HARDWARE IMPLEMENTATION OF NEURAL NETWORK FOR VEHICLE CLASSIFICATION USING FPGA

[1]**ANIKET U TRIVEDI** & [2]**VM UMALE**

[1&2]M.E. (Digital Electronics), Deptt.of Extc, S S G. M. C.E. Shegaon, India.;
Email: aniket.trivedi1989@gmail.com

Abstract: Vehicle detection and classification is an important and demanding application of WSN. The idea of utilizing small, almost invisible sensor nodes, equipped with a variety of passive and active sensors, for intrusion detection, force protection and border surveillance, though not new, remains fascinating. This project presents vehicle classification using acoustic signal and its FPGA implementation. For feature extraction TESPAR coding used to produce simple and fixed size matrices which are used for classification purpose. Classification is accomplished by using ANN. It is implemented on FPGA.

Keywords: *Vehicle classification, TESPAR, ANN, FPGA.*

1. I INTRODUCTION

Wireless Sensor Networks (WSN) is a technology field that is experiencing great progress. Vehicle detection and classification is an important and demanding application of WSN. The idea of utilizing small, almost invisible sensor nodes, equipped with a variety of passive and active sensors, for intrusion detection, force protection and border surveillance, though not new, remains fascinating. In many cases, sensor nodes detect and classify vehicles from their acoustic and/or seismic signature. In this project we use acoustic and seismic sensors for vehicle classification task.

Instead of using traditional spectral or wavelet techniques to extract a feature vector, representative of each vehicle, we use a time-domain feature extraction method, The encoding method, known as TESPAR, for Time Encoded Signal Processing and Recognition. The advantage of the method is its computational simplicity and low memory requirements. To validate the method, prerecorded acoustic and seismic vehicle signatures are encoded to form a histogram-like matrix of fixed dimensions, using a predefined alphabet. Classification is accomplished using implementation of an Artificial Neural Network (ANN). In this , classification model use multilayer feed forward neural network .The NN has 13input neuron ,6 hidden neuron ,2 output neuron for acoustic and 11input neuron ,6 hidden neuron ,2 output neuron for seismic signals. Network is designed and trained in software using MATLAB Neural Network processing toolbox .Once network is trained ,correct weights are determined, it has to hard coded on FPGA.

The VHDL code is compiled, synthesized and implemented in Quartus II. Version 7.2 software tool .Both software and hardware implementations using data from a real world experiment, which contains acoustic and seismic recordings from two vehicles, a heavy wheeled truck (Dragon Wagon) and a tracked vehicle (Assault Amphibian Vehicle). The dataset is available under the name Sitex02 .Finally it is implemented on Altera Cyclone II (EP2C20F484C7) FPGA.

2. CLASSIFICATION OF VEHICLES

The vehicles can be classified into 2 different categories depending on type of vehicle's weight; heavyweight and lightweight. Based on the threshold values assign to each one of the type vehicles are classified.

Heavyweight vehicle class

Lightweight vehicle class

Figure 1: Categories of Vehicles

2.1 Acoustic Sensors

Acoustic sensors measure vehicle passage, presence, and speed by detecting acoustic energy or audible sound produced by vehicular traffic from a variety of sources within each vehicle and from the interaction of a vehicle s tires with the road. When a vehicle passes through the detection zone, an increase in sound energy is recognized by the signal-processing algorithm and a vehicle presence signal is generated. When the vehicle leaves the detection zone, the sound energy level drops below the detection threshold and the vehicle presence signal is terminated. Acoustic frequencies between 8 and 15 KHz are processed by sensor, which accommodates mounting heights between 20 and 40 ft (6 to 12 m).

Fig 2.1 Acoustic Sensors

2.2 Seismic Sensors

Seismic sensor measures the vibration produced due to passage of vehicle. Seismic signal From ground vehicles are generated from the vibration of vehicle engine and their strike on ground. The frequency response of signals for different vehicle should be different. Seismic waves are less sensitive to different noise. Seismic sensors come under intrusive technology.

Fig 2.2 Seismic Sensor

3. FEATURE EXTRACTION

In this we used a different approach for the feature extraction. Instead of using traditional spectral or wavelet techniques to extract a feature vector, representative of each vehicle, we use a time-domain feature extraction method. The encoding method, known as TESPAR, for Time Encoded Signal Processing and Recognition, TESPAR coding is a good candidate for low-complexity and low-power target detection and classification applications of WSN.

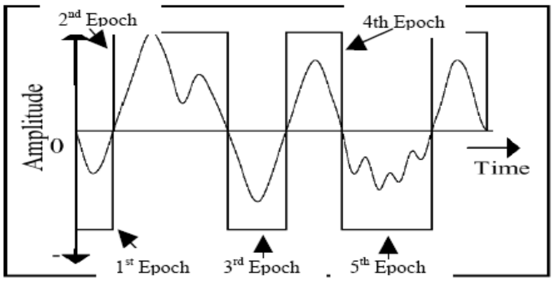


Figure 3 - Infinite Clipping Preserves Only Zero-Crossing Information

TESPAR is based on infinite clipping, proposed by Licklidder and Pollack. The infinite clipping coding is a direct representation of the duration between the zero crossings of the waveform, i.e. the real zeros of the waveform, thus depends on the waveform itself and not on the sampling frequency, as long as sampling is performed according to Shannon's theorem. Hence, a band limited waveform may be simply approximated by segmenting it into successive epochs.

4. TESPAR EXTRACTION AND CODING

As mentioned above, TESPAR coding is based on the real and complex zeros of the epochs in the speech waveform. An Epoch is the segment between two successive zero crossings as illustrated in Figure Duration (D) and Shape (S) parameters are derived from the epoch. The D parameter is derived from the duration between two successive zero crossing. The S parameter corresponds to the shape of the epoch and is represented by the number of positive minimas or negative maximas in the epoch. The selection of positive maximas or negative minimas depends on the polarity of the signal. For an epoch in the positive region, negative minimas are selected whereas for an epoch in the negative region, positive maximas are selected. Figure 3.3 shows the selection of positive minimas or negative maximas. Once the D/S parameters of each epoch are determined, the TESPAR coder pairs them up to generate the TESPAP symbol stream. The symbols are actually elements of the TESPAR alphabet which will be discussed in the next section.

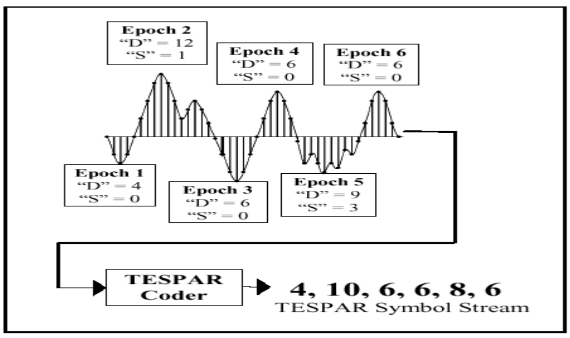


Figure 4: TESPAR Extraction and Coding

5. TESPAR ALPHABETS

A TESPAR codebook comprises of a symbol table of 28 different symbols and is used to map the duration/shape (D/S) parameters of each epoch into a single symbol. For most applications, a standard TESPAR alphabet comprising of 28 different symbols.

	"S" = 0	"S" = 1	"S" = 2	"S" = 3	"S" = 4	"S" = 5
"D" = 1	1					
"D" = 2	2					
"D" = 3	3					
"D" = 4	4	4				
"D" = 5	5	5				
"D" = 6	6	6	6			
"D" = 7	6	6	6			
"D" = 8	7	8	8	8		
"D" = 9	7	8	8	8		
"D" = 10	7	8	8	8	8	
"D" = 11	9	10	10	10	10	
"D" = 12	9	10	10	10	10	10
.	.	.	.	.	.	.
.	.	.	.	.	.	.
.	.	.	.	.	.	.
.	.	.	.	.	.	.
"D" = 37	23	24	25	26	27	28

Figure 5: Standard 29 Symbol TESPAR Alphabet

6. NEURAL NETWORK ARCHITECTURE

In this we used feed forward back propagation network for classification. In order to use ANNs for classification purposes, two phases must be completed. The first phase is the training phase, during which the ANN learns from the imposed input and the requested output. The second phase is the execution phase, during which the ANN is presented with an unknown input and provides an output. The training process is actually an optimization problem, where the mean square error (MSE) of the entire set of training data must be minimized. The algorithm used to solve this optimization problem is the back-propagation algorithm. After propagating an input through the network, the error is calculated and then propagated back through the network. Training is done by continually adjusting the weights so that the output of the ANN matches the output in the training file.

Here we have 13 sample outputs from the TESPAR encoder which is referred from the code book and there are 10 iteration for which the neural network is trained. This data is now converted to 8-bit binary number which is given to neural network by FSM (Finite State Machine).as shown in figure. There are 10 states require to process for 10 iteration. At the first state the system is reseted by keeping rst='1'

Input Vector	AAV(Example)					DW(Example)				
1	1	2	0	1	2	1	2	0	1	2
2	3	4	5	6	5	8	9	7	1	1
3	8	9	1	8	1	8	9	7	1	1
4	4	5	6	7	8	1	2	3	4	5
5	0	1	2	3	4	0	1	2	2	3
6	1	2	3	4	5	0	1	1	2	3
7	5	6	7	8	9	1	2	3	4	5
8	3	3	4	5	6	4	5	6	7	8
9	0	1	2	3	2	3	4	5	6	7
10	0	1	1	2	2	4	5	6	7	8
11	0	1	1	2	2	7	8	9	8	1
12	0	1	1	2	3	6	7	8	9	9
13	1	0	1	3	2	5	6	7	8	8

Table 1 .Training Data

Acoustic ANN has M neurons for the input layer, plus one bias neuron, one hidden layer with K neurons one bias neuron and 2 neurons for each-vehicle class (AAV: output1/DW: output2). M is the number of symbols in the alphabet . In this custom alphabets are used, so M=13,K=M/2=6 with log sigmoid transfer function.

Seismic ANN has also M neurons for the input layer, plus one bias neuron, one hidden layer with K neurons one bias neuron and 2 neurons for each-vehicle class (AAV: output1/DW: output2). M is the number of symbols in the alphabet . In this custom alphabets are used, so M=11,K=M/2=6.

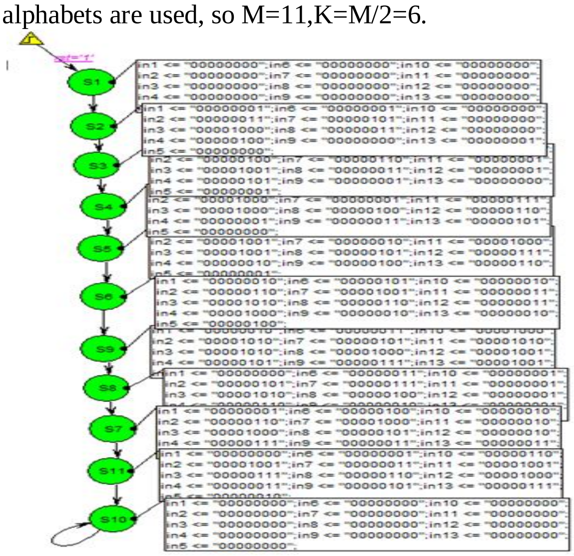


Figure 6: Finite State Machine (FSM) diagram for 13 input training samples

7. TRAINING

During the training phase, the network must "learn" to distinguish between different vehicles based on their feature vector (S-Matrix). For this purpose a fraction of the data- set is used for training purposes and the rest is used to test the performance of the system. The selection of the training set can be random or manual, by individually choosing the events that provide the more representative feature vector of each class. In a real implementation of a WSN for vehicle detection and classification, users would not have the luxury of training it on the best available data. Training the system on data gathered during an individual run of a vehicle, passing from a number of nodes, which clearly detect the event, is a more realistic approach. To simulate this situation the system is trained on data recorded from sensor nodes 1, 53, 56, 60 and 61 of the selected run, i.e. AAV4 and DW4 and tested with data from all other available nodes and runs (i.e. AAV5, AAV6 and DW5, DW6). The selection of the particular subset of sensor nodes is based on their position. A node situated in the middle of the crossroad has higher possibility to record the event with higher accuracy than a far-placed node. It is interesting to examine the performance of the system when trained with different runs, recorded under different conditions. It is apparent from performance characteristic.

Simulation result:

From simulation it is cleared that, FPGA implementation of ANN clearly classifying Vehicles in two categories. Out 1 represent AAV and out 2 represent DW. Vehicle Recognition rate is good.

RTL VIEW

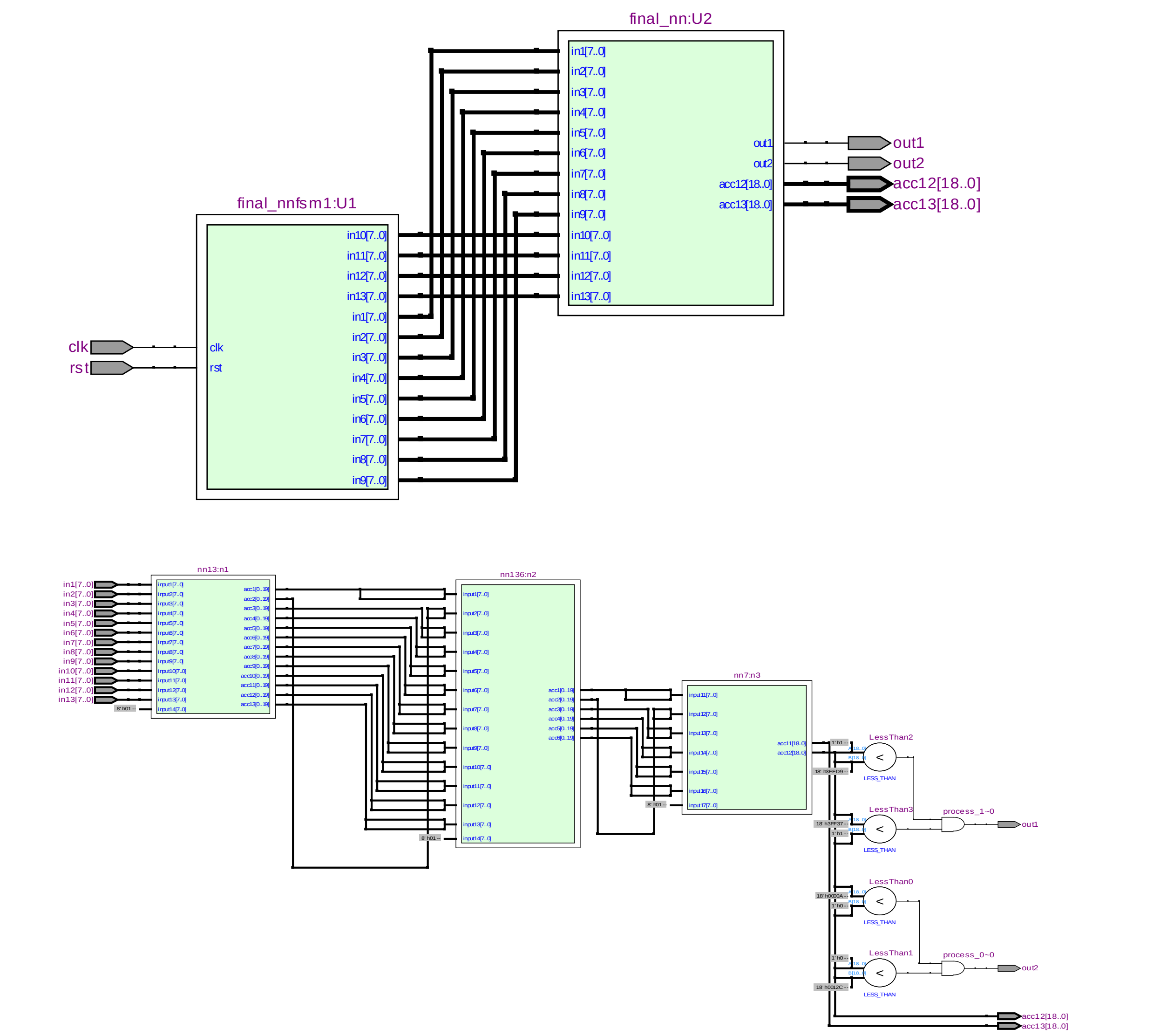


Fig 7. Synthesis (RTL Viewer)

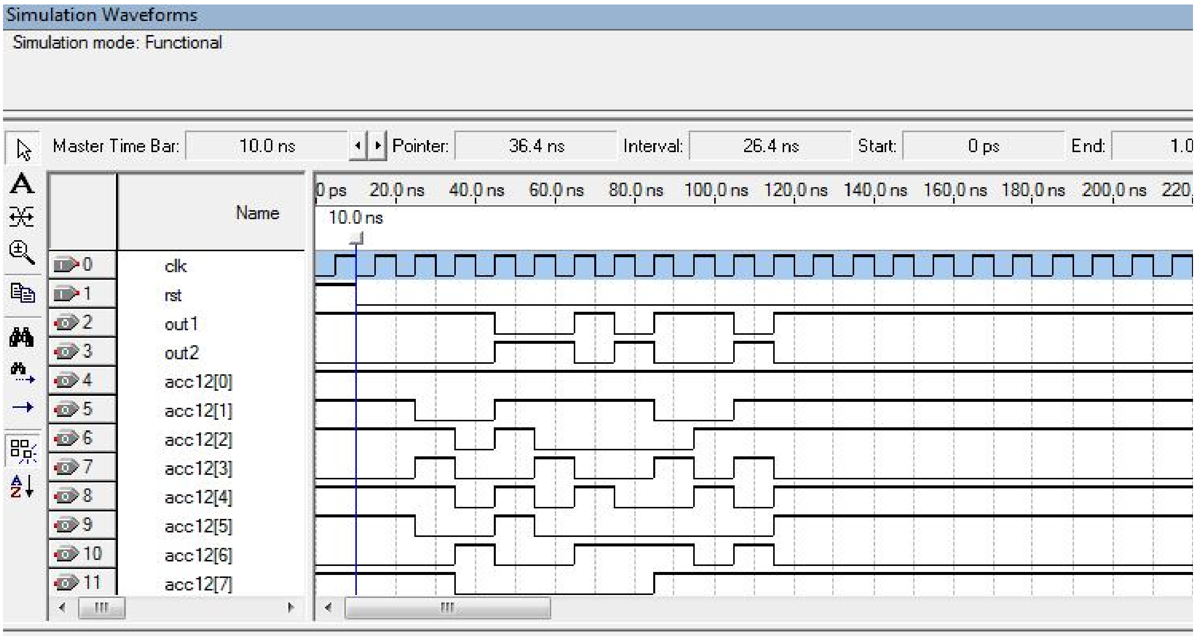
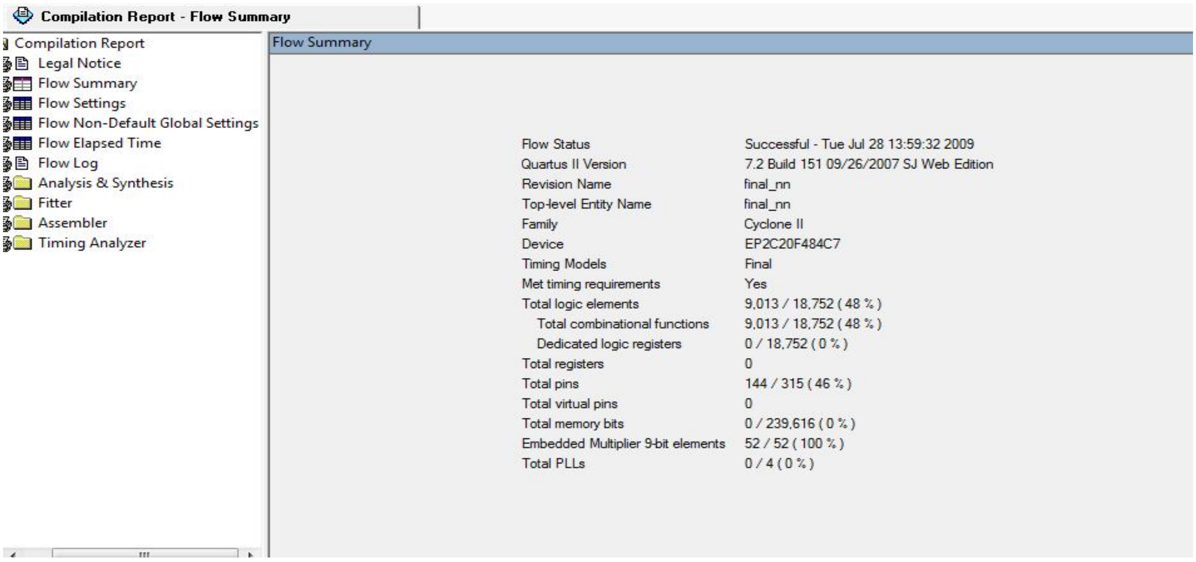


Fig 8. Simulation Result



8. CONCLUSION

This paper has presented the implementation of neural networks by FPGAs. The TESPAR coding has a main role in sensing the spaces in the duration and hence classify. The proposed network architecture is modular and is specially built for a military application and it is being possible to easily increase or decrease the number of neurons as well as layers because neural networks are inherently parallel structures, parallel architectures always result faster than serial ones.

REFERENCES

[1]. Rafid Ahmed Khalil, “Hardware Implementation of Backpropagation Neural Networks on Field programmable Gate Array (FPGA)”, Al-Rafidain Engineering, Vol.16, No.3, Aug.2008.

[2]. Aydoğan Savran, Serkan Ünsal,” Hardware implementation of a feedforward neural network using FPGA”.

[3]. S. Oniga, A. Tisan, D. Mic, A. Buchman and A. Vida-Ratiu,” Optimizing FPGA Implementation of Feed-Forward Neural Networks” Electronic and Computer Engineering Department, North University, Baia Mare, Romania.

[4]. R. A. King and T. C. Phipps, “Shannon, TESPAR and Approximation Strategies” HYDRALOGICA Ltd 2003. 4th December 2003.

◈◈◈