International Journal of Engineering Research and General Science Volume 3, Issue 3, May-June, 2015
ISSN 2091-2730

VLSI Modeling of Neural Pulse-Based Computations

1. Shyamal Kumar Roy

Assistant Professor, Department of Electrical Engineering, Guru Nanak Institute of Technology,

157/F, Nilgunj Road, Panihati, Kolkata – 700114, West Bengal, India.

Email: shyamal50@yahoo.com

Contact no: +91-833-480-8086

2. Suman Majumder

Assistant Professor, Department of Electrical Engineering, Guru Nanak Institute of Technology,

157/F, Nilgunj Road, Panihati, Kolkata – 700114, West Bengal, India.

Email: suman.armature@gmail.com

Contact no: +91-980-418-8980

Abstract— With the growing use of VLSI based neural networks and the development of spike-based multi-chip systems the design of spike-based learning algorithms and circuits compatible with existing solutions are becoming more important. This article clarifies some basic ideas and provides a common language for pulsed spiking neural systems and discusses pulse encoding and the relationship among various simplified models of spiking neurons in design and analysis of pulsed neural systems in VLSI.

Keywords— VLSI, Hodgkin-Huxley model, Pulse Encoding, Pulse Arithmetic, Pulse Generation, Pulse Neuron

I INTRODUCTION

Very Large-Scale Integration (VLSI) program has potential in building massively parallel arrays of interconnected neurons as tens of thousands of synapses integrated on a single chip. A pulse stream encoding technique performs analogue multiplication under digital control. This approach lends itself naturally to continuous, asynchronous computation, conceptually simple, but theoretically rich feedback networks introduced by Hopfield [1,5]. It is possible to use the same building blocks in a single or multiple layer feed forwarded networks and other neural architectures.

There are many schemes for the use of spike timing information in neural computation. The integrate-and-fire model is a very commonly used algorithm in neural networks in spiking neurons. This model is simple to understand and implement, however, as it approximates the very detailed Hodgkin-Huxley model it can capture generic properties of neural activity [3,4]. We can, therefore, form and characterize these by their firing times $t_i^{(f)}$. The lower index *i* indicates the neuron, the upper index *f* the number of the spike. We can then describe the spike-train as a function of neuron,

$$\mathbf{F_i = \{ (1) ,..., (n)\}}$$

The variable *ui* is used to refer the membrane potential, or internal state of a neuron *i*. If a neuron's membrane potential crosses threshold value ϑ as shown in Fig. 1, it generates a spike. Thus, we add the time of this event to F_i, defining as

$$\boldsymbol{Fi = \{ t \mid ui\,(t) = \vartheta \wedge u'(t) > 0 \}}$$

When a neuron generates an action potential, the membrane potential suddenly increases, followed by a long lasting negative after-potential (see fig. 1b). This sharp rise above the threshold value makes it is absolutely impossible for the neuron to generate another spike and is named absolute refractoriness.

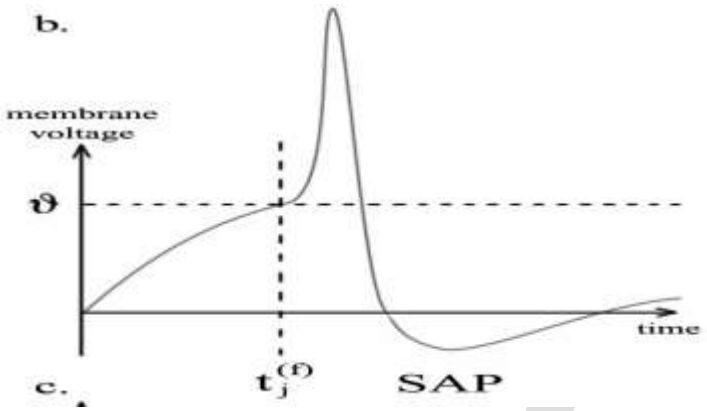

Fig 1: Incoming postsynaptic potentials alter the membrane voltage so it crosses threshold value ϑ; the neuron spikes and goes into a refractory state.

II PULSE ENCODING OF INFORMATION

Pulse encoding of electronic information is not a new idea. Communications systems have used pulse amplitude modulation, pulse width modulation and pulse code modulation for data transmission for some time. Horowitz and Hill [1,10] present a condensed review of pulsed techniques in communication. Pulse stream encoding was first used and reported in the context of neural integration in 1987 [6,9]. The process of pulse computations based down digital signals to carry information and control analogue circuitry, while storing further analogue information on the time axis is described below. There are different techniques available for coding a neural state in the range 0 < Vi < 1 on to a pulsed waveform Vi with frequency vi, amplitude Ai and pulse width δi. A representative selection of these is illustrated in Fig. 2, where a time varying analogue state signal Vi has been encoded in each of the following ways: pulse amplitude modulation, pulse width modulation, pulse frequency modulation, and pulse phase modulation.

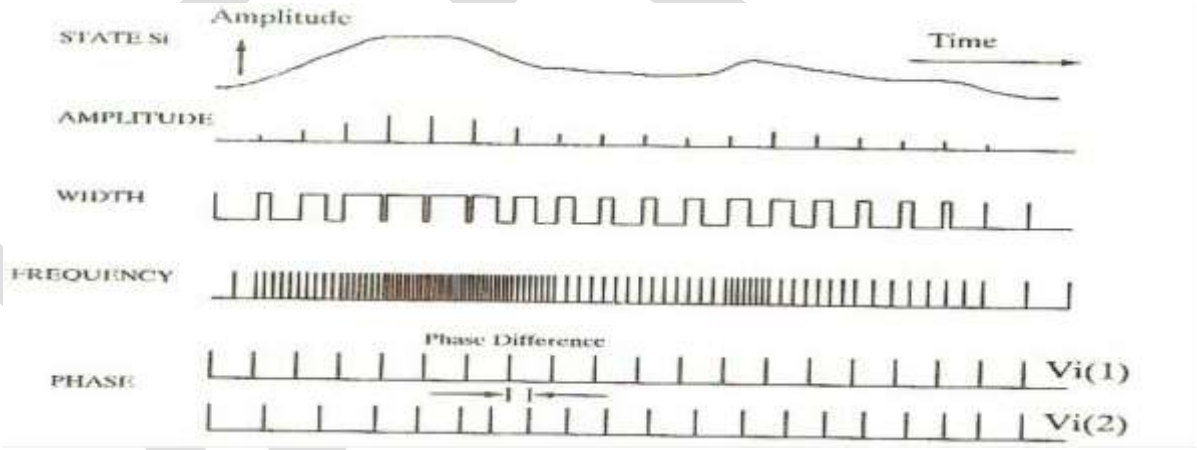

Fig.2 Methods for encoding a time-varying analogue neural state onto a pulsed signal

Pulse width, pulse frequency, pulse phase and pulse density modulation all encode information in the time domain, and can be viewed as variants of pulse rate.

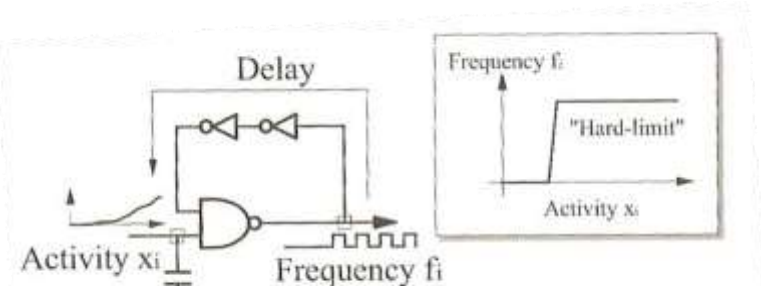

Fig. 3 Pulse Neuron

III. PULSE GENERATION IN VLSI

Fig. 3 describes the simplest possible form of pulsed neuron that can be generated on silicon chip.

When the activity voltage is 0 then NAND gate output is 1. So, ring is broken and there is no oscillation. Again, when voltage 1 then threshold is determined by the MOSFET NAND gate and output will oscillate at a frequency determined by the ring. It becomes so simple that the performance is not elegant. It makes the form hard limit neuron, where the output pulse frequency 0 maximum.

Pulse Arithmetic in VLSI

Addition and Multiplication are essential to the evaluation of $\sum TijVj$ in a neural network. These are the well defined functions. In analogue and pulse systems this may be implemented in details in several ways. Fig. 4 describes two different generic approaches in addition of pulsed and weighted in a natural state, using a frequency state signal Vi as an example.

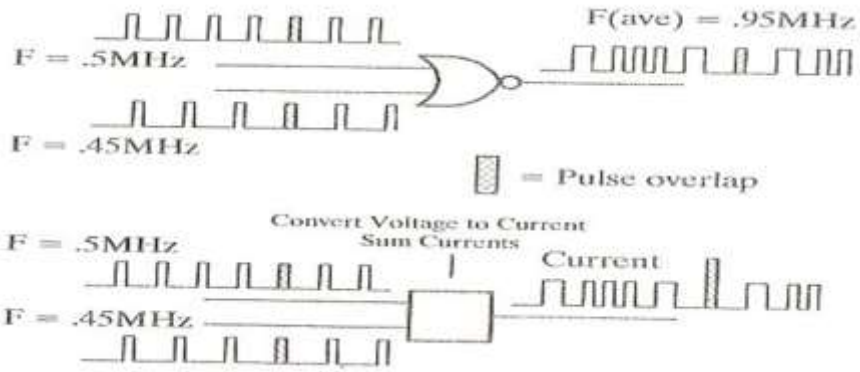

Fig. 4 Addition of pulses

Voltage pulse addition technique is based on the declaration that, if the frequency of a series of fixed width pulses perform a logical OR between two uncorrelated pulse streams is equivalent to addition of the signals.

Multiplication of pulse signals

When pulse frequency vj is used to encode presynaptic neural state Vj & vj can be compressed in time domain to represent multiplication. For example when presynaptic pulse width is Dtj after passing through a synaptic weight Tij< 1, the multiplication of pulse signals is Tij x Dtj. Multiplication of the postsynaptic pulses of a width in time proportional to the synaptic weight, and at a frequency controlled by the presynaptic neural state. So they would be accumulated in time by either of the methods described above.

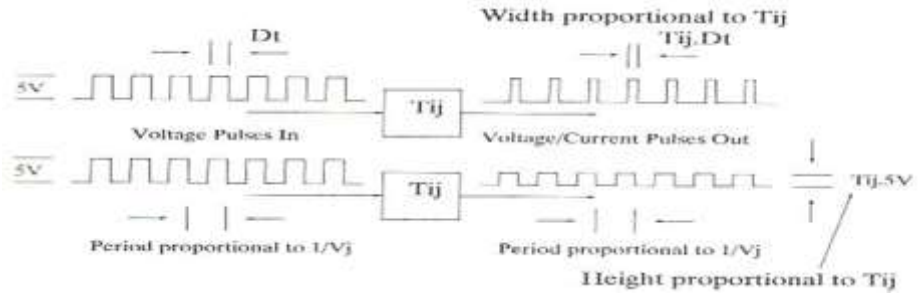

Fig. 5 Multiplication pulse width modulation and height modulation

IV. SPIKING NEURONS IN HARDWARE

The VLSI technology integrates many powerful features into a small microchip like a microprocessor. Such systems can use data representations of either binary (digital VLSI) or continuous (analog VLSI) voltages. Progress in digital technology has been tremendous, providing us with ever faster, more precise and smaller equipment. In digital systems an energy-hungry synchronization clock makes it certain that parts are ready for action. Analog systems consume much less power and space on silicon than digital systems (in many orders of magnitude) and are easily interfaced with the analog real world. However, their design is hard, due to noise computation. Fundamentally (slightly) inaccurate and sufficiently reliable non-volatile analog memory does not exist. Noise is influenced by random effects that affects *everything* in the real world that operates in normal (so, above the absolute zero) working environment temperatures. For digital systems this is not much of a problem, as extra precision can be acquired by using more bits for more precise data encoding. In analog systems such a simple counter-measure is not at hard. There are no practical ways of eliminating noise; at normal temperatures noise has to be accepted as a fact of life. Our brain is a perfect example of an analog system that operates quite well with noise, like neural networks do in general. In fact, performance of neural networks increases with noise level [11]. Spiking neuron models can easily be equipped with noise-models like noisy threshold, reset or integration. The interested reader can find more details on the modeling of noise in spiking neurons in Gerstner's excellent review on neuron models [8]. Hybrid systems can provide a possibly of perfect solution, operating with reliable digital communication and memory while using fast, reliable and cheap analog computation and interfacing. In such a solution, neurons can send short digital pulses, much like we've seen before in the integrate-and-fire model. This model can be implemented in VLSI systems quite well [2]. VLSI systems usually work in parallel, a very welcoming fact for simulation of neural systems, which are inherently massively parallel. A significant speed gain can be acquired by using a continuous hardware solution; by definition digital simulation which will have to recalculate each time-slice iteratively [20]. Though computer simulations have an advantage in adaptability, scaling, a network up to more neurons (1000+) often means leaving the domain of real-time simulation. VLSI systems can be specifically designed to be able to link up, easily forming a scalable set-up that consists of many parts operating like they are one big system [2, 11, 20].

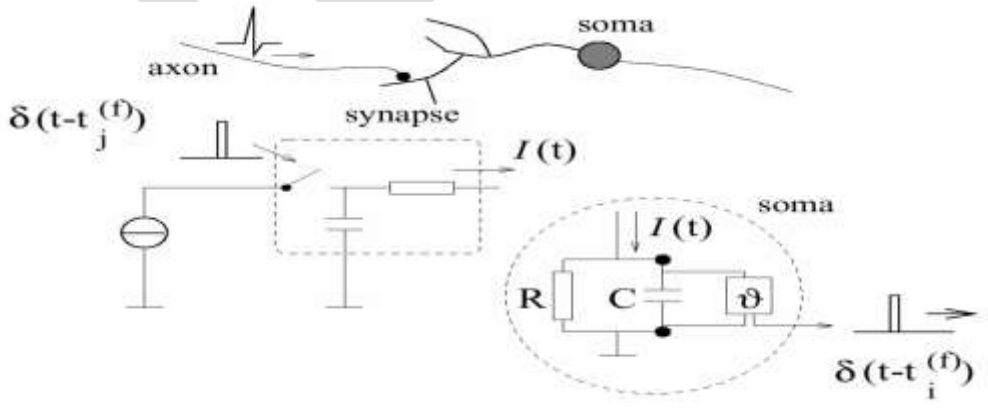

Fig. 5 Schematic of the integrate-and-fire neuron. Low-pass filter (left) that transforms a spike to a current pulse *I(t)* that charges the capacitor. Schematic version of the soma (right) which generates a spike when voltage *u* over the capacitor crosses threshold [10].

V. CONCLUSION

VSLI based neural pulse based computation modeling has advantageous over other modeling tools. VLSI systems are very powerful and capable of producing good results in presence of noise. The adaptability of VLSI systems in different environment and interfacing with other neural model makes this model multi-tasking.

REFERENCES:

[1] Christodoulou C., Bugmann, G. & Clarkson, T. G. *A spiking neuron model: applications and learning*, Neural Networks, in press (2002).

[2] Elias, J. G. & Northmore, D. P. M. *Building Silicon Nervous Systems with Dendritic Tree Neuromorphs* in Maass, W. & Bishop, C. M. (eds.) *Pulsed Neural Networks*, MIT-press (1999).

[3] Kempter, G. & van Hemmen, J. L. *Hebbian Learning and Spiking Neurons*, Physical Review E4, p. 4498-4514 (1999).

[4] Gerstner, W. *Spiking Neurons* in Maass, W. & Bishop, C. M. (eds.) *Pulsed Neural Networks*, MIT-press (1999).

[5] Horn, D. & Opher, I. *Collective Excitation Phenomena and Their Applications* in Maass, W. & Bishop, C. M. (eds.) *Pulsed Neural Networks*, MIT-press (1999).

[6] Murray, A. F. *Pulse-Based Computation in VLSI Neural Networks* in Maass, W. & Bishop, C. M. (eds.) *Pulsed Neural Networks*, MIT-press (1999).

[7] Bernd Richter, „*Development of a readout circuit for pulse-coupled artificial neural networks*," Diploma thesis, University of Technology Dresden, Endowed Chair for Neural Circuits and Parallel VLSI-Systems, June 2003.

[8] Christopher M. Bishop, "*Pulsed Neural Networks",* A Bradford book, the MIT Press Cambridge, Massachusetts, London, England.

[9] Allan Murray, Lionel Tarassenko, "*A Pulse Stream Approach",* Chapman & Hill, London, New York, Tokyo, Melbourn.

[10] David Zhang, "*Neural System Design*", Springer Verlag Singapore Pte. Ltd. 1999

[11] Jose G. Delgado, "*VLSI for Neural Networks and Artifical Intelligence*", Plenum Press, New York and London, 1992.

[12] J. Schemmel, K. Meier, and E. Mueller, "*A New VLSI Model of Neural Microcircuits Including Spike Time Dependent Plasticity*," Proceedings 2004 IEEE International Joint Conference on Neural Networks, vol. 3, pp. 1711 - 1716, July 2004