Current-Mode Computation with Noise in a Scalable and Programmable Probabilistic Neural VLSI System

Chih-Cheng Lu and H. Chen

The Dept. of Electrical Engineering,
The National Tsing-Hua University, Hsin-Chu, Taiwan 30013
hchen@ee.nthu.edu.tw

Abstract. This paper presents the VLSI implementation of a scalable and programmable Continuous Restricted Boltzmann Machine (CRBM), a probabilistic model proved useful for recognising biomedical data. Each single-chip system contains 10 stochastic neurons and 25 adaptable connections. The scalability allows the network size to be expanded by interconnecting multiple chips, and the programmability allows all parameters to be set and refreshed to optimum values. In addition, current-mode computation is employed to increase dynamic ranges of signals, and a noise generator is included to induce continous-valued stochasticity on chip. The circuit design and corresponding measurement results are described and discussed.

Keywords: Probabilistic VLSI, noise, scalable and programmable systems.

1 Introduction

Probabilistic models use stochasticity to generalise the natural variability of data, and have been shown promising for reasoning biomedical data or for solving weakly-constrained problems such as pattern recognition. Realising probabilistic models in the Very-Large-Scale-Integration (VLSI) is thus attractive for the application like intelligent sensor fusion in implantable devices [1] [2]. However, only a few probabilistic models are amenable to VLSI implementation [3] [4], and most of which relies greatly on precise computation of Bayesian rules or vector products, which becomes infeasible as transistor noise and hardware non-ideality grow.

The CRBM is a probabilistic model which has been shown capable of classifying biomedical data reliably [5] and has been realised as a probabilistic VLSI system [6], potential for being an intelligent embedded system in implantable devices. With a fixed number (six) of neurons, however, the prototype system is limited to model two-dimensional data, while biomedical signals in real-world applications are normally high-dimensional and complex. Therefore, modular design is employed in the VLSI implementation presented here, allowing the network size to be expanded by connecting multiple chips. All parameters of the system are stored in dynamic analogue memory which can be not only refreshed at optimum values reliably but also trained by chip-in-a-loop configuration. The full system has been designed and fabricated with the TSMC

C. Alippi et al. (Eds.): ICANN 2009, Part I, LNCS 5768, pp. 401–409, 2009.

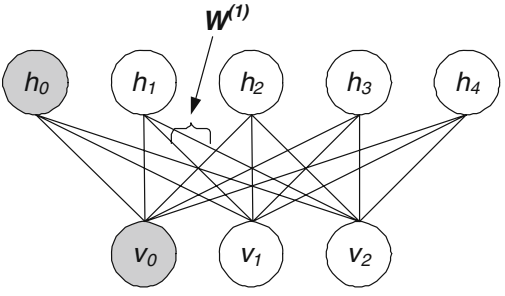


Fig. 1. The architecture of a CRBM model with two visible and four hidden neurons. v_0 and h_0 represent biasing units with invariant outputs $v_0 = h_0 = 1$.

$0.35\mu m$ CMOS technology. Following a brief introduction of the CRBM model, the architecture of the VLSI system, the circuit design, and the measurement results will be presented.

2 The CRBM Model

The CRBM consists of one visible and one hidden layers of stochastic neurons with inter-layer connections only, as shown in Fig.1. The number of visible neurons corresponds to the dimension of data, while that of hidden neurons is chosen according to data complexity [5]. Let w_{ij} represents the bi-directional connection between v_i and h_j. The stochastic behaviour of a neuron s_i is described by [5]

$$s_i = \varphi_i(a_i \cdot (\Sigma_j w_{ij} \cdot s_j + N_i(0,\sigma))) \tag{1}$$

where $N_i(0,\sigma)$ represents a zero-mean Gaussian noise with variance σ^2, and $\varphi_i(\cdot)$ a sigmoid function with asymptotes at ± 1 (e.g. $tanh(\cdot)$). Parameter a_i controls the slope of the sigmoid function and thus the variance of s_i.

As a generative model, the CRBM learns to "regenerate" the probabilistic distribution of training data at its visible neurons. Testing data can be subsequently classified according to the responses of hidden neurons [5]. Both $\{a_i\}$ and $\{w_{ij}\}$ can be trained by the simplified minimising-contrastive-divergence (MCD) algorithm, requiring only addition and multiplication of neurons' states to determine updating direction [7]. The simplicity and the locality make the training algorithm hardware-friendly.

3 System Architecture

Fig.2 shows the architecture of the scalable and programmable CRBM system [8], containing neuron modules (v_i and h_j), synapse modules (wij), a noise generator, and digital control circuits. The refreshing unit is designed to be realised by a microcontroller off-chip. Each synapse module contains two multipliers to calculate $w_{ij}h_j$ and $w_i jv_i$ as current inputs for neurons v_i and h_j, respectively. Each neuron $v_i(h_j)$ then sums up the

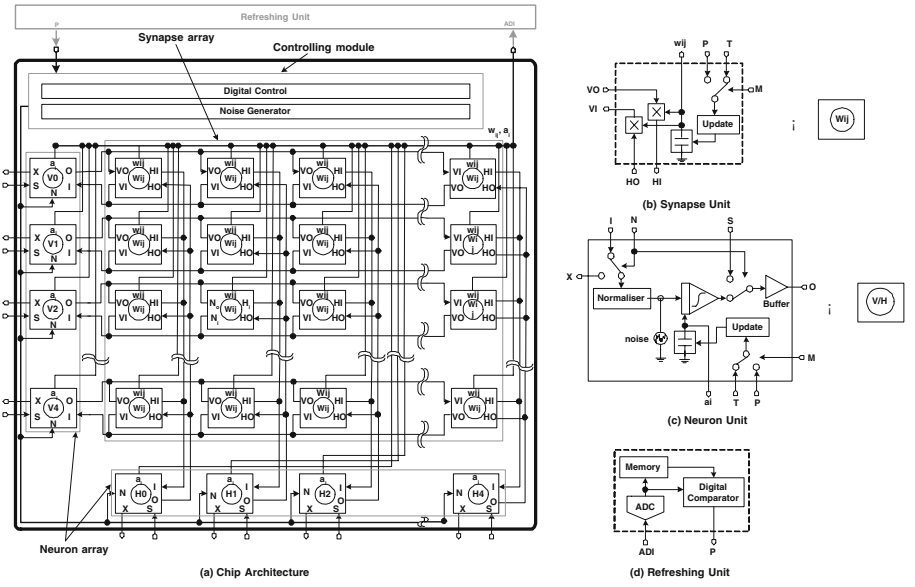


Fig. 2. The architecture of a scalable and programmable CRBM system and its functional units

currents on the same row (column) at the terminal I (Fig.2(c)), and passes the total current through sigmoid function to generate an output voltage at terminal O. In addition, each neuron includes a noise input to makes its output probabilistic.

The modular design enables the CRBM system to expand its network size easily by interconnecting multiple chips. For example, an MxN chip array forms a CRBM system with 5M visible and 5N hidden neurons. Synapse modules in the same row (column) transmit output currents to the left- (bottom-) most neurons in the row (column). Each neuron module $v_i(h_j)$ then transmits voltage output back to synapse modules in the same row (column). The control signal N in each neuron (Fig.2(c)) determines whether the neuron is enabled. When N=1, current inputs at terminal I are passed through sigmoid circuit to generate the neuron's output at terminal O. When N=0, the current inputs at terminal I are simply directed to terminal X, and the neuron output is buffered from terminal S into terminal O. A current normaliser is included to avoid the saturation of sigmoid circuit.

The parameters $\{w_{ij}\}$ and $\{a_i\}$ are stored locally as voltages across capacitors in the synapse and neuron module, respectively. The updating circuit employed in [9] is used to tune the capacitor voltages with infinitely small steps according to the digital input P or T. In training mode (M=1), the digital signal T is selected and calculated according to the simplified MCD algorithm. As soon as optimum levels are obtained, the analogue-to-digital converter (ADC) in the refreshing unit stores w_{ij} and a_i into digital memory (Fig.2(d)). Note that one ADC can be shared by all parameters. In refreshing mode (M=0), parameter values on capacitors are sampled periodically by the ADC, compared with optimum levels stored in the memory, and updated according to the output P of digital comparator.

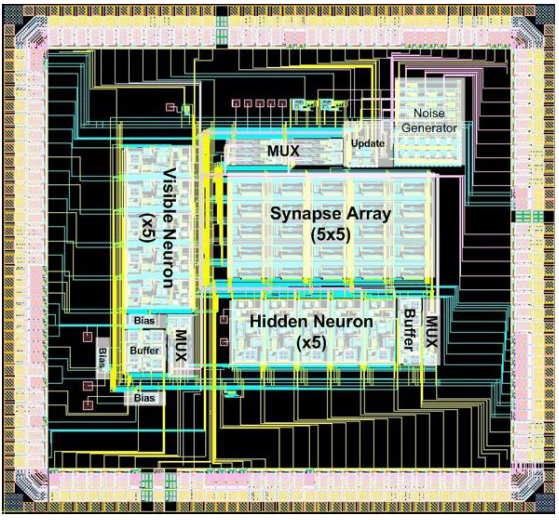


Fig. 3. The layout of the scalable and programmable CRBM system in VLSI

Table 1. The mapping of parameter values between software simulation and hardware implementation

	Matlab	VLSI(V)
s_i	[-1.0, 1.0]	[1.0, 2.0]
w_{ij}	[-3.0, 3.0]	[0.0, 3.0]
a_i	[0.5, 5.0]	[1.0, 2.5]

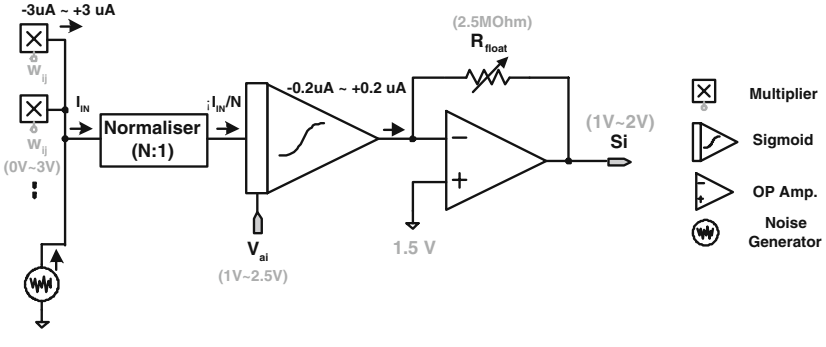


Fig. 4. The block diagram of the neuron with corresponding signal flows and the dynamic ranges of parameters in each stage

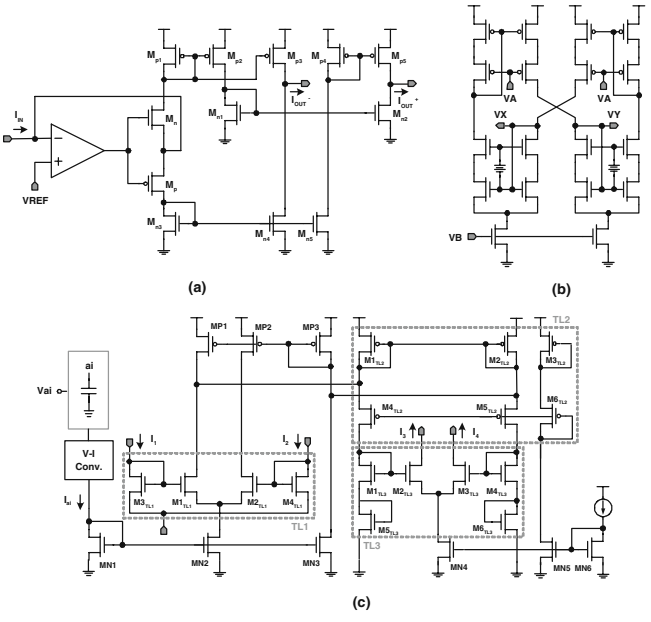


Fig. 5. The sub-circuits in the stochastic neuron. (a) Current conveyor with N:1 normaliser (b) Floating resistor (c) Sigmoid circuit.

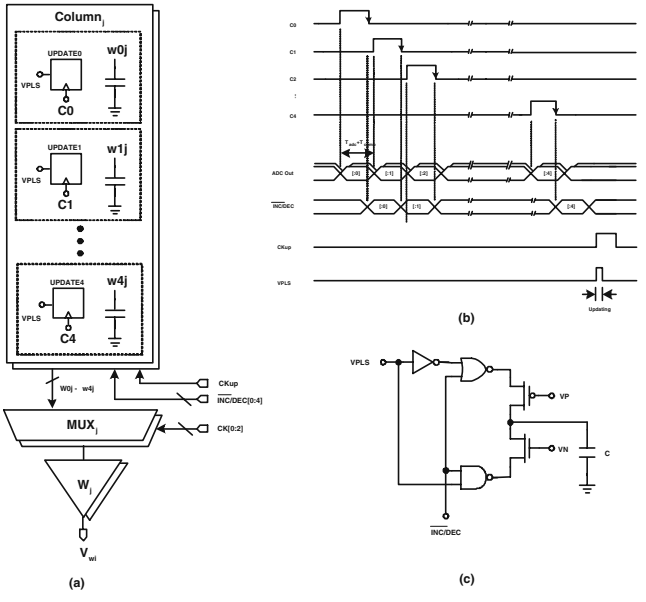


Fig. 6. (a) A multiplexing architecture for programming parameters in the CRBM system. (b) Clock signals for programming (c) Local updating circuit.

4 Circuit Design

The full system has been designed and fabricated with TSMC $0.35\mu m$ technology. Fig.3 shows the layout of the full system. the layout of the full system. The circuit area excluding the pads is $3100\mu m \times 2800\mu m$, and the power consumption is 6.8mA. With a supply voltage of 3V, the parameter mapping is defined as Table 1. The synapse module simply employs the "modified Chible multiplier" proposed in [7] as four-quadrant multipliers to calculate $w_{ij} \cdot s_j$. On the other hand, the noise generator proposed in [12] is implemented to generate 10 channels of uncorrelated noise on-chip. The following subsections describe the circuits of the neuron module and the programmable parameter array. In addition, Fig.4 shows the block diagram of the neuron indicating how hardware is implemented according to the mapping table (Table 1), and the details of the circuit design are described below.

4.1 The Neuron

Fig.5 shows the sub-circuits in the neuron module. The current conveyer with N:1 normalisation (Fig.5(a)) uses an operational amplifier (OPamp) with negative feedback to provide a low-impedance input point for all synapses connected to the neuron [10]. The noise voltage from the noise generator is also transformed into a noise current by a differential pair and then connected to the same node. By controlling the tail current of the differential pair, the noise current can be scaled as σ in Eq.(1) [7]. The current I_{IN} proportional to $(\Sigma_j w_{ij} \cdot s_j + N_i(0,1))$ in Eq.(1) is then normalised by N times through current mirrors, and subsequently transformed into a differential outputs, I^+_{OUT} and I^-_{OUT}. With $I_1 = I^+_{OUT}$ and $I_2 = I^-_{OUT}$, the sigmoid circuit in Fig.5(c) employs three translinear loops (TL1-TL3) to generate a differential output $(I_3 - I_4)$ whose value is approximately the sigmoidal function of $(I_1 - I_2)$ [13]. Moreover, the voltage representing a_i is converted into the current I_{ai} via the voltage-to-current converter (V-I converter) proposed in [9], controlling the slope of the sigmoid function. Finally, the current $(I_3 - I_4)$ is converted into a voltage representing s_i by an Op-amp with a negatively feedback resistor. The feedback resistor is implemented by the circuit shown in Fig.5(b) to achieve a resistance of more than several $M\Omega$. with a compact area [11].

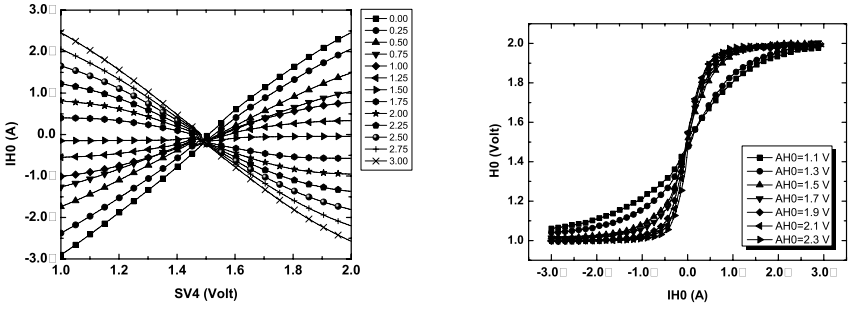
4.2 Programmable Parameter Array

The full microsystem contains 35 parameters (25 w_{ij} 10 a_i), which are arranged into a 5x7 array and multiplexed by the architecture shown in Fig.6(a). Fig.6(b) shows the corresponding digital-control signals. With C0-C4 decoded from CK[0:2], five parameter values in the same column $(w_{0j} - w_{4j})$ are selected sequentially by MUX_j and connected to the off-chip ADC. In refreshing mode, and connected to the off-chip ADC. In refreshing mode, w_{0j} is first selected and compared with its target value as C0=1. The signal (INC/DEC[:0]) representing update direction is then determined and stored in a register next to the updating circuit. With the same procedure, sequential activation of C1-C4 determines and stores the update direction for the other four parameters. Once the update directions of all parameters are obtained, CKup=1 triggers updating circuit

to tune all parameter once. The updating step is controlled by the pulse-width of the signal VPLS. In training mode, update directions are multiplexed and registered into the parameter array in a similar manner, except for that the update directions are calculated from the MCD algorithm.

5 Measurement Results

Fig.7(a) shows the measured output current of one multiplier in the synapse w40. With the output O of the neuron V4 sweeping from 1V to 2V, the output currents at IH, in response to different levels of w40, were measured. Obviously, the multiplier allowed the parameter w40 to have a rail-to-rail dynamic range and exhibited satisfactory linearity. Furthermore, the arithmetical zeros located at 1.5V precisely, agreeing with the mapping in Table 1.

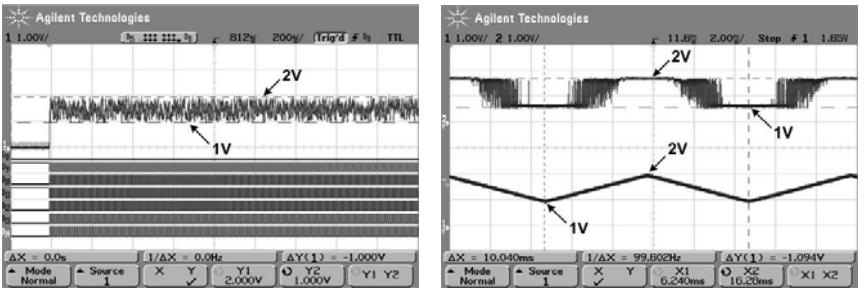


(a) (b)

Fig. 7. The measured DC characteristics of (a) a four-quadrant multiplier and (b) a sigmoid circuit with its slope controlled by the voltage AH0

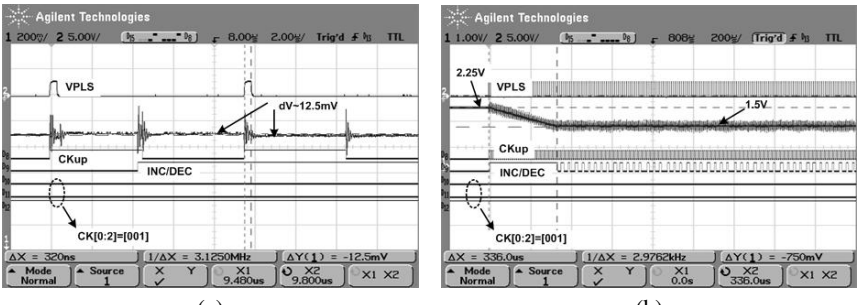
Fig.7(b) shows the measured characteristics of the sigmoid circuit in the neuron H0. With the current at the input I (called IH0) sweeping from $-3\mu A$ to $3\mu A$, the voltage at the output O (called H0) was measured. Different curves correspond to different levels of the voltage AH0, which controls the slope of the sigmoid function. As $\varphi(1) = 0.462$ and the unit values of IH0 and H0 are 1 A and 0.5V, respectively, the adaptable range of AH0 corresponds to an adaptable range of [0.5, 2.5] for the parameter a_i, covering the required range ([0.5, 5]) set in Table 1.

Fig.8(a) shows the noise voltage (the top trace) measured at one channel of the noise generator. The signal fell in [1, 2](V), corresponding to a numerical range of [-1, 1] in software simulation. As the noise signal was sent into the neuron H0 with w40=3V and AH0=1.9V, the measured neuron output, in response to all connected synapses having their VO sweeping between 1V and 2V, are shown in Fig.8(b). The neuron output (the upper trace) swept the sigmoidal curve periodically with VO (the lower trace), and the noise input perturbed the curve significantly. The continuous-valued stochastic behaviour of the neuron in accordance with Eq.(1) was clearly demonstrated.



(a) (b)

Fig. 8. The measured output of (a) one channel of noise voltage and (b) a stochastic neuron (upper trace) when its synapses had VO sweeping between 1.0 and 2.0 V (lower trace)



(a) (b)

Fig. 9. (a) The measured updating stepsize of 12mV with VP=2.46V, VN=0.57V, and a pulse width of 320ns. (b) The measured programming process of a parameter which was adapted from 2.25V towards 1.5V.

Fig.9(a) further shows the measured characteristic of the updating circuit in refreshing mode. With VP=2.46V, VN=0.57V, and a pulse width of 320ns for VPLS, an updating step of only 12mV was easily achieved for both incremental (INC/DEC=0) and decremental (INC/DEC=1) updates. The updating step could be further decreased by simply reducing the pulse width of VPLS, while the background noise of the oscilloscope and the switching noise made the updating step hardly visible. The programmability of parameter arrays was further tested by initialising a parameter to 2.25V and then adapting it towards 1.5V. Fig.9(b) shows the measured parameter voltage (the second trace from top) and corresponding digital control signals. The parameter value adapted from 2.25V to 1.5V within 336 sec. As soon as the target value was achieved, the directional signal (INC/DEC) started to alternate between 1 and 0, refreshing the parameter at 1.5 reliably.

6 Conclusion

The VLSI circuits realising a scalable and programmable CRBM system have been designed, fabricated and tested. The preliminary measurement results demonstrate

satisfactory functionality of the synapse module, the neuron module, and their programmable parameters. By interconnecting multiple chips, the capability of the system to model high-dimensional biomedical data, as well as the feasibility of using noise-induced stochastic behaviour to enhance the robustness of analogue computation will be further examined and discussed.

Acknowledgement

The authors would like to acknowledge the TSMC and CIC for fabrication of the chip, and the National Science Council (NSC) in Taiwan for funding this project (Grant code: NSC 95-2221-E-007-115).

References

1. Tong, B.T., Johannessen, E.A., Lei, W., Astaras, A., Ahmadian, M., Murray, A.F., Cooper, J.M., Beaumont, S.P., Flynn, B.W., Cumming, D.R.S.: Toward a miniature wireless integrated multisensor microsystem for industrial and biomedical applications. IEEE Sensors J. 2(6), 628–635 (2002)
2. Johannessen, E.A., Wang, L., Wyse, C., Cumming, D.R.S., Cooper, J.M.A.: Biocompatibility of a Lab-on-a-Pill Sensor in Artificial Gastrointestinal Environments. IEEE Transactions on Biomedical Engineering 53(11), 2333–2340 (2006)
3. Genov, R., Cauwenberghs, G.: Kerneltron: support vector machine in silicon. IEEE Trans. on Neural Networks 14(8), 1426–1433 (2003)
4. Hsu, D., Bridges, S., Figueroa, M., Diorio, C.: Adaptive Quantization and Density Estimation in Silicon. In: Advances in Neural Information Processing Systems, pp. 1083–1090. MIT Press, Cambridge (2002)
5. Chen, H., Murray, A.F.: Continuous restricted Boltzmann machine with an implementable training algorithm. IEE Proceedings-Vision Image and Signal Processing 150(3), 153–158 (2003)
6. Chen, H., Fleury, P.C.D., Murray, A.F.: Continuous-valued probabilistic behavior in a VLSI generative model. IEEE Trans. on Neural Networks 17(3), 755–770 (2006)
7. Chen, H., Fleury, P., Murray, A.F.: Minimizing Contrastive Divergence in Noisy, Mixed-mode VLSI Neurons. In: Advances in Neural Information Processing Systems (NIPS 2003). MIT Press, Camgridge (2004)
8. Lu, C.C., Hong, C.Y., Chen, H.: A Scalable and Programmable Architecture for the Continuous Restricted Boltzmann Machine in VLSI. In: IEEE International Symposium on Circuits and Systems (2007)
9. Cauwenberghs, G.: An Analog VLSI Recurrent Neural Network Learning a Continuous-Time Trajectory. IEEE Transactions on Neural Networks 7(2), 346–361 (2003)
10. Liu, S.-C., Kramer, J., Indiveri, G., Delbriick, T., Douglas, R.: Analog VLSI: Circuits and Principles. MIT Press, MA (2002)
11. Al-Sarawi, S.F.: A novel linear resistor utilizing MOS transistors with identical sizes and one controlling voltage. Microelectronics Journal 33(12), 1059–1069 (2002)
12. Cauwenberghs, G.: Delta-sigma cellular automata for analog VLSI random vector generation. IEEE Transaction on Circuit and System II: Analog and Digital Signal Processing 46(3), 240–250 (1999)
13. Diotalevi, F., Valle, M., Bo, G.M., Biglieri, E., Caviglia, D.D.: Analog CMOS current mode neural primitives. Circuit and System II: Analog and Digital Signal Processing. In: IEEE International Symposium on Circuits and Systems (2000)