Low-cost proportional automatic control valve

Mariana Arboleda Otero[a],

Santiago Builes Toro[b]

[a] *Process engineering student, EAFIT University, Medellin, Colombia*
[b]*Professor, Degree project advisor, Process engineering department, EAFIT University, Medellin, Colombia*

Sumary

Control and automation are the keys to escalate to industry 3.0, in a process the control valves are responsible for performing a corrective action in the process stream against a set point in the PID control loop, unfortunately its high cost does not allow the implementation of these equipment in academia and small industries. This project was based on the development of a functional prototype in hardware and software of a proportional valve for low-cost automatic control, coupling a conventional valve found in the market configured with microcontrollers, obtaining a low-cost device compared to those existing in the market that satisfy similar needs. The project was divided into four stages: i) mechanical coupling of the valve and the stepper motor, ii) electrical connection of the motor with the Arduino microcontroller, iii) programming the code to control the percentage of valve opening in direct relation with the number of steps of the motor, iv) transcription of the programming code to the Raspberry microcontroller to control the valve with the cell phone. A functional prototype with rapid response capacity was made and five buttons were set for the opening percentage of 0, 25, 50, 75, 100 %. In addition to this, it was possible to build a valve cell interface with a Raspberry microcontroller run through Google's mobile app, Home Assistant, to remotely control the valve.

1. Introduction

Today, industrial processes require control and automation, which is based on the implementation of hardware that performs corrective actions directly on the processes and software that allows decisions to be made through a programming code with certain established needs. They allow to increase efficiencies, minimize errors, obtain higher quality, standardization of processes and reduce low-skilled labor.

The figure 1. shows a reboiler system with PID control, it allows to control the temperature in the outflow *T(t),* determining the reference value *r(t)* sending the signal *s(t)* to the microcontroller to regulate the opening percentage in the valve. For instance, if the variable is lower than the desired temperature *T(t)* (is negative) the variable is the difference between the actual temperature and the desired one so the valve allows more recirculation in the boiler equipment [1]

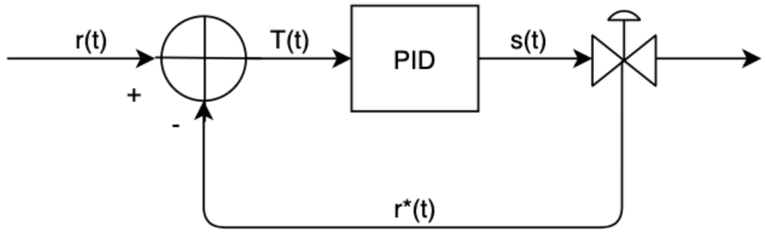


Figure 1. Closed loop PID control system

Valves are responsible for the action of controlling, and they are the only ones that have direct contact with the process streams in the control loop, to compensate for the load disturbance and keep the regulated process variable as close as possible to the desired set point [2]. The purpose of the valves is to regulate the flow, maintaining the desired system operating conditions. These control valves represent approximately 95% of the final control elements used in industry [3].

The operation of the valves consists of coupling the body of a control valve with an actuator that gives the electrical, pneumatic o hydraulic signals for the operation of the valve, and the accessories to be added such as pressure sensors, limit switches and whatever is required according to the user's needs [2].

The actuators can be a pneumatic, that requires a compressed air system, hydraulic, that works with water, or electrical, that needs electrical current to work; actuators that supplies force and motion to open or close a valve [4]. In the case of electric actuators, a good alternative is the use of microcontrollers, these are an integrated circuit that contains a central processing unit (CPU) inside, memory units (RAM and ROM), input and output ports and peripherals. These parts are interconnected within the microcontroller, and together they form what is known as a microcomputer [5], allowing to perform the necessary electrical functions through a programming code, Furthermore, most of these microcontrollers are inexpensive, priced below $ 10.

Microcontrollers have been used for the redesign of valves for automatic flow control, for example the work presented by Guerrero [6] who developed a control valve prototype controlled via a stepper motor and a microcontroller. Gears were used to increase the torque force and reduce the forces made by the motor with respect to the speed of rotation. This is the main difference with my prototype that eliminates the use of gears and makes direct connection of the motor and the valve, allowing to control the valve opening percentage in direct relation to the number of motor steps.

Henao et al. [7] developed an automatic control proportional valve, using a gate valve operated by an Arduino microcontroller, The prototype presented in this project is based on this work, however the main differences are that the buttons for certain opening percentages are already set and the interface with the cell phone is made to remotely control the valve. In addition, the project is developed at the EAFIT University of Medellin, Colombia, for the research hotbed of Development of industrial products - DPI.

The project presented in this work, present a proportional control valve prototype, programed with Arduino and Raspberry microcontrollers and adds an interface controlled by cellphone, allowing the remote control of the valve. A program with the values of five opening percentages of the valve in relation with the motor´s steps is implemented as default in the cellphone interface.

2. Materials and methods

Figure 2. shows the graphic design of the prototype of the initial valve, which shows the direct connection of the stepper motor with the gate valve and the connection with the microcontrollers.

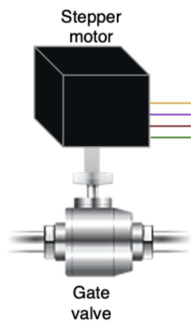


Figure 2. Prototype connection design of the valve with the stepper motor and the Arduino board

For the mechanical coupling of the valve, a Hammond gate valve LBS 300, 3/8” of diameter was purchased; a stepper motor 5.6kg cm Nema 17 of which the mobile end was welded with the upper part of the valve, the handle. Figure 3. shows the coupling of both equipment made with welding.

Figure 3. Mechanical accoupling of valve and stepper motor with welding

For the electrical connection of the motor to the Arduino [8], an Arduino UNO card was used, with a Driver A4988, which allows to control the motor, This controller is chosen because it is easy to find in the market and is recommended for its reliability and low cost. A heatsink[9] that helps prevent overheating in the driver, additionally an external power source with 12V voltage and 2 A current, is required because the motor consumes 1.68 A per phase. A 100μF electrolytic capacitor, recommended by the manufacturer of the A4988, that allows to store energy and provide it for periods if the motor requires it. Finally a breadboard that allows to carry out all the connections. Figure 4. shows the connection allowed to control the motor from the Arduino board. [8]

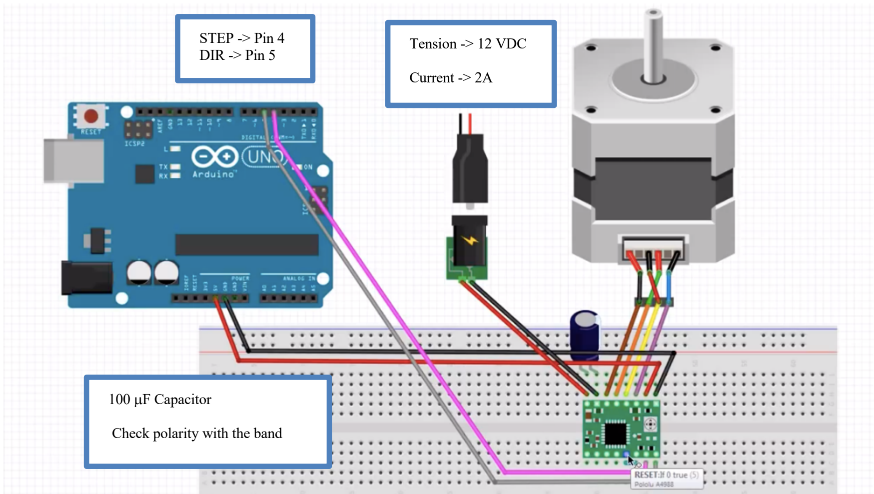
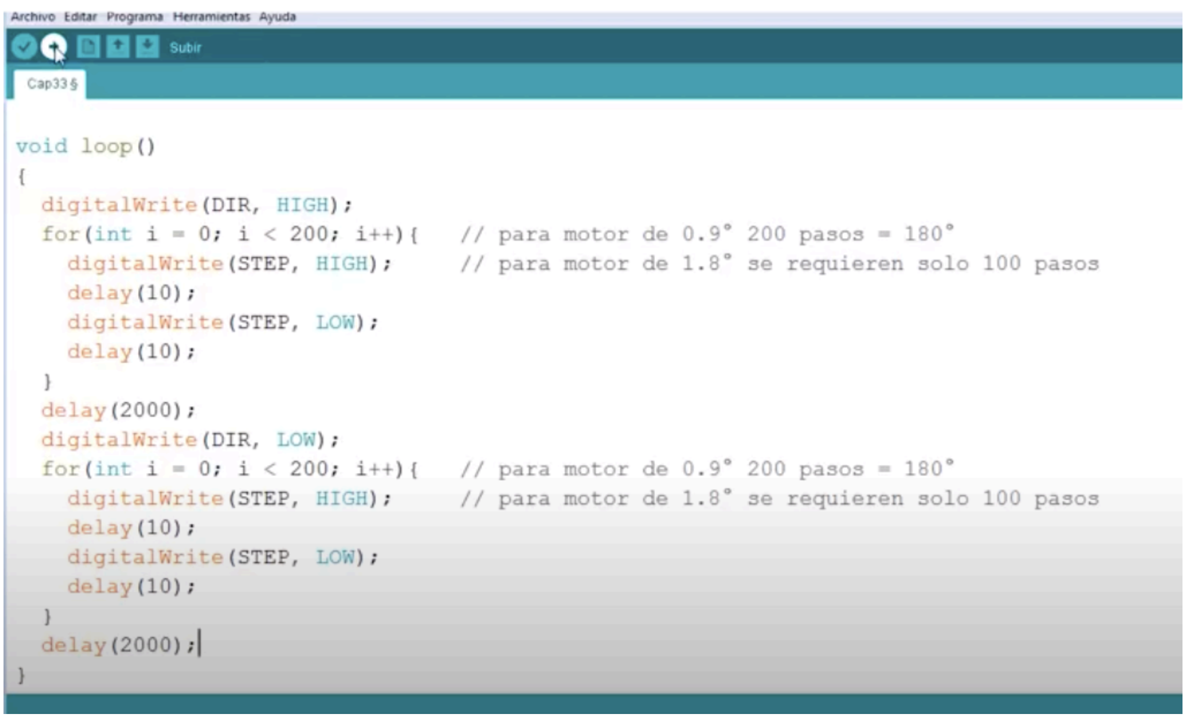


Figure 4. Connection of bipolar stepper motor and A4988 Driver with Arduino

The next step was to write the code to control the valve opening percentage, initially the steps of the motor are programmed to allow the valve to be closed and opened completely, For this, the code for a bipolar stepper motor found on the Arduino open platform shown in Figure 5. was taken as a reference[10]. Due to laboratory use restrictions to make the equipment generic for any fluid and process topology, it was only programmed the valve´s opening percentage instead of characterizing the flow of a particular fluid and piping topology.




```
void loop()
{
  digitalWrite(DIR, HIGH);
  for(int i = 0; i < 200; i++){    // para motor de 0.9° 200 pasos = 180°
    digitalWrite(STEP, HIGH);      // para motor de 1.8° se requieren solo 100 pasos
    delay(10);
    digitalWrite(STEP, LOW);
    delay(10);
  }
  delay(2000);
  digitalWrite(DIR, LOW);
  for(int i = 0; i < 200; i++){    // para motor de 0.9° 200 pasos = 180°
    digitalWrite(STEP, HIGH);      // para motor de 1.8° se requieren solo 100 pasos
    delay(10);
    digitalWrite(STEP, LOW);
    delay(10);
  }
  delay(2000);
}
```

Figure 5. Stepper motor reference code used for controlling the gate valve.

With the motor running, the programming code was modified for the exact number of steps to obtain a certain percentage of opening in the valve; setting five values for 0, 25, 50, 75 and 100% of the valve opening.

Once with the motor running, the code was transcribed to the Raspberry PI microcontroller, for this, the ESP8266 WiFi module was purchased, this low cost module of no more than 3 USD, allows to obtain WiFi connectivity capacity to our project [11], connect to a router, which may be the one at home. The Annex 1. shows the code programmed that allows to control the Arduino board with the ESPHOME, it controls de number of steps and the turning direction of the stepper motor. Figure 6. shows the connection that was made from the Arduino to the ESP8266 [12]. Then with the Google mobile application that works with ESPHOME, Home Assistant, which is an application in which household appliances are connected to have control of these from the cell phone, it has the cell-valve interface and gets remote control of it.

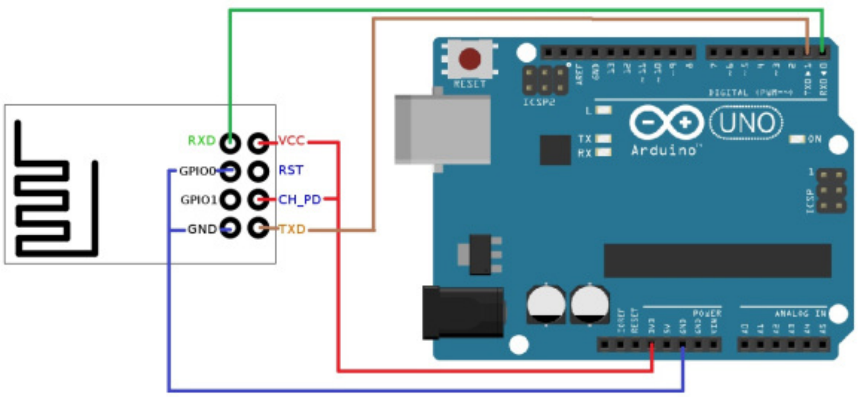


Figure 6. Arduino and ESP8266 connection

With the code running in ESPHOME continues with the controlling and automatization of the valve, shown in Annex 2. that allows to configurate the ESPHOME, with the code showed in Annex 1, to be read through Home Assistant application, and Annex 3. that automates the values of the opening percentages of the valve in relation to the number of motor´s steps. It allows to configure the five desired buttons corresponding to the opening percentage and also to set any other specific value of the opening percentage of valve.

Finally, the outer appearance of the valve was designed, The idea was based on introducing all the electrical part and the upper part of the motor in a container box that is installed on the prototype's head as shown in figure 7.

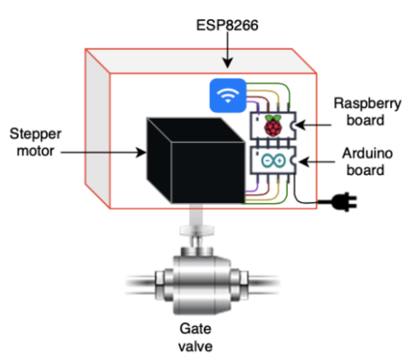


Figure 7. Final prototype design

3. Results and analysis

The purpose of the project presented here was the development of a low-cost automatic control proportional valve. For which a mechanical coupling was initially made with welding of the valve head with the upper part of the motor, the electrical connection of the motor and the Arduino to control the steps and the turning direction to determine the function of the stepper motor. Once the code was written and the motor was working, the Arduino code was transcribed to the Raspberry microcontroller through the ESP2866 wifi module that allowed to program the exact numbers of motor´s steps to achieve five values that indicate the valve opening percentage in relation of the number of the motor´s steps (0, 25, 50 75, 100%) having the code that controls the valve in the network, and thus configure on the cellphone with the application Home Assistant remote valve operation.

Finally, the collection of all the parts that make up the valve was carried out and they were placed on a wooden support that allowed the valve and the motor to be fixed and to be able to show its operation without requiring to connect it to a pipe. The initial idea was to put the parts into a container box on the head of the valve, but this support allows to show the electrical connection of the valve and the microcontrollers. Then it would be put into the box. Figure 8 shows the valve prototype in the wooden support.

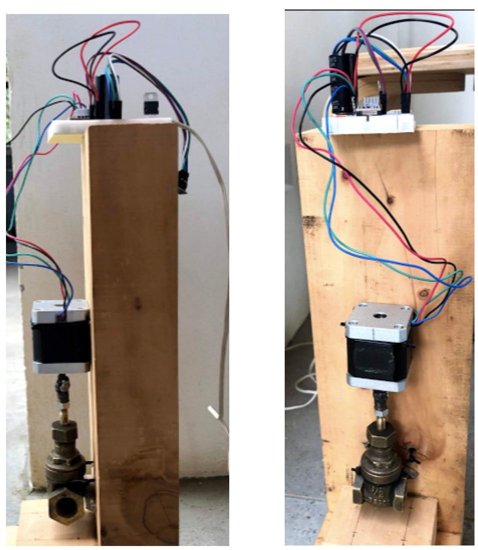


Figure 8. Low-cost proportional automatic control valve prototype

The cellphone interface was created in the Home Assistant application, it was carried out with the ESPHOME module and programed with the Arduino and Raspberry Pi microcontrollers, it has the five buttons of the selected values of the opening percentage of the valve and also a section where the user can write an specific percentage value, figure 9 shows the appearance of the interface in the computer and the application Home Assistant can be downloaded in the cellphone and the interface has the same appearance and the valve can be controlled through it.

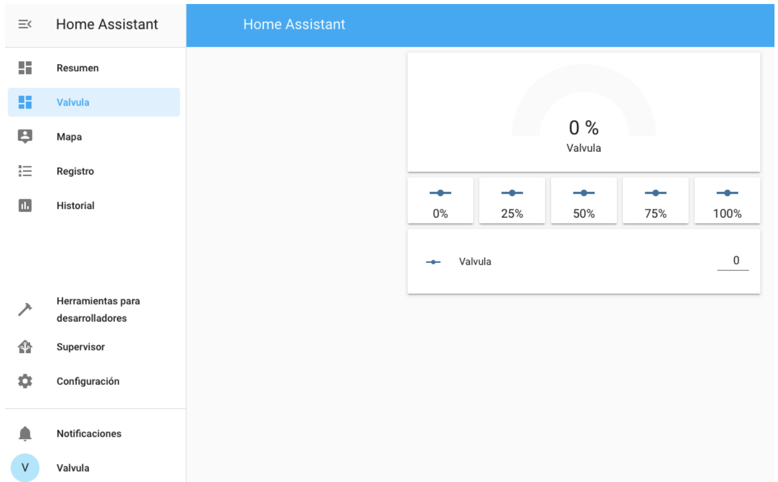


Figure 9. valve interface in a computer, the first box shows the actual opening percentage, the second box shows the five selected buttons for the opening percentage of the valve and in the third box the user can write an opening percentage of the valve different to the buttons

A flow test was done to compare the opening percentages fixed in the five preset buttons. The set-up used for testing is shown in figure 10. The time it takes to 300 ml of fluid (water + colorant) to pass from A to B with the five different opening percentages was determined.

Figure 10. Flow test to compare the times with the different opening percentages

The following results were obtained after doing the test ten times for each opening percentage Table 1.

Average time	Valve Opening %
7,35	25
7,10	50
6,97	75
6,62	100

Table 1. Average passage times from A to B for 300 ml of fluid, flow test for each percentage of opening

These values were graphed to buy the opening percentages and their times in the flow test, figure 11. In which it is possible to show the descent slope, which corroborates that the higher the opening percentage, the lower the passage time through the assembly done. This corroborates the characteristic curve of the valve, which has a linear behavior depending on the valve opening speed. In the control valve, all the opening percentages are carried out at the same engine speed.

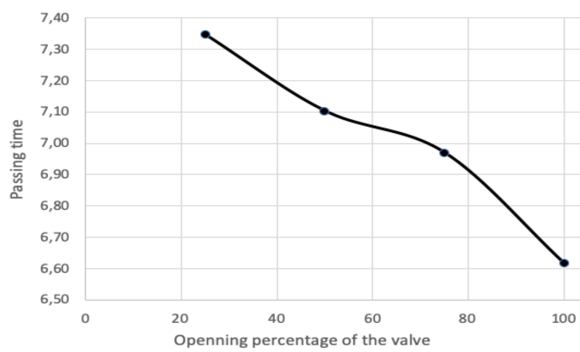


Figure 11. Graph of passage times vs. opening percentage in baldimetry test

To compare the cost of the developed valve with a valve with similar characteristics, a comparative analysis was performed using the cost of the materials shown in Table 2.

Table 2. Cost in USD of the elements requires for the construction of the prototype. (1USD= 3858,48COP)

Element	Cost (USD)
Gate valve	20,71
Stepper motor	19,44
Driver A4988	1,89
Arduino Uno R3	6,74
100µF electrolytic capacitor	0,05
Cables	1,30
Protoboard	1,43
ESP8266	2,58
Raspberry PI 4b 2g	58,90
Wireless router	25,92
total	**138,96**

The cost of an automatic control proportional valve in the market is around 500 USD[13], with an approximate savings of 72% in the purchase of the final control equipment, that is, it is possible to build around seven control valves as the ones developed in this work for less than the cost of one commercial valve..

The prototype elaborated in this work fulfilled the initial objectives, elaborating an automatic control proportional valve, programmed with the Arduino and Raspberry microcontrollers, with five buttons that control the valve opening percentage in relation to the motor steps, having the remote control of the valve with the cell-valve interface through Home Assistant and all this at low cost.

The characteristic valve curve with a pump system and full piping for a specific fluid was not performed because the restrictions of the use of the laboratories at the EAFIT university due to the Covid 19 confinement for that moment. That why only the opening percentage was calculated.

4. Conclusions

A functional prototype was developed for an automatic control proportional valve, with a coupling of a conventional gate valve found in the market with a stepper motor programmed with low cost microcontrollers, allowing to control the opening percentage of the valve in relation to the number of motor´s steps to regulate the flow and also make the remote connection for cell phones of the valve. The implementation of this prototype in production processes allows a reduction in equipment costs of around 70% per unit.
The prototype developed in this work allows to implement this type of automatic control technologies for small industry and academia, giving benefits to the end user and scaling processes to industry 3.0 and 4.0, reducing operating costs. Also, the use of these technologies in companies will be facilitated through everyday devices, such as cell phones. This prototype will allow to automate continue process in laboratory scale to the DPI hotbed in EAFIT University at Medellin, Colombia.
It is important to clarify that the tests carried out to prove the operation and the flow test in the valve had restrictions and measurement errors due to the restriction of laboratory use. due to the Covid 19 confinement for that moment.

5. Annexes

Annex 1. ARDUINO Programmed with ESPHOME

Annex 2. Configuration of the ESPHOME code to Home Assistant App

Annex 3. Automatization of the opening percentage of the valve in relation with the motor´s steps.

Available on https://github.com/parispimienta10/github-upload.git

References

[1] picuino, “Controlador PID.” [Online]. Available: https://www.picuino.com/es/arduprog/control-pid.html.
[2] E. Fisher, “Control Valve Handbook.Pdf,” vol. 5, pp. 14–25.
[3] A. Creus Solé, “Instrumentacón Industrial,” 8th ed., 2011, pp. 381–382.
[4] Y. Nishioka, K. Suzumori, T. Kanda, and S. Wakimoto, “Multiplex pneumatic control method for multi-drive system,” *Sensors Actuators, A Phys.*, vol. 164, no. 1–2, pp. 88–94, 2010.
[5] Electronica Estudio, “¿Qué es un microcontrolador?” [Online]. Available:

https://www.electronicaestudio.com/que-es-un-microcontrolador/.
[6] N. C. Guerrero Mesa, “Ingenieria e implementación de un prototipo parcial de una valvula de control de flujo,” *Univ. Pamplona*, vol. 33, 2006.
[7] E. E. Henao-Bravo *et al.*, “Low - Cost Electronic Opening Control Valve,” vol. 12, no. 22, pp. 87–96, 2017.
[8] Bitwise Ar, “Arduino desde cero en Español - Capítulo 33 - Paso a paso bipolar y controlador A4988.” [Online]. Available: https://www.youtube.com/watch?v=u0SG681s8aA.
[9] T. Electrónica, “A4988 con disipador de calor.” [Online]. Available: http://www.electronicathido.com/index.php?id_product=1008&controller=product.
[10] Arduino, “Bipolar Stepper Motor.” [Online]. Available: https://create.arduino.cc/projecthub/ingo-lohs/stepper-a-first-introduction-to-nema-17-ac15d7?ref=search&ref_id=stepper&offset=0.
[11] H. Libre, “ESP8266: el módulo WIFI para Arduino.” [Online]. Available: https://www.hwlibre.com/esp8266/.
[12] Tdrobotica.co, “Módulo Wifi - ESP8266.” [Online]. Available: http://tdrobotica.co/modulo-wifi-esp8266/350.html.
[13] H. Home, “Lista de precios 2020 Lista de preços 2020,” 2020.