Proc. of the 2nd International Conference on Electrical, Communication and Computer Engineering (ICECCE)
12-13 June 2020, Istanbul, Turkey

FPGA Based Hierarchical Fuzzy System

Mohammed A. Al-Ibadi
Computer Engineering Department
University of Basrah
Basrah, Iraq
m.a.al-ebadi@ieee.org

Aum alhuda Gani Abood
Computer Engineering Department
University of Basrah
Basrah, Iraq
aumalhuda_comp@ yahoo.com

Majid A. Alwan
Computer Engineering Department
. University of Basrah
Basrah, Iraq
altimimee@yahoo.com

***Abstract*— Multivariable fuzzy system implementation suffers from amount of computation resulting from the high number of rules especially for large number of input variables. As number of inputs increases, the situation becomes worse because the number of rules increases exponentially with respect to input variables resulting in problem commonly called "rule explosion problem". Another problem related to multivariable system is the complexity of rule construction of high dimensionality fuzzy system. Hierarchical system breaks high dimensionality fuzzy system into several low dimensional fuzzy systems such that the number of rules has linear proportion with the number of input variables. In this paper, hardware implementation of fuzzy system with hierarchical structure is performed to obtain powerful system take the advantages of both hardware speed and low dimensionality processing reduction of hierarchical implementation.**

Keywords—FPGA, Hierarchical systems, Fuzzy, Architecture, Hardware Accelerator.

I. INTRODUCTION

In the recent years, fuzzy logic theory is widely used to provide an appropriate tool that efficiently deals with imprecision and uncertainty typical for the reasoning system [1]. The great success of fuzzy logic led to the development of many dedicated tools for designing fuzzy systems. Most of these tools related to conviction of knowledge such as tuning parameters of a system by using learning algorithms, fuzzy system created by data set, and inference approaches [2].

The conventional systems require an accurate mathematical model to simulate the given system or plant. Fuzzy logic system is an exact option to establish a robust system with presence of disturbance, noise, and imprecise input information when it is difficult to achieve the mathematical model of the system. It incorporates a simple if-then rule-based approach to solve the problem instead of attempting to find a complete mathematical model of the system. Fuzzy logic system depends on the experiences of operators rather than system technical understanding [3].

Many commercial software packages provided for fuzzy system could generate optimized code of different specific device operations. The huge growth in application of the fuzzy logic needed to find an efficient way of hardware implementation [4]. The implementing of fuzzy logic system could be done by software, hardware, or mixed of both. The performance requirement and the type of application decide the suitable approach for implementing the fuzzy system. The solution can be obtained by dividing the design in to two parts: software and hardware depending on some criteria such as reliability, response time, price etc.

A processor needs for software part that it runs with. However, the hardware can be implemented in functional units treated as standard blocks used in design period in the form of building blocks [5].

Recently, a large development of fuzzy logic applications that required operating in physical systems have real time constraints for information exchange to interface with high speed operation circumstances [6]. FPGA technology solves world problems by using it for implementing hardware fuzzy logic systems [5].

As a theoretical basis, fuzzy logic models are able to operate as general logic system models. The generic approximation ownership of fuzzy model became well visible. However, all schemes of fuzzy approximation suffer from a trouble of increasing the number of fuzzy rules exponentially with the number of input variables. The rule explosion trouble takes too much area of computer memory and needs more computations [7].

The hierarchical design of fuzzy system deals with this trouble in order to make fuzzy techniques applicable for complex system. The hierarchical fuzzy system HFS puts the input system variables in a set of low dimensional fuzzy logic systems instead of using single high dimensional system [8]. In this paper, hierarchical design is implemented for fuzzy system to make fuzzy techniques applicable for complex system with less complexity, high speed, and less hardware components.

978-1-7281-7116-6/20/$31.00 ©2020 IEEE

II. HIERARCHICAL FUZZY SYSTEM HFS

One limitation of conventional fuzzy systems is the limited number of input variables fuzzy system can deals with efficiently. Fuzzy system can manipulate few input variables giving efficient reasoning with reasonable time and amount of calculations [8].

With multiple input fuzzy system variables, the number of fuzzy rule is increased in exponential growth relation with respect to the number of input variables. This problem is commonly called "rule explosion problem" [9] or "curse of dimensionality"[8]. This problem can be subdivided in to three detail sub-problems: i-rule dimensionality (rule number increases rapidly, exponentially, with respect to the number of input system variables), ii- parameter dimensionality (mathematical formulas parameters number of fuzzy system also exponentially increased with respect to system variables number), and iii- information dimensionality (the amount of knowledge or data needed for identifying fuzzy system operation increased in the same manner, exponentially, as with the two other dimensionality problems) [10].

To overcome dimensionality problems and make multi-input fuzzy system manageable, hierarchical fuzzy system format is developed [11, 12]. In hierarchical structure of fuzzy systems, the rules number is linearly increasing in a linear relation to the number of input system variables. If the number of variables is n and the number of membership is m, then it is needed m^n fuzzy rules for complete fuzzy system construction. For example if m=5 and n=4, the rules number is 54 = 625. If n is increased to 5, then the number of rules becomes 55=3125.

Hierarchical fuzzy system is splitting the whole high dimensionality system into number of low dimensionality conventional fuzzy systems. When hierarchical fuzzy system was firstly proposed, it represented as cascaded multilevel fuzzy systems. The first level produced output approximation from applying fuzzy system theory on portion of system variables (the most influent variables for example). The second level uses one or more variables and the output of the previous fuzzy level as input and so on until giving system output from the top level of hierarchy.

If a given system contains n system variable is to be modeled in L-level hierarchical structure such that with ni variable for ith level then:

$$n = n_1 + \sum_{i=2}^{L}(n_i - 1) \quad (1)$$

If a constant number of variables q is used in each level then the number of levels L is determined as:

$$L = \frac{n-q}{q-1} + 1 \quad (2)$$

And the complete set, total number, of rules k will be:

$$k = L \times m^q = \left(\frac{n-q}{q-1} + 1\right) m^q \quad (3)$$

It is proven that the number of rules for each level, and hence the total rules number of the system, is minimized if only two variables are used in each level [11]. Depending on the previous Equations, for fuzzy system with number of input variables n=4 and fuzzy set m=5 the number of level L=3 and the total number of rules=$3*5^2$=75 (rather than 625 in covenantal fuzzy system). For m=5 and n=5 the total rules in hierarchical =$4*5^2$=100 (rather than 3125 in covenantal fuzzy system).

The most known hierarchical structures are the serial and parallel hierarchical fuzzy structure. In serial structure, each layer receives one input and the output of the previous layer to produce layer output. The output of the last layer is the system output as shown in Fig 1.

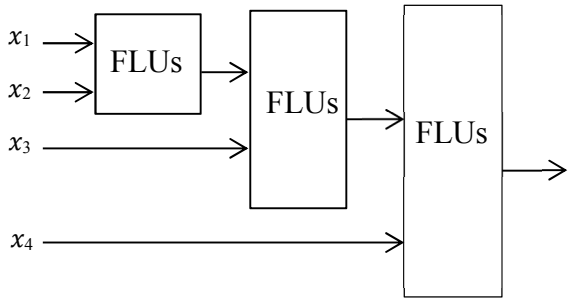

Fig. 1. Serial Hierarchical Structure

In parallel hierarchical fuzzy structure, all input variable $\{x_1, x_2, \ldots\ldots, x_n\}$ are used as inputs to the first layer. First layer contains $(n/2)$ fuzzy subsystems, each with two inputs. $(n/2)$ Intermediate outputs $\{y_1, y_2, \ldots\ldots, y_{n/2}\}$ are generated from the first layer. Intermediate variables from this layer represent the inputs to the second layer. The third layer input space become $(n/4)$ Intermediate variables. And so on until obtaining the last layer with single output [5, 32]. Fig. 2 illustrates the parallel hierarchical structure (for 4 input variables).

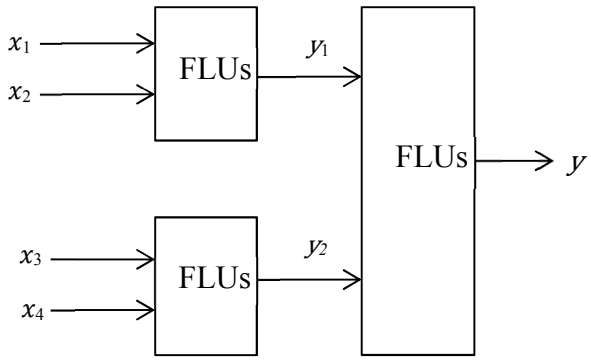

Fig. 2. Parallel Hierarchical Structure

Both serial and parallel hierarchical fuzzy logic systems suffer from little physical meaning intermediate output that makes the design to be difficult. This problem (intermediate layer output) can be managed by using rule mapping scheme. That deals with such intermediate outputs as the mapping intermediate variables. "Limpid- Hierarchical

Fuzzy System" (L-HFS) is proposed to obtain rule base that performs mapping process sufficient to deal with intermediate output difficulty. L-HFS is a method for mapping fuzzy rule –base of multi-input variables system (hierarchical fuzzy system) by mapping the intermediate variables to get a further result and ease to design middle layers fuzzy rules involved in the in hierarchical structure. By using this scheme, each fuzzy rule-base of fuzzy logic units doesn't need for redesigned. It also gives the same behavior (input-output) when using the standard fuzzy system with single layer. Fig. 3 illustrate 4- input L-HFS.

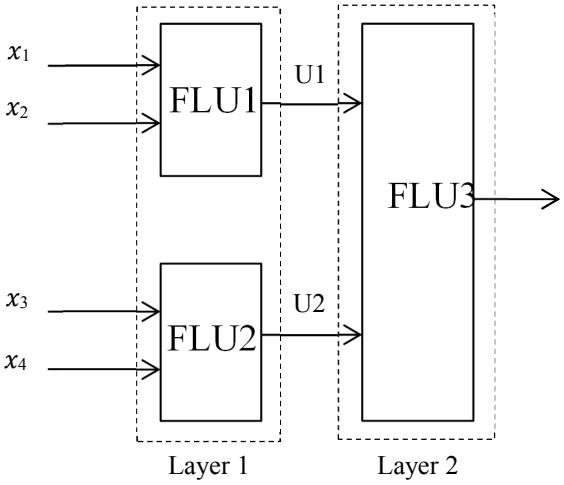

Fig. 3. The Structure of 4-Input L_HFS

III. The Proposed Hierarchical Fuzzy System

The proposed system includes the development of FPGA based 4-input variables ($x1$, $x2$, $x3$, and $x4$) system. Each one of these inputs has 5-membership functions and one output. This system is implemented in two schemes: the conventional single layer and the "Hierarchical Fuzzy System" with two cascaded layers. The used membership function is chosen as triangular shape which is used for all inputs, each input has five membership functions: NB(negative big), NS(negative small), ZE(zero), PS(positive small), and PB(positive big) as shown in Fig. 4.

Real value input in range [-1,1] is mapped to hexadecimal numbers in the range ["00","FF"], such that "00H" represents -1 and "FFH" represents 1. Mapping process is done by Equation 4:

$$X_R = \frac{2(x_H)}{FF} - 1 \qquad \ldots\ldots (4)$$

where
X_R: represent the real value.
x_H: represent equivalent hexadecimal value.

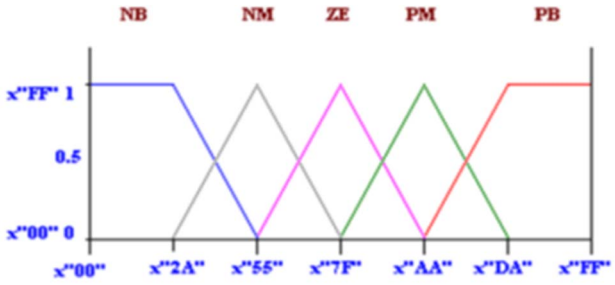

Fig. 4. The Input Membership Function

Fuzzification process has two arrays each one with five elements, fn and msf. The first array fn is represented as a binary vector (0, 1) indicates to which set , membership function set, the input is lie, while the second array ,msf, is represented the value of membership function belongs to this input.

A. Conventional Single Layer Implementation

The fuzzification part consists of four sub-units for each input ($x1$, $x2$, $x3$, and $x4$) operate simultaneously. The outputs of fuzzification part are fed as inputs for rule base part, as the antecedent linguistic terms. The total number of rules computed according [m^n] formula; therefore the rules number will become 54 = 625. Each rule is represented as:

If $x_1$is nb and x_2 is nb and x_3 is nb and x_4 is nb then out is nb

If $x_1$is nb and x_2 is nb and x_3 is nb and x_4 is ns then out is nb

.

.

If $x_1$is pb and x_2 is pb and x_3 is pb and x_4 is pb then out is pb

FPGA RTL-viewer is illustrated in Fig. 5. The output array of rule-base stage becomes an input to the defuzzification part. Final system output is determined using weighted average defuzzification method.

B. Hierarchical Implementation of 4-inputs HFS

The implementation of 4-input hierarchical fuzzy system using FPGA is done as a collection of two input fuzzy logic units. These logic units are arranged in two layers as shown in Fig. 3. The first layer includes two 2-inputs FLUs each of them take two input variables (x1and x2 for FLU1, x3and x4 for FLU2). Both FLUs have exact architecture. First layer FLUs outputs are represented in an array form. Layer1 FLUs operate in parallel manner. The outputs of each first layer FLU (U1 and U2) is fed to the second layer FLU3. The second layer FLU3 has the same architecture of the first layer FLU1 but without the fuzzification part. All these FLUs are implemented in a single FPGA chip.

The total rules number generated using" Limpid Hierarchical Fuzzy system L-HFS" is 75(25 rules for each FLU) instead of 625rules used for conventional fuzzy system. The intermediate variables result from the first layer FLU1 are mapped in to 5-mapping variables (A1, A2, A3,

A4, A5) while the intermediate variables result from the first layer FLU2 are mapped in to 5-mapping variables (B1, B2, B3, B4, B5). The second layer FLU3 rules determine the final rule base expressions from the two mapping variable sets U1 and U2 and produce the system output after defuzzification process.

These intermediate variables are grouped in to two sets of mapping variables U1 and U2, such that U1={A1,A2,A3,A4,A5} and U2={B1,B2,B3,B3,B4,B5}. The relationship for these two sets refers in Table I. The columns represent the variables of set U1, vertical mapping variables, while the rows represented the variables of set U2, horizontal mapping variables. Fig. 6 illustrates the FPGA RTL-viewer for hierarchical 4-inputs system resulting from the implementation of this system in FPGA.

IV. Results

"Multi input fuzzy logic system" is implemented using FPGA with Hierarchical architecture. The proposed hierarchical architecture is compared to the conventional implementation for 4-input fuzzy system. The proposed FPGA architecture is done using some families (Cyclone II, Cyclone III, Cyclone IV, Startix II, Startix III , and Startix IV). The performance is measured using circuit timing, speed up factor, logic elements, dedicated logic registers; number of LABs, total combinational function, and total used pins of the chip.

Both conventional and hierarchical implementation test bench waves of are shown in Fig. 7 and Fig. 8 respectively. As it is obvious from these two Figures:

- The required operation time for hierarchical implementation is 144.5 ns (17 clock pulses with 8.5 ns period).
- The required operation time for conventional implementation is 210 ns (21 clock pulses with 10 ns period).

- Software implementation of hierarchical 4-input need (109.774) us.
- Software implementation of conventional 4-input need (116.021) us.

Speed up factor (conventional) =759.681
Speed up factor (hierarchical) =552.48
Speed up factor (hierarchical FPGA over conventional software) =802.913

Table II shows the hardware characteristics of the proposed hardware architecture for both conventional and hierarchical FPGA implementations.

V. Conclusions

Results show that hierarchical implementation of fuzzy logic system using FPGA gives very attractive characteristics when compared to both the software FLS implementation and also the conventional FLS hardware implementation. Multi-variable FLS implementation consumption both run time and system resources. These requirements make conventional implementation improper for applications need fast executions or in the case of limited system resources.

For 4-inputs multi-variables FLS, the FPGA based implementation of conventional system is imposable using FPGA families such as Cyclone, Cyclone II, Cyclone III, and Stratix II because of the resources are not enough and less than that required. The implementation of such conventional system needs FPGA kit with more available resources (more prices) to implement 4-input conventional fuzzy systems. On the other hand, hierarchical fuzzy system FPGA implementation is possible for all the used test FPGA families with low usage ratio. As a result, hierarchical multivariable fuzzy systems can be implemented using wide range of FPGA families with high speed up factors. On the other hand, the conventional FLS implementation needs high resources ratio and cannot be implemented using simple or moderate specification FPGA chips especially when the number of variables increased.

References

[1] A. Titinchi ; N. Halasa “FPGA implementation of simplified Fuzzy LRU replacement algorithm”, 16th International Multi-Conference on Systems, Signals & Devices (SSD), Istanbul, Turkey, pp. 657 – 662, 21-24 March 2019.

[2] S. Guillaumea and B. Charnomordicb, “Learning Interpretable Fuzzy Inference Systems with FisPro”, Information Sciences, Special Issue on Interpretable Fuzzy Systems, vol. 181, no. 20, pp. 4409-4427, 2011.

[3] M. Lotfy, T. Senjyu, M. Farahat, A. Abdel-Gawad, and H. Matayoshi, “A polar fuzzy control scheme for hybrid power system using vehicle-to-grid technique,” *Energies*, vol. 10, no. 8, 2017.

[4] Y. Tan and B. Rosdi “FPGA-Based Hardware Accelerator for The Prediction of Protein Secondary Class Via Fuzzy K-nearest Neighbors With Lempel–Ziv Complexity Based Distance Measure”, Neurocomputing, Elsevier, vol.148, no.1, pp. 409-419, 2015.

[5] R.Harikumar and M. BalaSubramani , “FPGA Synthesis of Fuzzy Based SimpleTraffic Controller”, International Journal of Advanced Research in Computer and Communication Engineering,vol. 2, no. 7, pp. 2497-2502, 2013.

[6] N. Pathade and S. Ansari, “Speed Control of DC Motor for Robotic Application Using Fuzzy Controller on FPGA Platform”, International Journal of Advanced Research in Computer Science and Software Engineering , vol. 4, no. 12, pp. 410-414, 2014.

[7] E. Anarmarzi, M. Feyzi and M. Hagh, “Hierarchical fuzzy controller applied to multi-input power system stabilizer”, Turkish Journal of Electrical Engineering and Computer Science, vol. 18, no. 4, pp. 541-551, 2010.

[8] Chia-Wen Chang, Chin-Wang Tao, “A simplified implementation of hierarchical fuzzy systems”, Soft Computing, Springer, vol. 23, Issue 12, pp. 4471–4481, June 2019.

[9] G. Aidong, W. Yuzhen, L. Zhenbin, and J. Ping, “Hierarchical Fuzzy Control of Multivariable Systems via Semi-tensor Product Method”, Proceedings of the 33 rd Chinese Control Conference, pp. 4551-4556,2014.

[10] X. Zeng and J. Keane, “Approximation Capabilities of Hierarchical Fuzzy Systems”, IEEE Transections on Fuzzy Systems, vol. 13, no. 5, pp. 659-672, 2005.

[11] G. Raju, J. Zhou, and R. Kisner, “Hierarchical Fuzzy Control”, International Journal of Control, , vol. 41, no.5, pp. 1201-1216, 1991.

[12] R. Hmida, A. Ben Abdelali, and A. Mtibaa, “Hardware implementation and validation of a traffic road sign detection and identification system,” *Journal of Real-Time Image Processing*, vol. 15, no. 1, pp. 13–30, 2018.

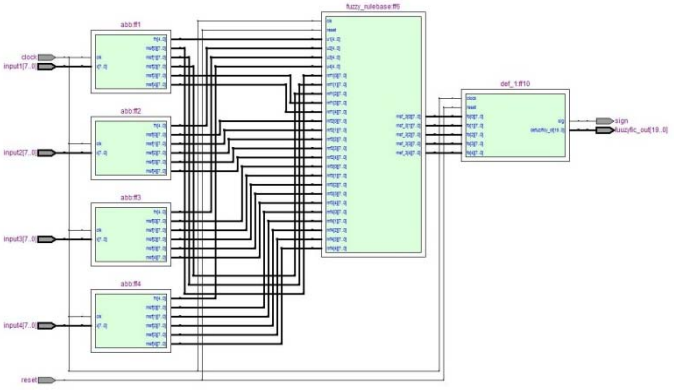

Fig. 5. FPGA RTL-viewer of Convintional 4-input fuzzy system

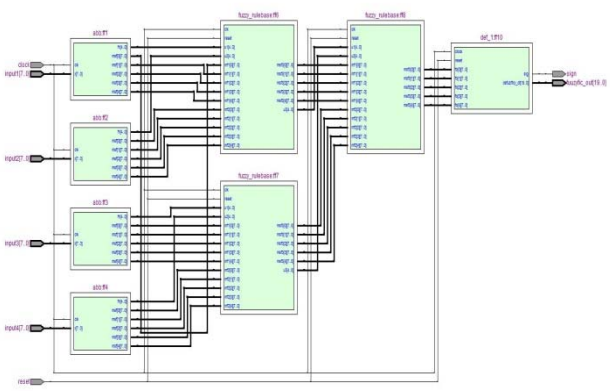

Fig. 6. FPGA RTL-viewer of hierarchiacal 4-input fuzzy system

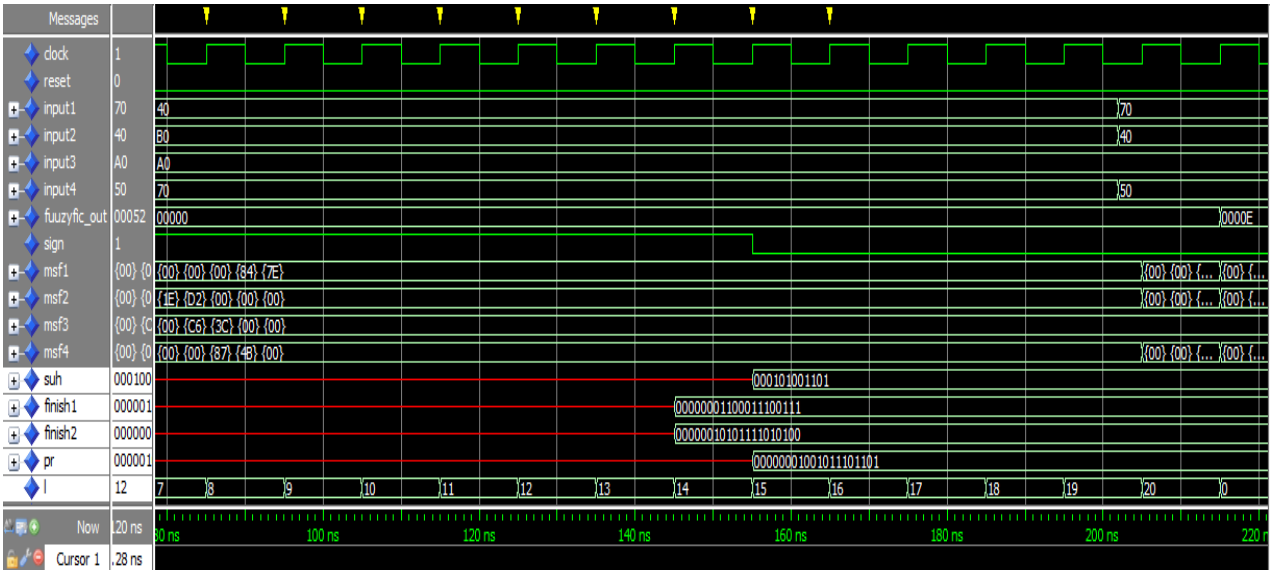

Fig. 7. Test Bench of Conventional 4-input Fuzzy System

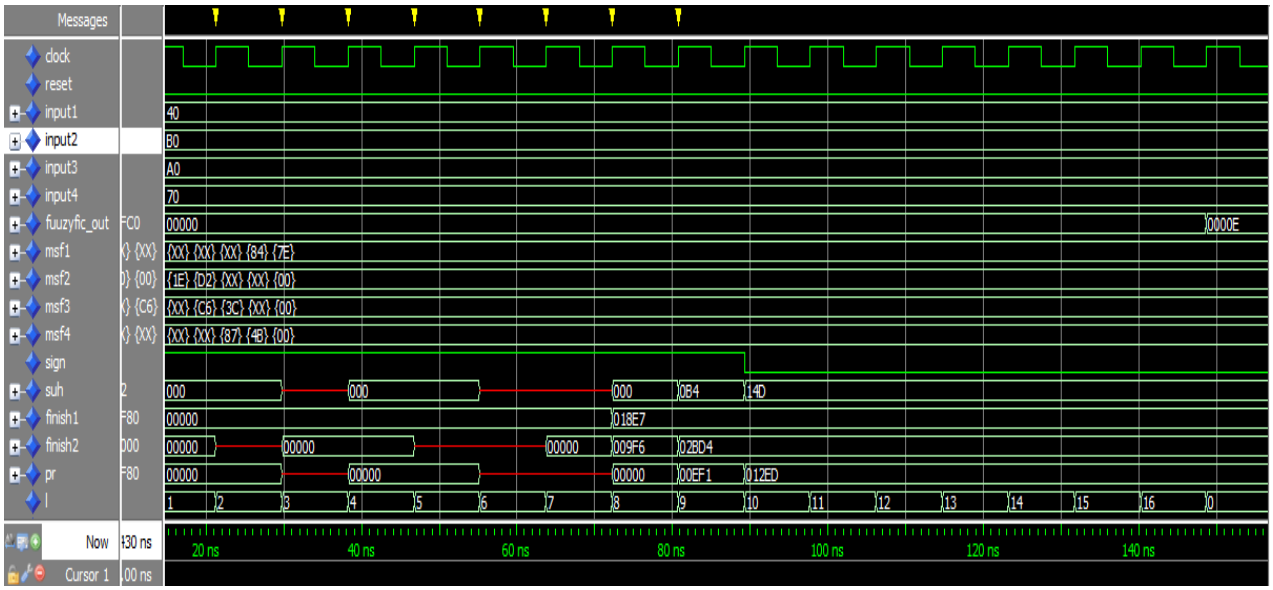

Fig. 8. Test Bench of Hierarchical 4-input Fuzzy System

Table I. Relationship Mapping of 4-input HFS

				X2																							
				NB					NS					Z					PS					PB				
				X1					X1					X1					X1					X1				
				NB	NS	Z	PS	PB	NB	NS	Z	PS	PB	NB	NS	Z	PS	PB	NB	NS	Z	PS	PB	NB	NS	Z	PS	PB	
X4	NB	X3	NB	NB	NB	NB	NB	NB	NB	NB	NB	NB	NS	NB	NB	NB	NS	Z	NB	NB	NS	NS	Z	NB	NS	Z	Z	Z	B1
			NS	NB	NB	NB	NB	NB	NB	NB	NB	NB	NS	NB	NB	NB	NS	Z	NB	NB	NS	NS	Z	NB	NS	Z	Z	Z	B1
			Z	NB	NB	NB	NB	NB	NB	NB	NB	NB	NS	NB	NB	NB	NS	Z	NB	NB	NS	NS	Z	NB	NS	Z	Z	Z	B1
			PS	NB	NB	NB	NS	NS	NB	NS	NS	NS	Z	NB	NS	NS	Z	PS	NS	NS	Z	Z	PS	NS	Z	PS	PS	PS	B2
			PB	NB	NB	NB	NS	Z	NB	NS	NS	Z	PS	NB	NS	Z	PS	PB	NS	Z	PS	PS	PB	Z	PS	PB	PB	PB	B3
	NS	X3	NB	NB	NB	NB	NB	NB	NB	NB	NB	NB	NS	NB	NB	NB	NS	Z	NB	NB	NS	NS	Z	NB	NS	Z	Z	Z	B1
			NS	NB	NB	NB	NS	NS	NB	NS	NS	NS	Z	NB	NS	NS	Z	PS	NS	NS	Z	Z	PS	NS	Z	PS	PS	PS	B2
			Z	NB	NB	NB	NS	NS	NB	NS	NS	NS	Z	NB	NS	NS	Z	PS	NS	NS	Z	Z	PS	NS	Z	PS	PS	PS	B2
			PS	NB	NB	NB	NS	Z	NB	NS	NS	Z	PS	NB	NS	Z	PS	PB	NS	Z	PS	PS	PB	Z	PS	PB	PB	PB	B3
			PB	NS	NS	NS	Z	PS	NS	Z	Z	PS	PS	NS	Z	PS	PS	PB	Z	PS	PS	PS	PB	PS	PS	PB	PB	PB	B4
	Z	X3	NB	NB	NB	NB	NB	NB	NB	NB	NB	NB	NS	NB	NB	NB	NS	Z	NB	NB	NS	NS	Z	NB	NS	Z	Z	Z	B1
			NS	NB	NB	NB	NS	NS	NB	NS	NS	NS	Z	NB	NS	NS	Z	PS	NS	NS	Z	Z	PS	NS	Z	PS	PS	PS	B2
			Z	NB	NB	NB	NS	Z	NB	NS	NS	Z	PS	NB	NS	Z	PS	PB	NS	Z	PS	PS	PB	Z	PS	PB	PB	PB	B3
			PS	NS	NS	NS	Z	PS	NS	Z	Z	PS	PS	NS	Z	PS	PS	PB	Z	PS	PS	PS	PB	PS	PS	PB	PB	PB	B4
			PB	Z	Z	Z	PS	PB	Z	PS	PS	PB	PB	Z	PS	PB	PB	PB	PS	PB	PB	PB	PB	PB	PB	PB	PB	PB	B5
	PS	X3	NB	NB	NB	NB	NS	NS	NB	NS	NS	NS	Z	NB	NS	NS	Z	PS	NS	NS	Z	Z	PS	NS	Z	PS	PS	PS	B2
			NS	NB	NB	NB	NS	Z	NB	NS	NS	Z	PS	NB	NS	Z	PS	PB	NS	Z	PS	PS	PB	Z	PS	PB	PB	PB	B3
			Z	NS	NS	NS	Z	PS	NS	Z	Z	PS	PS	NS	Z	PS	PS	PB	Z	PS	PS	PS	PB	PS	PS	PB	PB	PB	B4
			PS	NS	NS	NS	Z	PS	NS	Z	Z	PS	PS	NS	Z	PS	PS	PB	Z	PS	PS	PS	PB	PS	PS	PB	PB	PB	B4
			PB	Z	Z	Z	PS	PB	Z	PS	PS	PB	PB	Z	PS	PB	PB	PB	PS	PB	PB	PB	PB	PB	PB	PB	PB	PB	B5
	PB	X3	NB	NB	NB	NB	NS	Z	NB	NS	NS	Z	PS	NB	NS	Z	PS	PB	NS	Z	PS	PS	PB	Z	PS	PB	PB	PB	B3
			NS	NS	NS	NS	Z	PS	NS	Z	Z	PS	PS	NS	Z	PS	PS	PB	Z	PS	PS	PS	PB	PS	PS	PB	PB	PB	B4
			Z	Z	Z	Z	PS	PB	Z	PS	PS	PB	PB	Z	PS	PB	PB	PB	PS	PB	PB	PB	PB	PB	PB	PB	PB	PB	B5
			PS	Z	Z	Z	PS	PB	Z	PS	PS	PB	PB	Z	PS	PB	PB	PB	PS	PB	PB	PB	PB	PB	PB	PB	PB	PB	B5
			PB	Z	Z	Z	PS	PB	Z	PS	PS	PB	PB	Z	PS	PB	PB	PB	PS	PB	PB	PB	PB	PB	PB	PB	PB	PB	B5
				A1	A1	A1	A2	A3	A1	A2	A2	A3	A4	A1	A2	A3	A4	A5	A2	A3	A4	A4	A5	A3	A4	A5	A5	A5	

Table II. Hardware Characteristics of The Proposed Fuzzy System

FPFA Family	Fuzzy system Mode	Total Logic Element	Combinational Function	Dedicated Logic Register	Number of LABs	Total Pins
Cyclone II EP2C8AF25618	Conventional	Cannot Be Executed (*No Enough Recourses*)				
	Hierarchical	2,799/8,256 (34%)	2,779/8,256 (34%)	1,427/8,256 (17%)	223/516 (43%)	55/182 (30%)
Cyclone III EP3CI6F484C8	Conventional	Cannot Be Executed (*No Enough Recourses*)				
	Hierarchical	2,799/15,408 (18%)	2,779/15,408 (18%)	1,427/15,408 (9%)	225/963 (23%)	55/347 (16%)
Cyclone IV E EP4CE115F29C8	Conventional	13,583/114,480 (12%)	13,536/114,480 (12%)	10,551 /114,480 (9%)	1,388/7,155 (19%)	55/529 (10%)
	Hierarchical	2,808/114,480 (2%)	2,783/114,480 (2%)	1,429 /114,480 (1%)	236/7,155 (3%)	55/529 (10%)
Stratix II EP2515F48414	Conventional	Cannot Be Executed (*No Enough Recourses*)				
	Hierarchical	2,786/12,480 (22%)	2,637/12,480 (21%)	1,428/12,480 (11%)	220/780 (28%)	55/343 (16%)
Stratix III EP3SE50F780C4	Conventional	13,509/38,000 (36%)	13,393/38,000 (35%)	10,549/38,000 (28%)	1,366/1,900 (72%)	55/488 (11%)
	Hierarchical	2,844/38,000 (7%)	2,639/38,000 (7%)	1,427/38,000 (4%)	201/1,900 (11%)	55/488 (11%)
Stratix IV EP4SE820H40C4	Conventional	24,426/650,440 (4%)	13,433 /650,440 (2%)	10,551 /650,440 (2%)	1,348 /32,522 (4%)	55/976 (6%)
	Hierarchical	3,549/650,440 (< 1%)	2,680/650,440 (< 1%)	1,429 /650,440 (< 1%)	206 /32,522 (<1%)	55/976 (6%)