For a long time, pneumatic technology has played an important role in mechanical work, and also it has been used in the improvement of automation solutions in almost all industries.

The pneumatic systems transform the pressure and volume of compressed air into rectilinear and rotary movements through actuators such as cylinders and motors. The air compressor absorbs air from the atmosphere and stores it in a high-pressure tank called a receiver. Therefore it is supplied to the system through tubes and valves [1].

Many fields that require the automation of a process are applying pneumatic systems since this technology offers numerous advantages as follow [2][3]:

- Economical: the source is abundant since it uses air as a source of energy, and it is inexhaustible and free. Also, the components of a pneumatic system are not expensive and are durable. Likewise, the maintenance cost is lower than other methods.
- Easy to transport and store: the compressed air can is transported long distances in large volumes through pipes. Once used, there is no need for return ducts because the air is released into the atmosphere without a previous process.
- Resistant to temperature changes: the temperature variation alters in a lower degree the compressed air. Thus, it can work for a long time and with a wide working temperature range without the risk of reaching a dangerously high temperature.

- Safety aspects: It is flameproof so that it can be operated in a chemically aggressive, humid, and dusty environment without the risk of fire or explosion.

- Cleanliness aspects: A production with pneumatic technology does not produce pollutants. Its cleanliness feature is highly recommended for industries that required immaculate presentation like chemical, textile, and food industries.

- Easy regulation of force and speed: Both the force and speed are adjustable in a continuous way. The regulation is achieved through a pressure regulator without adding significant changes to the system. Pneumatics are useful for efforts that require precision and speed.

1.1.1 Servo Pneumatic Systems

A servomechanism is a closed-loop mechanism that provides feedback to correct the position of the actuator. This actuator must provide negative feedback through a sensor, and it allows the actuator to automatically correct the position error and with great precision. A pneumatic servo system combines a pneumatic actuator or cylinder and a control and feedback system [2].

Pneumatic systems have two main parts: the power and control subsystem (Figure 1.1).

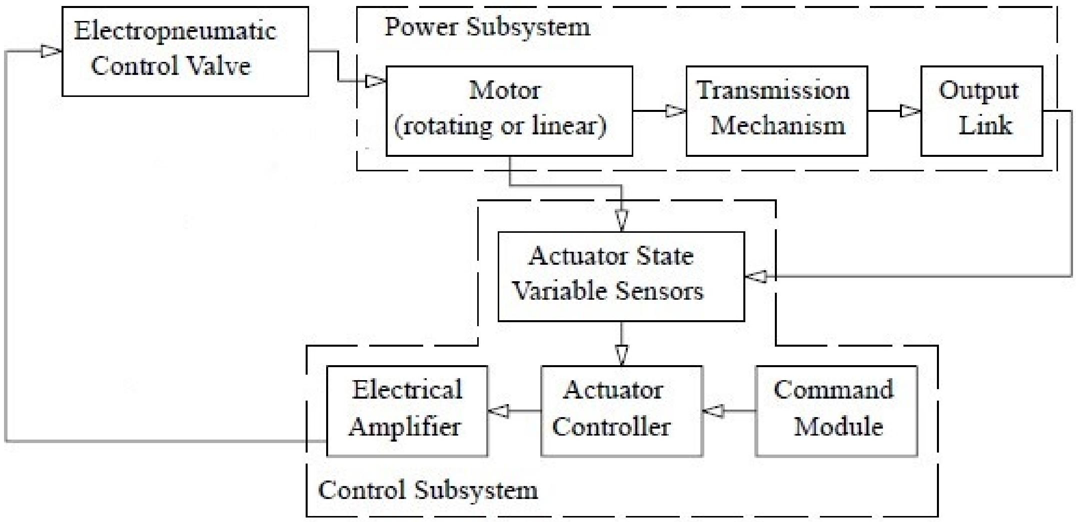


Figure 1.1: Block diagram of a pneumatic actuating system [1].

The motor is the most crucial part of the power subsystem as it converts pneumatic power into functional mechanical movement. This movement can be linear or rotary. In this thesis, a double-action pneumatic piston will be used, which performs a rectilinear movement in two directions, forward or backward. Usually, the output motor shaft is the actuator output link. It means that the pneumatic actuator has a direct drive structure [1].

All servomechanisms need a controller and an instrument to measure the position, speed, torque, and force for subsequent correction. The actuator state variable sensors take the input data to the control subsystem. Usually, the pressure, moment, displacement force, speed, and acceleration can be measured in the actuator. Linear-position sensors are used as feedback elements for motion control in a pneumatic drive system. Linear-variable-differential transformers (LVDTs), magnetostrictive sensors, precision linear potentiometers, and digital optical or magnetic encoders are examples of this sensor type [1].

Another important part of the control subsystem is the command module. This saves the input information and selects it by input commands. Desired positioning points, trajectory tracking, velocity, or force value are examples of the information that can be saved in the command module [1].

The main element of the control subsystem is the controller. It processes, compares, and provides the system's control functions, depending on the control algorithm applied. The controller can be of two types: analog and digital. Most of the industry controls are digital, and the most common algorithm is the PID controller (proportional + integral + derivative). After, the output signal coming from the controller has to pass through an electric amplifier before arriving at the electro-pneumatic valve. This device conditions the output signal to have a correct input signal for the electro-pneumatic valves [1].

The control valve is the interface between the power and the control subsystems in a pneumatic system. The valve is a crucial element where a signal with small amplitude and low power provides high response modulation in pneumatic power. Generally, there are three types of electro-pneumatic control valves (proportional, servo, and solenoid) used in the pneumatic actuator[5].

There is no consistent difference between servo and proportional valves. These valves have the same work's principle; they are spool-type precision valves in which the spool can be moved in proportion to an electrical control signal. However, servo valves generally have better dynamic characteristics than proportional valves and provide a higher degree of closed-loop control, making them more expensive.

On the other hand, proportional valves move the spool in proportion to a control signal but do not have automatic error correction within the valve. A proportional valve competes with servo valves in most characteristics except for dynamic characteristics [1][5].

Solenoid valves use a solenoid to control valve actuation. These valves have high reliability and compact size. The solenoid valve is an effective solution for repeated stops in two positions because it can work with an off/off algorithm control. The use of an on/off solenoid valve with the PWM (Pulse Width Modulation) control method achieves an equivalent performance in flow or pressure control operation. In this case, it is possible to replace the solenoid valve instead of the expensive servo or proportional valve [1][5].

Initially, pneumatic systems offer a cheap and straightforward solution for mainly point-to-point actions. An electromechanical system is a most accurate and more expensive alternative to control position, force-torque, and accuracy. Pneumatic servo systems are in the middle of these two extremes. A pneumatic servo system offers a high degree of control without being as expensive or complicated as an electromechanical system would be [2][4].

Servo Pneumatic control systems have three main applications[1][2]:

- **Pneumatic Positioning system:** The application of pneumatic systems in position control is highly used in robots and manipulators, welding and riveting machines, pick-and-place devices, vehicles.

- **Pneumatic Systems for Velocity Control:** The application fields of servo pneumatic actuators with speed motion control include arc welding machines, painting and printing equipment, scanning motion systems in inspection devices, cutting machines for plastic, wood, and fabric materials.

- **Pneumatic Systems for Force Control:** Servo pneumatic actuators with force control are applied in fields like dynamic and static material test systems, spot-welding equipment, vehicle suspensions, manipulator grippers, physiotherapy and assembly robots, paint spraying systems.

The servo pneumatic technique has the potential to replace less efficient electromechanical and costly hydraulic actuators in many applications.

1.1.2 Pneumatic Positioning Systems

Position Systems are fundamental and very attractive components in industries because they are cheap, clean, present an excellent force-weight ratio, and easy to assemble. Primarily, the applications that require high positioning accuracy used electromagnetic motors. However, nowadays, the servo pneumatic technique has the potential to replace less efficient electromechanical and costly hydraulic actuators in many applications[2].

Pneumatic positioning actuators are usually divided into two groups depends on the type of control:

- **Open-loop control:** Ordinarily, the pneumatic system that uses this control method has mechanical stops. Figure 1.2 shows a pneumatic actuator diagram with two adjustable hard stops; this feature allows the system to stop in two different positioning points. The proximity sensors indicate the position of the piston. When the piston approaches enough, the magnetic field closes the switch, closing an electrical circuit and producing an electrical signal, then is sent as the control system's input[1].

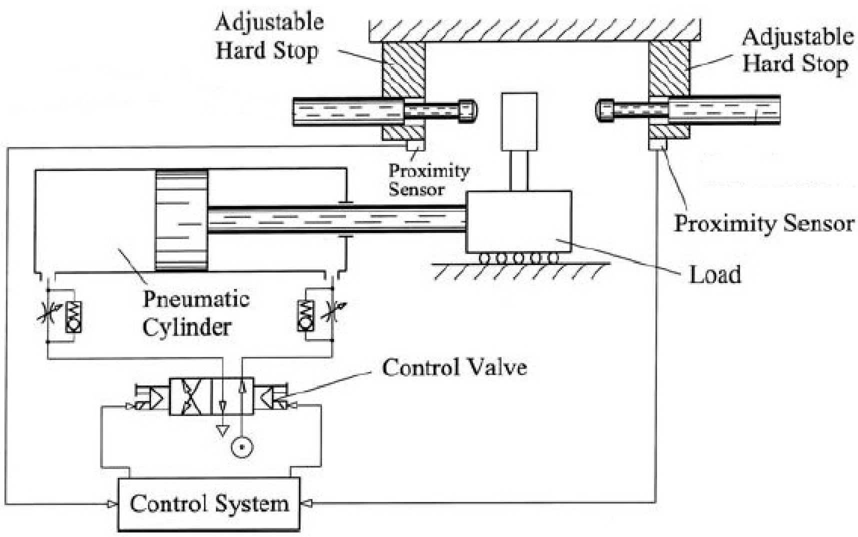


Figure 1.2: Schematic diagram of a open-loop pneumatic positioning actuator with two adjustable hard stops [1].

- **Closed-loop control:** a closed-loop pneumatic positioning system holds a transducer that measures and converts the actuator's output signal to an electrical signal. The controller compares the feedback signal coming from

the transducer and the command signal (set-point). The resulting signal is applied to reach the desired positioning or tracking of the movement [1].
Figure 1.3 shows the schematic of a positioning system with a servo or proportional valve. The system contains a pneumatic cylinder, a positioning transducer (sensor), a control valve, and an electronic control system. When the piston approaches the set-point, the valve moves to over-center to regulate the flow rate and volume into and out of the actuator, creating a pressure that opposes piston movement [1].

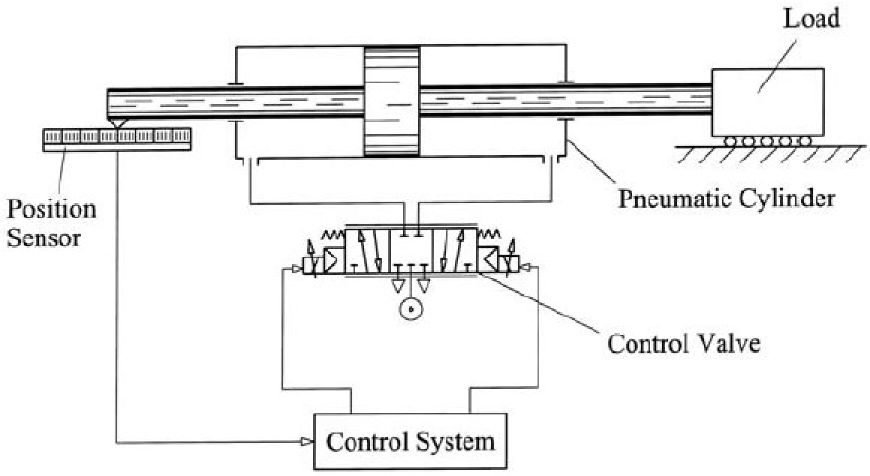


Figure 1.3: Schematic diagram of a closed-loop positioning system with servo or proportional control valve [1].

1.2 Company Overview - *Camozzi Automation S.p.A*

As all the test bench components in which the experiments will be carried out are provided from the same company, a brief description will be made about Camozzi S.p.A.

Founded in 1964 by Attilio Camozzi, Camozzi Automation currently present with 14 production plants in Europe and more than 50 exclusive distributors worldwide, is an Italian global leader in designing and producing components for motion and fluid control. The company, with more than 1900 employees, develops technology solutions principally in three fields[6].

- **Industrial Automation:** Packaging, food & beverage, automotive, woodworking machinery, assembly & robotics, and printing & paper.
- **Life Science**: Analytical devices such as Biomedical Analyzers, Environmental Analyzers, and Molecular Analysis) and Medical Devices such as dialysis devices, pressotherapy and vacuum therapy, oxygen concentrators, and clinical diagnostics).
- **Transportation**: Commercial vehicles, passenger cars, railway, and Off-Road Vehicles.

1.3 Problem Statement

The main objective of this thesis article is to control the position of a pneumatic cylinder actuator using different types of valves and control algorithms. The data collected is subsequently analyzed to define what type of valve and control have the best response.

Three different type of valves will be used to perform the exercise: the first scheme uses a 5/3 closed center digital valve, the second one uses four directly operated proportional valves, and the last one uses smart proportional valves either in MASTER-SLAVE or SLAVE-SLAVE configurations.

The most important features of a servo pneumatic positioning system are accuracy and speed. So, in order to analyze the behavior of the system, two types of test are executed [2][3]:

- The point-to-point changeover test consists of making a step-change in the reference signal from some known points within the cylinder's stroke length.
- The setpint tracking or trajectory tracking test consists of making the reference signal to follow a defined trajectory. Usually, sinusoidal and ramps signals are used to execute this test.

Chapter 2

Literature Review

As mentioned in the previous section, servo pneumatic position systems have a massive range of applications in automated systems and industries. Figure 2.1 shows the main research areas in servo pneumatic positioning systems. This section describes, for the most part, the current knowledge about controlling servo pneumatic positioning systems through the analysis of related published work and also a brief explanation of research in the other areas to understand how important and relevant this phenomenon is today.

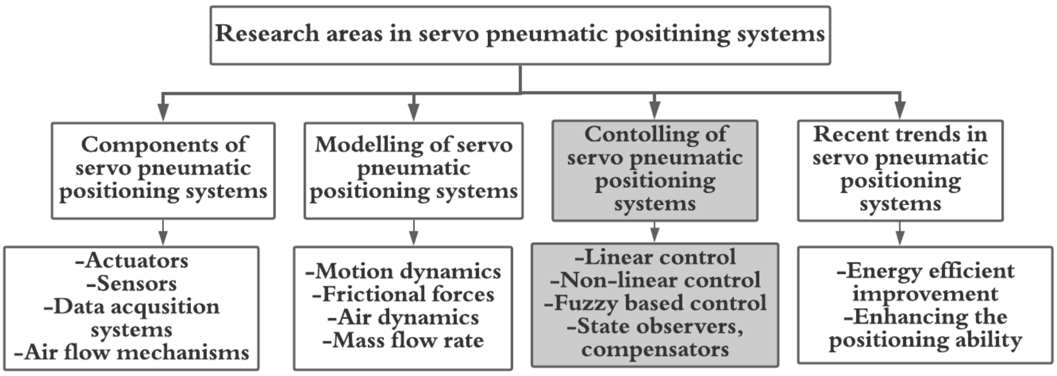


Figure 2.1: Research areas in servo pneumatic positioning system [2].

2.1 Controlling of servo pneumatic positioning systems

In the past few decades, the controlling of servo pneumatic systems has been an exciting research topic. Figure 2.2, shows the general structure of closed-loop feedback control. Naturally, the error signal ($e = x_{ref}$-x) feds the controller, and the valves receive the control signal u from the controller depends on the algorithm used [2].

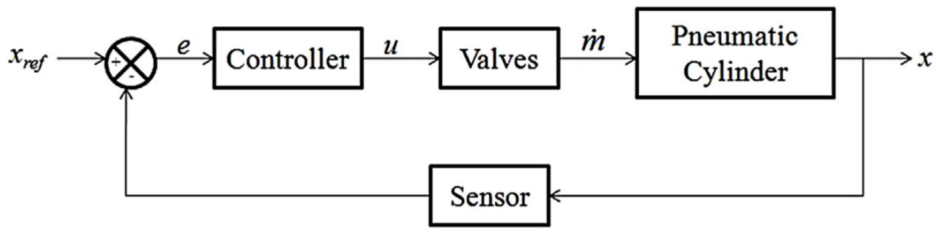


Figure 2.2: Structure of closed-loop feedback positioning system of pneumatic cylinder [2].

Due to the nonlinear nature of the pneumatic systems, different control algorithms have been studied over time. With this in mind, this literature review is divided into four categories according to the types of the most common control algorithm used in servo pneumatic positioning control systems: Linear control, Nonlinear control, Fuzzy based control, and State observers and compensators.

2.1.1 Linear Control of servo pneumatic systems:

The PID control is widely used for the position of pneumatic actuators. Wang et al. [7] propose an accurate position control strategy for servo pneumatic systems applied to food packaging. Wang implements a PID controller combined with an algorithm for time-delay minimization and target position compensation to be applied to a pusher mechanism in the packaging of confectionery products.

The control algorithms to achieve the minimization of time-delay and positioning accuracy uses the difference between the desired set-point and the current position as an amendment, which is then weighted and added to the current position in the next cycle of movement. Moreover, the point to highlight is the system's learning process. It determines a function $C(t, e)$, which is a time-varying coefficient that depends on the history of the position error. The function helps correct the difference between the set-point and the current position, making the error smaller in each duty cycle iteration. Figure 2.3, shows the learning process of the system.

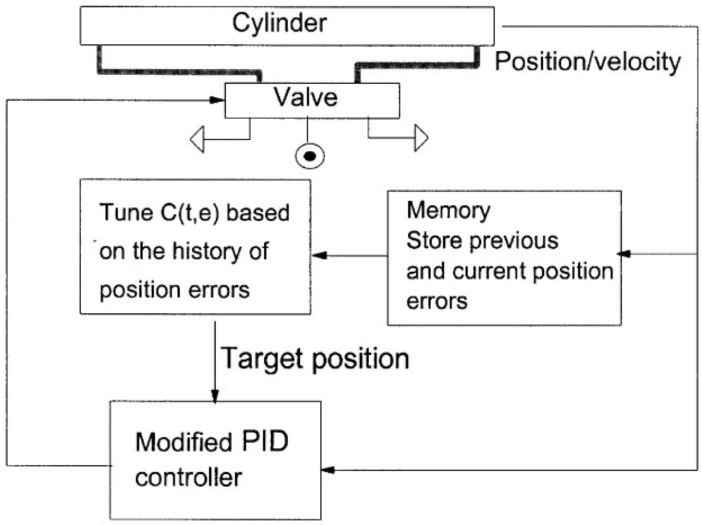


Figure 2.3: Learning process of the system [7].

The results show that the system is able to reach the set-point in an interval between 250 ms and 254 ms, with an accuracy of ± 1mm. Since the expected time to reach the desired position is 252 ms, Wang et al. conclude both the positioning and the time accuracy required by the production task are achieved using such a control strategy.

Chillari et al. [8] compared the performance of six different control algorithms: PID, Fuzzy, PID with pressure feedback, Fuzzy with pressure feedback, Sliding mode, and Neuro-Fuzzy control for a servo pneumatic system that consists of a cylinder, two or four ON-OFF or proportional valves, a displacement sensor, two pressure sensors, a controller, and a user interface to control and monitor the actuator.

The authors experimented with six different trajectories for each control algorithm in order to have a reliable comparison. The trajectories are three sinusoidal waves with the same amplitude and different frequencies, a square wave, a saw-tooth wave, and a staircase signal. These trajectories try to simulate many different types of conditions that can be experienced in real-world applications.

Regardless of the remarkable improvement both the PID and the Fuzzy control have with the pressure feedback, to avoid the increase in the global cost due to the introduction of a new sensor, Chillari et al. implemented a multilayer perceptron neural network as an estimation of the pressure signal.This estimation is based on the model equation. The result is that the pressure difference is a static function of piston position, velocity, and acceleration. Therefore, a position function and two past values should have a good approximation. The neural model was trained using previous measurements obtained by the real pressure sensor.

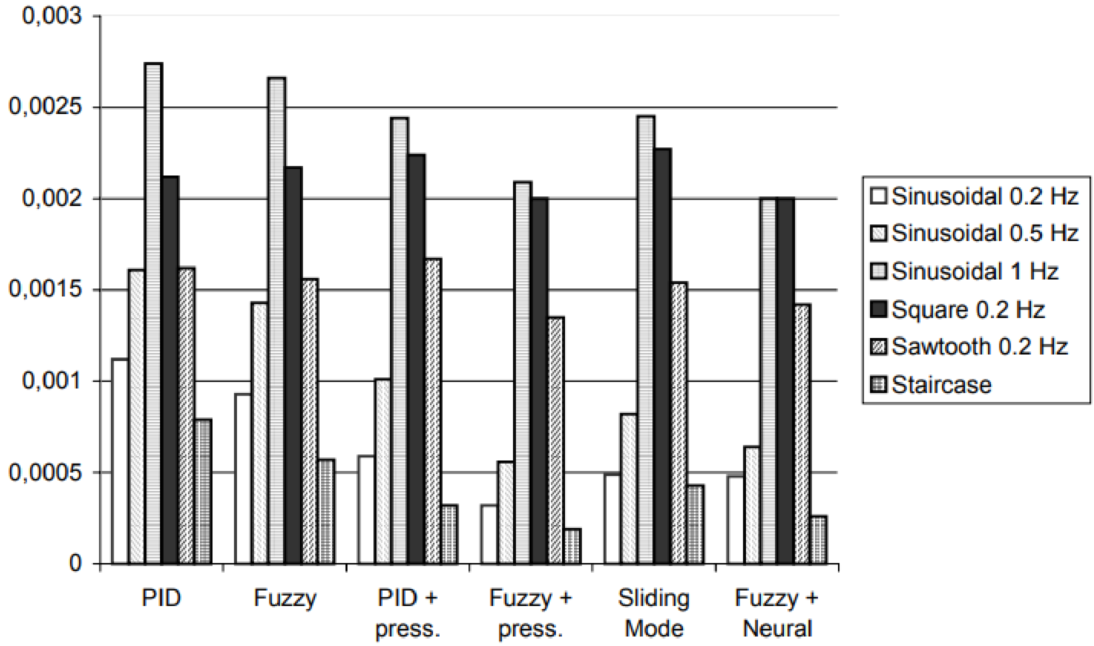


Figure 2.4: Comparison between all controls and references[8].

Figure 2.4 shows the comparison between the different types of controllers. Chillari et al. conclude not only the Fuzzy control betters the simple PID both with and without the pressure sensor's choice but also that the adoption of the neural network makes it possible to avoid using additional sensors, reducing the global performance insignificantly.

In the next article, the author approaches a linear double-acting pneumatic cylinder positioning differently from those already studied. In most studies on this topic, the authors use proportional valves to control air mass flow through the pneumatic cylinder's rear and front chambers. Almeida and Simiao [9] adopt proportional pressure regulator valves to control the air pressure and not the air mass flow.

Furthermore, the system works in a double-loop mode thanks to the linear resistive transducer inside the pneumatic cylinder that sends the output pressure signal to the controller and the internal pressure transmitter, which gives the input signal to an on-board PI controller that each valve has. This double-loop mode helps to better monitor and control the input and output signals of the process.

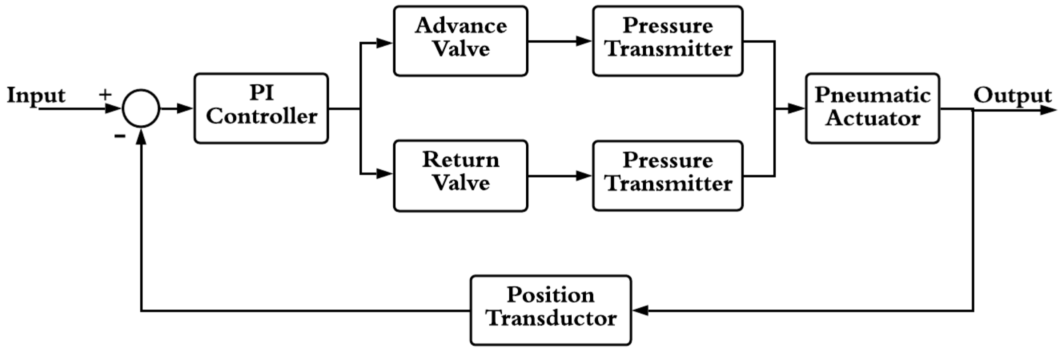


Figure 2.5: Simplified block diagram of the system [9]

Figure 2.5 shows a simplified system's diagram, where both controllers were tuned with the same parameters, so the block diagram is simplified with only one PI Controller to enable better understanding. Finally, the experimental data returned a maximum position error of 4,48mm, which was considered satisfactory for their applications.

2.1.2 Nonlinear control of servo pneumatic systems

Song and Liu [10] presented a CARIMA model-referenced adaptive GPC (Generalized Predictive Control) for a pneumatic system driven by PWM-controlled solenoid valves.

CARIMA (controlled auto-regression and integrated moving average) model is a statistical representation that uses variations and regression of statistical data to find patterns for predicting the future. It is a dynamic time series model, which means that past data explain future estimates rather than independent variables [11].

Moreover, the authors use the GPC based on a CARIMA model proposed by Clarke et al. [11]. GPC uses long-range predictive control, so it is more robust against plant dynamics, time-delay, and dead zone than conventional methods. However, some factors such as plant parameter variations and external disturbances may affect the performance of GPC. The recursive least-squares method (RLS) can be applied for on-line parameter estimation to adapt to time-varying model parameters and limit the influence of the past data.

The authors reached for a step reference input, a maximum and average absolute steady-state error of 0.25mm and 0.05mm, respectively. For a sinusoidal input, the maximum and average absolute dynamic tracking errors are 3.27mm and 0.76mm, respectively. Finally, facts have proved that RLS-based adaptive GPC can effectively adapt to time delays and plant dynamics and provide impressive results for steady-state and dynamic tracking, so adaptive GPC is a perfect solution for pneumatics systems that require precise control.

Shen et al. [12] develop on their paper a control methodology that allows nonlinear model-based sliding mode control of PWM-controlled servo pneumatic systems, capable of obtaining a robust control via relatively low-cost on-off solenoid valves.

In this practice, the authors first elaborate an average model-based PWM-control of nonlinear systems. This methodology explains the equivalent continuous-time dynamics of a PWM controlled nonlinear system, transforming this discontinuous system into an equivalent system in control canonical nonlinear form, adapted to many nonlinear control strategies.Having expressed the PWM system dynamics in a continuous input canonical form,a sliding mode control approach can be applied to the system's control.

Shen et al. conclude that despite tracking performance are corrupted at high frequencies due to the limited switching response time of the valves, the proposed control provides effective control

2.1.3 Fuzzy based control of servo pneumatic systems

Chen et al. [13] applied a fuzzy model-following control algorithm to a pneumatic servo position system subjected to an external load. Especially for systems in which their mathematical formulation is difficult to obtain, fuzzy logic controllers can perform better than conventional controllers.

In general, the design of the fuzzy logic controller is based on the characteristics of the system to be controlled. Although the conventional PID controller is popular in industrial applications, it is not practical to deal with nonlinear or time-variant systems. On the other hand, the fuzzy control system's resulting performance cannot be tuned very efficiently. To overcome this problem, the authors build a fuzzy controller that is composed of PID-like rules.

In order to have a fuzzy adaptive controller, the defuzzification process is selected such that the output of the "important" rule has a more decisive contribution to the control signal compared to the other rules, and the resulting control signal can be regarded as a PID-like controller with variable parameters at each sampling cycle.

Furthermore, it is difficult for conventional fuzzy control systems to predict the final performance of the controlled system. A fuzzy adaptive controller is introduced into the model-following control system to have the advantages of a fuzzy logic controller with expected performance. Figure 2.6 shows the structure of the fuzzy model-following control system, where U_m denotes the reference step input and x_m is the output trajectory of the model.

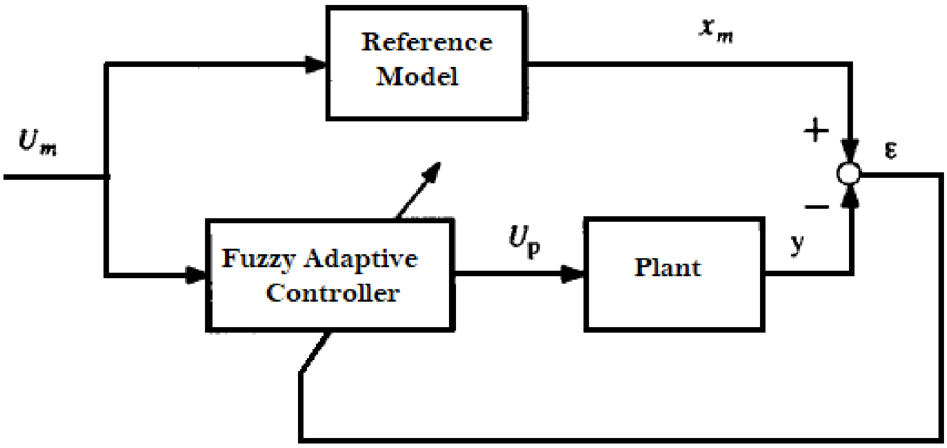


Figure 2.6: Structure of the Fuzzy model-following control system [13]

Due to the servo valve's hysteresis, the response of the plant is much slower than the reference's model output. By adding an extra fuzzy control loop, the asymmetric response of movement in different directions is improved. Chen et al. conclude that the proposed controller can properly control nonlinear systems and is insensitive to perturbances, external loads, and system parameter changes.

2.1.4 State observers and compensators in servo pneumatic systems

The system's state observer is used to estimate specific internal states (such as the velocity, acceleration, and pressure of the cylinder chamber) based on the displacement signal [2].

The compensator is a correction algorithm that uses the estimated values to serve the main controller to take corrective measures. Usually, friction, speed, and pressure compensators are used with the position feedback control [2].

The structure of the control system, including observers and compensators, is shown in Figure 2.7 .

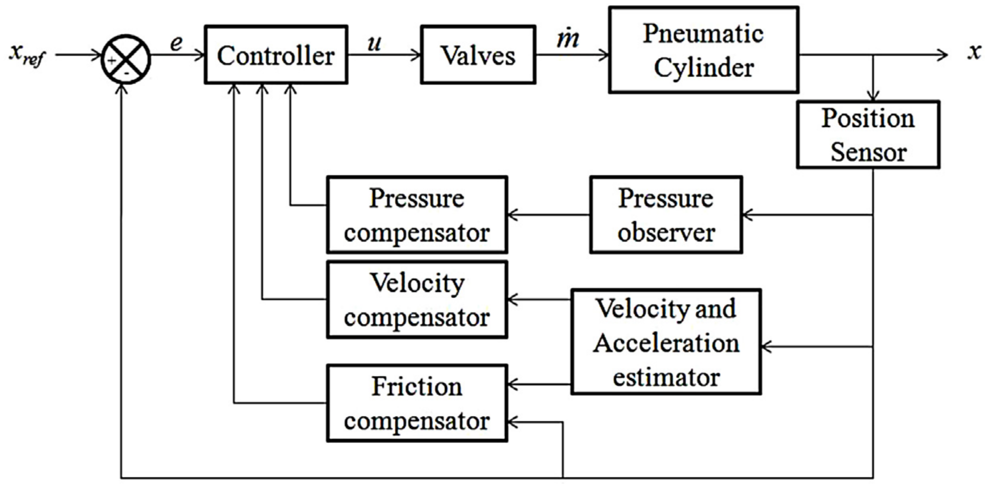


Figure 2.7: Structure of position control system with observers and compensators [2]

Song and Liu [10] implemented a proportional velocity acceleration (PVA) control with friction compensation to reduce the steady-state error for improving a pneumatic actuator's performance using two low-cost on/off solenoid valves driven with pulse-width modulation (PWM).

After studying the behavior of the system, the authors found that the friction force, which causes the steady-state error, results in the dead zone. Because of this, an effective approach of friction compensation is introduced. Figure 2.8 shows the diagram of the control system using PVA with friction compensation. The principle of operation of the friction compensation is to add an offset to the output of the PVA controller (u') so that the piston will continue moving in the dead zone.

Then, Song and Liu obtained that the value of the maximum absolute steady-state error is 0.28mm and the average absolute steady-state error is 0.07mm.

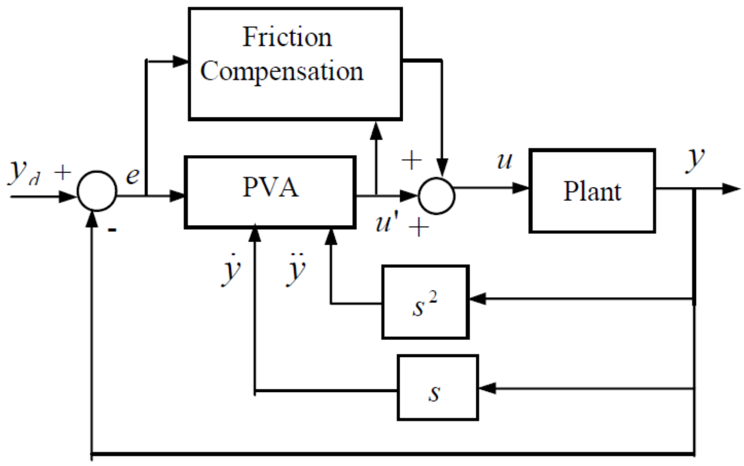


Figure 2.8: PVA control with friction compensation [10]

2.2 Recent trends in servo pneumatics positioning systems

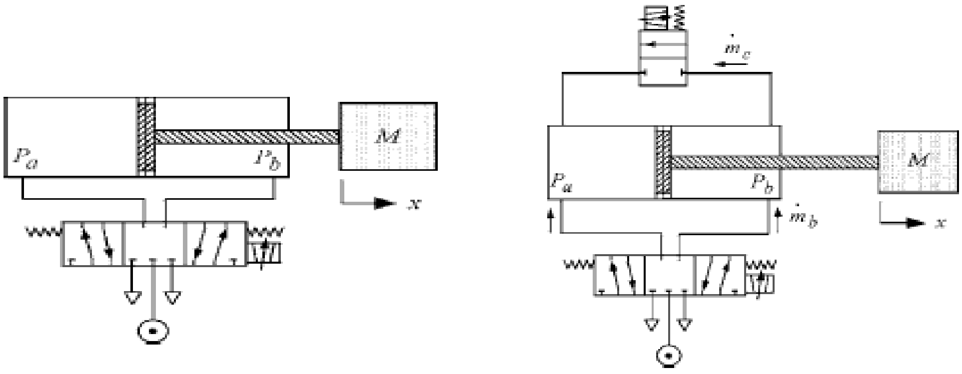
2.2.1 Energy efficiency improvement methods

Pneumatic systems have the advantage of better energy efficiency than electrical or mechanical systems. In order to further improve the efficiency of the system, a lot of research work is ongoing.

Shen and Goldfarb [14] proposed a structure and control method for the energy-saving servo control of a pneumatic servo system by allowing crossflow between the cylinder chambers. The flow between the cylinder chambers is achieved by adding an additional two-way valve (Fig 2.9b) to a typical pneumatic servo system that consists of a proportionally controllable four-way spool valve (Fig 2.9a).

Considering pneumatic actuators usually contain a large amount of stored energy due to their compressibility. Since this stored energy can theoretically be moved from one part of the actuator to another in a controlled manner through a valve, the pneumatic system provides a unique possibility and potential for energy-saving control and efficiency.

A control method is developed to enhance the mass flow rate required by the sliding mode controller with the recirculated mass flow provided by the crossflow valve to minimize compressed air consumption. Based on this model, by tracking a given trajectory, the mass flow rate into or out of the pneumatic cylinder's respective sides can be commanded. The crossflow valve is activated and provides



(a) Configuration of a typical pneumatic servo system **(b)** Modification of a typical pneumatic servo system to include an interchamber flow path.

Figure 2.9: Pneumatic servo system with and without interchamber flow plath [14]

the required mass flow rate when the pressure in the chamber being depressurized is greater than the pressure in the chamber being pressurized.

Shen and Goldfarb obtain results that prove reduced energy consumption by 25-52 % relative to the configuration showed in Fig 2.9a, without sacrificing the tracking performance.

2.2.2 Enhancing the positioning ability of pneumatic actuator

The utilization of miniature products in our daily activities is continuously increasing. In particular, with the advent of nanotechnology, the consumption of electronic products has increased exponentially. Semiconductors and flat panel display industries are examples of industries with massive demand in sensing and actuating technologies to such a small scale.

Chiang [15] proposed a novel pneumatic-piezoelectric hybrid actuator, shown in Fig 2.10. The pneumatic servo cylinder is used for high-speed and large-stroke positioning, and the piezoelectric actuator is used in fine stroke positioning.

Piezoelectric actuators have the advantages of high positioning accuracy and fast response and have been applied in many different fields, especially in high-precision positioning control. Nevertheless, the maximum stroke of the piezoelectric actuator can only reach the micrometer range. Although piezoelectric actuators can perform excellent response at nanometer positions, the stroke is limited to a range

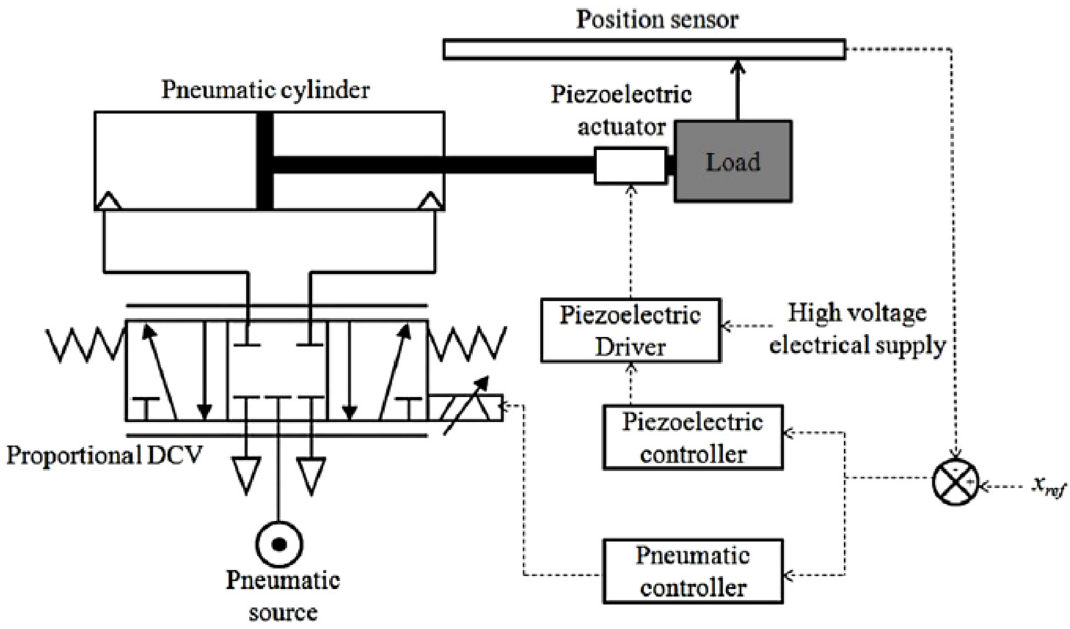


Figure 2.10: Schematic diagram of hybrid pneumatic-piezoelectric actuator[15]

of about 100 μ m. Therefore, in terms of positioning accuracy in the nanometer range and working range in centimeters, they cannot meet the positioning system's requirements.On the other hand, pneumatic non-linearities, such as friction, compressibility, make pneumatic servo control much more complicated than motor servo control and electro-hydraulic servo control, thus limiting the application of high positioning accuracy.

To meet these two goals, the author an intelligent X-Y dual-axial servo pneumatic-piezoelectric hybrid actuator for position control with dual-axes, high response, long stroke (250 mm), and nanometer accuracy (20 nm). Moreover, each axis comprises a servo pneumatic system, composed of a rodless pneumatic actuator and a proportional servo valve, and a piezoelectric actuator mounted in order to achieve nanometer positioning accuracy cascade on the pneumatic's actuator piston. Furthermore, a controller based on a fuzzy sliding mode controller with a self-organizing modifier and decoupling ability is implemented to control the system.

Chapter 3

Description of the test bench

This chapter is responsible for describing the test bench, such as the PLC S7-1200, the pneumatic cylinder and the control valves.

3.1 Programmable Logic Computer (PLC)

Programmable automation technology is an incredible tool used by engineers and technicians to improve their manufacturing systems in related industries. The expression programmable automation technology covers the three trends of programming. Programming methods can be divided into three trends: computer numerical control technology, robotics technology, and programmable logic control.

The International Electrotechnical Commission (IEC) defines a Programmable Logic Computer (PLC) as [19]:

"A digitally operating electronic system, designed for use in an industrial environment, which uses a programmable memory for the internal storage of user-oriented instructions for implementing specific functions such as logic, sequencing, timing, counting, and arithmetic, to control, through digital or analog inputs and outputs, various types of machines or processes. Both the PC and its associated peripherals are designed so that they can be easily integrated into an industrial control system and easily used in all their intended functions"

PLC-based systems are mainly used in the field of industrial automation. The main application areas include control of industrial machines, material handling, product assembly, control of machine tools. Siemens, Allen Bradley, and Omron are among the major manufacturers of programmable logic controllers.

Fig 3.1 shows the conventional functions and components present in an application with PLC. The sensors are responsible for obtaining the system's current state and sending the information to the input module responsible for converting the information into an understandable type of information for the PLC control unit. If it is an analog signal, the input module must be an analog input module. If it is a digital input signal, it must be a digital input module. The input module sends the information to the central control unit that, based on the program previously charged into the PLC by the programmer, determines the next action. This action can enable an output that can activate an actuator (motor, cylinder, light) or make an internal operation changing memory status, updating counters, or enabling timers. Therefore, the system's state will change to a new state to continue with the process [20].

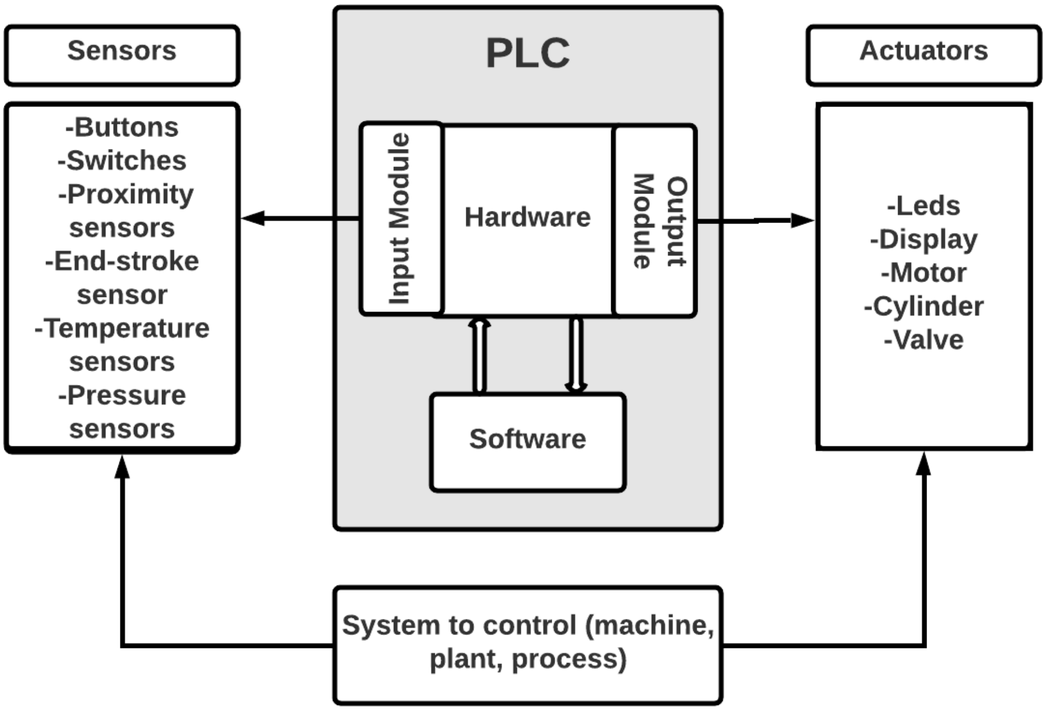


Figure 3.1: General components and functioning of a PLC system

Another fundamental aspect of the PLC is its programming language. The program can be written in different languages, which are understandable and accessible to the same operators who worked on the devices before the PLC. Three main languages are listed below:

- **LAD (ladder logic):** It is a graphic language that almost all PLCs can support. It is a graphical connection between Boolean type variables, similar to the old relay type controller, in which the energy flow is represented based on circuit diagrams. Therefore, this programming language is used for most Boolean signals and is not actually used for processing analog variables.

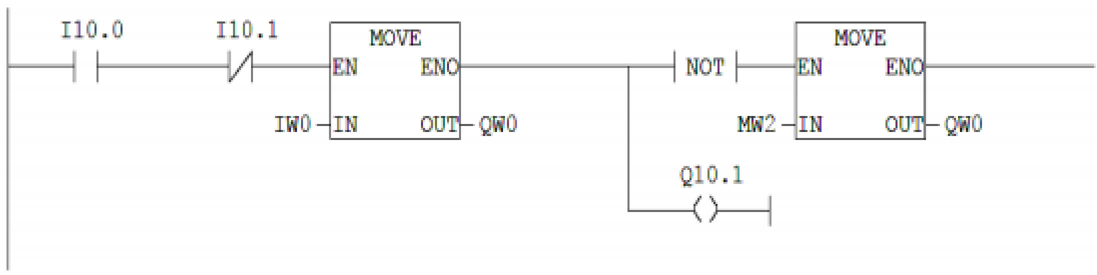


Figure 3.2: LAD programming technique

- **FBD (Function Block Diagram):** The second programming language is also graphical, is a programming language based on the graphical logic symbols used in Boolean algebra. FBD comes from the field of signal processing. It is very convenient to use when there is no cycle, but there are multiple branches in the program to be created. It is a high-level language that allows basic functions to be summarized in blocks so that users only worry about their routines' functional programming. In this way, it is ideal for users without advanced programming skills and low-complexity processes.

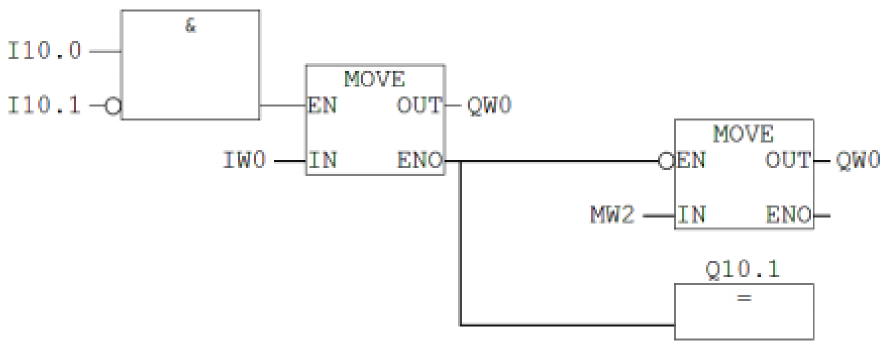


Figure 3.3: FBD programming technique

- **STL (Statement List):** This Is a text-based language that allows programmers to write programs by entering instruction mnemonics. It can help programmers create programs that other programming techniques cannot create because STL is a native program of PLC, and it works more or less like an assembler. The PLC follows the program's instructions from top to bottom and then restarts from the top. STL is a high-level programming language, so it is most suitable for experienced programmers.

```
      A     I     10.0
      AN    I     10.1
      JNB   _001
      L     IW     0
      T     QW     0
      SET
      SAVE
      CLR
_001: A     BR
      =     L      0.0
      AN    L      0.0
      JNB   _002
      L     MW     2
      T     QW     0
_002: NOP   0
      A     L      0.0
      BLD   102
      =     Q     10.1
```

Figure 3.4: STL programming technique

3.2 PLC Siemens S7-1200

The PLC used in this project work and implemented in the test bench is a Siemens S7-1200 with the CPU "CPU 1214 DC/DC/DC". The S7-1200 series contains various programmable logic controllers (PLCs) that can be used for many tasks in the automation field. The compact design, low cost, and extensive instruction set make the S7-1200 PLC ideal for controlling multiple applications [20].

The CPU combines a microprocessor, integrated power supply, input and output circuits, built-in PROFINET, high-speed motion control I/O, and onboard analog input in a compact enclosure to create a powerful controller. After downloading the program, the CPU contains the logic needed to monitor and control the application's devices. The CPU monitors the input and changes the output depending on the user's program. This program includes Boolean logic, counting, timing, complex mathematical operations, and communication with other smart devices.

The S7-1200 and its features are shown in Fig 3.5 and table 3.1, respectively. A signal board (SB) can be added in the front of the CPU.

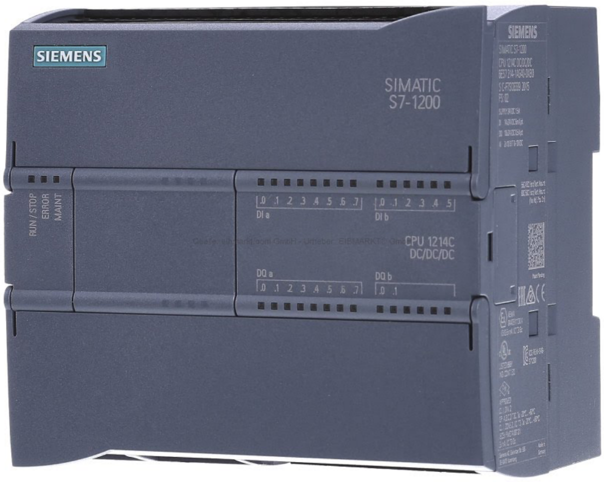


Figure 3.5: PLC Siemens S7-1200 [20]

Features PLC S7-1200	
CPU model	1214C DC/DC/DC
Digital Input (DI)	14 +(2 on SB)
Digital Output (DO)	10 +(2 on SB)
Analog Input (AI)	2 (0-10V DC)
Analog Output (AO)	1 on SB
Pulse Width Modulation (PWM)	2
Pulse Train Output (PTO)	2
High Speed Counter (HSC)	6
Closed loop controller (PID)	16
Power Supply	24 V DC

Table 3.1: Features PLC Siemens S7-1200

In addition to the CPU described, an Analog Output SM (Signal module) has been added to obtain the absent analog outputs necessary to control the PWM drivers of the Directly Operated Proportional Valves.

The analog signal module provides an input signal or the desired output value representing a voltage range or current range. These ranges are ±10 V or 0-20 mA, for the used module. The value returned by the module is an integer value, where 0 to 27648 represent the rated range of current, and -27648 to 27648 represent the range of voltage. Any value outside the range indicates overflow or underflow [20].The Analog Output SM and its features are shown below, respectively.

Figure 3.6: Analog Output SM

Analog Output SM features		
Model	6ES7232-4HB32-0XB0	
Analog Outputs (AO)	2	
Type	Voltage	±10 V
	Current	20 mA
Resolution	Voltage	14 bits
	Current	13 bits

Table 3.2: Features Analog Output SM

3.3 STEP 7 (TIA Portal) Programming software

STEP 7 (TIA Portal) presents a user-friendly environment to configure, program, test, and diagnose the logic required for control applications in all generations of basic, advanced, and distributed SIMATIC controllers, whether it is PLC-based or

PC-based controllers.

STEP 7 is the configuration and programming software component of TIA Portal. TIA Portal also includes WinCC, which is used to design and execute runtime systems Process visualization, including online help for WinCC and STEP 7. The term TIA Portal refers to the centralized configuration environment in which STEP 7 runs. TIA Portal is the English abbreviation of Totally Integrated Automation Portal and is also defined as an engineering platform for all fields related to automation. Moreover,its characteristic is that the user interface is the same for all areas of automation. It provides various shared services (such as configuration, communication, or diagnostics) and can be accessed by other software packages (such as SIMATIC WinCC V16, SINAMICS Startdrive o SIMATIC STEP 7 PLCSIM V16.

How was already explained, STEP 7 provides conventional programming languages for convenience and efficiency. It uses the three languages already described in the past section. They are briefly described again:

- LAD (ladder logic) is a graphical programming language. The representation is based on circuit diagrams.
- FBD (Function Block Diagram) is a programming language that is based on the graphical logic symbols used in Boolean algebra.
- SCL (structured control language) is a text-based, high-level programming language.

Furthermore, once the project is created, TIA portal gives a chance to the programmer to choose the programming language the program will be written for convenience and comfort.

3.3.1 Structuring the user program

When using TIA Portal, the instruction of the program for the automation task are inserted into code block, the blocks are divided in three categories and are explained as follow[20]:

1. **Organization Block (OB)** responds to specific CPU events and can interrupt the user program's execution. The default value of the cyclic execution user program (OB 1) provides the user program's basic structure and is the only code block required by the user program.There are several types of OB

blocks, for example:

- A "Startup" OB will execute one time when the operating mode of the PLC changes from STOP to RUN. After completion, the main "Program cycle" OB will begin implementing.
- A "Cyclic interrupt" OB allows you to start programs at periodic intervals, independently of cyclic program execution. The intervals can be defined in this dialog or in the properties of the OB. This OB is important because the PID block must be implemented in a Cyclic interrupt OB.
- A "Hardware interrupt" OB will interrupt cyclic program execution in reaction to a signal from a hardware event. The events must be defined in the properties of the configured hardware.
- A "Time error interrupt" OB will interrupt cyclic program execution if the maximum cycle time has been exceeded. The maximum cycle time is defined in the properties of the CPU.

2. **Function blocks (FB)** are code blocks whose values are permanently stored in the instance data block (DB), so the data is stored once the FB has finished. After the FB task is completed, the CPU returns to the code block called the FB. The instance data block retains the value of the FB instance. These values can be used for subsequent calls to function blocks in the same scan cycle or other scan cycles.

3. **Function (FC)** are code blocks or subroutines without dedicated memory, similar to the FB's. For example, use FC to perform standard and reusable operations (mathematical calculations) or technical functions (for example, individual controls that use bit logic operations). It is also possible to call an FC multiple times at different locations in the program. This reuse simplifies the programming of recurring tasks.

4. **Data blocks (DBs)** save program data. They, therefore, contain variable data which are used by the user program. Global data blocks acquire data that all other blocks can use. Their maximum size varies depending on the CPU, and the structure of the global DB's can be freely defined. As already discussed in the previous paragraphs, each functional block, each function, or organization block can read or write data from/to a global data block. A global data block and an instance data block can be open at the same time. Fig 3.7 shows how the DB global can be accessed and have access to all the blocks and the DB instance DB3 has access just for the FB3.

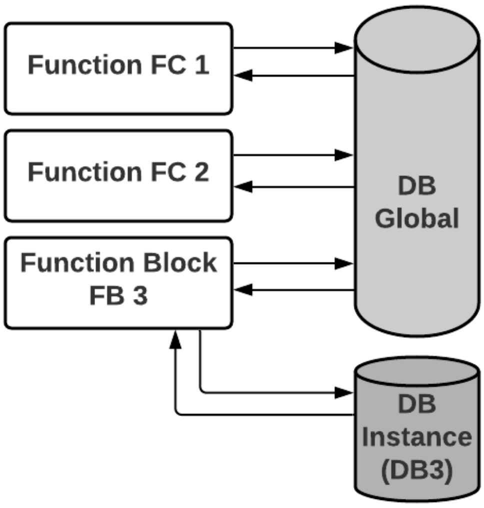


Figure 3.7: Representation of the various accesses to data blocks on TIA Portal [20]

Depending on the application specifications, it is possible to select a linear or modular structure to create the user program:

- A linear program executes all the instructions of the program sequentially. It means, one after the other. Usually, a linear program has all the instructions into the OB for the program's cycle execution (OB 1).

- A modular program calls different block functions to execute specific tasks. To create a modular program, the main task is divided into less complicated small subroutines. The program is structured by calling one code block from the block. Fig 3.8 shows the structures mentioned above[20].

By designing FBs and FCs that perform generic tasks, modular logic blocks are created. The program is then structured so that other code blocks call these reusable building blocks. The calling block passes the device-specific parameters to the called block. When one code block calls another code block, the CPU executes the program code in the called block. Once the called block's execution is completed, the CPU will resume the execution of the called block. Continue to execute the instruction processing after the block call. Fig 3.9 shows the process just described.

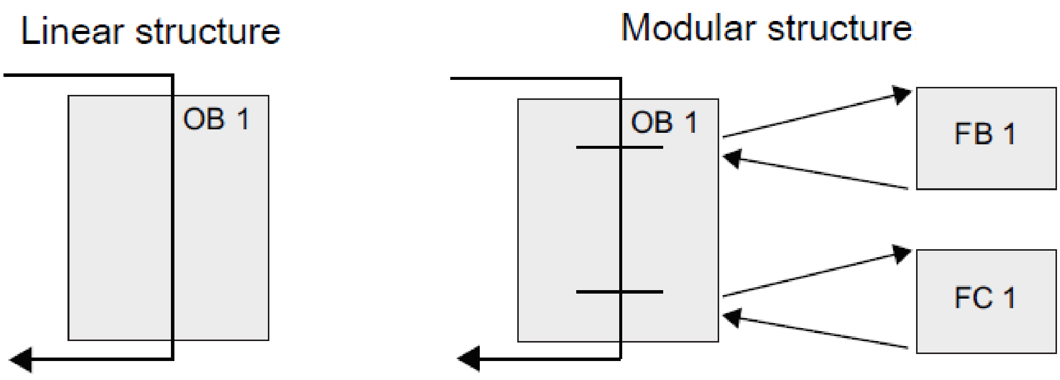


Figure 3.8: Linear and Modular structure program on TIA Portal [20]

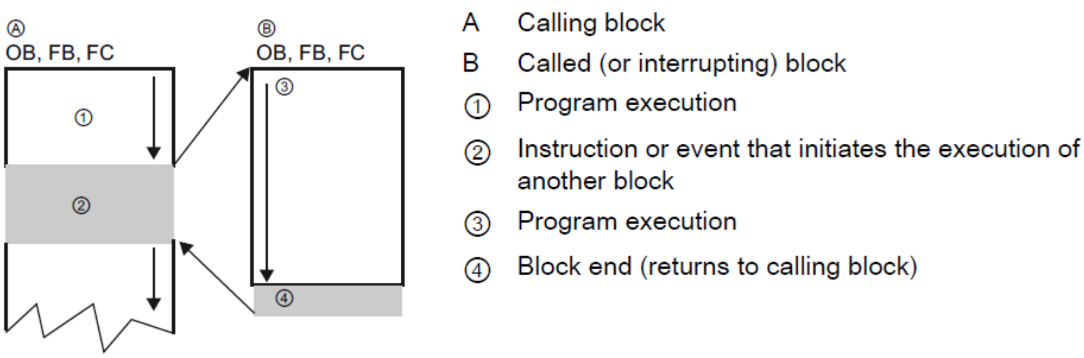


Figure 3.9: Code Block calling another Code Block on TIA Portal [20]

3.3.2 Trace and logic analyzer fuction

STEP 7 presents trace and logic analyzer functions, which can be used to configure variables for PLC tracking and recording. The recorded trace data can then be uploaded to the programming device and use STEP 7 tools to analyze, manage, and represent it graphically. Therefore, the trace and logic analyzer functions are suitable for monitoring highly dynamic processes. When the recording is activated again, the recorded value will be overwritten. This function is crucial in this thesis because it helps to understand the system's behavior in real-time [20].

Fig 3.10 shows the mode of operation of the trace function:

Step 1: **Trace configuration:** The user specifies the signal to be recorded, the trigger condition, and the duration. The trace configuration depends highly on the used device.

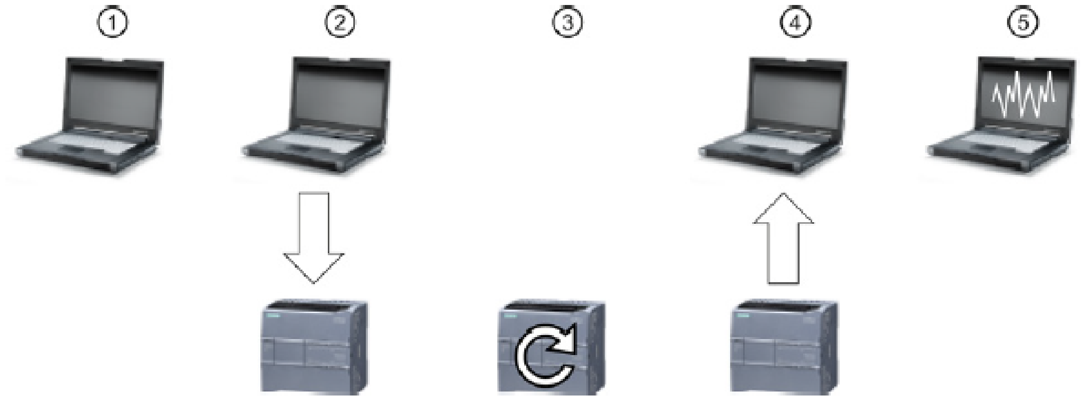


Figure 3.10: Mode of operation of the trace function[20]

Step 2: **Transferring the trace configuration:** the trace configuration is transferred to the PLC when an online connection is established.

Step 3: **Waiting for the recording:** The PLC executes the program. As soon as the trigger condition is satisfied, the recording starts.

Step 4: **Transfer measurement to programming device**: Once the recording finishes, the data is saved on the TIA portal's opened project. The measurement can be saved at any time after completing the recording.

Step 5: **Display, manage and save the measurement:** The trace function gives several options to analyze the measurement. Different ways to display the data can be executed, for example, a bit representation for binary signals. Besides, signal waves from diverse measurements can be analyzed together and compared with each other. The data can also be exported as a CSV (comma-separated value) file, which allows the user to analyze the measurement in a more dedicated software like Matlab or Spreadsheet.

3.3.3 Trace software user interface

The user interface of the trace function includes multiple areas. The layout of the user interface in the TIA portal is shown in Fig 3.11

1. The title bar of the working area shows the device to which the current display belongs.
2. The trace toolbar has the buttons for managing the project, for example, activation/deactivation of the traces, deletion, export of trace configuration, and measurement.

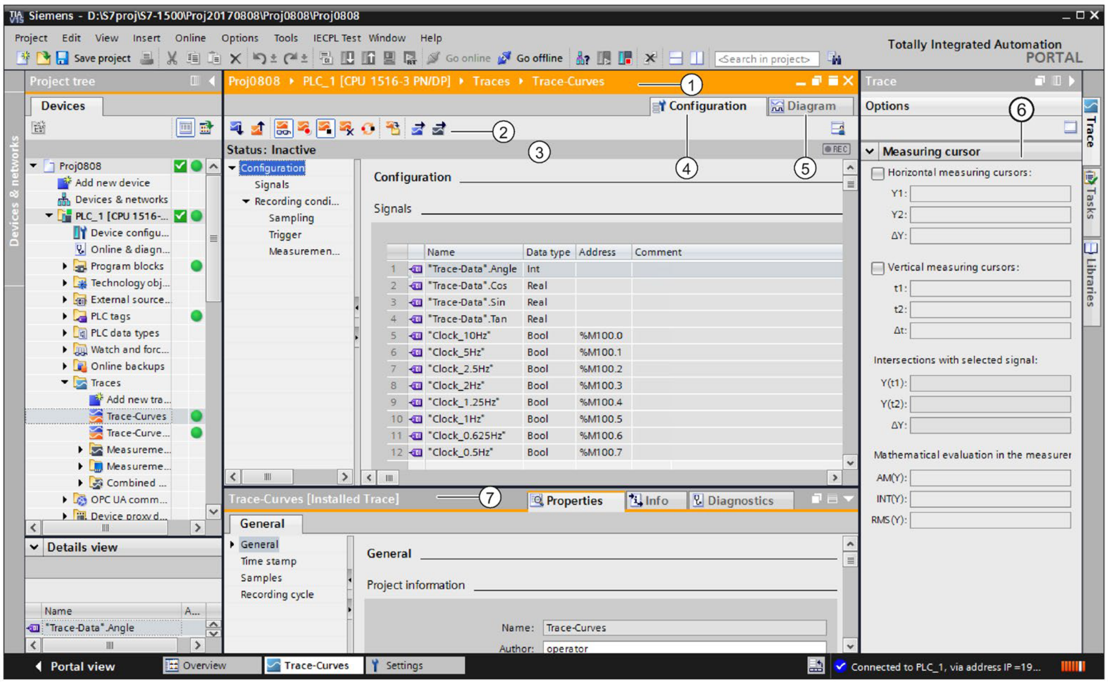


Figure 3.11: Layout of the Trace function [20]

3. The status display of the trace shows the current state of the recording.

4. The user can configure the recording duration, trigger condition, and signals to trace on the configuration tab.

5. The diagram tab displays the recorded values as a curve diagram and the signals from the displayed measurement.

6. The trace task card displays the measurement cursor data mathematical evaluation.

7. The inspector window shows the general information about the trace configuration.

As an example, Fig 3.12 displays four different signals. The fists one represent the position of the piston rod-end of the cylinder. The second signal shows the Pulse Width Modulation (PWM) used to control a 5/3 closed center digital valve. The third and fourth signals represent when the solenoids of the 5/3 valve are active.

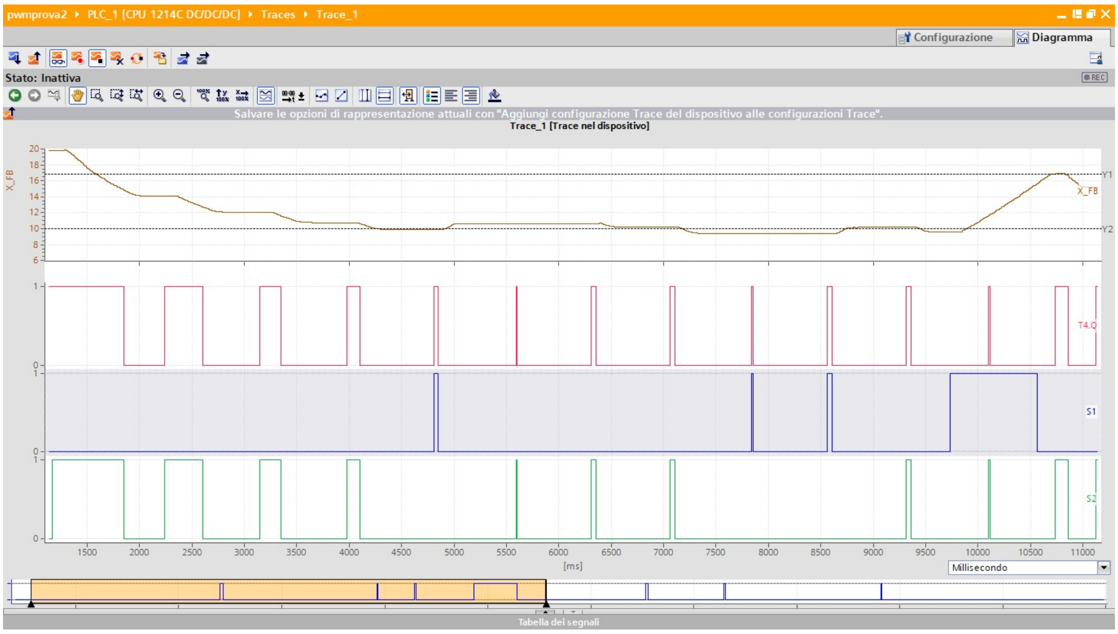


Figure 3.12: Snapshot of signals measurements

3.4 Camozzi CX06, Multi-serial Module

The serial module CX06 can manage traditional serial communication protocols and new generation ones such as EtherCAT, EtherNet / IP, and PROFINET.

The PLC is connected via ethernet to the CX06, and the two communicate via standard Profinet protocol. This configuration makes it possible to expand the number of inputs and outputs of the PLC up to a maximum of 128 bytes and 128 bytes, respectively (1024 digital inputs and 1024 digital outputs). The specifications can be observed in the table below. The memory of the module CX06 is divided into input and output sections, so it is impossible to have a configuration of 140 bytes dedicated to input and 116 bytes dedicated to output. Furthermore, CX06 can be up to 100m away from the PLC, which is the maximum distance covered by Profinet. The table above shows the main characteristic of the CX06 module [16].

General Features of the module CX06	
Number of digital outputs	1024
Number of digital inputs	1024
Maximum input absorption	1.5 A
Maximum output absorption	3 A
Voltage Supply	24 V DC +/-10%
Operating temperature	0-50°
Material	Aluminum

Table 3.3: Main features of the CX06 module

A series of modules must be added to the CX06 mainboard. It is possible to insert not only digital I/O modules but also analog I/O modules as well as valve islands and subnet modules. The CX06 installed on the test bench is shown in Fig 3.13.From left to right, the following modules are exposed and described.

Figure 3.13: CX06 and input-output modules

- **ME3-0800-DC**: The module has eight digital input, corresponding to 1 byte of the CX06 input memory,The features of the module are shown below[16].

Mod.	Number of digital inputs	Connection	Number of connectors	Sensor supply	Overvoltage protection	Absorption	Type of signal	Operating temperature
ME3-0800-DC	8	M8 3 pin female	8	24V DC	400 mA for 4 sensors	10 mA	PNP	0 ÷ 50°C

Table 3.4: ME3-0800-DC module features[16]

- **ME3-0004-DL:** The module has four high-power digital outputs (corresponding to 4 bits of the CX06 output memory) capable of powering large solenoid valves. Each five poles connector on the module corresponds to two outputs.

In the case of our test bench, two of the four outputs (outputs 3 and 4) are connected to the 5/3 digital valve. The connections, as well as the electric diagram, are shown in Fig 3.14 [16].

Mod.	Number of digital inputs	Connection	Number of connectors	Sensor supply	Max power for digital output	Absorption	Type of signal	Operating temperature
ME3-0004-DL	4	M12 A 5 pin female	2	24V DC	10 W	10 mA	NPN	0 ÷ 50°C

Table 3.5: ME3-0004-DL module features

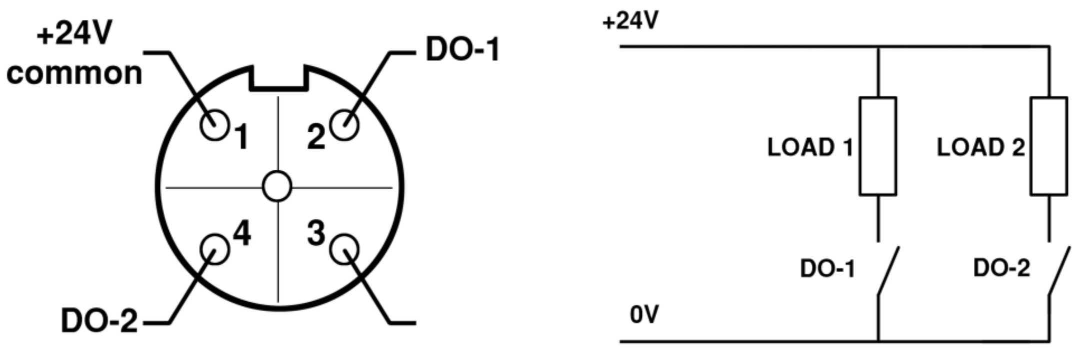
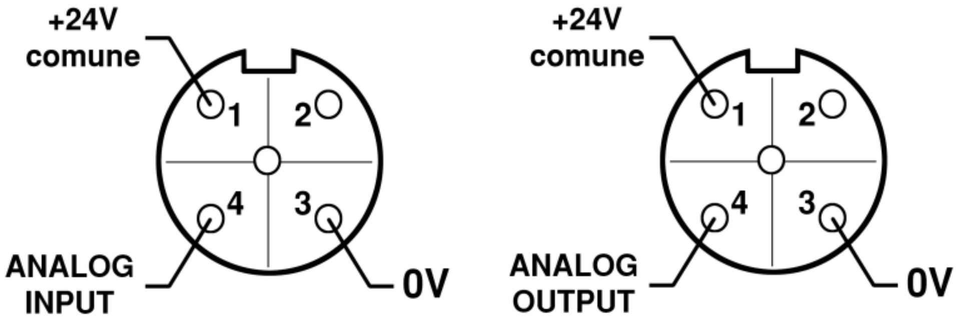


Figure 3.14: Connections and electrical diagram of the digital output module ME3-0004-DL[16]

- **ME3-00T0-AL**: The module has two analog outputs operating in the range 0-10V DC. Given the DAC resolution of 12 bits, each output occupies 2 bytes in the memory of the CX06, corresponding to a total of 32 bits. The electrical connections follow the same scheme as the ones presented in Fig 3.15b. The module is used to send a reference signal to the proportional valves present on the test bench when both are working in slave mode [16]. The features of the module are listed above.

Mod.	Number of analog inputs	Number of analog outputs	Connection
ME3-00T0-AL	-	2 outputs 0-10 V	2x M12 A 5 pin female

Table 3.6: ME3-00T0-AL module features [16]



(a) Analog Input Connector **(b)** Analog Output Connector.

Figure 3.15: ME3-00K0-AL analog input/output module connections

- **ME3-00K0-AL:** The module is similar to the ME3-00T0-AL, but this time it has one analog input and one analog output, both working in the range 0-10V DC. The resolution of the input can be modified, from 12 down to 8 bits, if need be. Fig 3.15 shows the proper connections for each five poles connector. Each I/O occupies 2 bytes of memory (or 1 byte in the analog input case with a resolution set to 8 bits). The features are shown in the table below.

Mod.	Number of analog inputs	Number of analog outputs	Connection
ME3-00K0-AL	1 input 0-10 V	1 output 0-10 V	2x M12 A 5 pin female

Table 3.7: ME3-00K0-AL module features [16]

When using 12-bit resolution, the analog data will occupy 2 bytes of memory, especially the 12 most significant bits (Fig 3.16). Another problem is that communication uses little-endian encoding. This means that the first byte is actually saved as the second byte (Fig 3.17). In order to reconstruct the data correctly, the easiest way is to use ROL or Roll Left block in TIA Portal: This block scrolls the specified number of bits to the left; every time a point is pushed out of the byte, it will occupy the right side to release s position. The operation results are shown in Fig 3.18.The four zeros now occupy the four most significant bits, and all other bits are arranged from right to right.

.

BYTE 1								BYTE 2							
BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
MSB											LSB	0	0	0	0
WORD															

Figure 3.16: Analog data inside CX06 memory

BIG ENDIAN	BYTE 1	BYTE 0
LITTLE ENDIAN	BYTE 0	BYTE 1

Figure 3.17: Big Endian and Little Endian encoding

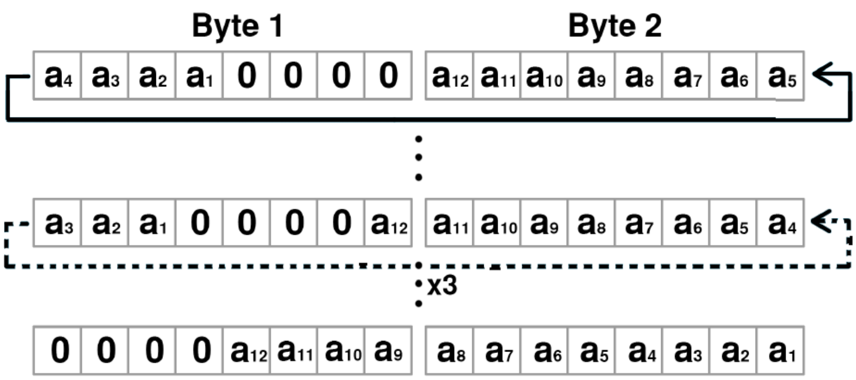


Figure 3.18: ROL operation

3.4.1 CX06 module configuration

The right configuration of the CX06 is crucial to communicate the module with the PLC. First, install the CX-Configurator software on any Windows PC. Then remove the plastic cover placed on the top cover of the CX06 CPU module, and connect the CX06 to the PC via a micro USB cable. It is recommended to disconnect the CX06 from the PLC so that the CX only communicates with the PC to obtain the best performance. This can be done via hardware, disconnecting the Ethernet cable, or via software via CX-Configurator. When the test bench is on, the window showed in the Figure 3.19 will appear.

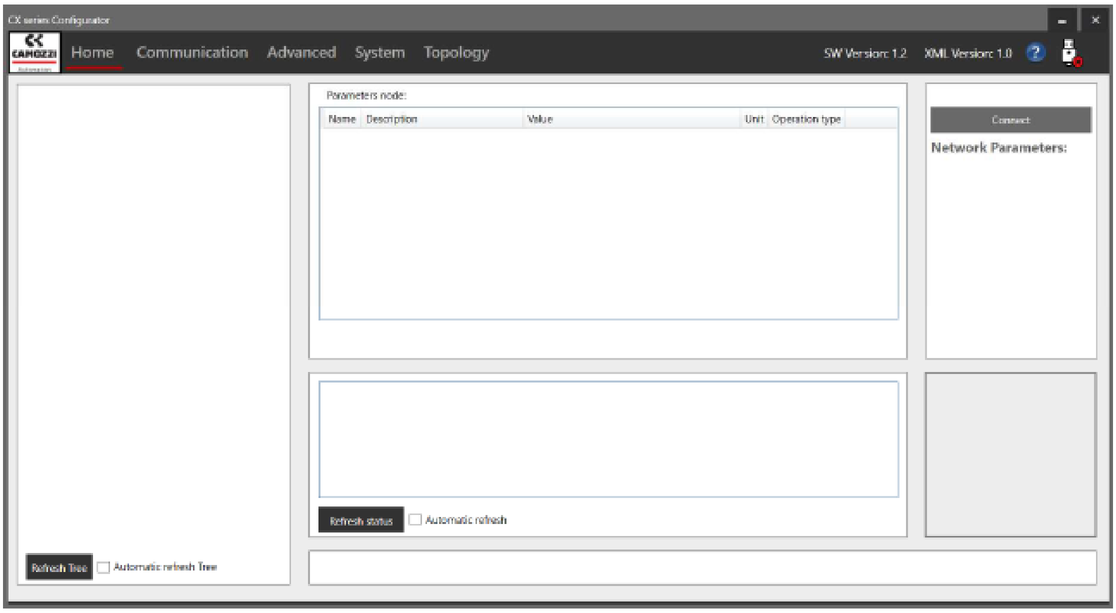


Figure 3.19: CX06-Configurator after bootup

Click on the connect button on the right and choose the correct COM port from the dropdown menu (Fig 3.20). Finally, hit connect and wait a few seconds for the communication to take place.

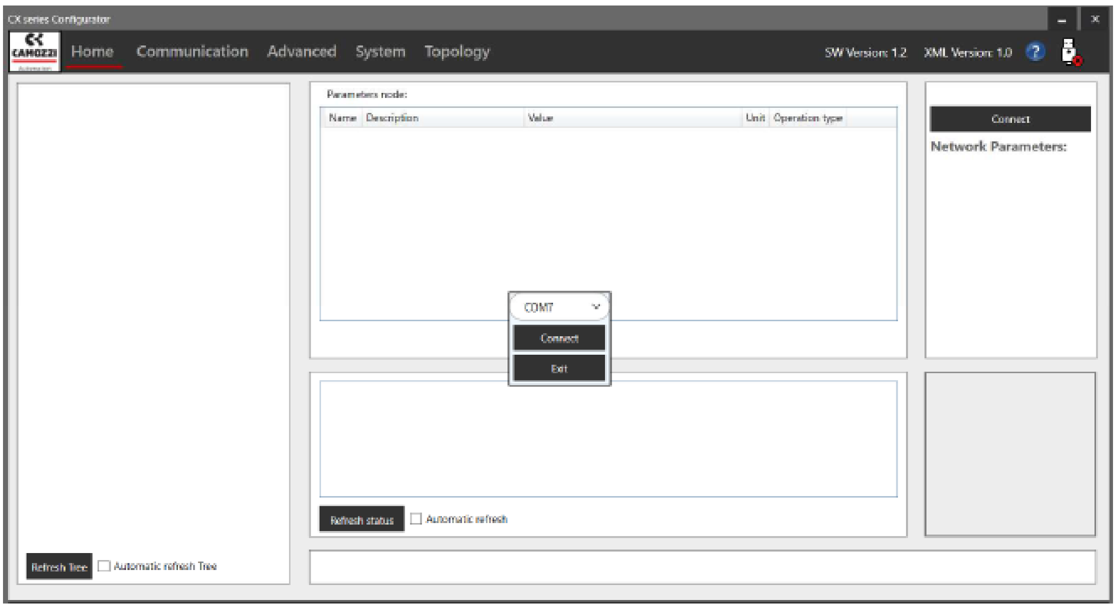


Figure 3.20: Connection to the CX06

Inside the left side panel, a tree represents the current CX06 configuration, with all the modules, including the subnet ones. Modules are numbered starting from the leftmost one. By clicking on each module in the tree, its properties will be displayed in the top center panel. The middle bottom tab shows the status of the selected module's input and output (if it exists). Fig 3.21 shows the first digital input module. In this case, the bottom panel displays the current status of the eight digital inputs.

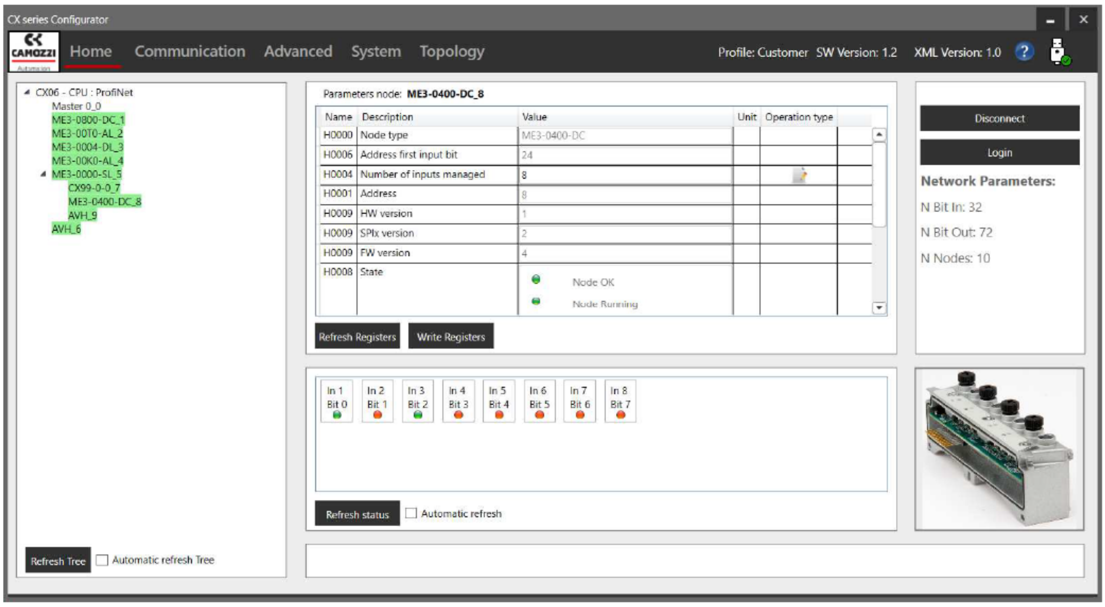


Figure 3.21: CX06 module properties

A vital step in the configuration is to check each module's starting address and write it down. This is useful when setting up the hardware configuration in the TIA Portal. Each module has a row in the property table with the title "Address of the first input/output bit" followed by a number. Dividing that number by eight will show the bytes associated with that particular module. Note that the PLC's input and output bytes are already occupied by its internal input and output, bytes 0 and 1. Therefore, the first byte (input and output) of CX06 will be byte 2 of the PLC. Fig3.21 displays the module ME3-0004-DL. The first input bit is the number 24. This translates to byte number 3 and therefore byte 5 inside the PLC input memory.

The last topic about CX06 configuration involves the mapping from memory bytes to the corresponding modules. As mentioned earlier, for each module, the corresponding byte can be found. A useful trick is to allocate no less than one byte for each module when less than one byte is sufficient. For example, the digital output module having just four digital outputs, four bits would be enough to store the module outputs state. The problem comes from the fact that the next module will start in the middle of a byte instead of a new byte. This is especially problematic when the following module is an analog input or output. Since PLC uses bits or bytes, it is very troublesome to manage data that takes up half a byte and then expands to the next byte, and it quickly becomes unprocessable. This is why it is always best to allocate at least one byte for each module.

3.4.2 TIA Portal v16, C06x and PLC communication

The hardware configuration must be set in the TIA Portal project to make the S7-1200 communicate with CX06. Proceed as follows.

Step 1: Create a new project on TIA Portal and insert the PLC inside the project.

Step 2: Double click on "Devices & networks." (Fig 3.22)

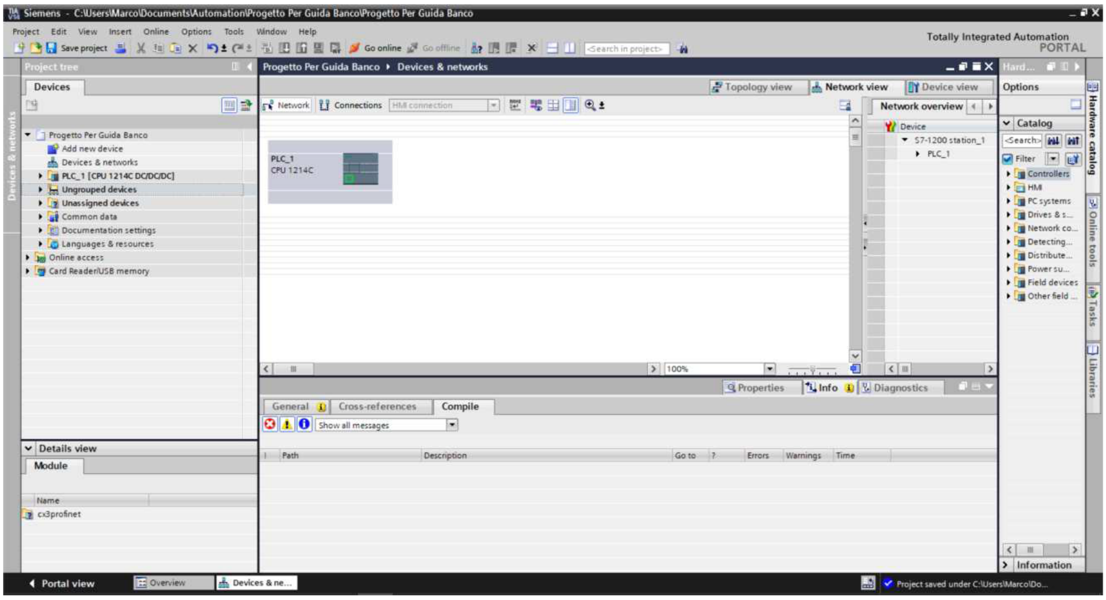


Figure 3.22: Devices & networks screen

Step 3: In the Catalog tab on the right, open "Other Field Devices" -> ”PROFINET IO”->”I/O”->" Camozzi Spa" ->" Serie CX"

Step 4: Drag and drop the "CX06-PNS Adapter rev1" device next to the PLC already created. (Fig Fig 3.23)

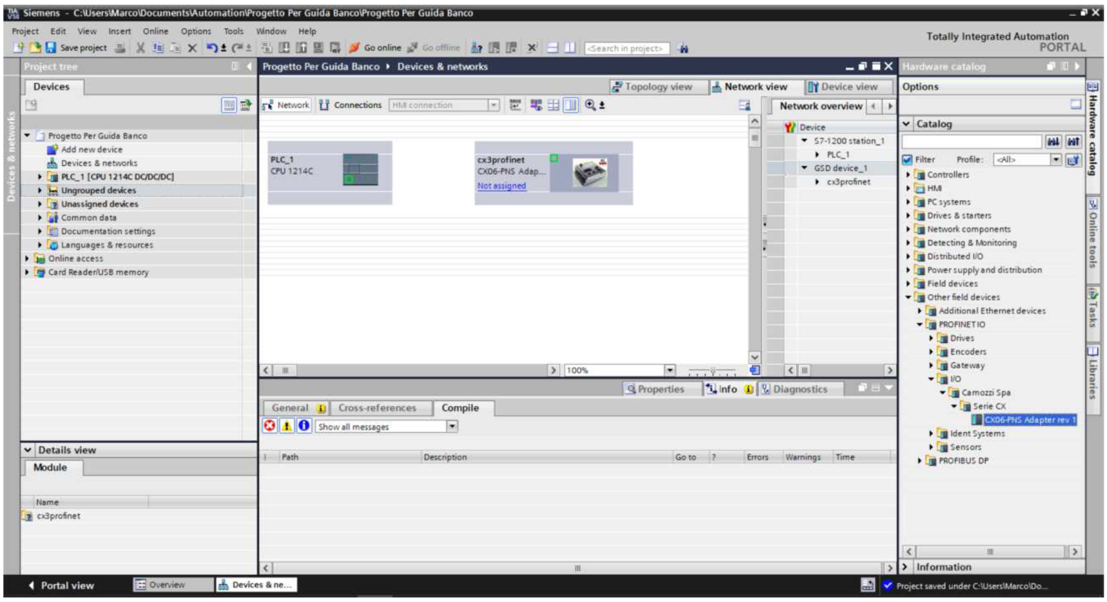


Figure 3.23: Insertion of the CX06 inside the TIA Portal

Step 5: Left-click and hold on the green Ethernet port on the PLC, drag, and release on the green square on the CX06 module. The PLC and the CX06 are now connected by a green line as shown in Fig 3.24

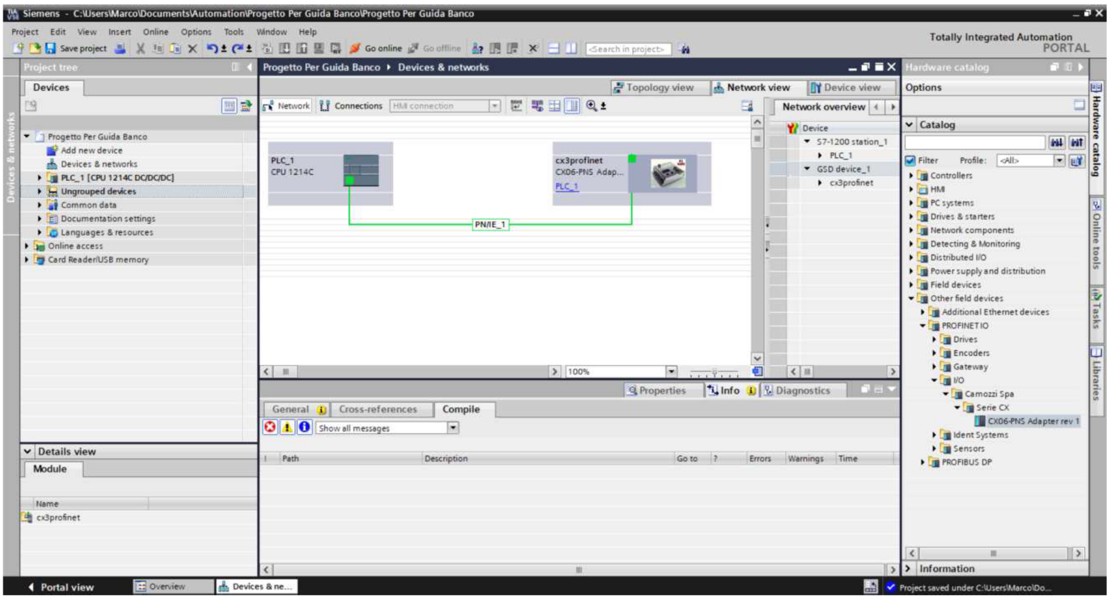


Figure 3.24: CX06 and PLC connection

Step 6: Open the CX06 by clicking on the image, and a window will appear (Fig 3.25). Then, expand the folder at the right to see the Input modules folder and Output modules.

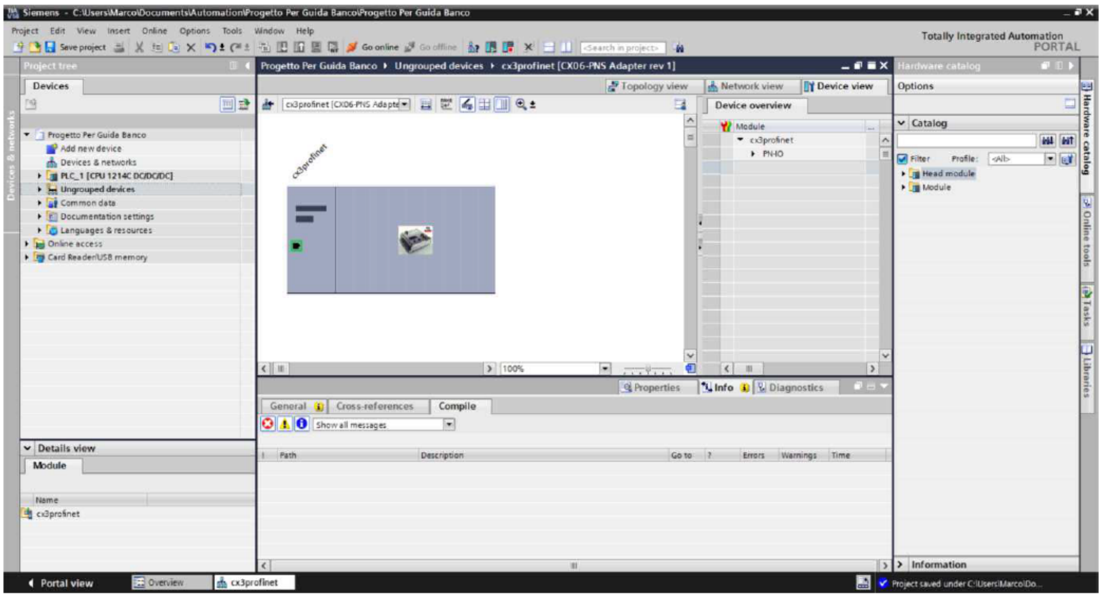


Figure 3.25: CX06 configuration inside the Tia Portal

Step 7: Drag and drop the correct number of input and output bytes inside the table next to the CX06.

Step 8: The first time, the table already contains 128 bytes of inputs and 128 bytes of outputs. Delete the two and populate the table according to the specific modules connected to the CX06.

This step must be performed correctly, so ensure the correct number of bytes of the correct type for each physical module connected to the CX06. Figure 3.26 shows the address of each byte/module of CX06 seen from the PLC.

3.5 6PF Series Positioning Feedback cylinders

The 6PF3P050A0200 (Fig 3.28) from Camozzi is a double-acting, single-rod cylinder with a stroke of 200mm and bore equal to 50mm. This actuator possesses a potentiometer linear position transducer inside the rod. The table below shows the general and technical features [18].

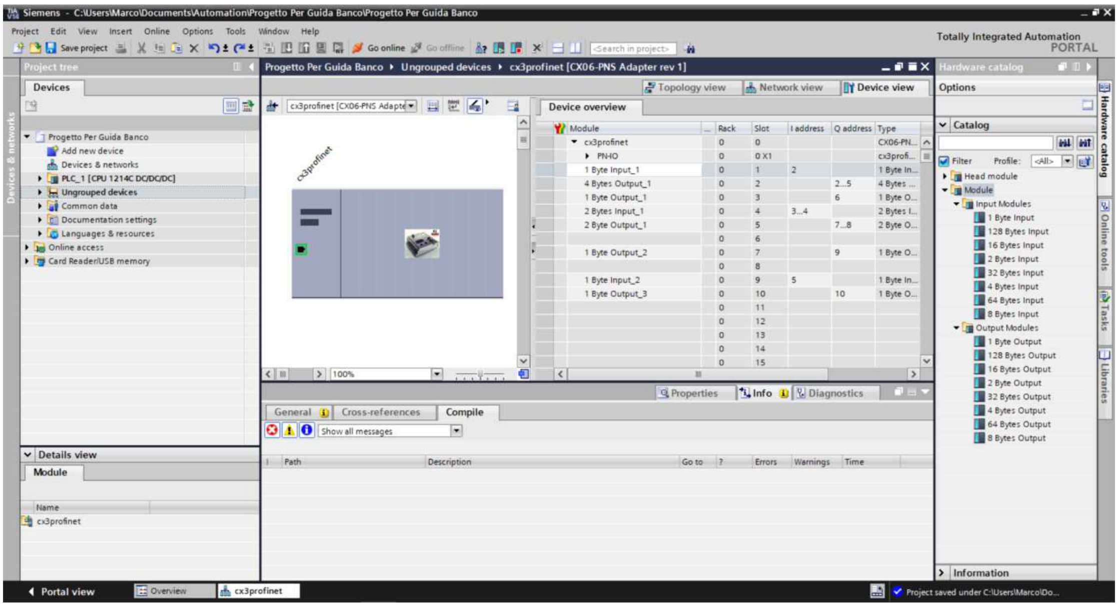
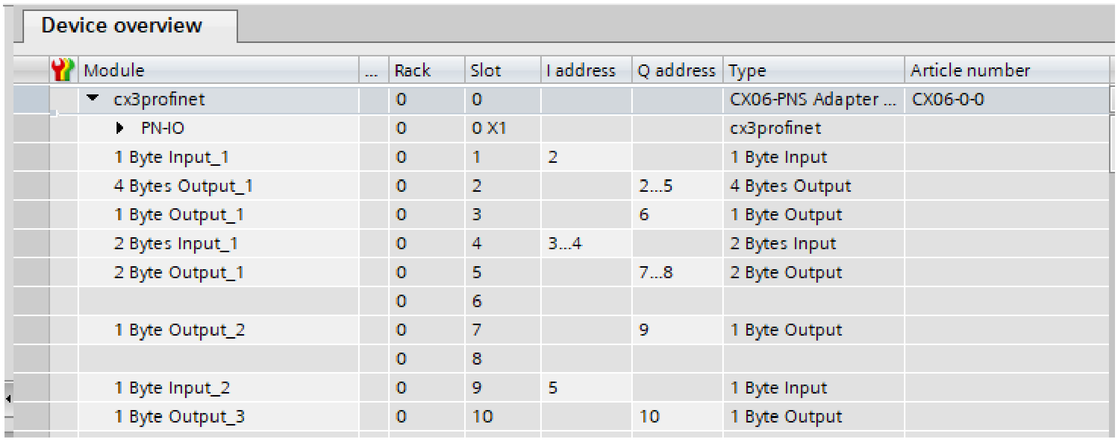


Figure 3.26: CX06 memory configuration inside the TIA Portal



Device overview

Module	...	Rack	Slot	I address	Q address	Type	Article number
cx3profinet		0	0			CX06-PNS Adapter ...	CX06-0-0
PN-IO		0	0 X1			cx3profinet	
1 Byte Input_1		0	1	2		1 Byte Input	
4 Bytes Output_1		0	2		2...5	4 Bytes Output	
1 Byte Output_1		0	3		6	1 Byte Output	
2 Bytes Input_1		0	4	3...4		2 Bytes Input	
2 Byte Output_1		0	5		7...8	2 Byte Output	
		0	6				
1 Byte Output_2		0	7		9	1 Byte Output	
		0	8				
1 Byte Input_2		0	9	5		1 Byte Input	
1 Byte Output_3		0	10		10	1 Byte Output	

Figure 3.27: CX06 memory configuration inside the TIA Portal (zoom)

This type of cylinder permits a constant control of the rod position by reading the transducer's internal resistance change along the entire stroke. The symbol and connection pins are shown in Fig 3.29.

For a proper operation, the potentiometer should be used as a voltage divider instead of a variable resistor. The measurement must detect the voltage and not the resistance. As already said, the transducer consists of a linear potentiometer; consequently, the output depends on the input voltage. Based on the fact that

PNEUMATIC SECTION	
Operation	double-acting low friction, not cushioned
Bore	50mm
Stroke	200mm
Linearity	0.1% of the stroke
Operating pressure	0.1 ÷ 10 bar
ELECTRICAL SECTION	
Max input voltage	60 V
Electrical resistance	10 k ohm
Tolerance on resistance	+/- 20%

Table 3.8: General and technical features

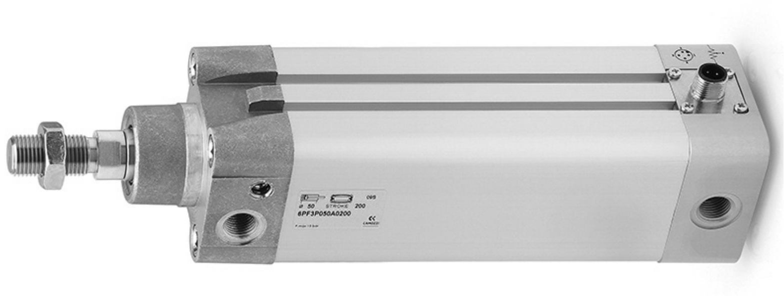


Figure 3.28: Positioning Feedback Cylinder 6PF3P050A0200 [18]

the analog input module sends out 24V DC, but the transducer only accepts 10V DC as maximum input voltage, a voltage divider was used to give 10V DC as input to the transducer in order to get an output in the valid range 0-10V DC [18].

The piston is also equipped with a permanent magnet which enables the use of external end-stroke sensors. Robust design and high performance make this cylinder suitable for tensioning cylinders, positioning cylinders, and filling, cutting and measuring systems [18].

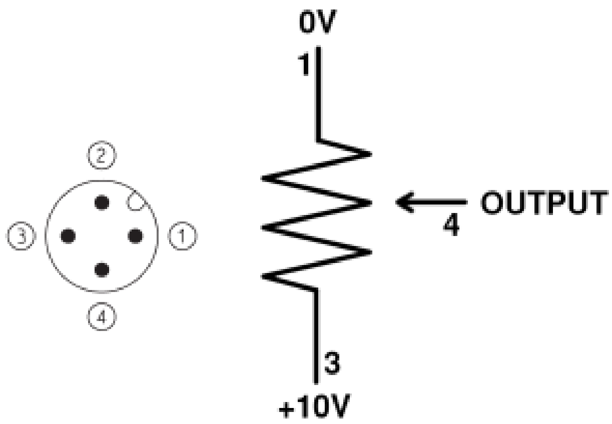


Figure 3.29: Position transducer symbol and connections[18]

3.6 Valves

3.6.1 5/3 closed center solenoid valve

The valve (code 368-011-02) is a digital 5/3 closed center valve. Figure 3.30 shows the valve and its ISO symbol. The command signals of the valve coils S1 and S2 are respectively connected to the digital output 3 and 4 of the ME3-0004-DL module through the same cable. This valve is used to control the induction cylinder placed on it, ensuring a 700 Nl/min ANR flow rate and a working pressure of 2÷10 bar [21].

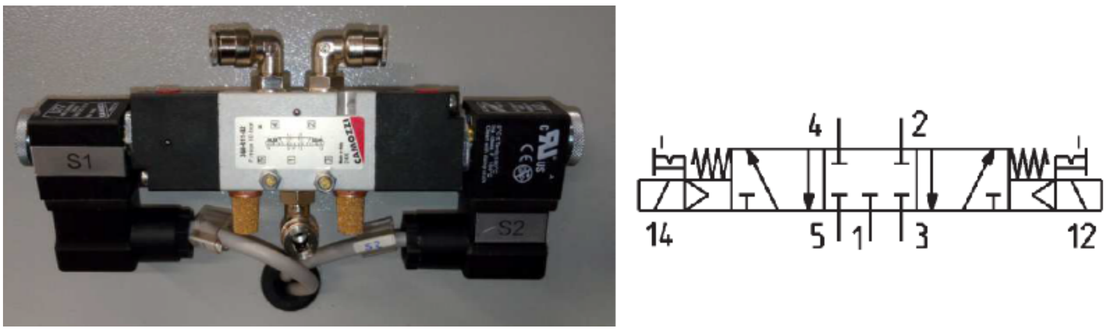


Figure 3.30: 5/3 closed center valve and relative ISO symbol [21]

The table 3.9 shows the features of the 5/3 closed center solenoid valve.

Model	Function	Flow Rate (Nl/min)	Operating pressure (bar)
368-011-02	5/3 CC	700	$2 \div 10$

Table 3.9: 5/3-way solenoid valve features

3.6.2 Directly Operated Proportional Valves

These valves are also produced by Camozzi and marketed under the code "AP-7211-LR2-U7". These are normally closed 2/2 solenoid valves, size 22 mm. Moreover, the valves can be controlled directly by the PLC through the Analog Output SM, and another alternative to control them is by means of a particular driver capable of guaranteeing proportional behavior. Fig 3.31 shows the valve and its relative ISO symbol, next, the table 3.10 below shows the main features of the mentioned valve.

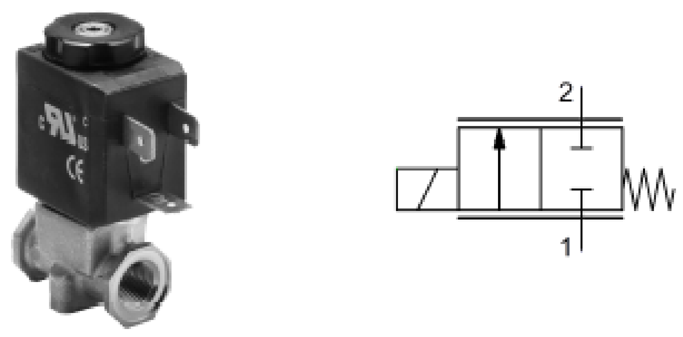


Figure 3.31: Valve "AP-7211-LR2-U7" and relative ISO symbol [22]

Model	Function	Orifice (mm)	Nominal flow Kv (l/min)	P Max (bar)
AP-7211-LR2-U7	2/2 NC	1.6	1.0	6 bar

Table 3.10: AP-7211-LR2-U7 valve features

3.6.3 Electronic Control Device for Proportional Valves

Series 130 electronic control device for proportional valves (Fig 3.32) allows the pilot control of any valve with a maximum current of 1A. The control device transforms a standard input signal (0-10V DC or 4-20 mA) into a PWM signal to obtain at the solenoid outlet a current which is proportional to the input signal. The particular driver used has the code "130-222", works at a frequency of 500 Hz, also confirmed by an oscilloscope, and has a maximum output of 24 V DC, compatible with the input of the valves used. The features of the control device are listed in the table 3.11.

Figure 3.32: Series 130 electronic control device for proportional valves [23]

Model	130-222
Power Supply	24 V DC
Power Consumption	6.5 W
Analog Input	0-10 V or 4-20 mA
Ouput	PWM 500 Hz
Max Current	1 A

Table 3.11: General Charecteristics of the Control Device

Device Functionality

The control device has an important feature that makes it possible to compensate for variations due to solenoid heating or supply voltage changes. The next figure shows the block diagram of the device functioning.

The Ramp time is the slot of time needed to pass from the minimum current value to the maximum current value supplied to the load. This ramp is changed by

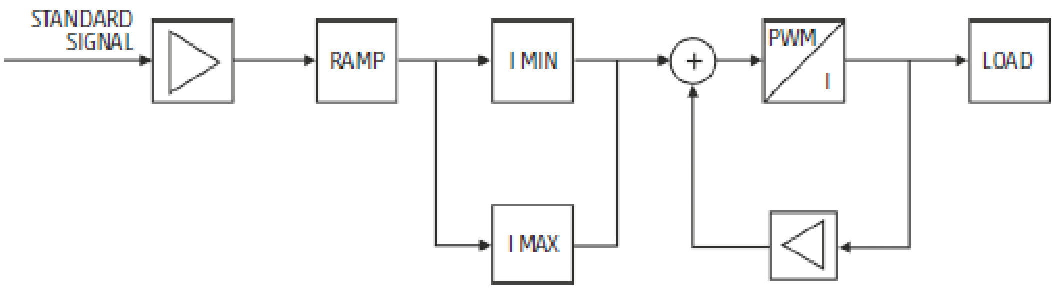
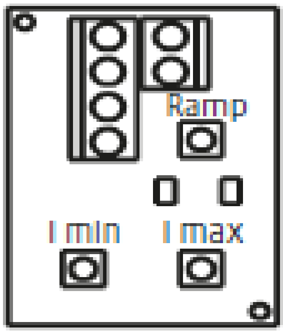
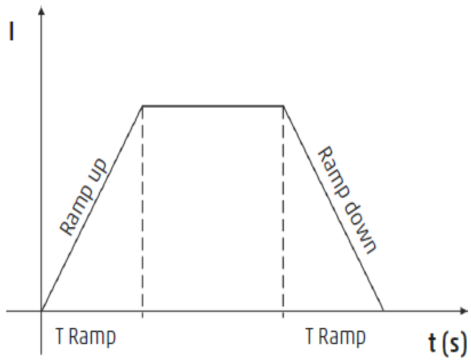


Figure 3.33: Block Diagram of the device functioning [23]

turning the RAMP timmer clockwise or counterclockwise. The output signal can have a ramp progression adjustable between 0, which is the default value, and 5 s that is the maximum [23]. Figure 3.34a shows the position of the device's trimmers and the Figure 3.34b shows the curve of the time's variation.



(a) Position of the adjustment trimmer [23]



(b) Ramp time calibration [23]

Figure 3.34: Schematic and Ramp time [23]

Moreover, the maximum and the minimum can be calibrated. The setting of the **minimum offset current** permits the valve's dead zone to be canceled. Even if there is no reference signal and an invalid reference signal (the reference signal 4÷20 mA in the current is lower than 4 mA), the device allows the minimum current setting, so the valve can also be opened. Therefore, it must be ensured that the valve's opening without a reference signal (or the current of the reference signal is less than 4 mA) will not cause property damage or personal injury during operation.

The trimmer I max determines the **maximum value of the current** supplied to the valve with a 100% reference signal. Therefore, the maximum current setting is used to limit the maximum flow value of the pilot valve. The curve of the current calibration is shown in the figure below.

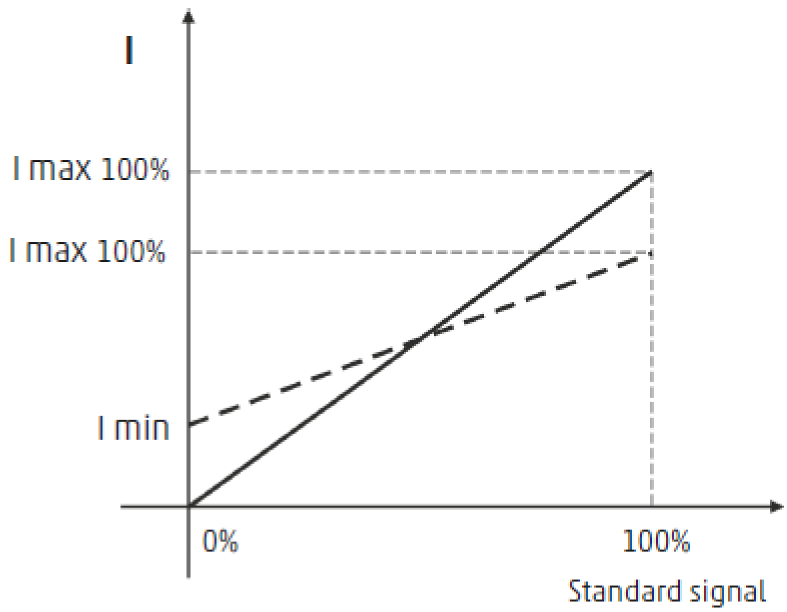


Figure 3.35: Current flow Calibracion [23]

3.6.4 Series LR digital proportional servo valves

Series LR digital proportional solenoid valves are 3/3-way, directly actuated with a patented rotary spool system, with a closed-loop control circuit. The electronic card is integrated into the valve body, ready to be connected. The LR Series digital proportional solenoid valves have been designed to be as compact as possible to save space and power. Thanks to this new digital version, the valve can be configured through a USB connector according to different requirements. The Fig 3.36 show the LR Series valve and it respective ISO symbol.

Figure 3.36: LR Series digital proportional solenoid valve and ISO symbol [24]

Power Supply	24 V DC, max absortion 1.5 A
Control signal	+/- 10 V
	0-10 V
	4-20 mA
Working temperature	0 to 50 °C
Supply pressure	-0.9 to 10 bar
Maximum flow rate 6.3bar ΔP 1bar	430 Nl/min
Hardware configuration port	micro USB

Table 3.12: General Characteristics of the Series LR Servo Valves

The next table illustrates the pins of the M12 8 pole connector located on the upper section of the LR series servo-valves (Fig 3.37).

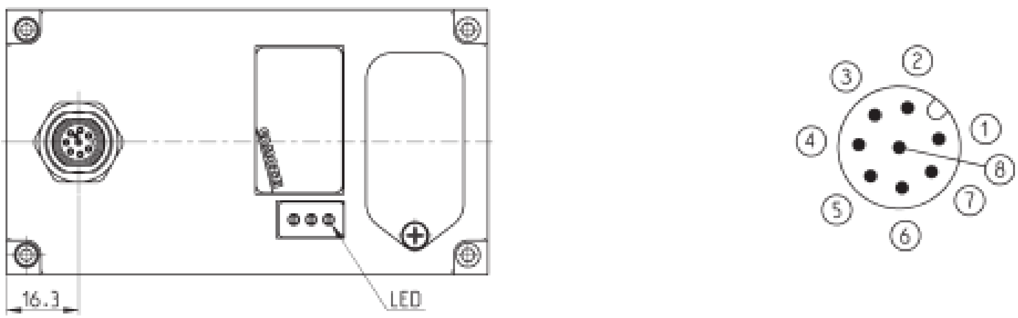


Figure 3.37: LR Series pins connection [24]

PIN	SIGNAL		DESCRIPTION
1	+5 V		+5V power supply for external potentiometer transducer (ref. GND). If used, it is necessary to connect RIF- with GND.
2	24 V DC		24V DC power supply (logic and motor): connect to the positive pole of the 24V DC power supply (ref. GND)
3	RIF -		GND reference or NEGATIVE pole of the command signal (0-10V / 4-20mA / ±10V)
4	RIF +		POSITIVE reference of the command signal (0-10V / 4-20mA / ±10V)
5	EXT	for LRXD	feedback signal of the external transducer 0-5V / 0-10V / 4-20mA (ref. RIF-)
		for LRWD	not used
6	FBK		feedback signal 0-10V / 4-20mA (ref. GND)
7	GND		common (reference pin 1 and 2): connect to the negative pole of the 24V DC power supply (compulsory)
8	ERR	for LRXD	command signal 0-10V for slave valve (ref. GND)
		for LRWD	error signal (output) 0-24V (ref. GND)

Table 3.13: Power supply connector

3.6.5 Servo valve for position control (LRXD2) - Master Valve

The LRXD2 servo valve is a proportional valve with high-precision integrated control for cylinder positioning. The valve includes a patented 3/3-way system based on the principle of a rotating spool and electronically controls the spool's position.

The closed-loop pneumatic servo system (Fig 3.38) allows position control via feedback from a position sensor or Camozzi 6PF cylinder with an integrated linear sensor. The speed and acceleration are directly managed by the electronic card integrated into the valve body. The master valve model LRXD2 is equipped with a special signal for controlling the LRWD2 valve, which will act as a slave-valve [24].

Configuration of the LRXD series servovalve [25]

- The software "LrxdConfigurator" should be downloaded from the website *http://www.camozzi.com* to configure the LRXD series servo valve, and then install it according to the on-screen instructions displayed in the process.

- When the software "LrxdConfigurator" is started, the system will verify the communication between the LRXD series servo valve and the PC with the installed configuration. If communication fails, the table 3.14 shows the possible errors that can occur.

- The "LrxdConfigurator" software can receive certain information and enter many settings. The screen displayed is shown in Fig 3.39, and the available

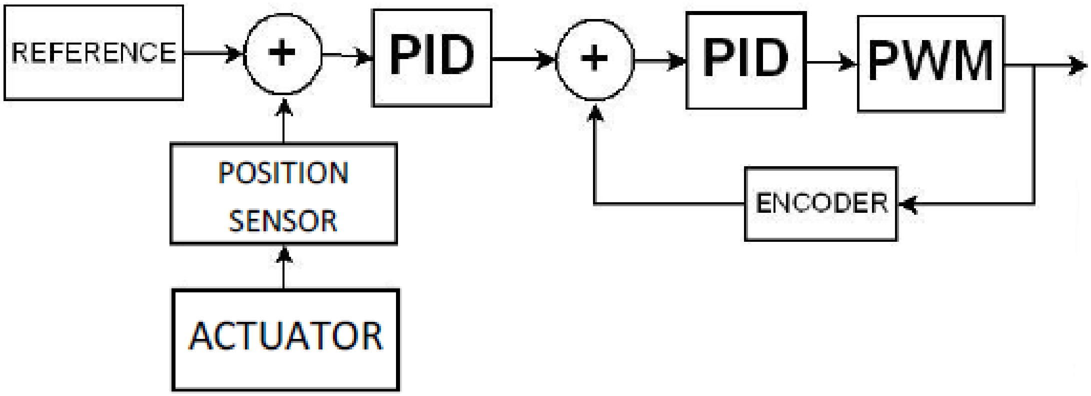


Figure 3.38: Electrical circuit of the LRXD valve [24]

Type of fault	Causes	Remedy
Communication failure between servovalve and PC	Electrical power supply not connected	Connect the Electrical power supply by means of the M12 connector and ensure that the green led PWR lights up.
	USB cable not connected	Connect the USB cable to one of the ports available on the PC and to the Micro USB connector under the transparent panel on the servovalve.
	USB drivers not installed	Contact the Camozzi technical assistance service.

Table 3.14: Types of faults

function levels will vary for different user types using the configurator. If the user is logged in as a "consumer," the servo valve parameters cannot be modified, nor can they be displayed. By entering the password and pressing the command "Apply," the user will be registered as a "producer," in this case, the servo valve parameters can be modified. The factory default value of this password is "INIT," which can be changed by writing a new password and pressing the command "Save" (only when the user is logged in as a "consumer"). Specific commands are disabled for "consumer" and "producer" users and can only be used in Camozzi Service.

- In the **"AUTO SETTING OF THE FEEDBACK EFFECT”** frame, you can select one from 3 different preset sets of K parameters of the internal PID controller: *Slow*, *Medium*, and *Fast*. When one of them is selected, the software will display the setting values six K parameters *(LKP, LKA, LKV, LKC, LKY* and *LKI)* in the relevant text box. In any case, in the **"MANUAL SETTINGS OF THE FEEDBACK EFFECT"** frame, you can modify

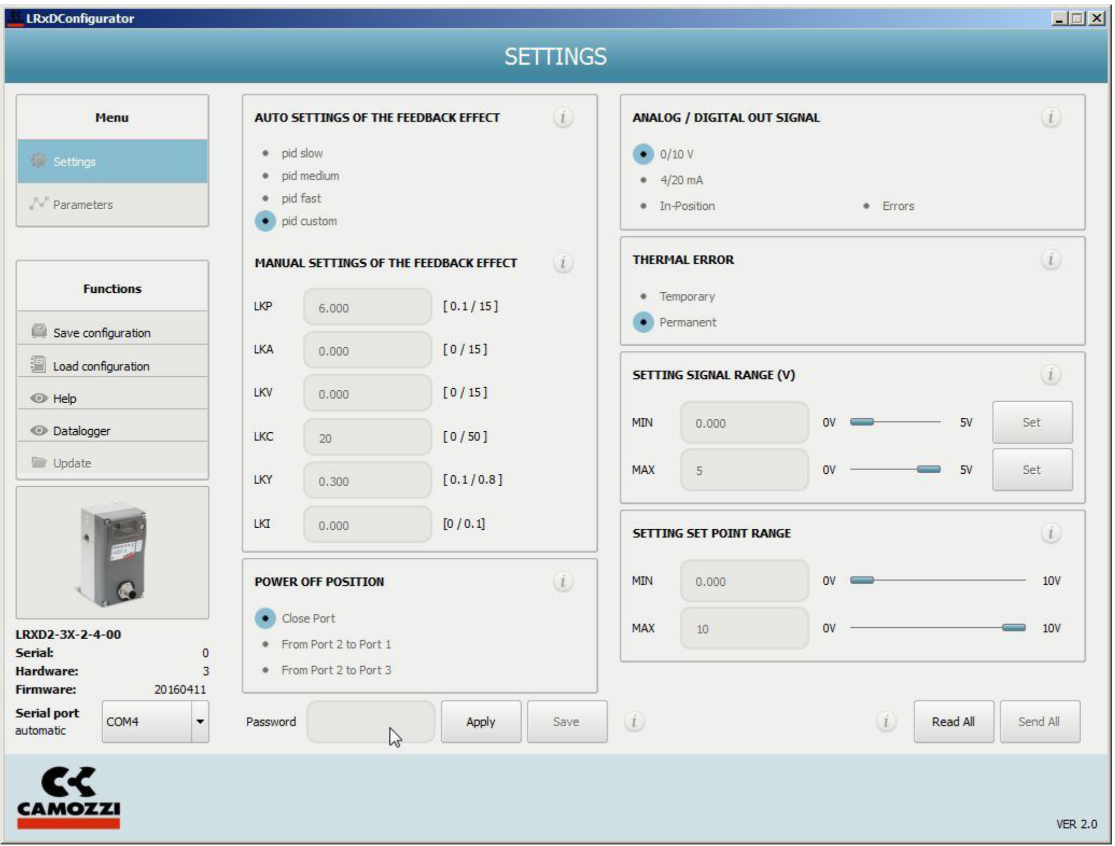


Figure 3.39: LrxdConfigurator Screen

the K parameter value by manually writing the value in the relative text box to improve the application's performance: set it to "*PID custom.*" The K parameter value is limited, and it is not possible to select a value outside these preset ranges (the minimum and maximum values of the K parameter are displayed near the relative text box). For valves with external sensors, the K parameter value must be manually selected.

- In the **"ANALOG/DIGITAL OUT SIGNAL"** frame, the user can choose the feedback signal format that the LRWD series servo valve provides to the control system through pin 8 of the M12 connector: 4-20 mA, 0-10V, in-position, or error. In the first two cases, the measured analog value will be proportional to the external position sensor's value. In the third case, there will be a digital signal indicating the achievement of the goal in real-time. The error tolerance range of the checked cylinder position can be set as a percentage of full scale. If the reference signal exceeds the set range, the digital output will assume an ON value, regardless of the value reached by the controlled variable (position).

- In the **"THERMAL ERROR"** frame, the user can select the servo valve response when a thermal error occurs. This error will occur when the servo valve's temperature is too high, which may be dangerous for the motor inside the LRXD series servo valve. After activating this protection, the motor's power consumption will be limited within a safe value so that the servo valve may lose its flow-related performance.

- In the case of power failure, the user can select the valve position in the **"POWER OFF POSITION"** frame:

  - **CLOSE**: air transit is shut off from 2 to 3 and from 2 to 1.
  - **Port 2 to Port 3**: air transit is opened from 2 to 3.
  - **Port 2 to Port 1**: air transit is opened from 2 to 1.

- In the **"SETTING SIGNAL RANGE"** frame, the user can use the signal from an external position sensor to set the maximum control point (position of the controlled cylinder). The value identifying this point can vary between the minimum and maximum values of the external sensor signal (this range depends on the external sensor signal selected when ordering the valve) and is expressed as an absolute value [V or mA]. The value can be set using the slider on the left side of the box or manually entering the value in the frame on the right side of the box. Using this function, the user can continue to use the entire range of the analog input signal on the LRXD series servo valve to limit the controlled cylinder's stroke within the nominal range.

- In the **"SETTING SET POINT RANGE"** frame, the user can set the reference signal's maximum and minimum points. The value of identifying these points varies between the minimum and maximum values of the reference signal (0÷10 V, or 4÷20 mA). The software "LrxdConfigurator" verifies the consistency of the two values and ensures that the minimum point is not greater than the maximum point. The values can be set using the slider on the box's left side or manually entering two values in the box on the box's right side. This function allows the user to limit the reference signal range within the nominal range while controlling the spool angle's entire range on the LRXD series servo valve. Therefore, the two maximum positive and negative values of the actuator position controlled by the LRWD series servo valve are associated with the set minimum and maximum reference signal values.

- The instruction **"READ ALL"** allows the user to update the valve's current configuration with the settings already set in the valve.

- The instruction **"SEND ALL"** allows the user to sent the setting currently shown on the software to the LRXD series servo valve.

PID Controller Settings

Starting from the fact that the optimum system response must be fast, stable, and accurate, the table below gives some general consideration about regulation:

Increase of	**Response velocity**	**Response stability**
Kp (proportional)	Increase	Decrease
Kv (velocity)	Decrease	Increase (*)
Ka (acceleration)	Decrease	Increase (*)
Ky (Prop. compensation)	Decrease	Increase
Ki (integral)	Increase	Decrease
*High gains of Kv and Ka can generate system instability		

Table 3.15: PID controller setting LRXD servo valve

When the parameter increases, the response velocity increases but the stability decreases. In general, the best situation is to have a controller slow but stable and not a controller fast but unstable.

LKY proportional compensation gain: When the rod cylinder position is close to the target position, the LKY gain directly modifies the control and multiplies the control action. This gain is mainly used to avoid overshoot of the system when the mass is to be moved. The LKY gain essentially allows the user to interrupt the rod's movement with a larger or smaller desired value. The value of LKY gain can be between 0.1 and 0.8 [25].

3.6.6 Servo valve for flow control (LRWD2)- Slave Valve

The servo valve LRWD2 works as a slave-valve controlled by the Master valve LRXDS through a dedicated signal. As the Fig 3.40 shows, the input reference signal comes from the LRXR valve as a 0-10V command signal.

The LRWD series servo valve's working mode is as follows: when the reference signal is lower than 50%, the servo valve joins connections 3 and 2 so that air can pass between the two pneumatic (Fig 3.41 right). Suppose the reference signal is higher than 50%, then the servo valve connects port 1 and port 2, so that air can pass between the two pneumatic ports (Fig 3.41 left).

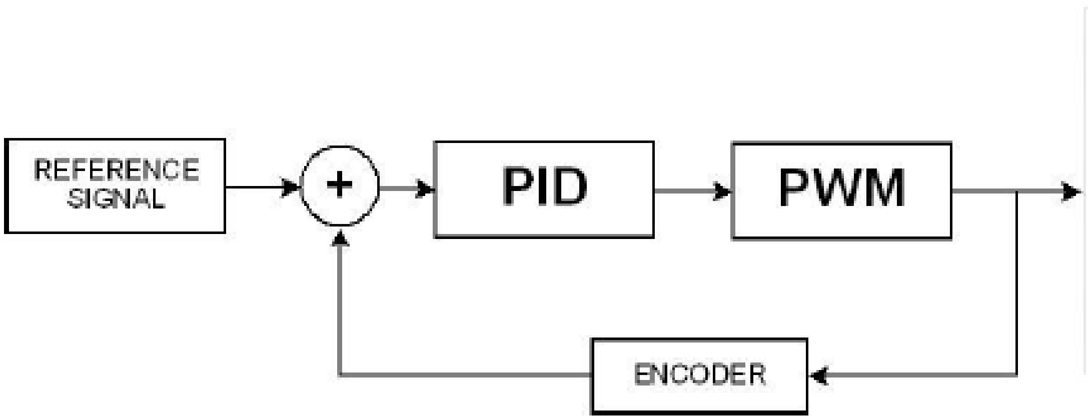


Figure 3.40: Electrical circuit of the LRWD valve [24]

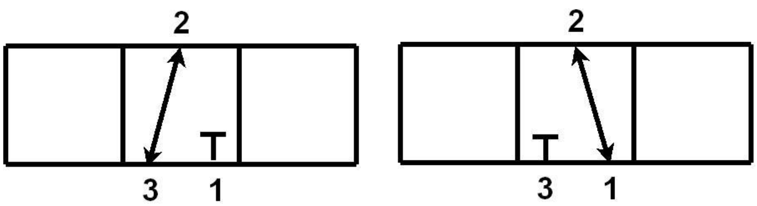


Figure 3.41: Working Principle Servo valve for flow control (LRWD2) [24]

Configuration of the LRWD series servovalve [26]

- The first steps are the same as in the LRXD valve. Whether it is the same software, there are some differences, and the user can notice them when the LRWD is connected. Fig 3.42 2 shows these changes.
- For the LRWD series servo valve, the **"AUTO SETTING OF THE FEEDBACK EFFECT"** and **"MANUAL SETTING OF THE FEEDBACK EFFECT"** frames are disabled. The value shown here will not have any influence on the operation of the LRWD series servo valve.
- In the **"ANALOG OUT SIGNAL"** FRAME, the user can select the feedback signal format that the LRWD series servo valve provides to the control system through pin 8 of the M12 connector: 4-20 mA or 0-10V. In both cases, the measured value is proportional to the angular position of the spool
- The user can set the maximum control point (for the LRWD series servo valve, the spool's angular position) in the **"SETTING FLOW RANGE"**

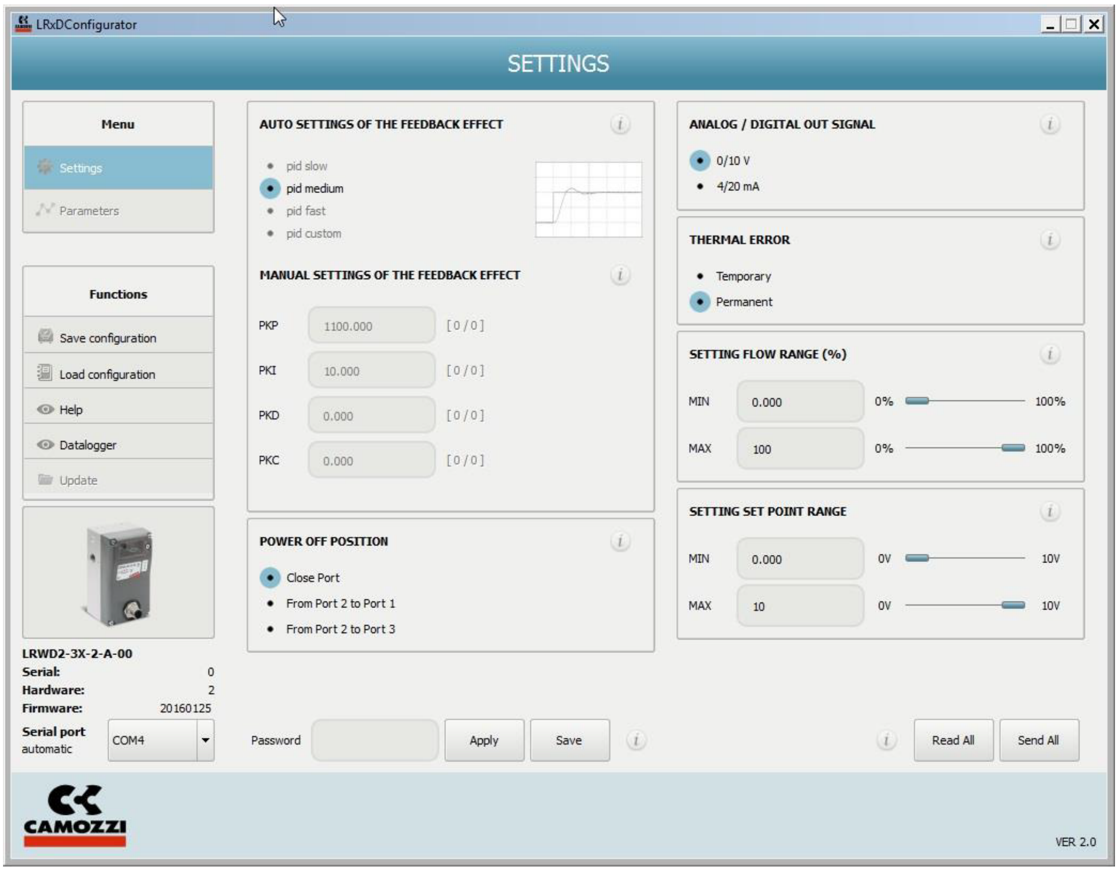


Figure 3.42: LrxdConfigurator Screen when LRWD is connected

frame. This point reads values that vary from 0 to 100 and is expressed as a percentage of the full scale. The value can be set using the slider on the left side of the frame or manually entering the value on the frame's right side. This function allows users to limit the LRWD series servo valve's flow rate to a value lower than its maximum flow rate while using the analog reference signal covering the entire range to control the LRWD series servo valve. Therefore, the reference signal's minimum and maximum values are related to the maximum point controlled in the corresponding flow direction (from 2 to 1 and 2 to 3). For example, if the reference signal type of the maximum control point is 0-10V, then the maximum control point is set to 80. When the reference signal is 0V, the spool moves from 2 to 80% of the final position of 1, and if the reference signal 10V, the spool will move. If the spool's final position is moved from 2 to 3, the spool moves to 80%.

- The **"THERMAL ERROR"**, **"POWER OFF POSITION"**, **"SETTING SET POINT RANGE"**, **"SEND ALL"** and **"READ ALL"** frames are already explained in the LRXD valve configuration. The functioning is the same.

3.7 Series PR precision regulators with manual override

PR series precision pressure regulators are ideal for applications that require precise and stable air pressure control. The working principle of using multiple diaphragms allows the PR series to respond even under the smallest pressure changes that may occur during use.The PR104 and its ISO symbol is shown in Fig 3.43[25].

Figure 3.43: Pressure Regulator and ISO symbol[25]

Moreover, the PR104 is used with the LRXD servo valve in two occasions:

- When the stroke movement is controlled in a backward direction, the master servo valve (LRXD) must be connected to the rear cylinder chamber. And in the front chamber is connected to the PR104 to generate the pneumatic spring.
- When the stroke movement is controlled in a forward direction, the master servo valve (LRXD) must be connected to the front cylinder chamber and the PR104 to the rear chamber.

Chapter 4

PLC programming

The PLC, Programmable Logic Controller, used for the activity and already presented in Section 3.2 of this document, is produced by Siemens with the code S7-1200. The programming was carried out using the TIA Portal v.16 software, supplied by the manufacturer and already described in Section 3.3. Next, the structure and function of the code written by the candidate are analyzed.

The PLC, Programmable Logic Controller, used for the activity and already presented in Section 3.2 of this document, is produced by Siemens with the code S7-1200. The programming was carried out using the TIA Portal v.16 software, supplied by the manufacturer and already described in Section 3. Next, the structure and function of the code written by the candidate are analyzed.

4.1 Structure of the PLC program

The language used for programming is Ladder Diagram, or LAD according to Siemens terminology. The code for the experiments changes depending on the valve type used and the control algorithm. Regardless of this, some functions remain the same for every kind of test realized, such as:

- Sensor's Reading [FC]
- Square Wave Generation [OB]
- Sine Wave Generation [OB]
- Pulse Generator PWM [FB]
- PID Cyclic interrupt [OB].

4.1.1 Sensor's Reading [FC]

This Function (Fig 4.1) receives as an input the word %IW3 ("X_FB_CX") coming from the Camozzi CX06 Module, which gets the feedback coming from the 6PF Series Positioning Feedback cylinders. The ROL block is needed because, as already explained in chapter 3.4, the communication uses little-endian encoding. This means that the first byte is saved as the second byte. This block scrolls the specified number of bits to the left; in this case, four times, every time a point is pushed out of the byte, it will occupy the right side to release s position.

Furthermore, on the second network, the blocks "NORM_X" and "SCALE_X" are used to conditioning the signal. The "Normalize" command normalizes the already roled "X_ANALOG" input variable value by expressing the variable on a linear scale. The MIN and MAX parameters are used to define the limits of a series of values reflected on the scale. The MIN and MAX values were set manually. When the cylinder piston is fully retracted, the minimum value is 53, and when the piston is fully extended, the maximum value is 4058. If the value to be normalized is equal to the MIN input value, 53 in this case, the output returns the value 0.0. If the value to be normalized is equal to the MAX input value, 4058, the output returns the value 1.0. Depending on the value's position to be normalized in this range of values, the result is calculated and deposited as a floating-point number in the "TEMP" variable.

Later on, the "Scale" instruction scales the input value by mapping it into a specific range of values. When executing the "Scalar" instruction, the floating-point number in the input is scaled to the range of values defined by the MIN and MAX parameters. This range is 0-20, which represents the total stroke of the piston in centimeters. The result of the scale is an integer written in the "X_FB" variable, which is the piston's position, giving the system a feedback signal.

4.1.2 Square Wave Generation [OB]

The block in question (Fig 4.3) is of the OB type and Cyclic Interrupt subtype. These blocks are executed at constant and predetermined time intervals, regardless of the scan cycle duration. They are, therefore, perfect for generating waveforms. The block in question allows generating a square wave signal of selectable amplitude and frequency, with a duty-cycle equal to 0.5. The two timers TON, SW_Timer_1 and SW_Timer_2, necessary for the square wave generation, are inserted on Network 1. The Q output of a TON timer is activated once the time set in the PT input ("SW_period") has elapsed, in this case, equal to a half-period, and is deactivated, with reset of the timer when there is no power supply on the branch

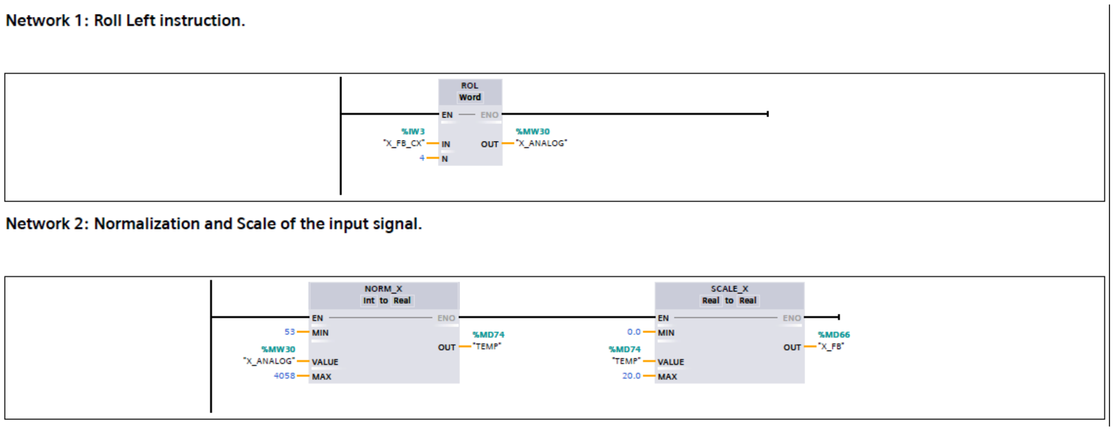


Figure 4.1: Sensor's Reading [FC]

is placed. The Fig4.2 below shows the trend of the two timers' signals when the "SW_period" variable is equal to one second.

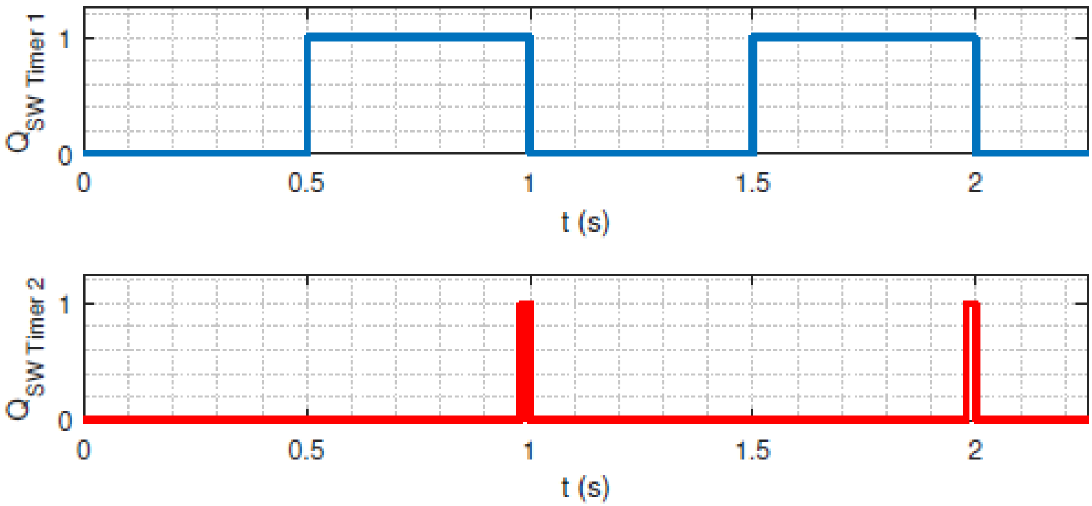


Figure 4.2: Outputs Q in the generation of a square wave with a period of 1s

At the start of the routine, the second timer's Q signal is absent; consequently, the first is powered. After a half-period has elapsed, the Q output of SW_Timer_1 is activated so that also the timer SW_Timer_2 begins to operate. After another half-period, the second's output Q is activated, cutting off the power supply to the first. A chain reaction is thus generated, leading to the disappearance of the Q signal of SW_Timer_1, to the consequent deactivation of SW_Timer_2, and finally to the disappearance of the Q signal of the latter. At this point, the whole procedure repeats itself from the beginning. The last two code segments allow updating the reference in the piston position, which varies between a minimum saved in the "SW_POS_DOWN" variable and a maximum saved in the "SW_POS_UP" variable.

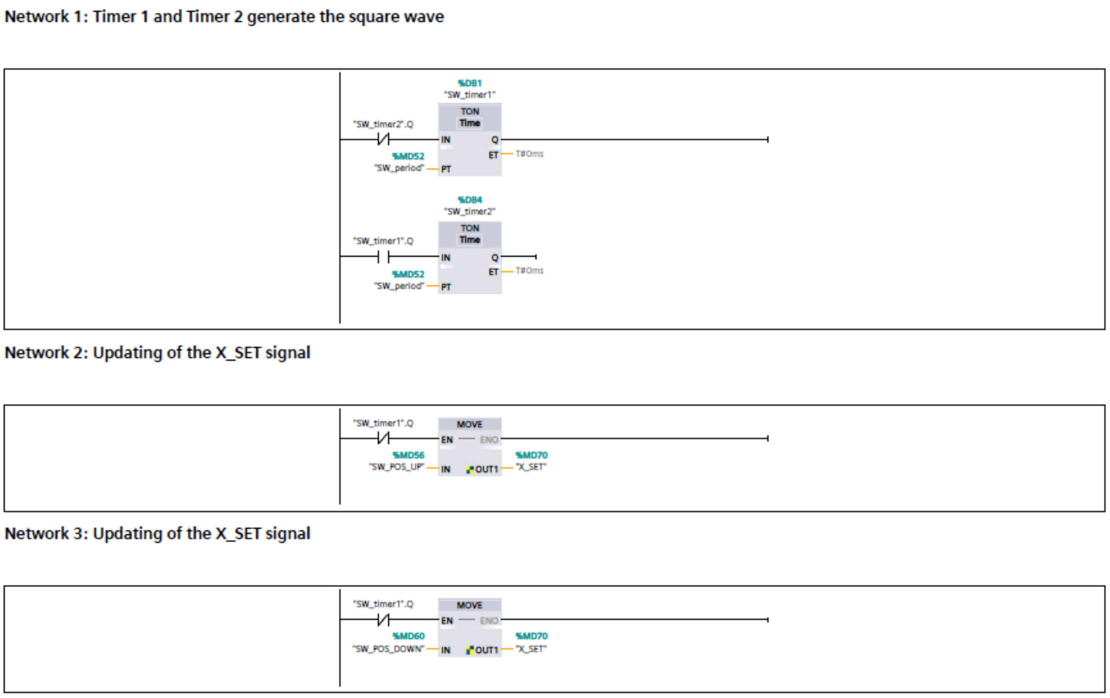


Figure 4.3: Square Wave Generation [OB]

4.1.3 Sine Wave Generation [OB]

The Sine Wave Generation block shown in Fig 4.4 , like the previous one, is of the "Cyclic Interrupt" type and allows the piston setpoint to be varied according to a sine wave of selectable amplitude and frequency and equation:

$$X\_SET = Offset + A \times Sin(\omega t) \tag{4.1}$$

On Network 1 there is a timer that restarts every time a period equal to a period of the sine wave in question passes. Next, on network 2 the product (ωt) is calculated, and the conversion from ms to s is carried out. Furthermore, on network 3, the product $A \times sin(\omega t)$ is calculated and then add the Offset. Finally on the last network, the X_SET variable is updated according to the above equation.

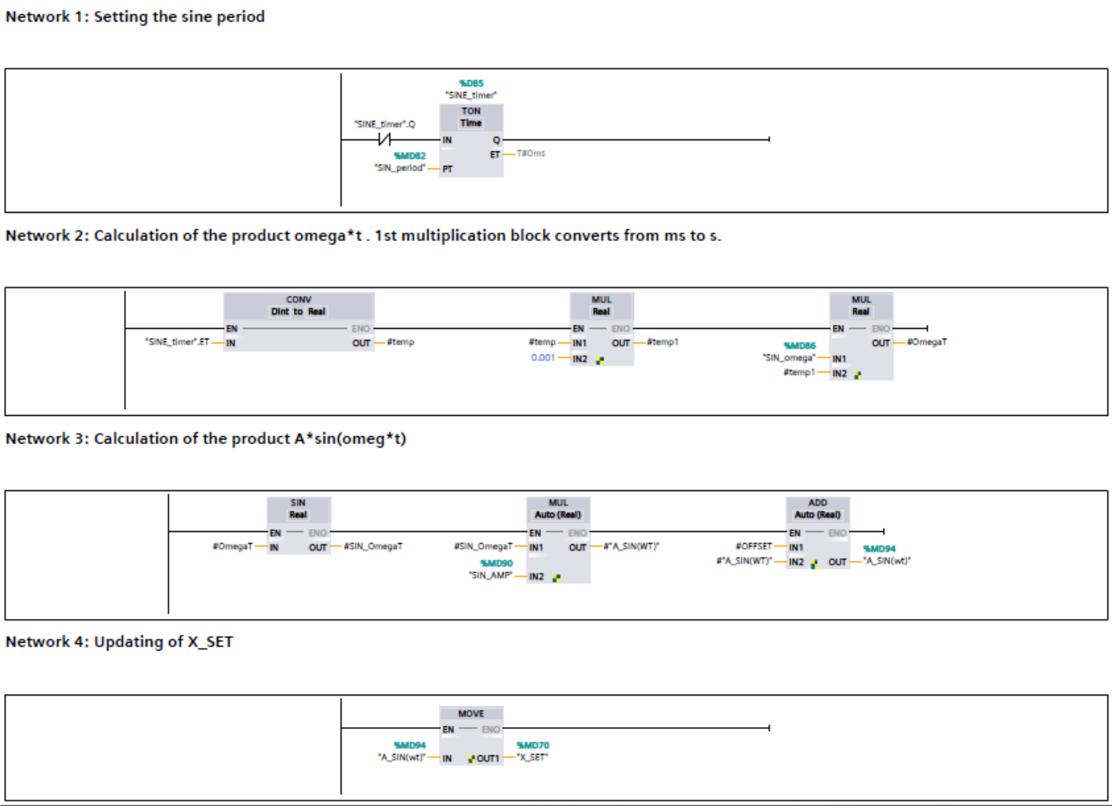


Figure 4.4: Sine Wave Generation [OB]

4.1.4 Pulse Generator PWM [FB]

The PWM signal is a pulse sequence with a fixed frequency and amplitude and a variable pulse width. There is only one pulse of fixed size in each PWM cycle. However, the width of the pulse varies from pulse to pulse following the modulation signal.

The duty cycle is the amount of time that a digital signal is "active" relative to the signal period. The duty cycle is usually given as a percentage. For example, a perfect square wave with equal high time and low time has a duty cycle of 50%. Fig 4.5 shows a representation of a duty cycle. The calculation of the active time obey the following equation [29]:

$$T_{on} = (\frac{Period}{100}) \times DutyCycle \tag{4.2}$$

$$T_{off} = (\frac{Period}{100}) \times (1 - DutyCycle) \tag{4.3}$$

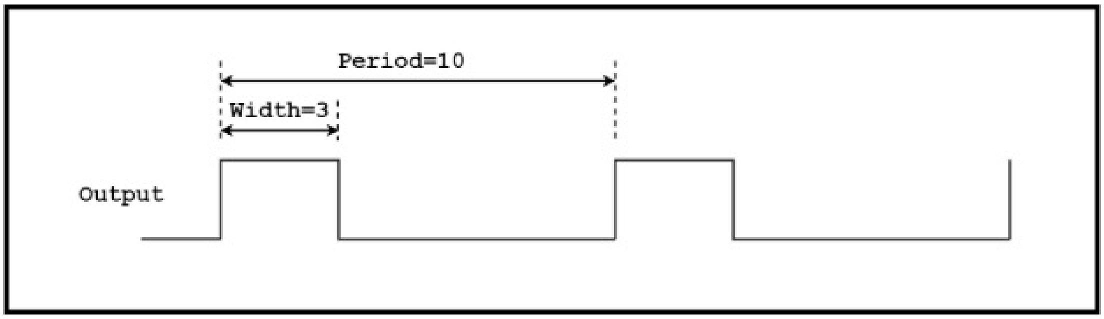


Figure 4.5: Representation of the concept of duty cycle

A Function Block is created for generating a PWM signal. It should be noted that it is better to create a Function Block FB and not an FC function as it could be helpful to have different instances with different values of the Period "T" and the duty cycle.

Network 1 is in charge of calculating the maximum error possible to have. It compares the setpoint entered by the user. If it is higher than $10cm$ (half of the total stroke), the maximum error possible is the setpoint itself; but if the setpoint is lower than $10cm$, the maximum error is equal to $20 - X_{SET}$. For example, if the setpoint is 12 cm and the piston is at 0 cm, the maximum error is 12 cm, but if the setpoint is 5 cm and the piston is at end stroke, what is 20 cm, the maximum possible error is $20cm - 5cm = 15cm$.

Network 2 is in charge of calculating the T_OFF of the duty cycle. The duty cycle is proportional to the error, so the first step is to calculate the error $U = X_{SET} - X_{FB}$. Since what is needed is time, this error U must be a positive value, so the absolute value is obtained. Then, the error is normalized between 0 and MAX error to finally be scaled between 0 and the maximum period defined by the user (to achieve good performance a period not lower than 750ms is suggested by the valve's user manual). Network 1 and Network 2 are shown in Fig 4.6

Network 3 (Fig 4.7)is in charge of generating the PWM. The timer "T3" drives the signal period, which means that the signal will repeat every "T" millisecond. Timer "T4" generates the time the signal remains inactive, which means that the signal will last OFF "T_ON_24" milliseconds. The last branch has the normally closed contact output of the Timer "T4," which activates the coil "ON." The time this coil remains active is the active time of the PWM signal.

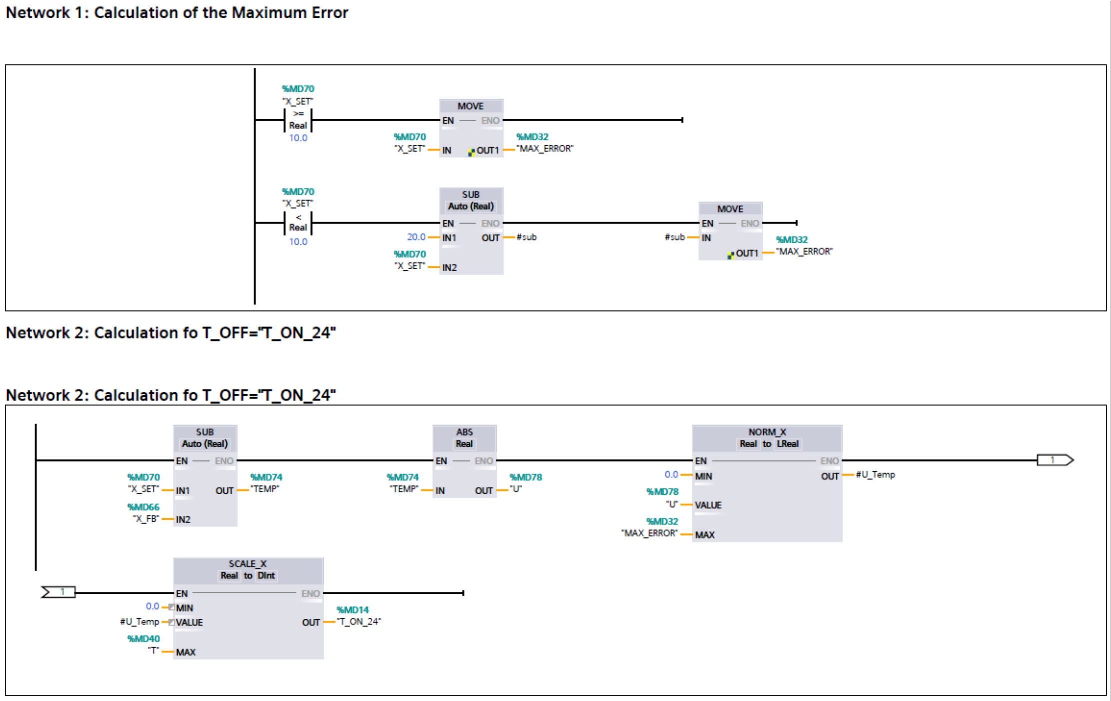


Figure 4.6: Network 1 and Network 2 of the PWM generator [FB]

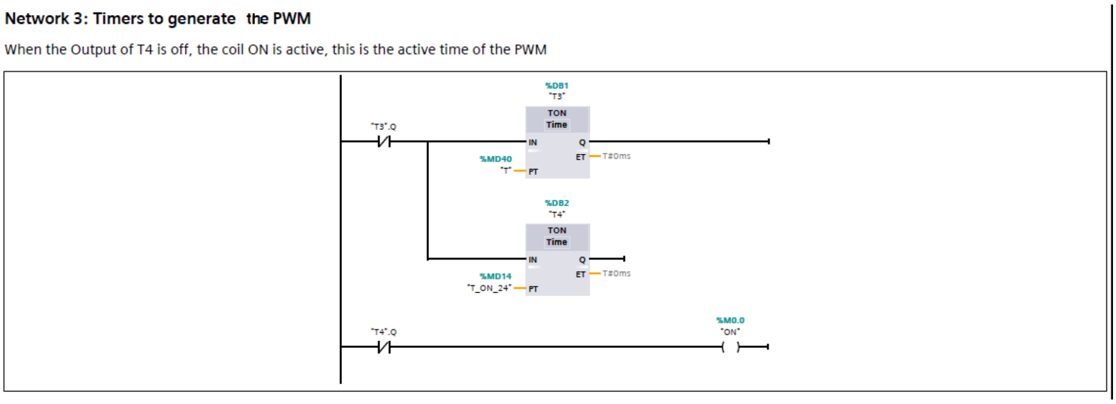


Figure 4.7: Network 3 of the PWM generator [FB]]

4.1.5 PID Cyclic interrupt [OB]

The PID block is used in various ways in this thesis. For example, when working with the 5/3 valve, the PID controller is implemented inside the PLC program and, given the setpoint and the feedback coming from the sensorized cylinder, the digital valve is actuated using the PWM technique or when working with the proportional valves, the output of the PID is the input of the driver, which processes this signal internally.

Before analyzing the block's various segments, the procedure for building a PID controller inside a Siemens PLC is briefly summarized. The first step consists of creating a Cyclic Interrupt type block, inside which the actual PID network will be inserted. The frequency with which the interrupt is called coincides with the frequency with which the PID algorithm is calculated. The latter is all contained within a PID Compact component, which is part of the Technologic Instructions family. Once the interrupt has been created, the PID Compact block must be positioned inside it, and it is possible to set its operating parameters, the most important of which is the sampling period. It determines how many times the algorithm is executed before the outputs are updated. In other words: set a one-millisecond interval for the interrupt and set this property to four milliseconds; the PID algorithm is executed four times before the PLC output is updated. This technique allows not to clog the output bus and at the same time allows the system to respond to the command signal.

The slower a system responds to a change in the setpoint, the longer the sampling period can be. In the case in question, the PID algorithm is executed every millisecond, and the updating of the outputs takes place every two.

Once the piston position has been read and converted, it is sent, together with the desired set point, to the PID block, which compensates for the error between the two inputs and saves the result in a temporal variable, called "temp."

The PID output then enters an ABS block, with which the absolute value of the input signal is obtained and stored in the "PID_Output" variable. The ABS function is executed because the output of the PID is different in several ways in the thesis. For example, when the PID output commands a PWM, this output has to be converted into a time variable, so there is no negative time. Another example is that the analog module's output is limited to the range 0 - 10 V, and the drivers of the proportional valves do not accept negative command signals.The PID Cyclic interrupt [OB] is shown in the Fig 4.8.

4.1.6 Configuration Windown PID Compact

The user can configure the PID Compact in the Configuration Window. The basic settings, Process Value Settings, and Advanced Settings.

- **Basic Settings:**on this window, the user can configure properties as the physical quantity, the control logic, and the start-up behavior after reset.

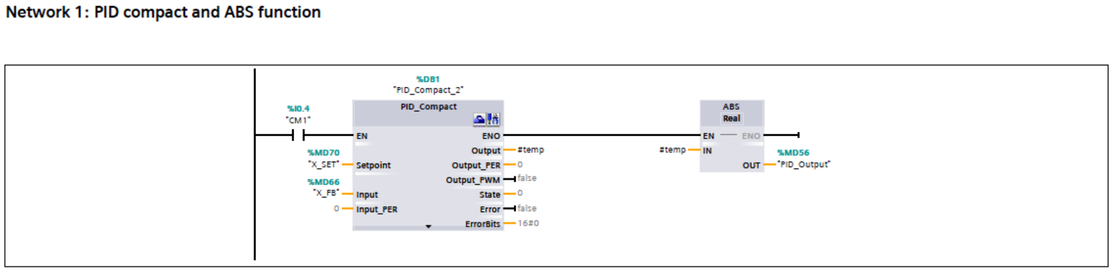


Figure 4.8: PID Cyclic interrupt [OB]

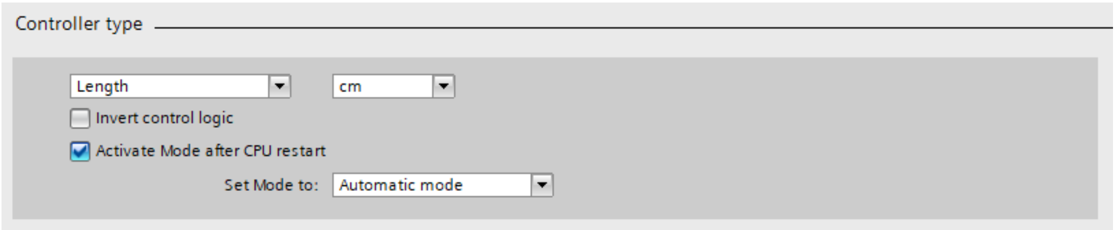


Figure 4.9: Control Type Windonw

As shown in the Fig 4.9, the chosen physical quantity is Length, and the measurement unit is centimeters because a positioning control is realized. The process value and setpoint and will be displayed in this unit.

In the input/output parameters (Fig 4.10), the user can choose the input and output type. In this case, the entry "Input" is selected from the drop-down list "Input," then the user must write the name of the variable in which the processed process value is saved. PID Compact offers three output values, and the type is chosen depending on the application and the actuator.

- Output PER: The actuator is triggered by an analog output and controlled by a continuous signal. For example, 0-10V, 4-20mA.
- Output PWM: A digital output controls the actuator. Pulse width modulation creates minimum ON and minimum OFF times.
- Output: The user program's output value needs to be processed, for example, because of nonlinear actuator response.

- **Process Value Settings**: The user must specify appropriate absolute upper and lower limits for the process value as the controlled system's limit values.

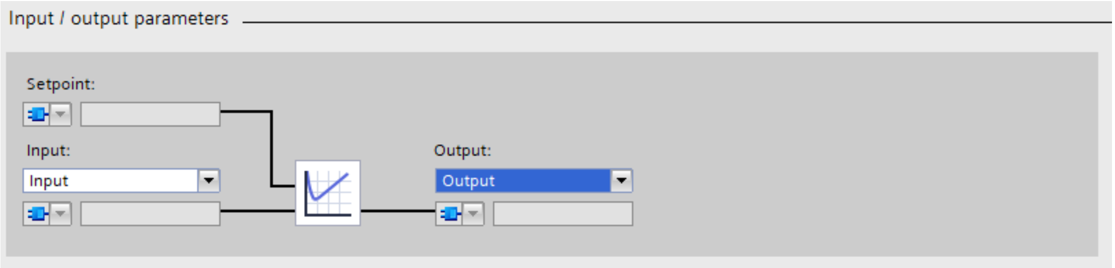


Figure 4.10: Input/Output Parameters

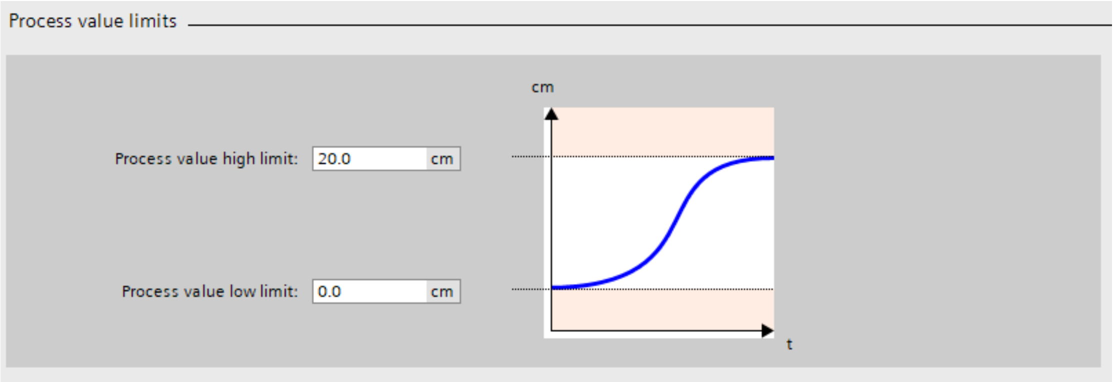


Figure 4.11: Process Value Limits

Once the process value violates these limits, an error will occur (ErrorBits = 0001h). When the process value limit is violated, the tuning is canceled.

In this case, as shown in Fig 4.11, the high limit is set at 20 cm and the low limit at 0 cm because the total stroke of the cylinder is 20 centimeters, so the process value will always stay inside that range.

- **Advanced Settings**:
  - **Output value limits:** In this configuration window (Fig 4.12), the user configures the absolute limit of the percentage form's output value. The absolute output value limit will not be violated in manual mode or automatic mode. If an output value that exceeds the limit is specified in manual mode, the practical value is limited to the CPU's configured limit.
  - **PID parameters:** The parameters are shown in the "PID parameters" configuration window (Fig 4.13 . The parameters will adapt to the controlled system during the controller tuning but also can be changed manually.

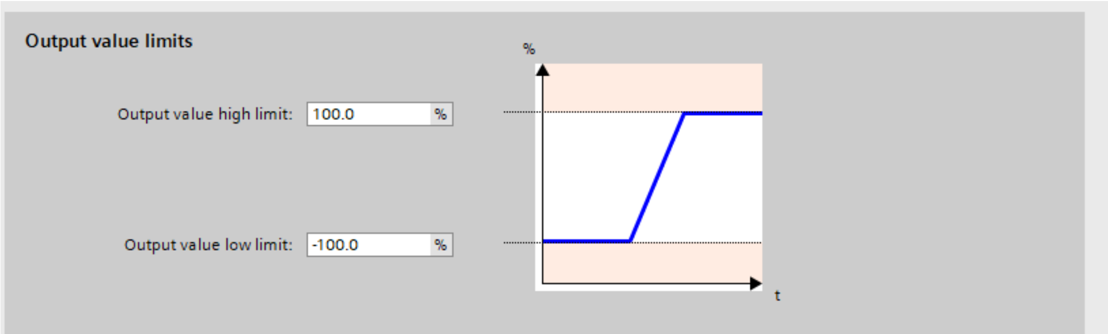


Figure 4.12: Output Value Limits

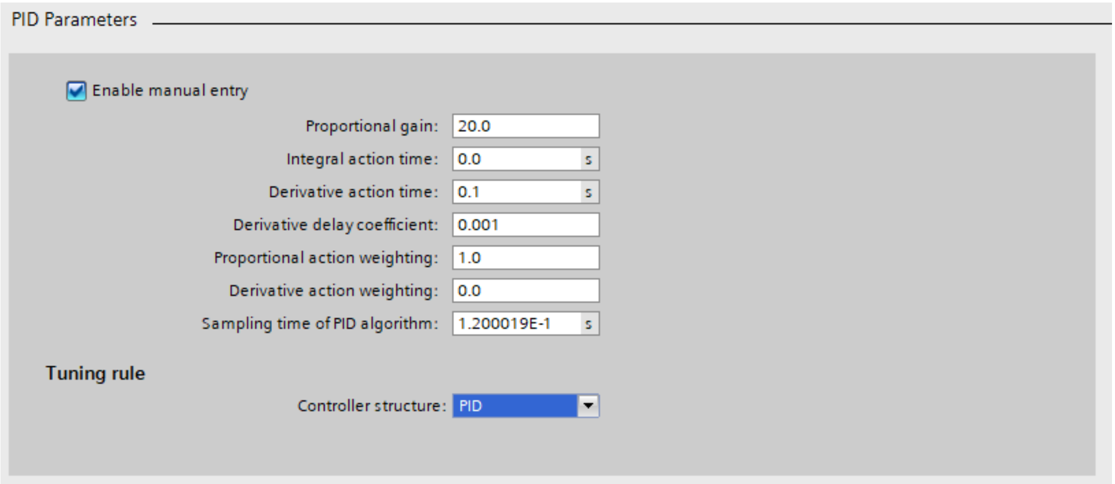


Figure 4.13: PID Parameters

In the "Controller structure" drop-down list, the user can choose the type of controller that better matches the requirements, and the options are PID and PI.

4.2 Structure of the Compact PID controller

As mentioned, the PID controller available within the TIA Portal takes the name of PID Compact. The control law used within it has the following structure:

$$y_{PID} = K_p[(w_p x_{SET} - x_{FB}) + \frac{1}{T_i s}(x_{SET} - x_{FB}) + \frac{T_d s}{\tau T_d s + 1}(w_d x_{SET} - x_{FB})] \quad (4.4)$$

Where x_{SET} and X_{FB} indicate a generic reference signal and the corresponding feedback coming from the transducer. In terms of time constant, it is very similar to the classic PID controller. Instead of the two coefficients wp and wd, the "weighting coefficient of proportional part" and "weighting coefficient of differential part" respectively. The value range of wp and wd is 0-1, and the purpose is to exclude or cause the proportional and derivative components to intervene in a limited way (if a change in the reference value causes the error input to the controller) rather than the process variable. In terms of simplicity, the coefficients are set to 1 to work with the commonly known PID control law, which becomes as follow:

$$y_{PID} = K_p[(x_{SET} - x_{FB}) + \frac{1}{T_i s}(x_{SET} - x_{FB}) + \frac{T_d s}{\tau T_d s + 1}(x_{SET} - x_{FB})] \quad (4.5)$$

Each term in the equation is explained in the following table.

Symbol	Description
y_{PID}	Output value of the PID algorithm
K_p	Proportional gain
s	Laplace operator
w_p	Proportional action weighting
x_{SET}	Setpoint
x_{FB}	Feedback value (Process value)
T_i	Integral action time
T_d	Derivative action time
w_d	Derivative action weighting
τ	Derivative delay coefficient

Table 4.1: Terms of the PID Compact equation

Thus, it was possible to determine the equivalence between the form implemented within the Siemens PLC and the classical one using k_p, k_i, k_d. The conversion formulas are as follows:

$$K_p = k_p \quad (4.6)$$

$$T_i = \frac{k_p}{k_i} \quad (4.7)$$

$$T_d = \frac{k_d}{k_p} \quad (4.8)$$

$$(4.9)$$

Thanks to this convention, it will be easy to analyze and compare the system's response when these values vary.

Chapter 5

Experimental results

In this chapter, the results obtained for the different configurations of the plant will be presented. The control methods and how they were implemented in the TIA portal software will be briefly explained by describing the ladder language programs. The comparison between the results is discussed in detail in the next section.

5.1 5/3 closed center solenoid valve

The difference between the setpoint and the process output (measured value) is caused by disturbances affecting the plant (process). The controller's role is to eliminate these interferences and keep the plant output (process value) at a predefined value (setpoint).

The way the controller reacts to errors is called control law or control mode. Various control laws are used in industrial applications, mostly on-off control, PID control, or other more advanced laws (fuzzy, neuro-fuzzy, and optimal).

For the configuration of the test bench that uses the 5/3 closed center solenoid valve, the PLC and the Camozzi CX06 module, shown Fig 5.1, three types of control were implemented:

- On-Off Control
- PWM Method
- PWM driven by a PID control

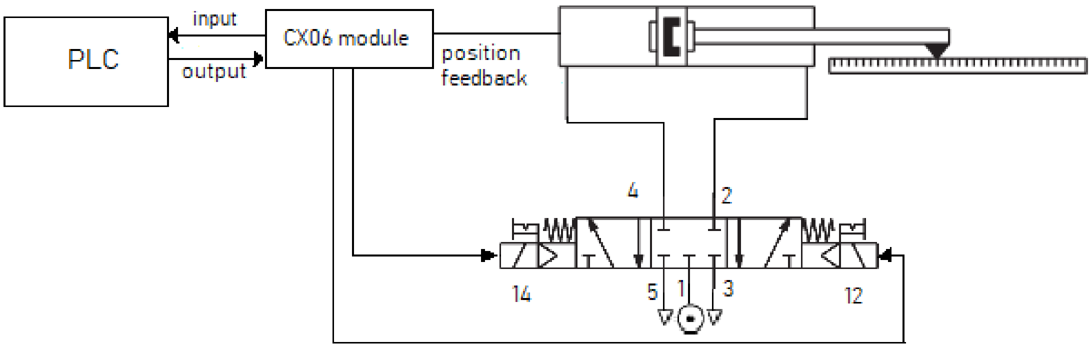


Figure 5.1: Configuration using 5/3 closed center solenoid valve

5.1.1 ON-OFF control

An on-off controller, bang-bang controller, or two-step controller is the simplest feedback control method. The on-off controller easily drives the process value from fully closed, which switches OFF when the error is zero or negative to fully open, which switches ON when the error is positive according to the position of the controlled variable relative to the setpoint value [28].

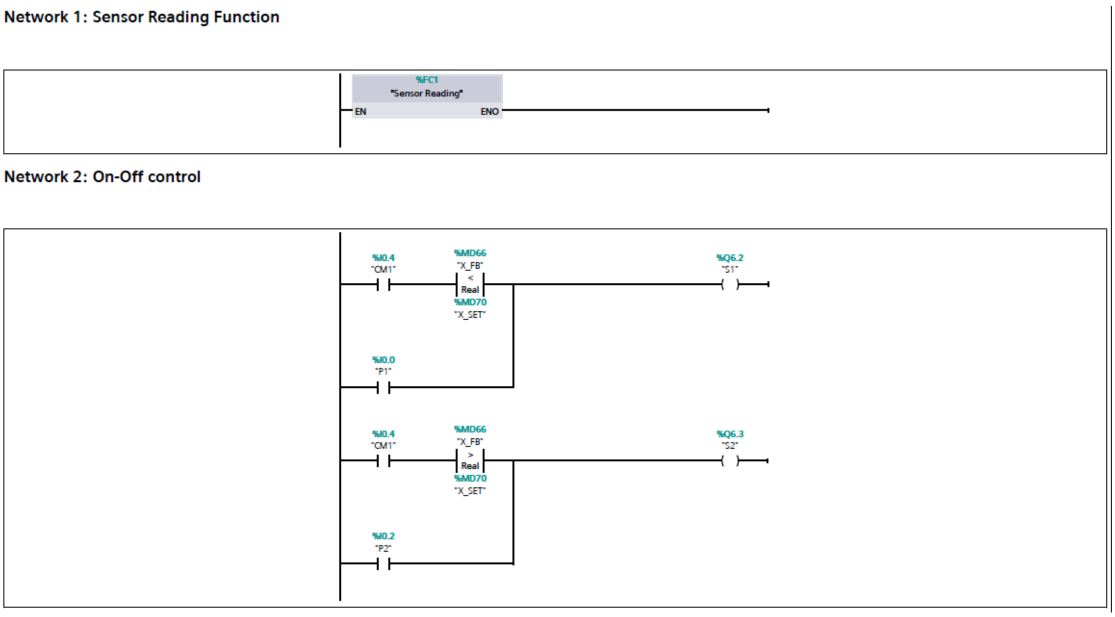


Figure 5.2: Ladder Diagram On-Off controller

Furthermore, Fig 5.2 shows the ladder diagram for the on-off controller made in

TIA portal.As explained in Chapter 4, the first network is in charge of acquiring the sensorized cylinder signal and giving it the proper conditioning to work comfortably. The second network has the logic of the controller. The first branch is in charge of the airflow in the rear chamber, and the second branch is in charge of the pneumatic cylinder's front chamber. The process value stored in the X_FB variable is compared with the setpoint value entered manually by the user and stored in X_SET variable. When the process value has not reached the setpoint, the S1 coil that commands solenoid 14 of the 5/3 valve is activated until the setpoint is reached. On the other hand, if the value goes beyond the setpoint, coil S2 that commands solenoid 12 of the valve is activated.

The Fig 5.3 shows the dynamic position response while tracking a step change in reference signal from 0 mm to 120 mm and its error $u(t) = X_{SET} - X_{FB}$. As seen in the figure, not much can be concluded as the system never stabilizes. This significant error is caused because the type of control is not robust enough to control the valve's airflow. As this flow is huge, the momentum it gives to the pistons is so strong that it never stabilizes.

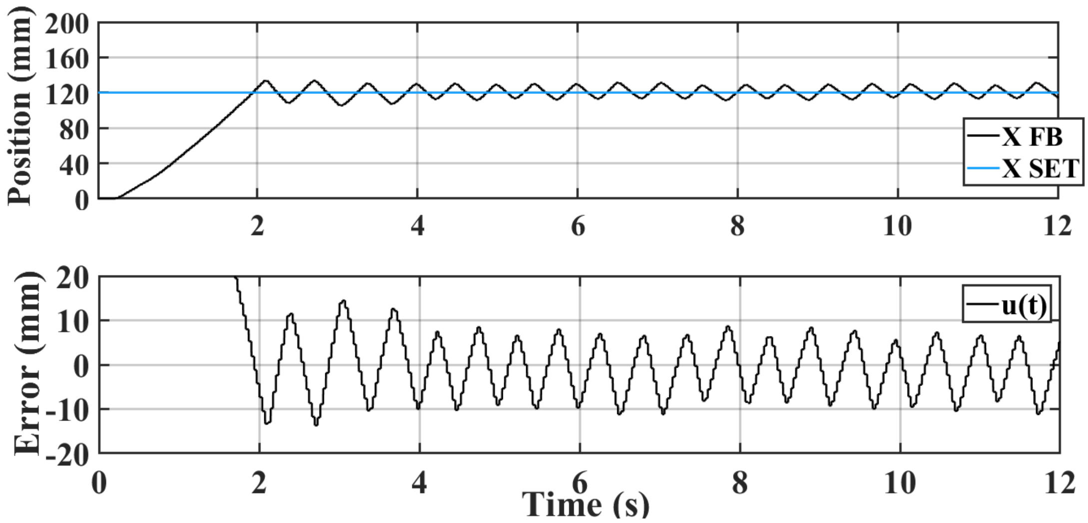


Figure 5.3: Dynamic position response while tracking a step change in reference signal from 0 mm to 120 mm.

5.1.2 PWM Method

Figure 5.4 shows network 1 and network 2 of the program. These blocks are already explained in detail in section 4.

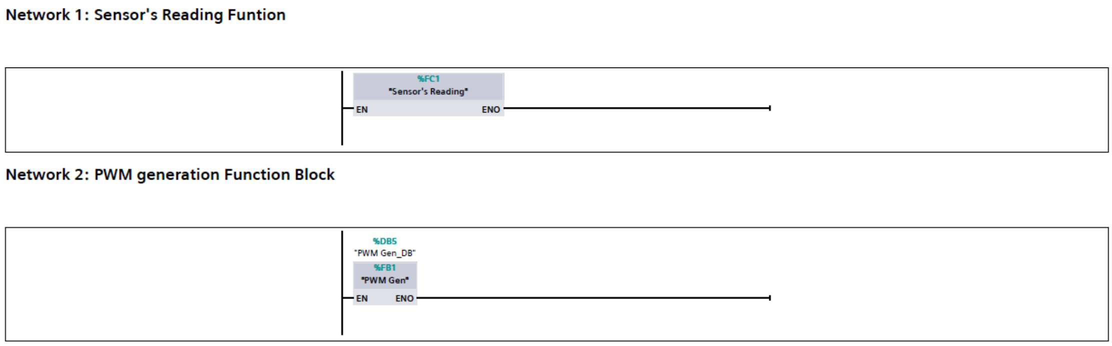


Figure 5.4: Network 1 and Network 2 of the PWM Method program.

Network 3 and network 4, shown in Fig 5.5 are responsible for the control of the solenoids S1 and S2. When the error "U_TEMP" is higher than zero, the solenoid S1 that commands the airflow in the rear chamber will remain active for the time the contact "ON" is close. It is worth it to remain that the contact "ON" represents the active time of the PWM signal generated by the PWM Gen Function Block. On the other hand, the same happens with the solenoid S2, and this will remain active when the error "U_TEMP" is lower than zero.

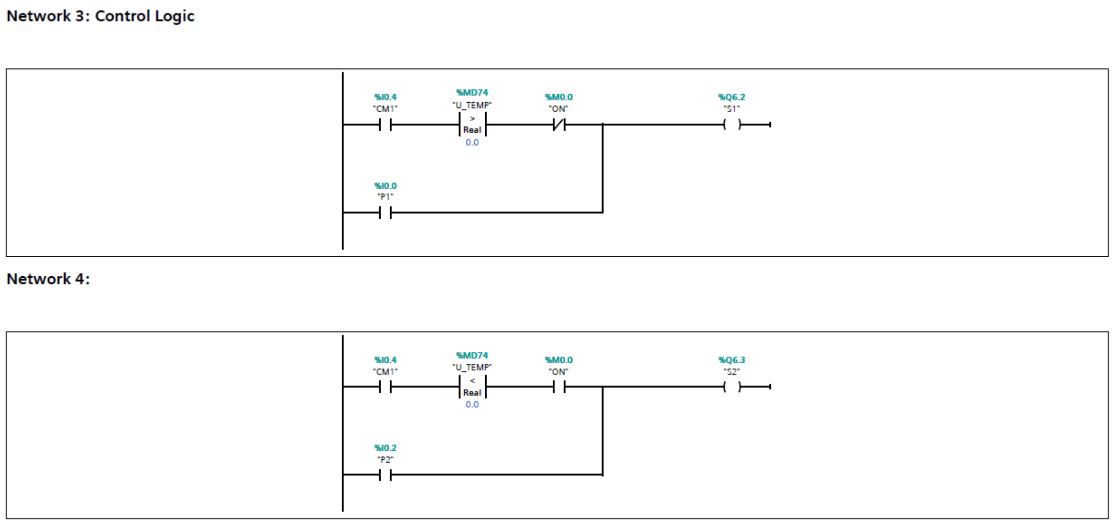


Figure 5.5: Network 3 and Network 4 of the PWM Method program

Figure 5.6 shows the dynamic position response while tracking a step-change in reference signal from 0 mm to 120 mm and period 650, 750, and 850 milliseconds, respectively. As the images show, the behavior is much better than the previous exercise. The system remains stable for an uncertain time, but the difference in pressure between the cylinder chambers means that the piston never stays entirely still. For this reason, the system remains vibrating around the setpoint.

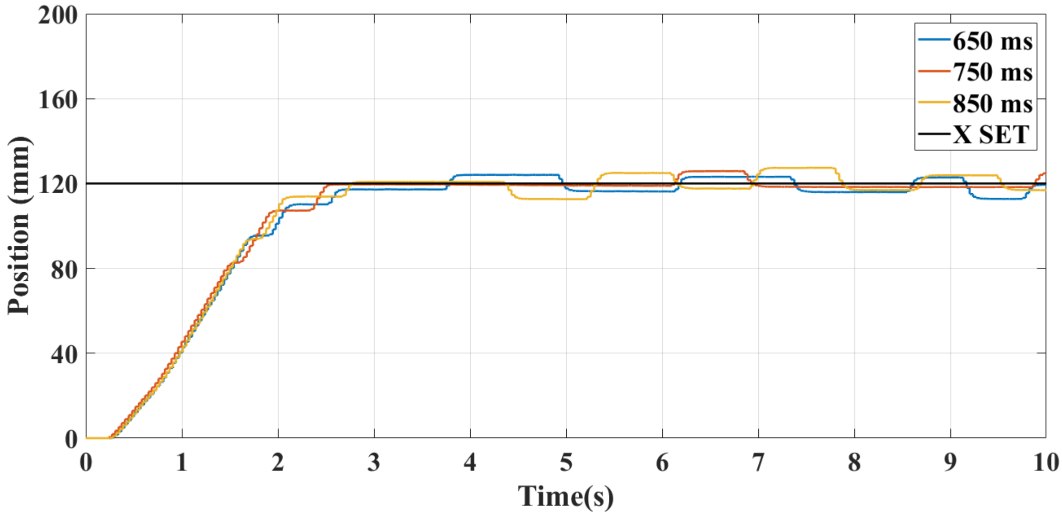


Figure 5.6: Dynamic position response while tracking a step change in reference signal from 0 mm to 120 mm with perido of 650, 750 and 850 ms.

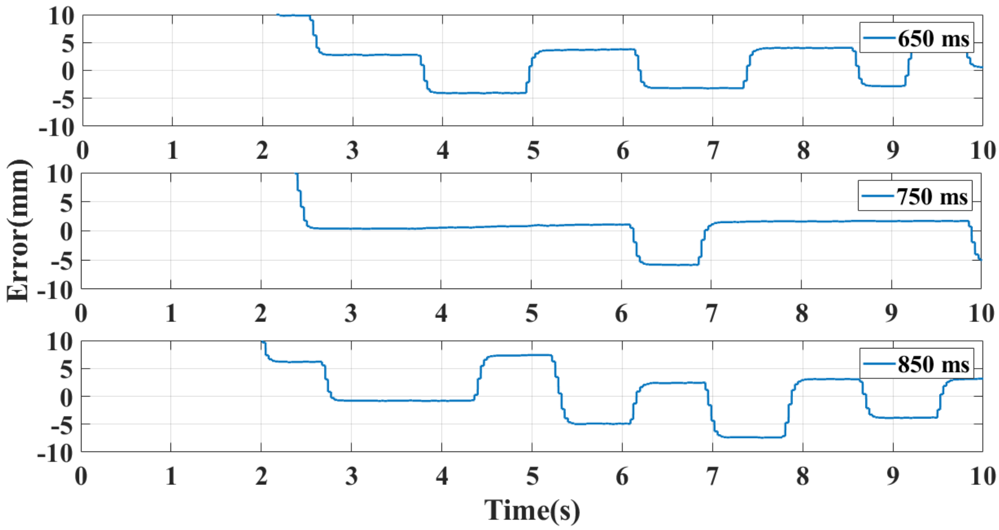


Figure 5.7: Error varying the period.

At this time, it is important to introduce the term "settling time". **Settling time (ts)** refers to the time it takes for the response curve to reach an error band around the setpoint and stay within this band. The error band is usually +-2% or

+-5% of the final value [30]. In this thesis, an error band of 5% was chosen. Table 5.1 , summarizes the results obtained.

	650 ms	**750 ms**	**850 ms**
ts (s)	2.55	2.45	2.30
max error (mm)	7.29	5.87	7.42

Table 5.1: Results PWM Method

5.1.3 PWM driven by a PID controller

An adequately set PID controller will reach this set value as soon as possible and keep it at a constant value. After the output value changes, the process value usually changes only with time. The controller must compensate for this response.

The output value of the PID controller consists of three parts:

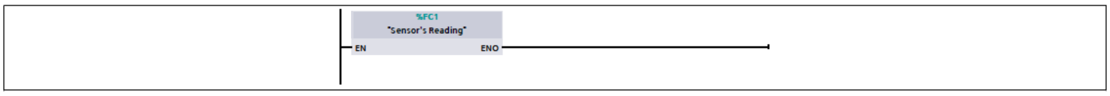
- P (Proportion): the output value is proportional to the difference between the set value and the process value (input value). As the name suggests, this control action is proportional to the error signal. Internally, the proportional action multiplies the error signal by a constant and tries to minimize the system error. When the error is significant, the control action is large and tends to minimize this error [31].

- I (integer): the output value increases in proportion to the duration of the difference between the set value and the process value (input value) to finally correct the difference. As its name implies, this control action calculates the error signal's integral; that is, it corresponds to the sum or accumulation of the error signal. As time goes by, small errors add up to make the integral action ever greater, reducing the error of the process that is in production. The disadvantage of using the integral action is that it adds certain inertia to the system and therefore makes it more unstable.

- D (Derivative): the output value increases according to the rate of increase in the difference between the set value and the process value (input value). Correct the output value to the set value as soon as possible. This control action is proportional to the derivative of the error signal or, in other words, to the speed of the error. When the system or process controlled is moving at high speed towards the desired point (reference value), the system will pass by due to its inertia, producing overshoots and oscillations.

Fig 5.8 shows the first two networks of the program. As in past programs, the first network acquires and conditions the sensor signal. The second network is responsible for generating the active duty cycle for the PWM. Unlike the previous program, the duty cycle is not directly proportional to the error $u = X_{SET} - X_{FB}$; in this case, the PWM depends on a PID controller. The Function Block called "PID_PWM," which gets as input the variable "PID_output" from the PID Cyclic interrupt [OB], explained in chapter 4, and the period "T" which determines the period of the PWM and gives as output the duty cycle proper to command the valves.

Network 1:



Network 2: PID_Output Function Block

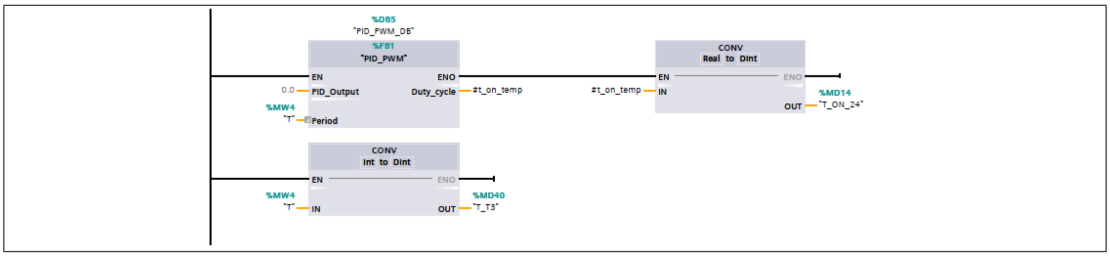


Figure 5.8: Network 1 and 2 PWM driven by a PID controller

Figure 5.9, shows what is inside the PID_PWM [FB]. Firstly the variable "PID_output" is normalized between 0-100 because this variable is needed in percentage format, and then this percentage is multiplied by the period, thus obtaining the desired duty cycle.

Fig 25.10images the PWM generator, which works in the same way as the one explained in the past method.

Fig 5.11 shows the control logic, which works pretty similar to the previously explained. When the error "U_pid" is positive, the solenoid S1 is activated when the contact "ON" remains active. The contact "ON" represents the duty cycle of the PWM. The same happens with the solenoid S2 when the error "U_pid" is negative. When the error "U_pid" is zero, both the coils are deactivated.

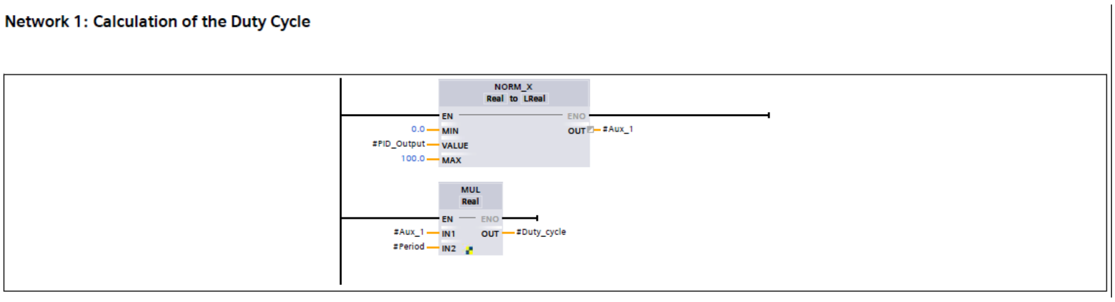


Figure 5.9: PID_PWM Function Block

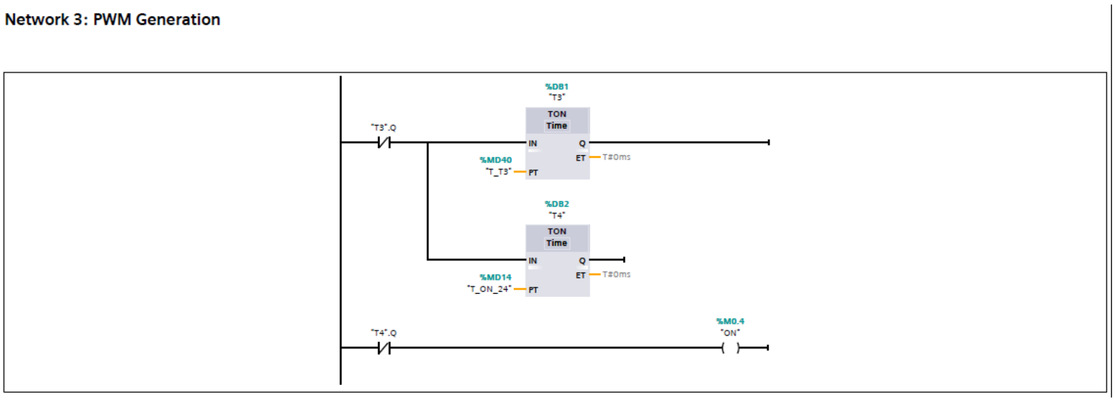


Figure 5.10: Network 3 of the PWM driven by a PID controller programm

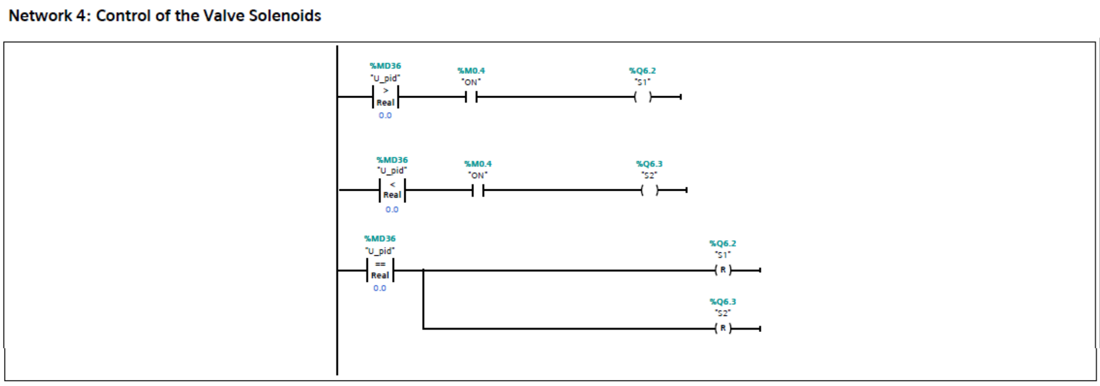


Figure 5.11: Network 4 of the PWM driven by a PID controller programm

The obtained results are shown in the Fig 5.13. The effect of the parameter Kp is considerably notable because by increasing the parameter, the system's response increases, but the stability decreases.Fig 5.13 shows the error $u(t) = X_{SET} - X_{FB}$

for each K_p. It is observable the improvement of the system. The table 5.2 summarizes the results obtained in this experiment in terms of settling time and error.

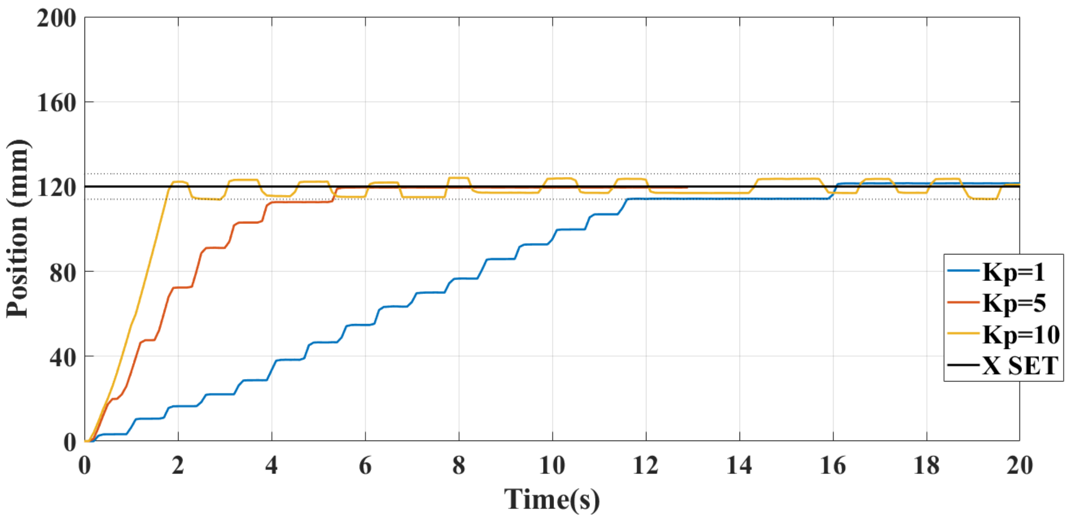


Figure 5.12: Dynamic position response while tracking a step change in reference signal from 0 mm to 120 mm varying Kp parameter

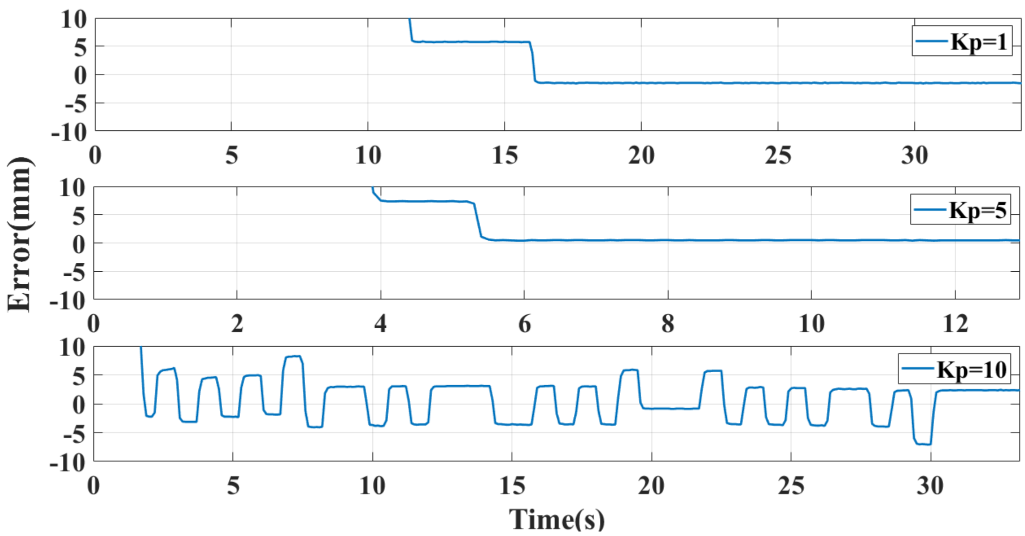


Figure 5.13: Error varying Kp parameter

The Fig 5.14 shows the dynamic position response while tracking a step change in reference signal from 0 mm to 120 mm varying K_i parameter and keeping $K_p = 5$

	Kp=1	Kp=5	Kp=10
ts (s)	11.7	5.3	1.7
error (mm)	1.48	0.45	6

Table 5.2: Results PID control varying KP parameter

constant.Moreover, the addition of an integral action to the system was supposed to be accompanied by a decrease in the steady-state system error, but the response was not as expected, it is shown in Fig 5.15. The table 5.3 summarizes the results obtained.

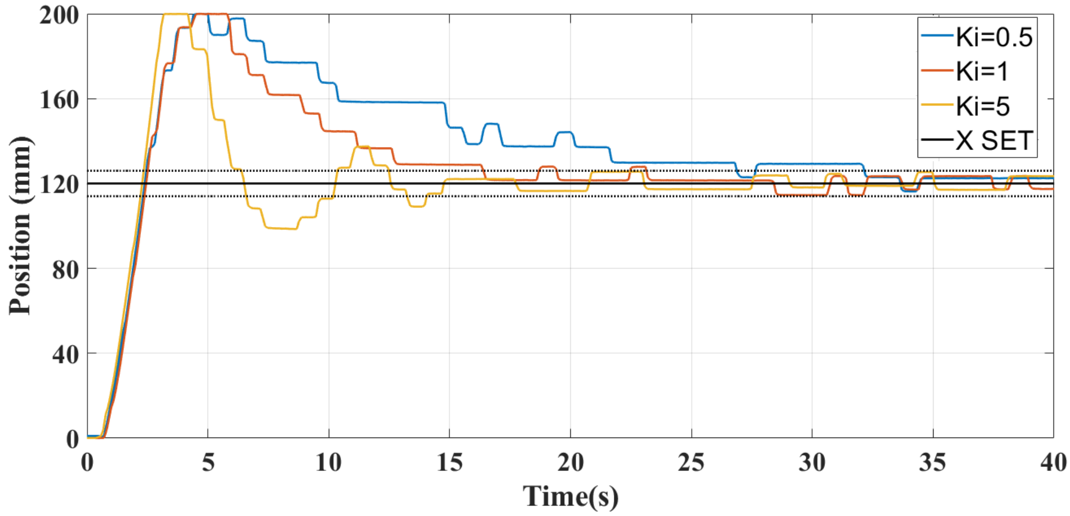


Figure 5.14: Dynamic position response while tracking a step change in reference signal from 0 mm to 120 mm varying Ki parameter

The Fig 5.16 shows the dynamic position response while tracking a step change in reference signal from 0 mm to 120 mm varying K_d parameter and keeping $K_p = 5$ and $K_i = 5$ constant. In addition, it is remarkable the action of the derivative parameter in the system. Compared to the previous results, this one is more stable but a little less fast. On the other hand, the error continues to be similar to the previous results. The steady-state error is shown in Fig 5.17. The table 5.4 summarizes the results obtained.

After seeing the different proportional, integral, and derivative actions of a PID control, a few simple rules can be applied to tune this controller manually. The proportional action is gradually increased to decrease the error and increase

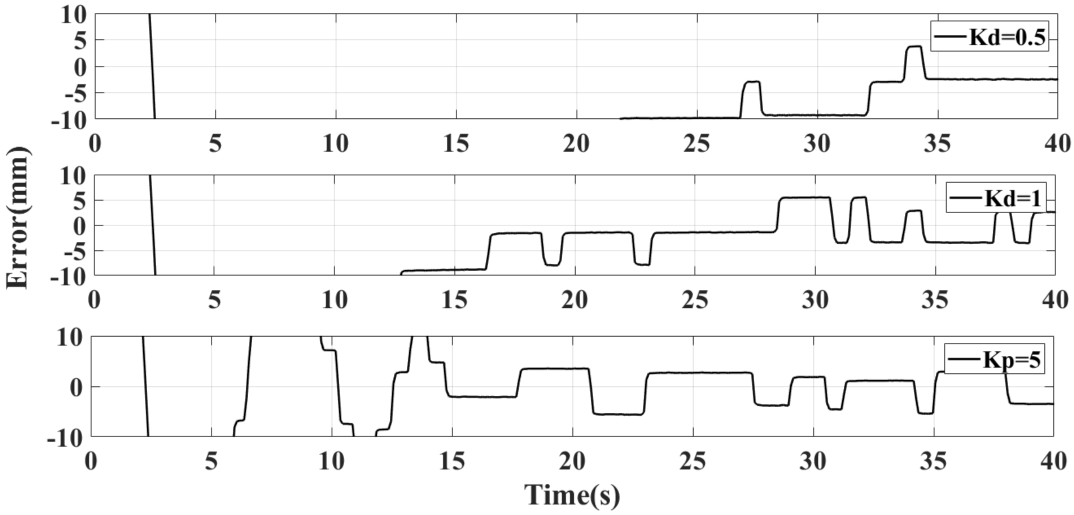


Figure 5.15: Error varying Ki parameter

	Ki=0.5	Ki=1	Ki=5
ts (s)	32.1	23.3	14.1
error (mm)	3.753	5.5	5.58

Table 5.3: Results PID control varying KI parameter

response speed. With $K_p = 6$ a suitable response in speed and error is reached, so the PID is already tuned. Fig 5.18 and the table 5.5 shows the dynamic position response while tracking a step change in reference signal from 0 mm to 120 mm with $K_p = 6$ and the error, respectively.

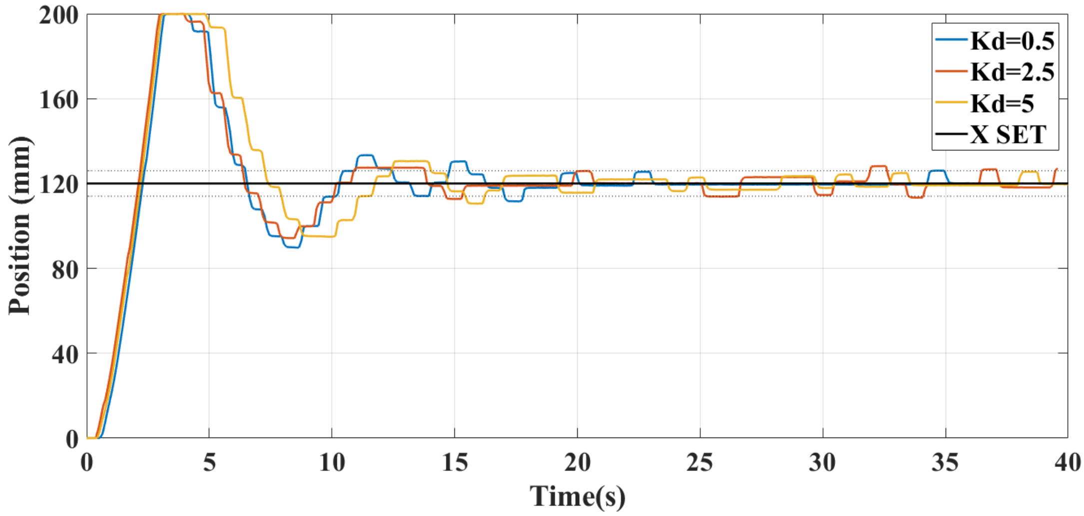


Figure 5.16: Dynamic position response while tracking a step change in reference signal from 0 mm to 120 mm varying Kd parameter

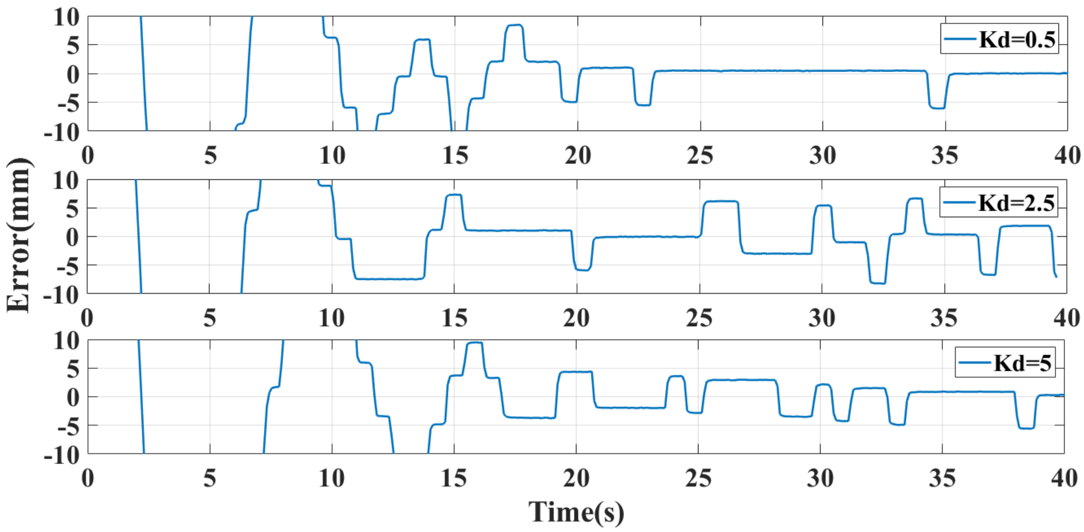


Figure 5.17: Error varying Kd parameter

	Kd=0.5	Kd=2.5	Kd=5
ts (s)	17.87	15.27	16.34
error (mm)	5.4	5.8	5.8

Table 5.4: Results PID control varying Kd parameter

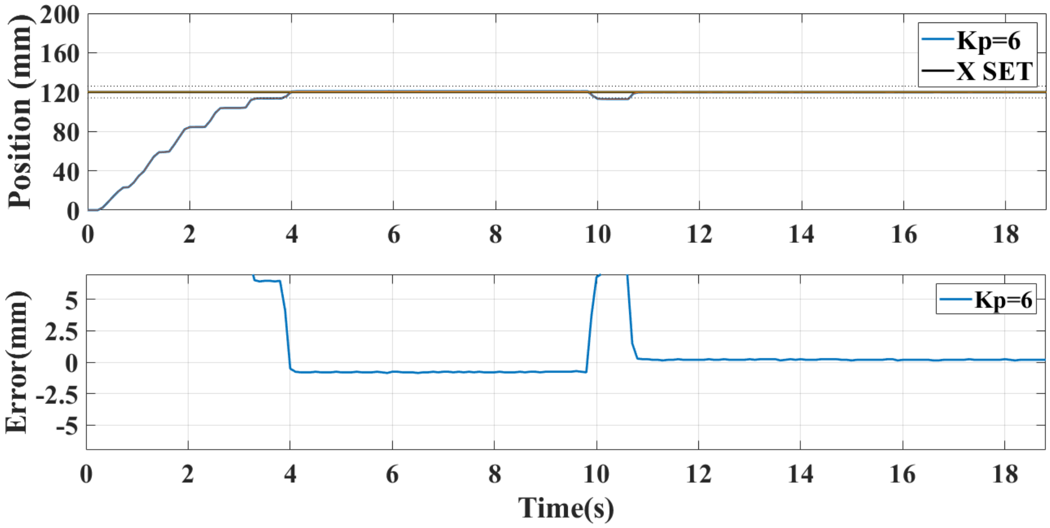


Figure 5.18: Dynamic position response while tracking a step change in reference signal from 0 mm to 120 mm with Kp=6

	Kp=6
ts (s)	3.71
error (mm)	0.72

Table 5.5: Result PID control with Kp=6

5.2 Directly Operated Proportional Valves

For the configuration of the test bench that uses the Directly Operated Proportional Valves, the CX06 Camozzi Module and the PLC, shown Fig 5.19, three types of control were implemented:

- PWM Method using the CTRL_PWM instruction
- PWM Method using the CTRL_PWM instruction driven by a PID controller

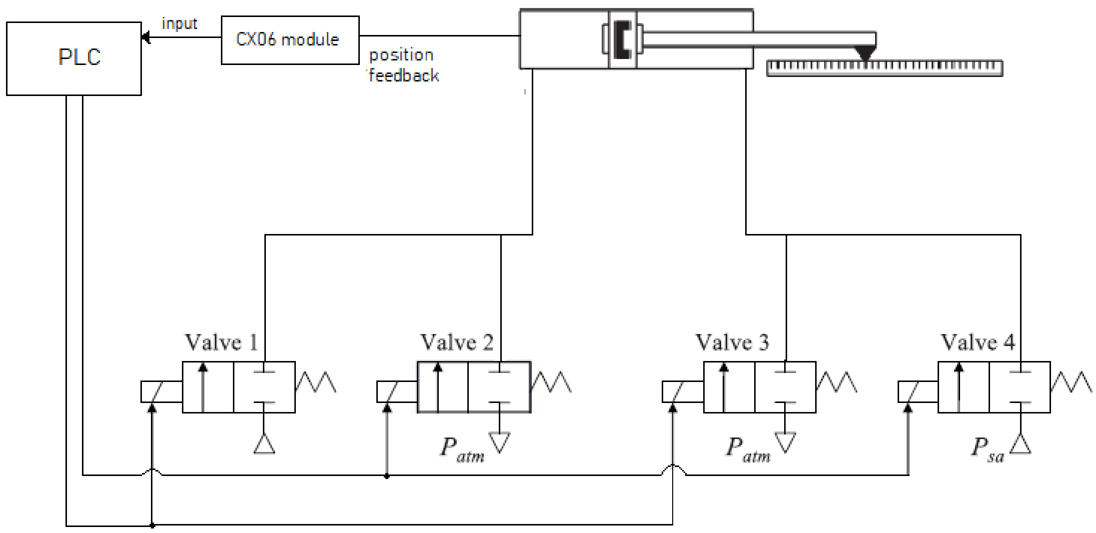


Figure 5.19: Configuration Using Directly Operated Proportional Valves

5.2.1 PWM Method using the CTRL_PWM instruction

At this time, it is essential to talk about the **CTRL_PWM** instruction that the PLC has incorporated. The "CTRL_PWM instruction" provides a fixed cycle time output with a variable duty cycle. After starting at the specified frequency (cycle time), the PWM output will run continuously. The pulse width is changed as needed to achieve the required control. The instruction has two important parameters:

- PWM identifier: The activated pulse generator's name will become a tag in the "constant" tag table and can be used as a PWM parameter. (Default value: 0)

- ENABLE: When the EN input is TRUE, the PWM_CTRL instruction starts or stops the identified PWM according to the value on the ENABLE input. The value specifies the pulse width in the relevant Q word output address.

Fig 5.20 shows the configuration window for the PTO/PWM instruction. The main parameters of the PWM are set on the "Pulse option" section, where the user can choose the signal type: PWM (Pulse Width Modulation) or PTO (Pulse Train Output), the time base, for this experiment, the base is Milliseconds.

When the CPU enters RUN mode for the first time, the pulse width will be set to the initial value configured in the device configuration. The user writes the value to the Q word position specified in the device configuration ("output address" / "start address:") according to the needed to change the pulse width. Instructions such as move, transform, math, or PID can be used to write the required pulse width into the appropriate Q word.

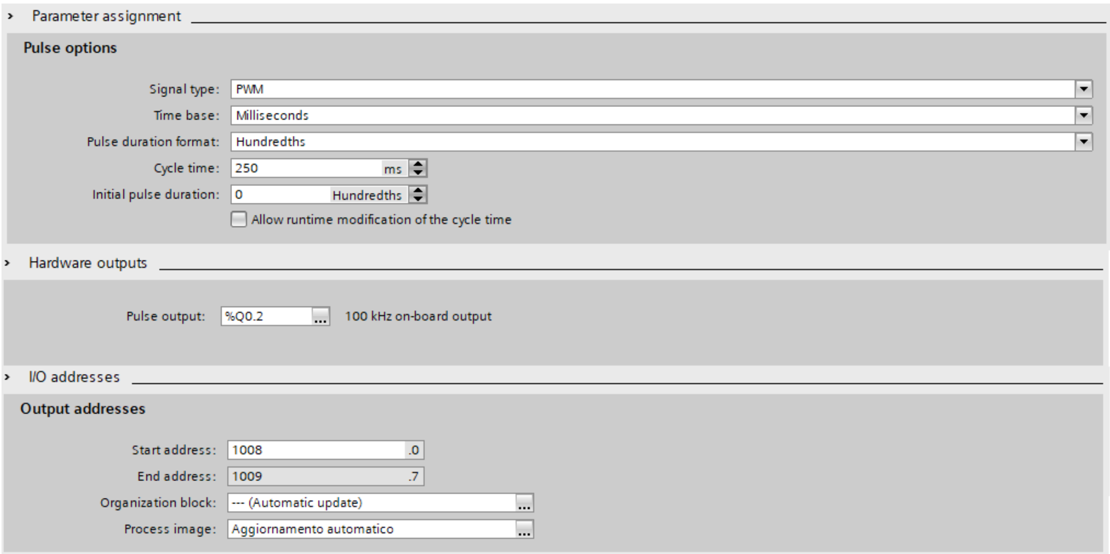


Figure 5.20: Pulse generator window configuration

The ladder program carried out for this experiment is shown below. The first three networks shown in Fig 5.21 are already explained in chapter 4. They are in charge of acquiring the signal and calculating the maximum error possible to get the PWM period.

The last network is in charge of sending the signal of activation to the CTRL_PWM instruction. When the $error = X_{set} - X_{fb}$ is higher than zero, the coil "Carico_Enable" is activated; this coil enables the CTRL_PWM, which drives the rear chamber of the cylinder, in the other hand, when the error is less or equal than zero, the coil

"Scarico_Enable" is activated; this coil enables the CTRL_PWM, which drives the front chamber of the cylinder.

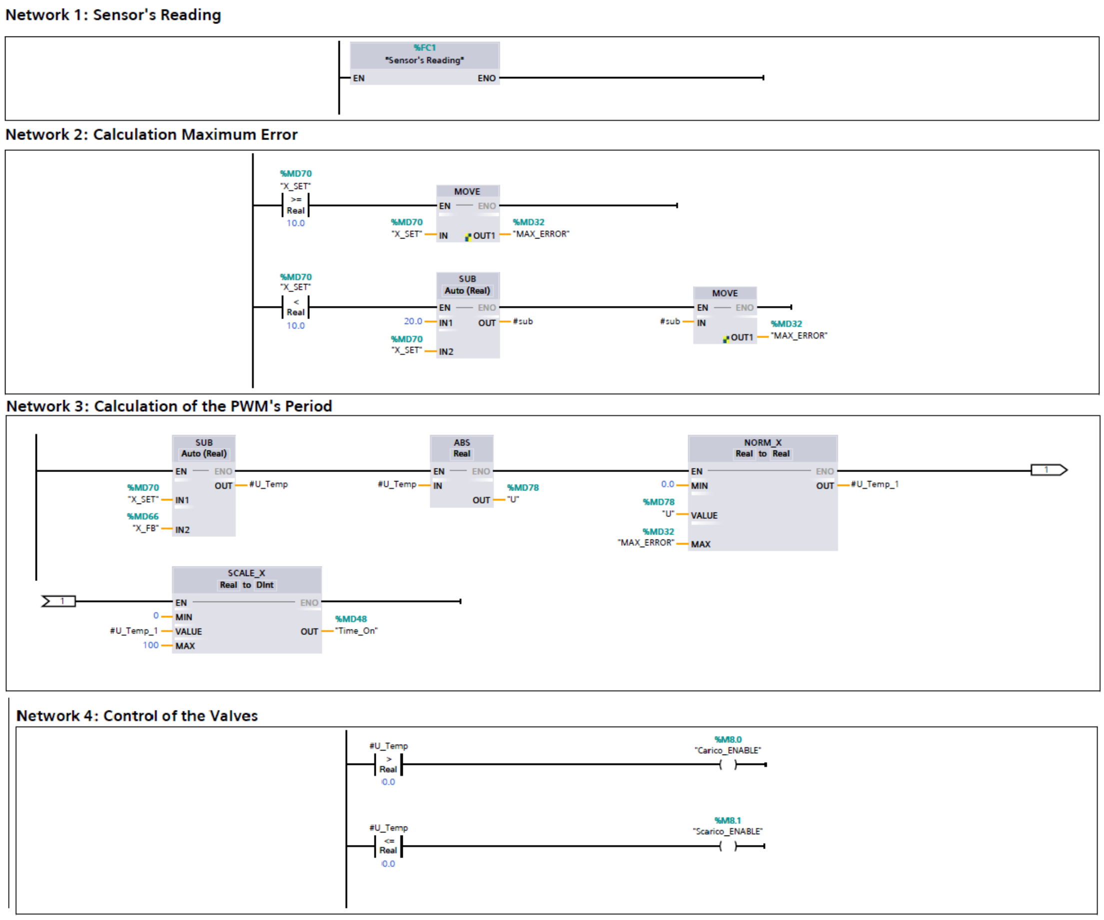


Figure 5.21: Main Organization Block of the PWM Method using the CTRL_PWM instruction program

Fig 5.22 shows an organization block implemented to activate the CRTL_PWM block for the cylinder's front and rear chamber. Network 1 gets the "Time_On" value, which is the duty cycle of the PWM proportional to the error. The block "MOVE" sends this value to the words "QW1008" and "QW1002," which were chosen in the configuration window of the instruction. Network 2 and 3 hold the CTRL_PWM Block. When the "ENABLE" parameter of the block is activated, it sends an analog signal proportional to the duty cycle to command the opening of the Directly Operated Proportional Valves.

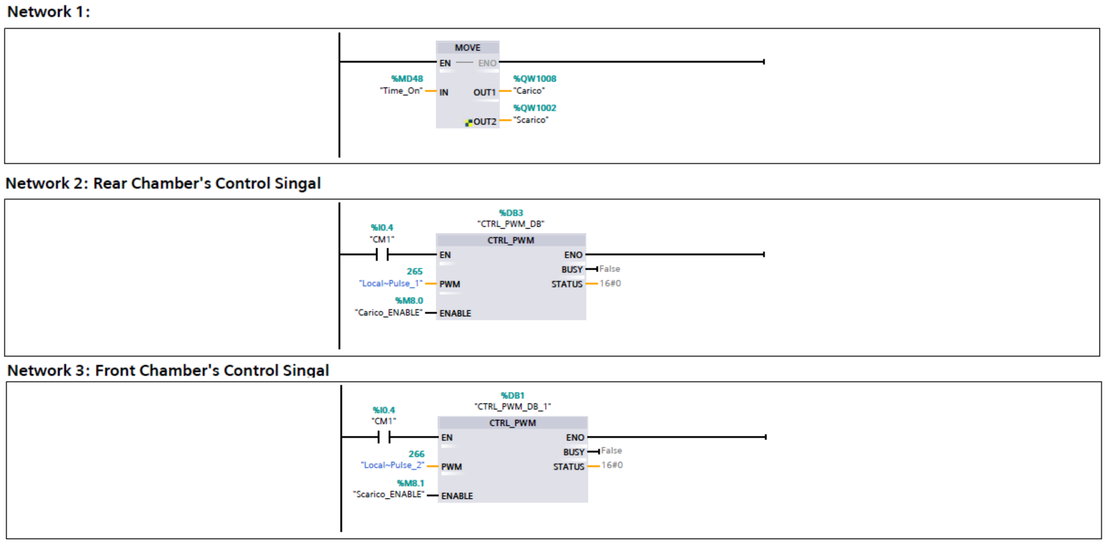


Figure 5.22: CTRL_PWM Organization Block

In addition, in the tests made with this method, the PWM period was varied to see and analyze the system's behavior. The periods used were 100ms, 200ms and 300ms. Fig 5.23 shows the dynamic position response while tracking a step-change in reference signal from 0 mm to 120 mm. The improvement with these valves is notable concerning the valves used in the previous exercise. With this configuration, steady-state errors of 4.9 mm were reached for 100ms, 2.8mm for 200ms, and 0.34mm for 300ms (Fig 5.24). The table 5.6 summarizes the results obtained in this case.

	100ms	**200ms**	**300ms**
ts (s)	2.8	1.8	1.3
error (mm)	4.9	2.8	0.34

Table 5.6: Results PWM Method using the CTRL_PWM instruction

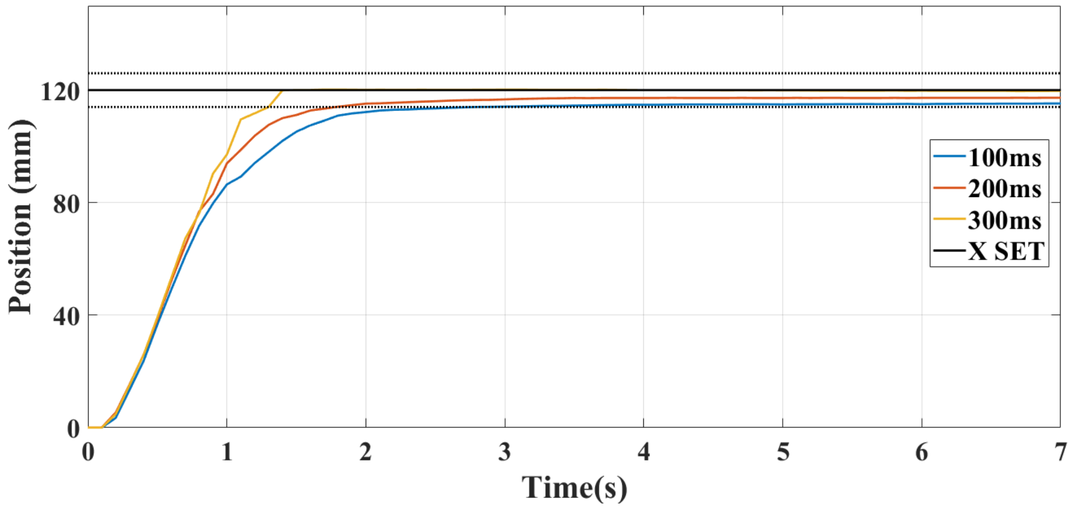


Figure 5.23: Dynamic position response while tracking a step-change in reference signal from 0 mm to 120 mm varying the period of the PWM

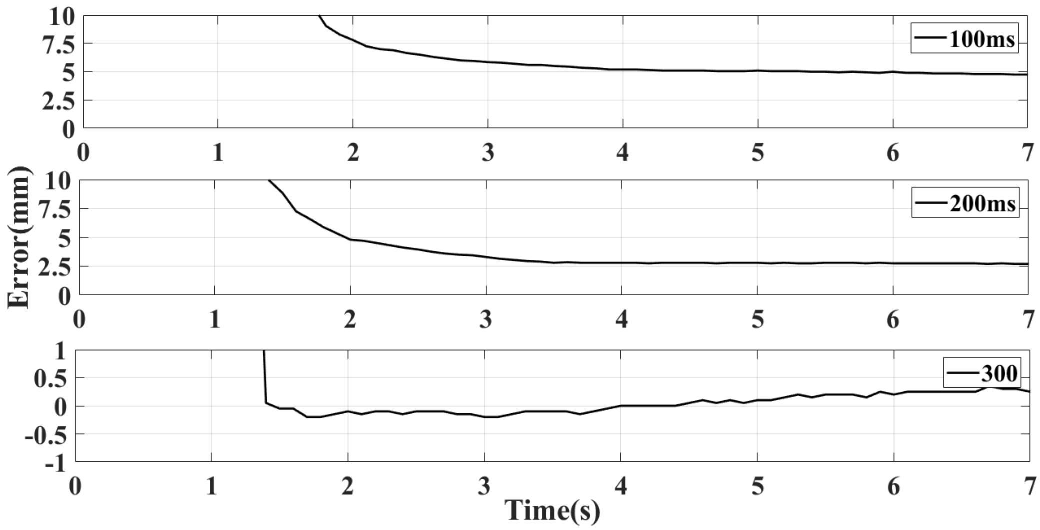


Figure 5.24: Error varying the period of the PWM

5.2.2 PWM Method using the CTRL_PWM instruction driven by a PID controller

This method works in the same way as the previous one, with the difference that, in this case, the PWM duty cycle is modulated by a PID controller. Fig 5.25 shows the Cyclic Time Interrupt where both the PID_Compact and the CRTL_PWM blocks are implemented. The Main organization block is the same shown in Fig **??**

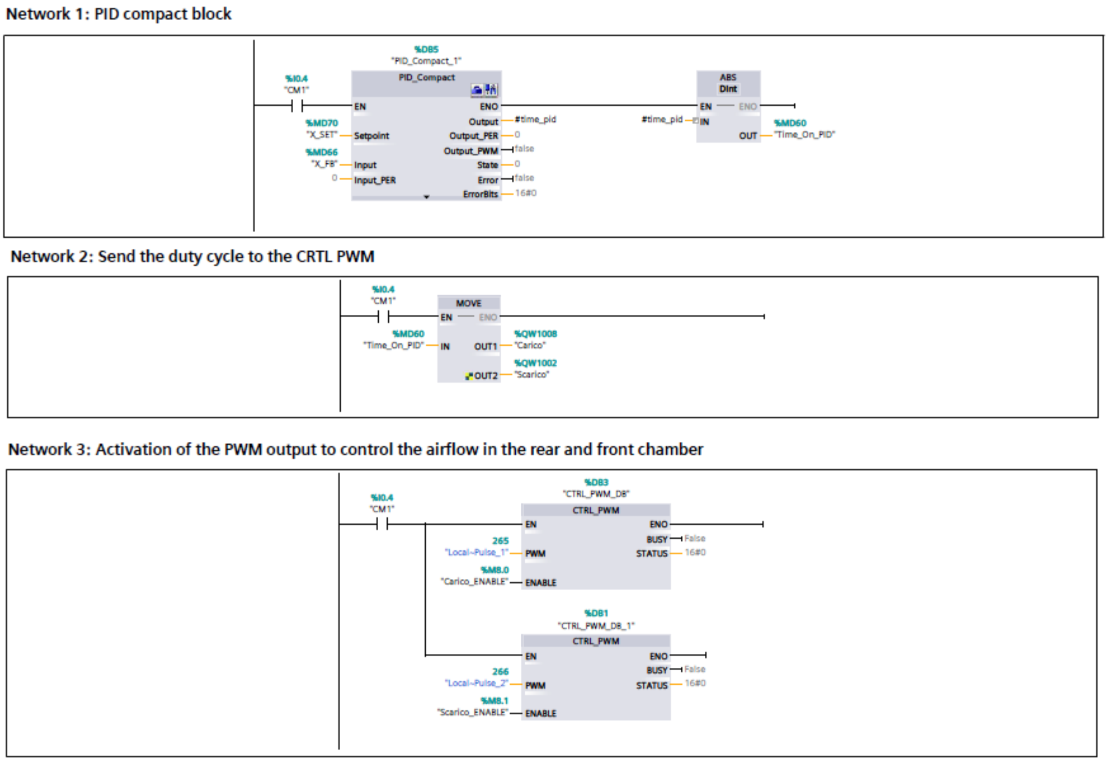


Figure 5.25: PWM Method using the CTRL_PWM instruction driven by a PID controller

Fig 5.26 and Fig 5.27 shows the dynamic position response while tracking a step-change in reference signal from 0 mm to 120 mm varying the K_p parameter and its error, respectively. The effect of the parameter Kp can be clearly seen. When $K_p = 3$, there is no overshoot, but the system responds slowly. On the other hand, with $K_p = 5$ and $K_p = 10$, the system responds faster. Regarding the error, when $K_p = 10$, the error is almost zero; therefore, for the following tests, K_p will be left fixed, and a variation of the other parameters will be made. Results are summarized on the table 5.7.

	$K_p = 10$	$K_p = 15$	$K_p = 30$
ts (s)	1.32	1.2	1.56
error (mm)	2.2	0.39	0.34

Table 5.7: Results of the PWM Method using the CTRL_PWM instruction driven by a PID controller varying Kp

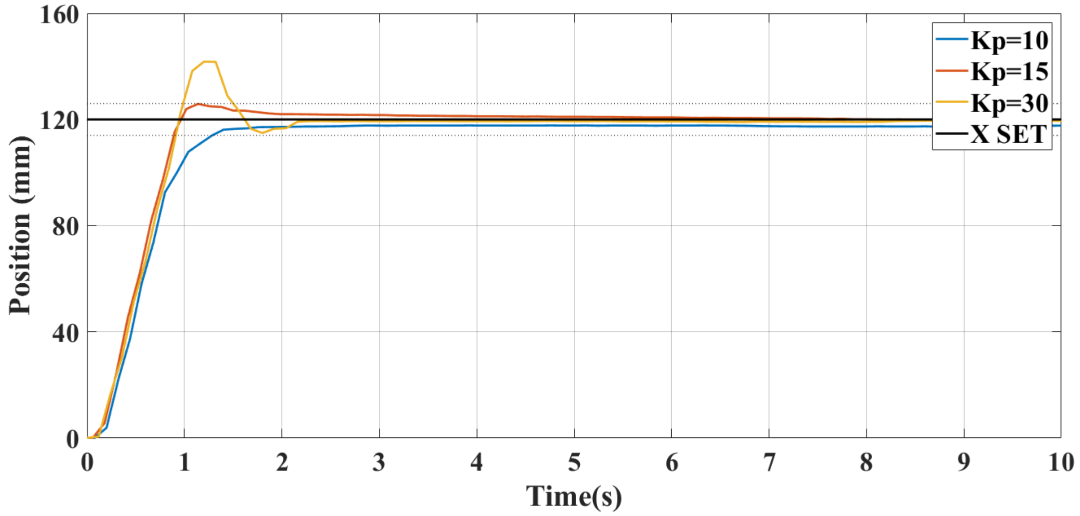


Figure 5.26: Dynamic position response while tracking a step-change in reference signal from 0 mm to 120 mm varying the K_p parameter

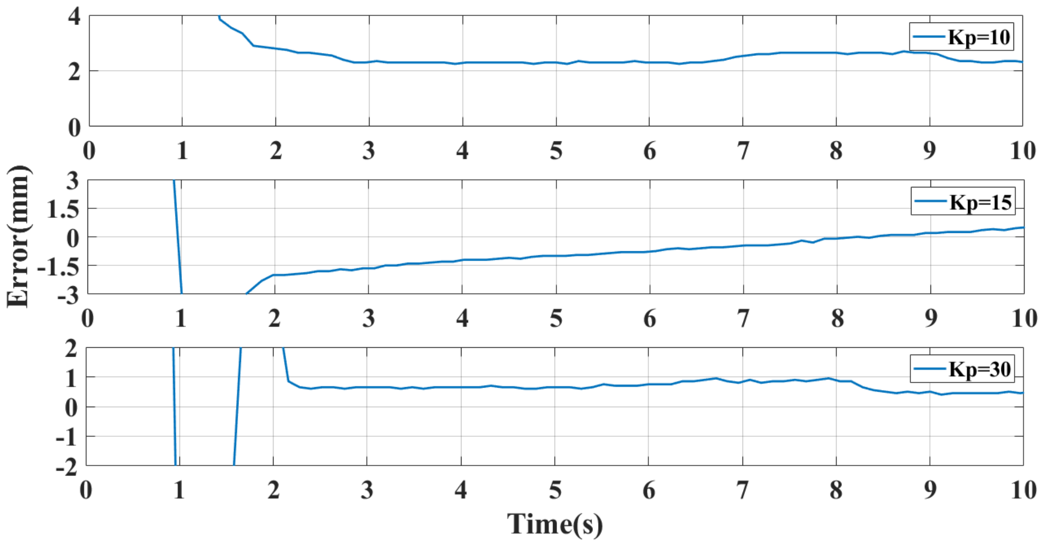


Figure 5.27: Error varying the K_p parameter

Fig 5.28 and Fig 5.29 shows the dynamic position response while tracking a step-change in reference signal from 0 mm to 120 mm varying the K_d parameter and its error, respectively, with $K_p = 15$ fixed.It is remarkable that with the addition of the K_d parameter, the system becomes more stable and a slower; the error almost does not change. Table 5.7 summarizes the results obtained varying K_d.

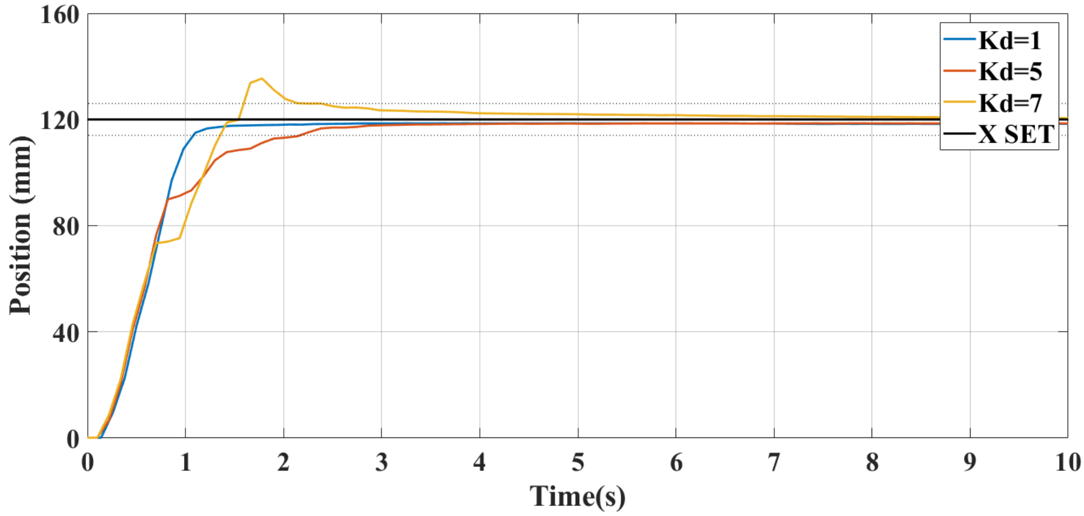


Figure 5.28: Dynamic position response while tracking a step-change in reference signal from 0 mm to 120 mm varying the K_d parameter

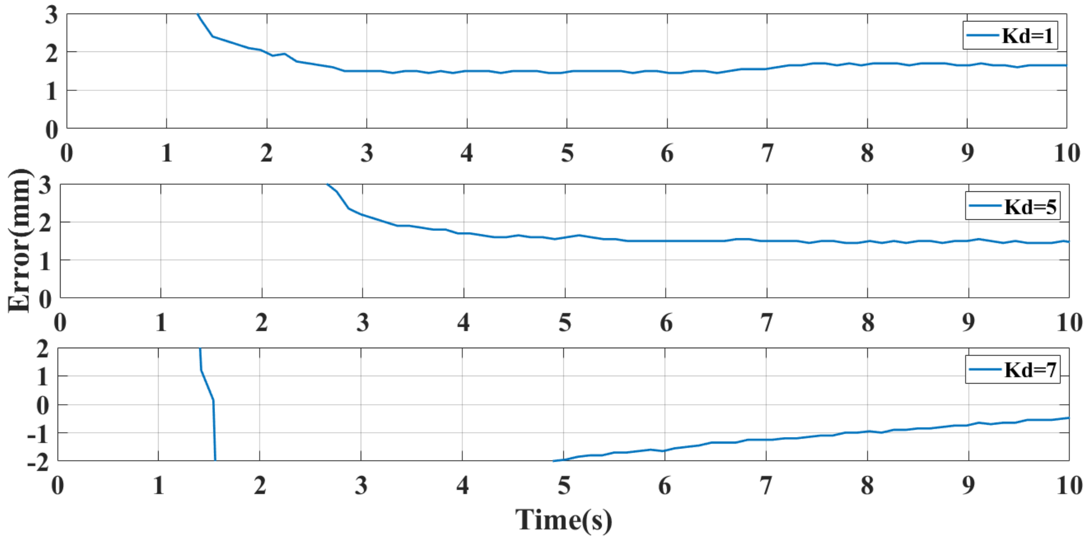


Figure 5.29: Error varying the K_d parameter

Fig 5.30 and Fig 5.31 shows the dynamic position response while tracking a step-change in reference signal from 0 mm to 120 mm varying the K_i parameter and its error, respectively, with $K_p = 15$ and $K_d = 1$ fixed. With the introduction of the Ki parameter, it is visible that the system improves its speed and the error in a steady-state compared to the previous result. Table 5.9 summarizes the results obtained varying K_i

	$K_d = 1$	$K_d = 5$	$K_d = 10$
ts (s)	1.2	2.06	2.18
error (mm)	1.4	1.5	0.5

Table 5.8: Results of the PWM Method using the CTRL_PWM instruction driven by a PID controller varying Kd

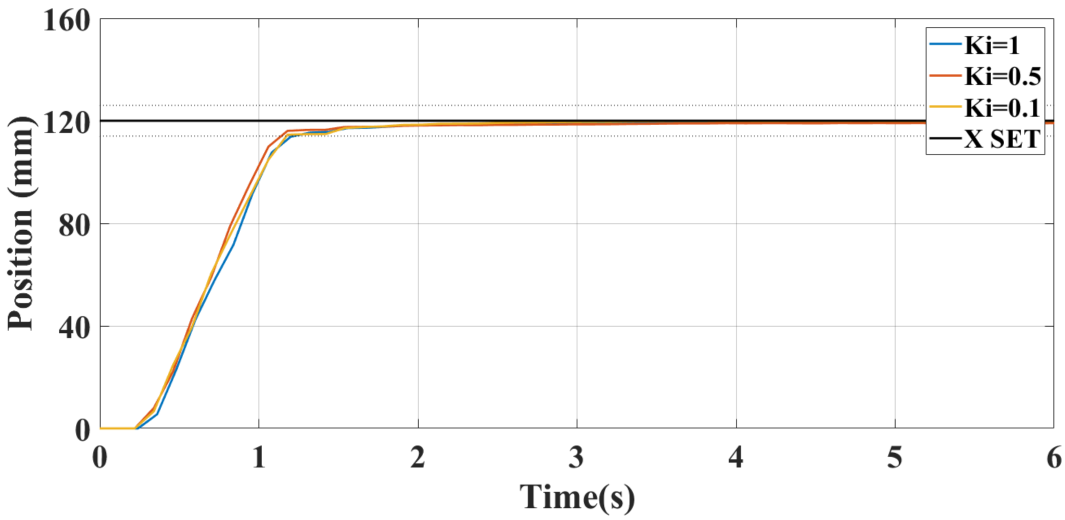


Figure 5.30: Dynamic position response while tracking a step-change in reference signal from 0 mm to 120 mm varying the K_i parameter

Ziegler-Nichols Tuning Rule

After analyzing the system's behavior as the main PID parameters (Kp, Ki, Kd) vary, it is necessary to find the right combination of values that make the system optimal. Different techniques for the calibration of the PID parameters that allow identifying this triad and for this positioning control, the Ziegler-Nichols method in the closed-loop have been chosen. However, this method cannot be applied to all control systems, but only to those which, if brought to the threshold of instability, do not cause damage. This is a purely experimental calibration method, and in order to apply it, the following steps must be performed in the order in which they are listed [31]:

Step 1: Cancel in the closed-loop control system all the parameters that characterize a PID controller (Kp, Ki and Kd)

Step 2: Gradually increase the value of the proportional coefficient (Kp) until, by

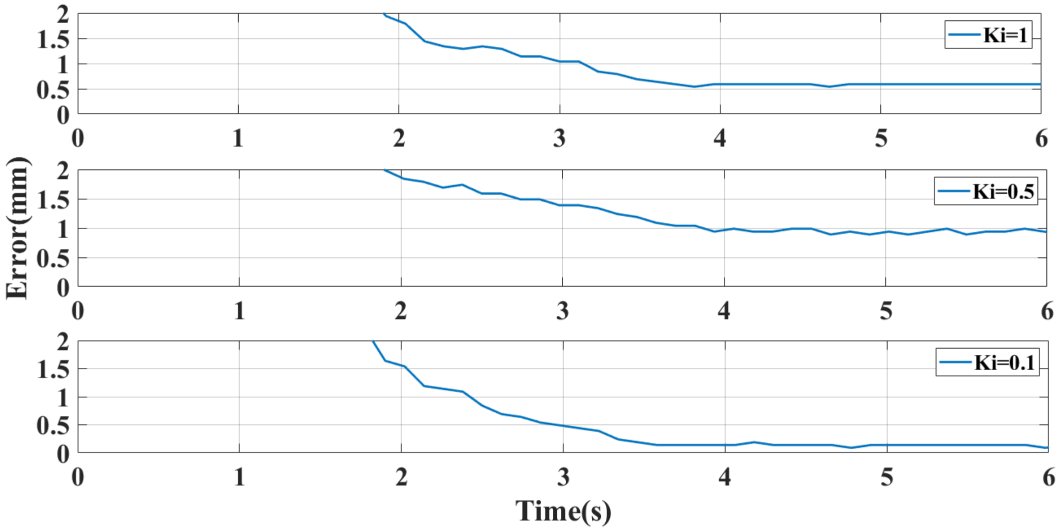


Figure 5.31: Error varying Ki

	$K_i = 1$	$K_i = 0.5$	$K_d = 0.1$
ts (s)	1.2013	1.2026	1,.2057
error (mm)	0.53	0.83	0.13

Table 5.9: Results of the PWM Method using the CTRL_PWM instruction driven by a PID controller varying Ki

introducing a small step perturbation of the input into the system, a permanent oscillation of the output is generated. The value of the Kp obtained will be called critical gain or K_c, while the period of oscillation generated T_c (if the system never enters oscillation, the method is not applicable)

Step 3: Insert in table 5.10 the values of K_c and Tc, according to the type of controller to be used, and then extrapolate the calculated PID parameters' values.

This procedure was applied to this positioning control in order to optimize it. After varying the K_p parameter, values of $K_c = 55$ and $Tc = 1.2s$ were found. Table 5.11 reports the values of the parameters calculated using the table mentioned above.

Fig 5.33 shows the dynamic response of Dynamic position response while tracking a step-change in reference signal from 0 mm to 120 mm whit the controllers calculation using the Ziegler-Nichols tuning rule and its respective steady-state error.The table 5.12 shows the result obtained from this exercise.

Control Type	K_p	K_i	K_d
P	$0.5K_c$		
PI	$0.45K_c$	$1.2K_p/T_c$	
PID	$0.60K_c$	$2K_p/T_c$	$K_pT_c/8$

Table 5.10: Look-up table for PID-controller tuning with the Ziegler-Nichols method

Control Type	K_p	K_i	K_d
P	32.5		
PI	29.25	31.33	
PID	38.35	68.48	5.46

Table 5.11: Calculated values by Ziegler-Nichols Tuning Rule

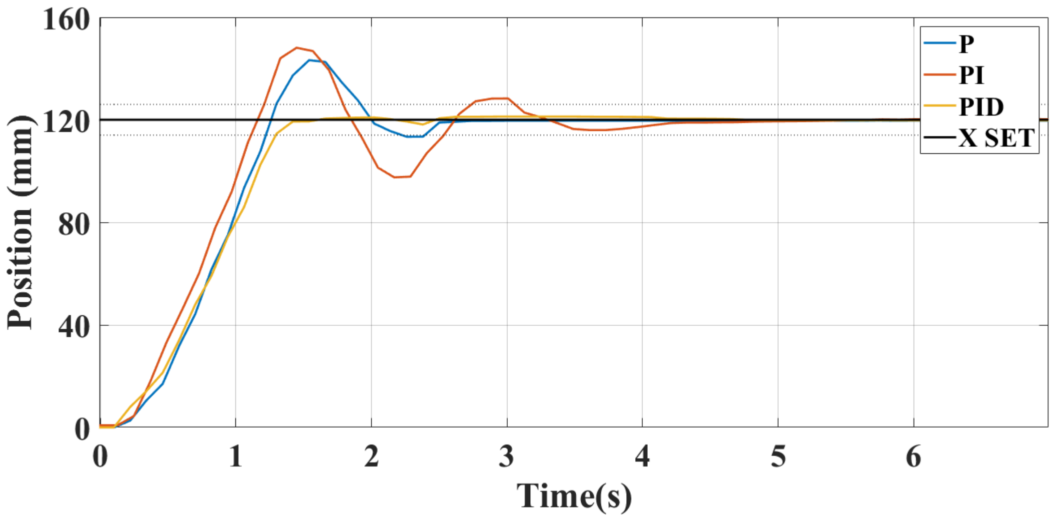


Figure 5.32: Dynamic response of Dynamic position response while tracking a step-change in reference signal from 0 mm to 120 mm Ziegler-Nichols Method

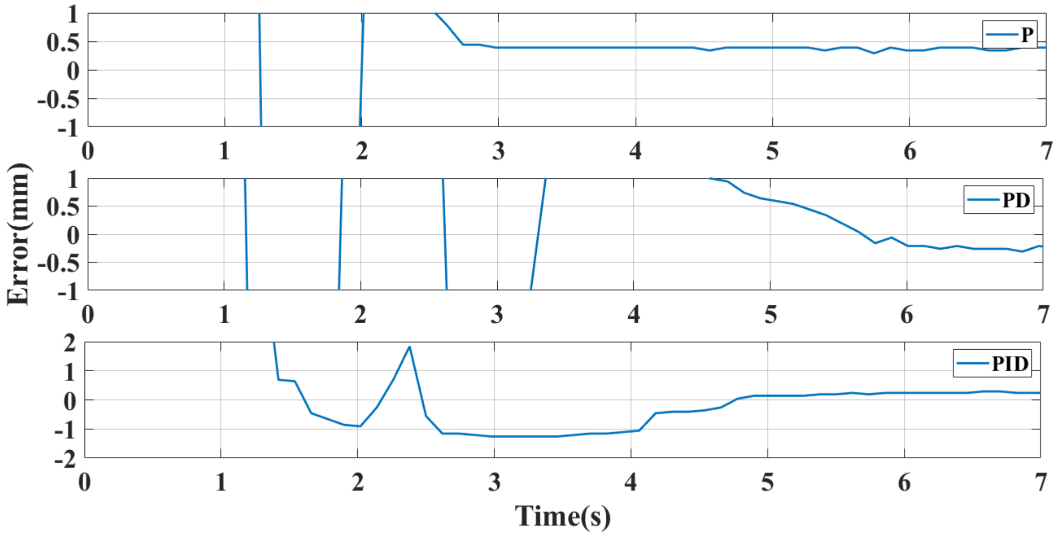


Figure 5.33: Error Ziegler-Nichols Method

	P	PI	PID
ts (s)	2.48	3.1	1.3
error (mm)	0.28	0.25	0.23

Table 5.12: Resultes of the PID Ziegler Nichols Tuning Method

By comparing the three responses obtained using the Ziegler-Nichols method, it is possible to state that, although the three options stabilize quickly and have a low steady-state error, the best would be the PID controller, since, of the three, it is the only one that does not present an overshoot.

5.3 Directly Operated Valves driven by an Electronic Control Device for Proportional Valves

This section discusses the drivers' operation coupled to the valves in the test bench. As mentioned in chapter 3, the purpose of the driver is to obtain a proportional flow behavior. Although it was not possible to disassemble a valve to observe its internal structure, it is logical to assume that this is constituted by a shutter kept in position by a spring, which guarantees proportionality.

This configuration of the test bench is similar to the last one, but in this case, the PLC's output goes first to the Electronic Control Device, and then a PWM signal commands the Directly Operated Proportional. Figure 2 shows the system mentioned above. For this experiment,

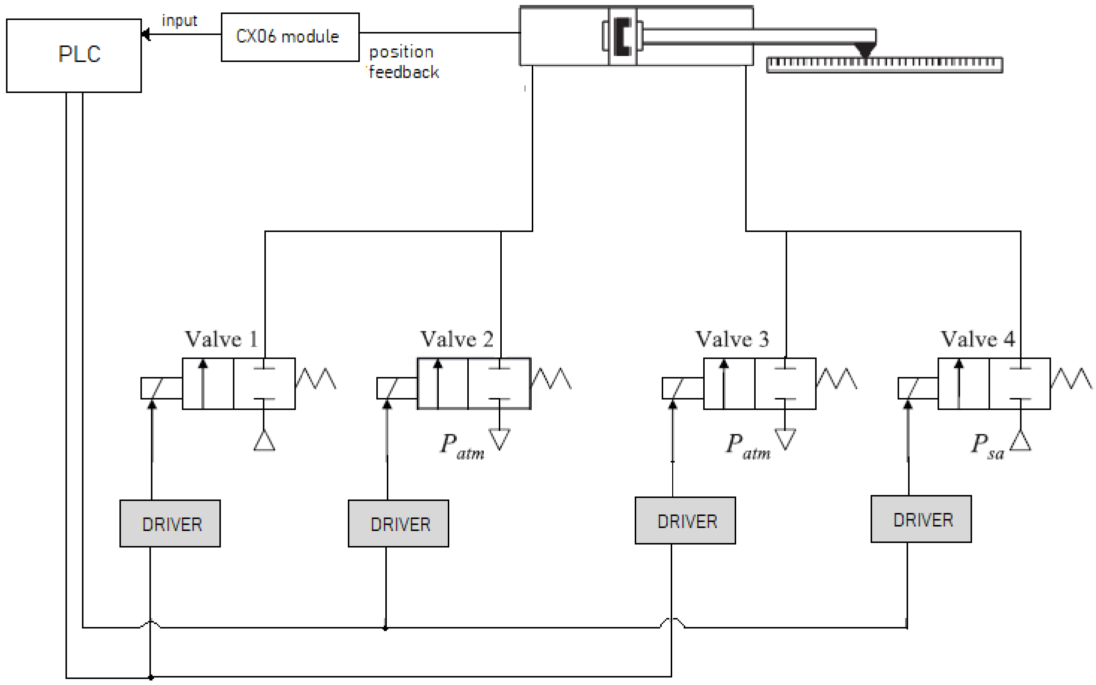


Figure 5.34: Directly Operated Valves driven by an Electronic Control Device for Proportional Valves

5.3.1 Duty Cycle of the PWM inside the Electronic Control Device is proportional to the error.

Fig 5.35 shows Network 1 and Network 2 of the program developed for this exercise. Network 1 contains the Sensor's Reading FB already explained in chapter 4. Network 2 is helpful to choose the type of signal the system will follow. Through the variable "SELECT" the user can choose between a Step Reference Signal, Sine Wave Reference Signal, and Square Wave Reference Signal. The Sine and Square signal are generated in their respective organization block explained in detail in chapter 4. This signal is sent employing the block MOVE, which sends the signal to the PID_Compat as the SETPOINT input.

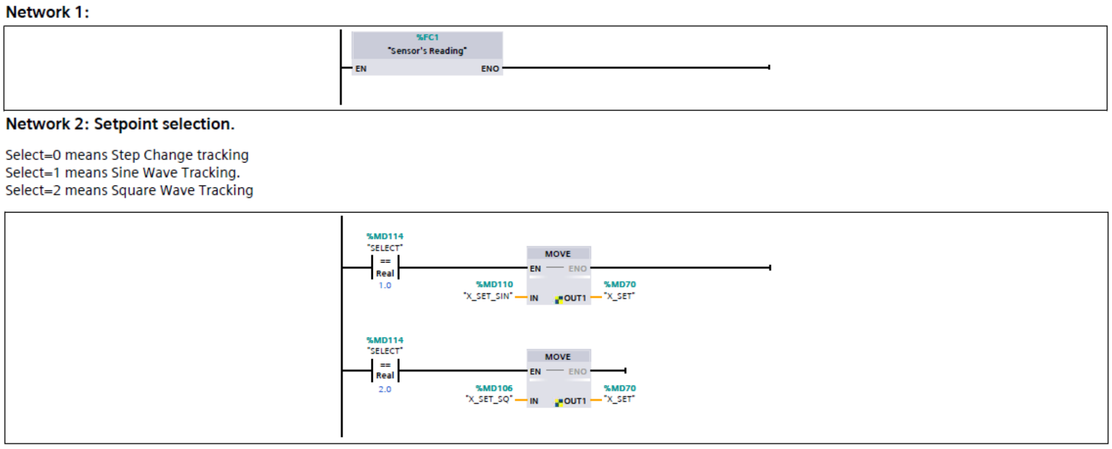


Figure 5.35: Network 1 and 2 of the program for the Electronic Control Device

Network 3, shown in Fig 5.36 converts the output of the PID Controller into the actual analog value sent to the valves. During the tests carried out, it was verified that the minimum voltage that guarantees flow through the valves is equal to 3.5 V, with a supply pressure of 6 bar. This aspect leads to a high non-linearity in the valves' response, overcome by mapping the PID output in the range 3.5V to 10 V, or 9680-27648 for the PLC.

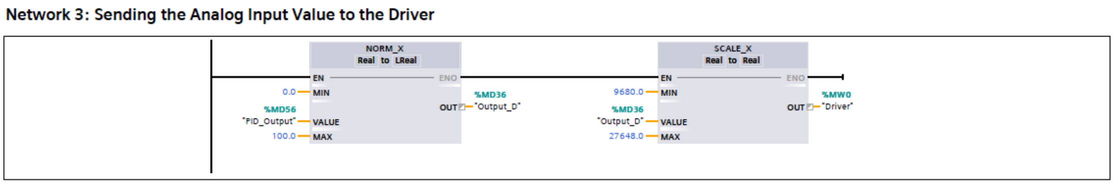


Figure 5.36: Network 3 of the program for the Electronic Control Device

Network 4 and 5, shown in Fig 5.37, are in charge of sending the analog value into the driver. A microprocessor translates the reference value of the shutter position, imposed in the form of voltage through the analog output "QW96" and "QW98," into a current value that must flow in the coil of the solenoid. This value is then compared with the feedback coming from the valve, and the resulting error is compensated through a specific control law. The result of the compensation operation is the duty-cycle value of the driver output signal.

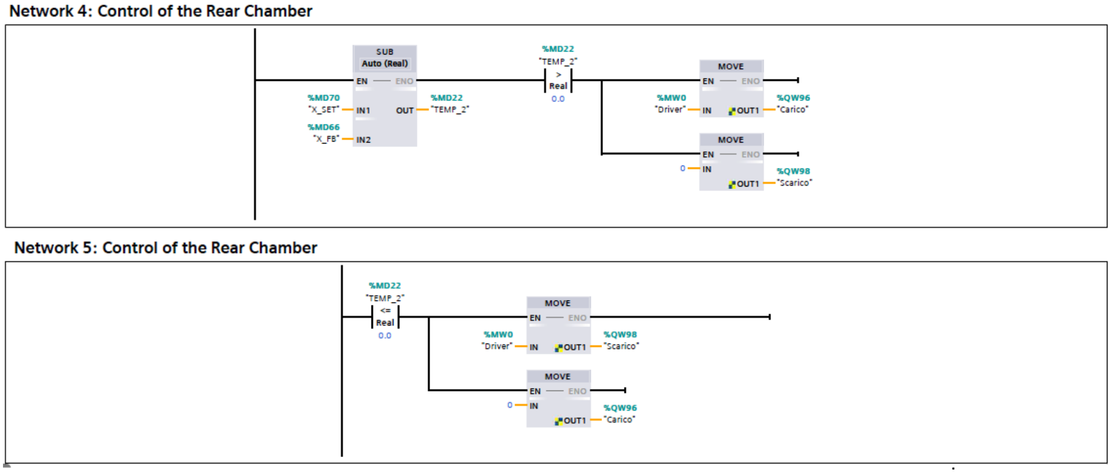


Figure 5.37: Network 4 and 5 of the program for the Electronic Control Device

Fig 5.38 and Fig 5.39 show the dynamic position response while tracking a step-change in reference signal from 0 mm to 120 mm, varying the Kp parameter and its error. It is easy to see the effect of the parameter Kp over the system. When $K_p = 8$, there is no overshoot, but the system responds slowly. On the other hand, with $K_p = 15$ and $K_p = 20$, the system responds faster and and the overshoot is minuscule. Besides, $K_p = 30$ generates an overshoot that is not a desirable result for this test. Results are summarized in table 5.13.

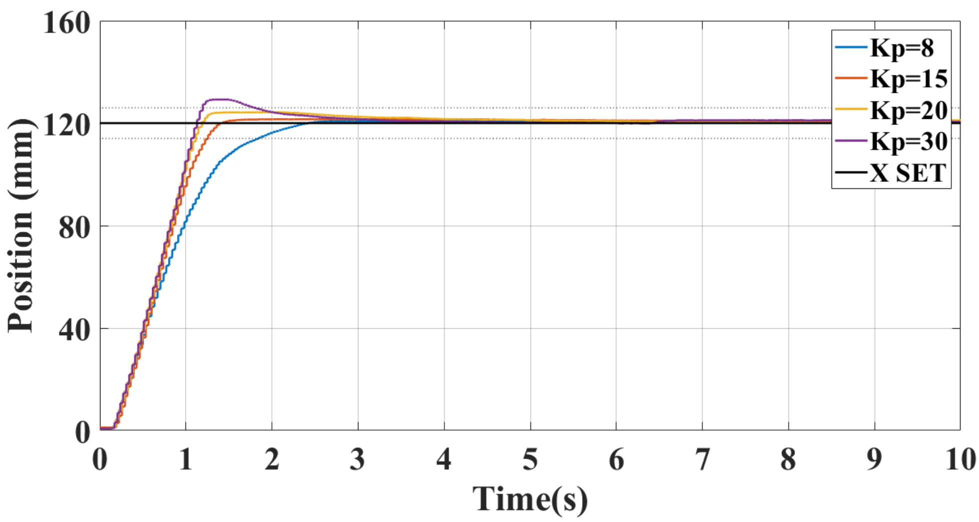


Figure 5.38: Dynamic position response while tracking a step-change in reference signal from 0 mm to 120 mm varying the K_p parameter

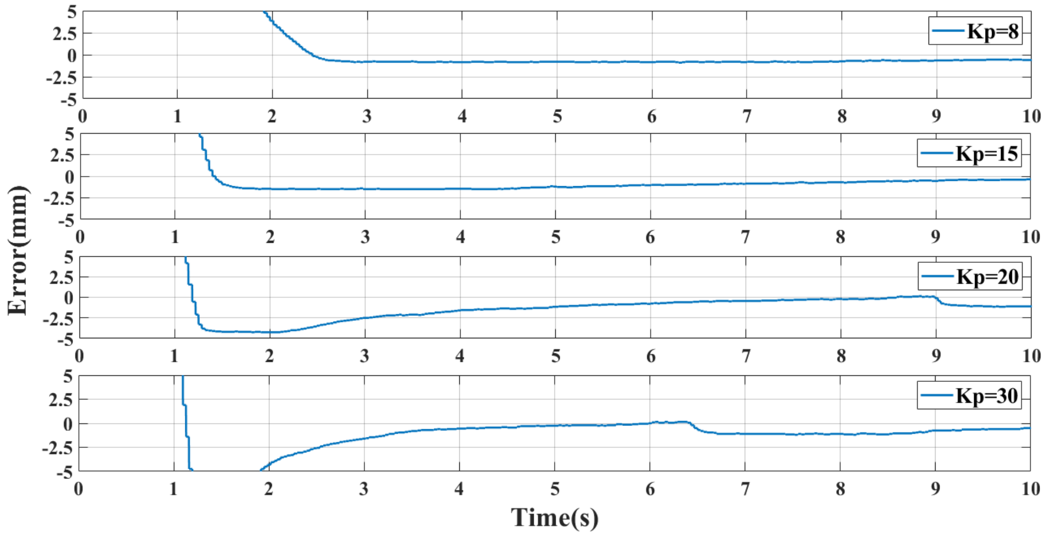


Figure 5.39: Error varying K_p parameter

	$K_p = 8$	$K_p = 15$	$K_p = 20$	$K_p = 30$
ts(s)	1.84	1.24	1.12	1.061
error(mm)	0.7	0.4	0.18	0.6

Table 5.13: Results using the Electronic Control Device varying K_p

Fig 5.40 and Fig 5.41 show the dynamic position response varying the Kd parameter and its error, respectively, with $K_p = 20$ fixed. It is remarkable that with the addition of the Kd parameter, the system becomes more stable, and the stade-state error goes almost to zero. Table 5.14 summarizes the results obtained varying Kd.

	$K_d = 1$	$K_d = 5$	$K_d = 10$
ts(s)	1.25	1.66	2.12
error(mm)	0.3	0.22	0.26

Table 5.14: Results using the Electronic Control Device varying K_d

Fig 5.42 and Fig 5.43 show the dynamic position response while tracking a step-change in reference signal from 0 mm to 120 mm varying the Ki parameter and its error, respectively, with $K_p = 20$ and $K_p = 5$ fixed. With the introduction

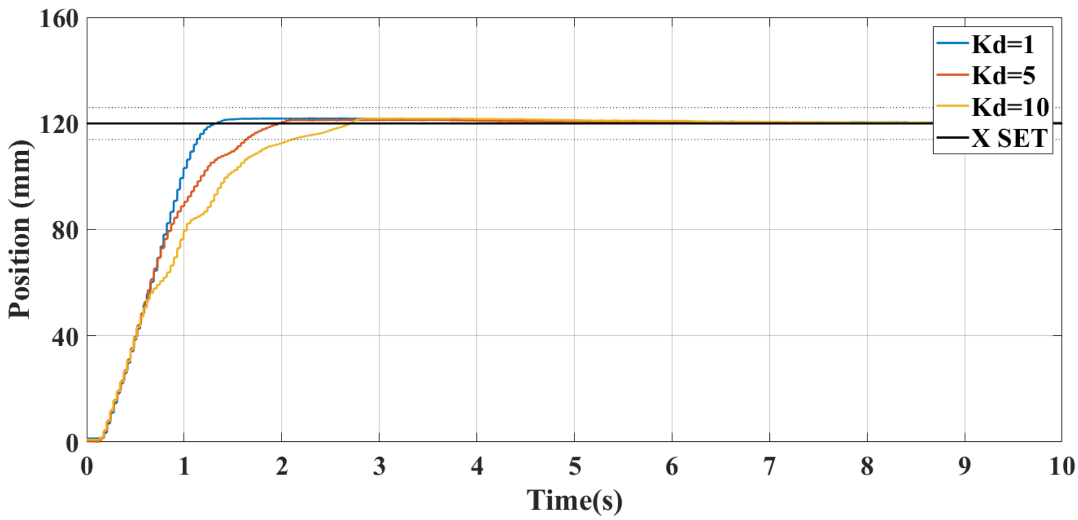


Figure 5.40: Dynamic position response while tracking a step-change in reference signal from 0 mm to 120 mm varying the K_d parameter

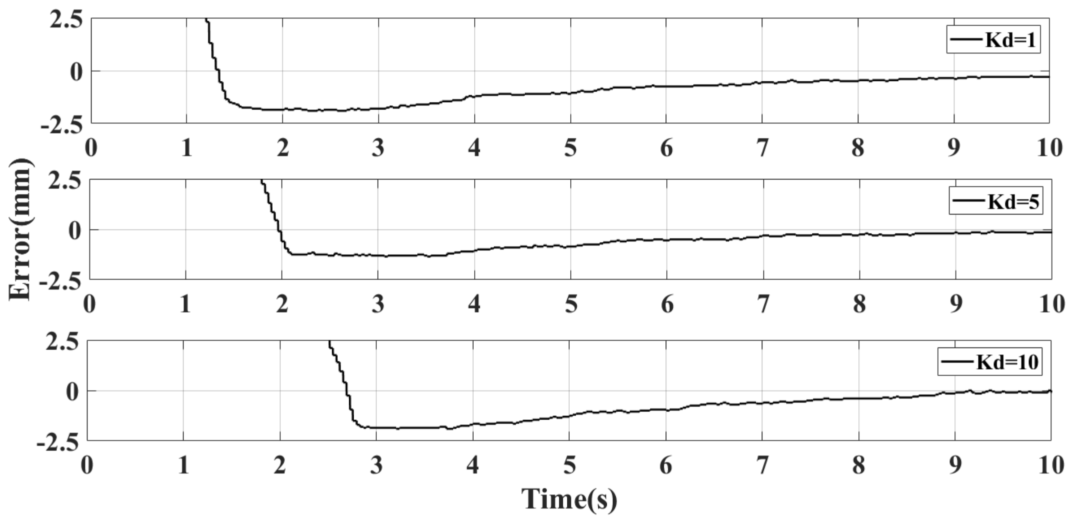


Figure 5.41: Error varying K_d parameter

of the Ki parameter, it is visible that the system improves its speed and the error in a steady-state compared to the previous result. Table 5.15 summarizes the results obtained varying Ki.

This procedure was applied to this positioning control in order to optimize it. After varying the Kp parameter, values of Kc = 100 and Tc = 0.6s were found. Table

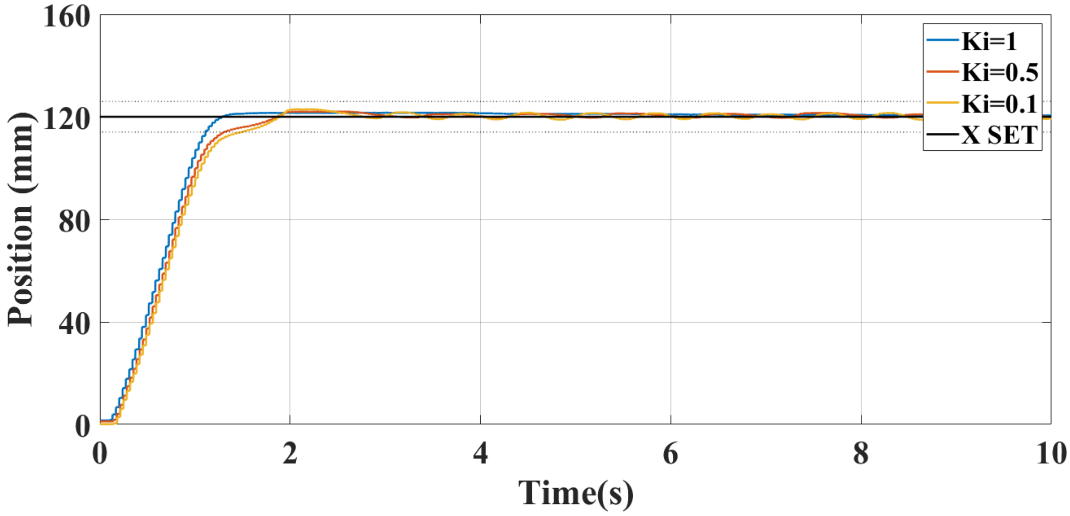


Figure 5.42: Dynamic position response while tracking a step-change in reference signal from 0 mm to 120 mm varying the K_i parameter

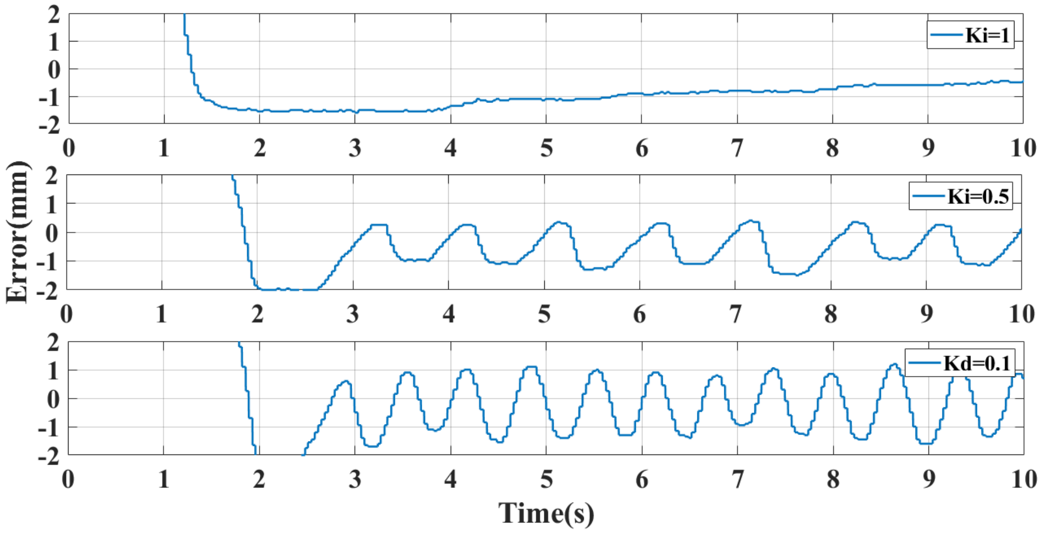


Figure 5.43: Error varying K_d parameter

5.16 reports the values of the parameters calculated using the table mentioned early.

Fig 5.44 and Fig 5.45 shows the dynamic response of Dynamic position response while tracking a step-change in reference signal from 0 mm to 120 mm whit the controllers' calculation using the Ziegler-Nichols tuning rule and the steady-state error of each controller. Table 5.17 shows the result obtained from this exercise.

	$K_i = 1$	$K_i = 0.5$	$K_i = 0.1$
ts(s)	1.19	1.13	1.4
error(mm)	0.4	1.2	1.1

Table 5.15: Results using the Electronic Control Device varying K_i

Control Type	K_p	K_i	K_d
P	50		
PI	45	90	
PID	59	196.66	4.5

Table 5.16: Calculated values by Ziegler-Nichols Tuning Rule

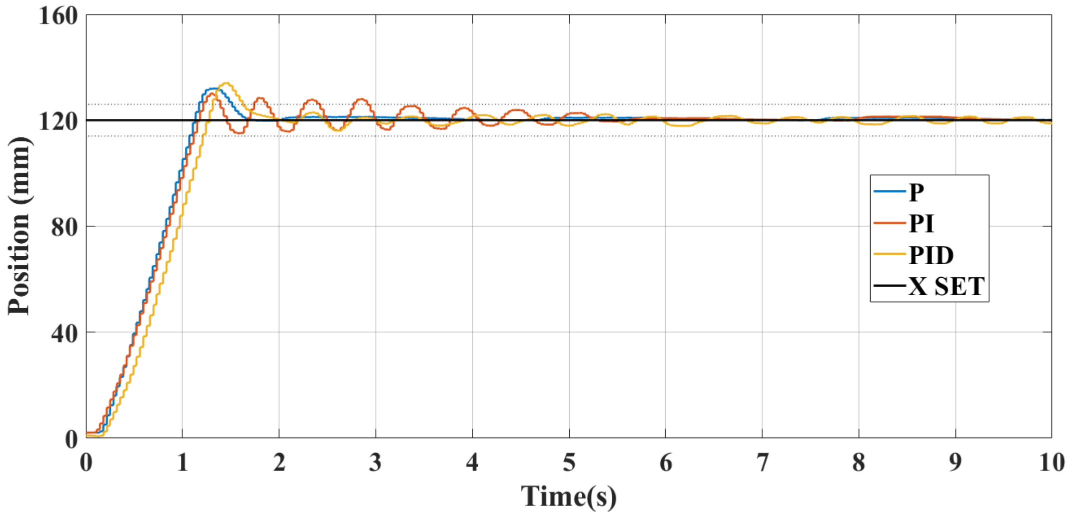


Figure 5.44: Dynamic position response while tracking a step-change in reference signal from 0 mm to 120 mm Ziegler-Nichols Method

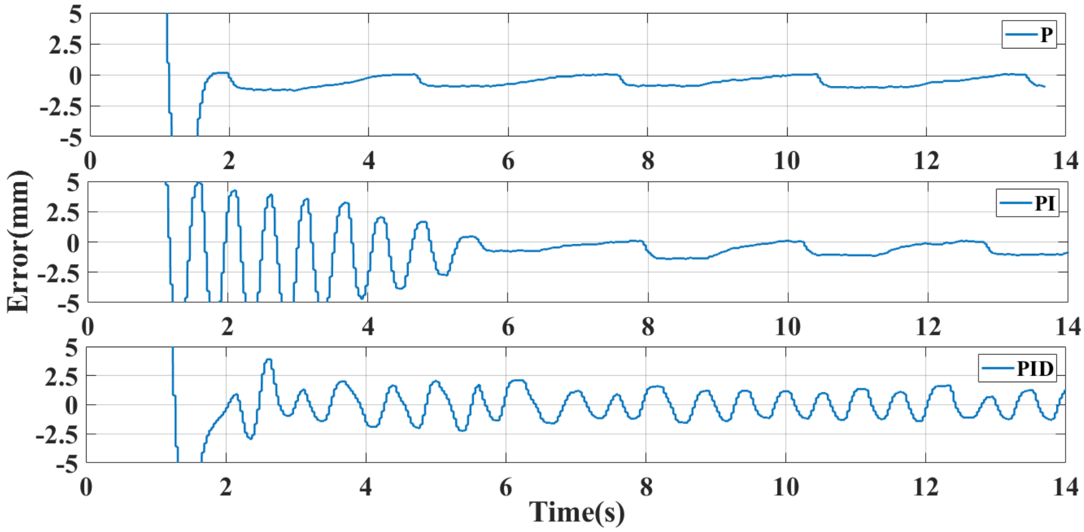


Figure 5.45: Error Ziegler-Nichols Method

	P	PI	PID
ts(s)	1.51	2.93	1.6
error(mm)	1.05	1.35	1.65

Table 5.17: Resultes of the PID Ziegler Nichols Tunning Method

5.3.2 Square Wave Tracking

Two tests of tracking a square wave signal are presented below. In the firsts two cases, shown in Fig5.46, the amplitude of the square wave is 120mm peak-to-peak for both and 2s and 10s, respectively. In the first case, the system cannot follow the reference signal due to the square signal period. The tests carried out previously discovered that the minimum settling time is more or less 1.2 seconds. The system does not reach the setpoint before the signal changes again because the square wave has a 2 seconds period. On the other hand, when the square wave period is 10s, the system reaches the setpoint and remains stable until the reference signal changes again. Fig 5.47 shows the test's error, making visible how the error remains near zero for the second case until the setpoint changes.

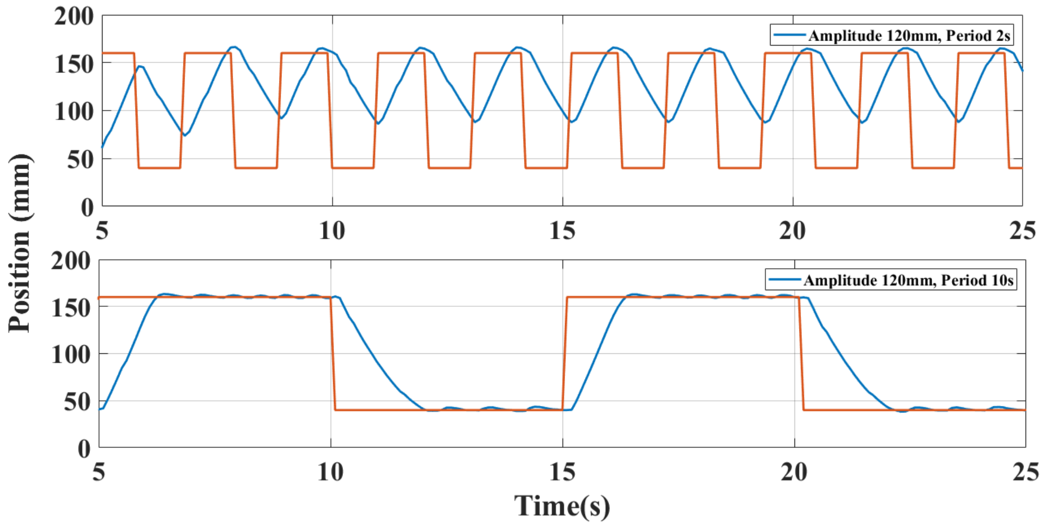


Figure 5.46: Dynamic position response while tracking a square wave in reference signal of 120mm peak-to-peak amplitude

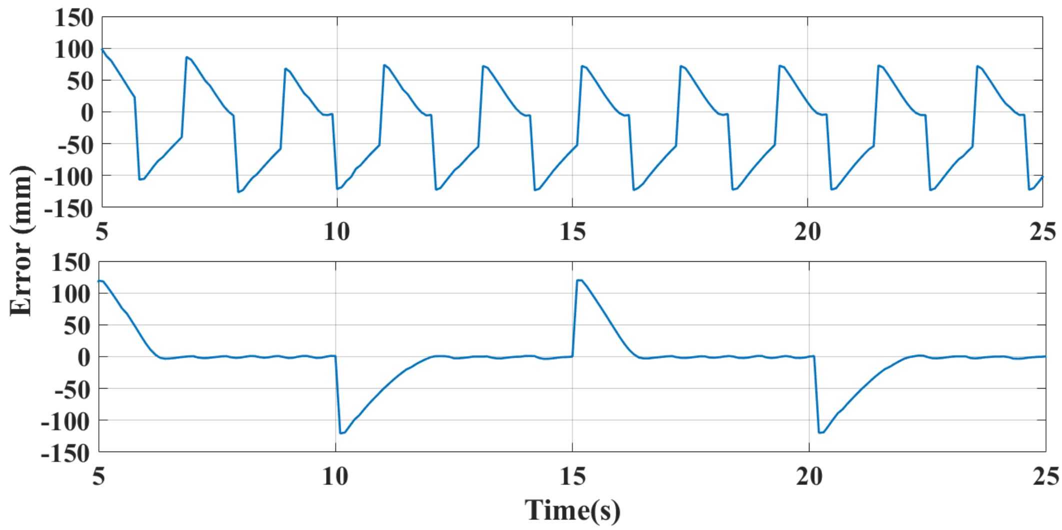


Figure 5.47: Error of square wave in reference signal

5.3.3 Sine Wave Tracking

The sinusoidal signal following tests allows to easily observe the phase shift phenomenon, which decreases as the setpoint variation frequency decreases. This phase shifth is calculated as follows:

- t_d: is the time difference between a wave point and the equivalent of another wave.

- ϕ: is the phase angle or phase shift (in degrees or radians) that the waveform has shifted either left or right from the reference point.

$$\phi = \omega \times t_d$$
$$\omega = 2\pi F$$

Furthermore, in the first case shown in Fig 5.48, the period is 3s, so ω is equal to $\frac{2}{3} \times \pi$ and t_d=0.5s. This leads to a phase shift equal to $\phi = \frac{\pi}{3}\ \frac{rad}{sec}$. This means that the system reaches to follow the sine wave form with a retard of 60°.

Moreover, in the second case, with $T = 10s$, the period is big enough to get almost perfect tracking. Fig 5.49 shows the error of both experiments. Regarding the second one, it is notable that the error never exceeds 5mm.

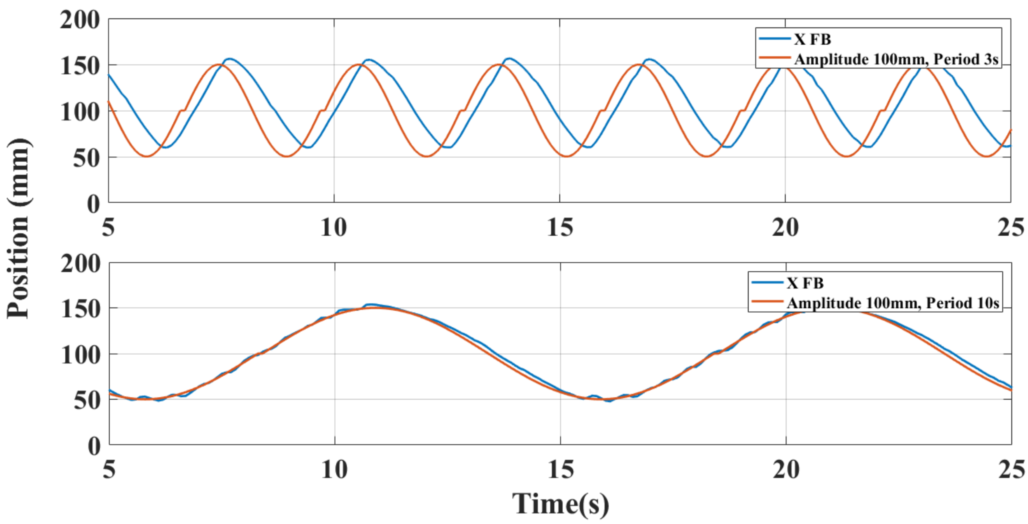


Figure 5.48: Dynamic position response while tracking a sine wave in reference signal of 100mm peak-to-peak amplitude

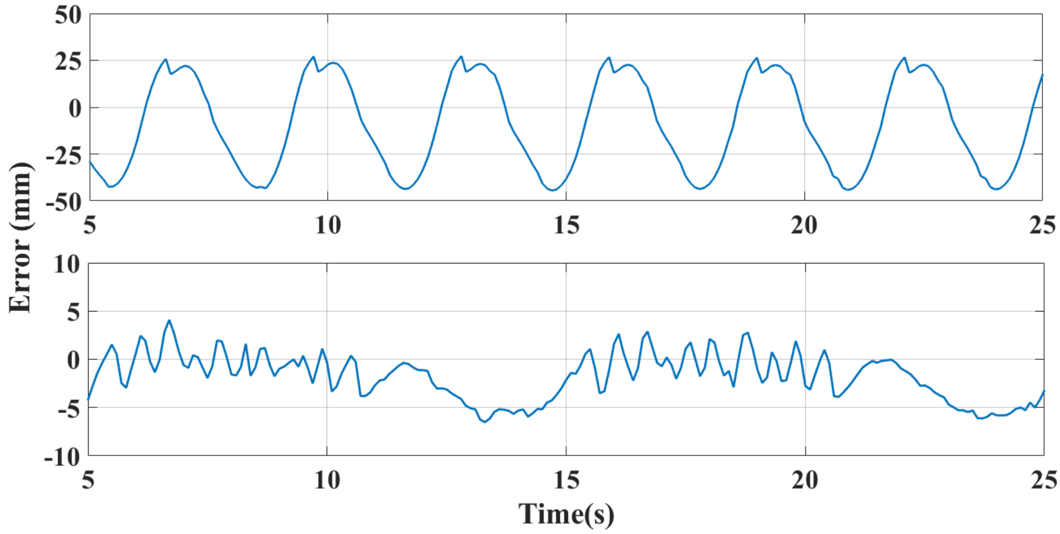


Figure 5.49: Error of sine wave in reference signal

5.4 Series LR digital proportional servo valves

The final position control method consists of using the Series LR digital proportional servo valves. Three configurations are used for this experiment.

- Master-Slave configuration
- Master Valve and Pressure Regulator in the front chamber of the cylinder
- Master Valve and Pressure regulator in the rear chamber of the cylinder

When using the Series LR servo valves, the position setpoint is sent as a 0-10Vdc signal from the PLC through the CX06 module to the MASTER valve, which also gets the feedback from the cylinder; the control loop is programmed inside the MASTER valve, which automatically decides the airflow though itself and the slave valve in order to move the actuator to the desired setpoint.

Below is the ladder program, which is the same for the three different configurations.The first two networks are shown in Fig 5.50. Network 1 is in charge of reading the analog input from the sensor. Since it is a 0-10V signal, the normalization must be done between 0 and 27648. In this case, the feedback value is acquired to plot the results because the PID controller is implemented inside the Master valve, so there is no need for the PLC process value. Network 2 is helpful to choose the type of signal the system will follow. Through the variable "SELECT," the user can choose between a Step Reference Signal, Sine Wave Reference Signal, and

Square Wave Reference Signal. The Sine and Square signal are generated in their respective organization block explained in detail in chapter 4.

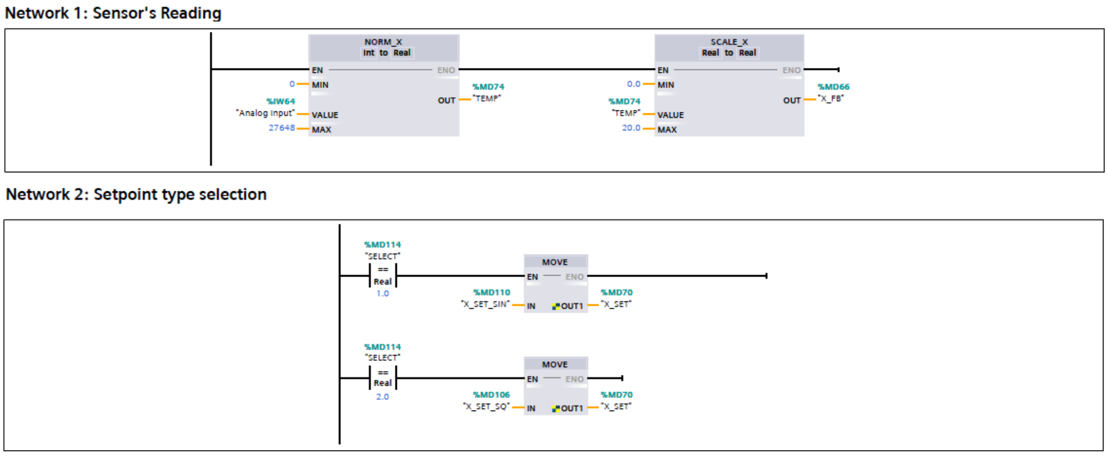


Figure 5.50: Network 1 and 2 of the ladder program for the Series LR digital proportional servo valves

Network 3, shown in Fig 5.51 transforms the value of the setpoint into the voltage sent to the Master Valve. First, the setpoint value is normalized between 0 and 20, that is, the piston's total stroke, then, this normalized value is escalated between 0 and 4096, which 0 is the minimum value and 4096 is the maximum value that can be expressed with 12-bit.

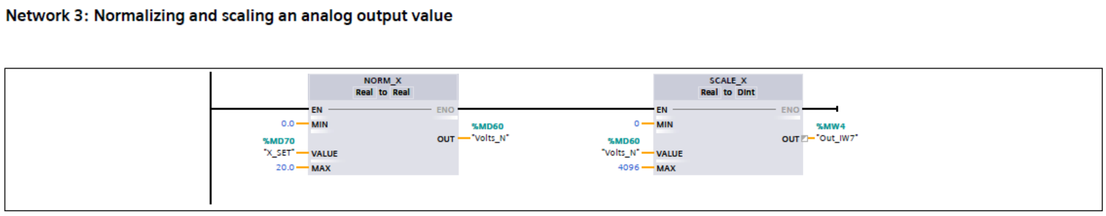


Figure 5.51: Network 3 of the ladder program for the Series LR digital proportional servo valves

When using 12 bits of resolution, the analog data occupies 2 bytes of memory and, in particular, the 12 most significant bits. The problem lies in the fact that the C0X6 module uses the little-endian encoding, and the PLC uses the big-endian encoding. This means that the first byte is actually saved as a second (already explained in chapter 3). The value is Rolled Right four times to reconstruct the data correctly to be read for the Camozzi module. After this, the Master valve gets a 0-10 signal reference input This process is shown in Fig 5.52

Network 4: Roll Right instruction

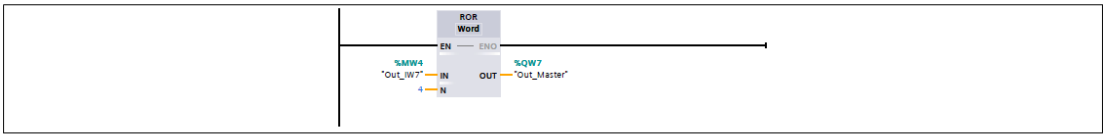


Figure 5.52: Network 4 of the ladder program for the Series LR digital proportional servo valves

5.4.1 Master-Slave configuration

Figure 5.53 represents the system's Master-Slave configuration and its respective connection with the PLC and the cylinder. The Master servo valve commands the airflow going through the cylinder's rear chamber, and the Slave valve commands the airflow of the front chamber. As can be seen in the figure, the PLC only sends the reference signal to the master valve. The pins of each valve are explained in chapter 3.

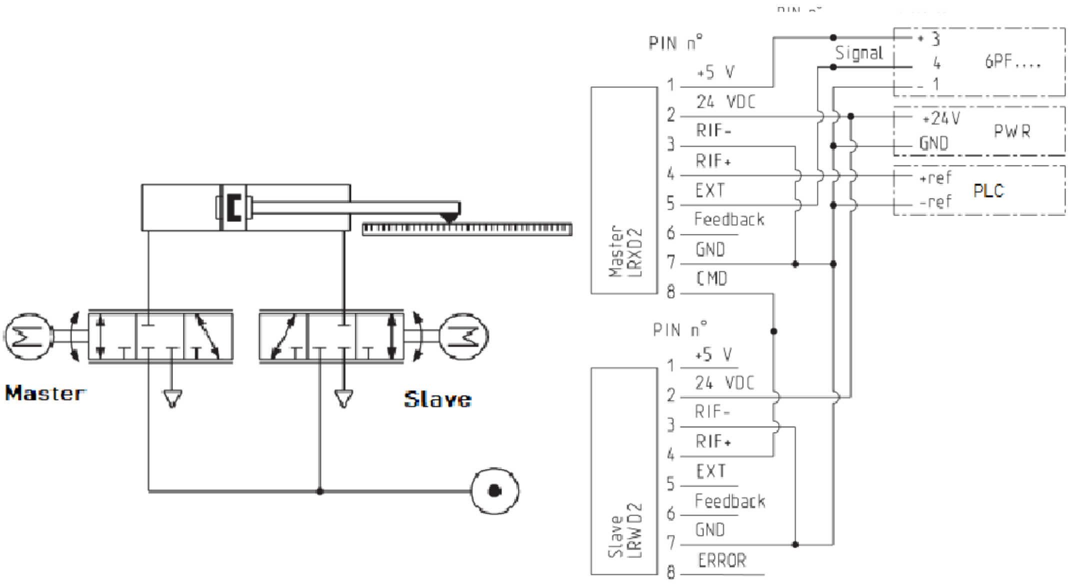


Figure 5.53: Master-Slave Configuration

As explained in chapter 3, the servo valve has its software called "LrxdConfigurator." This configurator can set the PID controller parameters in the "AUTO SETTING OF THE FEEDBACK EFFECT" window. The table 5.18 shows the different arrangement of PID parameters used for this system configuration.

- The K_c gain defines the distance from the samples used to calculated the

	K_p	K_c	K_y	K_i
PID SLOW	1	20	0.3	0
PID MEDIUM	3	20	0.5	0
PID FAST	8	20	0.6	0

Table 5.18: Auto settings Master-Slave Configuration

speed and acceleration of the system; this means that low KCL values create actions noisy but effective to dampen the abrupt output signal changes, while the high KCL values create less noisy actions but suitable for not excessively fast dynamics systems.

- The K_y proportional compensation gain modifies directly the control multiplying the control action when the position of the rod cylinder is close to the target. This gain is mainly used to avoid overshoots of the system in the case of masses to be moved. The K_y gain essentially allows to brake the movement of the rod with greater or lesser anticipation.

Fig 3 shows the dynamic position response while the system tracks a step-change in reference signal from 0 mm to 120 mm, Fig 2 shows the system's steady-state error, and the table 5.19 below summarizes the results obtained for this configuration.

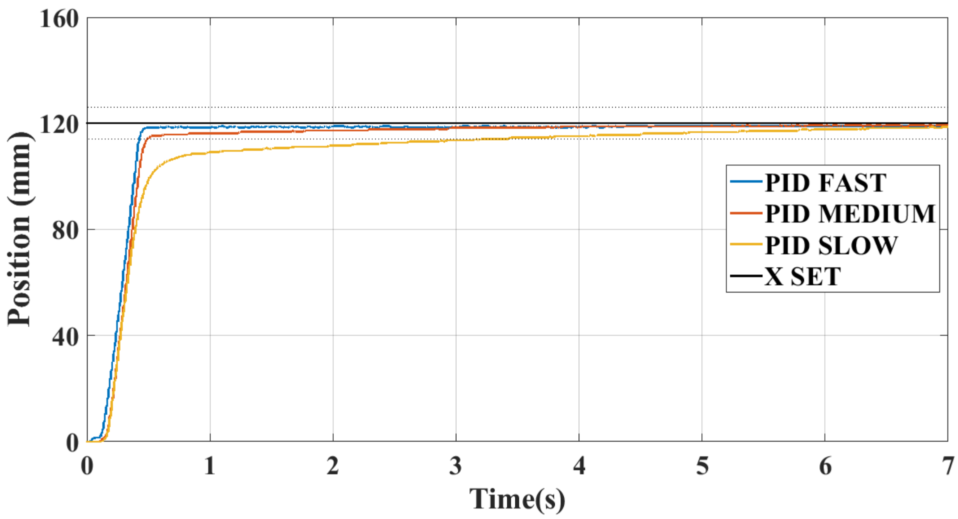


Figure 5.54: Dynamic position response while tracking a step-change in reference signal from 0 mm to 120 mm Master-Slave configuration

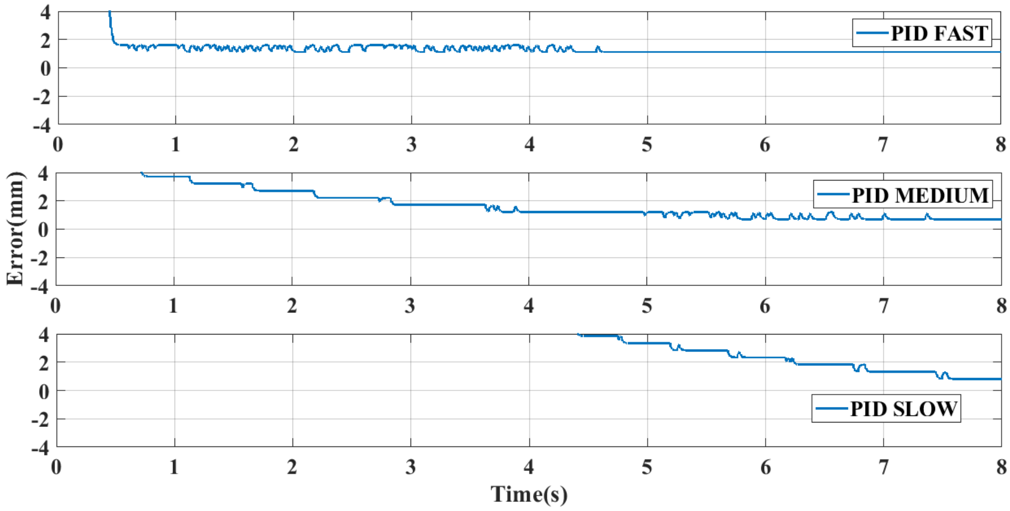


Figure 5.55: Error Master-Slave configuration

	SLOW	MEDIUM	FAST
ts(s)	3.36	0.51	0.42
error(s)	0.92	0.82	0.97

Table 5.19: Results using the Master-Slave Configuration

Fig5.56 shows the system's dynamic response when it tracks a sine wave of 140mm peak-to-peak amplitude and a period of 10 seconds and another of 140mm of amplitude and period 0.5s, using the PID FAST configuration. Moreover, it is easy to see that the system can track almost perfect the sinusoidal input when the period is big enough. Fig 5.57 shows that the error never exceeds the 10mm of error. On the other hand, with a period of 0.5 seconds, the system response presents an angle phase shift equal to $0.164\pi\frac{rad}{seg}$ or 29.52°.

Besides, Fig 5.58 shows the system's dynamic response when it tracks a square wave with 100mm of amplitude peak-to-peak and period of 0.5 seconds, and the second one is a square wave with 100mm of amplitude and a period of 4 seconds. It is clear that in the first experiment, the system barely reaches the setpoint before it changes again. On the other hand, whit the reference signal of period 4 seconds, it is notable that the system can track the input well enough. Fig 5.59 shows how the error remains at zero for almost the entire period of the square wave.

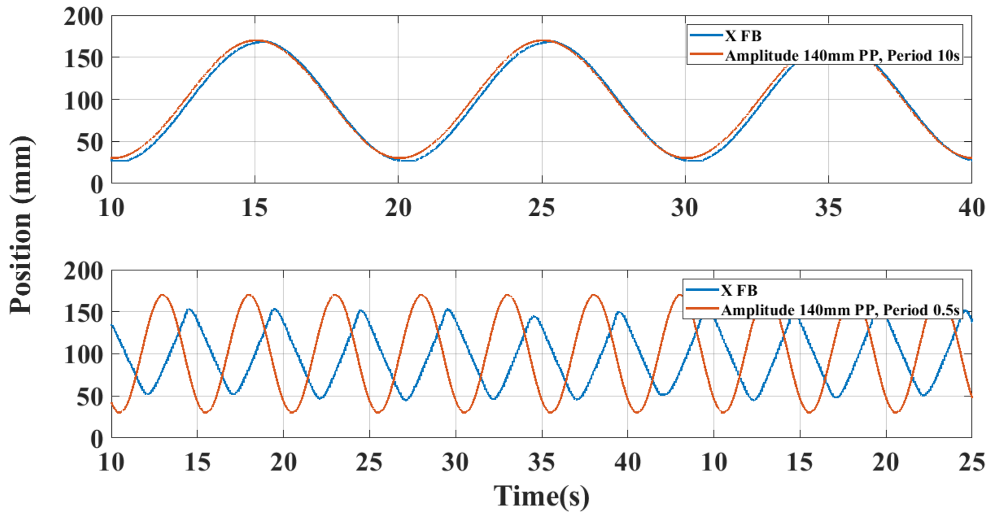


Figure 5.56: Dynamic position response while tracking a sine wave in reference signal of 140mm peak-to-peak amplitude

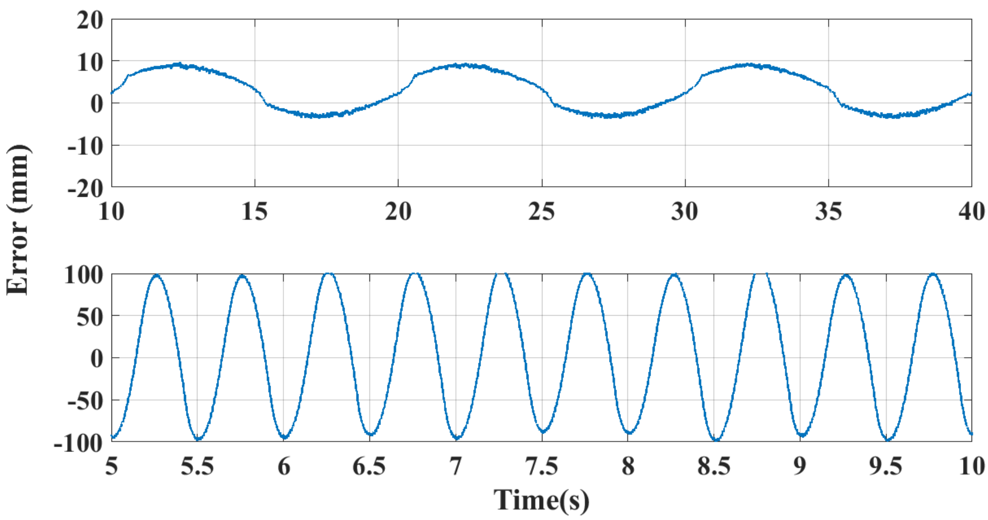


Figure 5.57: Error of sine wave as reference signal

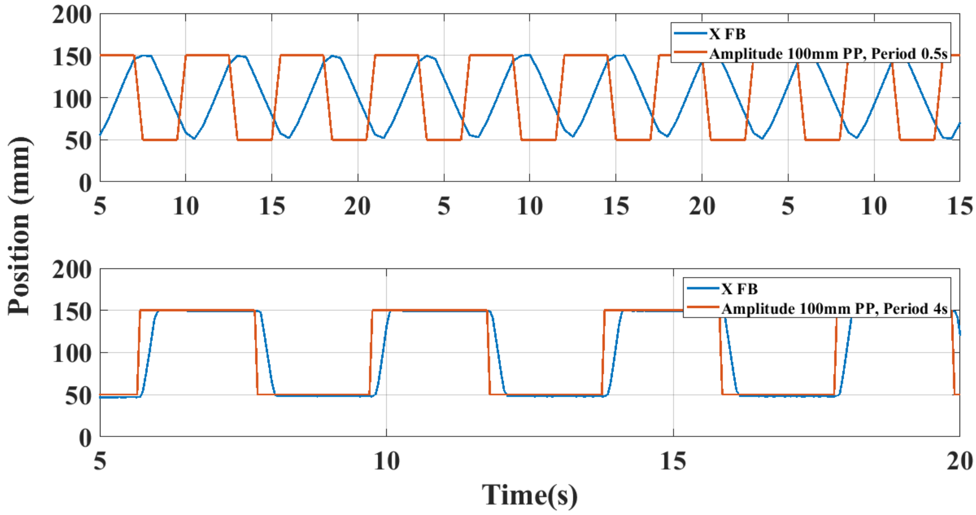


Figure 5.58: Dynamic position response while tracking a square wave in reference signal of 100mm peak-to-peak amplitude

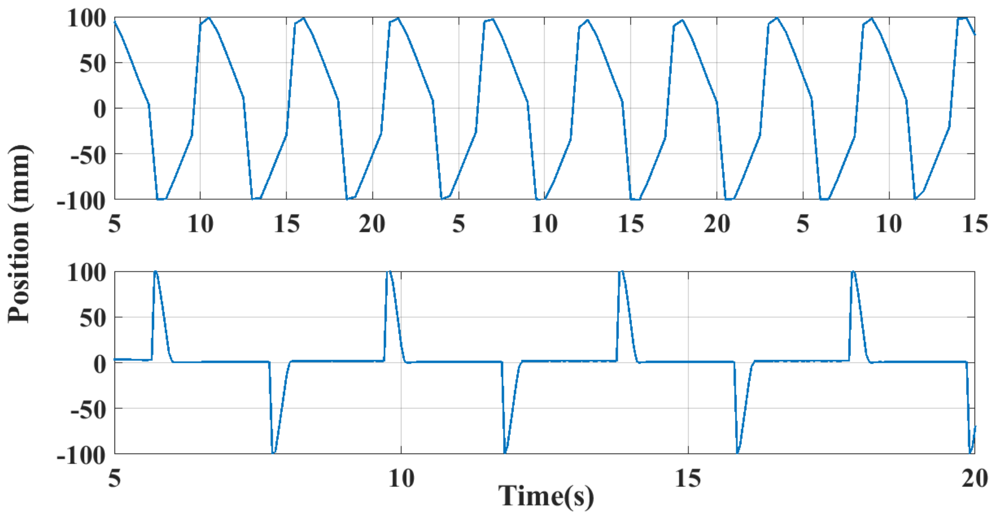


Figure 5.59: Error of square wave as reference signal

5.4.2 Master Valve and Pressure Regulator at the front chamber of the cylinder

This configuration allows working with just one servo valve. In this case, the Master Servo Valve is connected at the cylinder's rear chamber, and at the front chamber, there is a precision regulator. The pressure regulator imposes a fixed pressure in one of the cylinder chambers. When the master valve operates and moves the piston to reach the setpoint, the pressure regulator maintains a constant pressure that simulates a valve's operation in the other chamber of the cylinder. Fig 5.60 shows the system configuration and its respective wiring.

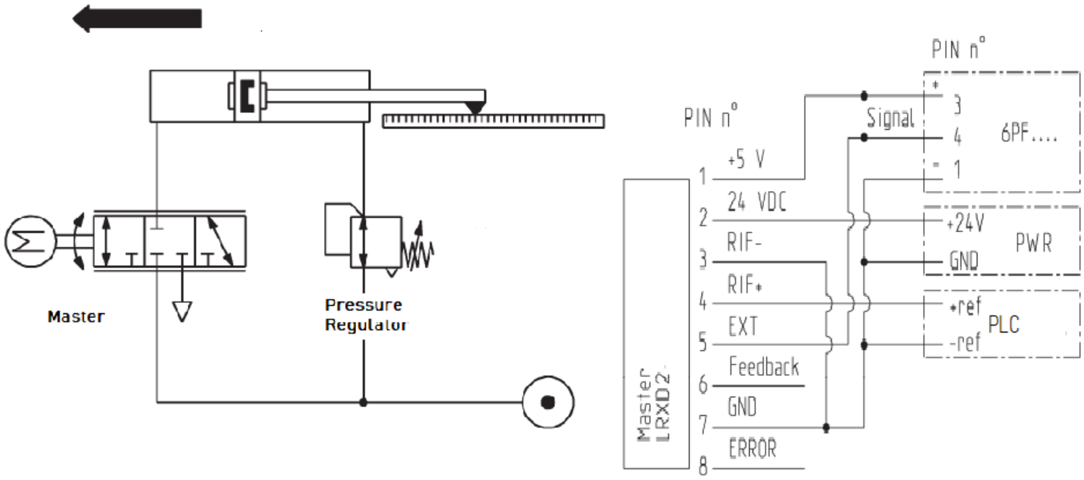


Figure 5.60: Master Valve and Pressure Regulator at the front chamber of the cylinder

Fig 5.61 and Fig 5.62 show the dynamic position response varying the regulator's pressure and the obtained error. It can be seen that as the pressure of the regulator increases, the response is more stable. In the first case, the pressure applied by the valve is greater than that of the regulator; therefore, an overshoot is generated, the system stabilizes when the master valve reduces the airflow to reach the setpoint. However, when the regulator's pressure is 5 bar, the response is slower, but overshoot is no present. Table 5.20 summarizes the results of this experiment.

	1 bar	**3 bar**	**5 bar**
ts(s)	0.503	0.48	0.75
error(mm)	0.38	0.4	2.125

Table 5.20: Results using the Master-PR at front chamber Configuration

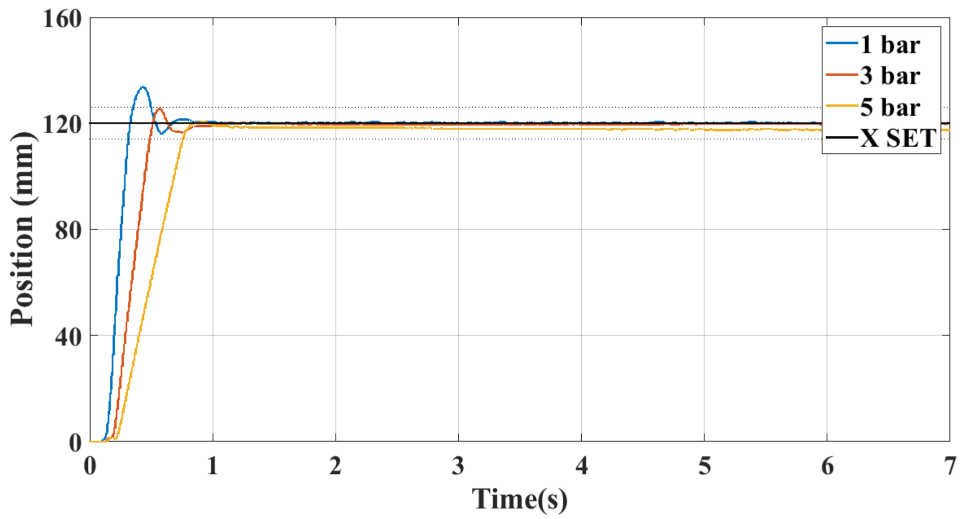


Figure 5.61: Dynamic position response while tracking a step-change in reference signal from 0 mm to 120 mm Master - Pressure Regulator at front chamber configuration

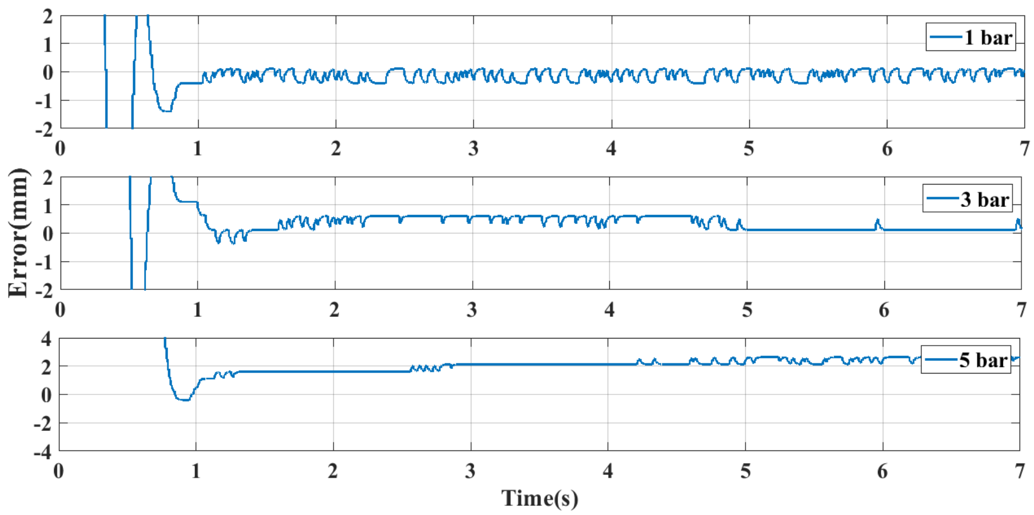


Figure 5.62: Error Master - Pressure Regulator at front chamber configuration

5.4.3 Master Valve and Pressure Regulator at the front chamber of the cylinder

This configuration (Fig 5.63) works like the last one; in this case, the pressure regulator is located at the cylinder's rear chamber. Figure 5.64 and Fig 5.64 shows the dynamic position response varying the regulator's pressure and the respective obtained error. As expected, when the pressure is set at 1 bar, the response is slow, and when the pressure is at 5 bar the response is faster but with an overshoot, the contrary as the past case.

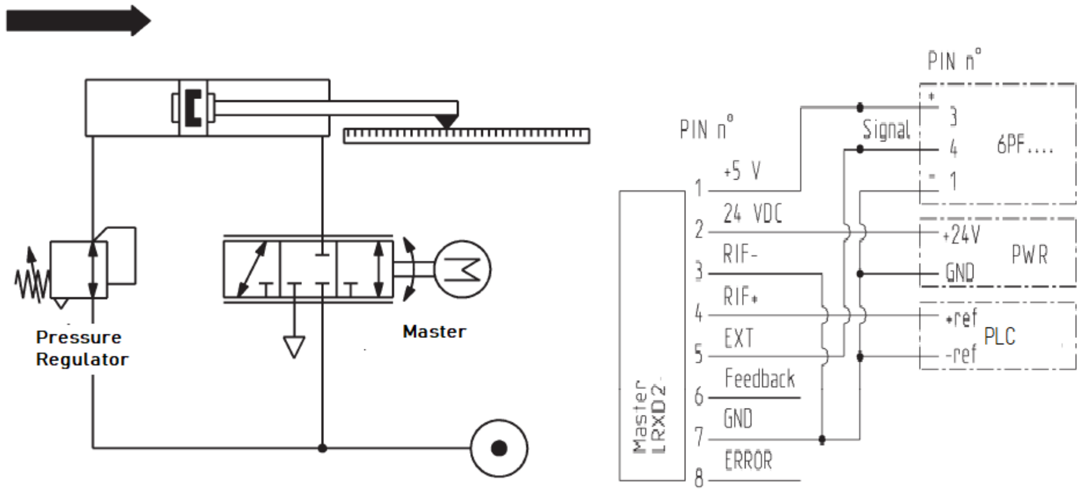


Figure 5.63: Master Valve and Pressure Regulator at the rear chamber of the cylinder

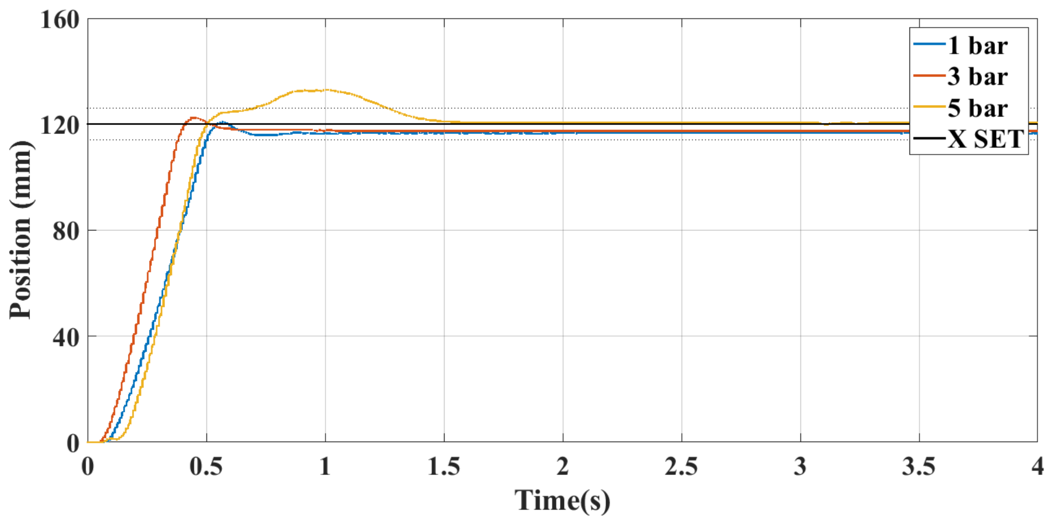


Figure 5.64: Dynamic position response while tracking a step-change in reference signal from 0 mm to 120 mm Master - Pressure Regulator at rear chamber configuration

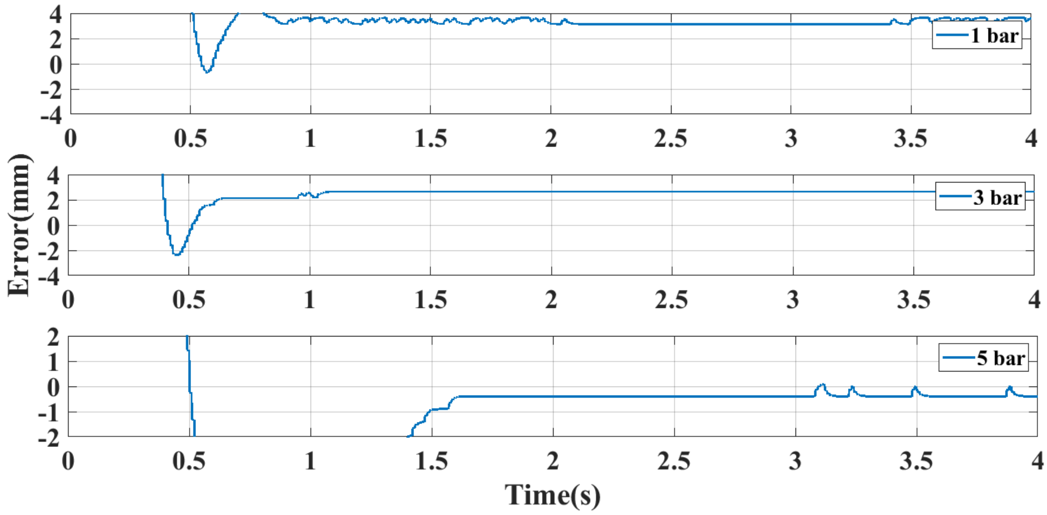


Figure 5.65: Error Master - Pressure Regulator at rear chamber configuration

	1 bar	3 bar	5 bar
ts(s)	0.58	0.39	1.26
error(mm)	2.95	2.62	0.39

Table 5.21: Results using the Master-PR at rear chamber Configuration

Chapter 6

Comparison between the experimental results

Below is a comparison employing a histogram, which clearly explains the results obtained in the different experiments.

6.1 5/3 closed center solenoid valve results

Figures 6.1 and 6.2 graphically compare the results obtained using the 5/3 valves. It can be noticed that despite being digital valves that do not have an airflow regulator, that is, they are fully open or fully closed, excellent results were achieved with this configuration. Regarding the settling time, times of 1.7 seconds were obtained, and regarding the error, errors of 0.45 mm were received. The best result taking into account time and error, was obtained using a proportional controller with $K_p = 6$, with this a ts of 3.71 seconds and a maximum error of 0.72 millimeters were obtained.

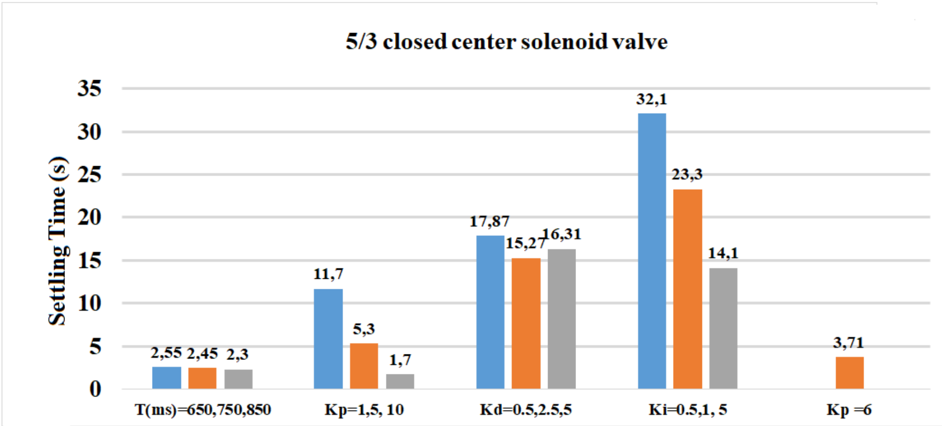


Figure 6.1: Settling Time Comparison Chart Using 5/3 closed center solenoid valve

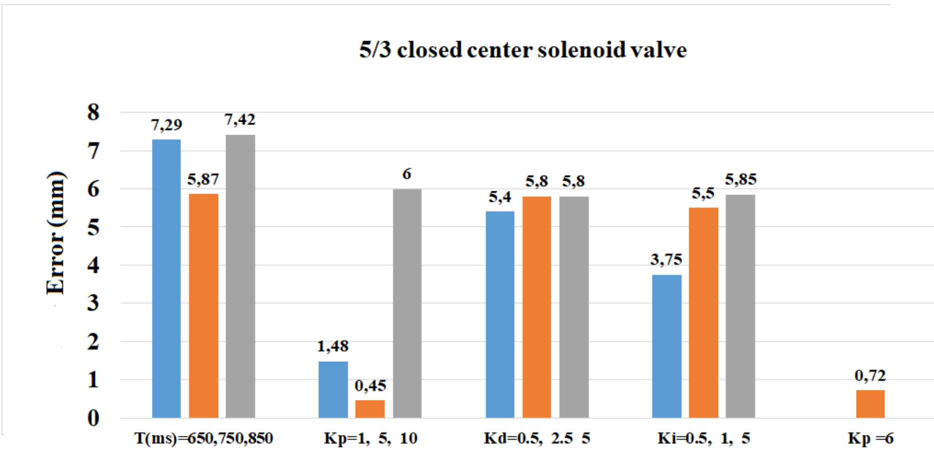


Figure 6.2: Error Comparison Chart using 5/3 closed center solenoid valve

6.2 Directly Operated Proportional Valves results

Figures 6.3 and 6.4 graphically compare the results obtained using the Directly Operated Proportional Valves. The improvement is notable with this type of valve concerning the previous configuration since its opening is proportional to an analog value inlet. In this case, settling times equal to 1.2 seconds and steady-state errors of 0.1 millimeters were achieved. These results were obtained with a PID controller with parameters equal to Kp = 15, Kd = 1, and Ki = 0.1.

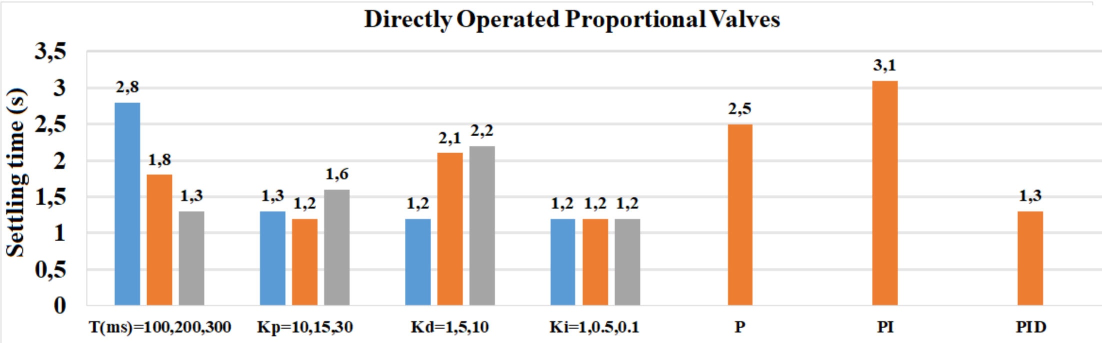


Figure 6.3: Settling Time Comparison Chart Using Directly Operated Proportional Valves

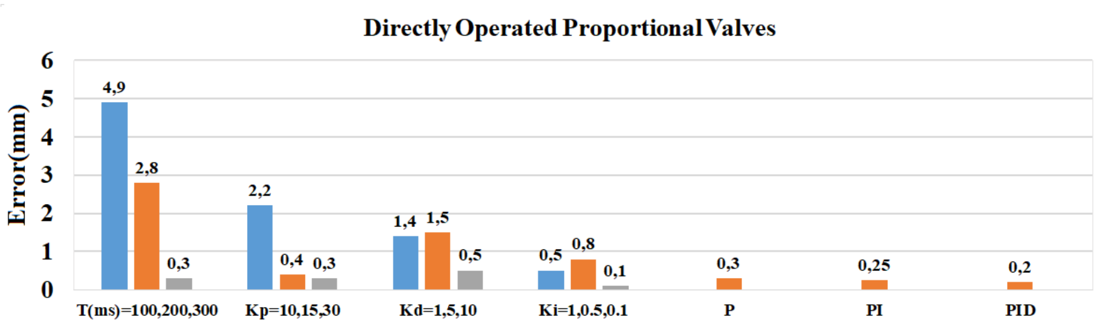


Figure 6.4: Error Comparison Chart Using Directly Operated Proportional Valves

6.3 Electronic Control Device for Proportional Valves results

Figures 6.5 and 6.6 graphically compare the results obtained using the Electronic Control Device for Proportional Valves. The results using the driver for the proportional valves are not very different from the previous results. Regarding the settling time, times of 1.12 seconds were obtained, and regarding the error, errors of 0.18 mm were reached. The best result taking into account time and error, was obtained using a PDI controlled with Kp = 15, Kd=1, and Ki=1. With these parameters, a ts of 1.19 seconds and a steady-state error of 0.4 millimeters were obtained.

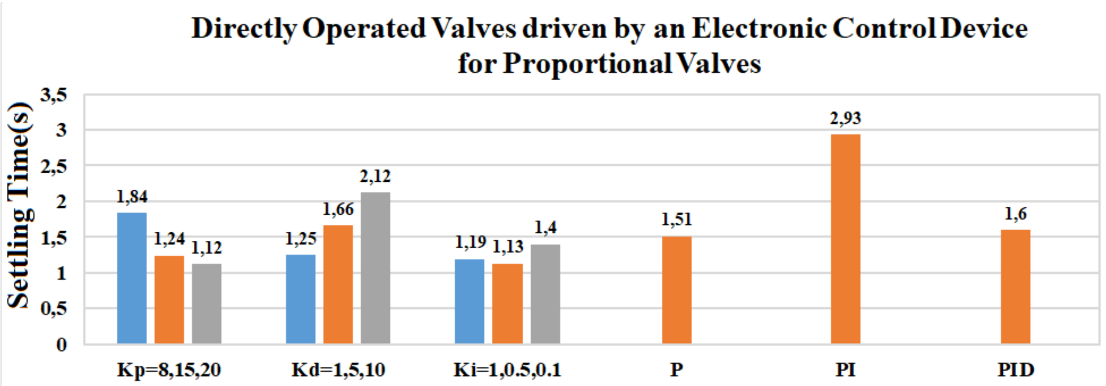


Figure 6.5: Settling Time Comparison Chart Using Directly Operated Valves driven by an Electronic Control Device for Proportional Valves

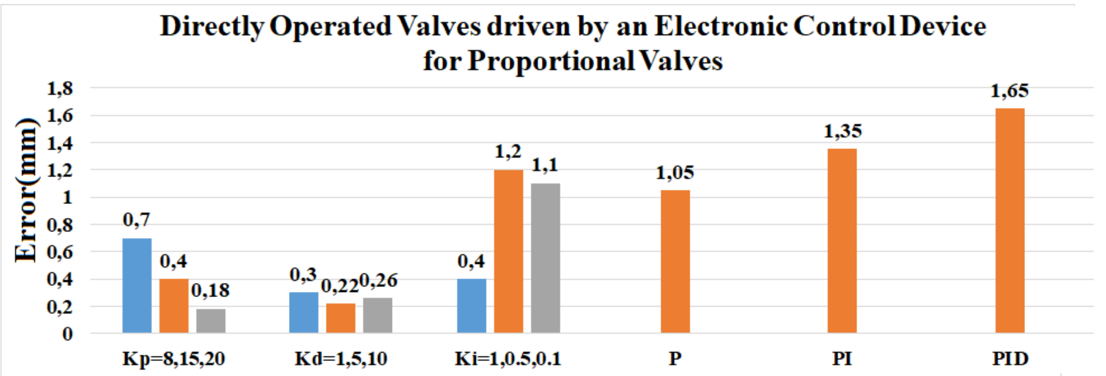


Figure 6.6: Error Comparison Chart Directly Operated Valves driven by an Electronic Control Device for Proportional Valves

6.4 Series LR digital proportional servo valves results

Figures 6.7 and 6.8 graphically compare the results obtained using the Series LR digital proportional servo valves. The results using the servo valves show an improvement for the settling time, as it is reduced by almost one second to the past settings. Regarding the settling time, times of 0.39 seconds were obtained, and regarding the error, errors of 0.38 mm were reached. The best result taking into account time and error, was obtained using a Master Valve and a Pressure regulator at the front chamber with a fixed pressure of 3 bar. With configuration, a ts of 0.48 seconds and a steady-state error of 0.42 millimeters were obtained.

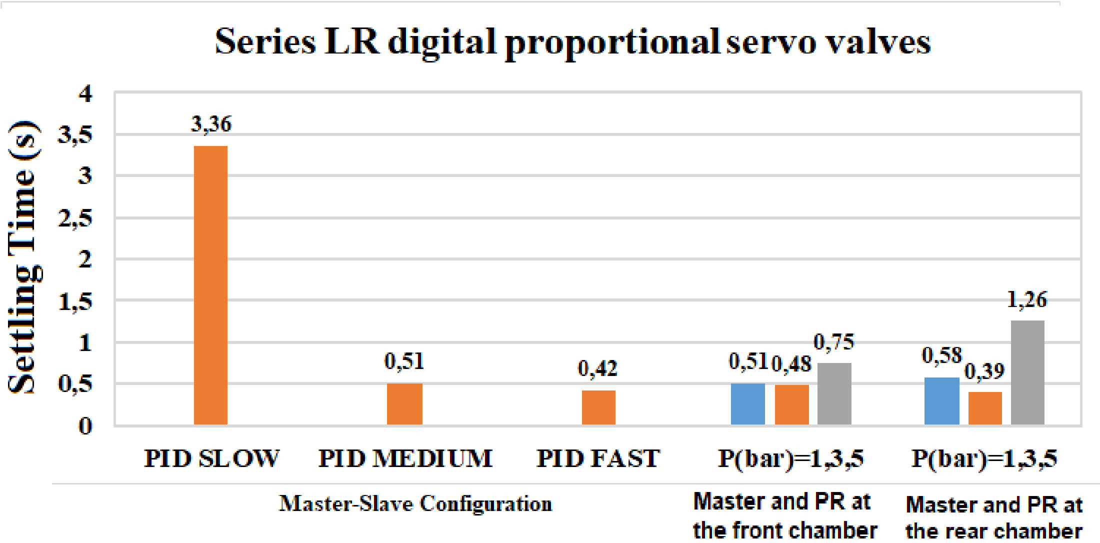


Figure 6.7: Settling Time Comparison Chart Using Series LR digital proportional servo valves

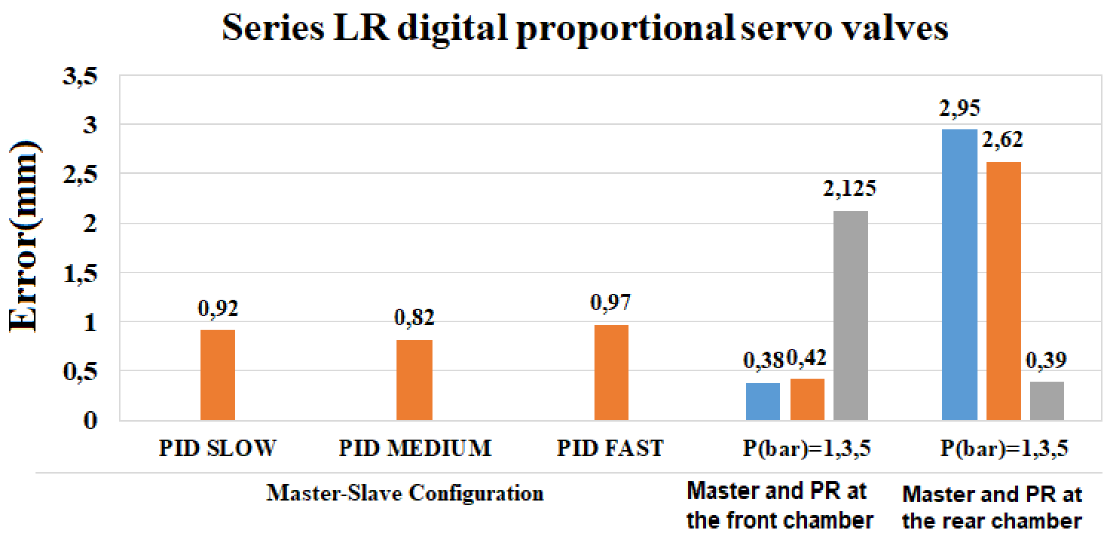


Figure 6.8: Error Using Series LR digital proportional servo valves

Chapter 7

Conclusions and future developments

The present thesis work was born with the idea, as already mentioned, of developing a single-rod, double-acting cylinder positioning control by means of a PLC. The experiments were carried out on a test bench provided by DIMEAS, which has different pneumatic actuators, making it possible to analyze the behavior of the different configurations applying diverse control methods.

The system composed of the PLC, the cylinder, the serial module, and the different valves presented good behavior. Moreover, the experiments clarify that Camozzi CX06 serial module does not introduce any delay to the system since the behavior was the same when the valves were controlled through the serial module or directly from the PLC.

As shown in the previous chapter, for the configuration that uses the 5/3 closed center solenoid valve, taking into account the settling time and the error, the best results are 3.71 seconds and a maximum error of 0.72 millimeters, using as control method a PWM driven by a PID controller. On the other hand, for the Directly Operated Proportional Valves, results of 1.2 seconds for the settling time and steady-state errors of 0.1 millimeters were achieved, using the CTRL_PWM instruction driven by a PID controller. Finally, for the Series LR digital proportional servo valves, times of 0.39 seconds were obtained, and regarding the accuracy, errors of 0.38 mm were reached.

Therefore, it is possible to conclude by saying that the best performed is the one managed by the Directly Operated Proportional Valves using the CTRL_PWM instruction driven by a PID controller.

Thinking about future developments, however, it would be interesting to control the cylinder using the two servo valves in slave mode. In this case, the control loop must be programmed inside the PLC. This time the valves receive a signal in the range 0-10Vdc from the PLC, and they, in turn, regulate the airflow in and out of the actuator's chambers based on that signal. Since the experiments realized on the didactic bench exposed during the thesis will help students with academic exercises during fluid automation and industrial automation courses, the addition of an HMI would help understand its configuration and functioning. The HMI will make the system user-friendly, and once the program is upload to the PLC, there is no need for a computer since the system can be controlled, monitored, and changed through this Human Machine Interface. Finally, it would be interesting to analyze how the system behaves when it has to move an external load.

Chapter 8

References

1 Krivts, I. and Krejnin, G., 2006. Pneumatic actuating systems for automatic equipment. Boca Raton: Taylor & Francis.

2 Saravanakumar, D., Mohan, B. and Muthuramalingam, T., 2017. A review on recent research trends in servo pneumatic positioning systems. Precision Engineering, 49, pp.481-492.

3 Satyendra, K., 2015. Basics of Pneumatics and Pneumatic Systems – IspatGuru. [online] Ispatguru.com. Available at: <https://www.ispatguru.com/basics-of-pneumatics-and-pneumatic-systems/>s/

4 Ryan, K., 2020. What Is Servo Pneumatics? - Pneumatic Guides - Rowse Pneumatics. [online] Rowse-pneumatics.co.uk. Available at: <https://www.rowse-pneumatics.co.uk/blog/post/what-is-servo-pneumatics>].

5 Lab-Volt, L, 2002. APLICACIONES DE HIDRÁULICA CONTROL SERVO PROPORCIONAL. 1st ed.

6 It.automation.camozzi.com. 2021. Camozzi Automation. [online] Available at: <https://it.automation.camozzi.com/chi-siamo/camozzi-automation.kl>.

7 Wang, J., Pu, J. and Moore, P., 1999. Accurate position control of servo pneumatic actuator systems: an application to food packaging. Control Engineering Practice, 7(6), pp.699-706.

8 Aziz, S. and Bone, G., 1998. Automatic tuning of pneumatic servo actuators. Advanced Robotics, 13(6), pp.563-576.

9 Muzy, Gabriel & Caporali, Alexandre. (2018). Positioning System of a Pneumatic Actuator Driven by Proportional Pressure Regulator Valves. 11-16.

10 Song, Qiang & Liu, Fang. (2006). Improved Control of a Pneumatic Actuator Pulsed with PWM. pp. 1 - 4.

11 D.W. Clarke, C. Mohtadi and P.S. Tuffs, "Generalized Predictive Control Part 1: The Basic Algorithm, Part 2: Extensions and Interpretations", Automatica, Vol. 23, No. 2 1987, pp.137-160

12 Shen, Xiangrong & Zhang, Jianlong & Barth, E.J. & Goldfarb, M. (2004). Nonlinear averaging applied to the control of pulse width modulated (PWM) pneumatic systems. Proceedings of the American Control Conference. 5,pp 4444 - 4448 vol.5.

13 Chieh-Li Chen, Pey-Chung Chen, Cha'o-Kuang Chen, A pneumatic model-following control system using a fuzzy adaptive controller, Automatica, Volume 29, Issue 4, 1993, pp 1101-1105.

14 Shen, X., and Goldfarb, M. (October 4, 2006). "Energy Saving in Pneumatic Servo Control Utilizing Interchamber Cross-Flow." ASME. J. Dyn. Sys., Meas., Control. May 2007; 129(3): pp.303–310.

15 Chiang MH. Development of X-Y servo pneumatic-piezoelectric hybrid actuators for position control with high response, large stroke and nanometer accuracy. Sensors (Basel). 2010; pp.2675-93.

16 Camozzi, 2019. Modulo multi-seriale Serie CX. Use and Maintenance Manual.

17 Politecnico di Torino, n.d. PLC Siemens S7-1200 & Camozzi serial node (Profinet protocol).

18 Camozzi, 2019. Cilindri Positioning Feedback Serie 6PF. Use and Maintenance Manual.

19 IEC, IEC 61131-1, Programmable Controllers, Geneva: International Electrotechnical Commission, 2003-2005

20 Siemens, 2012. S7-1200 Programmable controller. SIMATIC System Manual.

21 Camozzi, 2019. Series 3 Valves and Solenoid Valves. Use and Maintenance Manual.

22 Camozzi, 2019.Series AP Directly Operated Proportional Valves. Use and Maintenance Manual.

23 Camozzi, 2019.Series AP Directly Operated Proportional Valves. Use and Maintenance Manual.

24 Camozzi, 2019. Tecnologia proporcional. Catalogo.

25 Camozzi, 2017. Istruzioni d'uso e manutenzione Servovalvola serie LRXD. Use and Maintenance Manual.

26 Camozzi, 2016. Istruzioni d'uso e manutenzione Servovalvola serie LRWD. Use and Maintenance Manual.

27 Camozzi, 2019. Series PR precision regulators with manual override. Use and Maintenance Manual.

28 Ryniecki, A., 2015. Basics of Process Control: the On-Off Control System. PRZEMYSŁ SPOŻYWCZY, 1(11), pp.28-31.

29 Patel, Mahesh & Patel, Ankit & Vyas, Dhaval & Patel, Ketul. (2009). Use of PWM Techniques for Power Quality Improvement. POSTER PAPER International Journal of Recent Trends in Engineering. 1. 99-102

30 Escuela Politécnica Nacional, 2015. ANALISIS DE LA RESPUESTA TRANSITORIA.

31 Morilla García, F., 2007. Fundamentos de los controladores PID.

32 Viktorov Vladimir, Colombo Federico (2013), Automazione dei sistemi meccanici: corso di base, CLUT, Torino.