70 INPUT, 20 NANOSECOND PATTERN CLASSIFIER

P. Masa, K. Hoen, H. Wallinga
MESA Research Institute, University of Twente,
P. O. Box 217, 7500 AE Enschede,
The Netherlands, E-mail: masa@ice.el.utwente.nl
Phone: 31 53 892753, Fax: 31 53 341903

Abstract - **A CMOS neural network integrated circuit is discussed, which was designed for very high speed applications. This full-custom, mixed analog-digital chip implements a fully connected feedforward neural network with 70 inputs, 6 hidden layer neurons and one output neuron. The neurons perform inner product operation and have sigmoid-like activation function. The 70 network inputs and the neural signal processing are analog, the synaptic weights are digitally programmable with 5 bit (4 bits + sign) precision. The synaptic weights are stored on on-chip static RAM cells. The combination of analog and digital techniques results unique computing power with ease of use. Programming can easily be performed with the help of a spreadsheet or other suitable interface program from a PC. The resolution of the input signals is mainly determined by the signal to noise ratio which lies typically between 8-12 bits. Therefore the equivalent input bandwidth can be as high as 28-42 Gbits/second. The system is designed for very high speed vector classification and the feasibility of a single chip neural network photon trigger for nuclear research is shown. Because of the fully parallel architecture and the fast analog signal processing the network achieves unique computing performance and classifies up to 70 dimensional vectors within 20 nanoseconds, performing 20 billion ($2*10^{10}$) multiply-and-add operations per second. The circuit occupies 10x9mm^2 silicon area with 1.5μm CMOS process and dissipates only 1W at 5V supply.**

I. INTRODUCTION

Although neural networks (NNs) compute exceptionally parallel manner, this valuable characteristic has not been exploited as successfully as their learning capability. In case of fully parallel hardware, the processing time is independent of the amount of data to be processed by the network. Furthermore only a few computing steps have to be performed in serial manner, therefore computation time can be extremely short. This work concentrates on the benefits of unique parallel processing. One of the most challenging tasks of hardware realisation of neural nets is the inner product operation. Since it consumes too large chip area with digital circuitry, fully parallel digital architectures do not exist for large NN-s. If high precision is not required, the compact and high speed analog approach has great advantage. With analog technique low cost, low power dissipation, single chip architectures of complex neural networks are possible. Although such systems are commercially available [4], [5], offering as low as several microsecond processing time for as large as 128 dimensional input vector, it is almost impossible to find any solution for application domain demanding tens of nanoseconds processing delay for similarly large input vectors. The integrated circuit presented here is intended to provide the high computing performance needed for such applications.

II. NETWORK ARCHITECTURE

The implemented NN architecture is shown on figure 1. It is a fully interconnected feedforward structure with 70 analog inputs, 6 hidden layer neurons and one output neuron. The neurons are inner product type, and have sigmoid-like activation function. The neural signal processing is fully analog, yielding high speed operation and compact circuitry for inner product operation. The synaptic weights are stored digitally on static RAM (SRAM) cells, to enable simple programming even from a personal computer. Digital weight storage also helps to eliminate weight decay and increases reproducibility. The SRAM cells are located nearby each synapse circuit to minimise wiring for communication. Downloading the approximately 3.5 Kbit synaptic weight and configuring information is relatively slow compared to normal operating speed of the NN circuit, and takes a few milliseconds. The chip block diagram is shown on figure 2. The largest area is occupied by the 70x6 synapse array, including the 70x6 differential voltage to current converters as synapses and 70x6x5 SRAM cell for weight storage. Each 5 bit synaptic weight can be selected, red and written by the row-, column decoders and read, write circuitry. A programmable voltage source array is located nearby the synapse array which enables programmable biasing and control of the gain of neural activation functions.

0-7803-1901-X/94 $4.00 ©1994 IEEE

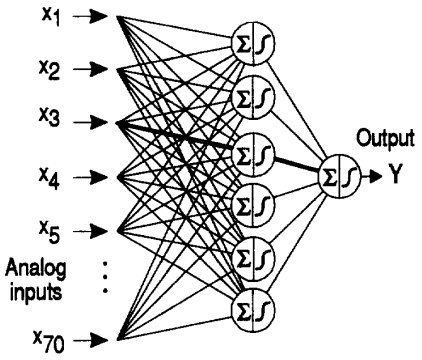

Figure 1. Implemented NN architecture

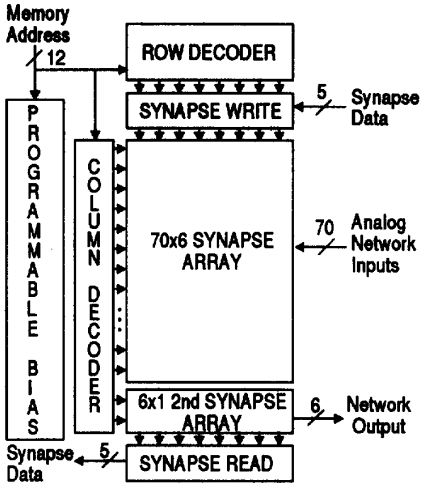

Figure 2. System block diagram

III. CIRCUIT DESCRIPTION

Figure 3 shows the analog circuitry along the signed signal path of figure 1. The processing delay of the NN pattern classifier is merely the delay introduced by this circuitry, since the rest of signal paths are parallel. The synapse circuit is a differential pair formed by T1 and T2, with a single ended voltage input and a reference voltage, which is equal for all the synapses in the NN. The outputs of synapses are differential currents, which are summed on the (differential) summing node of the corresponding neuron. Variable synaptic weight is achieved by programmable current source for the differential pair. The current source transistors T6, T7... T8 are properly sized to deliver current with respect to the smallest, or unity current, according to ascending powers of 2. Any combination of these currents can be obtained by using the switch transistors T3,T4... T5. The sign of the synapse can be varied by interchanging Vin and Vref, using an 8 transistor switch, which is not shown in figure 3. Synapse characteristic, obtained by PSPICE simulation is shown in figure 4. The sum of synaptic currents is transformed to voltage by the load transistors T9 and T10. T11 controls the differential load. The saturating, sigmoid-like activation function, shown in figure 5, is obtained by the saturating characteristic of the second layer synapse, rather than by a separate non-linear circuit.

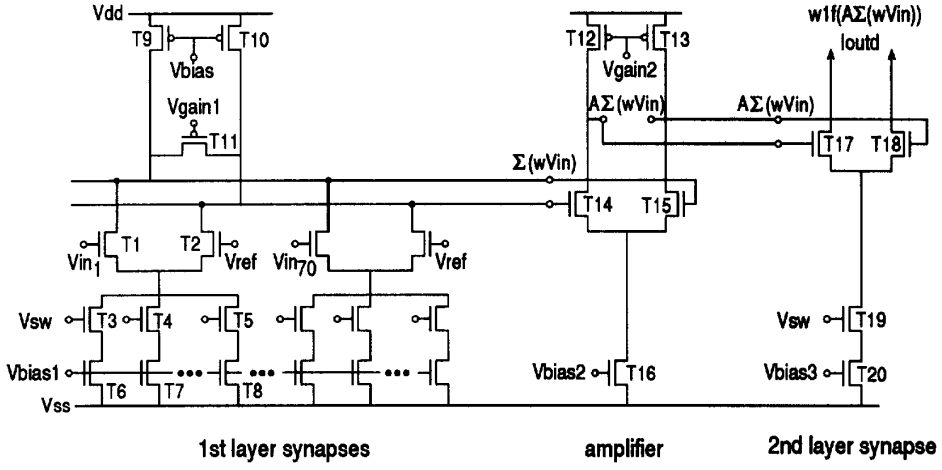

Figure 3. Circuitry along the signed signal path of figure 1

This method simplifies the circuitry and increases speed. Note however, that this simplification is valid only for NNs with one neuron in the second layer. The second layer synaptic weight is obtained by the number of parallel connected active synapse stages, formed by T17, T18, T19, T20. This stage is activated by the switch transistor T19. Every switch transistor of the circuit is wired to a separate SRAM cell. There are altogether 3750 SRAM cells on chip.

The current summing node has intrinsically large parasitic capacitance, since all the synapse outputs and the common load are connected to this node. To increase the speed of the circuit, the node impedance has to be kept low. The consequence of low node impedance is a small voltage swing on the summing node. A voltage amplifier stage scales this voltage properly for the second layer synapse stage.

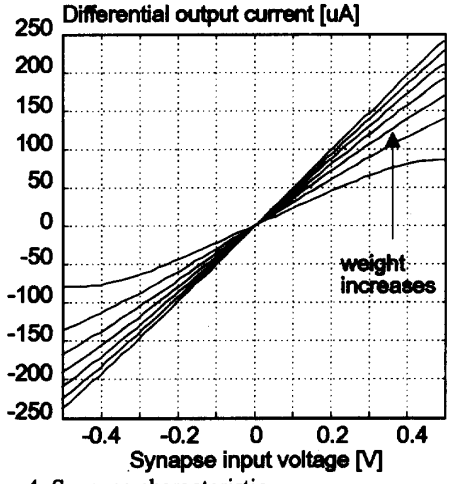

Figure 4. Synapse characteristic

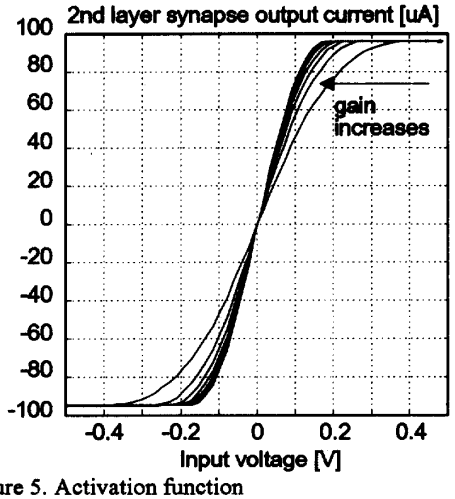

Figure 5. Activation function

IV. NETWORK PERFORMANCE

CASE STUDY: *"A Neural Network Photon Trigger"*

The feasibility of a single chip NN photon trigger for the LHC[1] experiment in CERN[2] have been studied. We used a database containing "TGT calorimeter preshower" information[3], generated by photons and pions. The NN photon trigger should recognise data which was generated by photons. Real-time data processing allows less than 25 nanosecond time period for the decision making. Within this time period a 32 dimensional analog input vector has to be evaluated. A similar high-energy physics application is described for an experiment at DESY[4], in [1], [3], with higher dimensional analog, time discrete input vectors.

We trained a BackProp. network with the "labelled" database. The total set with more than 7000 samples, was divided into training and test set. The learning curve is shown in figure 6. The small difference between curves of training- and test set indicate good generalisation. The percentage of incorrectly classified patterns is smaller than 4%, even for the test set. Synaptic weights obtained by the training procedure, can be downloaded to the NN chip.

Examining the decision making process of feedforward neural nets for pattern classification, reveals why and how this type of computation tolerates the non-ideal effects of analog hardware. Here only quantitative results are presented for the discussed application, one may refer to [1], [2], [3] for more detailed discussion. Figure 7 shows the effect of discretization. As it is expected, classification error decreases with increasing number of discrete levels. Surprisingly, the performance changes only slightly above 20 discrete levels for synaptic weights. This is due to the large distance between the pattern classes in the input space. The applied, simple discretization procedure results "lucky" and "unlucky" cases, therefore the curve on figure 7 is not monotonous. Careful discretization improves performance and would result smoother curve.

---

[1]LHC: *Large Hadron Collider at CERN*

[2]CERN: *Conseil européen pour la recherce nucléraire*

[3]*"TGT calorimeter preshower" information, developed on the basis of CERN RD33 project.*

[4]DESY: *Deutsches Elektronen Synchrotron (Hamburg)*

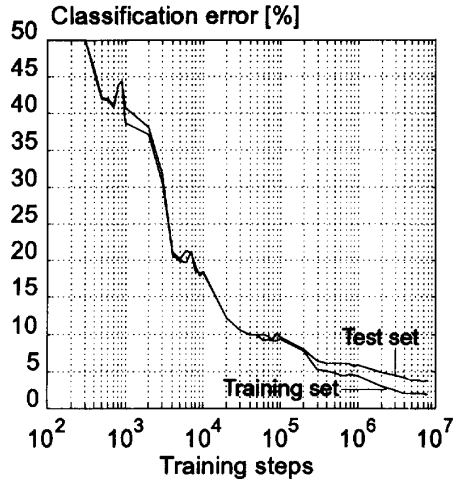

Figure 6. Learning curve

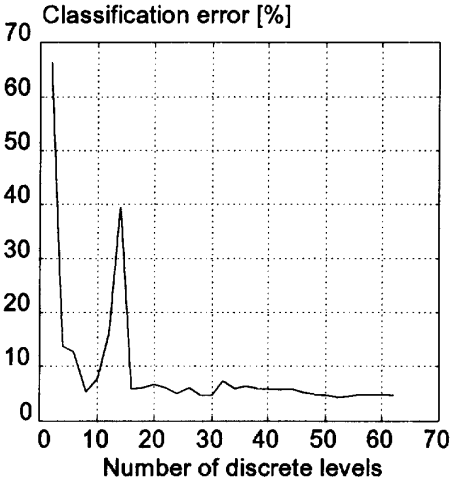

Figure 7. Error due to discretization (test set)

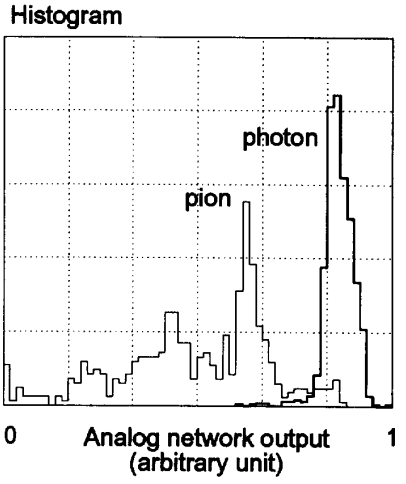

Figure 8. Histogram of classifier responses (test set, optimal)

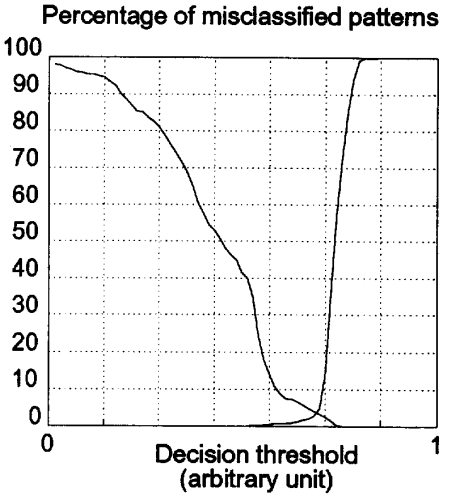

Figure 9. Classification error (test set, optimal)

Figure 8 shows, the histogram of analog classifier outputs over the test set. We can see, that although there is an overlap between the two classes, photon and pion data is clearly separated. Above or under a certain network output the data is classified "photon" or "pion" respectively. We call this value of network output "decision threshold". Figure 9 shows the classification efficiency as a function of the decision threshold. For example if we choose the decision threshold, where the percentage of misclassified photon patterns equals the percentage of misclassified pion patterns, the correctly classified data is 96% for both classes. When decreasing the decision threshold to correctly classify 99% of photon data, 15% of pion data is misclassified.

Figure 8 and figure 9 show performance in case of ideal hardware. Simulations have been made to examine the non-ideal effects, introduced by our analog NN hardware. Noise, synapse non-linearity, weight discretization and the effect of sigmoid-like shape of the activation function have been taken into account. Figure 10 and figure 11 show the results. The overlap between the two classes increases compared to the ideal case. In contrast to the 96% correctly classified data at the crossing of curves in figure 9, we get 93% with our hardware. The 75% increase of misclassified patterns is mainly due to the applied simple discretization technique. We expect even better performance with careful weight discretization.

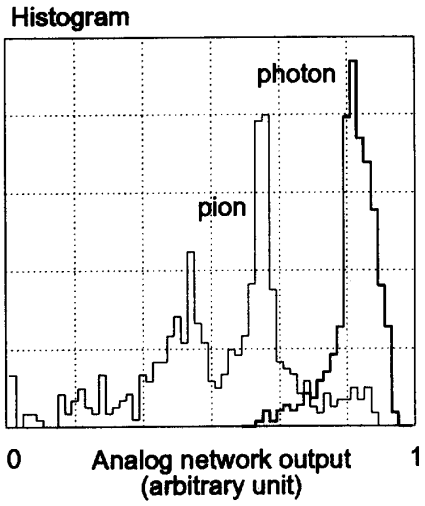

Figure 10. Histogram of classifier responses (test set, hardware)

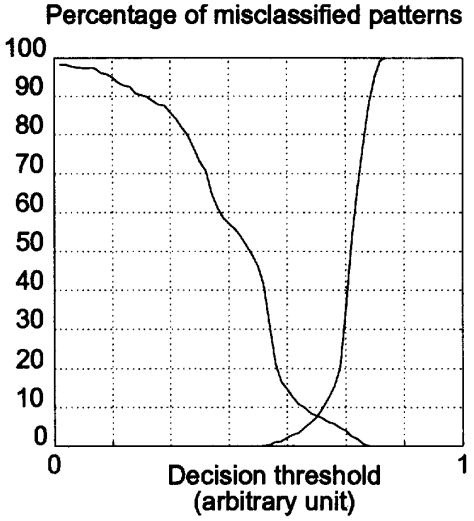

Figure 11. Classification error (test set, hardware)

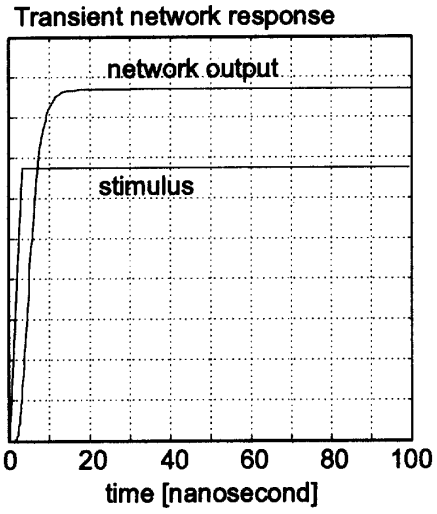

Figure 12. Transient response

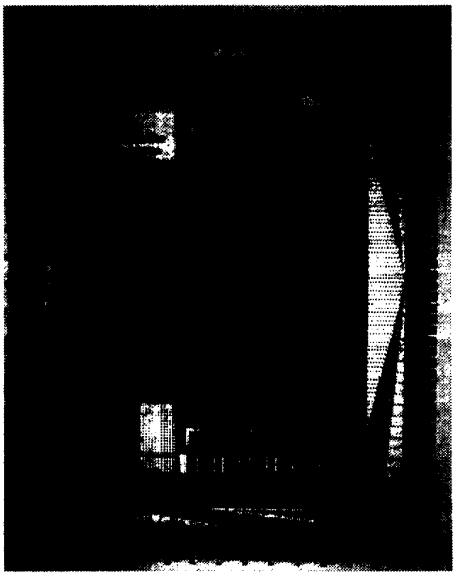

Figure 13. Chip Layout

PSPICE simulations, based on elaborate layout extraction, including parasitic effects and parameter spread, verify the expected 20 ns processing delay for the entire NN circuit. We can see the result of transient analysis on figure 12. Both the stimulus and the NN hardware response are shown on the figure. Table 1. shows more details about the chip specifications. Equivalent input bandwidth was calculated by assuming 12 bit resolution (signal to noise ratio) and 50 MHz input rate for the time discrete analog input signals. Dividing the number of synapses on chip by the processing delay we get the computing performance in terms of multiplications and additions per second. The chip layout is shown in figure 13.

Table 1. Specifications of the programmable chip under fabrication

Network architecture:	70x6x1 feedforward
Equivalent input bandwidth	4 GBytes/second
Chip size	10mmx9mm (1.5μm DLM CMOS)
Number/resolution of synapses	426, 5 bits (4 bits + sign)
Synapse size	400x70μm^2
No. of transistors:	40 000
PGA package:	144 pins
Total processing delay:	< 20 nanoseconds
Computation speed:	20 billion multiplications and additions per second
On-chip static RAM	3750 bits
Power dissipation:	1W

V. CONCLUSION

A digitally programmable, analog neural network processor is presented. Although the chip does not take advantage of a state-of-the-art technology, it provides unique computing performance, due to the architecture and analog processing. Considering the attractive points of analog approach such as high speed, compact inner product operation, we conclude, that analog hardware is attractive for the implementation of high speed neural networks. The feasibility of a single chip neural network photon trigger for nuclear research have been confirmed. The circuit clearly demonstrates the strong attributes of analog VLSI neural networks. The single chip pattern classifier performs 20 billion ($2*10^{10}$) multiplications per second, with merely 1W power dissipation and has 4 GBytes per second input bandwidth. With this performance classification up to 70 dimensional vectors within tens of nanoseconds becomes possible.

Acknowledgements
This work in the program of the Foundation for Fundamental Research on Matter (FOM) have been supported by the Netherlands Technology Foundation (STW)

REFERENCES

[1] P. Masa, K. Hoen, H. Wallinga, "High-Speed Analog Neural Network Processor", Submitted to *IEEE Micro, Special Issue on Analogue VLSI and Neural Networks*, June 1994

[2] P. Masa, K. Hoen, H. Wallinga, "20 Million Patterns Per Second Analog CMOS Neural Network Pattern Classifier", *Proc. European Conf. on Circuit Theory and Design*, Davos, Switzerland, 1993

[3] P. Masa at. al., "20 Million Patterns Per Second VLSI Neural Network Pattern Classifier" *Proc. International Conference on Artificial Neural Networks*, Amsterdam, 1993

[4] M. Holler et al., "An Electrically Trainable Artificial Neural Network," *Proc. Int. Joint Conf. Neural Networks* , 1989

[5] Hernan A. at al., "Implementation and Performance of an Analog Nonvolatile Neural Network," *Analog Integrated Circuits and Signal Processing* 4, 97-113 (1993)