International Journal of the Physical Sciences Vol. 6(23), pp. 5575-5585, 9 October, 2011
Available online at http://www.academicjournals.org/IJPS
ISSN 1992 - 1950 ©2011 Academic Journals

Full Length Research Paper

Fuzzy logic applied to control a one degree of freedom (DOF) pneumatic robot

Ramos-Arreguin Juan-Manuel*, Gorrostieta-Hurtado Efren, Pedraza-Ortega Jesus-Carlos, Aceves-Fernandez Marco-Antonio and Vargas-Soto Jose-Emilio

Facultad de Informatica, Universidad Autonoma de Queretaro, Av. De Las Ciencias S/N, Juriquilla, Queretaro, México.

Accepted 18 August, 2011

For many years, the vast majority of robots have been built with electrical actuators, and pneumatic actuators are not considered to control the position of a manipulator robot. Instead, electrical actuators are used. The main concern with pneumatic actuators is the highly non linear behavior, due to the high compressibility of the air. This work is focused on a one degree of freedom (DOF) robot, with pneumatic actuator to control the angle of the arm. A dynamic mathematical model of the manipulator is used, including the thermo-mechanical model. The air flow through the pneumatic cylinder is controlled by a 5/3 electro-valve. A pulse width modulation (PWM) signal is used to activate each coil of the electro-valve, where the duty cycle is set by a fuzzy logic algorithm. The fuzzy system is implemented into field-programmable gate array (FPGA), including the digital interface with a computer, an optical encoder and four seven-segment displays. This work is focused on the hardware implementation of fuzzy logic algorithm into FPGA system using the Sugeno algorithm, into a FPGA device, applied to the pneumatic electro-valve, with 25 ms of sample time.

Key words: Field-programmable gate array (FPGA) hardware, fuzzy logic, pneumatic robot, dynamic.

INTRODUCTION

Fuzzy logic is widely used to control many industrial systems, especially for systems with a no linear behavior, where a proportional integral derivative (PID) control is not enough to get satisfactory results. Moreover, the use of fuzzy logic algorithms implemented in hardware, has been used in different ways. Barriga et al. (1996) developed an automatic synthesis of fuzzy logic controllers, and its implementation has been developed by Lago et al. (1997). A fuzzy logic algorithm is used to control a car trajectory (Daijin, 2000), and its hardware implementation provides long computing time, but it offers high performance. Later, a hardware implementation of fuzzy system type 2 was developed (Melgarejo and Peña-Reyes, 2004), where the architectural proposal is used for specifying a type-2 fuzzy processor with reconfigurable rule base. Furthermore, fuzzy controller has been implemented for electric vehicle, by fuzzy systems with dynamic reconfiguration, implemented with an FPGA (Poorani et al., 2005). The need to implement fuzzy logic controllers is increasing with time for all kind of applications, especially when the mathematical models are not available or inaccurate. Therefore, a hardware implementation of the fuzzy algorithm is necessary with FPGA technology (Deliparaschos et al., 2006). Finally, fuzzy logic implementation can be used for traffic control (Karakuzu and Demirci, 2010).

In robotic field, the pneumatic actuators has been used previously to control a pneumatic manipulator (Ramos et al., 2006), including neural networks (Gorrostieta et al., 2007) and simulation with practical results was presented by Ramos-Arreguin et al. (2008a, b), where several kind of controllers are used for position control, including PID, fuzzy logic and neural networks. This paper is focused into fuzzy logic hardware implementation for position control of a pneumatic robot with one degree of freedom with small steps. A hardware implementation of fuzzy logic for pneumatic robot is reported by Ramos et al. (2010), where the fuzzy algorithm is designed for step response, with high variations. Figure 1 show the mechanical structure for the pneumatic robot, with one degree of freedom, where X is the rod displacement and θ is the arm position with vertical reference. The main

*Corresponding author: jramos@mecamex.net

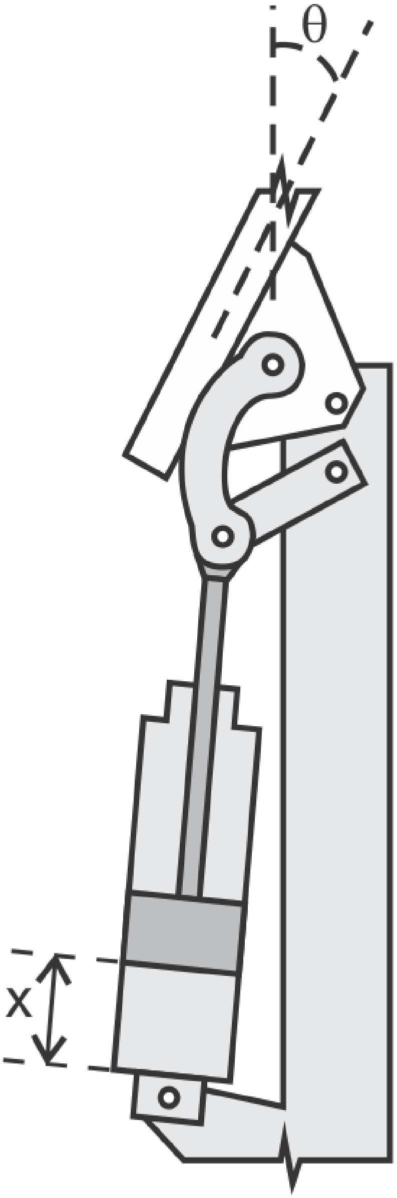

Figure 1. Pneumatic robot with one degree of freedom.

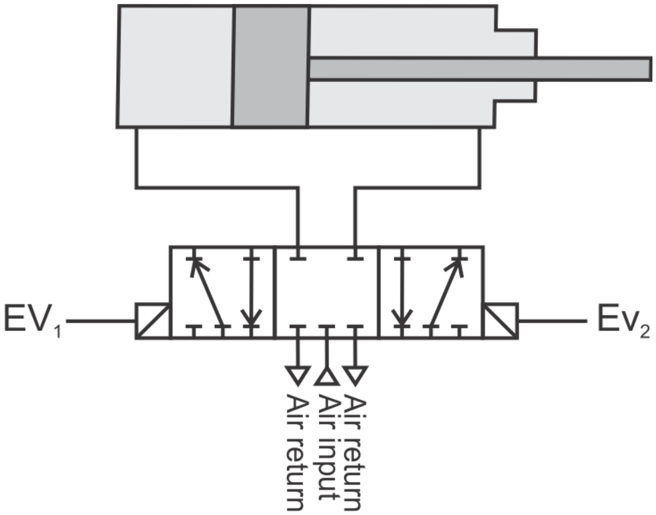

Figure 2. Pneumatic hardware used for the pneumatic robot with one degree of freedom.

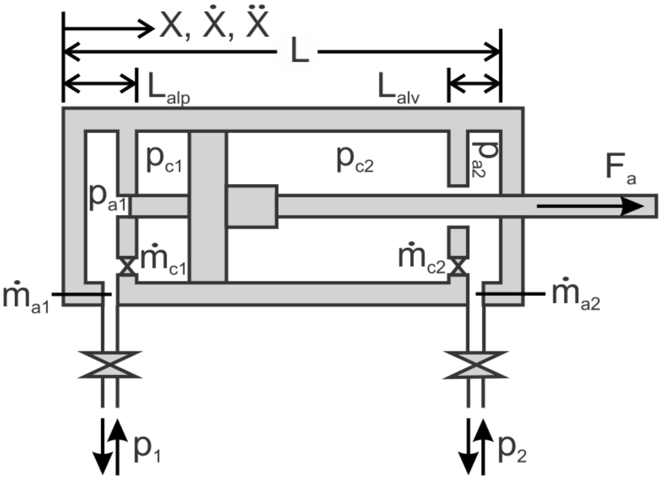

Figure 3. Pneumatic cylinder and its internal variables.

contribution of this work is the intelligent control process that generates small changes for the rod displacement into pneumatic cylinder. This process includes the use of fuzzy algorithm to set the duty cycle for the PWM signal, applied to the electro-valve. The fuzzy algorithm is implemented onto a FPGA device, to obtain high performance.

THE PNEUMATIC SYSTEM

The pneumatic system used for the pneumatic robot is shown in Figure 2. The digital signals EV_1 and EV_2 are used to control the air flow through the pneumatic cylinder. The hardware used to implement the pneumatic system is a pneumatic cylinder with chambers in both sides, and electro-valve 5/3 (5 ports 4 way and 3 position) with closed center, used for the rod position control.

The cylinder used in this work, includes two damping chambers in both sides. Figure 3 shows a schematic of such cylinder. The mathematical model is explained subsequently.

The thermo-mechanical model

The mathematical model used in this work is called “Simplified Thermo-Mechanical” model, and corresponds to the pneumatic cylinder shown in Figure 3, defined by Equations (1) to (10).

For the interval $0 \le X \le L$:

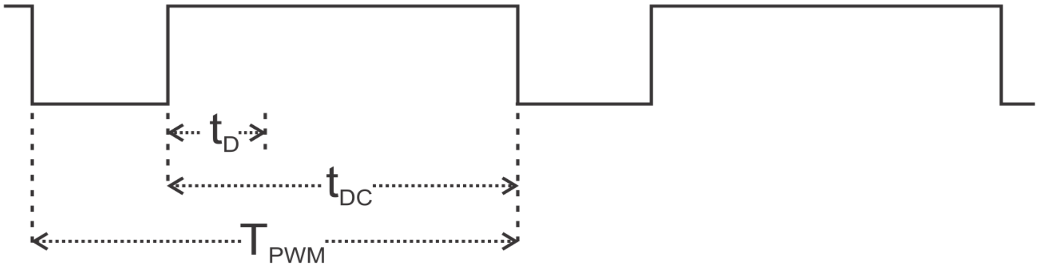

Figure 4. Timing diagram of PWM signal.

Table 1. Variables involved in PWM timing diagram.

Variable	Description	Unit
T_{PWM}	Period of the PWM	Seconds
t_{DC}	Duty cycle time for PWM signal	Seconds
t_D	Delay time for each coil of electro-valve	Seconds

$$\dot{X} = \frac{d}{dt}X \tag{1}$$

$$D\dot{X} = \frac{d^2}{dt^2}X \tag{2}$$

For the interval 0 ≤ X ≤ L_{alp}:

$$\dot{P}_{a1} = g_{21}(X)\left(\dot{m}_{a1} - \dot{m}_{c1} - 9.176\times10^{-10}P_{a1}DX\right)\times10^{8} \tag{3}$$

$$\dot{P}_{c1} = g_{31}(X)\left(\dot{m}_{c1} - 3.608\times10^{-8}P_{c1}DX\right)\times10^{6} \tag{4}$$

For the interval L_{alp}< X ≤ L:

$$\dot{P}_{a1} = g_{22}(X)\left(\dot{m}_{a1} - 3.7\times10^{-8}P_{a1}DX\right)\times10^{11} \tag{5}$$

$$\dot{P}_{c1} = g_{32}(X)\left(\dot{m}_{c1} - 3.7\times10^{-8}P_{c1}DX\right)\times10^{11} \tag{6}$$

For the interval 0 ≤ X ≤ (L-L_{alv}):

$$\dot{P}_{c2} = g_{41}(X)\left(\dot{m}_{c2} + 3.469\times10^{-8}P_{c2}DX\right)\times10^{11} \tag{7}$$

$$\dot{P}_{a2} = g_{51}(X)\left(\dot{m}_{a2} + 3.469\times10^{-8}P_{a2}DX\right)\times10^{11} \tag{8}$$

For the interval (L - L_{alv}) < X ≤ L:

$$\dot{P}_{c2} = g_{42}(X)\left(\dot{m}_{c2} + 3.352\times10^{-8}X_4X_6\right)\times10^{13} \tag{9}$$

$$\dot{P}_{a2} = g_{52}(X)\begin{bmatrix} 9.983\times10^{3}\left(\dot{m}_{a2} - \dot{m}_{c2}\right) + \\ 1.168\times10^{-5}X_5X_6 \end{bmatrix}\times10^{4} \tag{10}$$

Due to the non-linear behavior of the air, and the non-linear mathematical model expressed by the second order differential equations shown before, the system behavior is complex. For this reason, we propose the use of a fuzzy logic algorithm.

Air mass flow control

With the use of pulse width modulation (PWM) method, the air flow can be controlled, and the position of the rod can be changed inside the pneumatic cylinder. The electro-valve is a SY5320-6LZ-01, manufactured by SMC Company, with 19 s time delay. A PWM cycle with its timing data is shown in Figure 4; this PWM signal is applied on the electro-valve to control the rod speed.

To control the displacement direction of the rod, digital signals EV_1 and EV_2 are used. If the rod has to spin backwards to the cylinder, signal EV_2 must be set HIGH and signal EV_1 is set LOW. However, if the rod has to shift out of the cylinder, signals EV_1 and EV_2 are set HIGH and LOW, respectively. The rod speed depends directly of t_{DC} time, and this time is set by the fuzzy logic algorithm. The t_{DC} cannot be less than t_D. Table 1 shows the meaning of each variable involved in Figure 4. The fuzzy Logic is develop to adjust the t_A time, and control the air flow. The Fuzzy logic algorithm gives the output needed by the controller, considering the time of duty cycle is divided in 256 parts.

THE KINEMATIC SYSTEM

The mechanical structure is shown in Figure 5, and parameters values are in Table 2. From the mechanism shown on Figure 5, two kinematic chains can be obtained. The first chain includes R, L_2, A and B vectors, where R is given in Equation (11); and the second chain includes L_2, L_3, L_4 and L_5. Figure 6a and b shows the kinematic chains.

$$R = Lv + H + X \tag{11}$$

By considering the vectorial analysis of the kinematic system shown in Figure 5, the kinematic chains are obtained, as shown in Figure 6.

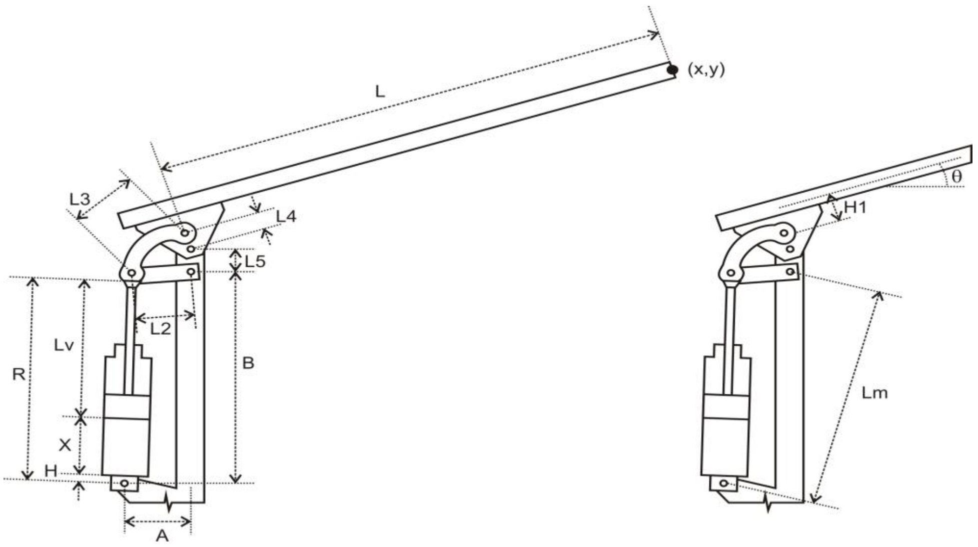

Figure 5. Mechanical structure of one degree of freedom pneumatic robot.

Table 2. Parameters of pneumatic robot.

Param	Description	Value (m)
A	Difference between cylinder and crank	0.108
B	Distance between pivot cylinder – crank	0.450
H	Distance form pivot to cylinder bottom	0.042
H1	Distance between the arm and the pivot of the follow bar	0.073
L2	Crank length	0.074
L3	Coupled bar length	0.099
L4	Follow bar length	0.043
L5	Distance between crank and follow pivot	0.041
Lm	Distance between cylinder and crank pivots.	0.463
L	Arm length	1.500
Lv	Rod length	
X	Rod displacement	
θ	Arm angle	

The vector system of the kinematic chain in Figure 6a is shown in (12).

$$\bar{A}+\bar{B}+\bar{L}_2=\bar{R} \qquad (12)$$

The components *x* and *y* are shown in (13) and (14).

$$\mathrm{A}+\mathrm{L}_2\cos\propto_2=\mathrm{R}\cos\propto_1 \qquad (13)$$

$$\mathrm{B}+\mathrm{L}_2\sin\propto_2=\mathrm{R}\sin\propto_1 \qquad (14)$$

For the kinematic system in Figure 6b, we have the Equations (15) to (17).

$$\bar{\mathrm{L}}_2+\bar{\mathrm{L}}_3+\bar{\mathrm{L}}_4=\bar{\mathrm{L}}_5 \qquad (15)$$

$$\mathrm{L}_2\cos\propto_2+\mathrm{L}_3\cos\propto_3+\mathrm{L}_4\cos\propto_4=0 \qquad (16)$$

$$\mathrm{L}_2\sin\propto_2+\mathrm{L}_3\sin\propto_3+\mathrm{L}_4\sin\propto_4=\mathrm{L}_5 \qquad (17)$$

Finally, the final position (*x*,*y*) is obtained by (18) and (19).

$$\mathrm{x}=\mathrm{L}\cos\theta+(\mathrm{L}_4+\mathrm{H}_1)\cos\alpha_4 \qquad (18)$$

$$\mathrm{y}=\mathrm{L}\sin\theta+(\mathrm{L}_4+\mathrm{H}_1)\sin\alpha_4+\mathrm{L}_5+\mathrm{B} \qquad (19)$$

The values of α_j are shown in (20) through (23), and are computed with Newton Raphson's method.

$$\alpha_{1i+1}=\alpha_{1i}-\frac{\mathrm{R}^2+\mathrm{A}^2+\mathrm{B}^2-\mathrm{L}_2^2-2\mathrm{RA}\cos\alpha_{1i}-2\mathrm{RB}\sin\alpha_{1i}}{2\mathrm{RA}\sin\alpha_{1i}-2\mathrm{L}_2\mathrm{RB}\cos\alpha_{1i}} \qquad (20)$$

$$\alpha_{2i+1}=\alpha_{2i}-\frac{-\mathrm{R}^2+\mathrm{A}^2+\mathrm{B}^2+\mathrm{L}_2^2+2\mathrm{L}_2\mathrm{A}\cos\alpha_{2i}+2\mathrm{L}_2\mathrm{B}\sin\alpha_{2i}}{-2\mathrm{L}_2\mathrm{A}\sin\alpha_{2i}+2\mathrm{L}_2\mathrm{B}\cos\alpha_{2i}} \qquad (21)$$

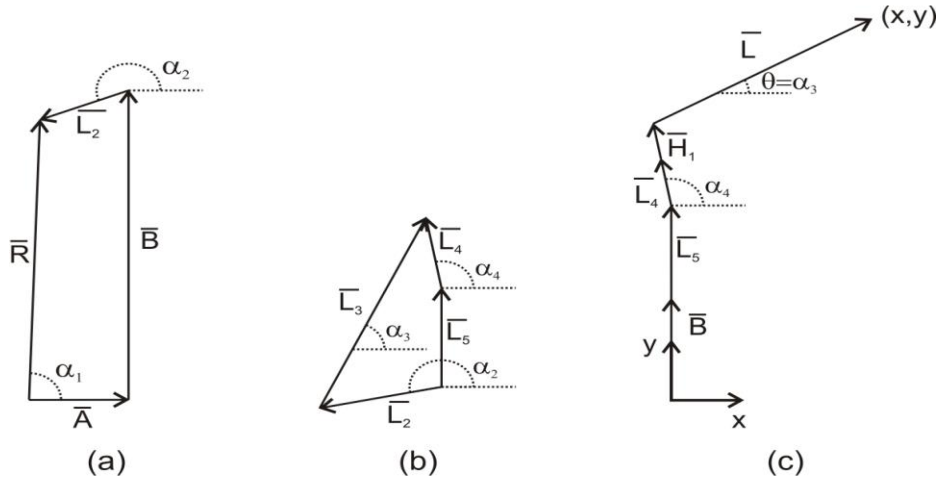

Figure 6. Kinematic chains for pneumatic robot; (a) for cylinder part; (b) the output part; (c) to compute the final position.

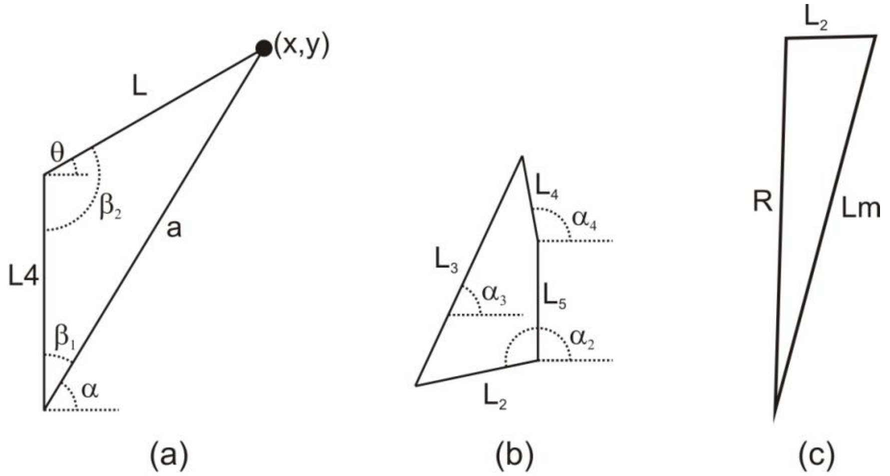

Figure 7. Structure for inverse kinematic; (a) output kinematic chain; (b) four bars mechanism, and (c) pneumatic cylinder.

$$\alpha_{4i+1} = \alpha_{4i} - \frac{\begin{bmatrix} L_5^2 + L_4^2 - L_3^2 + L_2^2 - 2L_4L_2(\cos\alpha_4\cos\alpha_2 + \sin\alpha_4\sin\alpha_2) \\ +2L_5L_4\sin\alpha_4 - 2L_5L_2\sin\alpha_2 \end{bmatrix}}{2L_4L_2(\sin\alpha_4\cos\alpha_2 - \cos\alpha_4\sin\alpha_2) + 2L_5L_4\cos\alpha_4} \quad (22)$$

$$\alpha_{3i+1} = \alpha_{3i} - \frac{\begin{bmatrix} L_5^2 - L_4^2 + L_3^2 + L_2^2 + 2L_3L_2(\cos\alpha_3\cos\alpha_2 + \sin\alpha_3\sin\alpha_2) \\ -2L_5L_3\sin\alpha_3 - 2L_5L_2\sin\alpha_2 \end{bmatrix}}{2L_3L_2(\cos\alpha_3\sin\alpha_2 - \sin\alpha_3\cos\alpha_2) - 2L_5L_3\cos\alpha_3} \quad (23)$$

These equations are used to compute the final position of the robot. Next, the inverse kinematic of the pneumatic robot system is presented.

The inverse kinematic system

For the inverse kinematic system, an equations set show the inverse kinematic in according to Figure 7. The variable values into Figure 7a are computed with the Equations (24) to (28), where θ is the output angle.

$$a^2 = x^2 + y^2 \quad (24)$$

$$\alpha = tg^{-1}\frac{y}{x} \quad (25)$$

$$\beta_1 = cos^{-1}\left(\frac{L^2 - a^2 - L_4^2}{2aL_4}\right) \quad (26)$$

$$\beta_2 = cos^{-1}\left(\frac{a^2 - L^2 - L_4^2}{2LL_4}\right) \quad (27)$$

$$\theta = \beta_2 - 90 \quad (28)$$

To compute the angles of the four bars mechanism, in Figure 7b, Equations (29) to (32) are used.

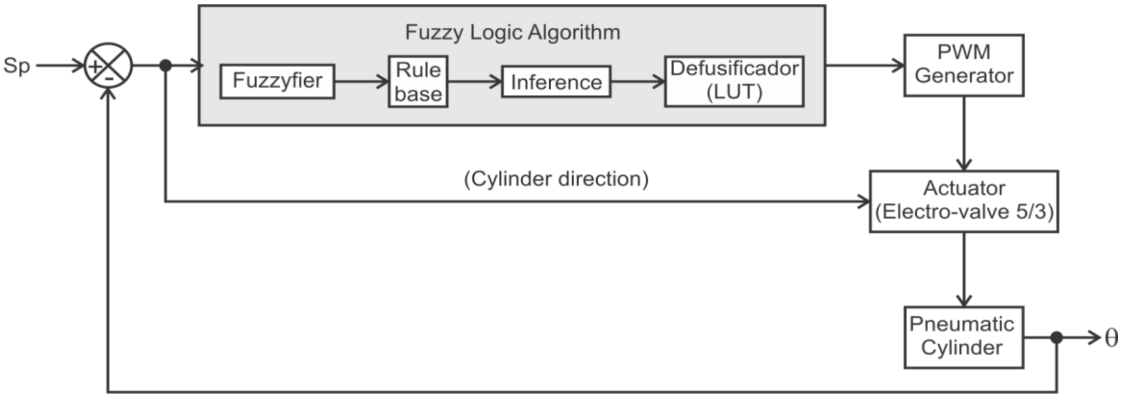

Figure 8. Block diagram to control the rod displacement *X*.

$$\alpha_4 = \beta_1 + \alpha \qquad (29)$$

$$\alpha_{1i+1} = \alpha_{1i} - \frac{R^2 + L_2^2 - L_m^2 - 2L_2R(cos\alpha_2 cos\alpha_{1i} + sin\alpha_2 sin\alpha_{1i})}{2L_2R(cos\alpha_2 sin\alpha_{1i} - sin\alpha_2 cos\alpha_{1i})} \qquad (30)$$

$$\alpha_{2i+1} = \alpha_{2i} - \frac{\begin{bmatrix} L_5^2 + L_4^2 - L_3^2 + L_2^2 - 2L_4L_2(cos\alpha_4 cos\alpha_{2i} + sin\alpha_4 sin\alpha_{2i}) \\ +2L_5L_4 sin\alpha_4 - 2L_5L_2 sin\alpha_{2i} \end{bmatrix}}{2L_4L_2(cos\alpha_4 sin\alpha_{2i} - sin\alpha_4 cos\alpha_{2i}) - 2L_5L_2 cos\alpha_{2i}} \qquad (31)$$

$$\alpha_{3i+1} = \alpha_{3i} - \frac{\begin{bmatrix} L_5^2 + L_4^2 + L_3^2 - L_2^2 - 2L_4L_3(cos\alpha_4 cos\alpha_{3i} + sin\alpha_4 sin\alpha_{3i}) \\ +2L_5L_4 sin\alpha_4 - 2L_5L_3 sin\alpha_{3i} \end{bmatrix}}{2L_4L_3(cos\alpha_4 sin\alpha_{3i} - sin\alpha_4 cos\alpha_{3i}) - 2L_5L_3 cos\alpha_{3i}} \qquad (32)$$

Finally, for Figure 7c, Equations (33) to and (11) are used to compute the rod displacement (x) to get the desired position of the arm.

$$R = \sqrt{L_2^2 + L_m^2 + 2L_2L_m(cos\alpha_2 cos\alpha_m + sin\alpha_2 sin\alpha_m)} \qquad (33)$$

$$x = R - Lv - H \qquad (34)$$

The equations described subsequently are used to know the angle of each mechanism part. Next, a system description was presented.

System description and problem statement

A mechanical system with n degree-of-freedom described by Euler-Lagrange equations is considered as shown in Equation (35).

$$\frac{d}{dt}\left(\frac{\partial L(q,\dot{q})}{\partial \dot{q}}\right) - \frac{\partial L(q,\dot{q})}{\partial q} = \tau \qquad (35)$$

where $q(t) \in \boldsymbol{R}^n$ are the generalized positions of the system, the Langrangian function $\partial L(q,\dot{q})$ and the τ are the generalized forces applied to the system.

Denoting by M(q), the n×n inertia matrix, by the Coriolis $C(q,\dot{q})\dot{q}$ and centrifugal forces, and considering the applied forces τ=*u-F* with *F* representing frictions, and u the control inputs to the system, as shown Equation (36).

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) + F = u \qquad (36)$$

The symmetry and bounderness of *M(q)* are set next

$$\bar{m} \geq m > 0, mI \leq M(q) = M^T(q) \leq \bar{m}I, \forall q \in \boldsymbol{R}^n \qquad (37)$$

Skew symmetry is

$$M(\dot{q}) - 2C(q,\dot{q}): s^T[M(q) - 2C(q,\dot{q})] = 0\ \forall s \in \boldsymbol{R}^n \qquad (38)$$

The friction *F* depends predominantly on speed, but generally also on position, time and external forces. It constitutes one of the major sources of the uncertainty in the mechanical system represented by Equations (37).

In Equation 36, let $q_d(t) \in \boldsymbol{R}^n$ be a given smooth $(q_d, \dot{q}_d, \ddot{q}_d bounded)$ desired trajectory and define the tracking error as $\tilde{q}(t) \triangleq q(t) - q_d(t)$. The problem considered in this paper is to design a control law for (36) to ensure the tracking error to be uniformly ultimately bounded. Also, the ultimate bound should be made arbitrarily small by choosing appropriately controller parameters, while maintaining all signals in the system uniformly bounded. For the fuzzy logic system, a Mandami type is considered in (39).

$$R_n: If x_i is B_i^n(x_i) and \ldots and x_i is B_i^n(x_i) then y_i is S_i^n(y_i) \qquad (39)$$

where R_n denotes the *n*th rule, $x = (x_i, \ldots, x_m) \in X \subset \mathbb{R}^n$and inputs variable, control variable and $B_i^n(x_i)$are defined as

$$B_i^n(x_i) = \begin{cases} \dfrac{x-a}{b-a} & a \leq x \leq b \\ 0 & if x < a and x > c \\ \dfrac{c-x}{c-b} & b \leq x \leq c \end{cases} \qquad (40)$$

In similar way, the triangular membership function $S_i^n(y_i)$ is defined.

THE FUZZY ALGORITHM

The duty cycle of PWM signal (t_{DC}) is set by the fuzzy algorithm, and t_{DC} must to be great than t_D. Figure 8 shows a block diagram of the system to control the arm position, controlling the angle θ. With

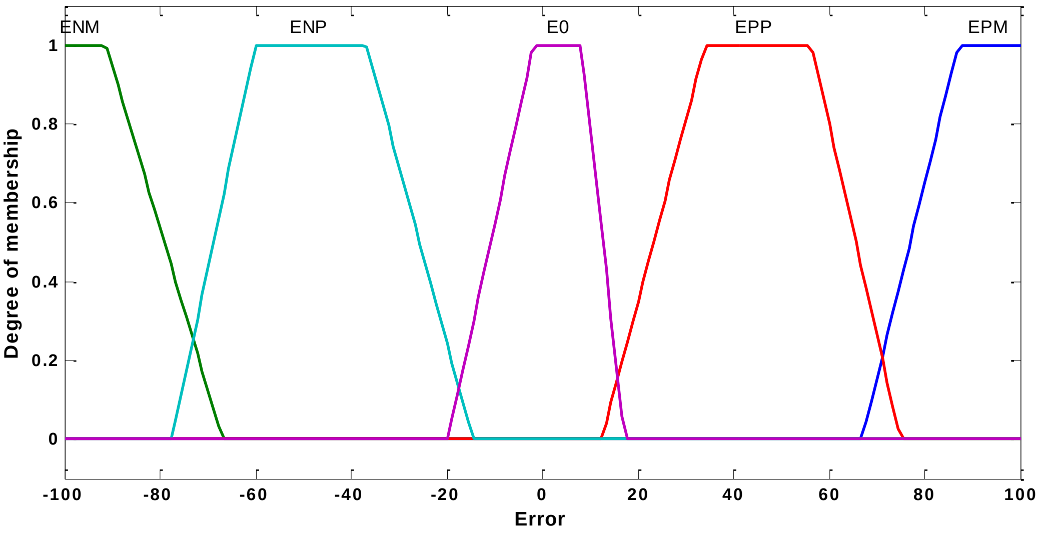

Figure 9. Membership functions for the input (Error).

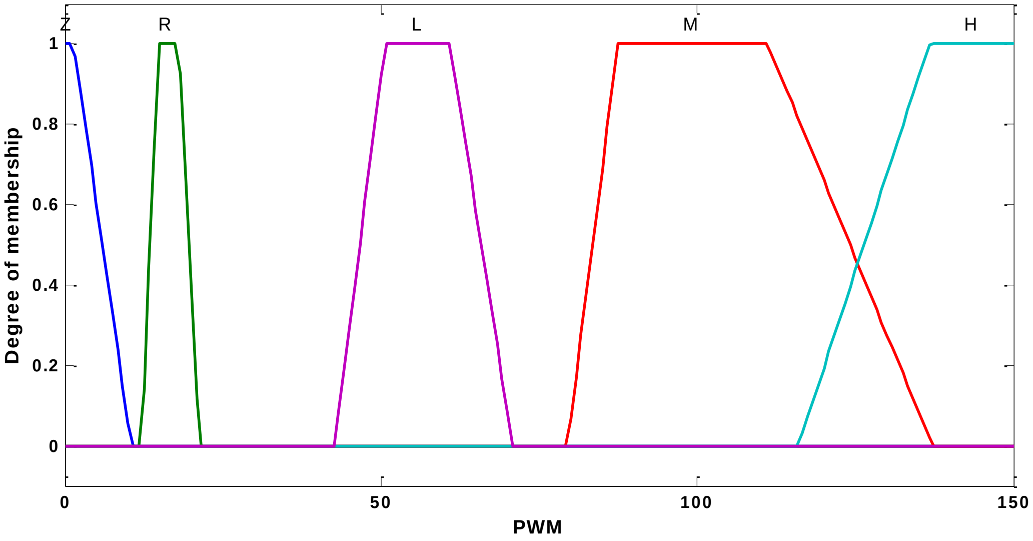

Figure 10. Membership functions for the output (DC).

the sign of the error signal, the direction of rod movement is defined, and the electro-valve obtains the PWM signal from the PWM generator, and the air flow is used into the pneumatic cylinder. Due to the robot topology, the most important variable is the output angle value. Therefore, the path planning for the robot and the angle speed control are not a concern.

As shown in Figure 8, the first step to consider is to compute the error. The error is the input of the fuzzy logic algorithm, where the output is the duty cycle value, needed for the PWM Generator block. This block delivers the PWM signal to the electro-valve, and joint with the error, the rod movement is generated in the right direction.

The fuzzy logic algorithm is designed using Matlab software. The algorithm considers just the error (E) as the input, and the output is the duty cycle for the PWM generator. Figures 9 and 10 show the membership functions for the input and the output variables. The direction of rod movement is defined by the error in Equation (41).

$$E = Sp - X \tag{41}$$

where: E = error; Sp = set point, and X = actual position
The rules used in this case are shown next, obtained using the member functions of Figures 9 and 10.

Rule 1: If error is EPM (Error Positive Medium) then PWM is H
Rule 2: If error is EPP (Error Positive Small) then PWM is M
Rule 3: If error is ENP (Error Negative Small) then PWM is R
Rule 4: If error is E0 (Error Cero) then PWM is Z
Rule 5: If Error is ENM (Error Negative Medium) then PWM is L

The actual position is sensed by an optical encoder. The interval of movement of the arm is 0 to 170 degrees, represented by 0 to 1820 counts of the encoder. Therefore, the resolution of movement measure is 0.0934° by each count of the encoder. The output of

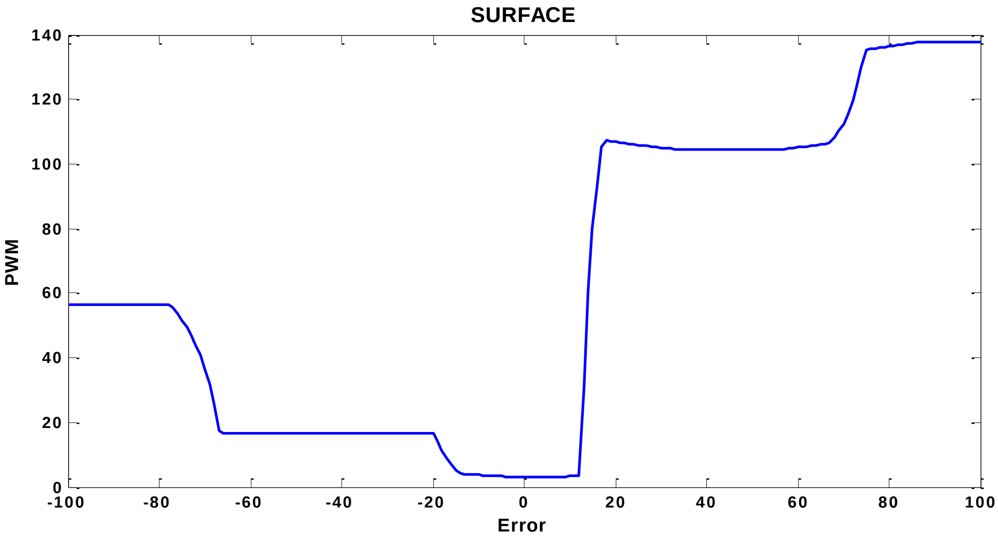

Figure 11. Surface of the fuzzy rules (Error input versus DC output).

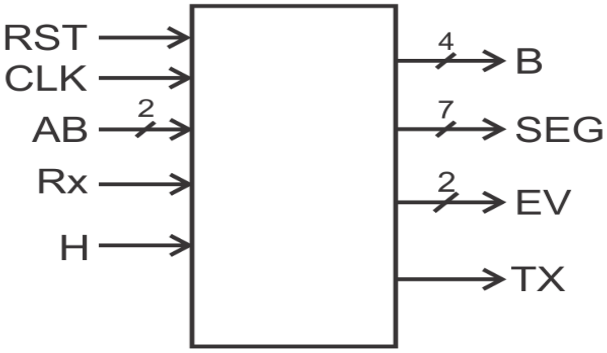

Figure 12. Entity of fuzzy_electrovalve.

defuzzifier represent the duty cycle value for PWM generator is 0 to 255. This value gives us a good resolution for the air flow control. The method used for rules evaluation is Mamdani, and the defuzzification method is centroid. Figure 11 shows the surface of behavior for the fuzzy logic algorithm.

The hardware implementation of the fuzzy logic algorithm is explained subsequently.

HARDWARE IMPLEMENTATION

The fuzzy algorithm hardware implementation has been developed in several ways, as proposed in Ramos-Arreguin et al., (2008a), where a hardware realization is presented, including fuzzification and inference engine. The used signals are:

1. RST => Reset
2. CLK => Clock (50 MHZ)
3. AB => Encoder signals
4. Rx => RS232 receive line
5. H => Enable
5. B => displays control
6. SEG => Signals for seven segments display
6. EV => Electro-valve control
7. Tx => RS232 transmit line.

Their proposal to improve the fuzzification is based on the arithmetic calculation method. Also, this realization reduces the hardware cost by means of reducing its complexity. However, when the hardware cost is reduced, the time solution is increased. The proposal of this work has the main advantage in the computing speed, due that the output of fuzzy algorithm is obtained in just one clock cycle. The very high speed integrated circuit hardware description language (VHDL) description is developed in accordance with the entity shown in Figure 12.

The architecture of fuzzy_electrovalve is shown in Figure 13. The hardware implementation is developed using a Xilinx board Spartan-3 Starter Kit, with an FPGA device XC3S200, with 200K gates. A hardware description language is needed to implement the fuzzy algorithm, and VHDL is the description language used.

The fuzzylut module shown in Figure 13 contains the results of the fuzzy rules and the output value. The process to obtain the LUT is explained in Ramos-Arreguin et al. (2010).

The EncoderCounter module takes the pulses from the optical encoder (AB signal), and signals B and SEG are used to show the arm position with 4 seven segments display. The counter output is

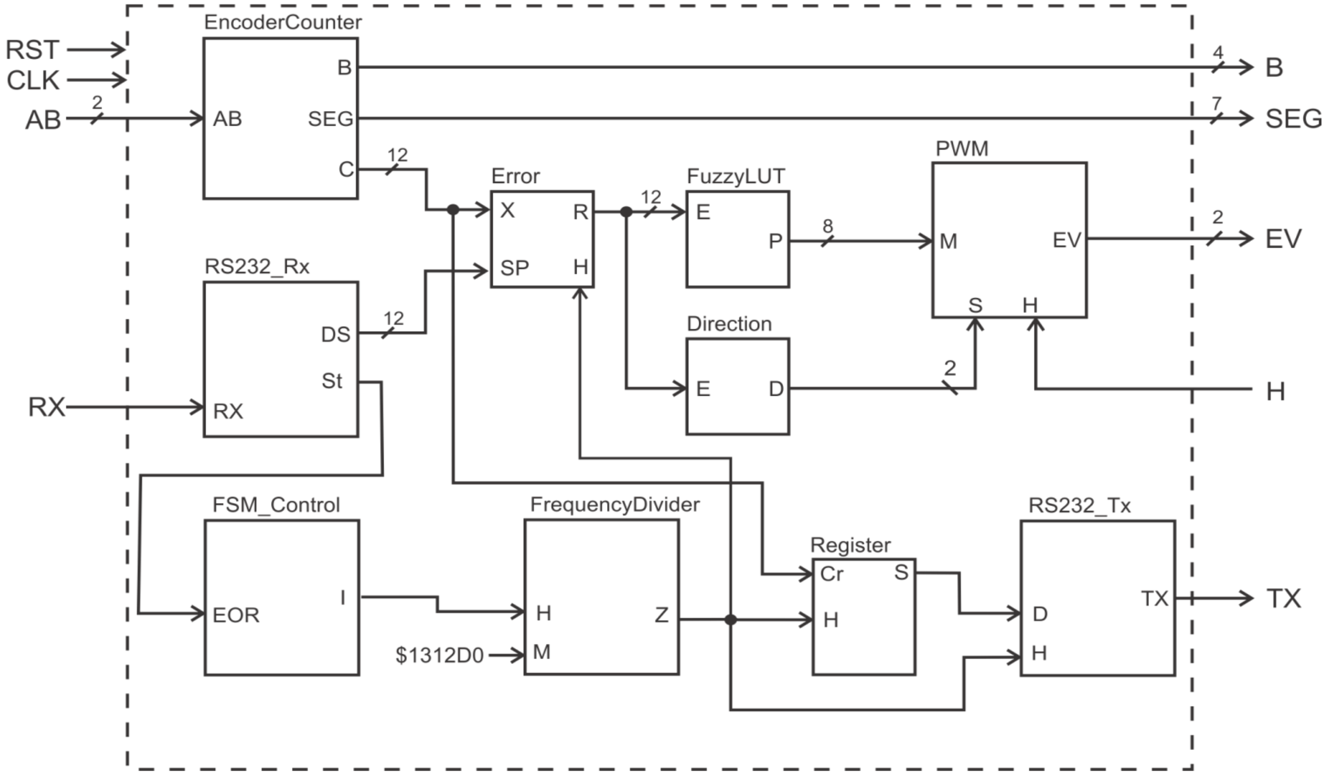

Figure 13. Architecture of fuzzy_electrovalve.

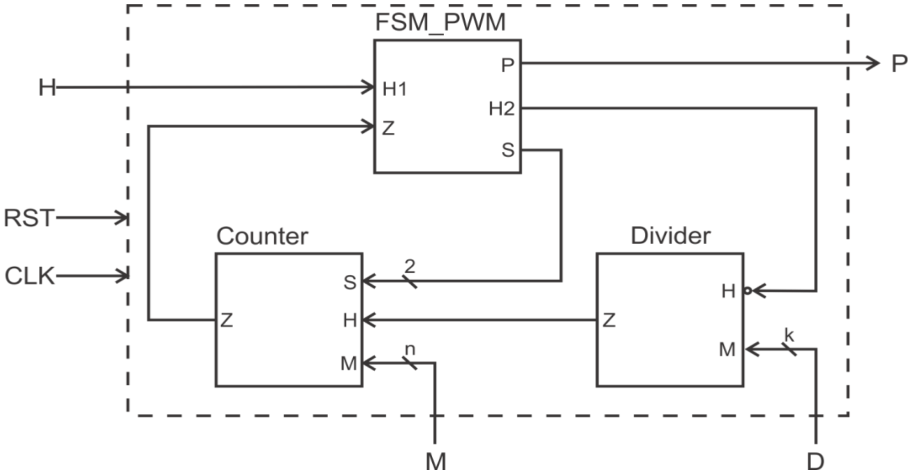

Figure 14. PWM hardware description.

the signal C with 12 bit wide. The signals C and SP are the input of the error module, where SP is received through the RS232_Rx module; the error is the signal R and H is the signal to enable the output register. The signal R is connected to the look-up table (LUT) module, to get the output for the PWM duty cycle; also, the more significant bit of signal R is used to determine if the rod goes outward or goes inward. If the error is positive, then PWM signal goes out through the EV_0 signal; when the error is negative, the PWM signal is assigned to EV_1. The PWM module gets the value of duty cycle from fuzzylut module (signal M), and the rod direction is set by signal S.

PWM module

A descriptive block diagram of 8 bit PWM module is shown in Figure 14. The timing values used for PWM signal in Figure 4 are shown in Table 3.

Table 3. Timing values assigned for Table 1.

Variable	Value	Unit
T_{PWM}	25	Miliseconds
t_{DC}	0 a 25	Miliseconds
t_D	19	Miliseconds
t_A	0 a 7	Miliseconds

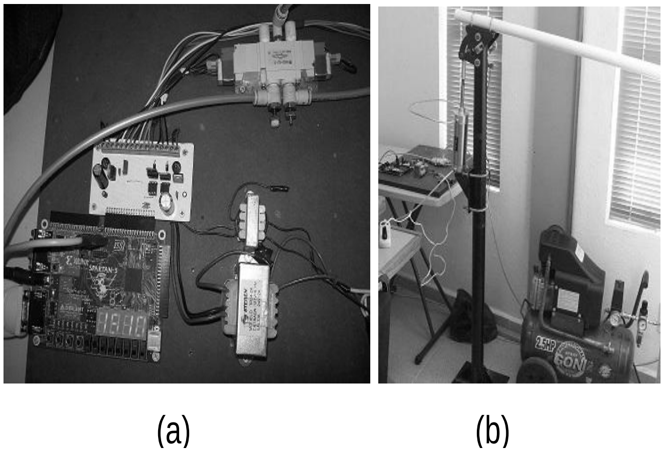

Figure 15. Pneumatic robot system (a) Spartan-3 kit; (b) pneumatic robot with 1 dof.

In Figure 14, the signal output P is the PWM signal. Signal D is set in accordance with Equation (40), where T_{CLK} is the clock period (20 ns), the PWM period is T_{PWM} = 25 ms and n is the number of bit in the counter module.

$$D = \frac{T_{PWM}}{2^n T_{CLK}} \tag{42}$$

Therefore, D value computed is D = 4882 (H1312).

RESULTS

The Spartan-3 board and power interface to control the electro-valve are shown in Figure 15; also pneumatic manipulator with one degree of freedom is shown in the same figure. The air pressure to work with the cylinder is 0.4 MPa, and the step response is shown in Figure 16.

The implementation of the system into FPGA allows accurate verification of the system behavior. Figure 16 shows the test, with different values of set point.

In Figure 16, the flat line is the set point, the error in all cases are around 1°, which is good for this test of implementation. The system is tested with step response, and the error is ±5% of the set point. The set point change is executed every 2 or 4 s.

The fuzzy rules are using just one input, due to the vibrations and the arm speed is not considered.

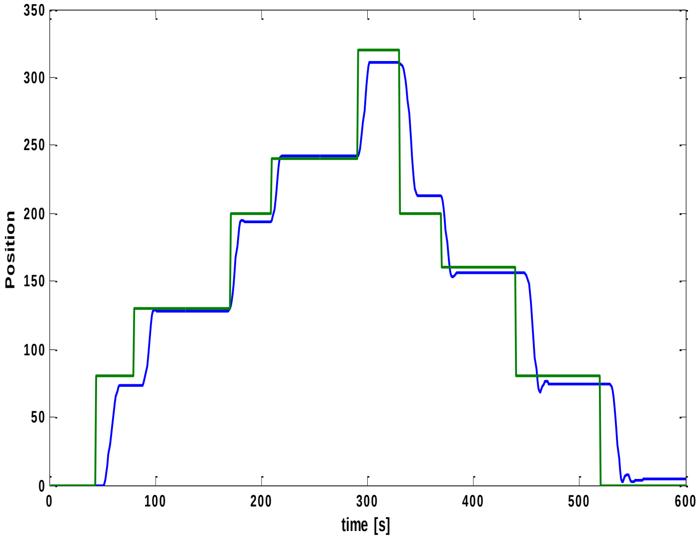

Figure 16. Test of the system response, with different set point.

DISCUSSION

In this work, a fuzzy controller design is presented, and it is applied to a pneumatic manipulator robot with one degree of freedom. Due the air behavior, the mechanical system has a nonlinear model. The mathematical model is a set of second order differential equations as could be seen on "The pneumatic system" section. In this particular case the development of fuzzy control allowed to work with the nonlinearities to design and propose a control and follow a specific path as shown in the results on "Hardware implementation". The development of another type of fuzzy controller could also work; for example, a neural control or neuro-fuzzy control. On the other hand, the FPGA implementation allows the development of a comprehensive control in an embedded system in which there is a field of development for the implementation of intelligent control systems. With FPGA devices and this kind of architecture, we can get a fuzzy algorithm evaluation in just one clock cycle, 20 ns, which allows to get a real-time control.

Conclusions

This work proposed the use of a Fuzzy logic embedded algorithm to control the arm position, showing that the hardware implementation is of paramount importance, because the system has not any dependency of a personal computer. In other works developed, the hardware implementation of fuzzy logic has been oriented with electrical motors, systems without a mathematical model, but works with hardware implementation applied to pneumatic cylinder are not reported. The major advantage of this effort is the decreasing of computing speed, due that the fuzzy algorithm is computed in just one clock cycle. This permitted to make other process in parallel, such as RS232 communication, data acquisition, etc.

For future work, a second degree should be added to the actual robot, with planning trajectory. Also, an embedded control will be included, considering a neural algorithm, a fuzzy PID control, a neural-PID control, and other kind of controllers algorithms.

Furthermore, a multivariable control ought to be implemented, with more degrees of freedom for the robot, considering more complex path planning and the pressure control into the chambers of the pneumatic cylinder.

REFERENCES

Barriga A, Sánchez-Solano S, Jiménez CJ, Gálan D, López DR (1996). Automatic Synthesis of Fuzzy Logic Controllers.Mathware Soft Computing. 3:425-434

Daijin K (2000). An Implementation of Fuzzy Logic Controller on the Reconfigurable FPGA System. IEEE Transaction Indust. Elect., 47(3):703-715

Deliparaschos KM, Nenedakis FI, Tzafestas SG (2006). Design and Implementation of a Fast Digital Fuzzy Logic Controller Using FPGA Technology. J. Intelligent and Robotic Syst., 45:77-96.

Gorrostieta E, Ramos JM, Pedraza JC (2007). Fuzzy and Neuronal Control to Flexible Manipulator. Int. J. Factory Automation, Robotics and Soft Comput., Pp 155-160.

Karakuzu C, Demirci O (2010). Fuzzy Logic Based Smart Traffic Light Simulator Design and Hardware Implementation. Appl. Soft Comput., 10-1:66-73

Lago E, Hinojosa MA, Jiménez CJ, Barriga A, Sánchez-Solano S (1997). FPGA Implementation of Fuzzy Controllers. XII Conference on Design of Circuits and Integrated Syst., (DCIS'97). Pp 715-720

Melgarejo MA, Peña-Reyes CA (2004). Hardware Architecture and FPGA Implementation of a Type-2 Fuzzy System. Proceedings of the 14th ACM Great Lakes symposium on VLSI, pp 458-461.

Poorani S, Urmila-Priya TVS, Kumar KU, Renganarayanan S (2005). Fpga Based Fuzzy Logic Controller for Electric Vehicle. J. Institut. Eng., Singapore, 45-5, pp 1-14.

Ramos-Arreguin JM, Guillen-Garcia E, Canchola-Magdaleno S, Pedraza-Ortega JC, Gorrostieta-Hurtado E, Aceves-Fernández MA, Ramos-Arreguin CA (2010). Fuzzy Logic Hardware Implementation for Pneumatic Control of One DOF Pneumatic Robot.Advances in Soft Computing, Springer, Part II, LNAI 6438, pp. 500–511.

Ramos-Arreguin JM, Pedraza-Ortega JC, Gorrostieta-Hurtado E, Romero-Troncoso RJ, Vargas-Soto JE, Hernandez-Hernandez F (2008a). Pneumatic Fuzzy Controller Simulation vs Practical Results for Flexible Manipulator.Automation and Robotics. I-Tech Edu. Publ., Pp. 191-200.

Ramos-Arreguin JM, Pedraza-Ortega JC, Gorrostieta-Hurtado E, Romero-Troncoso RJ (2008b). Artificial Intelligence Applied into Flexible Manipulator. Seventh Mexican Int. Conf. Artif. Intell.,.Pp. 339-345.

Ramos JM, Gorrostieta E, Pedraza JC, Romero RJ, Ramírez B (2006). Pneumatic Cylinder Control for a Flexible Manipulator Robot.12th IEEE International Conference on Methods and Models in Automation and Robotics.Pp. 637-641.