Silicon synapse designs for VLSI neuromorphic platform

Nguyen Duc Bui Phong[1], Masoud Daneshtalab[1,2], Sergei Dytckov[1], Juha Plosila[1], Hannu Tenhunen[1,2]
[1]Department of Information Technology, University of Turku, Finland
[2]Department of Electronic Systems, Royal Institute of Technology, Sweden

Abstract— Analog silicon neurons were proven to be a promising solution for VLSI neuromorphic platform to implement massively scalable computing systems. They possess the advantages of consuming less power and silicon area than digitally designed neurons. This paper compares the differences in power and area consumption between two methods of synapse design for analog neuron models: time-based modulation and current-based modulation. The obtained results demonstrate that under the same technology process (ST CMOS 65nm), the neuron that uses time-based modulation consumes less power (almost six times) and silicon area (about thirty times) but higher energy (twelve times) than that of the current-based modulation.

I. Introduction

Since being adopted, spiking neural networks have shown to be an emerging topic in academic scholarly. The system is well-proven by a wide range of applications across multiple research areas [1][2][8][10], for instance understanding its impact in the neuroscience field, or in the case of the neuromorphic engineering field, constructing of computational systems that mimic the brain. As a result of the popularity of the spiking neural network, various simulators, including both strategies and software tools, have been carried out in the effort to simulate those networks [3]. Although such simulators would theoretically work well to investigate or verify neural networks' behavior, one of the major issues is there is no system powerful enough to capture the real-time behavior of some cerebral cortex regions with multiple characteristics of different classifications of neuron cells without taking in a huge amount of resources. In response to the obstacle created by the lack of scalable operations of those simulators, custom digital systems are proposed as a solution. Despite being rather limited to small- and medium-sized networks and having the bottleneck in memory interconnection, the capability of digital systems includes taking advantage of the speedup from GPUs and FPGAs or the energy and area saving aspect of the ASIC devices. In spite of that it is not verified that whether such a system could be able to qualify for the energy consumption, area overhead and robustness, which are the key parameters for modelling the neurons. On the other hand, there is the silicon neuron, a type of hybrid analog/digital very large scale integration (VLSI) circuit, having the ability to accelerate the function of the hardware emulation in terms of power, silicon area and speed. Instead of manipulating the simulation on normal computers, silicon neurons allow the hardware to emulate directly on its network. This fits well with the concept of neuromorphic, which was first introduced by C. Mead in "Neuromorphic electronic systems" in 1990 [12]. Neuromorphic refers to an "artificial neural system" which is organized and functions with high similarity to the operation of a biological nervous system. By using silicon neurons as the basic building blocks in a neural network, we can achieve a scalable concurrent system with promising area-wise and energy-wise aspects [15].

In this paper, two design methods of the analog neuron are presented and compared. Time-based modulation and current-based modulation are presented for the Leaky Integrate and Fire neuron model. Both implementation methods of analog neuron designs are compared based on speed, power and area through circuit-level simulation and layout estimation under the ST CMOS 65nm technology process.

II. Motivation

Digital neurons may provide faster time and more precision outcomes but analog neurons, on the other hand, had been proven to be much more power efficient and much less area overhead than that of digital neurons [11]. However; the shortcomings of analog neurons lie in their difficulty in designing and inaccuracy which is caused by noise. At the system level, the reason noise could have influence on analog systems is because the systems themselves do not have any mean to remove those random effects and such influence by noise could not be shaded away but rather being accepted as a part of the operation. Digital systems, as developed as they are, could totally remove the noise influence by utilizing extra bits in order to achieve more accuracy in calculation. As a matter of fact, a highly desirable neural network is a network that can perform well with the existence of noise and the noise models can be integrated into the spiking neuron without much hassle. In such a situation, the hybrid system could be considered as one of the possible alternatives. Such a system is a combination of both digital communication and analog computation which shares the integrated memory, process and interface. In this case, the noise could be suppressed at each individual neuron circuit while on the digital domain noise is less likely an issue. Furthermore, VLSI hybrid system can be used to implement neural networks (in terms of transmitting digital signal pulses, a representative of a neuron's spikes, throughout the network) [4]. As a result, the system will be similar to that of a neural system where the operations of each distinguish neuron in such a system could happen simultaneously.

Different implementation methods may bring different pros and cons of each neuron as an individual as well as the system as a whole. Thus, it is in within the scope of this paper that the keys parameters of each design of the neuron will be explored and analyzed to pave the road to build such a neuromorphic system. Consequently, a massively scalable system can be achieved by designing each of the unit components separately and then connecting them together to form one consistent system [4][7][13].

978-1-4799-6890-9/14/$31.00 ©2014 IEEE

III. LEAKY INTEGRATE AND FIRE MODEL (LIF)

A regular neuron unit has three main parts which are differentiated with each other by their operation and mission. These parts include: the soma, the dendrite and the axon. Soma – also known as cell body – Is the largest part and is considered as the central control unit of each neuron where most of the primary operation activities of a neuron take place, for instance the integration of membrane potential or the firing of spikes. The signal receiver is the second part which is normally called dendrites. As the name says, the dendrites' main function is to receive incoming signals which are initiated or transmitted from other neurons. The last part is axon of which main function is to transmit the outgoing signal from corresponding neuron to another neuron. Normally, axon situates at the axon terminals which are also at the end of their respective neuron. It is noted that each neuron may have many different types of dendrite but could only have one single axon. The axon of one neuron is connected to multiple receiving dendrites of other neurons and out of these connections a neural network is effectively formed. In other words, the operation of each neuron allows that neuron to generate only one outgoing signal whilst receiving multiple incoming signals from many other different neurons in the same network. There is another element which is not counted as one of the three parts above is the synapse which handles all transmitting and receiving of information between one neuron and the others. The synapses take the role as the linkage between the transmitting part of the neuron, the axon, and the receiving part of another neuron, the dendrite or the soma.

The neuron model which is used here is the LIF [6][9]. Practically, the LIF model is easier to implement in hardware and it is also simple enough so that if there are problems, they can be easily tracked down and "troubleshot".

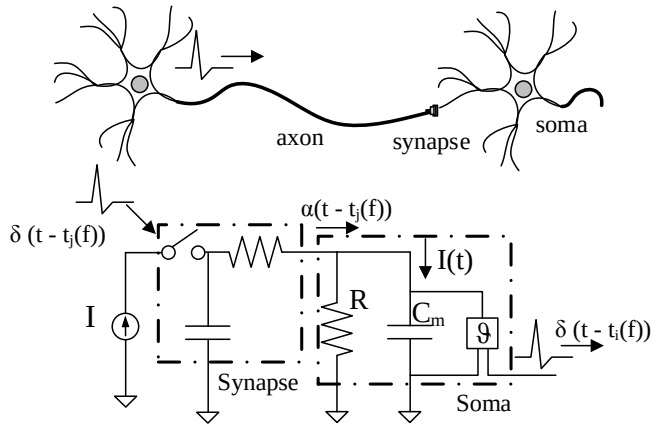

Fig. 1. Schematic diagram of the Leaky Integrate and Fire model [6](Gerstner W. & Kistler W. M., 2002)

Fig. 1 shows the schematic diagram of the basic Leaky Integrate and Fire model. Inside the dashed circle on the right side of Fig. 1 is the soma circuit. On the left side of Fig. 1, the synapse which contains the low pass filter circuit acts as the connection between the input spike $\delta\ (t - t_j(f))$ and the soma. The input spike is filtered out and a current $\alpha(t - t_j(f))$ is the result of the process and gets transferred to the soma. The current $I(t)$ charges the RC circuit thus changing the membrane potential of the neuron. The membrane potential of the neuron, presented as voltage different between the anode and cathode of the capacitor $u(t)$ is then compared against a threshold of value ϑ. There are two cases as to what happens: if $u(t)$ reaches ϑ at time $t_i(f)$, the neuron will generate (or fire) an output pulse of $\delta(t - t_i(f))$, otherwise if $u(t)$ is smaller than ϑ, no output spike will be generated. There are also descriptions of how the "integrate" part and the "leaky" part of the neuron works in the schematic diagram. The capacitor is the main storage of the membrane potential of the neuron which is the integration of input current $I(t)$ while the resistor, in parallel with the capacitor, is where the leakage of membrane potential takes place. Although the firing (a spike) event and the operation sequences are not specifically presented in the schematic diagram, it should be noted that as soon as the membrane potential $u(t)$ reach the threshold ϑ, the membrane potential of the neuron will be reset to the reset potential ur instantly and the next integration begins with the initial value of the membrane potential of ur. The LIF neuron model can be expressed as the leaky integration of the input spike and its membrane potential over time:

$$\tau_m\ (du/dt) = -u(t) + RI(t)$$

Where on the right hand side of the equation, $u(t)$ is the membrane potential of the neuron at time t, R is the membrane resistance, $I(t)$ is the current which is generated from the synaptic connection of the synapse charges the capacitor, the representative of the membrane potential. The element on the left side of the equation, τ_m, is the membrane time constant calculated by the multiplication of membrane resistance and membrane capacitance ($\tau_m = R * C$).

To have a quick view of how silicon neurons operate one can think of them as circuits consisting of at least one synapse block, a leaky integrator block and a comparator block. The synapse block functions as both the dendrites and synaptic connection of the neuron while the leaky integrator block and the comparator block functionalities are similar to that of the neuron's soma as presented in Fig. 2.

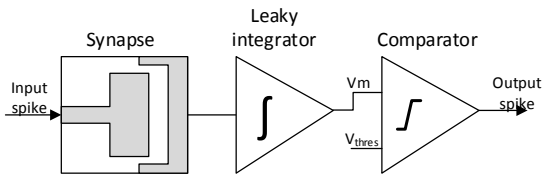

Fig. 2 block diagram of the Leaky Integrate and Fire neuron

IV. ANALOG DESIGN APPROACHES OF THE LIF NEURON

For the analog implementation, current injection method is utilized for the weight injection where the current I_{weight} representing the weight of the input spike will be injected to the membrane capacitor C_m. The differences between two methods of current injection will be explored through the operating principle and the results obtained through power and area simulation. The synapse, which consists of a digital to analog converter and a weight injection circuit, is the only differences between the two methods of analog designs for the LIF neuron. The remaining components of the neuron model, the leaky integrator and the comparator, are the same for both designs.

A. *The Leaky Integrator component of LIF neuron*

The incoming spike excites or inhibits a neuron toward or away from firing a spike while the membrane potential of the neuron slowly decreases over time. This process can be interpreted as a leaky integrator where the synaptic connection charges or discharges a capacitor *Cm* storing the model's membrane potential V_m that integrates the incoming spike and its weight overtime while the internal potential of the integrator continuously dissipates by the present of a leakage current I_{leak}. Furthermore, when the membrane potential V_m of the integrator reaches the threshold $V_{threshold}$, the membrane potential V_m gets immediately pulled down to initial potential V_{reset} (or reset voltage).

The leaking rate of the LIF model is required to be customizable to some extent; this is done by replacing the leaking resistor with a leaking current source I_{leak} or leaking voltage source V_{leak} controlled by a digital input. The principle of the leaky circuitry remarkably bears a resemblance of a digital to analog converter, where the resolution of the converter is the resolution of the leaking rate which is digitally encoded. Fig. 3 demonstrates the leakage current I_{leak} or leakage voltage V_{leak} discharging from the membrane capacitor C_m.

The n-channel MOSFET controlled by the input signal "leak" will determine whether or not the leakage happens. Another n-channel MOSFET is put into place to control the reset mechanism of the neuron model by a "reset" signal, when closed will initialize V_m to V_{reset}.

The charging curve of is dependent on the membrane time constant ($\tau_m = R * C$) which in turns depends on the membrane capacitance itself, expressed by the equation:

$$V_m = \tau_m * (1 - exp(-t/\tau_m))$$

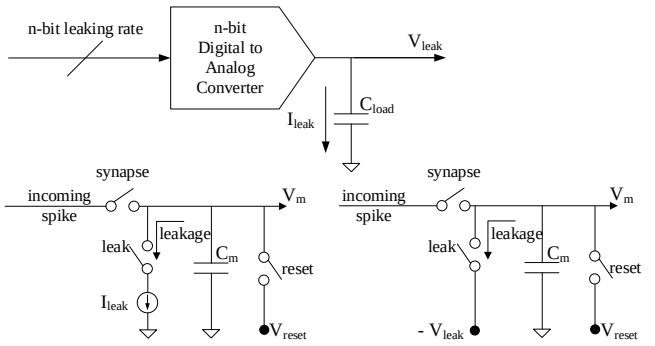

Fig. 3. schematic diagram of the leaky integrator with necessary signals and components

B. *The comparator component of LIF neuron*

The comparator block receives the internal potential V_m as input voltage and compares it with the threshold voltage which is treated at the reference voltage. If the input voltage is larger than the reference voltage, the comparator output will be active.

The comparator is one of the most critical building blocks since the speed and accuracy of the comparator is the speed and accuracy of the design model itself. For this design, the comparator should be designed to be sufficiently fast, accurate and low area and power consumption. Hence the latched comparator is chosen where the operation is dependent on a clock signal as illustrated in Fig. 4.

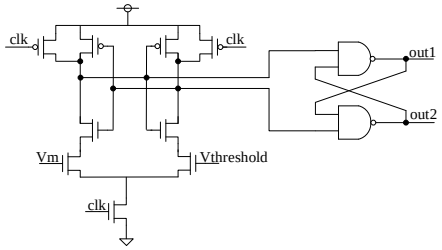

Fig. 4. schematic diagram of the latched comparator

C. *The Synapse component of LIF neuron*

A spike generated by a neuron can be transferred through the synapse. From the analog circuit point of view, the function of the synapse is taking in the input spikes and weights and converting them into an electrical potential then transferring it to the soma of the neuron. The synapse can then be interpreted as a digital to analog converter that converts the digitally coded weight of a spike to the analog value of the weight, either as a current value or a voltage value. The resolution of the digital to analog converter is equal to the resolution of the weight of the input spike while the amplitude of current or voltage at the output node of the digital to analog converter represents the weight of the input spike.The n-bit Digital to Analog Converter will convert the n-bit weight value (which is digital) of the incoming spike to an analog current value of I_{weight} or a voltage value of V_{weight}. After that, the converted weight value will be injected to the soma via the weight injection circuit. The neuron designs in this paper will have the weight resolution of 7 bits, which makes the maximum weight reach 127 unit weights. In the weight injection circuit, the excitatory input spike can be realized by a positive current or voltage being applied to the load and vice versa for the inhibitory input spike, a negative current or voltage is applied to the load, as in Fig. 5.

Fig. 5A and Fig. 5B show the incoming spike is presented as the input signal "spikein" and the membrane potential of the neuron will be stored in the capacitor C_m. The input signal "spikein" controls a CMOS Inverter (logic NOT gate) which consists of two transistors, the p-channel MOSFET M1 and the n-channel MOSFET M2. In the case of excitatory incoming spike, the "spikein" signal will be LOW, leading to the p-channel MOSFET M1 get closed (or M1 is conductive) and the n-channel MOSFET M2 get opened (there is no current flowing through M2), then the capacitor C_m will be charged with an inflow current of value I_{weight} or a positive voltage of value $+V_{weight}$. On the other hand, the inhibitory input weight can be easily achieved by changing the input signal "spikein" to HIGH, the capacitor Cm will be discharges by an outflow current of value I_{weight} or a negative voltage of value $-V_{weight}$.

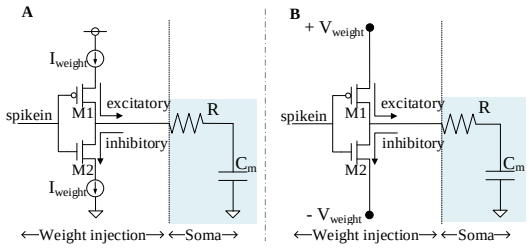

Fig. 5. basic schematic diagram of the weight injection

1) Time-based modulation synapse

This method for the synapse is based on the principle of the simple yet effective pulse-width modulation digital to analog converter such that the amplitude of the input signal will be expressed as the percentage of time which the pulse (pulse width) stays as "High" state in one clock cycle (duty cycle) as presented in Fig. 6. To further reduce the power, another technique called step wise charging [5] is applied, basically, by dividing the width of one pulse into smaller equal width steps, which is illustrated in Fig. 7. Since basing on the step wise modulation principle, the elementary circuits of the time-based synapse consist of one pulse width modulation digital to analog converter and the step wise weight injection circuit as shown in Fig. 8.

The injection period for one input spike may reach up to 127 clock cycles (one clock cycle for one unit weight) if the input spike have maximum weight.

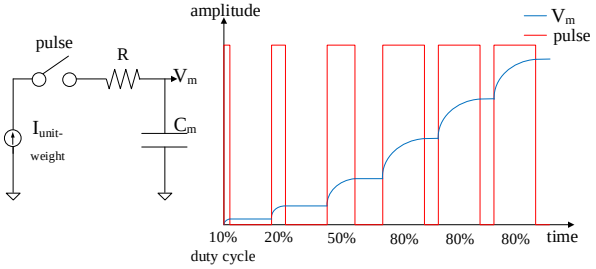

Fig. 6. the weight of the input spike is modulated as the width of the pulse that controls the injecting period. The longer the switch is closed, the larger the amount of current getting injected to the capacitor

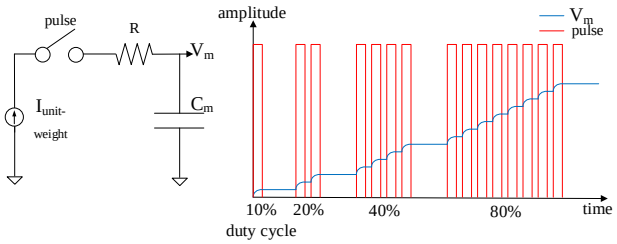

Fig. 7. step wise pulse width modulation waveform, the improved version of PWM with the pulses got divided into equally sized smaller pulse

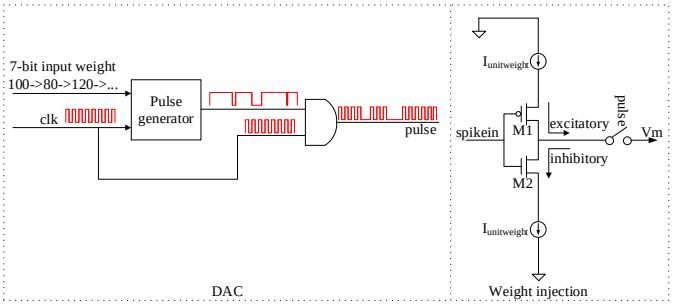

Fig. 8. time-based modulation synapse. Signal spikein determines if the input weight is positive or negative. The weight of the input spike is realized by using the switches controlled by the pulses generated by a pulse generator

2) Current-based modulation

This method is based on the principle of the binary weighted digital to analog converter which is more straightforward as the amplitude of the inputs signals is represents as the amplitude of the current injecting to the capacitor.

The major difference between the current-based modulation synapse and the time-based modulation synapse which had been addressed during the design step is that the current-based modulation consumes a very large amount of silicon area since the transistors have to be sufficiently large to house the extra current flowing through them. On the bright side, the injecting period of one input spike regardless of its weight costs only two clock cycles for the current-based modulation neuron model; it makes a difference in speed when comparing with the injecting period of 127 clock cycles for the time-based modulation neuron model.

Being originated from the binary weighted modulation, the time-based synapse circuit includes one pulse width modulation digital to analog converter and the step wise weight injection circuit as shown in Fig. 9.

Since the clock frequency is high and the maximum value of I_{weight} can reach up to 127 times the unit weight current $I_{unitweight}$, the weight switches (*w(0), w(1)*, …, *w(6)*) alone are not enough to handle the task of injecting a large current to the membrane capacitor.. Thus, another switch is put into the weight injection circuit and controlled by the signal "inject". Staying closed for a sufficient amount of clock cycles, the weight switches will ensure that the injecting weight current I_{weight} reaches the stable value while the inject switch will do the actual injecting in one clock cycle as illustrated in Fig. 10.

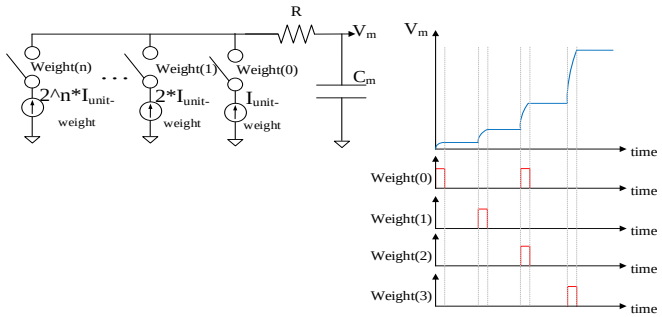

Fig. 9. binary weighted waveforms, the input sequence is as follow: 0001, 0010, 0101, and 1000

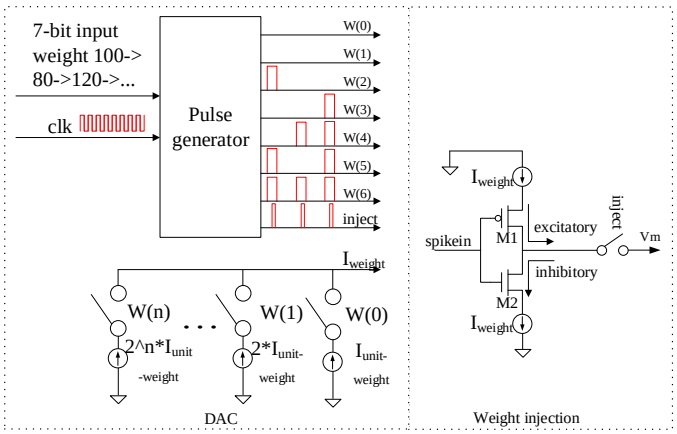

Fig. 10. current-based modulation synapse

V. EXPERIMENTAL RESULTS

In this work, the circuits are designed using Cadence's icfb and simulated using Spectre simulator using ST CMOS065 (65nm).The desired goal is for the neuron models to operate with a 500MHz clock with minimal power and area consumption. The weight has the resolution of 7 bits, which makes the maximum weight reach 127 unit weights.

Both design models have the maximum threshold of one maximum weight, which are 127 unit weights. This threshold means that the neuron models will fire as soon as their membrane potential reach the voltage value equivalent to that of 127 unit weights (after resetting: 127 pulses for time-based modulation model and $127*I_{unitweight}$ for current-based modulation model). The leaking rate is fixed at a tenth of $I_{unitweight}$ ($I_{leak} = 1/10 * I_{unitweight}$).

A. Time-based modulation neuron model

In Fig. 11, after the membrane potential being initialized to the reset voltage, 127 pulses were injected by one excitatory input spike during a 254 nanosecond-long injecting period. During the injecting period, the membrane potential V_m raises until V_m meet the threshold voltage and then a spikeout is fired.

Poly-NW capacitor is chosen for area saving purpose since this type of capacitor has relative high capacitance density. In total, the neuron model that uses time-based modulation method costs a total of 174 um^2. The layout is presented in Fig. 12.

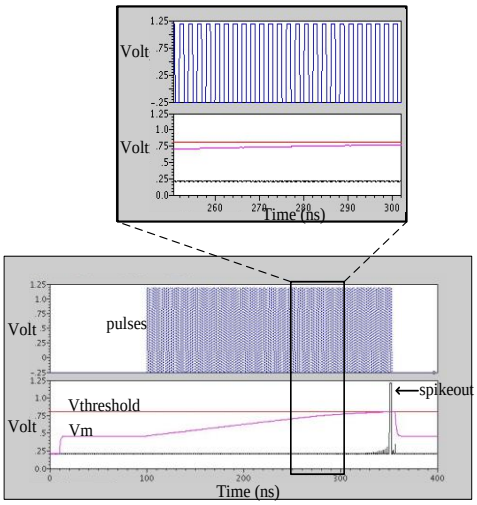

Fig. 11. waveforms from the simulation of time-based modulation neuron model

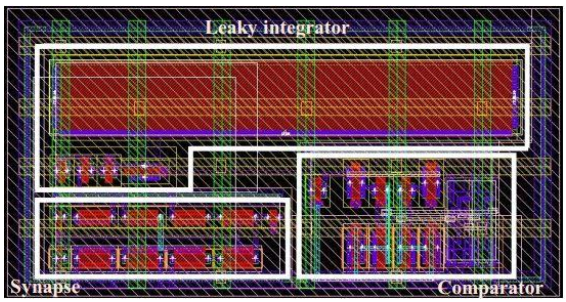

Fig. 12. layout of time-based modulation neuron model

B. Current-based modulation neuron model

In Fig. 13, between time t = 60ns and t = 70ns, all weight switches are closed, ensuring a stable current of $I_{weight} = 127 * I_{unitweight}$. During that time, an injecting period of four nanoseconds is realized by the inject switch, effectively inject the current I_{weight} into the membrane capacitor which raises the membrane potential and then a spikeout is fired when the membrane potential is higher than the threshold voltage.

The current-based modulation neuron model is estimated to cost 5000 um^2 of silicon where the major part (more than 90%) of the neuron circuit is the synapse.

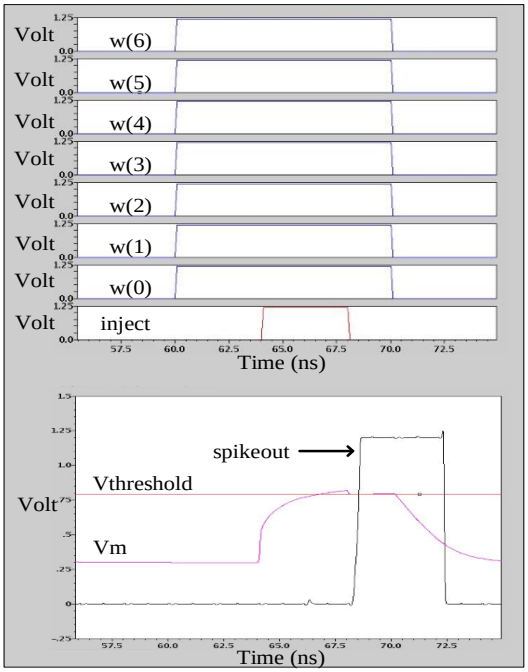

Fig. 13. waveforms from the simulation of current-based modulation neuron model

C. Results and Comparison

Table I gives the average power consumption of both methods by each component as well as the total power values of both models. In the case of time-based modulation, the power hungry component is the comparator which accounts for three forth of the total power consumption while the leaky integrator consumes little to no power. Consequently, it is advisable to use multiple leaky integrators and/or synapse while utilize one comparator when there is a need for designing a cluster of neurons arises.

TABLE I. AVERAGE POWER CONSUMPTION FOR EACH COMPONENT OF BOTH NEURON MODELS

	Synapse (DAC + weight injection)	**Leaky Integrator**	**Comp - arator**	**Total**
Time-based	2.7uW	0.21uW	8.8uW	11.7uW
Current-based	53.6uW	0.83uW	9.1uW	63.5uW

TABLE II. STATISTICS OF THE TWO NEURON MODELS

	Total Average Power	Leakage Power	Time per spike	Total Energy	Total Area
Time-based	11.73uW	1.8uW	254ns	2.98pJ	174um2
Current-based	63.5uW	2uW	4ns	0.25pJ	5000um2

In Table II, for each neuron model, the total energy is calculated by multiplying the total power (average power) with the time it takes to fire a spike. The experiment shows that the neuron circuit that uses current-based modulation method only needs a short amount of time compared with the one using time-based modulation method (4ns versus 254ns). Hence, despite the higher power consumption, the total energy of the current-based modulation model is much less than that of the time-based modulation model. On top of that, when there is no circuit activity the leakage powers of both neuron models are the same. It could also be reduced as both neuron circuits would use up the same amount of energy when they are both idle. The silicon area result from the time-based variant is obtained by doing the layout in full custom flow. The layout includes the synapse (which contains DAC and weight injection), the leaky integrator and the comparator. The silicon area of the current-based neuron circuit (also contains the same building blocks as the time-based modulation neuron circuit), however, is based on the estimation of the floorplan.

Other factors that may affect the power consumption and silicon area of the neuron circuits are the threshold value, the weight resolution and the clock speed. Firstly, the larger threshold was experimented on and the consensus is that the higher the threshold, the larger the membrane capacitor needed and the more time it takes for the neuron model itself to fire a spike (assuming the membrane potential have to raise from the reset voltage to the threshold voltage) which in turn increase the energy dissipation and area overhead respectively. Secondly, a higher weight resolution may not affect the time-based modulation synapse circuit in terms of power or area but it will definitely affect the energy consumption since the time it takes to inject the weight value (*maximum weight = 2^weight_resolution - 1*). On the other hand, a higher weight resolution will definitely affect every statistics of the current-based modulation synapse circuit since a bigger current or a longer injection time will be needed. Finally, a different clock frequency will literally affect all of the statistics of both methods. It is vice versa for the neurons that have lower weight resolution. Works targeting at the weight resolution are ongoing while some of those show that at the lower weight resolution, SNN can still function appropriately.

VI. CONCLUSION

Two different designs of analog silicon neurons are brought out and compared. The current-based modulation method can operate quite effective in systems where performance is the first priority; however, the total energy per spike is worth considering in the long run. On the other hand, it is an optimal choice to build a massively scalable computing system with the time-based modulation neuron circuit as the core processing element due to the lower cost for silicon and less power consumption (heat dissipation will also less likely be an issue).

VII. ACKNOWLEDGEMENT

This work was supported by VINNOVA (Swedish Agency for Innovation Systems) within the CUBRIC and ERoT projects, and academy of Finland.

REFERENCES

[1] Ananthanarayanan, R., Esser, S. K., Simon, H. D., & Modha, D. S. (2009). The cat is out of the bag: Cortical simulations with 109 neurons, 1013 synapses. High Performance Computing Networking, Storage and Analysis, Proceedings of the Conference on, 1-12.

[2] Belhadj, B., Joubert, A., Li, Z., Héliot, R., & Temam, O. (2013). Continuous real-world inputs can open up alternative accelerator designs. Proceedings of the 40th Annual International Symposium on Computer Architecture, 1-12.

[3] Brette, R., Rudolph, M., Carnevale, T., Hines, M., Beeman, D., Bower, J. M., . . . Harris Jr, F. C. (2007). Simulation of networks of spiking neurons: A review of tools and strategies. Journal of Computational Neuroscience, 23(3), 349-398.

[4] Christodoulou, C., Bugmann, G., & Clarkson, T. G. (2002). A spiking neuron model: Applications and learning. Neural Networks, 15(7), 891-908.

[5] D.J. Mlynek, and Y. Leblebici, "Design of VLSI Systems", web-based advanced course (Chapter 7). http://www.vlsi.wpi.edu/webcourse/

[6] Gerstner, W., & Kistler, W. M. (2002). Spiking neuron models: Single neurons, populations, plasticity Cambridge university press.

[7] Horn, D., & Opher, I. (1999). Collective excitation phenomena and their applications. Pulsed Neural Networks, , 297-320.

[8] Indiveri, G. (2001). A neuromorphic VLSI device for implementing 2D selective attention systems. Neural Networks, IEEE Transactions on, 12(6), 1455-1463.

[9] Izhikevich, E. M. (2004). Which model to use for cortical spiking neurons? IEEE Transactions on Neural Networks, 15(5), 1063-1070.

[10] Izhikevich, E. M., & Edelman, G. M. (2008). Large-scale model of mammalian thalamocortical systems. Proceedings of the National Academy of Sciences of the United States of America, 105(9), 3593-3598. doi:10.1073/pnas.0712231105 [doi]

[11] Joubert, A., Belhadj, B., Temam, O., & Heliot, R. (2012). Hardware spiking neurons design: Analog or digital? Neural Networks (IJCNN), the 2012 International Joint Conference on, 1-5.

[12] Mead, C. (1990). Neuromorphic electronic systems. Proceedings of the IEEE, 78(10), 1629-1636.

[13] Murray, A. F. (1999). Pulse-based computation in VLSI neural networks. Pulsed Neural Networks, 87-109.

[14] Northmore, D. P., & Elias, J. G. (1998). Building silicon nervous systems with dendritic tree neuromorphs. Pulsed Neural Networks, , 135-156.

[15] Poon, C. S., & Zhou, K. (2011). Neuromorphic silicon neurons and large-scale neural networks: Challenges and opportunities. Frontiers in Neuroscience, 5, 108. doi:10.3389/fnins.2011.00108 [doi]