NEUCOM 201

Design and performance of a prototype analog neural computer

Paul Mueller, Jan van der Spiegel, Vincent Agami, David Blackman, Peter Chance, Christopher Donham, Ralph Etienne, Jason Flinn, Jinsoo Kim, Mike Massa and Supun Samarasekera

Depts. of Elect. Engrg. and Biophysics, University of Pennsylvania & Corticon Inc., Philadelphia, USA

Received 21 November 1991
Revised 21 August 1992

Abstract

Mueller, P., J. van der Spiegel, V. Agami, D. Blackman, P. Chance, C. Donham, R. Etienne, J. Flinn, J. Kim, M. Massa and S. Samarasekera, Design and performance of a prototype analog neural computer, Neurocomputing 4 (1992) 311–324.

A prototype programmable analog neural computer and selected applications are described. The machine is assembled from over 100 custom VLSI modules containing neurons, synapses, routing switches and programmable synaptic time constants. Connection symmetry and modular construction allow expansion to arbitrary size. The network runs in real time analog mode, however connection architecture as well as neuron and synapse parameters are controlled by a digital host that monitors also the network performance through an A/D interface. Programming and monitoring software has been developed and several application examples including the dynamic decomposition of acoustical patterns are described. The machine is intended for real time, real world computations including ATR. In current configuration its maximal speed is equivalent to that of a digital machine capable of more than 10^{11} flops. A much larger machine is currently under development.

Keywords. Analog neural computer; optical character recognition; phoneme recognition; real time computation.

1. System overview

We have assembled and tested a programmable Analog Neural Computer designed for real time computation of neural network problems. A photograph of the machine is seen in *Fig. 1*. The system is a prototype intended to validate the design concept, a larger machine for commercial applications is being developed.

The computer contains directly interconnected arrays of *electronic neurons*, *modifiable synapses*, *modifiable synaptic time constants* and *analog routing switches*. The arrays are

Correspondence to: Paul Mueller, Depts. of Elect. Engrg. and Biophysics, University of Pennsylvania & Corticon Inc., Philadelphia, USA.

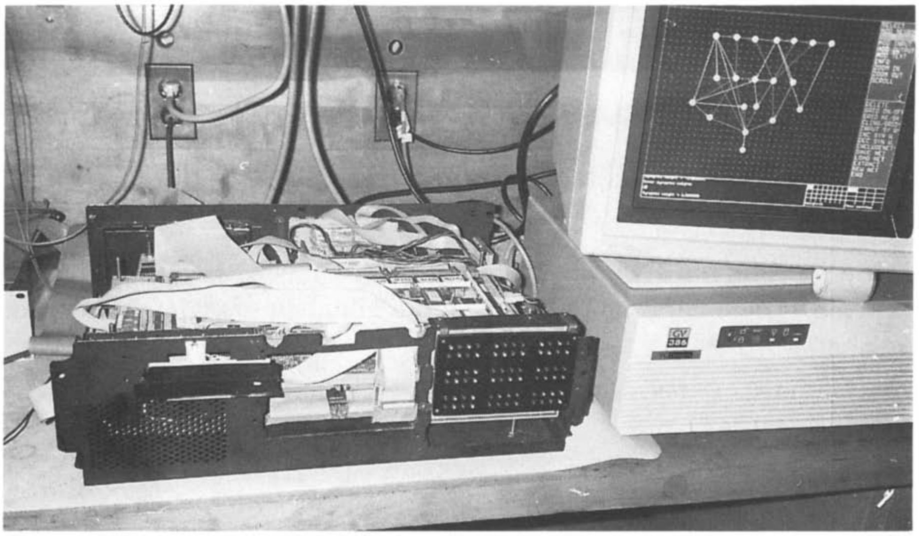

Fig. 1. Photograph of the neural computer, showing the main circuit boards, I/O buffer, power supplies and LED display panel. The host computer is at right.

fabricated on VLSI chips and packaged in planar chips carriers that form separate modules and are mounted on interconnected circuit boards. Neuron arrays are arranged in rows and columns and surrounded by synapse and routing switch arrays. The switches select the connections between neurons. *Figure 2* illustrates the machine architecture.

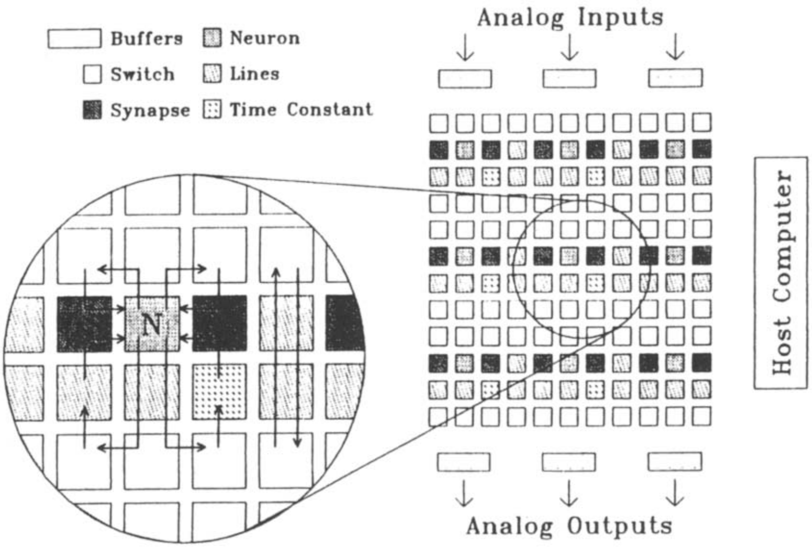

Fig. 2. General architecture and data flow. Neuron modules receive inputs from synapse modules east and west and feed outputs into switch modules north and south. Due to the connection symmetry the network can be expanded to arbitrary size.

The computer runs in analog mode. However, interconnections, synaptic gains, time constants and neuron parameters are set by a digital host computer through an interface board either from the keyboard or from stored programs. For the display of neuron activity and the implementation of learning algorithms, selected time segments of the outputs from all neurons are multiplexed on each neuron chip, digitized and stored in host memory. The multiplexing operation is independent of and does not interfere with the analog computations of the network.

Adjustments of synapse parameters and connection architectures are computed by the learning algorithms on the basis of the stored neuron activity and loaded serially into the neural computer via a digital interface board.

The machine has several unique attributes. First, both the gains and the time constants of synaptic transfer are separately programmable, a feature that is indispensable for analog computation of time domain phenomena such as visual motion, target tracking and analysis of acoustical patterns. Second, it is constructed from interchangeable modules with two-dimensional symmetrical pinout that permits easy modification of the numerical ratios and positions of neurons, synapses and routing switches. Third, the network is expandable to arbitrary size. Finally, the coupling of analog and digital hardware adds the flexibility of digital methods.

In its current configuration the machine is composed of 3 circuit boards each containing 35 directly interconnected VLSI modules (*Fig. 3*). Each module contains a separate array of neurons, modifiable synapses, modifiable synaptic time constants or analog routing switches. The neurons have an adjustable threshold and minimum output at threshold, synapse gains are programmable from 0 to ±15 with logarithmic 6 bit control and the time constants from 5 ms to 2s with 4 bit resolution.

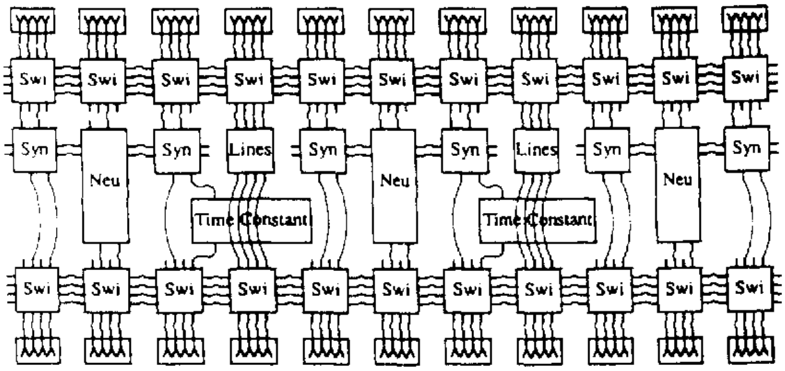

Fig. 3. Block diagram of the processor boards. The major components and 1/4 of the analog data paths are shown. Support chips, digital data buses and power connections are omitted. I/O buffers and digital interface are on separate boards.

2. Description of the modules

The four different VLSI modules containing arrays of neurons, synapses, time constants or routing switches were designed and fabricated in 2μ CMOS. For this prototype the synapse chip contained 8 × 16 synapses, the switch chip 16 × 16 analog crosspoint switches and the neuron chip 8 neurons. The relatively low component count was dictated mainly by economic considerations. For details of the VLSI components see [1] and [2].

2.1 The neuron module

The module contains 8 neurons, an analog multiplexer and digital control logic for addressing the chip and driving the multiplexer. The neuron input is a current provided by the synapse and the output a voltage (0–4V). The neuron transfer function is shown in *Fig. 4.*

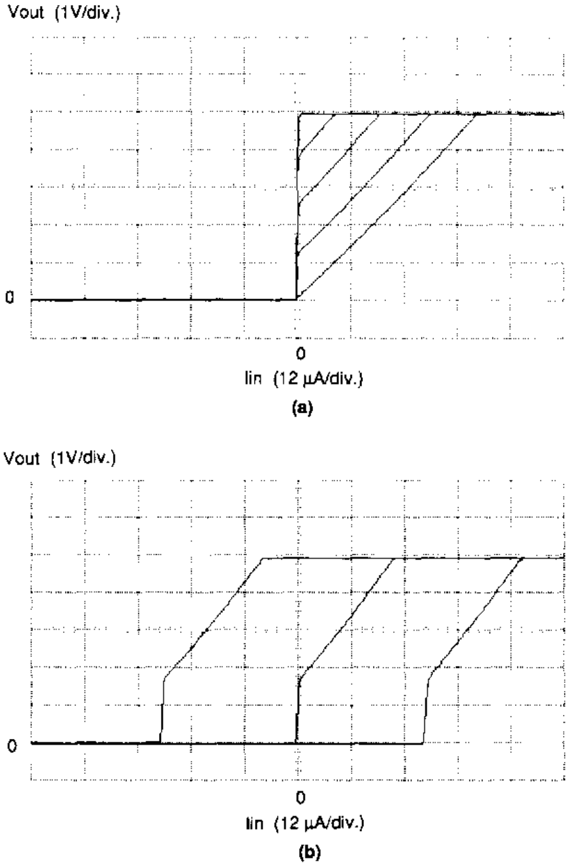

Fig. 4. The measured neuron output voltage versus input current: (a) transfer characteristics for different minimum outputs at threshold ranging from 0 to 5 V: (b) characteristics for different threshold bias currents: −30, 0, and +30 μA with a 1.7 V step at threshold.

The reasons for choosing this particular transfer function which permits variable degrees of arithmetic and logic operation by the same unit have been described elsewhere [1]. In order to avoid possible crosstalk in the switch lines the neuron bandwidth was designed for 150 KHz and provisions were made to further reduce the bandwidth at the system level. However, no crosstalk problems were encountered and in future designs the bandwidth shall be increased to 1 MHz. The analog multiplexer which permits sampling of the neuron activity by the host computer is of conventional design and does not interfere with the neuron operation. The sampling rate of the multiplexer is under software control with a maximal rate of 150 KHz. Thus the activity of all neurons can be sampled at up to 0.5 ms per point.

2.2 The synapse module

The synapse converts the neuron output voltage into a current through a V/I converter and scales the current mirrors into 5 values. A current recombination unit controlled by a shift register memory combines the currents to the appropriate value specified by the user programmable weight. In order to achieve a large dynamic range for the weights (4 orders of magnitude) a floating point scheme was chosen for the current scaling and recombination. The block diagram of the synapse module and some measurements of synapse performance are shown in *Figs.* 5 and 6.

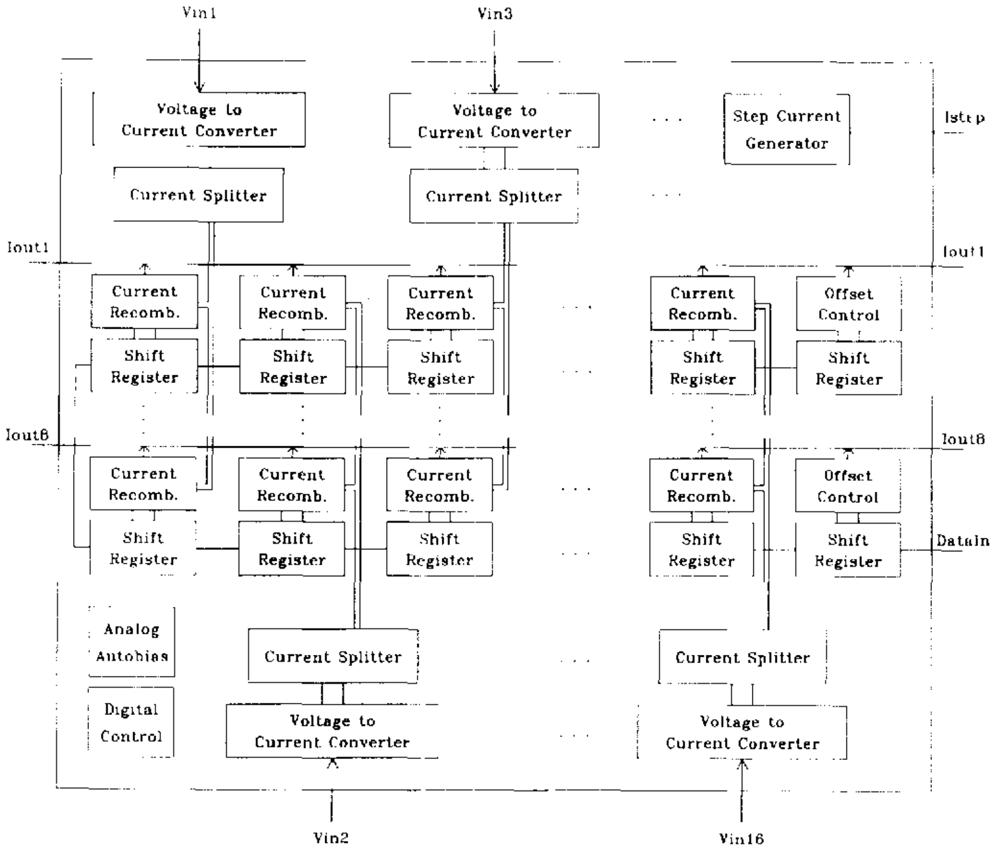

Fig. 5. Block diagram of the synapse module; the inputs are common to all synapses in the same column and the output lines sum the currents of all synapses in the same row.

2.3 The time constant module

For time domain operations, programmable time constants over the range from 5 ms to

1 s are desirable. These time constants were implemented fully on chip with no external capacitors by using an operational transconductance amplifier as a high resistance. The time constant values are set by a 4 bit local memory. The values vary logarithmically with digital code as seen in *Fig. 7.*

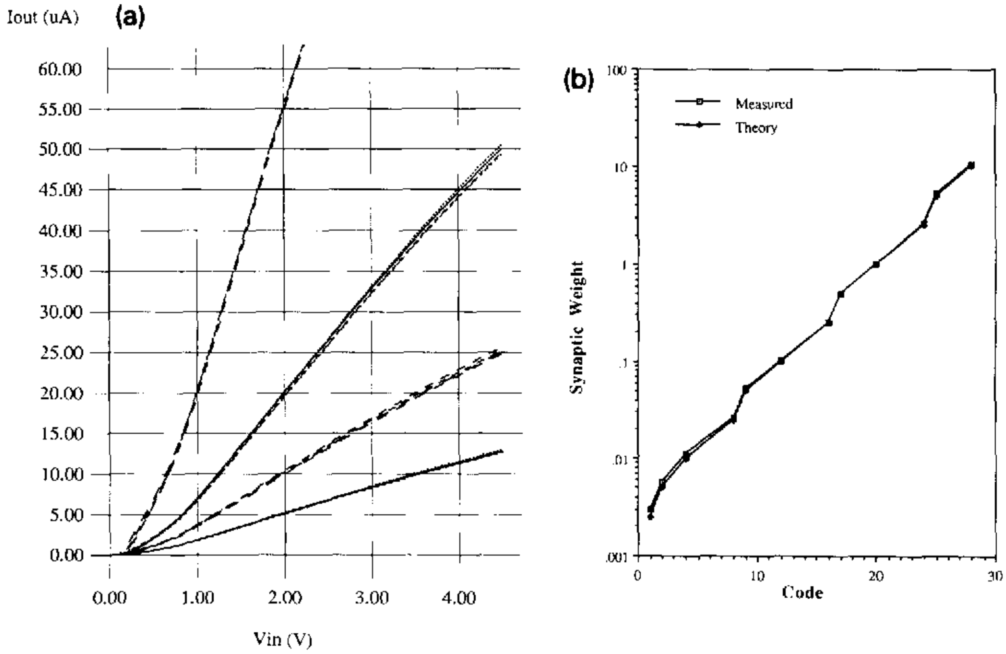

Fig. 6. *Left*: Output current versus input voltage of a synapse, measured from 4 chips from the same run for weights of 2.5, 1, 0.5 and 0.25. *Right*: Measured and theoretical weight factors of the synapse versus the digital code programmed in synapse memory. The measured values are normalized by the measured weight value for a gain of 1.

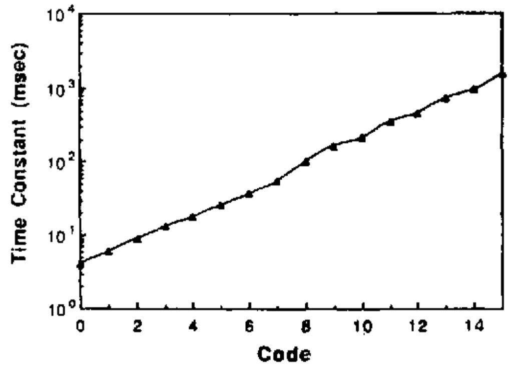

Fig. 7. The average time constant values versus the digital code measured from 16 circuits on 4 different chips. The standard deviations vary from 2.5 to 7.5%.

2.4 The switch module

This module is used to route the analog signals between the neurons and the synapses and thus specify the network architecture. The module consists of an array of crosspoint switches with additional cut switches at the vertical and horizontal lines. Each crosspoint switch is set by a local one bit memory, the cut swtiches also allow grounding either output or input and have a two bit memory. On-resistance is 2 to 3 KΩ and On-resistance > 1 TΩ. The block diagram of the module is shown in *Fig. 8*.

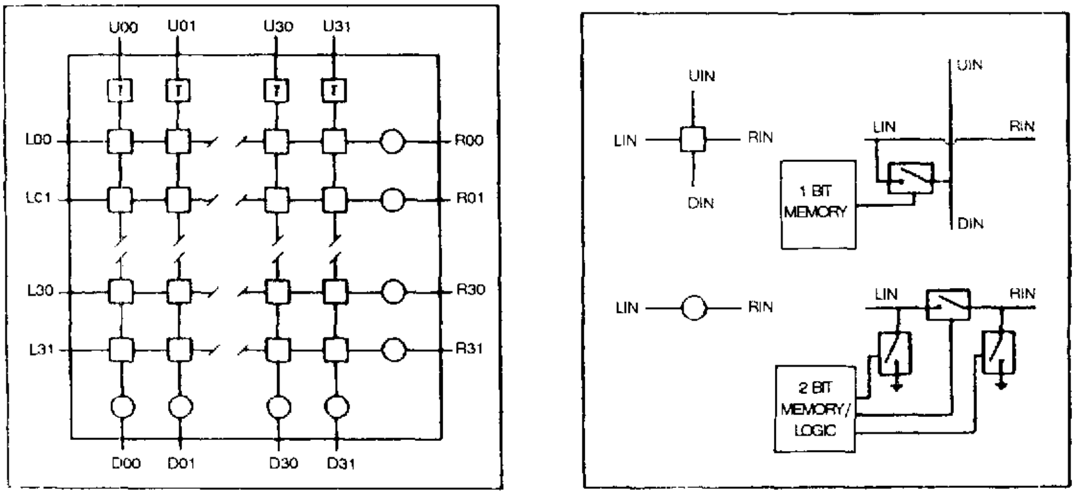

Fig. 8. Block diagram of the switching fabric.

3. System operation

The machine is controlled by software residing in the digital host. An overview of the different tools is shown in *Fig. 9*.

The connection architecture, synapse and neuron parameters are set and verified serially at 2 MHZ from a digital host through a special interface board and graphic control software. At the lowest level of control the network parameters are set manually through a mouse and graphic display that highlights the connections and displays the synapse parameters. Parameters are loaded either for the entire network, a selected board or a selected module.

Higher level control software, allowing the automatic transfer of conceptual networks into the machine has also been developed. In this case the neurons in the conceptual network are first partitioned and grouped according to the degree to which their inputs originate from common sources. The groups are then assigned to specific modules in the physical network by a placement routine and subsequently interconnected by an autorouting program based on min-cut procedures. This program generates settings of switches and component parameters that are loaded into the neural computer.

The network is connected directly to the outside world through parallel analog I/O buffers. There are 352 analog I/O lines available. In addition to the analog I/O, each neuron chip contains an analog multiplexer that enables the digital host to monitor and store the neuron activity for feeds over a common line into a 250 KHz A/D board located in the host. The monitor software generates either arrays of selected time segments of each neuron output that are displayed as separate graphs as shown in *Fig. 10*, or it generates a continuous gray scale

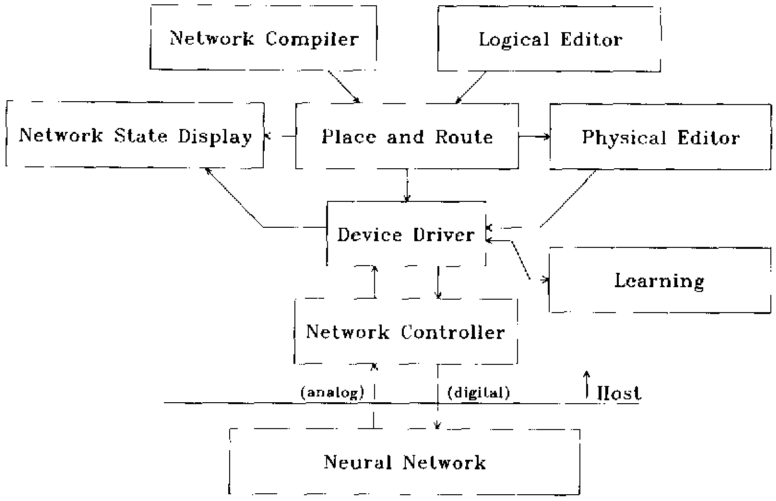

Fig. 9. Overview of the software for the neural computer.

display of activity for all neurons. Neuron activity is also displayed directly by an LED Panel seen in *Fig. 1*.

The neuron activity patterns read by the host also form the basis for implementation of different learning algorithms in which the network and the host form a closed loop. The learning of a simple OCR task by packpropagation has been demonstrated (see below).

4. Performance of simple computational tasks

The machine has been programmed for several tasks in order to evaluate its performance. A few examples are discussed. In all cases the actual neural computations (not learning) are performed in real time, with response time limited only by the bandwidth of the neurons which was maximally 300 KHz.

The first example (*Fig. 10*) is a 'Winner Take All' net which, depending on the inhibitory feedback gain, either extracts the largest output among 32 neurons or enhances the contrast between the different outputs. The network settles within the time constant of the neuron outputs and shows no oscillations. Simulations of this small network on a SUN 4 are slower by a factor of 100. Since this is a fully connected net the speed ratio would scale with n^2 on the number of neurons.

Other small circuits, not shown, have been programmed, including an associative net of 24 neurons, a motor control circuit that models the last stage of the control of saccadic eye movements, a neural integrator and several circuits for the computation of time domain pattern primitives.

An example of a time domain operation is shown in *Fig. 11* which shows small circuits that are tuned to the frequency of amplitude modulation of a carrier frequency. Such circuits are useful in the recognition of sonar patterns. Several coupled oscillators and central pattern generators have also been programmed. By making use of threshold bias, feedback and synaptic time constants, the networks can generate arbitrary dynamic activity patterns in the

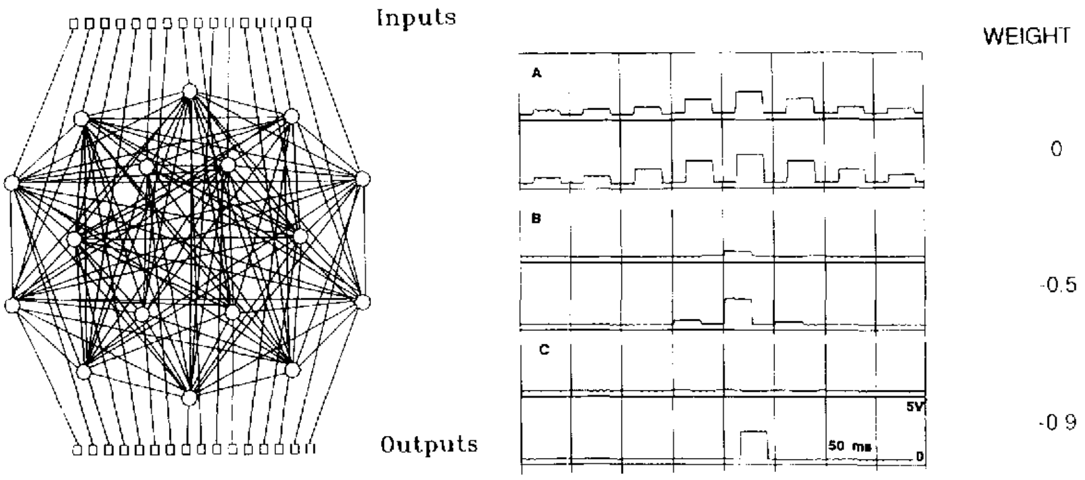

Fig. 10(a). Conceptual network and performance of a 'Winner Take All' network. 16 neurons receive different inputs and are connected by mutual inhibitory connections. The records at right show outputs from 16 neurons in response to a square wave input of different amplitude to each neuron. At A the weights of the inhibitory connections were set to zero. In B these gains were 0.5 and C, 0.9. The 0.5 inhibition results only in contrast enhancement. The records were obtained through the analog multiplexers on the neuron modules and were displayed with the monitor software by the digital host.

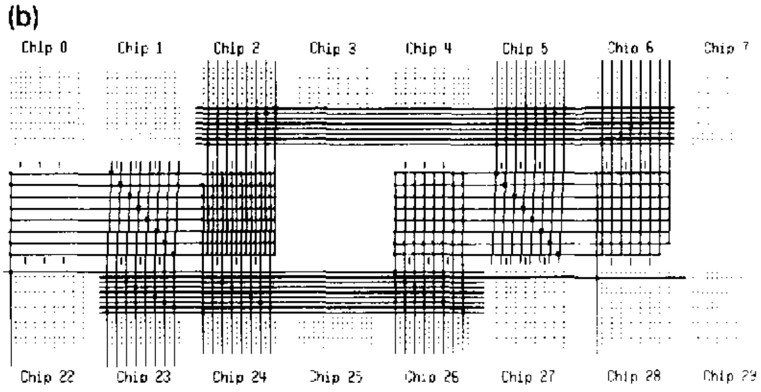

Fig. 10(b). Routing configuration of the 'Winner Take All' net. This example shows part of the graphic displays used for routing of connections and setting of synapse parameters. Parameters are selected from graphic menus for each chip module.

absence of external inputs.

The example of character recognition in *Fig. 12* demonstrates that the system can learn by backpropagation in a simple 4 layer network. Development of alternative learning algorithms for the machine is still at an early stage. Various algorithms can be run either by the host alone or with the neural computer in loop. Since the weights are computed and updated by the host, the learning speed for error correcting algorithms is improved only in those cases

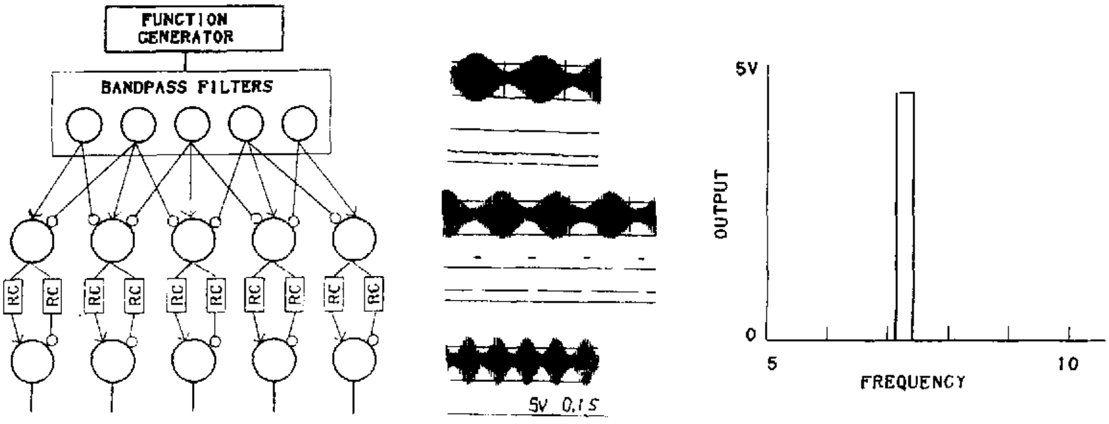

Fig. 11. Network for recognition of amplitude modulation frequency of a specific carrier frequency. The carrier frequencies, which are separated by a set of bandpass filters, are amplitude modulated by a function generator and fed into an array of first stage of neurons. Their outputs connect with inhibitory and excitatory synapses to second stage neurons. By adjusting the synaptic transfer time constants and weights, individual neurons can be sharply tuned to a narrow band of amplitude modulation frequency as seen in the middle records which show the amplitude modulated signal and the output from one of the second stage neurons. This unit responds only in a phase locked manner to a narrow band of modulation frequency (middle record). The tuning curve of this unit is shows at right.

where the actual performance computation is the rate limiting factor such as in networks with extensive front ends or networks having lateral or feedback connections or different synaptic time constants. However, we are developing single shot learning schemes for pattern recognition where the synapse parameters are determined from the input patterns without iterative adjustments.

The machine is especially suited for the real-time analysis of dynamic patterns such as speech. Another example of a more elaborate network for time domain computation is shown in *Figs. 13* and *14*. This network, involving most of the available neurons, performs an initial decomposition of acoustical patterns and the real time recognition of several phonetic speech sounds. The net receives analog input from a set of 8 bandpass filters (200–3000 Hz) and extracts local maxima of amplitude vs. frequency, (d2E/dS2), local rates of rise and fall of amplitude (+ dE/dt), and local rise and fall of frequency i.e. motion (+ dS/dT) of activity along the frequency axis (see also [3] and [4]). The activity patterns generated in this feature space by individual sounds are decoded by connection matrices to individual neurons that respond in real time selectively to a specific sound such as a phoneme.

5. Conclusions

Several conclusions can be drawn from these tests. First of all the machine performance has met all expectations regarding noise levels, accuracy, stability, programming flexibility and processing speed. Particularly the routing space proved entirely adequate. There is therefore no reason to project an increase in the ratio of switch modules to neuron and synapse modules

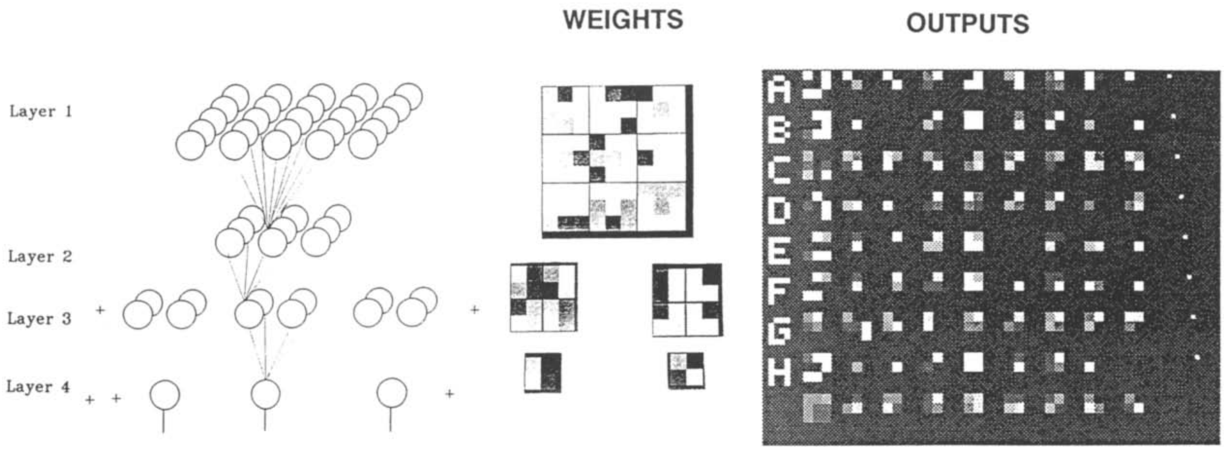

Fig. 12. Architecture and performance of a network for optical character recognition. The network consists of 5 × 5 input layer, a 3 × 3 second layer and a separate 2 × 2 array for each character. The recognition units form layer 4. Each unit in the second layer is connected to 9 units in layer 1 and each unit in layer 3 connects to 4 units in layer 2. The output units receive inputs from the 4 units in layer 3. The network was trained by backpropagation in conjunction with the neural computer for the letters A through H at shifted positions in the input layer and a Null input. The resulting weights as loaded in the neural computer are shown as gray scale pixels in the middle figure. At right are the outputs from the neurons in the different layers for the characters A to H. Although the outputs from the second and third layer units bear no resemblance to the input patterns, the recognition by the 4th layer units was 100%. The recognition is shift invariant.

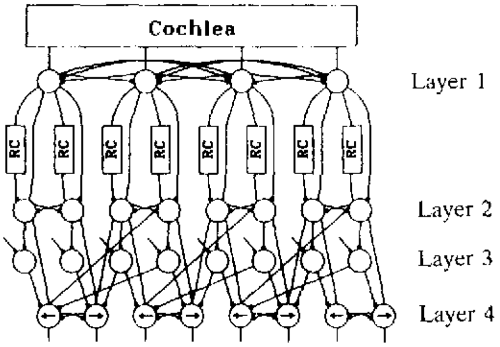

Fig. 13. Section of the conceptual design of a network for the primary decomposition of acoustical patterns. The primary neurons receive input from eight band pass filters. These neurons are connected with mutually inhibitory inputs in a center-surround scheme with spatially decaying gains. They extract the maxima of the sound amplitude. The next stage extracts separately the temporal rise and decay of the sound amplitudes. These neurons receive delayed excitatory or inhibitory inputs from the previous stage. The third stage neurons are normally 'on' through positive bias input and compute the complement of the activity of the second stage neurons. The fourth stage units compute the changes of frequency maxima and their direction through a combination of the second and third stage neurons. In essence they are motion detectors. The outputs from all these neurons provide inputs selectively to individual neurons that respond specifically to a particular phoneme as shown in *Fig. 14*. These connections are now shown in this figure.

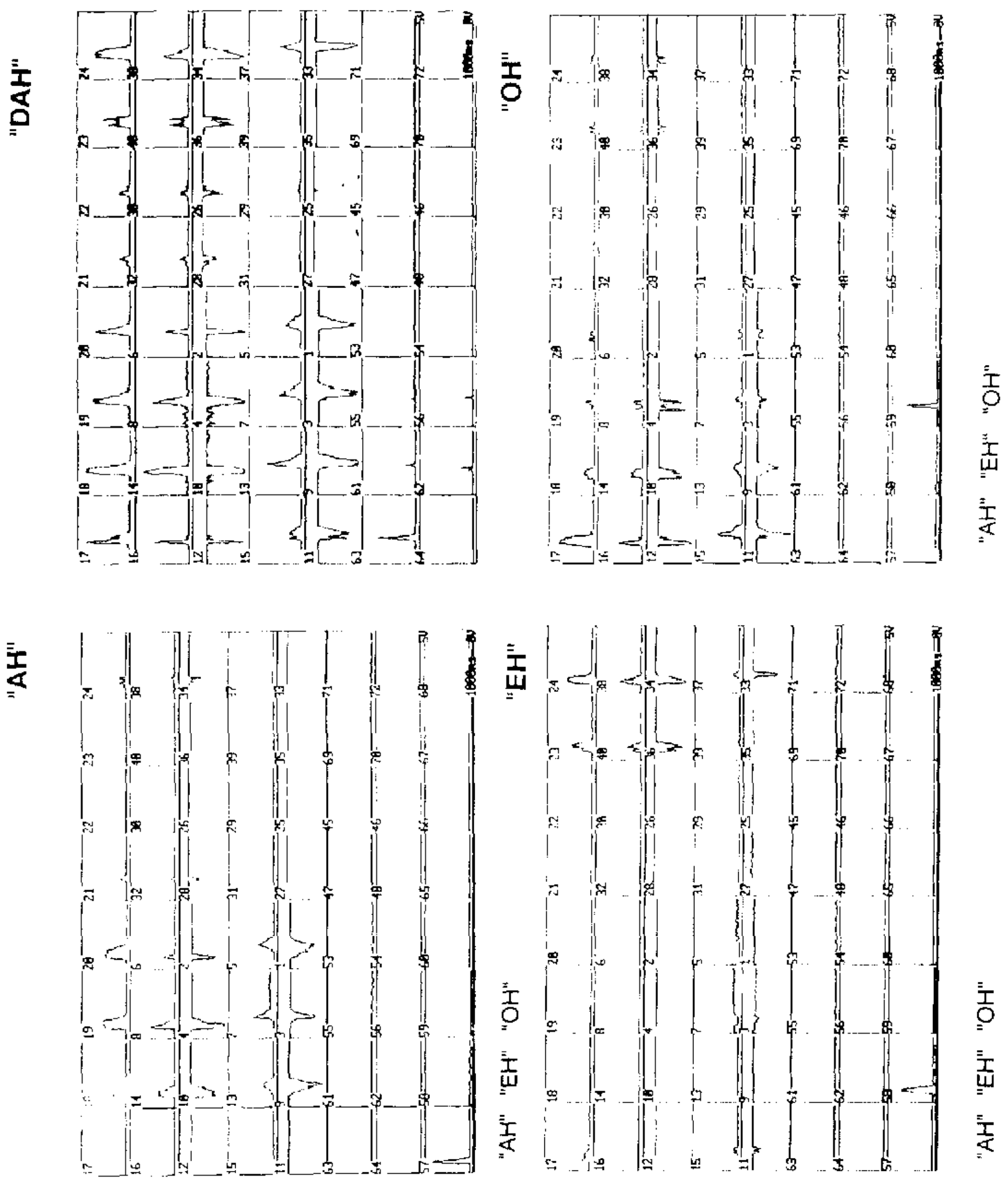

Fig. 14. Neuron outputs from the network shown in *Fig. 13* for the phonemes 'AH', 'DAH', 'EH', and 'OH', illustrating the differences of the patterns and their recognition by single neurons. Each block shows the output of a neuron as function of time during the pronounciation of the phoneme. The colums represent increasing frequency from left to right (from 200 to 3000 Hz). The activity of the neurons in the first row is proportional to the sound energy at the different frequencies. The second and fourth rows represent the positive and negative rate of change of the sound amplitude at the different frequencies and the third and fifth rows are the inverse of rows two and four. The outputs of neurons in rows five and six represent up and down changes of frequency as function of time as they occur, for example, in diphones during formant transitions in diphones. Such a transition is indicated by the activity of unit 63 for the diphone 'DAH'. The last row (units 57, 58,

59) for phonemes 'AH', 'EH' and 'OH' shows the outputs from neurons that receive excitatory and inhibitory inputs from the active neurons in rows one to seven such that they respond only to a particular phoneme, in this case 'AH', 'EH' or 'OH'. Notice that the outputs of the recognizing neurons have a shorter duration than the spectral components of the sound patterns seen in the top row. This arises from the fact that the 'ON' units are excitatory whereas the 'OFF' units are inhibitory. A detailed discussion of the decoding strategy used to determine the connection to the recognizing units is beyond the scope of this paper.

for larger machines. No major design change for the neurons is needed. The synapse gain (weighting) scheme could be improved by increasing the resolution for middle range gains (between 0.1 and 1). The range and resolution of synaptic time constants seems adequate.

The major advantage of analog hardware implementations of neural computation is speed. Simulations show that the matching of real time performance of even small networks similar to that in *Fig. 13* requires digital processing speeds exceeding 10^{11} flops and larger machines of this type in which the neuron band width is increased to 1 MHz, can be expected to approach 10^{14} equivalent flops.

Acknowledgement

This work was supported by grants from the Office of Naval Research and the National Science Foundation.

References

[1] P. Mueller, J. van der Spiegel et al., Design and fabrication of VLSI components for a general purpose analog neural computer, in: C. Mead, ed., *IEEE Workshop on VLSI and Neural Systems*, Portland, OR (1989) (Kluwer, Boston, MA).

[2] P. Mueller, J. van der Spiegel et al., Design and performance of a prototype general purpose analog neural computer, IJCNN, Seattle, WA (1991).

[3] P. Mueller and J. Lazzaro, A machine for neural computation of acoustical patterns with application to real time speech recognition, *AIP Conf. Proc.* 151 (1986) 321–326.

[4] P. Mueller, Computation of pattern primitives in a neural network for acoustical pattern recognition, *Proc. IJCNN* Washington (1990) 149–151.

Paul Mueller received his MD degree from Bonn University, Germany. He has worked in molecular and systems neuroscience since 1953 and has been involved in theoretical studies and hardware implementation of neural networks since the early sixties.

Jan van der Spiegel was born in Aalst, Belgium. He received his engineering degree in Electro-mechanical Engineering, and a Ph.D. degree in Electrical Engineering from the Katholieke University of Leuven, Belgium in 1974 and 1979, respectively. From 1980 to 1981, he was a postdoctoral fellow at the University of Pennsylvania where he became an assistant professor of Electrical Engineering in 1981. Currently, he is an associate professor of Electrical Engineering and of Material Science and Engineering. He holds the Bicentennial Chair of the Class of 1940 and is Director of the Center for Sensor Technologies at the University of Pennsylvania. His research interests are in artificial neural networks, CCD sensors, microsensor technology and low noise, low power, and high speed analog integrated circuits. He is the recipient of the Presidential Young Investigator Award, an IBM Award for Young Faculty Development, the Reid S. Warren Award and the C.M. Lindback Award for distinguished teaching. He is editor for Sensors and Actuators (North and South America). He is a senior member of the IEEE, and member of Tau Beta Pi and Phi Beta Delta.

David Blackman received BS degree in Computer Science and Engineering from University of Pennsylvania in 1981, and an MS in Computer Science From Rochester Institute of Technology in 1986. From 1981 to 1986, he worked as an engineer at Xerox Corporation in Rochester, NY. He received his PhD degree in 1992 from the Department of Neuroscience at the University of Pennsylvania where he worked on models of the visual cortex and VLSI implementation of neural networks. He is currently with Corticon Inc.

Peter Chance received a BS and MS degree in Computer Science from the University of Pennsylvania. From 1974 to 1984 he worked in robotics and machine learning. His current interests include speech recognition and robotic control using neural networks.

Christopher Donham was born in Greenwich, CT in 1967. He received a BSE and MSE degree in Electrical Engineering from the University of Pennsylvania in 1989 and 1991, respectively. He is currently working towards a PhD in electrical engineering at the University of Pennsylvania where he is researching neural network based speech recognition. In 1988, he received the Outstanding Student Member Award for the Philadelphia branch of the IEEE.

Ralph Etienne-Cummings was born in Victoria, Seychelles in 1967. In 1988 he graduated Valedictorian and Summa Cum Laude from the Lincoln University, PA with a BSc in Physics. He received his MSEE in 1990 from the University of Pennsylvania, where he is currently pursuing a PhD degree. From 1988 to 1990, he was a Fountain Fellow and is currently a Harris Fellow. His research interests include hardware implementation of neural networks and visual tracking systems.

Jinsoo Kim received a BSE in Electrical Engineering from the University of Pennsylvania in 1991. He is currently with KEI in Korea working on ASIC design.

Michael Mass was born in Cincinnati, Ohio in 1967. He obtained BSE in Computer Science Engineering from the University of Pennsylvania in December of 1990. He worked in the Distributed Systems Laboratory at the university before joining the Microsoft Corporation in July of 1991. He currently works on the development of network transport protocols.

Supun Samarasekera was born in Sri Lanka. He obtained a BSc in Computer Science and Engineering and a MS in Computer Information Systems from University of Pennsylvania in 1991. He worked for the Pennsylvania College of Podiatric Medicine from September 1991 to February 1992. Since then he has been working with the Medical Image Processing Group at the University of Pennsylvania.