International Journal of Reconfigurable and Embedded Systems (IJRES)
Vol. 10, No. 1, March 2021, pp. 37~46
ISSN: 2089-4864, DOI: 10.11591/ijres.v10.i1.pp37-46

Hamming neural network application with FPGA device

Liqaa Saadi Mezher
Department of Computer Engineering, Al-Mustansiriyah University, Iraq

Article Info

Article history:

Received May 4, 2020
Revised Oct 29, 2020
Accepted Jan 25, 2021

Keywords:

Artificial neural network
Feed forward layer
FPGA
Hamming neural network
Recurrent layer
VHDL

ABSTRACT

The Hamming neural network is a kind of counterfeit neural system that substance of two kinds of layers (feed forward layers and repetitive layer). In this study, two pattern entries are utilization in the binary number. In the first layer, two nerves were utilization as the pure line work. In the subsequent layer, three nerves and a positive line work were utilization. The Hamming Neural system calculation was also implemented in three reproduction strategies (logical gate technique, programming program encryption strategy and momentary square chart technique). In this study in programming of VHDL and FPGA machine was utilization.

Corresponding Author:

Liqaa Saadi Mezher
Department of Computer Engineering
Al Mustansiriyha University
Baghdad, Iraq
Email: iqa35@uomustansiriyah.edu.iq

1. INTRODUCTION

Artificial neural networks (ANN) are in an innovation of man-made brainpower imaginative by the human cerebrum and nerves framework and generally use to the shape and troublesome upgrade marvels including a huge estimation of procedure factors [1-3]. Counterfeit neural systems are computational approach's that lead to in various analyzers [4]. Roused by organic neural systems, it is a phony neural system that contains layers of straightforward computational hubs that go about as individuals summing up the nonlinear models of systems [5]. Field programmable gate arrays (FPGA) machine is an equipment apply enormous estimation of counterfeit nerves arrange [6]. Exceptionally rapid integrated circuits hardware description language (VHDL) the product actualized to murmuring nerves organize use numerous ways [7]. The examination is portrayal as in the accompanying. In the segment two and a concise audit a presentation hamming neural network about the components for examples, loads and what's more, the exchange work unadulterated line and pos line. In section three, portray hamming neural network's calculation. In segment four, utilization ANN structure. Hamming neural network appended by numerous reenactment techniques description applied by VHDL programming and FPGA machine. Segment five, portray the last yield of Hamming neural network, end segment six, are given some significant ends.

2. HAMMING NEURAL NETWORK

In this examination, the primary elements of this system utilized, a neural network architecture contains multiple neural networks (numerous layers), as is explained in Figure 1.

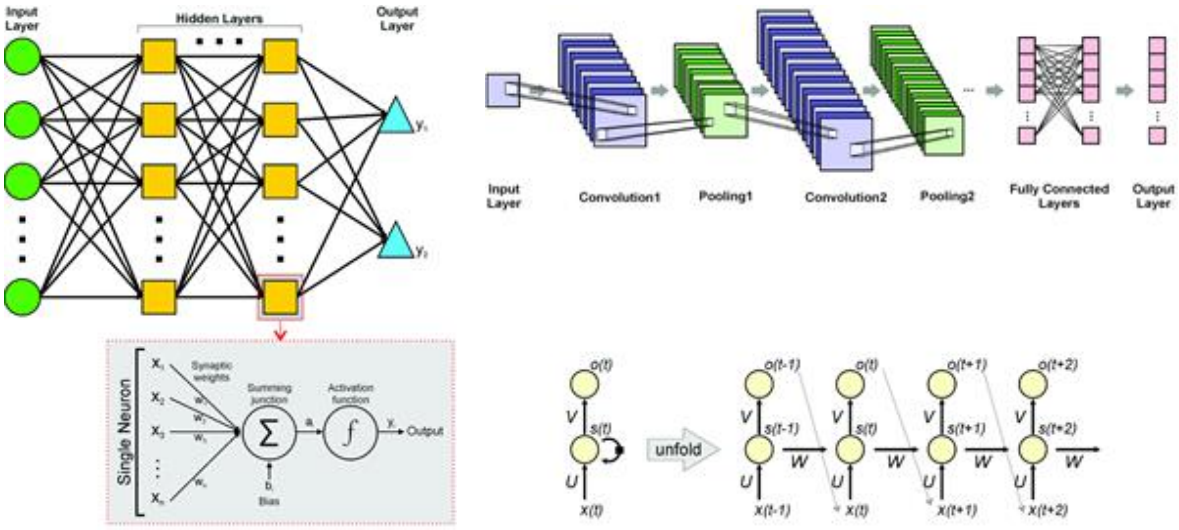
2.1. The first layer: feed forward layer (FFL)

In this layer, feed forward layer FFL consists of a number of simple processing units that resemble nerves, organized in two layers [8]. In this study, two inputs to one output were used. When entering binary number, layer one (Forward Input Layer): It takes the input and then we put it into the input layer without changing anything.

Layer two (yield layer from feed forward): yield layer, consisting of (n) nerves, which processes (n) inputs, pattern (1),…., pattern (n), and multiplied in the matrix weights encoded by these connections encode the network knowing the network and are made up of two nerves and merge with a bias (b), then inserted into the capacity of straight actuation (Purelin) and extract the arrangement (a1), in FFL there are no concealed layers, as is explained in the Figure 1(a).

2.2. The second layer: recurrent layer

In this layer, recurrent layer (RCL), repeating layer neural systems contains three layers [9, 10], the principal layer is called input layer of repeating Layer: In this layer, the yields from the past layer (feed forward) were utilized and a crude contribution with one yield was utilized (prime a2=0). Layer two is known as the shrouded layer of repeating layer: the anterior feeding layer turned into the contribution to the concealed layer in repetitive layer, which consists of n number of nerves [11], which processing (n) inputs, pattern (1),…., pattern (n), and it is multiplied with a weight matrix and consists of three nerves and summation with the bias (b2) and afterward goes into a positive direct enactment work (Poslin). Also, the third layer (the resulting t layer of duplicate layer): the outcome (F) is drawn, as is explained in in the Figure 1(b).



(a) (b)

Figure 1. Planned of humming neural network; (a) Feed forward neural network (b) Repetitive neural network

3. ALGORITHMS FOR NETWORK

a. Set the enactment and arrange loads and predisposition numbers.
   I: Values of information focuses (vector input p).
   J: The length esteems for p.
b. All vector has an input number patterns, row=1, 2… I, col. =1, 2…, J, and then repeat the stage three to four.
c. All nerves, multiplied every entry with its corresponding weight and then combined the bias numbers.
d. Refresh activation of row=1, 2…, I, and col. =1, 2…, J.

$$X_m(0) = F_{hardlim}(X_{(ni,m)})$$

e. Effectively implements the activation function and store the current activations for use in the next repetition for m=1, 2…, j.

$$X_m(old) = X_m(0))$$

f. Select the absolute value of the weight matrix.

$$\text{Weight: } W_{n,m} = \begin{matrix} 1 & for\ n = m \\ -\epsilon & otherwise \end{matrix}$$

g. All t=1,2,3, … rehash stages eight to eleven, while the stop condition test is wrong.

h. All nerves, row=1, 2,3,4,…, I and col=1, 2,3,4…, J, ascertain the system signal, it gets for the subsequent stage.

$$X_{(ni,m)}(new)) = W_{ni,m} * P_n + B_n$$

i. Refresh activation of row=1, 2…, I, and col. =1, 2…, J.

$$X_m(new) = F_{Poslin}(X_{(ni,m)}(new)))$$

j. Store the actuation to use for the following redundancy.

$$X_m(old) = X_m(new))$$

k. Stop condition test. On the off chance that there is more than one non-zero resultant hub yield, continue to stage eight, in any case, as is clarified in Figure 2.

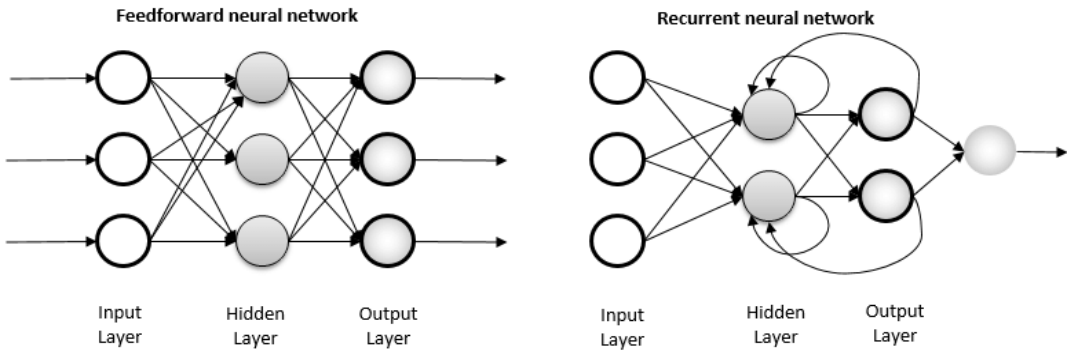


Figure 2. Structure for the network (HNN)

4. SIMULATION AND RESULT

In this study, three methods were utilized for VHDL programming and FPGA device usage as is clarified beneath:

4.1. Method 1: Logical gates technique

Hamming neural network is manufactured by interfacing it utilizing a NAND entryway with the information sources (pattern1, pattern2, weight 11, weight 12, weight 21, weight 22, weight 23, b1, b2) and the yields (F) are utilization, as is explained in the Figure 3.

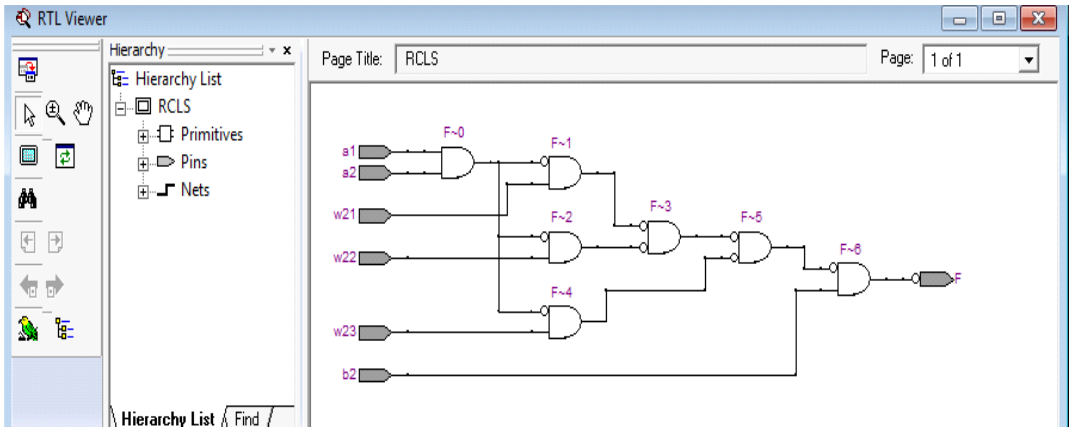


Figure 3. Logic gate technique

4.2. Method2: Encoding software

In this method, programming with code was usage and the spiraling the hamming neural system was stratified.

4.2.1. Feed forward layer (FFL)

In this layer, programming VHDL was utilized by entering values for the inputs (pattern1, pattern2, weight 11, weight 12 and b1) and yield (a1), and the rationale work in geometry was usage to for the extraction the yield as is explained in the Figure 4. A brief of program implementation review and specific forefoot neuron feedback network (FFL) ratios, as is explained in Figure 5.

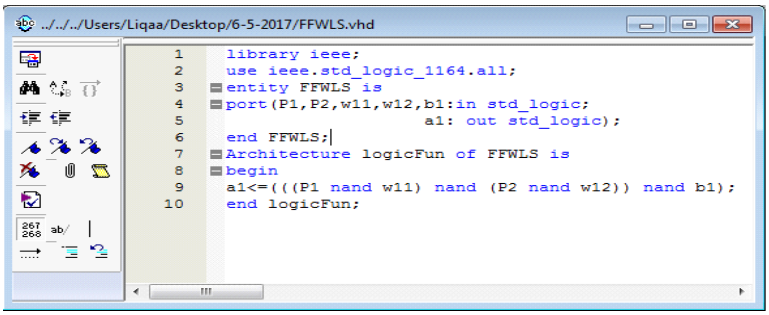


Figure 4. Encoding software of feed forward

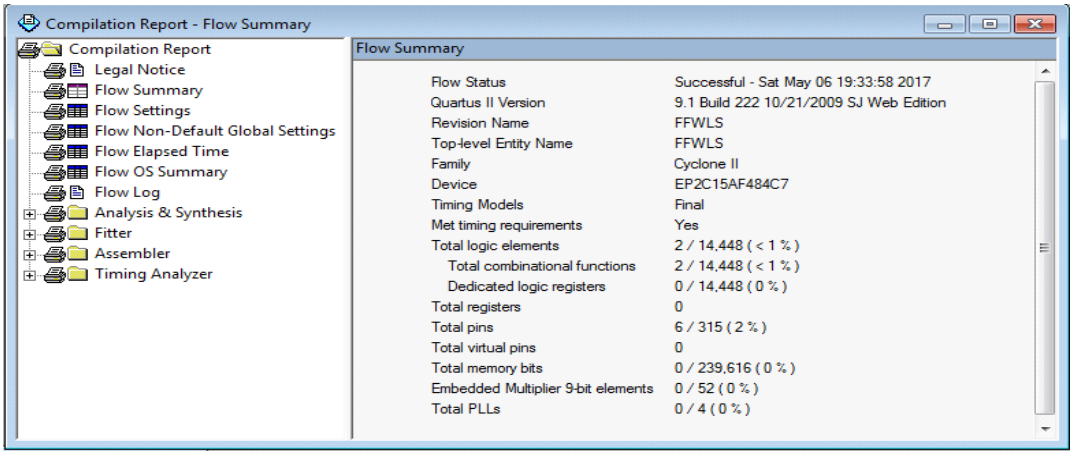


Figure 5. Brief review of feed forward neural network

The following steps are applied to review the feed forward neural network connection (choose program – tools – Netlist watchers – RTL watcher), as is explained in Figure 6. The accompanying advances are applied to show the forward neural network connectivity of the feed yet implementation and this is made by (choose program – tools – Netlist watchers – technology map watcher – post mapping), as is explained in the Figure 7.

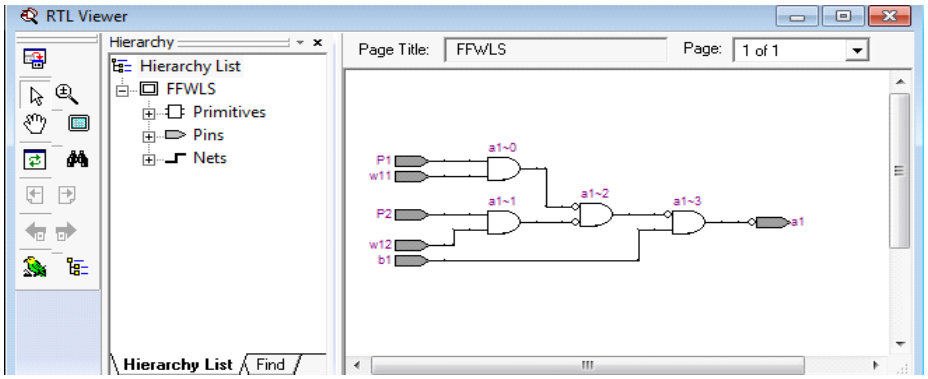


Figure 6. RTL watcher of neuron feedback network layer

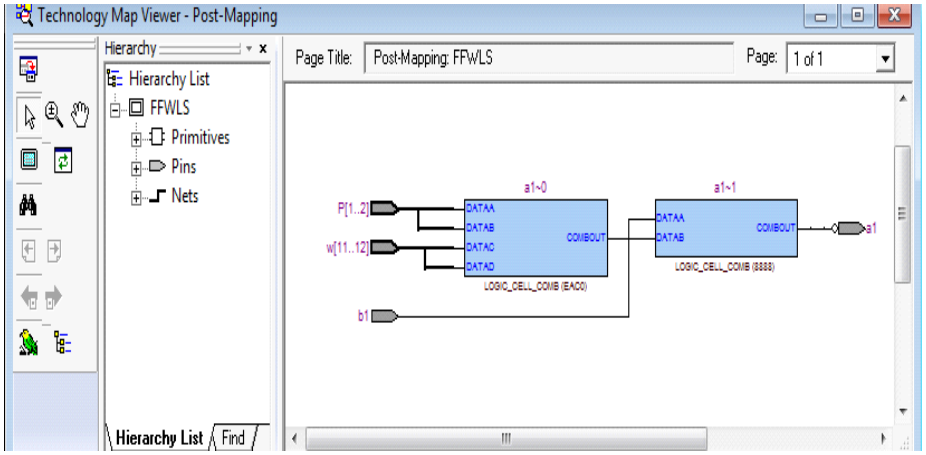


Figure 7. Moment block diagram of feed forward neural network

4.2.2. Repetitive neural network (RNN)

In this layer, programming VHDL was utilized by entering values for the inputs (a1, a2, weight 21, weight 22, weight 23 and b2) and yield (F and the rationale work in calculation was usage to for the extraction the yield as is explained in the Figure 8.

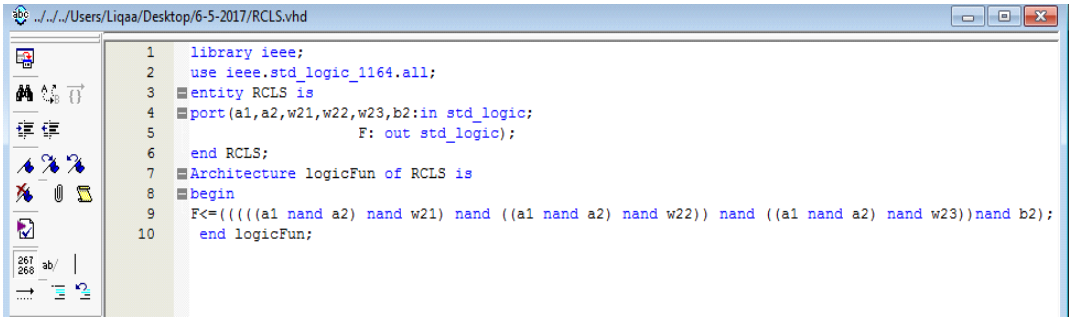


Figure 8. Encoding software of recurrent

Figure 9 shows a brief of program implementation review and specific forefoot neuron recurrent network (RCL) ratios. Figure 10 shows the step which are applied to review the recurrent neural network connection (choose program – tools – Netlist watchers – RTL watcher). Figure 11 shows the accompanying advances are applied to show the repetitive neural system connectivity of the feed yet implementation and this is made by (choose program – tools – Netlist watchers – technology map watcher – post mapping), as is explained in the Figure 11.

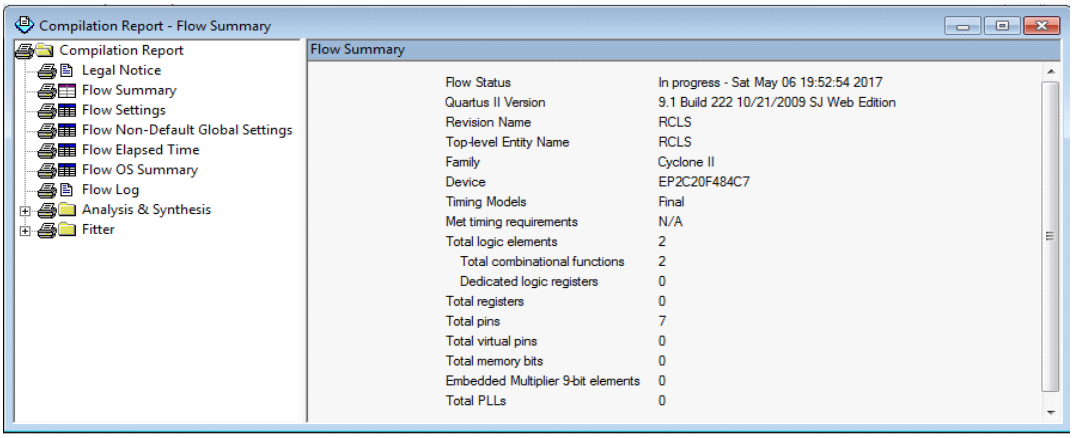


Figure 9. Brief review for recurrent neural network layer

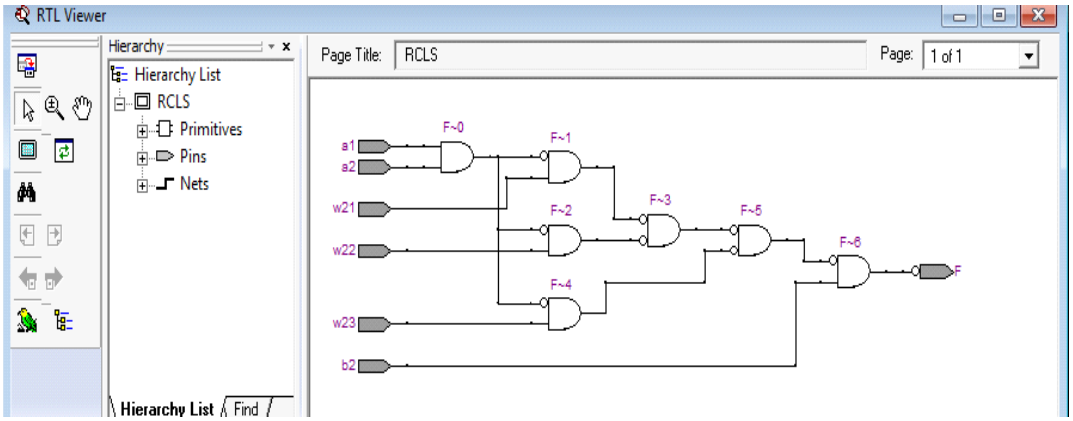


Figure 10. RTL of repetitive neural network layer

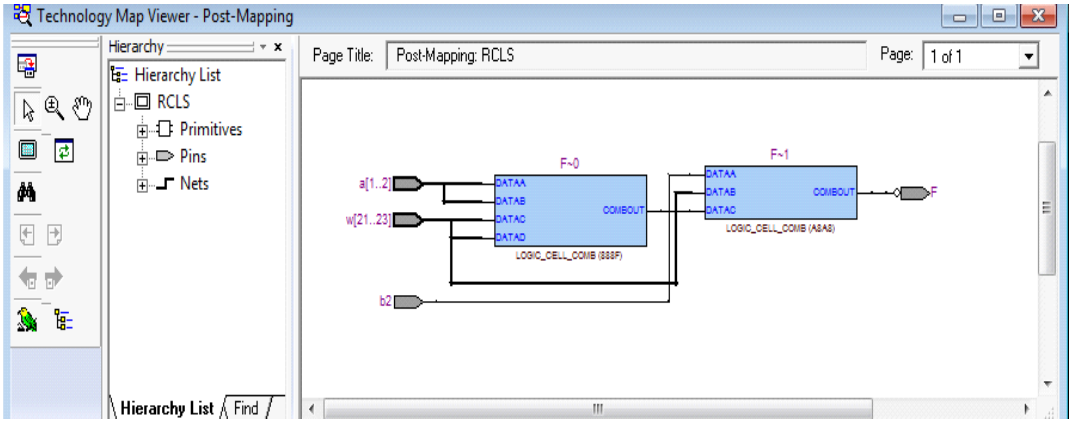


Figure 11. Moment block diagram of repetitive neural network

4.2.3. Hamming neural network

It is the original program of the HNN where the two-layer software (FFL & RCL) is called by the part and afterward blends them and determine the info and yield, use the structure in the engineering and extract of yield, as is explained in the Figure 12. At that point convert from a programmable programming for blocking and applied it to the FPGA device as a tool process and extraction results.

HNNSS.vhd

```
LIBRARY ieee;
USE ieee.std_logic_1164.all;

LIBRARY work;

ENTITY hnns IS
    PORT
    (
        P1 :  IN  STD_LOGIC;
        P2 :  IN  STD_LOGIC;
        W11 :  IN  STD_LOGIC;
        W12 :  IN  STD_LOGIC;
        B1 :  IN  STD_LOGIC;
        W21 :  IN  STD_LOGIC;
        W22 :  IN  STD_LOGIC;
        W23 :  IN  STD_LOGIC;
        b2 :  IN  STD_LOGIC;
        a1 :  OUT  STD_LOGIC;
        F :  OUT  STD_LOGIC
    );
END hnns;

ARCHITECTURE bdf_type OF hnns IS

COMPONENT ffwls
    PORT(P1 : IN STD_LOGIC;
         P2 : IN STD_LOGIC;
         w11 : IN STD_LOGIC;
         w12 : IN STD_LOGIC;
         b1 : IN STD_LOGIC;
         a1 : OUT STD_LOGIC
    );
END COMPONENT;

COMPONENT rcls
    PORT(a1 : IN STD_LOGIC;
         a2 : IN STD_LOGIC;
         w21 : IN STD_LOGIC;
         w22 : IN STD_LOGIC;
         w23 : IN STD_LOGIC;
         b2 : IN STD_LOGIC;
         F : OUT STD_LOGIC
    );
END COMPONENT;

SIGNAL  SYNTHESIZED_WIRE_0 :  STD_LOGIC;
SIGNAL  SYNTHESIZED_WIRE_1 :  STD_LOGIC;


BEGIN
a1 <= SYNTHESIZED_WIRE_0;
F <= SYNTHESIZED_WIRE_1;


b2v_inst : ffwls
PORT MAP(P1 => P1,
         P2 => P2,
         w11 => W11,
         w12 => W12,
         b1 => B1,
         a1 => SYNTHESIZED_WIRE_0);


b2v_inst1 : rcls
PORT MAP(a1 => SYNTHESIZED_WIRE_0,
         a2 => SYNTHESIZED_WIRE_1,
         w21 => W21,
         w22 => W22,
         w23 => W23,
         b2 => b2,
         F => SYNTHESIZED_WIRE_1);


END bdf_type;
```

Figure 12. Program encoding methods for hamming

4.3. Method3: Instant block diagram

In the wake of closure, the software programming process for all layers in isolation, the accompanying advances are applied to change over the program to square Diagram.

4.3.1. The feed forward neural network layer

Choose each software program of feed forward neural network - file - create / update - create symbol file for current file, and open new document - square diagram / schematic file – ok – at that point select a moment obstruct from the undertaking, as is explained in the Figure 13.

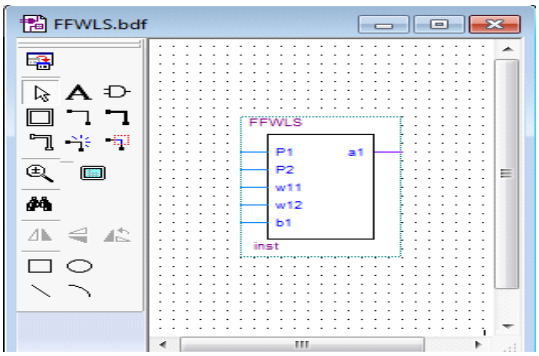


Figure 13. Block diagram of moment of feed forward neural network

4.3.2. The repetitive neural network layer

Choose each software program of recurrent neural network - file - create / update - create symbol file for current file, and open new document - square diagram / schematic file – ok – at that point select a moment obstruct from the venture, as is explained in Figure 14. The two past immediate charts, an incorporated system, consist of the information sources, the weight grid and its final outputs are integrated, as is explained in the Figure 15.

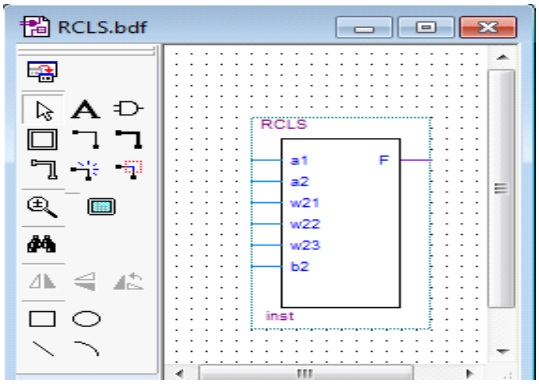


Figure 14. Block diagram of moment for RCL

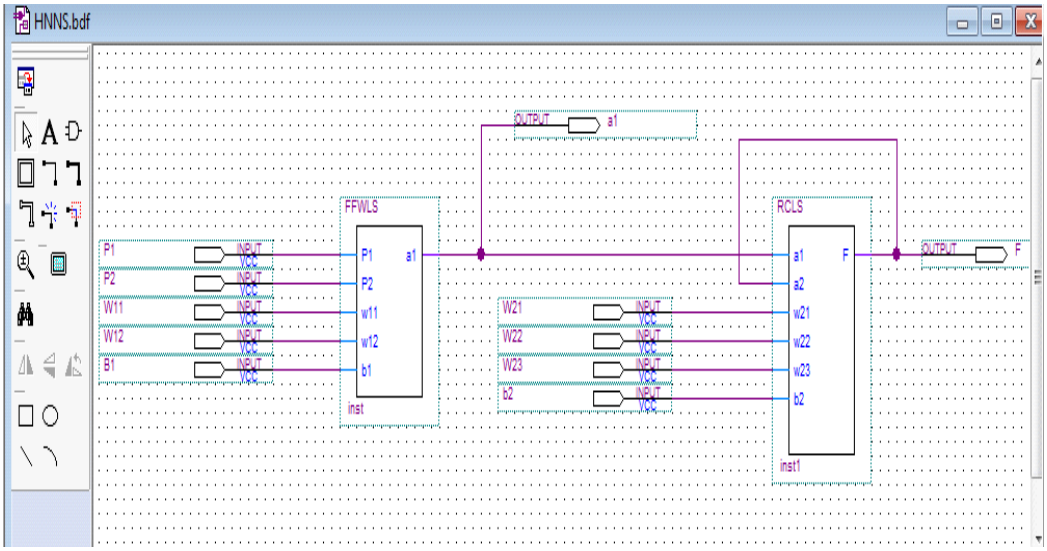


Figure 15. Block diagram of represent moment of humming

In FPGA machine, nine input pins speaking to (two patterns, five weights and two biases) were used as is explained in Table 1 representing the utilization pins.

Table 1. Input pins for hamming

Switch	Input	Represented
SW0	PINS_M22	P1
SW1	PINS_V12	P2
SW2	PINS_M1	W11
SW3	PINS_M2	W12
SW4	PINS_W12	W21
SW5	PINS_U12	W22
SW6	PINS_U11	W23
SW7	PINS_L22	b1
SW8	PINS_L21	b2

Two yields were utilization. The principal result speaks to the extraction of the interface layer from the primary layer and the different speaks to the conclusive outcome of the fundamental system, as is explained in Table 2 representing the pin usage.

Table 2. Output pins for hamming

LED	Output	Represented
LEDR1	PINS_R20	F
LEDR0	PINS_R19	a1

5. THE RESULT OF NETWORK

5.1. The input of hamming neural network

The passage numbers of patterns and weights in two layers, the first layer is feed forward layer and the second layer is repetitive layer, as is explained in Table 3.

Table 3. Input numbers in hamming neural network

Patterns		Weights (FFL)		Weights (RCL)		
P1	P2	W11	W12	W21	W22	W23
0	0	0	0	0	0	0
0	1	0	1	1	0	0
1	0	1	0	0	1	0
1	1	1	1	1	1	0
0	0	0	0	0	0	1
0	1	0	1	1	0	1
1	0	1	0	0	1	1
1	1	1	1	1	1	1

5.2. The output of hamming neural network

The output values for baise, a1 and F in hamming neural network, as is explained in Table 4.

Table 4. Output values in hamming neural network

B1	B2	a1	F
0	0	11111111	11111111
0	1	11111111	00111111
1	0	00010001	11111111
1	1	00010001	00000000

6. CONCLUSION

In this examination, a hamming neural system is implemented. This system contains (2) kinds for networks (FFL and RCL). In the first layer, (2) pattern sections (pattern1, pattern2) and just one yield (a1) were utilization. Where the passage for the two-digit patterns in the weight's matrix (weight11, weight 12) was increased and summation with the baise (b1), at that point it was entered on the pure line functions. In

the second layer, the yield from the preceding layer (feed forward) was utilized and utilized as the primary passage (a1 and primary notes a2) with a single yield (F). The entry was duplicated by the weight matrix (weight12, weight22 and weight23) and summation with the baise (b2) and afterward applied into the POS line work. Three strategies determined (coherent entryway technique, programming program encryption strategy and momentary square chart technique) are applied utilizing VHDL software and associated with an FPGA machine to extraction the yield result.

REFERENCES

[1] Ibrahim M. Nasser, Mohammed O. Al-Shawwa, Samy S. Abu-Naser, "artificial neural network for diagnose autism spectrum disorder," *International Journal of Academic Information Systems Research (IJAISR)*, vol. 3, no. 2, pp. 27-32, Feb 2019.

[2] Binghui Wu, Tingting Duan, "A performance comparison of neural networks in forecasting stock price trend", *International Journal of Computational Intelligence Systems*, vol. 10, pp. 336-346, 2017.

[3] Osama M. Al-Mubayyed, Bassem S. Abu-Nasser, Samy S. Abu-Naser, "predicting overall car performance using artificial neural network," *International Journal of Academic and Applied Research (IJAAR)*, vol. 3, no. 1, pp. 1-5, Jan 2019.

[4] Mohd Nazrul Effendy Mohd Idrus, Kim Seng Chia, "Artificial neural network and partial least square in predicting blood hemoglobin using near-infrared spectrum," *Indonesian Journal of Electrical Engineering and Computer Science (IJEECS)*, vol. 16, no. 2, pp. 701-708, Nov 2019.

[5] Joko Siswantoro, Anton Satria Prabuwono, Azizi Abdullaha, Bahari Idrus, "A linear model based on kalman filter for improving neural network classification performance," *Journal of Expert Systems with Applications*, vol. 49, pp 112–122, Jan 2016.

[6] Dinda P., Takashi M. and Hakaru T., "Implementation of multi-FPGA communication using Pulse-Coupled phase oscillators," *International Conference on Artificial Life and Robotics (ICAROB)*, Seagaia Convention Center, Miyazaki, Japan, pp. 128-131, Jan 2017.

[7] J. Steckert, A. Skoczen, "Design of FPGA-based radiation tolerant quench detectors for LHC," *Journal of Instrumentation*, vol. 12, no. 4, Apr 2017.

[8] Pierre B., Roman V., "The capacity of feed forward neural networks," *Journal of Neural Networks*, vol. 116, pp. 288-311, Aug 2019.

[9] Stefan Horoi, Guillaume Lajoie, Guy Wolf, "Internal representation dynamics and geometry in recurrent neural networks," *Journal of Computer Science, arXiv preprint arXiv: 2001. 03255*, Jan 2020.

[10] Sangeun Kum and Juhan Nam, "Joint detection and classification of singing voice melody using convolutional recurrent neural networks," *Journal of Applied Sciences*, vol. 9, no. 7, pp 1324, Mar 2019.

[11] Siti Aisyah Mohamed, Muhaini Othman, Mohd Hafizul Afifi, "A review on data clustering using spiking neural network (SNN) models," *Indonesian Journal of Electrical Engineering and Computer Science (IJEECS)*, vol. 15, no. 3, pp. 1392-1400, Sep 2019.

BIOGRAPHY OF AUTHOR

Assistant lecturer Liqaa Saadi Mezher, Master Sc. in Computer Engineering, B. Sc. in Computer & Software Engineering, she has 14 years of teaching experience. She published 6 research papers at International level. Worked with Al-Mustansiriyah University, Baghdad, Iraq.