31st Annual International Conference of the IEEE EMBS
Minneapolis, Minnesota, USA, September 2-6, 2009

An Analog Circuit Implementation of a Quadratic Integrate and Fire Neuron

Eric J. Basham, *Student Member, IEEE*, David W. Parent, *Member, IEEE*

***Abstract*— Silicon neurons are of importance both to implement hybrid electronic-biological system as well as to develop fundamental understanding of the neurobiological systems they emulate. We have implemented a hardware version of the quadratic integrate and fire neural model. The quadratic integrate and fire neuron differs from the more common integrate and fire neuron in that the model, and thus the hardware, intrinsically generate spikes. Readily available discrete surface mount components are used to make the hardware available to a wider audience and facilitate experimentation.**

I. Introduction

IMPLEMENTATION of neuronal networks *in silico* can facilitate understanding of biological systems and result in hardware systems with similar performance to biological systems. An excellent review covering implementation of neural models *in silico* is presented in [1]. Software systems, while flexible and accurate, are often too slow to interface directly with neural tissue or to operate on experimentally collected data in real time. Physical circuit implementation has several advantages. First, circuits that are designed to perform specific operations typically operate much faster than general purpose hardware. One approach to developing specialized hardware is to develop computational models that are more easily implemented in hardware. Indeed, this was the motivation for the development of the Fitzhugh-Nagomo model [2], i.e. the simplification of the Hodgkin-Huxley active membrane equations allowed implementation of neural spiking behavior in analog circuitry as specialized type of analog computer.

Determining if a neural model is biologically relevant presents a significant challenge [3, 4]. One approach to verification is direct interface of the neural model to biological neural systems. The direct interface to simple biological systems has met with some success. In [5], the authors interfaced a multiconductance hardware neural model directly to a leech heart neuron via an electrophysiological instrumentation interface. Direct interface of hardware to neural tissue can also provide a means to advance prosthetics technology. In [6], the authors employ a VLSI (Very Large Scale Integration) integrate and fire (IF) *in silico* neural network to demonstrate the use of dexterous hand movement in simulation.

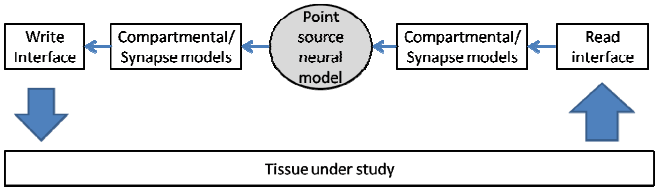

Fig 1. Components of a hardware-biological hybrid system. The point source neural model (shown in grey) is the focus of this work.

Thus, the second advantage of hardware implementation is that hardware may be more easily interfaced to biological tissue under study. If specialized hardware-biological hybrid systems are to be realized several conditions must be met. The hardware model must have internal dynamics matching those of a biological neuron i.e. operate on the same time basis as the biological system, the hardware model must receive input from the neuron, the hardware model must deliver biologically relevant signals and the hardware model must allow for real time tuning to match biological parameters during experimentation. A reduced set of parameters would also facilitate implementation ease the collection of experimental data. A general overview of such a system is shown in Fig 1.

The third advantage of physical circuit implementations is that it enables integration into a robotics system. Robotic systems are an ideal complement to simulation in that complex physical laws need not be simulated and modeling may be verified through observing the operation of the robot. Point source neural models implemented in central pattern generators (CPGs) are of particular interest in robotics, for review see [7].

Neural Hardware models fall into roughly three categories. Conductance models are based upon biophysical properties of the membrane [8-11]. Modeling relies on a biophysical explanation of the neuron. Typical challenges associated with these implementations are the complexity of parameter tuning, circuit complexity and lack of a biological time base.

Integrate and Fire (IF) and Leaky Integrate and Fire (LIF) models have been referenced in the literature in principle since 1907 [12]. A recent review integrate and fire neurons is provided in [13]. These linear models do not produce true spiking behavior.

Manuscript received April 7, 2009. This work was supported in part by the Defense Microelectronics Activity Cooperative Agreement #H94003-07-2-0705-SJSU.

D. W. Parent is with San Jose State University, San Jose, CA 95192 (email: dparent@email.sjsu.edu).

E.J. Basham is now with University of California Santa Cruz, Santa Cruz, CA 95064 (email: basham.eric@gmail.com).

Even upon inclusion of a threshold assigned spike mechanism they do not display behavior observed during electrophysiological experimentation such as spiking latencies, activity dependant threshold or tonic spiking modes [14]. The LIF models are prevalent in the literature due to the ease of computational modeling.

In the third general group, are mathematical models derived based upon bifurcation analysis. Of these, the quadratic integrate and fire model (QIF) is the simplest model capable of producing true spiking behavior [15]. The QIF demonstrates true Class I and Class II behavior. The QIF is also interesting in that it serves as the basis for incrementally more complex models, which show an improved fit to observed neural behavior [4, 16] including the exponential model [17]. The exponential model is of particular interest in this application as dynamic I-V curves were used to extract parameter data from electrophysiological experiments [18]. However, the exponential form is not analytically solvable as is the QIF form [15].

II. Methods

A. The quadratic integrate and fire model

The quadratic integrate and fire follows from a reduced form

$$\frac{dV}{dt} = F(V) + I \tag{1}$$

where F(V) is a voltage dependant function which aims to capture the voltage dependant current flow in active membranes. I is the applied current and dV/dt is the time varying membrane voltage. Substituting V^2 for F(V) leads to the one dimensional system

$$\frac{dV}{dt} = V^2 + I \tag{2}$$

which is the topological normal form for saddle node bifurcation. A reset condition is required else the solution would increase to infinity, thus

$$\text{If } V \geq V_{Peak}, \text{ then } V \Rightarrow V_{Reset} \tag{3}$$

If the graph of F(V) vs. V is above the y axis, there is one stable equilibrium point and tonic spiking results. If F(V) is below the y axis, then two stable equilibrium points are present. Bistability results when V_{reset} is above zero and intermittent spiking occurs as applied current is integrated until a spike occurs and the system is reset to the initial equilibrium point. This classification scheme provides a convenient method to evaluate both biological experimental data and hardware performance.

Class I neurons as defined by Hodgkin as those where the firing frequency response to injected current is linear from an arbitrarily low frequency. Class II neurons are those which encompasses all behavior that leads to a discontinuous firing frequency to injected current curve. In comparison to Class I neurons, spiking stops at some arbitrary low frequency.

A more convenient representation is used in [15] where dynamical systems are defined as either integrators or resonators and also either bistable or monostable leading to four classifications. Understanding the proper classification of neural dynamical systems is important to properly model computational properties and properly design testing sequences of systems under development.

B. Implementation of circuits from the quadratic integrate and fire equation

Discrete and integrated versions of integrate and fire neurons are common in the literature. However, these implementations typically require a mechanism to generate spikes. Implementations of quadratic integrate and fire neurons are less common. To our knowledge, there are no discrete implementations of the QIF reported in the literature.

Before the advent of high speed digital computers, it was common to solve differential equations in general purpose analog hardware. The techniques for implementation of differential equations in analog computers are well laid out in [19]. Excellent historical perspectives on analog computation may be found in [20, 21].

Our approach differs in that a switch with hysteresis was used in conjunction with a noninverting switched integrator [22] to implement the reset function defined in (3) and allow the integrator node to completely discharge. Without this hysteretic behavior aberrant spike shapes may be observed during large amplitude input stimulation.

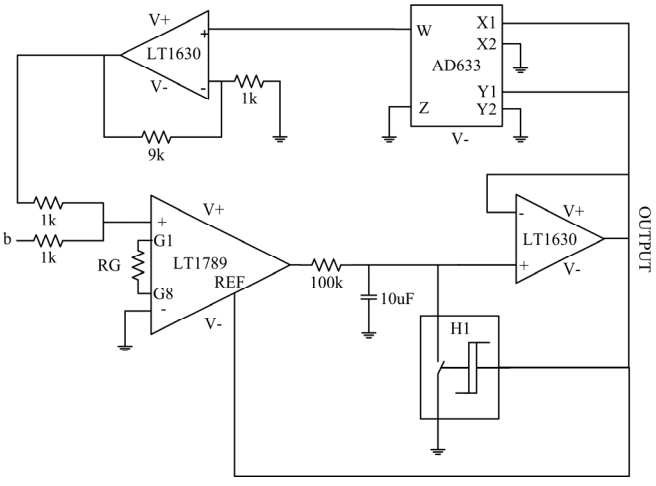

Fig. 2. Circuit schematic of the analog QIF. The measured output is buffered by the LT1630 shown as "output x". The switch encapsulates hysteretic behavior. The details of the hysteretic switch are shown in Fig. 3.

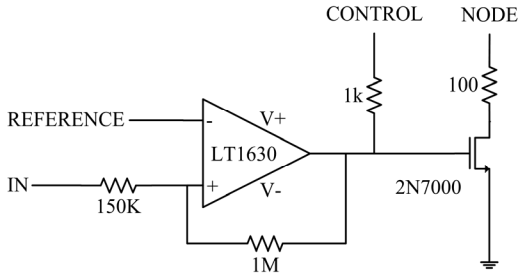

Fig. 3. Circuit Schematic for the hysteretic switch. R1 is 130kΩ, R2 is 1MΩ, Vref can be varied to alter the excitability of the circuit and the resting voltage and R3 is 1kΩ. The switch is a discrete nFET 2N7000.

An Analog Devices AD633 discrete multiplier chip is used to provide the quadratic behavior. The AD633 is low cost and also supports single rail operation. Further simplification may be possible with the AD835. The AD835 multiplies without attenuation and has a simplified transfer function. However, because it has a 5V supply maximum it may not be appropriate for driving a stimulator directly due to compliance limitations common in neural stimulation systems. The switch was implemented using a 2N7000 discrete transistor as shown in Fig. 3. A noninverting hysteretic comparator implemented with the second LT1630 drives the switch.

C. *Evaluation of circuit performance*

The quadratic integrate and fire model (2) is reformulated as

$$RC\frac{d}{dt}V = V^2 + b \tag{4}$$

where R is the resistance of the integrator, and C is the capacitance of the integrator. Assuming that the RC product is greater than any parasitic resistances (switch) or capacitances (op amp input capacitance) (4) can be used to determine the period of spiking for the quadratic integrate and fire circuit for a dc value of b. Given that (4) is a first degree nonlinear differential equation it can be solved by separation of variables

$$RC\frac{RC}{V^2 + b}dV = dt \tag{5}$$

The left side of (5) can be integrated and is found to be equal to an inverse tangent function (6). The integration of dt is simply time. All integration constants are assumed to be equal to zero as the capacitance storing the integrated value is reset for each spike. The imaginary number j is used (j=√-1).

$$\frac{RC}{j2\sqrt{b}}\int\frac{1}{V - j\sqrt{b}}dV + \frac{-1}{V + j\sqrt{b}}dV = \frac{RC}{\sqrt{b}}\tan^{-1}\frac{V}{\sqrt{b}} \tag{6}$$

The voltage as a function of time is then given as

$$V = \sqrt{b}\tan\left(\frac{\sqrt{b}}{RC}t\right) \tag{7}$$

Subject to the reset condition

$$\text{If } V \geq V_{\text{Trip High}}\text{, then } V \Rightarrow V_{\text{Trip Low}} \tag{8}$$

where $V_{\text{Trip High}}$ is the voltage at which the switch discharges the charge of the integrating capacitance, and $V_{\text{Trip Low}}$ is the voltage at which the integrating capacitance is allowed to start integrating.

Equation 8 can be used to solve for the time it takes for the signal to rise from one voltage to another for a given dc value of b. Assuming that the rise time of the signal is much greater than the fall time (this is easily accomplished by setting the integrator resistance R to be much greater than the internal resistance of the discharge switch), the period of the spike is given by (9).

$$Period \cong T_{Rise} = \frac{RC}{\sqrt{b}}\left(\tan^{-1}\frac{V_{TripHigh}}{\sqrt{b}} - \tan^{-1}\frac{V_{TripLow}}{\sqrt{b}}\right) \tag{9}$$

III. RESULTS

Spiking behavior of the neuron circuit is shown in Fig 4 and Fig. 5.

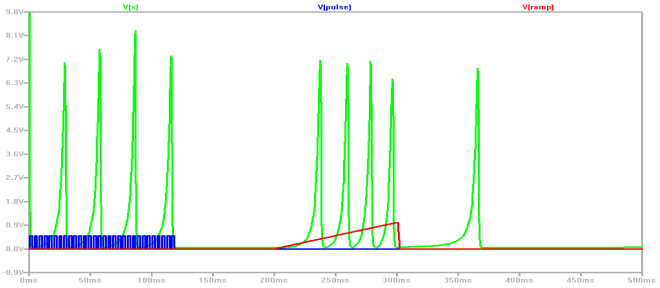

Fig 4. Spiking response to pulsed input (shown in blue) ramp input (shown in red) and accumulation (last spike to the right of graph with no input). This demonstrates the system operates as an integrator with a single stable equilibrium.

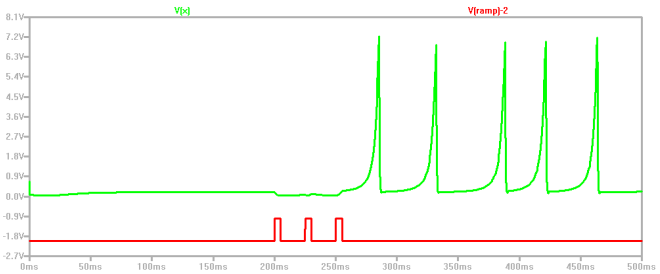

Fig 5. Demonstration of bistability. System bistability occurs when the reset value is above the second stable equilibrium.

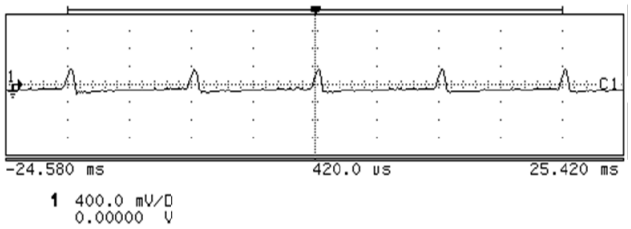

Fig 6. Tonic spiking response of implemented circuit. Tonic spiking at a fixed frequency is typical behavior of a class I excitable system.

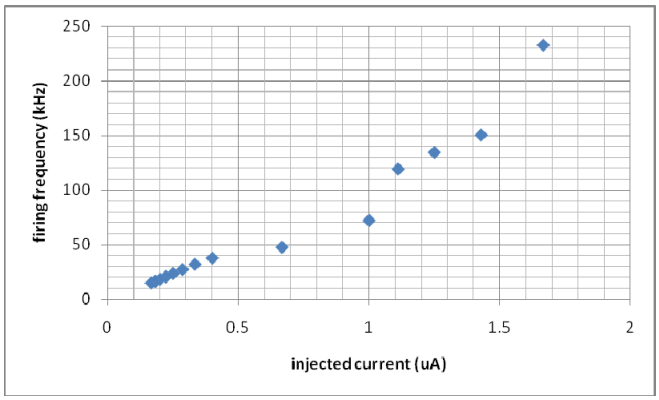

Fig 7. Spiking frequency as a function of input. A nearly linear relationship between frequency and injected current demonstrates this circuit is operating as a class I excitable system.

Circuit testing focused on Class I behavior because, as mentioned earlier, there is only one set of conditions that lead to Class I behavior. Testing is accomplished by varying the input current and the V_{reset} value. Observing Class I behavior (Fig. 6 and 7) proved to be more challenging that observing class II behavior (Fig 4 and Fig. 5) in a circuit that demonstrates both Class I and Class II behavior.

IV. Conclusion

We have presented a mathematically based hardware neural implementation that displays behavior consistent with excitable membranes. A discrete circuit implementation makes this available to experimentalist in robotics, neuroscience and control systems research. The circuit was designed using single rail capable components and circuit topologies to allow single supply operation. Single supply operation is easily obtained by tying all –Vss terminals to ground. Further improvements are possible by using the outlined approach to reducing component count and improve circuit performance.

Acknowledgment

We would like to thank Dr. David Tauck for useful discussions on neural behavior.

References

[1] L. Smith, "Implementing Neural Models in Silicon," in Handbook of Nature-Inspired and Innovative Computing, 2006, pp. 433-475.
[2] E. M. Izhikevich and R. FitzHugh, "FitzHugh-Nagumo Model, http://www.scholarpedia.org/article/FitzHugh-Nagumo_model," Scholarpedia, vol. 1, p. 1349, 2006.
[3] "Data sharing and modeling challenges in neuroscience - a first step towards predictive neuron models?," in Computational and Systems Neuroscience 2008, http://www.cosyne.org/wiki/Cosyne_08, Snow Bird, Utah, 2008.
[4] R. Jolivet, F. Schürmann, T. Berger, R. Naud, W. Gerstner, and A. Roth, "The quantitative single-neuron modeling competition," Biological Cybernetics, vol. 99, pp. 417-426, 2008.
[5] M. Simoni, G. Cymbalyuk, M. Sorensen, R. Calabrese, and S. DeWeerth, "Development of hybrid systems: Interfacing a silicon neuron to a leech heart interneuron," Advances in neural information processing systems, pp. 173-179, 2001.
[6] A. Russell, F. Tenore, G. Singhal, N. Thakor, and R. Etienne-Cummings, "Towards control of dexterous hand manipulations using a silicon Pattern Generator," in Engineering in Medicine and Biology Society, 2008. EMBS 2008. 30th Annual International Conference of the IEEE, 2008, pp. 3455-3458.
[7] A. J. Ijspeert, "Central pattern generators for locomotion control in animals and robots: A review," Neural Networks, vol. 21, pp. 642-653, 2008.
[8] L. Alvado, J. Tomas, S. Saïghi, S. Renaud, T. Bal, A. Destexhe, and G. Le Masson, "Hardware computation of conductance-based neuron models," Neurocomputing, vol. 58-60, pp. 109-115, 2004.
[9] G. Roy, "A Simple Electronic Analog of the Squid Axon Membrane: The NEUROFET," Biomedical Engineering, IEEE Transactions on, vol. BME-19, pp. 60-63, 1972.
[10] L. Alvado, J. Tomas, S. Saighi, S. Renaud, T. Bal, A. Destexhe, and G. Le Masson, "Hardware computation of conductance-based neuron models," Neurocomputing, vol. 58-60, pp. 109-115, 2004.
[11] M. F. Simoni, G. S. Cymbalyuk, M. E. Sorensen, R. L. Calabrese, and S. P. DeWeerth, "A multiconductance silicon neuron with biologically matched dynamics," Biomedical Engineering, IEEE Transactions on, vol. 51, pp. 342-354, 2004.
[12] N. Brunel and M. van Rossum, "Lapicque's 1907 paper: from frogs to integrate-and-fire," Biological Cybernetics, vol. 97, pp. 337-339, 2007.
[13] A. Burkitt, "A Review of the Integrate-and-fire Neuron Model: I. Homogeneous Synaptic Input," Biological Cybernetics, vol. 95, pp. 1-19, 2006.
[14] D. Mishra, A. Yadav, and P. K. Kalra, "Learning with single quadratic integrate-and-fire neuron," 2006, pp. 424-429.
[15] E. M. Izhikevich, Dynamical systems in neuroscience : the geometry of excitability and bursting. Cambridge, Mass.: MIT Press, 2007.
[16] E. M. Izhikevich, "Which model to use for cortical spiking neurons?," Ieee Transactions on Neural Networks, vol. 15, pp. 1063-1070, Sep 2004.
[17] B. Ermentrout, "Ermentrout-Kopell canonical model," Scholarpedia, http://www.scholarpedia.org/article/Ermentrout-Kopell_canonical_model, vol. 3, p. 3, 2008.
[18] L. Badel, S. Lefort, T. Berger, C. C. H. Petersen, W. Gerstner, and M. J. E. Richardson, "Extracting non-linear integrate-and-fire models from experimental data using dynamic I-V curves," Biological Cybernetics, vol. 99, pp. 361-370, Nov 2008.
[19] G. Rizzoni and T. T. Hartley, Principles and applications of electrical engineering, 5th ed. Boston: McGraw-Hill Higher Education, 2007.
[20] R. M. Howe, "Fundamentals of the analog computer: Circuits, technology, and simulation: The history of analog computing," IEEE control systems, vol. 25, pp. 29-36, 2005.
[21] K. Lundberg, "The history of analog computing: introduction to the special section," IEEE Control Systems Magazine, vol. 25, pp. 22-25, 2005.
[22] G. Brisebois, "Instrumentation amp makes noninverting integrator," EDN, p. 2, sept. 19, 2002.