FPGA Implementation of Adaptive Neuro-Fuzzy Inference Systems Controller for Greenhouse Climate

Charaf eddine LACHOURI

Electronics department
University Badji Mokhtar
Annaba, Algeria

Aissa BELMEGUENAI

Electronics research Laboratory
University 20 August 1955
Skikda, Algeria

Khaled MANSOURI

Electronics department
University Badji Mokhtar
Annaba, Algeria

Mohamed mourad LAFIFI

Electronics department
University Badji Mokhtar
Annaba, Algeria

Abstract—**This paper describes a Field-programmable Gate Array (FPGA) implementation of Adaptive Neuro-fuzzy Inferences Systems (ANFIS) using Very High-Speed Integrated Circuit Hardware-Description Language (VHDL) for controlling temperature and humidity inside a tomato greenhouse. The main advantages of using the HDL approach are rapid prototyping and allowing usage of powerful synthesis controller through the use of the VHDL code. The use of hardware description language (HDL) in the application is suitable for implementation into an Application Specific Integrated Circuit (ASIC) and Field tools such as Quartus II 8.1. A set of six inputs meteorological and control actuators parameters that have a major impact on the greenhouse climate was chosen to represent the growing process of tomato plants. In this contribution, we discussed the construction of an ANFIS system that seeks to provide a linguistic model for the estimation of greenhouse climate from the meteorological data and control actuators during 48 days of seedlings growth embedded in the trained neural network and optimized using the backpropagation and the least square algorithm with 500 iterations. The simulation results have shown the efficiency of the implemented controller.**

Keywords—Neuro-Fuzzy; ANFIS; VHDL; FPGA; Quartus; ASIC

I. INTRODUCTION

Under greenhouse production, the climate control is a tool used for yield crop manipulation that maximizes the entrepreneurial benefits. Once the objectives that optimize crop growth and development are defined, the control engineer must design and implement automatic control systems that make possible to obtain a maximum crop yield at minimum production costs. In this sense, control engineering has undergone a considerable development. Researchers have used many control techniques in different fields, from the conventional or classic strategies [proportional integral derivative (PID) control, cascade], artificial intelligence (AI) (fuzzy control, neural networks and genetic algorithms), advanced control techniques (predictive control, adaptive), to robust control strategies, non-linear and optimal control. Specifically, they have been applied in the area of greenhouse climate control [1][2][3]. Conventional control techniques are difficult to implement in greenhouse systems due to their multi-variable and non-linear nature. Where interrelations between internal and external variables are complex (non-linear physical phenomena that govern these systems dynamics are complicated). This provides justification for the use of intelligent control techniques as a good alternative. In this way, fuzzy logic as part of AI techniques is an attractive and well-established approach to solving control problems [4].

We were brought to develop a Neuro-Fuzzy control of the internal humidity and internal temperature of the greenhouse. This last characterizes the operation of the complex system that the greenhouse constitutes. The identification that is in the center of this step is a process of search for a mathematical representation that minimizes the variations of the real system compared to the modeled system. The development of the plant is influenced mainly by the environmental, climatic variables. The greenhouse, which is a closed circle in which the climatic variables can be controlled, constitutes the ideal medium for the control of the plants growth. The greenhouse must not only create the favorable conditions of the plants growth, but it must moreover be able to ensure certain flexibility in the calendar of production: precocity and spreading out of the calendar. To carry out this objective a robust model using the Artificial Neural Networks and the fuzzy logic can be well adapted to control the nonlinear comportment of greenhouse climate accurately is more than necessary [5].

For the implementation of agricultural technologies (innovations in control systems, remote monitoring, information management), robustness, low-cost and real-time capabilities are needed. In this sense, field programmable gate arrays (FPGAs) proved as a good option for greenhouse technology development and implementation, because FPGAs allow fast development of prototypes and the design of complex hardware systems. These devices are used in many real applications [6]. Through FPGAs, rapid tests, modifications accomplishment, up-dates using single software modifications and an effective production cost (relation performance-price is very favorable) are obtained. In the same sense, reduction in development and commercialisation time is

accomplished. On the other hand, for neuro-fuzzy control implementation, which based on software or hardware, FPGAs are an alternative that keep both benefits, hardware speed and software flexibility. Research made about these devices has experienced an enormous development, in the academic field as well as in the industrial area. There is a great number of contributions about FPGAs applications in different fields [7][8][9]. Also, there are some contributions reported about hardware implementations of neuro-fuzzy control [10]. Moreover, problems of digitized neuro-fuzzy control have been studied [11]. The approach proposed here is focused on greenhouse technologies development, based on AI techniques, particularly fuzzy logic, cascaded with a feed-forward a neural network, and system-on-a-chip (SoC) applications using FPGA technology, with the purpose of obtaining complete engineering solutions on a single Integrated circuit. In our case, an intelligent SoC was developed to carry out the perfect functionality for the greenhouse climate control due to an ANFIS system that seeks to provide a linguistic model for the estimation of greenhouse climate from the meteorological data and control actuators during 48 days of seedlings growth embedded in the trained neural network and optimized using the backpropagation and the least square algorithm with 500 iterations.

II. NEURONAL METHODS IN THE FUZZY SYSTEMS

In a conventional fuzzy inference system, the number of rules is decided by an expert who is familiar with the system to be modeled. In this particular case study the rules generated by an agriculture expert and the number of membership functions assigned to each input is chosen from real data. This is carried out by examining the desired and real input-output data. This situation is much the same as ANN's. In this section ANFIS topology and the learning method used for this neural network are presented. Both neural network and fuzzy logic are model-free estimators and share the common ability to deal with the uncertainties and noise. It is possible to convert fuzzy logic architecture to a neural network and vice versa [12]. This makes it possible to combine the advantages of neural network and fuzzy logic [13-14].

Layer 1: Every node in i in this layer is a square node with a node function

$$o_i^1 = \mu_{A_i}(x)$$ (1)

Where x is the input node i, and A_i is the linguistic label (Minimum, Moderate, Maximum) associated with this node function. In other words, o_i^1 is the membership function and it specifies the degree to which the A_i given x satisfies the quantifier A_i. Usually we choose $\mu_{A_i}(x)$ to be bell shaped with maximum equal to 11, moderate equal to 00 and minimum equal to 10, such as

$$\mu_{A_i}(x) = \frac{1}{1 + \left[\left(\frac{x - c_i}{a_i} \right)^2 \right]^{bi}}$$ (2)

Where $\{a_i, b_i, c_i\}$ is the parameter set. As the values of these parameters change, the best bell-shaped functions vary

accordingly, thus exhibiting various forms of membership functions on linguistic label A_i. In fact, any continuous and piecewise differentiable functions, such as commonly used trapezoidal or triangular-shaped membership functions are also qualified candidates for node functions in this layer. Parameters in this layer are referred to as premise parameters.

Layer 2: Every node in this layer is a circle node labeled $\prod$ which multiplies the incoming signals and sends the product out. For instance,

$$w_i = \mu A_i(x) * \mu A_i(y), i = 1, ..., 40$$ (3)

Each node output represents the firing strength of a rule (In fact, other T-norm operators that perform generalized AND can be used as the node function in this layer).

Layer 3: Every node in this layer is a circle node labeled N. The ith node calculates the ratio of the ith rule's firing strength to the sum of all rules firing strengths:

$$\overline{w_i} = \frac{w_i}{w_1 + ... + w_{40}}, i = 1, ..., 40$$ (4)

For convenience, outputs of this layer are called normalized firing strengths.

Layer 4: Every node in this layer is a square node with a node function

$$O_i^4 = \overline{w_i} f = \overline{w}(p_i x + q_i y + r_i)$$ (5)

Where $\overline{w_i}$ is the output of layer 3, and (p_i, q_i, r_i) is the parameter set. Parameters in this layer will be referred to as consequent parameters.

Layer 5: The single node in this layer is a circle node labeled Σ that computes the overall output as the summation of all incoming signals.

$$O_1^5 = overall output = \sum \overline{w_i} f = \frac{\sum_i w_i f}{\sum_i w_i}$$ (6)

Thus we have constructed an adaptive network which is functionally equivalent to a fuzzy inference system [14-15]. The hybrid algorithm is applied to this architecture. This means that, in the forward pass of the hybrid learning algorithm, functional signals go forward up to fourth layer and the consequent parameters are identified by the least squares estimation. In the last backward and the premise parameters are updated by the gradient descent [14].

A. ANFIS Predictive Architecture

Using a given input/output data set, the ANFIS method constructs a fuzzy inference system (FIS) whose membership function parameters are tuned (adjusted) using either a backpropagation algorithm alone, or in combination with a least squares type of method. This allows fuzzy systems to learn from the data they are modeling. FIS Structure is a network-type structure similar to that of a neural network, which maps inputs through input membership functions and associated parameters, and then through output membership functions and associated parameters to outputs [16].

In our case ANFIS is a four-layer neural network that simulates the working principle of a fuzzy inference system. The linguistic nodes in layers one and four represent the input and output linguistic variables, respectively. Nodes in layers two are term nodes acting as membership functions for input variables. Each neuron in the third layer represents one fuzzy rule, with input connections representing preconditions of the rule and the output connection representing consequences of the rules. Initially, all these layers are fully connected, representing all possible rules.

Six feature variables, internal temperature, internal humidity, external temperature, external humidity, global radiation and wind speed, are selected as inputs of the ANFIS. Three membership functions (Mfs) are assigned to each linguistic variable. The suggested ANFIS model is shown in "Fig. 1".
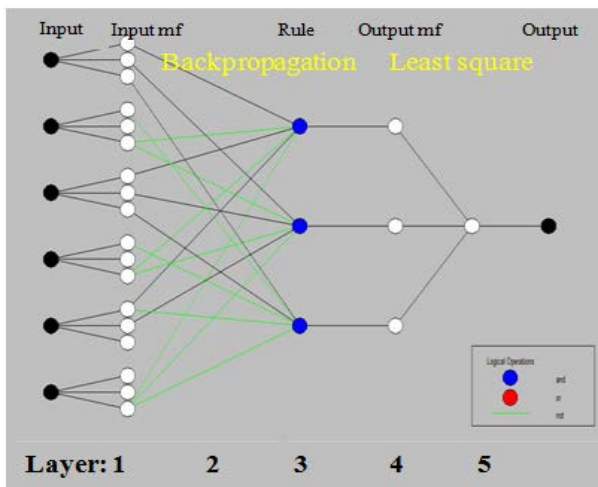

Fig. 1.   ANFIS model structure of greenhouse climate

B. ANFIS Modeling, Training and Testing

ANFIS modeling process starts by obtaining a data set (input-output data) and dividing it into training, testing and checking data sets. Training data constitutes a set of input and output vectors. The data is normalized in order to make it suitable for the training process. This was done by mapping each term to a value between 00, 01 and 10 using the Min, moderate and Max method. This normalized data was utilized as the inputs (internal climate conditions and meteorological data) and outputs (actuators conditions) to train the ANFIS. In other words, two vectors are formed in order to train the ANFIS. Input vector = [internal temperature, internal humidity, external temperature, external humidity, global radiation and wind speed]. The output vector = [Ventilating and heating]. The ANFIS registers the input data only in the numerical form therefore the information about the control actuators, internal and external climate of the greenhouse must be transformed into numerical code.

The training data set is used to find the initial premise parameters for the membership functions by equally spacing each of the membership functions. A threshold value for the error between the actual and desired output is determined. The consequent parameters are found using the least-squares method.

Then an error for each data pair is found. If this error is larger than the threshold value, update the premise parameters using the gradient decent method as the following (Qnext=Qnov+ηd, where Q is a parameter that minimizes the error, η the learning rate, and d is a direction vector). The process is terminated when the error becomes less than the threshold value. Then the checking data set is used to compare the model with actual system. A lower threshold value is used if the model does not represent the system.

"Fig. 2", shows the uniform falling of the value of testing error ETest with the number of iterations during the testing process for the ANFIS configuration with traingular Mf and with gaussian Mf. The smallest error of testing (ETest) is reached at iteration 145 (traingular Mf) and at iteration 107 for Gaussian Mf. It can be seen in the "Fig. 2", that error converges not to zero but to 12% and 2%. This is caused by the presence of some contradicting examples in the training and testing set.
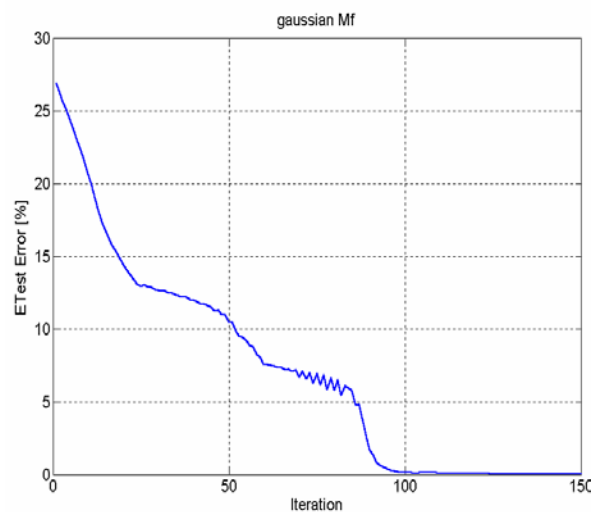

Fig. 2.   Decrease of error during the testing process for the ANFIS configuration with Traingular Mf and with Gaussian Mf

Training of the ANFIS can be stopped by two methods. In the first method, ANFIS will be stopped to learn only when the testing error is less than the tolerance limit. This tolerance limit would be defined at the beginning of the training. It is obvious that the performance of the ANFIS that is trained with lower tolerance is greater than ANFIS that is trained with higher tolerance limit. In this method the learning time will change with the architecture of the ANFIS. The second method to stop the learning is to put constraint on the number of learning iterations. In our study, the ANFIS architecture is stopped to learn after 500 training iterations.

III.   NEURO-FUZZY CLIMATE CONTROLLER

As is already known from neuro-fuzzy principles, a neuro fuzzy controller acts as a non-linear system capable of implementing expert reasoning for computation of the control values. Indeed, a neuro fuzzy controller which is defined by a set of linguistic rules and fuzzy sets were trained by neural network and optimized using the back-propagation and the least square algorithm is able to compute appropriate values

for greenhouse actuators (heating, ventilating) taking into account information data from the system for control proposes. In the experimental greenhouse, the temperature is controlled by means of heaters, while the humidity is controlled indirectly with the ventilating index regulation. That affects on the temperature and the humidity. Using the physical model, a complete system simulator is shown in "Fig. 3". With this simulator, a first experiment was carried out using a conventional controller (on-off) with a dead band of 2°C; this is based on a heating system that is activated or deactivated when the error exceeds the fixed regulation range. The humidity depends on the internal air temperature and the ventilating index. This variable is regulated by windows opening in the greenhouse according to the wind speed measurements.

In this case, a multiple inputs, multiple outputs (MIMO) non-linear controller for temperature regulation was used. A MIMO neuro fuzzy controller can be distributed in several multiple inputs, single output (MISO) controllers keeping the same performance. These controllers are independent and can be executed in parallel, which is advisable for the climate controller implementation in a FPGA. The Neuro Fuzzy Controller has six input variables and two output variables, characterized by three fuzzy sets in the universe of discourse.

Input variables are inside and outside temperature (Ti, Text), inside and outside humidity (Hi, Hext), global radiation (Gr) and wind speed (Ws). Membership functions sets and their appropriate modifications were obtained following a test and error strategy by making exhaustive simulations in Matlab until reaching a good performance through a careful tuning. "Fig. 4", shows an example of a membership functions set for the input. For this one, three linguistic variables were used (MIN, minimum; MOY, medium; MAX, maximum). The set of fuzzy rules to develop the controller for each variable has been obtained from the expert grower. For tuning the fuzzy rules as well as for membership functions sets a trial-and-error strategy (manual tuning) was used, this is modifying control rule sets until we reaching a good performance of the controller by using the ANFIS editor (simulation system). Each possible linguistic value of inputs is assigned to a consequential action.
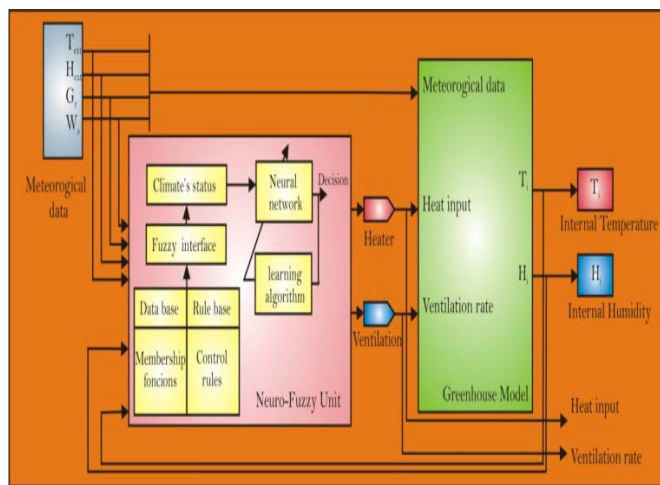

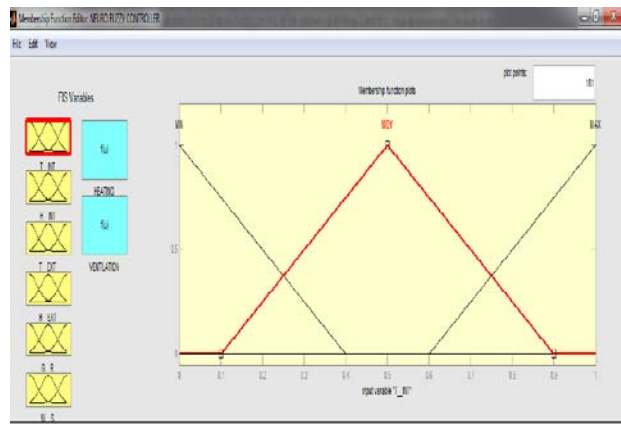Fig. 3.   The model of the greenhouse control system

Fig. 4.   Membership function of internal temperature

IV.   DESIGN AND HARDWARE IMPLEMENTATION

The Neuro Fuzzy Controller shown in "Fig. 5", has been implemented on an FPGA. The hardware platform used is the Altera DE2 development and education board that is based on the Altera Cyclone II EP2C35F672C6 FPGA.
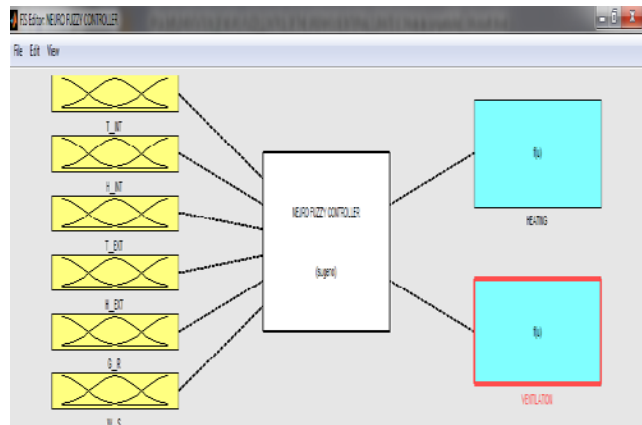

Fig. 5.   Neuro Fuzzy Controller

In order to implement our application effectively the design is broken down into modules.

A. The de-multiplexer component

The system should accept multiple inputs with 8-bits in total of 48-bits. In order to reduce the number of pins used in FPGA we have made a de-multiplexing as shown in "Fig. 6", it has one input of 8-bits and three selection lines, in order to learn at each clock pulse one input and he settles it into a buffer. After six clock top it will acquire all inputs. At the seventh clock pulse it delivers the enable signal and the values of multiples inputs to the rest of the system.

B. The Fuzzier module

In this section we have realized six blocks, where each block is intended for one of memberships functions. The example of such block is presented in the figure it used for the external temperature given in *"Fig. 7"*. The blocks transformed the numerical data to three linguistic variables (MIN, MOY, MAX). For easy implementation and as we have three cases two bits are used to materialize these case as follows (min => 10, moy => 00 and max => 11).

Fig. 6.   The de-mulplixer component

Fig. 7.   The fuzzier module

C. The command module

The following operation is the order of the ventilating and the heating. This component shown in "Fig. 8", admits at the inputs the various decisions for the multiple inputs and it will computing the rules of our FIS structure obtained by Matlab Fuzzy Logic Toolbox. To reduce the use of the hardware resource, finite state machine (FSM) is adopted to model this computing   process. Finally it will transform the linguistics values on the binary values.

Fig. 8.   The command component

V.   RESULTS AND DISCUSSION

The next step is the simulation of the design to illustrate how it works. "Fig. 9", shows the global simulation timing obtained by Quartus II version 8.1 SJ Web edition. 'Data' are the input values information, (T_int, T_ext, H_int, H_ext, W_s and R_g) are the values of the deferent parameters, (fuz_Tint, fuz_Text, fuz_Hint, fuz_Hext, fuz_Ws and fuz_Rg) are the resultant of all the membership functions.

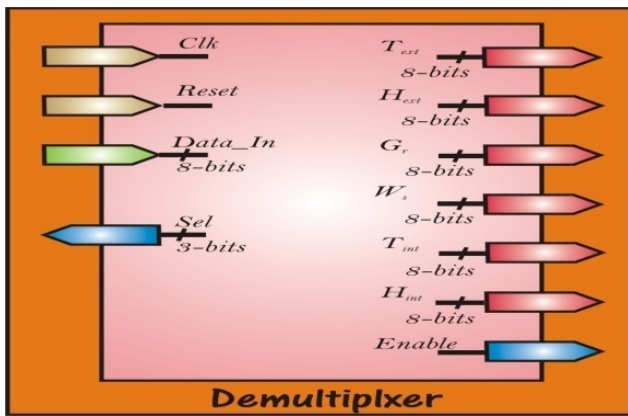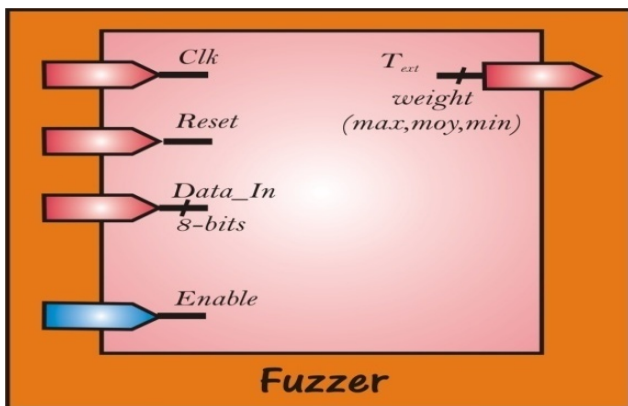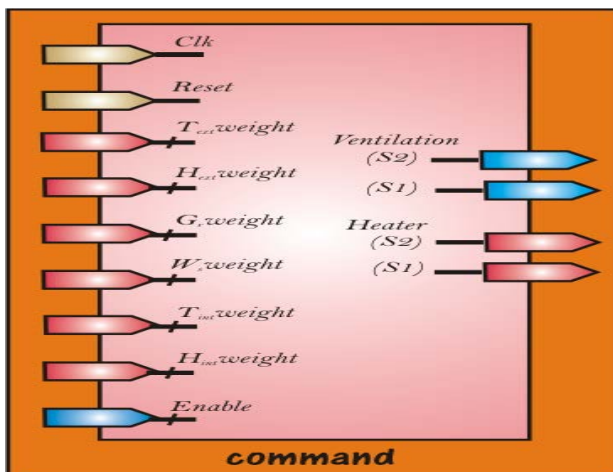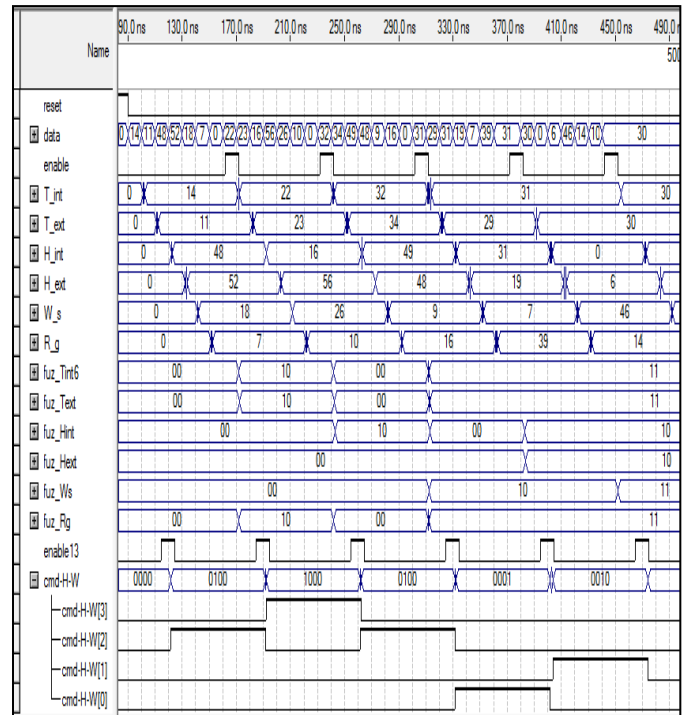'Cmd-H-W' is the finally output value represent the ventilating and heating.

Fig. 9.   Neuro Fuzzy Controller Quartus II simulation

The table I shows the strong similarity between the results   obtained   by Matlab Fuzzy Logic Toolbox environment and those obtained in "Fig. 9". It shows the best operation of all modules. We can also see how the transformation of these data from the linguistic values to numerical values.

Synthesis of fuzzy neural network on FPGA:

We have implemented the design using the DE2 board, contain Cyclone®II 2C35Altera FPGA device, EP1C6Q240. The principal features of Cyclone II EP2C35 FPGA are as follows:

- 33216 Logic elements.
- 105 M 4K RAM blocks.
- 483,840totalRAMbits.
- 35 embedded 1818 multipliers.
- Four PLLs.
- 475 user I/O pins.

TABLE I. COMPARISON OF THE RESULTS GIVEN BY MATLAB FUZZY LOGIC TOOLBOX AND THOSE OBTAINED WITH QUARTUS II

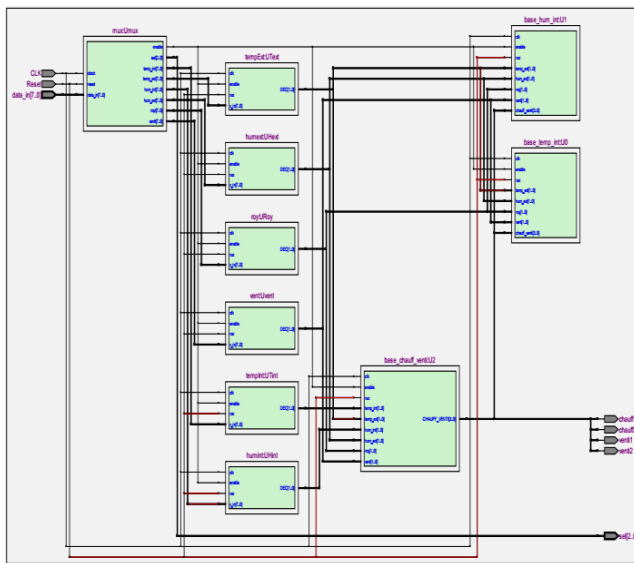| Meteorological data				Greenhouse data		Decision control			
T_ext	H_ext	G_r	W_s	T_int	H_int	Heating S2	Heating S1	Ventilation S2	Ventilation S1
MIN	MOY	MIN	MOY	MOY	MOY	MAX		MIN	
10	00	10	00	00	00	1	0	0	0
MOY	MOY	MOY	MOY	MOY	MIN	MOY		MIN	
00	00	00	00	00	10	0	1	0	0
MAX	MOY	MAX	MIN	MAX	MOY	MIN		MOY	
11	00	11	10	11	00	0	0	0	1
MAX	MIN	MAX	MIN	MAX	MIN	MIN		MAX	
11	10	11	10	11	10	0	0	1	0

Fig. 10. RTL schematics of neuro-fuzzy controller

VI. CONCLUSION

The current work focuses on the application of neuro-fuzzy control of a greenhouse internal climate. It successfully demonstrated the performance through co-simulation by using ANFIS and ModelSim. This implementation accurately reproduces the theoretical behavior of the system, thus is ready to be used. The future work will be destined to improve the design of our work including the number and the type of inputs membership functions.

REFERENCES

[1] Lees M J; Taylor J; Chotai A; Young Z S; Chalabi Z S(1996). Design and implementation or a proportional-integral plus (PIP) control system for temperature, humidity and carbon dioxide in a glasshouse. Acta Horticulturae, 406, 115–124.

[2] Hanan J(1998). Greenhouses: Advanced Technology for Protected Horticulture, 1st Edn. CRC Press, New York, USA.

[3] Tap F(2000). Economics-based optimal control of greenhouse tomato crop production. PhD Thesis, Wageningen Agricultural University, The Netherlands.

[4] Lee C C(1990). Fuzzy logic in control systems: fuzzy logic controller—(parts I and II). Transactions on Systems, Man, and Cybernetics,20,404–432.

[5] Draoui B(1994). Characterisation and analyses hydrous assessment thermo of a horticultural greenhouse. In situ identification of the parameters of a dynamic model, Thesis of Doctorate of the university of Nice Sophia Antipolis, France, 1994.

[6] Vega-Rodriguez M A; Sanchez-Perez J M; Gomez-Pulido J A (2004). Special issue on FPGAs: applications and designs. Microprocessors and Microsystems,28,193–195.

[7] Ali L; Sidek R; Aris Ishak; Ali A M; Suparjo B S(2004). Design of a micro-UART for SoC application. Computers and Electrical Engineering,30,257–268.

[8] Romero-Troncoso R; Herrera-Ruiz G; Terol-Villalobos I; Jauregui-Correa J C (2004). FPGA based on-line tool breakage detection system for CNC milling machines. Mechatronics,14,439–454.

[9] Mendoza-Jasso J; Ornelas-Vargas G; Castan˜eda-Miranda R; Ventura-Ramos E; Zepeda-Garrido A; Herrera-Ruiz G (2005). FPGA-based real-time remote monitoring system. Computers and Electronics in Agriculture, 49(2), 272–285.

[10] Dipali L.Gaikwad; Prabha Kasliwal (2013). FPGA Based Critical Patient Health Monitoring Using Fuzzy Neural Network. International Journal of Scientific & Engineering 2229-5518.

[11] Dhananjay E. Upasani (2010). FPGA implementation of intelligent climate control for greenhouse. International Journal of Computer Applications (0975 - 8887).

[12] H. C. Cho, and K. S. Lee, "Adaptive control and stability analysis of nonlinear crane systems with perturbation," *J. Mech. Sci. Techn.*, vol. 22, pp. 1091-1098, 2008.

[13] M. Y. El Ghoumari, H. J. Tantau, and J. Serrano, "Nonlinear constrained MPC: real-time implementation of greenhouse air temperature control," *Comput. Elect. Agric.*, vol. 49, pp. 345–356, 2005.

[14] F. Lafont, and J. F. Balmat, "Fuzzy logic to the identification and the command of the multidimensional systems," *International J. Comput. Cognition,* vol. 2, pp. 21–47, 2004.

[15] F. Fourati, and M. Chtourou, "A greenhouse control with feed-forward and recurrent neural networks," *Simulation Modeling Pract. and Theory,* vol.15, pp. 1016–1028, 2007.

[16] J.S.R. Jang "ANFIS: Adaptive-Network-Based Fuzzy Inference System", *IEEE Trans. Systems, Man, Cybernetics*, 23(5/6):665-685, 1993.