FPGA-based Implementation of Digital Logic Design using Altera DE2 Board

Zeyad Assi Obaid, Nasri Sulaiman and M. N. Hamidon
Department of Electrical & Electronic Engineering, Faculty of Engineering,
University Putra Malaysia,
43400 UPM Serdang, Selangor Darul Ehsan, Malaysia.
eng.alhamdany@yahoo.com

Abstract: In control applications, most of the physical systems require a real-time operation to interface high speed constraints; higher density programmable logic devices such as field programmable gate array (FPGA) can be used to integrate large amounts of logic in a single IC. This paper presents an Experimental implementation of digital logic designs on the Altera DE2 board which presented as an educational and development board, in order to check the flexible implementation with FPGA and to get the better and safely ways to use these specifications during any design implementations. The implementation in this paper contains of two types of digital logic design, the first one is Digital UP-counter design which designed using *Verilog* language, the experimental results for this design displayed on the 7-segment with the sequence of *HEX0* and *HEX1*. The second design is the proportional-derivative (PD) fuzzy logic controller which contain of three parts, Fuzzifier, inference engine and Defuzzifier. The PD Fuzzy logic controller was designed using *VHDL* language. Groups of two membership function with 5 linguistics variable and rule table of 25 rules were used to generate the control surface of the PD fuzzy logic controller, and to generate the simulation before the implementation, these groups are stored in the memory blocks, this block is generated using *MegaWizard Plug-in Manager* provided by Altera Quartus II program, the design using *MegaWizard* is important to ensure the good design specifications, the surfaces with these groups compared with the same design using *MATLAB*. By subtracting both type of surfaces, a results have been obtained, these results proved that the FPGA-based fuzzy controller is very close to the software-based controller using *MATLAB*.

Keywords: *PD Fuzzy Logic Controller, Digital up-counter, 7-segments Display, Altera Quartus II, Hardware implementation.*

1. Introduction

FPGA becomes one of the most successful of today's technologies for developing the systems which require a real time operation [1], [2], [3], [4]. Semi-custom and full-custom application specific integrated circuit (ASIC) devices are also used for this purpose but FPGA provide additional flexibility: they can be used with tighter time-to-market schedules. FPGA places fixed logic cells on the Wafer and the FPGA designer construct more complex functions from these cells [5].

The term field Programmable highlights the customizing of the IC by the user, rather than by the foundry manufacturing the FPGA. Several researchers discussed the design of hardware systems. Numbers of these works were specialized in control application, and were aim to get better control responses, [6], [7]. FPGA are two dimensional arrays of logic blocks and flip-flops with an electrically programmable interconnection between logic blocks. The interconnections consist of electrically programmable switches which is why FPGA differs from Custom ICs, as Custom IC is programmed using integrated circuit fabrication technology to form metal interconnections between logic blocks. In an FPGA logic blocks are implemented using multiple level low fan in gates, which gives it a more compact design compared to an implementation with two-level AND-OR logic. FPGA provides its user a way to configure and these specifications offer the fuzzy logic a large amount of logic in single IC: The intersection between the logic blocks and the function of each logic block. Logic block of an FPGA can be configured in such a way that it can provide functionality as simple as that of transistor or as complex as that of a microprocessor. It can used to implement different combinations of combinational and sequential logic functions [8].

2. DE2 development board

Altera DE2 board become one of the most widely development FPGA board which is used to development of FPGA design and implementations [9]. The purpose of the Altera DE2 Development and Education board is to provide the ideal vehicle for learning about digital logic, computer organization, and FPGAs. It uses the state-of-the-art technology in both hardware and CAD tools to expose students and professionals to a wide range of topics [10]. The board offers a rich set of features that make it suitable for use in a laboratory environment for university and college courses, for a variety of design projects, as well as for the development of sophisticated digital systems. Altera provides a suite of supporting materials for the DE2 board, including tutorials, "ready-to-teach" laboratory exercises, and illustrative demonstrations [11]. Figure (1) gives the block diagram of the DE2 board. To provide maximum flexibility for the user, all connections are made through the Cyclone II FPGA device. Thus, the user can configure the FPGA to implement any system design [9].

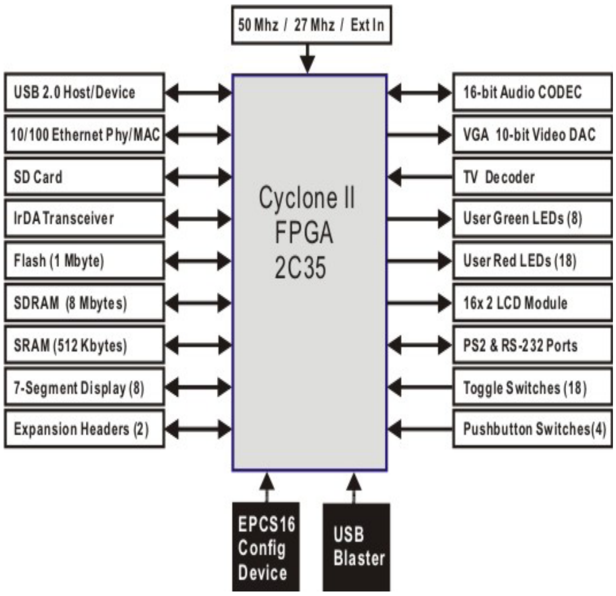


Fig. 1: Block diagram of the DE2 board [9].

Software provided with the DE2 board features the Quartus® II Web Edition CAD system, and the Nios® II Embedded Processor. Also included are several aids to help students and professionals experiment with features of the board, such as tutorials and example applications. Traditionally, manufacturers of educational FPGA boards have provided a variety of hardware features and software CAD tools needed to implement designs on these boards, but very little material has been offered that could be used directly for teaching purposes. Altera's DE2 board is a significant departure from this trend [10] and [11].

3. Structure of the Fuzzy Logic Controller

The fuzzy controller, (as explained in Figure (2)), have four main components:

(1) The Rule-Base holds the knowledge, in the form of a set of rules, of how best to control the system.
(2) The Inference Mechanism evaluates which control rules are relevant at the current time and then decides what the input to the plant should be.
(3) The Fuzzification Interface simply modifies the inputs so that they can be interpreted and compared to the rules in the rule-base.
(4) The Defuzzification Interface converts the conclusions reached by the inference mechanism into the inputs to the plant [13] and [14].

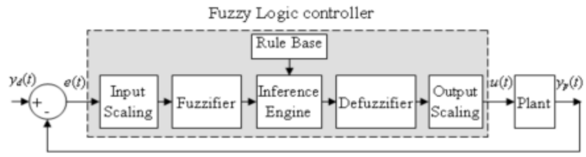


Fig. 2: Typical Fuzzy Controller Structure.

3.1 Rule-Base

Rule-base plays a key role in representing expert control knowledge and experience and in linking the input variables of fuzzy controllers to output variable (or variables). It contains a fuzzy logic quantification of the expert's linguistic description of how to achieve good control, represented in if-then statements involving fuzzy sets, fuzzy logic, and fuzzy inference [12] and [13]. A single fuzzy if-then rule assumes the form.

IF <antecedent > THEN < consequent > … (1)

3.2 Fuzzification

Fuzzification is a mathematical procedure for converting an element in the universe of discourse into the membership value of the fuzzy set. This procedure converts controller inputs into information that the inference mechanism can be easily used to activate and apply rules [12]. Suppose that fuzzy set $\tilde{A}$ is defined on [a,b]; that is the universe of discourse is [a,b]; for any $x \in [a,b]$, the result of Fuzzification is simply $\mu\tilde{A}(x)$. Figure (3) below shows an example in which the Fuzzification result for x =7 is 0.4.

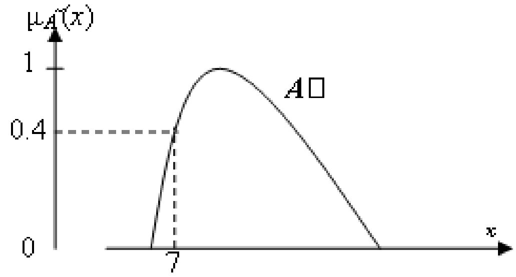


Fig. 3: an example showing how Fuzzification works.

3.3 Fuzzy Inference Engine

Fuzzy inference is sometimes called fuzzy reasoning or approximate reasoning. It is used in a fuzzy rule to determine the rule outcome from the given rule input information. In control, it emulates the expert's decision-making in interpreting and applying knowledge about how best to control the plant. A number of fuzzy inference methods are available [15] and [16] but only few of these methods are used frequently. Table (1) lists four of the most popular fuzzy inference methods. These methods are graphically illustrated in Figure (4).

Table 1: Common Fuzzy Inference Methods for Fuzzy Control System.

Fuzzy Inference Method	Definition
Mamdani minimum inference, R_M	$\min(\mu, \mu_A(x))$, for all x
Larsen product inference, R_L	$\mu \times \mu_A(x)$, for all x
Drastic product inference, R_{DP}	μ , for $\mu_A(x)=1$ $\mu_A(x)$, for $\mu=1$ 0 , for $\mu<1$ and $\mu_A(x)<1$
Bounded product inference, R_{BP}	$\max(\mu+\mu_A(x)-1, 0)$

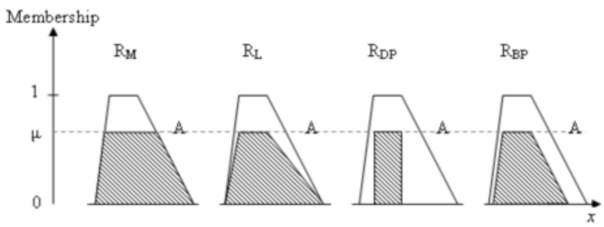


Fig. 4: Graphical illustration of the common fuzzy inference methods provided in Table (1).

When singleton output fuzzy set is used, the four different inference methods produce the same inference result, as shown in Figure (5). If the output fuzzy set in some rules are the same, fuzzy logic OR operation is sometimes used to combine the memberships. This step, however, is not essential; many fuzzy controllers can function properly without it [12].

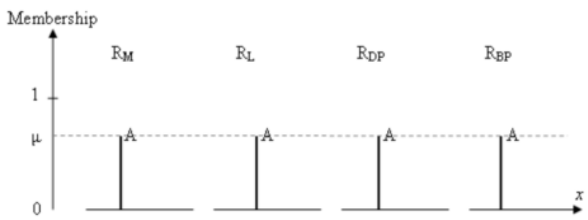


Fig. 5: The outcome, of Mamdani fuzzy controllers with singleton fuzzy sets in the rule consequent, using the inference methods provided in Table (1).

3.4 Defuzzification

Defuzzification is a mathematical procedure used to convert a fuzzy set or fuzzy sets to a real number. This procedure converts the conclusions of inference mechanism into an actual input for the process to be controlled. Different types of Defuzzifier are suitable for different circumstances. It is even hard to introduce these methods, as different authors sometimes name the same method differently. Centroid method (also known as the center of mass, or center of gravity method), is probably the most commonly used Defuzzification method because it has several desirable properties:

1. The Defuzzification values tend to move smoothly around the output fuzzy region, that is, changes in fuzzy set topology from one model frame to the next usually result in smooth changes in the expected value.
2. It is relatively easy to calculate.
3. It can be applied to both fuzzy and singleton output set geometries.

The Defuzzifier output, z is defined by

$$z = \frac{\sum_{k=1}^{N} \mu_k * \beta_k}{\sum_{k=1}^{N} \mu_k} \quad \ldots (2)$$

Where N represents the number of the rules, μk is the degree of the applicability of the kth rule; βk is the Defuzzifier value of the output membership function of the kth rule. This relation is also true for systems using singleton output fuzzy sets that are nonzero only at z = β1… βN [13], [16] and [17]. Figure (6) shows a typical fuzzy reasoning method of two rules for a fuzzy system with two inputs and one output, using singleton membership function for the output variable, minimum operation for fuzzy inference, and Centroid for Defuzzification. Note that x0 and y0 are input instances, and z0 is the final output. The rules are:

Rule 1: IF x is $\tilde{A}1$ AND y is $\tilde{B}1$ THEN z is z1
Rule 2: IF x is $\tilde{A}2$ AND y is $\tilde{B}2$ THEN z is z2

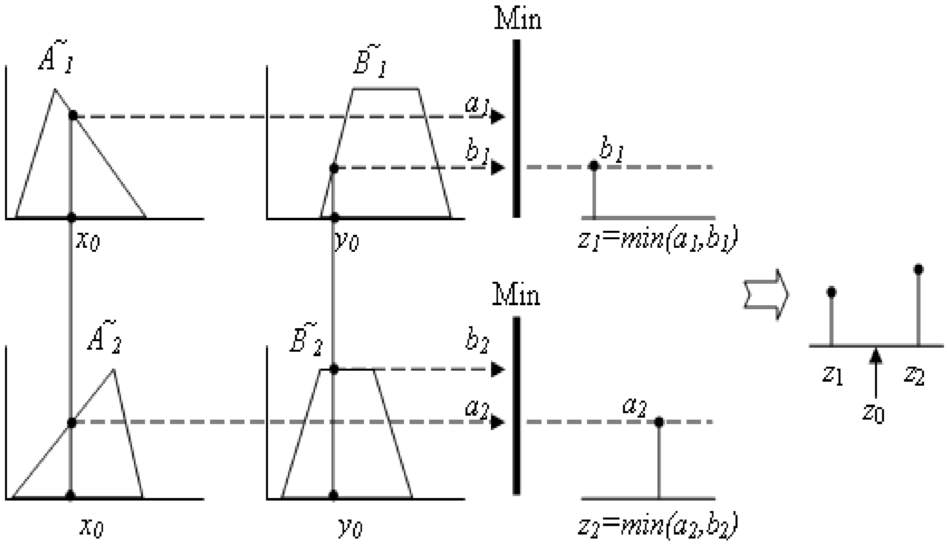


Fig. 6: Simplified fuzzy reasoning method.

4. Experimental Setup and Results

The purpose of this work is to implement a digital design using Hardware description language using DE2 Altera Board and to check the work flexibility in this development board which used in wide range of experiments development of the FPGA-based systems, as well as design of hardware controllers. Experimental environment shown in figure (7) in this paper used with all FPGA-based implementation steps. Altera DE2 board is connected the PC using USB-blaster provided with DE2 package. The VGA display connected with DE2 using VGA port available as one of DE2's output ports, other ports such as serial PS/2 used in either mouse or Keyboard use.

4.1 Implementation of Digital UP-counter with 7-segments display

Digital circuits implementations is important in many applications, we used this design to get more check about DE2 board, and it is necessary to get the safely and easy ways before implements the second design in section (4.2). this design contain of an input pins (clock and KEY0) and the output is connected with HEX0 and HEX1 for 7-segments display, Figure (8) shows the synthesis circuit of digital up-counter, and figure (9) shows the experimental results for this design which displayed on 7-segments with the reset step.

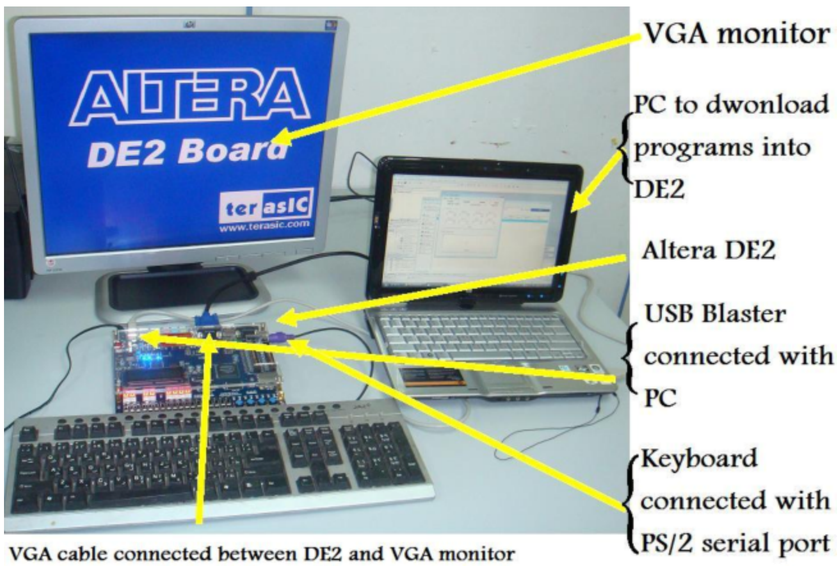


Fig. 7: Experimental environments.

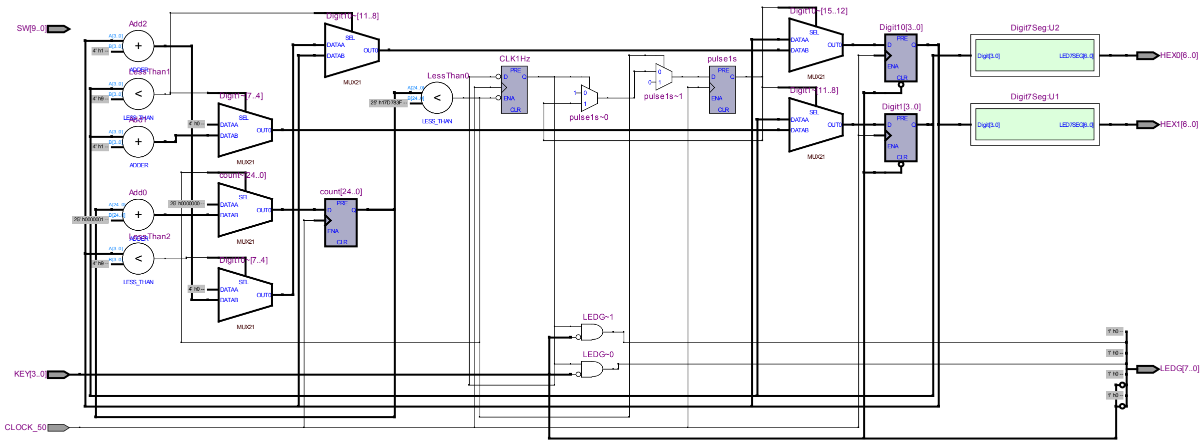


Fig. 8: Synthesized Design of the Digital up-counter.

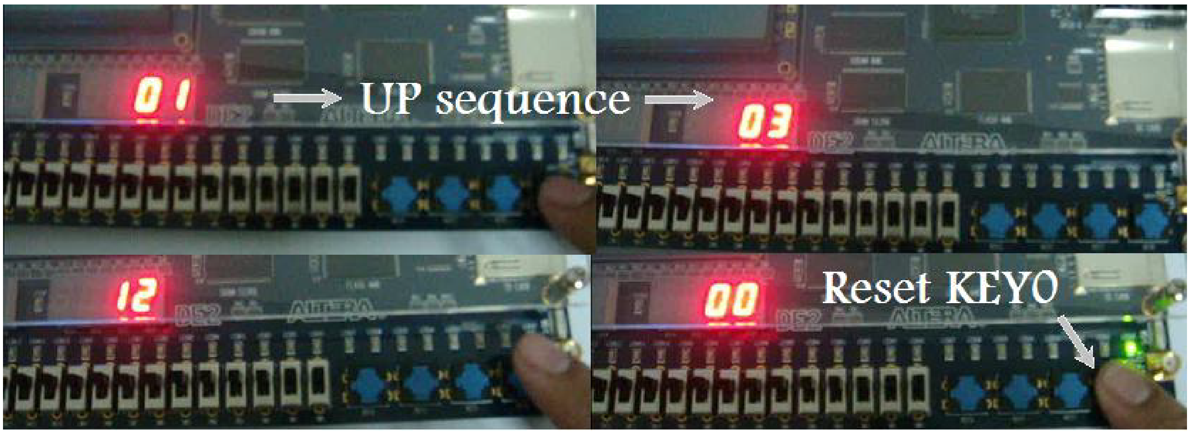


Fig. 9: Experimental result of digital up-counter implementation.

4.2 Implementation of PD Fuzzy Logic Controller

This section presents an implementation of the PD fuzzy logic controller as explain in section (3), the fuzzy logic controller has widely used with many systems especially with the control applications, fuzzy logic controller presented with three parts, Fuzzifier, inference engine and Defuzzifier, the typical is highlighted the number of inputs with the first part of the fuzzy logic controller, since it contain of one-input and one-output, in this paper, a PD fuzzy logic controller with two-input one-output was implemented using Altera DE2 board. For the purpose of simulation during the design we used five triangular membership functions for input variable, five singleton membership functions for output variable, and rule table of 25 rules, this group is shown below in figure (10) and table (2).

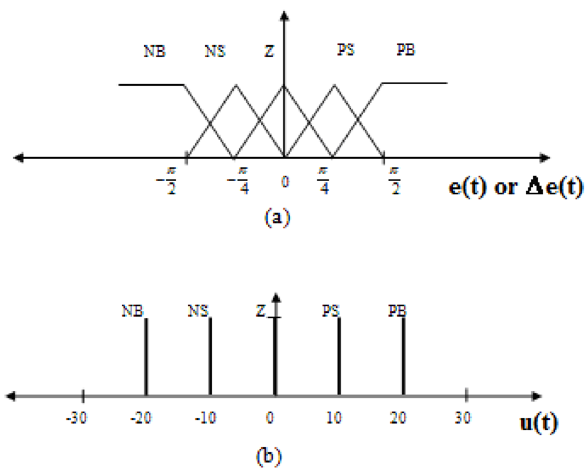


Fig. 10: (a) Inputs fuzzy sets, (b) Output fuzzy sets

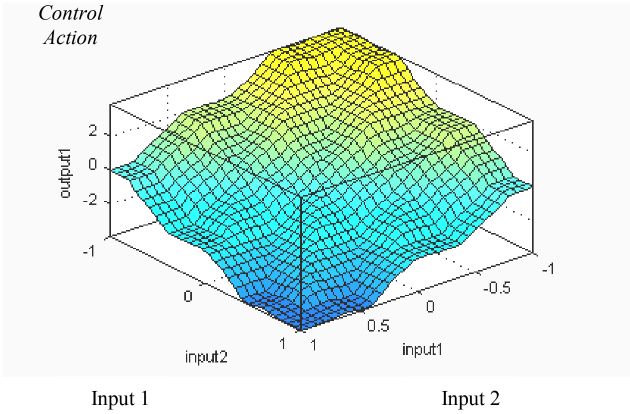
Table 2: Fuzzy rules used in the PD fuzzy logic controller.

		e				
		NB	NS	Z	PS	PS
de	NB	PB	PB	PB	PS	Z
	NS	NS	PB	PS	Z	NS
	Z	PB	PS	Z	NS	NB
	PS	PS	Z	NS	NB	NB
	PB	Z	NS	NB	NB	NB

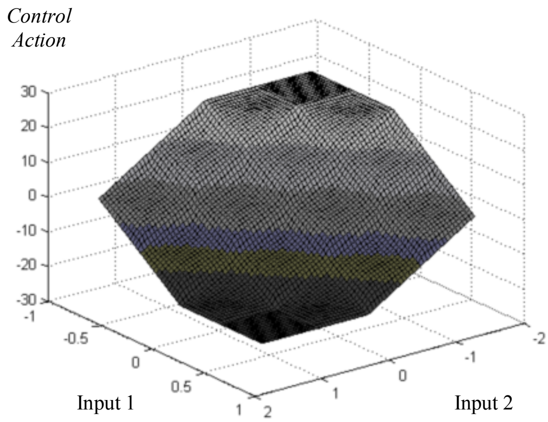
At first, a test is performed to make sure that the fuzzy inference system used inside the *FPGA-based* design is working properly. This test involves generating control surface using fuzzy sets and rule table shown in Figure (10) and table (2); figure (11-a) shows the surface of the fuzzy inference which implemented using VHDL, figure (11-b) shows Control surface of the PD fuzzy logic controller using MATLAB. This surface reveals the effect of rounding and approximation processes (inside the FPGA design) on the result, and by subtracting this surface with the one generated using MATLAB, the results shown in table (3) were obtained from this comparison. Figure (12) shows the synthesized design results of the PD fuzzy logic controller.

Table 3: Statistics of the difference surfaces.

	Difference percentage results
maximum error	0.5349
minimum error	-0.5349
mean error	3.89E-17
mean square error	0.0196
range of values	-30 to 30



(a)



(b)

Fig.11: a) Control surface of the PD fuzzy logic controller using VHDL.
b) Control surface of the PD fuzzy logic controller using *MATLAB*.

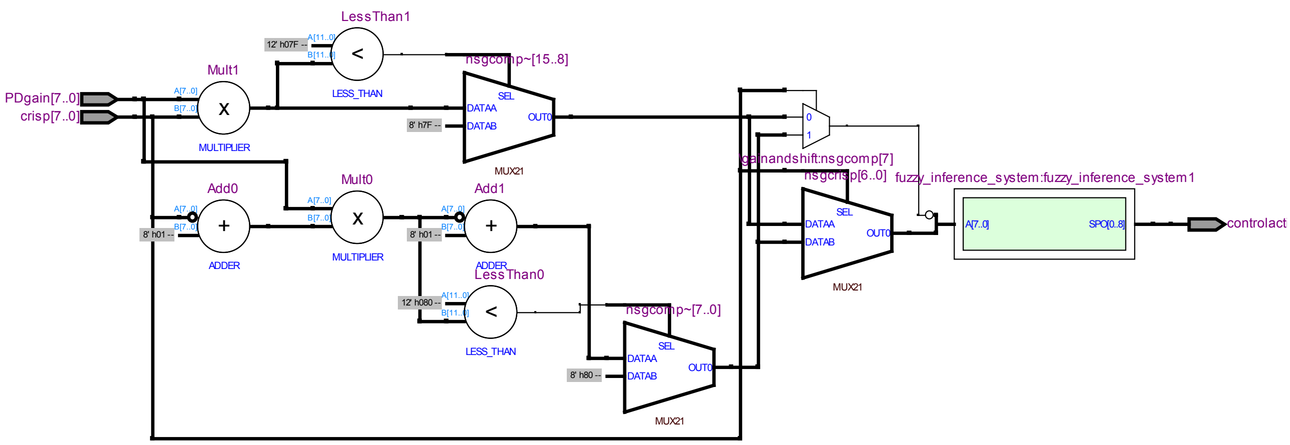


Fig. 12: Synthesized design results of the PD fuzzy logic controller.

works, from this comparison, it seem that these results is much closed to those obtained from MATLAB. The design shown in figure (12) consist of two input pins and four output pins, the input pins contain of proportional gain pin and the derivative gain pin, for the output pins, all these pins connected with the 7-segments display. Other way we can use the LCD to display this action. The FPGA design and implementations (using the environment showing in figure (7)) consist of many steps, starting with Design coding and end with pin assignments. After compiling the design code and get the Results, The user must assign the pins in the FPGA chip by using pin planner provided by Altera *Quartus II* program.

two inputs, these inputs represent the proportional-derivative (PD) controller gains (e(t) and Δe(t)). This controller can be used in unity feedback control as shown in figure (13). This design can be used to control many of the industrial application. And it approved the flexibilities with the FPGA-based implementations; Figure (14) shows the simulation waveform results of this design by using ModelSim Simulator, this test represents the controller with unity feedback control system. This simulation step is necessary before any design implementation in order to check the timing behavior of our design. Figure (15) shows the control action displayed on LCD, which built in DE2 board.

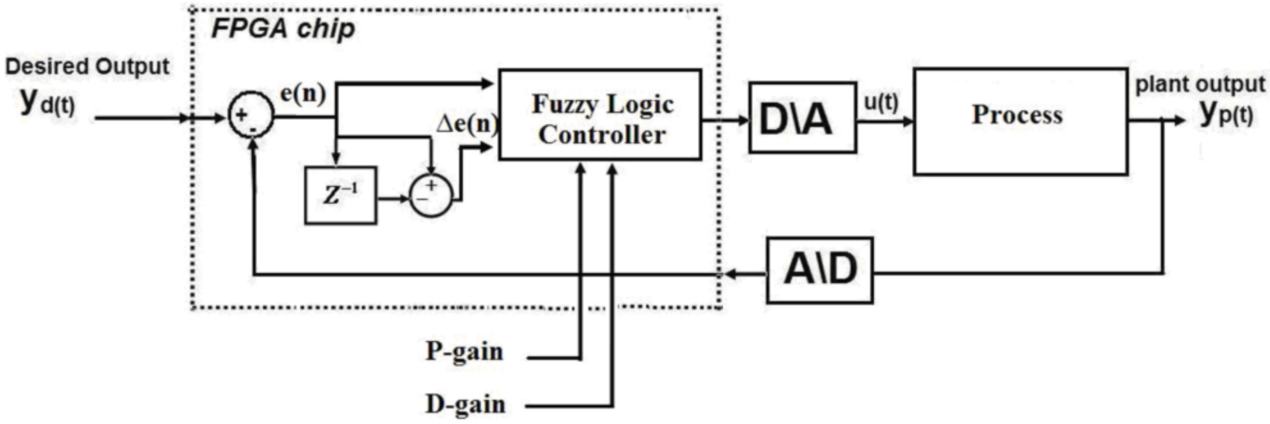


Fig. 13: Layout representation of the FPGA-based PD fuzzy logic controller with unity feedback control system.

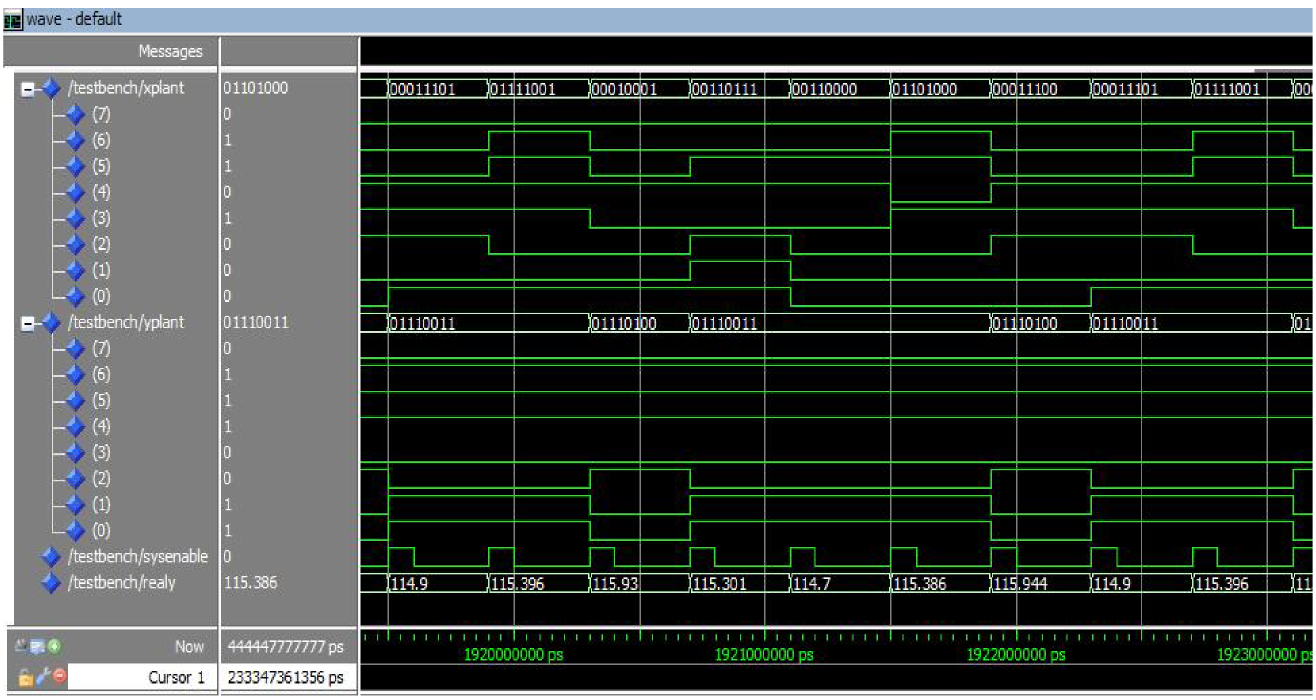


Fig. 14: Simulation waveform results of the PD fuzzy logic controller.

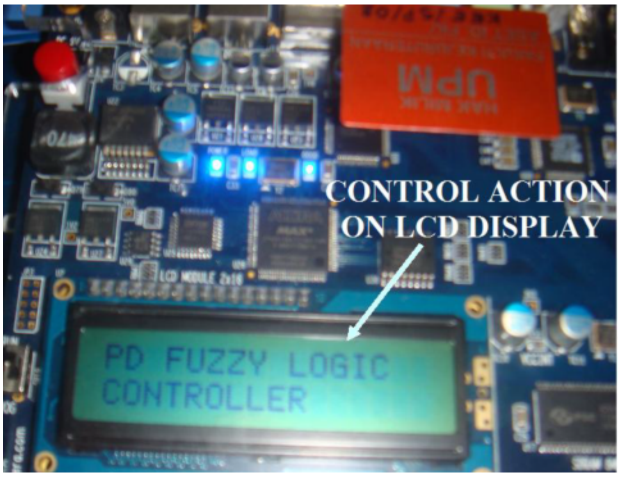


Fig. 15: LCD display in DE2 board.

5. Discussion

This paper presents experimental implementations of digital logic design using Altera DE2 developments and education boards, the purpose of the Altera DE2 development and education board is to provide the ideal vehicle for learning about digital logic, computer organization, and FPGAs. It uses both hardware and CAD tools to offers students and professionals to a wide range of topics for FPGA design implementations. The board offers a rich set of features that make it suitable for use in a laboratory environment for university and college courses and can used for any design implementations, as well as for the development of sophisticated digital systems by using hardware description language (HDL) . All connections are made through the Cyclone II 2C35 FPGA device in order to provide maximum flexibility for the user. Thus, the user can configure the FPGA to implement any system design. The first design used to check and prove the DE2 features before implementing the second design. The results obtained by subtracting the surface for our PD fuzzy logic controller with the surface generated using MATLAB, we got that the maximum error, minimum error, mean error, mean square error and range of values are: 0.5349, -0.5349, 3.89E-17, 0.0196 and -30 to 30, respectively. These results proved that the FPGA-based fuzzy controller is very close to the software-based controller using *MATLAB*. It is worth to notice that the precision of any control surface also depends on choosing the right memory words that represent the desired membership functions correctly. The implementations results for the first design proved the flexible implementation with DE2 board. Fuzzy logic controller is presented as one of the controllers to deal with many of the industrial applications, and can be implemented in many of the hardware implementation ways, FPGA offer this design a large amount of fuzzy logic in single IC, so that it's considered as one of the best ways to implement fuzzy logic controller especially with the applications require real-time operation. This type of implementation can be used with different applications, such as temperature control, industrial furnaces and process control applications.

Acknowledgment

The authors would like to thank firstly, our god, and all UPM staff and all friends who gave us any help related to this work. Finally, the most thank is to our families and to our countries which born us.

References

[1] Gerard Aranguren, Mariano Barron, Joseba, Arroyabe And Gonzalo Garcia-Carreira" a pipe-line fuzzy controller in FPGA", IEEE conference in fuzzy systems (FUZZ-IEEE), Volume: 2, p- 635- 640, Barcelona, Spain, 1997.

[2] Valentin Mure_San, Dan and Xiaojun Wang" from VHDL to FPGA. A case study of a fuzzy logic controller" proceedings of the international conference of young lecturers and PHD students, pp. 83 - 90, Miskolc, Hungary, 11-17 august 1997.

[3] Oscar Montiel, Yazmin Maldonado, Roberto Sepulveda, and Oscar Castillo," Simple Tuned Fuzzy Controller Embedded into an FPGA", Annual Meeting of the North American, Fuzzy Information Processing Society, NAFIPS, ISBN: 978-1-4244-2351-4, p. 1-6, New York City, NY, 19-22 May 2008.

[4] Shabiul Islam, Nowshad Amin, M.S.Bhuyan, Mukter Zaman, Bakri Madon and Masuri Othman, "FPGA Realization of Fuzzy Temperature Controller for Industrial Application", WSEAS Transactions On Systems And Control, ISSN: 1991-8763, Issue 10, Volume 2, p. 484- 490, Sep. 17, 2007.

[5] Philip t. Vuong, asad m. Madni and jim b. Vuong" VHDL implementation for a fuzzy logic controller", BEI technologies, inc. 13100 telfair avenue, sylmar, california 91342 USA, 2006. http://www.wacong.org/wac2006/allpapers/isiac/isiac_248.pdf

[6] Mohammed Y. Hassan and Waleed F. Sharif, " Design of FPGA based PID-like Fuzzy Controller for Industrial Applications", IAENG International Journal of Computer Science, 34:2, IJCS_34_2_05, 17 November 2007.

[7] Masmoudi n., hachicha m. and kamoun l." Hardware Design of Programmable Fuzzy Controller on FPGA", IEEE International Fuzzy Systems Conference Proceedings, Seoul, Korea, p. -1675- -1679, August 22-25, 1999.

[8] FPGA tutorial "Over view on FPGA", www.Tutorial-reports.com, 2008. http://www.tutorial-reports.com/computer-science/fpga/overview.php

[9] Altera "DE2 Development and Education Board, user manual" Version 1.4, 2006, [available online at] http://www.altera.com.

[10] Handouts "Introduction to FPGAs" June 2009, [available online at] http://saaubi.people.wm.edu/TeachingWebPages/Physics351_Fall2008/Week2/Week2_overheads.pdf
[11] Altera "DE2 Development and Education Board" DE2 Package, or [available online at] http://www.altera.com.
[12] H. Ying, "Fuzzy Control and Modeling, Analytical Foundations and Applications", Institute of Electrical and Electronic Engineers Inc., USA, 2000.
[13] K. M. Passino and Stephen Yurkovich, "Fuzzy Control", Addison-Weslwey Longman Inc., USA, 1998.
[14] J. Huang, "Hybrid Fuzzy PID Controller with Adaptive Genetic Algorithms for the Position Control and Improvement of Magnetic Suspension System", M. Sc. thesis, Mechanical and Electro-Mechanical Engineering, June, 2004. http://etd.lib.nsysu.edu.tw/ETD-db/ETD-search/view_etd?URN=etd-0624104-182807
[15] L. Reznik, "Fuzzy Controllers", Newnes, first edition, 1997.
[16] E. Cox, "The Fuzzy Systems Handbook", AP Professional, second edition, USA, 1999.
[17] S. S. Farinwata, D. Filev, and R. Langari, "Fuzzy Control: Synthesis and Analysis", John Wiley & Sons, Ltd., 2000.

Eng. Zeyad Assi Obaid was born in Dyala, Iraq. He received his B. Sc. Degree in control and systems engineering from the University of Technology, Baghdad, Iraq, in 2006. Cisco Certified Network Association (CCNA) from (Cisco Computer center), Doha, Qatar, 2007. From July 2008 currently is Master Student in control and automation engineering department, university putra Malaysia. His current research interests include fuzzy modeling and fuzzy control systems with FPGA design applications. He has many of Professional Affiliations like, Member in IAENG, Member in IEEE, Member in ACM, Member in IACSIT and Member in the Program committee of the IEEE International Conference on Signal and Image Processing Applications (ICSIPA), by the IEEE Signal Processing Society. He has an industrial experiences such as, from February, 2004 to October, 2006, working as: *Network administrator* in Alkawarizmy Computer Services Company, Baghdad, Iraq and From 20/12/2006 to 31/5/2008 working as *control and computer engineer in* Ezdan Real estate Company (international constructions and trading company), Doha, Qatar.

Dr. Nasri Sulaiman was born in Malaysia; he received his B. Sc. Degree in (Elect & Comp.) from university putra Malaysia, 1994. Master degree from University of Southampton, UK, 1999. PhD from University of Edinburgh, UK, 2007. He is currently a Lecturer at Department of Electrical and Electronics Engineering, UPM. His current research interests include. Evolvable Hardware (EHW) and Digital Signal Processing (DSP).

Dr. M. N. Hamidon was born in Malaysia; he received his B. Sc. Degree from university of Malay, Malaysia, 1994. Master degree from the university Kebangsaan, Malaysia, 2001. PhD from the University of Southampton, UK, 2005. He is currently a Lecturer at Department of Electrical and Electronics Engineering, UPM. His current research interests include. Microelectronics (Sensor Technology), MEMS, Wireless System Devices Fabrication and Packaging.