$$V_{C=} = V_i - V_T - \lambda_1 V_{BT} \text{ and } V_{O=} = V_{DD} - V_T - \lambda_2 V_{BT} \qquad (1)$$

where $\lambda_1 = \sqrt{}(\beta_1/\beta_2)$, and $\lambda_2 = \sqrt{}(\beta_1/(n-1)\beta_3)$, and $V_{BT} = V_B - V_T$. From eqn. 1 we verify that, for the saturation condition to be satisfied for any input level, the transistor parameters must obey $\lambda_1 \leq (V_{imin} - V_B)/V_{BT}$ and $\lambda_2 \leq (V_{DD} - V_{imax})/V_{BT}$, so establishing the following input dynamic range:

$$V_B + \lambda_1 V_{BT} \leq V_i \leq V_{DD} - \lambda_2 V_{BT} \qquad (2)$$

If we let now one of the inputs (say V_1) increase, while all the others stay the same (worst case), V_{O1} decreases toward $V_C \simeq V_C$, whereas $V_{O2} = V_{O3} = \ldots = V_{On}$ increase. As mentioned earlier, if the transistor parameter ratios are small enough they may cause the latter voltage to be > $V_{DD} - V_T$; this situation exists if

$$\lambda_2 < (2V_{DD} - 2V_C - 3V_T)V_T/V_{BT}^2 \qquad (3)$$

which is independent of n, as expected, because in this state each active cell has just one active load.

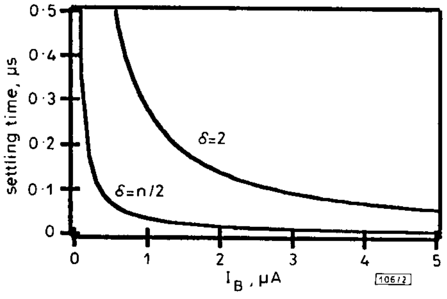

Fig. 2 *Settling time against power consumption*

$n = 32$, $(W/L)_3 = 6/6$, $\Delta V = 1$V

We return now to the time response. Although the total bias current nI_B is constant, only a fraction δI_B $(1 < \delta < n)$ is destined to discharge the winning node (V_{O1}), the magnitude of δ depending on how far V_1 is from the other input voltages. The settling time of the system is therefore given by

$$T = C\Delta V/\delta I_B \qquad (4)$$

where C is the capacitance of the output node ($\sim(n-1)(WL)3 \times 0.5$fF/μm^2) and ΔV ($\simeq V_T$) is the difference between V_{PRE} ($\simeq V_{DD}$) and the transition voltage of the output inverter ($\simeq V_{DD} - V_T$). This behaviour is illustrated in Fig. 2 for practical lower and upper bounds of δ.

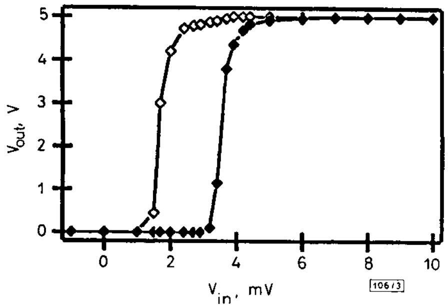

Fig. 3 *System gain*

Experimental results: The performance of the network was measured on an $n = 32$ WTA circuit fabricated on a 2.0μm CMOS chip, with $L = 10$μm, $\lambda_1 = 0.7$, and $\lambda_2 = 0.4$. The experiments agreed consistently with the predictions, showing always only one winner present at a time, as imposed by the equilibrium state of the circuit. A wide system resolution was obtained (~50dB), with a sensitivity better than 10mV in the worst case. Two separate gain measurements are shown in Fig. 3, where the low offset of the system is apparent; these tests were performed with $V_B = 1.0$V and V_{REF} (all inputs but one) = 2.0V.

An application of the circuit is shown in Fig. 4. The WTA was used to detect the winning output of a Hamming network [5]. All rows of the system processed the same random vector shown on the left-hand side of Fig. 4, except one row, which processed the vector shown on the right-hand side. (The latter vector has only one bit distinct from the former, which circulates over the 8 × 8 block and is always in agreement with the corresponding input bit.) As expected, the circuit yielded a fixed winner.

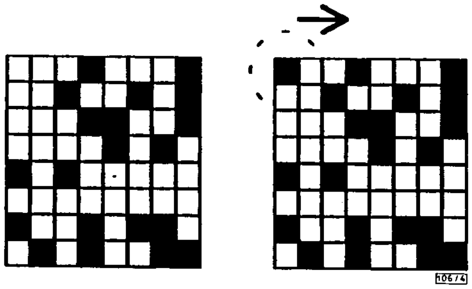

Fig. 4 *WTA application*

Conclusion: We have made use of a neural architecture to introduce a new winner-take-all network. Basic properties of the circuit were qualitatively and quantitatively discussed and experimental results were presented. Positive-feedback, high gain, and small offset make possible the detection of very small perturbations, which are immediately decoded through digital outputs directly available on the network.

© IEE 1994 29 July 1994
Electronics Letters Online No: 19941182

V. A. Pedroni (*California Institute of Technology, Dept. of Electrical Engineering, 128-95 Pasadena, CA 91125, USA*)

References

1 LAZZARRO, J., RYCKEBUSH, S., MAHOWALD, M., and MEAD, C.: 'Winner-take-all networks of O(N) complexity', *Neural Information Processing Systems - NIPS*, **1**, 1989

2 ANDREOU, A., BOAHEN, K., POULIQUEN, P., PAVASONIC, A., and JENKINS, R.: 'Current-mode subthreshold MOS circuits for analog VLSI neural systems', *IEEE Trans.*, **NN-2**, 1991, pp. 205-213

3 HE, Y., CILINGIROGLU, U., and SANCHEZ-SINENCIO, E.: 'A high-density and low-power charge-based Hamming network', *IEEE Trans., VLSI*, 1993, pp. 56-62

4 HOPFIELD, J.: 'Neural networks and physical systems with emergent collective computational ablitites', *Proc. Nat. Ac. of Sciences*, **79**, 1982, pp. 2554-2558

5 PEDRONI, V., AGRANAT, A., and YARIV, A.: 'Pattern matching and parallel processing with CCD technology'. Int. Conf. on Neural Networks, 1992, Baltimore

Real time output derivatives for on chip learning using digital stochastic bit stream neurons

M. van Daalen, J. Zhao and J. Shawe-Taylor

Indexing terms: Learning systems, Neural chips, Stochastic automata

The authors present the hardware design of an extremely compact and novel digital stochastic neuron, that has the ability to generate the derivative of its output with respect to an arbitrary input. These derivatives may be used to form the basis of an on chip gradient descent learning algorithm.

Introduction: An artificial neuron is required to calculate a single output value by applying an 'activation function' to the weighted sum of its inputs. Such neurons are intended to operate in massively parallel networks, often processing real time data. Conventionally, feedforward networks containing neurons of this type are trained off-line using learning algorithms such as back propagation, but recently some research has focused on building the learning algorithms directly into the neural hardware [1, 2].

In this Letter we present the design of an enhanced stochastic bit stream neuron that contains additional circuitry that allows the real time calculation of the neuron's output derivative with respect to an arbitary input. This derivative may then be used as the basis for an 'on chip' gradient descent learning algorithm. The detailed hardware design and operating principles of standard stochastic bit stream neurons and their networks is given in [3–5, 7].

Stochastic bit stream neuron: To describe the process required to calculate the output derivative of a stochastic bit stream neuron, we will begin by giving a brief description of the basic operation of such a neuron.

All signals processed by these neurons are real values represented by stochastic bit streams in the interval [0,1] for unsigned values, and [-1,1] for signed values. A neuron has only one physical input and weight connection, but by the use of time division multiplexing, may have many logical connections. The core of the neuron is a simple counter, which may be preloaded with a threshold value. Each input bit is weighted by either ANDing, when operating on unsigned values, or XORing, when operating on signed values, with a corresponding weight bit. Thus this weighted input contributes 0 or 1 to the counter on each operational cycle. Details of signed and unsigned stochastic bit stream neurons may be found in [4, 8]. The unique threshold values supplied to the counter are chosen such that they will cause an overflow into the top-most bit, when a given input count is achieved or exceeded. Thus the top bit of the counter provides the output of the neuron.

The activation function applied by the neuron, which requires no additional circuitry, is formed by the interaction of the probability distribution of the weighted input values, and the probability distribution of the chosen threshold values. A sigmoid like activation function is achieved by using a fixed threshold value, and a linear activation function is obtained when using a uniformly distributed threshold value; see [6] for the precise mathematical definitions.

Calculating the derivative: The probability of generating a '1' as the output bit on a given operational cycle of a bit stream neuron, with weighted inputs i_1 to i_m, and preloaded threshold value t_n, may be written as shown in eqn. 1.

$$O_n = \Pr(i_1+i_2+i_3+\cdots+i_m > t_n) \qquad (1)$$

$$= \Pr(i_1+i_2+i_3+\cdots+i_{k-1}+i_{k+1}+\cdots+i_m > t_n)$$
$$+\Pr(i_1+i_2+i_3+\cdots+i_{k-1}+i_{k+1}+\cdots+i_m = t_n)$$
$$\times \Pr(i_k = 1) \qquad (2)$$

This function may be rewritten as eqn. 2, which is now easily differentiated with respect to the arbitary input i_k, giving eqn. 3:

$$\frac{\partial O_n}{\partial i_k} = \Pr(i_1+i_2+i_3+\cdots+i_{k-1}+i_{k+1}+\cdots+i_m = t_n) \quad (3)$$

To implement this result for a given neuron, additional hardware will be required. This circuitry must prevent the input i_k from contributing to the internal counter, and also must detect the condition that the counter exactly matches the preloaded threshold value. This is easily arranged, as the preloaded threshold value is chosen such that it sets the most significant bit of the counter when achieved. So if i_k is '0' then the circuitry must detect the counter value '1000...00', or if i_k is '1', it must detect the value '1000...01'. This functionality can be achieved with a simple combinatorial circuit. The number of counter bits required by a typical bit stream neuron with m inputs, which must now be checked, is given by $2\log_2 m$, which for most applications will be small, ~8 bit. Actual circuit details are not given here, as these will depend largely on the final hardware implementation platform, the most efficient being fully custom VLSI.

Results: Four graphs showing output functions and their derivatives with respect to input 1 are presented in Fig. 1. The bit stream neurons used had 15 inputs with either linear or sigmoidal activation functions. Each function is displayed twice. In the first instance all of the inputs to each neuron are distinct 1 Mbit streams of the same value following the linear ramp function $y = x$. The second set of two graphs shows neurons with the same activation functions, but with input 1 set to sin $2x$, and the remaining inputs 2 to 15 set to $y = x$ as before.

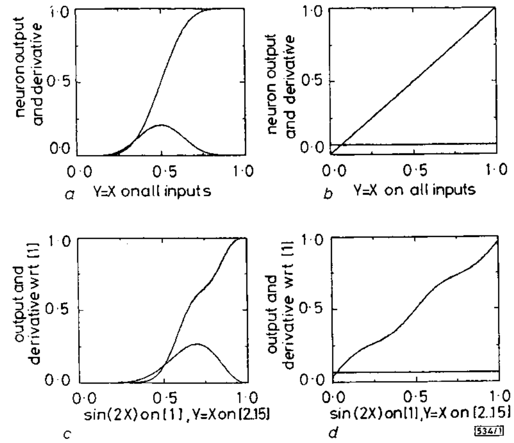

Fig. 1 *Neural activation functions, and their derivatives*

a Sigmoid activation function, 1 Mbit on each of 15 inputs
b Linear activation function, 1 Mbit on each of 15 inputs
c Sigmoid activation function, 1 Mbit on each of 15 inputs
d Linear activation function, 1 Mbit on each of 15 inputs

For a neuron with a linear activation function the derivative of the output with respect to a given input will always be constant, irrespective of the inputs. This is illustrated by the two graphs showing outputs and derivatives of a linear neuron.

In the case of a bit stream sigmoid neuron, the actual activation is a complex function of the inputs [6]. A fully symmetric sigmoidal activation function is only achieved when all the inputs are the same value, (each one must be represented as a distinct bit stream) and the threshold value is chosen as the midpoint of the input range on a given operational cycle, i.e. 7 for a 15 input device. A direct consequence of this is that the derivative of the sigmoidal activation is also a function of the inputs. Two examples of this behaviour can be seen in the appropriate graphs shown in Fig. 1.

The next two graphs, shown in Fig. 2, show the same results as the first two of the last set, but here the lengths of the bit streams presented to the inputs of the neuron have been reduced to 10k and 1k bit. The resulting increase in noise caused by the random variance errors inherent in stochastic bit streams is easily apparent from the graphs.

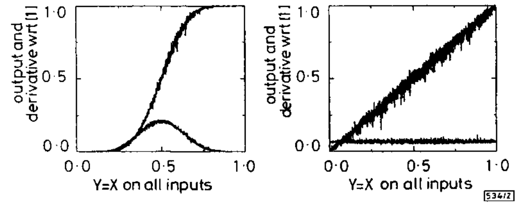

Fig. 2 *Sigmoid and linear activations with derivatives at 10k and 1kbit*

a Sigmoid activation function, 10kbit on each of 15 inputs
b Linear activation function, 1kbit on each of 15 inputs

Conclusion: We have demonstrated that it is possible to construct linear or sigmoidal stochastic bit stream neurons with the ability to generate real time output derivatives using only simple digital circuitry. We are currently evaluating an 'on chip' learning scheme designed for feedforward networks that is closely modelled on back propagation [9]. In this scheme each bit stream neuron contains additional circuitry to produce the appropriate δ values. These are then aggregated and passed back through the network to their respective weights by additional layers of simplified linear neurons.

© IEE 1994 5 September 1994
Electronics Letters Online No: 19941233

M. van Daalen, J. Zhao and J. Shawe-Taylor (*Connection Science and Machine Learning Group, Royal Holloway, University of London, Egham, Surrey, United Kingdom*)

References

1 EGUCHI, H., FURUTA, T., HORIGUCHI, H., OTEKI, S., and KITAGUCHI, T.: 'Neural network LSI chip with on chip learning'. IJCNN, 1991, (Seattle), Vol. 1, pp. 453–456

2 BOGASON, G.: 'Generation of a neuron transfer function and its derivatives', *Electron. Lett.*, 1993, **29**, (21), pp. 1867–1869

3 VAN DAALEN, M., KOSEL, T., JEAVONS, P., and SHAWE-TAYLOR, J.: 'Emergent activation functions from a stochastic bit stream neuron', *Electron. Lett.*, 1994, **30**, (4), pp. 331–333

4 VAN DAALEN, M., JEAVONS, P., and SHAWE-TAYLOR, J.: 'A stochastic neural architecture that exploits dynamically reconfigurable FPGAs'. IEEE Workshop on FPGAs for Custom Computing Machines, 1993, (Napa, CA), pp. 202–211

5 STANFORD TOMLINSON, M., WALKER, D.J., and SILVILOTTI, M.A.: 'A digital neural network architecture for VLSI'. IJCNN, 1990, (San Diego), Vol. 1, pp. 545–550

6 SHAWE-TAYLOR, J., JEAVONS, P., and VAN DAALEN, M.: 'Probabilistic bit stream neural chip: Theory', *Connection Science*, 1991, **3**, (3), pp. 317–328

7 VAN DAALEN, M., JEAVONS, P., SHAWE-TAYLOR, J., and COHEN, D.: 'Device for generating binary sequences for stochastic computing', *Electron. Lett.*, 1993, **29**, (1), pp. 80–81

8 VAN DAALEN, M., JEAVONS, P., and SHAWE-TAYLOR, J.: 'Probalilistic bit stream neural chip: Implementation'. Int. Workshop on VLSI for Artificial Intelligence and Neural Networks, September 1990, (Oxford University)

9 WERBOS, P.J.: 'The roots of backpropagation' (John Wiley and Sons, 1993)

80Gbit/s soliton data transmission over 500km with unequal amplitude solitons for timing clock extraction

M. Nakazawa, E. Yoshida, E. Yamada, K. Suzuki, T. Kitoh and M. Kawachi

Indexing terms: Soliton transmission, Optical communication

Single-polarisation 80Gbit/s soliton data signals have been successfully transmitted over 500km. The soliton source was a modelocked fibre laser and a planar lightwave circuit was used for stable optical multiplexing. A nonlinear loop mirror was used for demultiplexing, in which unequal amplitude solitons were used for clock extraction.

There are two interesting fields of application for soliton communication. One is long distance transoceanic communication over 10000km, in which the transmission speed is limited to 5–40Gbit/s under the various dispersion and amplifier spacing conditions [1–4]. The other is relatively short distance communication over 1000km, where the transmission speed is 100Gbit/s–1Tbit/s. This may prove useful as a high speed information highway.

The experiments for the former application have been undertaken by many groups using loop circulations or straight line transmissions, however there have been few experimental reports of the latter type. This is because the pulse width of a soliton source should be much shorter than 10ps, but it is not easy to generate such a pulse using a gain-switched laser diode or electron absorption modulators. In such a high speed soliton system, the amplifier spacing is longer than the soliton period. In addition, dispersion irregularities should be reduced, i.e. the standard deviation of the soliton periods should be smaller than that with longer soliton pulses.

As long as we use some form of soliton control such as synchronous amplitude or phase modulation with optical filtering, it is possible to send a high speed soliton signal over unlimited distances [5]. However, it is also becoming increasingly important to realise soliton transmission of the order of 100Gbit/s over a relatively long distance, as a linear signal has already been transmitted at 100Gbit/s over 200km [6].

In this Letter we show for the first time that it is possible to transmit an 80Gbit/s soliton data signal through a 500km straight fibre line. This transmission distance is the longest yet reported for such high speed soliton communication.

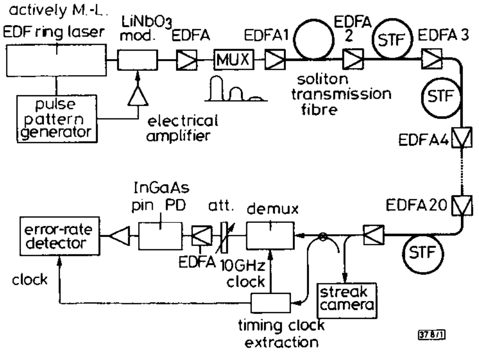

Fig. 1 *Experimental setup for 80Gbit/s soliton transmission over 500km*

The experimental setup for the 80Gbit/s-500km soliton transmission is shown in Fig. 1. The soliton source was an actively modelocked 10GHz erbium fibre ring laser, which could emit a 2.7–3.0ps soliton pulse. The pulse was a transform-limited pulse and modulated at 10Gbit/s with a 2^{15}–1 pseudorandom binary sequence using an $LiNbO_3$ (LN) intensity modulator. A planar lightwave circuit was used as a stable optical multiplexer to obtain a 80Gbit/s pulse train. To obtain a 10GHz clock signal easily from the transmitted 80Gbit/s signal, 10GHz soliton units were superimposed on each other with slightly different soliton amplitudes. This technique is also useful for reducing the soliton-soliton interaction [4, 7].

The soliton transmission fibres (STFs) were dispersion-shifted fibres with an average dispersion as low as –0.19ps/km/nm at 1.552μm. The average soliton period was 19.0km, which meant that the amplifier spacing had to be shortened to as little as 25km. The coded pulses were amplified by EDFAs to an average soliton power level of +8.2dBm when the mark rate of the pseudorandom signal was 1/2. The average $N = 1$ soliton peak power was as high as 31.5mW. The average fibre loss including the connector loss for one span was ~6.0dB. A narrowband optical filter with a pass band of 3nm was installed every 50km to stabilise the soliton train.

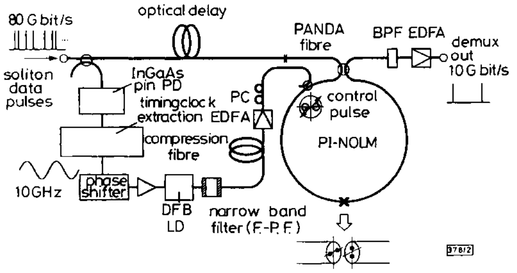

Fig. 2 *Optical demultiplexing circuit using a polarisation-insensitive nonlinear optical loop mirror*

An 80Gbit/s soliton data signal was demultiplexed to a 10Gbit/s signal using a polarisation-insensitive nonlinear optical loop mirror (PI-NOLM) [8]. The demultiplexing circuit is shown in Fig. 2. Part of the transmitted 80Gbit/s soliton signal was detected with a high speed InGaAs *pin* photodiode and a 10GHz clock signal was extracted. A high SN clock was obtained because of the 10GHz component resulting from our use of an unequal amplitude soliton train. Then, the sinusoidal clock signal drove the DFB LD under a gain-switching condition, and the generated optical pulse was converted to a transform-limited 9 ps pulse with a combination of spectral filtering and linear compression techniques. The 9 ps pulse