MICRONTROLLER BASED NEURAL NETWORK REALIZATION AND IRIS PLANT CLASSIFIER APPLICATION

Mutlu Avcı, Tülay Yıldırım
Yildiz Technical University Electronics and Communication
Engineering Dept. 34349 Besiktas Istanbul
mavci@yildiz.edu.tr, tulay@yildiz.edu.tr

Abstract : General neural network realizations are in integrated circuit forms. Therefore they are expensive to produce for many non commercial applications. Instead of designing a chip, using a cheap, standard microcontroller for neural network realization will be an optimum solution for many neural network applications. In this work, a Multi Layer Perceptron (MLP) neural network was realized by programming the ATMEL 89C51 microcontroller. Training of the network was performed in MATLAB 6.1 environment and weight and bias values were taken from the program. Microcontroller programmed was tested using both training and test data and the results were concluded.

Keywords: Neural Networks Hardware, Multi Layer Perceptron, Iris Classification

1. INTRODUCTION

Neural networks hardware implementations are generally classified as neurocomputers or neurochips. These can be application specific or general purposed very large scale integrated (VLSI) circuit technology realizations [1]. Of course producing an integrated circuit is more expensive for small amounts. Instead of producing integrated circuits usage of a standard microcontroller is more economical way. Field programmable gate arrays (FPGA) are also used for neural network realizations [1,2,3]. They are also standart chips and cheap applications [2,4]. Microcontroller based realizations are not so popular since mathematical operators in standard microcontrollers are limited and many of them do not have floating point arithmetical unit. The sigmoid based network topologies are hard to build on these chips. FPGAs with floating point unit are seem to be more suitable for neural network synthesis. Although software based realizations in programmable standard chips are more flexible and cheaper than the integrated circuit implementations, they are not as fast as the hardware solutions. With respect to application type, both have different importance. The limitations in the number of input-output pins and in memory exist in standard chips. Although these disadvantages, for many applications, microcontroller based neural network realizations are perfectly suitable and optimum solutions.

In this work, using ATMEL 89C51 standard microcontroller MLP neural network was realized by programming. In MATLAB 6.1 environment, MLP network was trained with iris plant data set and test data were applied to check the network performance. After programming the microcontroller with the same weight and bias values, train and test data were checked and the realization results were concluded by giving the simulation results.

2. THE IRIS PLANT CLASSIFIER

2.1. Iris Plant Data Set

One of the most popular and best known database of the neural network application is the iris plant data set. Created by R. A. Fisher in 1950 [5,6]. There are 150 instances with 50 in each one of three classes. Existing 4 numerical attributes for identification of the classes are sepal length, sepal width, petal length and petal width in centi meters. The classes are iris setosa, iris versicolor, iris virgica [7].

In this application 40 data of each class are used for training and 10 are seperated for test purpose.

2.2. Multi Layer Perceptron Neural Network Structure

Multilayer Perceptron (MLP) is the most common neural network model, consisting of successive linear transformations followed by processing with non-linear activation functions. MLP represents a generalisation of the single layer perceptron, which is only capable to construct linear decision boundaries and simple logic functions. However, by cascading perceptrons in layers complex decision boundaries and arbitrary Boolean expressions can be implemented. MLP is also capable to implement non-linear transformations for function approximations. [8], [9], [10].

The network consists of a set of sensory units (source nodes) that constitute the input layer, one or more hidden layers of computation nodes, and an output layer. Each layer computes the activation function of a weighted sum of the layer's inputs. The input signal propagates through the network in a forward direction, on a layer-by-layer basis. The learning algorithm for multilayer perceptrons can be expressed using generalised Delta Rule and gradient descent since they have non-linear activation functions. [8,11].

In general form of an MLP network, the x_i inputs are fed into the first layer of $x_{h,1}$ hidden units. The input units are simply 'fan-out' units: no processing takes place in these units. The activation of a hidden unit (neuron j) is a function f_j of the weighted inputs plus a bias, as given in equation (1).

$$x_{pj} = f_j\left(\sum_i w_{ji}\, x_{pi} + \theta_j\right) = f_j\left(y_{pj}\right) \qquad (1)$$

Where w_{ji} is the weight of input i to neuron j, x_{pi} is input i, that is, output i from the previous layer, for input pattern p and θ_j is the threshold value. The output of the hidden units is distributed over the next layer of $x_{h,2}$ hidden units until the last layer of hidden units, of which the outputs are fed into a layer of x_o output units [8,9,10,11].

In MATLAB 6.1 environment, an MLP net with three hidden layers and one output layer was designed as shown in Fig. 1. The activation functions for the hidden and ouput layers were pure linear because of the difficulty of sigmoid function realization on microcontroller. All hidden layer had three neurons like the output layer.

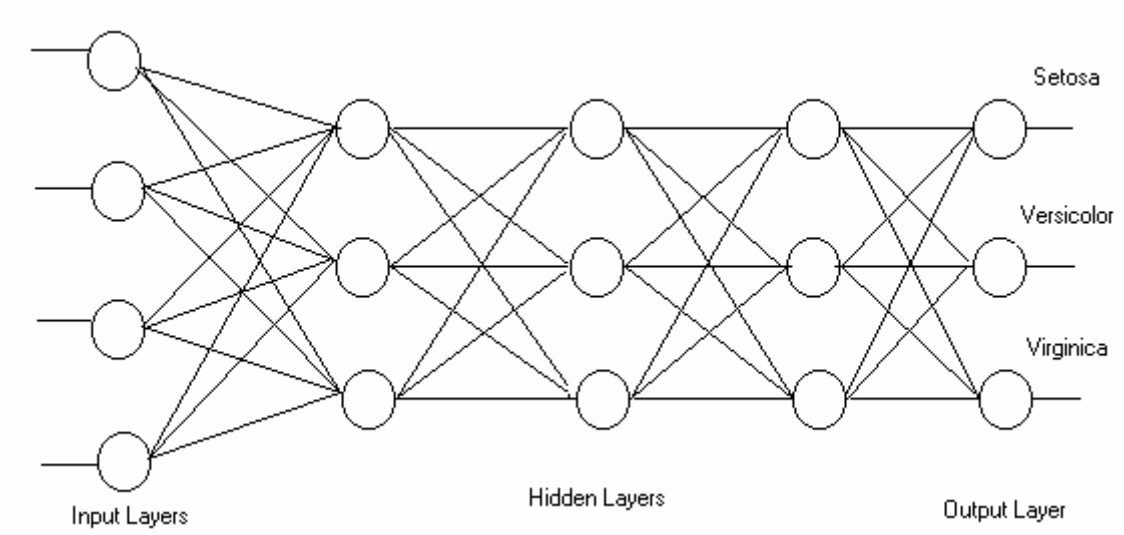

Fig. 1. The MLP Neural Network Topology

3. THE PROPOSED HARDWARE

Block diagram of the neural network hardware evaluation board in Fig. 2 was designed for general purpose neural network realizations. Since the ATMEL 89C51 microcontroller has only digital inputs, an analog to digital (ADC) converter ADC0808 with 8 channels was connected to increase the inputs. The microcontroller in the block diagram controls the input selection with 3 pins. The results were serially sent to a PC to check the performance of the classification.

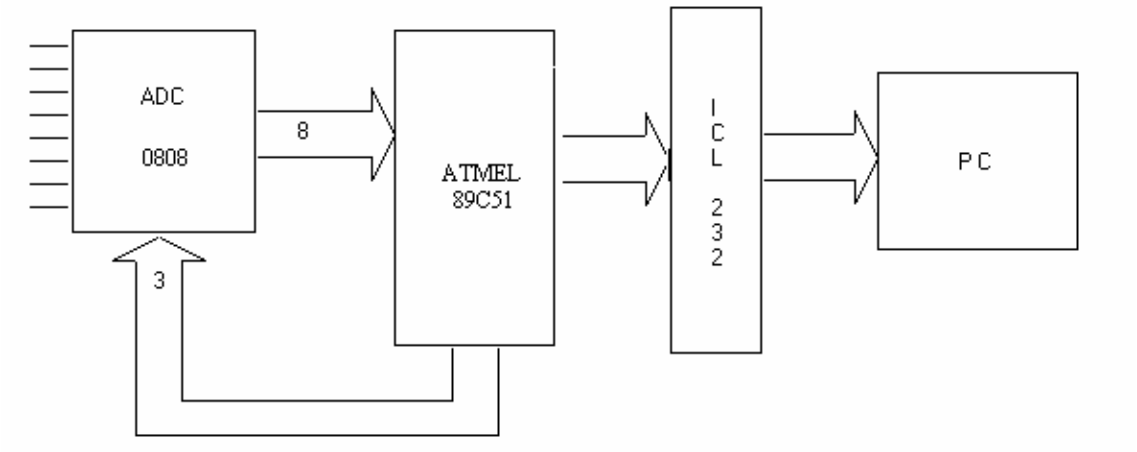

Fig. 2. Block diagram of the neural network implementation circuit

4. THE IMPLEMENTATION RESULTS

The hardware implementation results of training and test data were shown in Table 1 and Table 2, respectively.

Table 1. Training data classification

Iris Plant	True	False	Total	Performance
Setosa	40	0	40	100%
Versicolor	39	1	40	97.5%
Virginica	40	0	40	100%

Table 2. Test data classification

Iris Plant	True	False	Total	Performance
Setosa	10	0	10	100%
Versicolor	10	0	10	100%
Virginica	10	0	10	100%

5. CONCLUSION

The application performance proved that microcontroller based realization of MLP can find many application areas. Although sigmoid function is not used, by increasing hidden layer size, microcontroller based neural network implementations can be effectively used. The main disadvantage of this kind of application is the reduction at the classification speed because of the increased hidden layers. Through this application a neural network is realized with the cheapest hardware and effectively operated. This does not mean microcontroller based realizations generally works. Many classification tasks may not be possible without sigmoid activation function.

References

1.I. Aybay, S. Çetinkaya, and U. Halıcı, Classification of Neural Network Hardware, Neural Network World, IDG Co., Vol. 6, No 1, 1996, pp 11-29

2.V. Beiu, How to Build VLSI-Efficient Neural Chips, Proceedings of the International ICSC Symposium on Engineering of Intelligent Systems, EIS'98, Teneferie, Spain, February 1998, pp. 9-13.

3. A. W. Fisher, R. J. Fujimoto, and R. C. A. Smithson, A Programmable Analog Neural Network Processor, IEEE Transactions on Neural Networks, Vol. 2, No. 2, March 1991, pp. 222-229.

4. P. D. Moerland, and E. Fiesler, Hardware-Friendly Learning Algorithms for Neural Networks: An Owerview, Proceedings of 5th International Conference on Microelectronics on Neural Networks and Fuzzy Systems: MicroNeuro'96, Lausanne, Switzerland, February 1996.

5. R. A. Fisher, The use of multiple measurements in taxonomic problems, Annual Eugenics, 7, Part II, 179-188 (1936); also in "Contributions to Mathematical Statistics" (John Wiley, NY, 1950).

6.B. V. Dasarathy, Nosing Around the Neighborhood: A New System Structure and Classification Rule for Recognition in Partially Exposed Environments. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. PAMI-2, No. 1, 67-71, 1980.

7. http://ftp.ics.uci.edu/pub/ml-repos/machine-learning-databases/

8.M. Avcı, T. Yıldırım, Classification of Escherichia Coli Bacteria by Artificial Neural Networks, IEEE Conferance on Intelligent Systems,

9.R..J. Schalkoff, Artificial Neural Networks, McGraw-Hill Inc., Singapore, 1997.

10.J..M. Zurada, Introduction to Artificial Neural Systems, PWS Publishing, Boston, 1995.

11.S. Haykin, Neural Networks: A Comprehensive Foundation, Macmillan College Publishing, New York 1994.