Dynamically Reconfigurable Silicon Array of Spiking Neurons With Conductance-Based Synapses

R. Jacob Vogelstein, *Student Member, IEEE*, Udayan Mallik, Joshua T. Vogelstein, and Gert Cauwenberghs, *Senior Member, IEEE*

***Abstract*—A mixed-signal very large scale integration (VLSI) chip for large scale emulation of spiking neural networks is presented. The chip contains 2400 silicon neurons with fully programmable and reconfigurable synaptic connectivity. Each neuron implements a discrete-time model of a single-compartment cell. The model allows for analog membrane dynamics and an arbitrary number of synaptic connections, each with tunable conductance and reversal potential. The array of silicon neurons functions as an address–event (AE) transceiver, with incoming and outgoing spikes communicated over an asynchronous event-driven digital bus. Address encoding and conflict resolution of spiking events are implemented via a randomized arbitration scheme that ensures balanced servicing of event requests across the array. Routing of events is implemented externally using dynamically programmable random-access memory that stores a postsynaptic address, the conductance, and the reversal potential of each synaptic connection. Here, we describe the silicon neuron circuits, present experimental data characterizing the 3 mm $\times$ 3 mm chip fabricated in 0.5-μm complementary metal–oxide–semiconductor (CMOS) technology, and demonstrate its utility by configuring the hardware to emulate a model of attractor dynamics and waves of neural activity during sleep in rat hippocampus.**

***Index Terms*—Address–event representation (AER), dynamically reconfigurable network, membrane conductance, mixed-signal very large scale integration (VLSI), neural emulator, neurotransmitter quantal release, switched capacitor.**

I. Introduction

NEUROMORPHIC systems engineering [1], [2] emulates both structure and function of biological neural systems in silicon, and correspondingly achieves high levels of efficiency in the implementation of artificial sensory systems. To date, facsimiles of the initial stages of visual and auditory information processing have been implemented on single microchips (e.g., [3]–[7]). However, the complexity of neural computation beyond sensory perception requires a multichip approach and a proper communication protocol between chips to implement higher levels of processing and cognition (see, e.g., [8]).

The common language of neuromorphic chips is the address–event representation (AER) communication protocol [9]–[11], which uses time-multiplexing to emulate extensive connectivity between neurons. In its original formulation, AER effects a one-to-one connection topology; to create more complex neural circuits, convergent and divergent connectivity are required. Several authors have discussed and implemented methods of enhancing the connectivity of AER systems toward this end [5], [12]–[18]. These methods generally employ AER "transceivers" [12], [15], [16] and call for a memory-based projective field mapping that enables routing an address–event (AE) to multiple receiver locations. Accordingly, the chip described in this paper contains 2400 neurons but no hardwired connections between cells, rather depending on an external infrastructure to route events to their appropriate targets.

Neural transceivers are useful alternatives to dedicated (hardwired) architectures for implementing large scale spiking neural networks and models of spike-timing dependent synaptic plasticity [20]–[23] because they are reconfigurable. In a typical hardwired implementation, the size of the network is limited by the number of cells integrated on-chip, where each cell implements a dedicated synapse in a fully connected network. This approach is appropriate to study the dynamics of small, densely interconnected networks, but does not scale to more complex models involving large, sparsely connected networks, such as those responsible for attention, object recognition, and language processing in various areas of cortex. Generally, implementations of cortical models have so far been limited to software simulations—because none of the cortical areas being studied have been fully characterized, the models are always in flux and it is often counterproductive to develop a chip hardwired to a particular network architecture.

There are a few examples of reconfigurable neural array transceivers in the literature [15], [16], [24]–[30]. The one described here differs in some important aspects. First, the silicon neuron implements a more biologically realistic discrete-time model of membrane dynamics that includes conductance-based synapses. Conductance-based synapses allow for different neural dynamics than what can be emulated with standard integrate-and-fire (I&F) models (including those that contain leakage, an activity-independent conductance to the rest potential, in the neural dynamics). Second, the design permits a virtually unlimited number of connections between neurons, with independent control of synaptic strength (conductance) and reversal potential on a per-connection basis. The synaptic

Manuscript received July 21, 2005; revised July 24, 2006. This work was supported in part by the National Science Foundation, the National Institute on Aging, and the Office of Naval Research. The work of R. J. Vogelstein was supported by the National Science Foundation Graduate Research Fellowship.

R. J. Vogelstein is with the Department of Biomedical Engineering, The Johns Hopkins University, Baltimore, MD 21205 USA (e-mail: jvogelst@bme.jhu.edu).

U. Mallik was with the Department of Electrical and Computer Engineering, The Johns Hopkins University, Baltimore, MD 21218 USA. He is now with Goddard Space Flight Center, NASA, Greenbelt, MD 20771 USA (e-mail: udayan@gmail.com).

J. T. Vogelstein is with the Department of Neuroscience, The Johns Hopkins University, Baltimore, MD 21205 USA (e-mail: joshuav@jhu.edu).

G. Cauwenberghs is with the Division of Biological Sciences, University of California at San Diego, La Jolla, CA, 92093 USA (e-mail: gert@ucsd.edu).

Digital Object Identifier 10.1109/TNN.2006.883007

"wiring" and parameters can be changed on the fly by reprogramming digital random-access memory. This ability to rapidly alter synaptic connectivity and synaptic parameters also supports the implementation of synaptic dynamics (Section III-B) and spike-timing dependent synaptic plasticity [31]. From a circuits and systems perspective, our approach combines advantages of analog and digital processing by dividing the network architecture into two main components: neural membrane dynamics implemented in an analog VLSI array, and synaptic connectivity implemented in dynamically reconfigurable digital memory. Also, the silicon neuron uses a switched-capacitor architecture to efficiently and reliably implement the membrane dynamics in discrete time, obviating the need for precisely matched linear transconductance elements in a continuous-time implementation.

II. Virtual Synapses

Rather than hardwiring synapses to form a physically connected network of neurons, we implement synapses as soft (virtual) connections. In this system, each neuron is given a unique address, and an off-chip lookup table (LUT) represents synapses as a mapping between presynaptic and postsynaptic neuron addresses (Fig. 1) [16], [19]. When a neuron fires an action potential, its address is transmitted over a shared bus to a decoder that has access to the LUT—this is called an AE. Depending on the data stored in the lookup table and the mechanism used to determine postsynaptic targets, this technique can be used to emulate one-to-one [9] or one-to-many [12] connections. Additionally, it allows for arbitrary and reconfigurable connectivity, as any neuron can connect to any other, and synapses can be rerouted dynamically by changing entries in the table. As long as the bus speed is significantly faster than the event generation rate or the spike processing dynamics, a single connection can be multiplexed in time to emulate extensive connectivity between neurons [12].

A schematic representation of the lookup table used to store the network connectivity is shown in Fig. 1. Each entry represents a single "virtual synapse" and specifies both the presynaptic and postsynaptic neurons' addresses, as well as the following four synaptic parameters:

n	number of release sites;
p	probability of release;
q	quantal postsynaptic response;
E	reversal equilibrium potential.

The overall strength of the synaptic connection can be expressed as a conductance, g_{net}, which is the product of the first three parameters [32]

$$g_{\text{net}} = npq. \tag{1}$$

That is, the conductance is proportional to the product of the number of synaptic release sites on the presynaptic neuron, the probability of synaptic release, and the postsynaptic response to a quantal amount of neurotransmitter. The polarity of the synaptic connection (excitatory or inhibitory) is determined by the synaptic reversal (equilibrium) potential E, which functions as an electrochemical "battery" across the cell membrane and

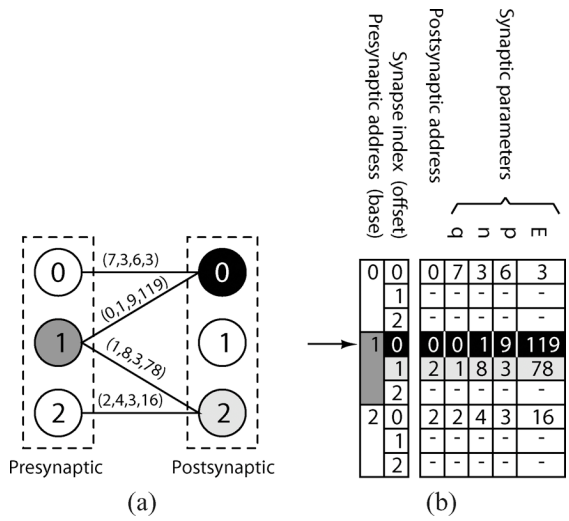


Fig. 1. (a) Virtual synapses: address-domain connectivity between presynaptic and postsynaptic neurons. Numbers in parentheses correspond to the synaptic parameters (q, n, p, and E) defined in Section II. (b) Lookup table storing addresses and parameters of the synaptic connections in random-access memory. The address of a presynaptic action potential "event" is used as an index into the table to identify postsynaptic target addresses and their corresponding synaptic parameters [1].

varies according to the ionic species (K^+, Na^+, Cl^-, etc.) conducted by the synapse. A model of how all of these parameters interact at the postsynaptic site, adapted for analog VLSI implementation, is presented in Section III.

III. Neural Model

A number of silicon neurons have been presented in the literature with varying degrees of biological accuracy [33]–[40]. The most detailed and accurate silicon models feature many parameters and are very flexible, but occupy a large on-chip area and, therefore, limit the number of cells that can be fabricated on a single chip. The simplest models contain only a few transistors and are well suited for implementation in a large scale network, but deviate significantly from the biology and have few adjustable parameters. Many applications would benefit from a balance between these two extremes: A more biologically accurate neural model allows for more sophisticated emulation of cognitive functions, but only in the context of a sophisticated network architecture (see [15] and [25] for some examples along these lines). We have, therefore, designed a small-footprint, highly configurable, "general-purpose" silicon neuron that implements a standard extensible model of biological neural membranes.

The question of whether to use a linear I&F model or a conductance-based model has become a popular subject among neural modelers recently, with no clear resolution [41]–[43]. However, it is known that ion channels in real neurons act as nonlinear conductances [44], and this may play an important role in certain network computations (c.f. Section VI). For example, conductance-based models exhibit a concave-down charging characteristic, which has been shown to be useful for synchronizing populations of neurons [45]–[47]. Conductance-based models also exhibit a strong dependence on the order of input events, which may be important for neural

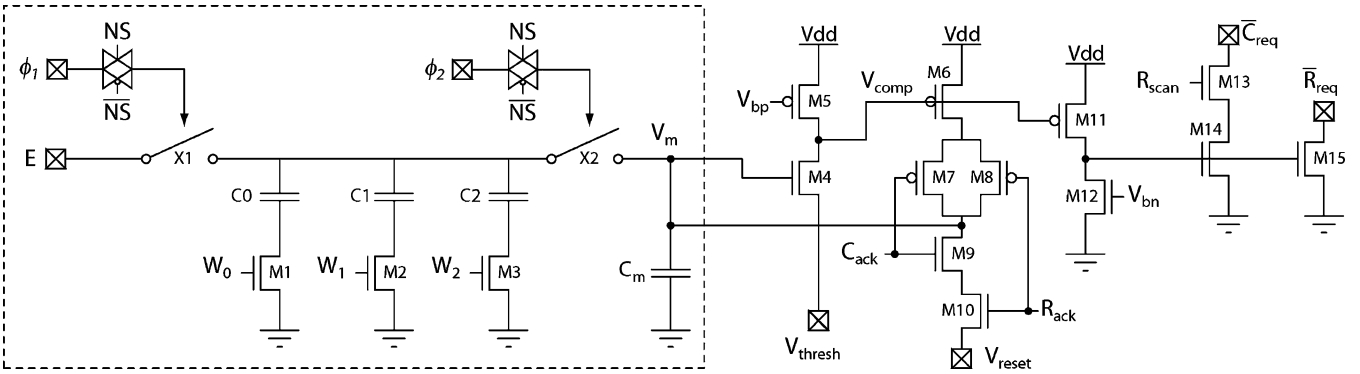


Fig. 2. Silicon single-compartment neuron and "general-purpose" synapse (inside dashed box, [19]), with event generation circuitry (shown right, [16]). When the address of an incoming event is decoded, row- and column-select circuitry activate the cell's neuron select (NS) line, the global signals W_0–W_2 and E are established, and the spike is registered by a pulse on ϕ_1 followed by a pulse on ϕ_2.

calculations based on spike timing (e.g., [48]–[52], and the references therein). For these reasons, we have decided to implement a conductance-based model in our silicon neuron.

A. Membrane Dynamics

The single-compartment model (Fig. 3) is commonly used in computational neuroscience to describe the ionic flux through biological neural membranes. In this model (which has many variations), a neuron is represented as a large membrane capacitance C_m in parallel with a number of conductances and batteries in series. Currents resulting from transient and static inputs are integrated on the capacitor until the potential V_m exceeds a threshold, at which point the cell fires an output (called an *action potential*, or a *spike*). Each type of input to the cell is represented by a conductance-battery combination $\{g, E\}$; transient inputs (e.g., synapses) are implemented with time-varying conductances $g(t)$ whereas static inputs have constant conductances. The primary static input is a very small "leak" conductance g_{leak}, and in the absence of any other inputs, the voltage stored on the membrane capacitor C_m relaxes to the "resting potential" E_{rest}. In contrast, at any given synapse the conductance is usually near zero except for a brief period of time following an input event (modeling the transient opening of ion channels triggered by neurotransmitter binding). The battery potential (or synaptic reversal potential) E_i and the maximum value of conductance (synaptic strength) g_i vary on a per-synapse basis, with some synapses being excitatory (suprathreshold reversal potential) and others being inhibitory (subthreshold reversal potential).

The single-compartment model is specified by the membrane equation

$$C_m \frac{dV_m}{dt} = g_1(t)\cdot(E_1 - V_m) + g_2(t)\cdot(E_2 - V_m) + g_i(t)\cdot(E_i - V_m) + \cdots + g_{\text{leak}}\cdot(E_{\text{rest}} - V_m). \quad (2)$$

Although biology operates in continuous time, most neural interactions occur on the millisecond time scale, so it is possible to simulate the internal dynamics of a neuron using fast, discrete-time steps. Similarly, while multiple synaptic inputs can be active simultaneously in a real neuron, it is essentially equivalent to activate a group of synapses in rapid succession due to biology's low precision in the time domain. We exploit both of these observations in the implementation of our silicon neuron.

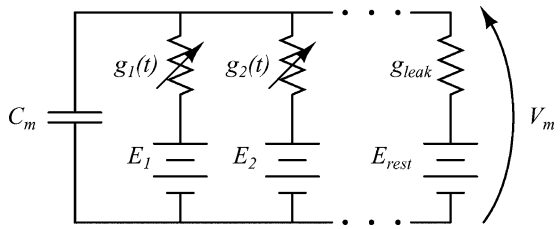


Fig. 3. Single-compartment model of a biological neuron with transmembrane voltage V_m, lumped membrane capacitance C_m, multiple synapses represented by $\{g_i(t), E_i\}$ pairs, and a static leak conductance to the resting potential $\{g_{\text{leak}}, E_{\text{rest}}\}$.

B. Implementation

Multiple different excitatory and inhibitory synaptic inputs to the neuron are implemented by event-based time-multiplexing of a single time-varying supply $E(t)$ and the corresponding series conductance $g(t)$. For the ith synaptic input event to the neuron, the pair $\{g_i, E_i\}$ is generated from the corresponding entries in the LUT (Fig. 1) and presented to the neuron's inputs. The three parameters described by the model in (1) can be emulated, if desired, by repeatedly generating n identical events of magnitude q, conditioned on the binary outcome of a random draw with probability p.

The neural cell schematic is shown in Fig. 2, along with event generation circuitry to trigger and communicate output spikes (Section IV). The cell size, including the event generation circuitry, is 40 μm $\times$ 60 μm in 0.5-μm double-poly triple-metal CMOS technology. Using a simple switched-capacitor architecture [53], [54], this circuit implements a discrete-time version of the membrane equation (see Appendix for details)

$$C_m \frac{\Delta V_m}{\Delta T} = g_i \cdot (E_i - V_m). \quad (3)$$

The conductance g is discrete in amplitude. The amount of charge transferred [q in (1)] depends on which of the three geometrically sized synaptic weight capacitors ($C0$–$C2$) are active. These elements are dynamically switched on and off by binary control voltages W_0–W_2 on the gates of transistors $M1 - M3$. Therefore, the binary coefficients W_0–W_2 provide

eight-level control of conductance values. A larger dynamic range in effective conductance can be accommodated by modulating the multiplicity n and probability p of synaptic events according to (1). Although the synaptic updates in (3) are instantaneous, synaptic dynamics mimicking the time course of postsynaptic potentials in biological neurons can be emulated (at the cost of bandwidth) by repeatedly stimulating the same synapse with gradual changes in n, p, and q.

Incoming spikes are represented by a packet of data containing the target neuron's address, the synaptic weight, and the synaptic reversal potential [Fig. 1(b)]. The address is processed by row- and column-decoders, which activate the neuron select (NS) lines to identify the postsynaptic target(s). The q-component of the weight (1) corresponds to the values of W_0–W_2, which are shared by all on-chip neurons. The synaptic reversal potential is converted to an analog voltage (E) off-chip and also shared by all of the neurons. After these signals are established, the spike is registered when an off-chip clock generates a pulse on ϕ_1 followed by a pulse on ϕ_2, which allows a packet of charge proportional to the difference between the synaptic reversal potential (E) and the membrane potential (V_m) to be accumulated onto the membrane capacitor (C_m). These discrete-time updates (3) model the transient opening of membrane channels in the limit of a very short time window ΔT of postsynaptic response.

To conserve bandwidth, reserved address words allow the row- and column-decoders to select the entire chip or any individual row or column of neurons. Because the signals W_0–W_2 and E are global, many neurons can process events with the same synaptic parameters simultaneously. This allows for an efficient implementation of the continuous leak conductance g_{leak} in the membrane model (2): Leak is approximated by chip-wide periodic (or otherwise repeated) events with low weight and a synaptic equilibrium potential equal to the rest potential E_{rest}.

IV. Event Generation and Arbitration

Information encoded by neurons in the array is represented by the timing of spike events. Therefore, event generation, arbitration, and communication are essential elements of the design.

A. Event Generation

The event generation circuitry of [16] is embedded in every cell (Fig. 2, right). An event—signaled by a low voltage on $\overline{R_{\mathrm{req}}}$—is generated each time the total accumulated charge on C_m causes the neuron's membrane potential (V_m) to exceed the spike threshold voltage (V_{thresh}). Because V_m can rise very slowly, the threshold comparator is implemented as a current-starved inverter ($M4$–$M5$), with $M5$ biased in weak inversion for reduced power dissipation. V_{thresh} is applied to the source of $M4$; this value is shared by all cells in the array and is externally controlled. The corresponding input-referred threshold is approximately equal to $V_{\mathrm{thresh}} + V_{T_n}$, where V_{T_n} is the threshold voltage of $M4$. When V_m exceeds this value, a positive feedback loop implemented by transistors $M6$–$M8$ is activated, triggering a spike event by driving V_m to the positive rail.

A high voltage on $M15$ activates $\overline{R_{\mathrm{req}}}$, the output node of a row-wise wired-NAND, and indicates to the row arbitration circuitry that a cell in that row has generated an event and needs to be serviced. Until this occurs, the row and column acknowledge signals R_{ack} and C_{ack} remain low, maintaining the positive feedback loop and preventing any further inputs from affecting the cell. The row arbitration circuitry indicates it has selected a row by driving one pair of R_{scan} and R_{ack} signals high. All cells in that row with pending events then pull their $\overline{C_{\mathrm{req}}}$ signals low, indicating to the column arbitration circuitry that they have generated events and need to be serviced. Finally, the column arbitration circuitry indicates which column it has selected by driving one column's C_{ack} signal high. At that point, both R_{ack} and C_{ack} are asserted (for one cell only) so the positive feedback loop is inactivated and the reset circuit implemented by nMOS transistors $M9$ and $M10$ causes V_m to become V_{reset} (like V_{thresh}, V_{reset} is shared by all cells in the array and is externally controlled). As V_m drops below the comparator's threshold voltage, V_{comp} is pulled high by $M5$ and the column and row requests ($\overline{C_{\mathrm{req}}}$ and $\overline{R_{\mathrm{req}}}$) are removed. This completes the handshaking sequence between a cell and the arbitration circuitry.

B. Address Encoding and Event Arbitration

The arbitration circuitry on the periphery of the neural array serves two purposes. First, it identifies the location of a spiking cell and converts this location into a row and column address. Second, whenever there are two or more neurons with pending events, it "decides" which one to service. The design is completely asynchronous and consists of a number of arbitration stages, each consisting of a chain of multiple "request propagation" (RP) subcircuits and one "selection unit" (SU) circuit.

The arbitration process begins when one or more neurons activate their row request line(s), $\overline{R_{\mathrm{req}}}$ (Fig. 2). Within each stage of arbitration, the RP subcircuits selectively transmit or prevent the request from proceeding to the next stage, based on the decisions made by the SU. The row that is eventually serviced is the one whose request is propagated through all the stages of arbitration. This row is then activated for column arbitration, which proceeds in an identical fashion. At the conclusion of this process, a single neuron is selected and its address is placed on the external AER bus. When an external receiving device has latched the address, it asserts the chip acknowledge line, which resets the arbiter. Details on the circuit implementation of the arbitration, and how it differs from other AER arbiters (e.g., [55]), will be presented elsewhere [56].

V. Experimental Characterization

We constructed an array of 60×40 conductance-based I&F neurons, with associated event coding and arbitration circuits, on a 3 mm $\times$ 3 mm chip in 0.5-μm CMOS technology. A micrograph of the I&F array chip is shown in Fig. 4. With 5-V supply voltage, the chip consumes 645 μW of power at 10^6 synaptic events per second.

A. Hardware Architecture

Because there are no hardwired connections between neurons on-chip, the silicon I&F neuron array must be embedded

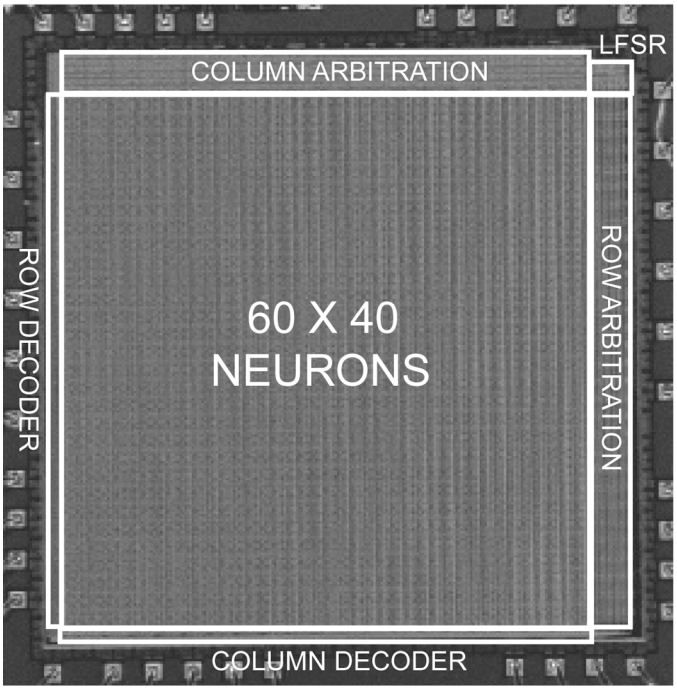


Fig. 4. Conductance-based I&F array chip micrograph. This paper focuses on the 2400-element neural array.

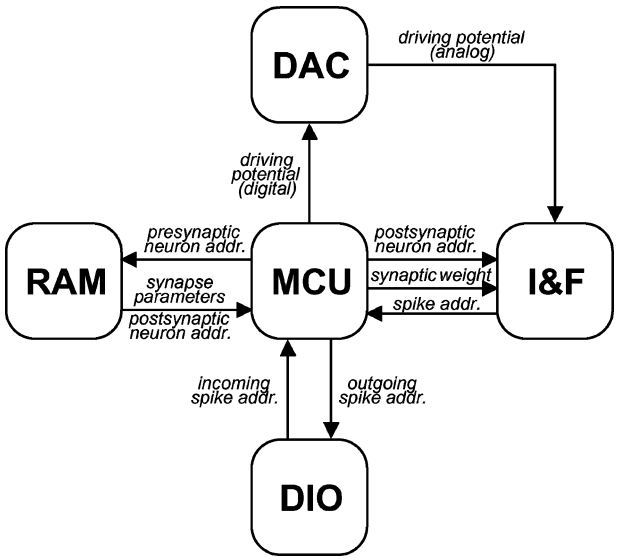


Fig. 5. Block diagram of IFAT system. Incoming and outgoing AEs are communicated through the digital I/O port (DIO), with handshaking executed by the microcontroller (MCU). The MCU generates the random numbers for p in (1) and provides the synaptic conductance g (binary coefficients W_0–W_2) and driving potential E (through DAC) to the conductance-based I&F silicon neuron array, according to the synapse parameters stored in memory (RAM).

in an architecture that can store a network topology and route incoming and outgoing AEs to their appropriate targets (Section II). For this purpose, we developed a board-level system [dubbed I&F array transceiver (IFAT)] that allows up to 4 194 304 synapses to be implemented between up to 9600 neurons located on four I&F chips while processing up to 10^6 synaptic events per second [57].

A block diagram of the IFAT system is shown in Fig. 5. Its components include a 100-MHz field-programmable gate array (FPGA) (Xilinx XC2S100PQ208), 128 MB of nonvolatile SRAM (TI bq4017), a high-speed digital-to-analog converter (DAC) (TI TLC7524CN), a 68-pin digital input–output (I/O) interface (DIO) to communicate with neuromorphic sensors or a computer workstation, and four custom analog VLSI (aVLSI) chips with 2400 I&F neurons each. The FPGA controls access to both an internal and external AE bus, and is responsible for routing incoming, outgoing, and recurrent events to their appropriate postsynaptic targets according to the lookup table stored in RAM.

For the following experiments, a computer was connected to the IFAT's digital I/O port in order to provide the inputs to the system and monitor the output. The digital I/O link to the computer limited the speed of operation, but was helpful to observe and analyze internal network dynamics in the experiments.

B. Results

Fig. 6 illustrates the general functionality of one neuron in the array as it receives a sequence of events while both the synaptic reversal potential and the synaptic weight are varied. It also clearly reveals the strong impact of the order of events on the neural output due to the operation of the conductance-based model, as opposed to a standard I&F model. As in biological neurons, an inhibitory event following a sequence of excitatory events has a greater impact on the membrane potential than the same inhibitory event following many inhibitory events (and vice versa; compare events at arrows A versus A^* and B versus B^* in Fig. 6). This effect is predicted by (2) and (3): For a fixed conductance g, as the difference between the reversal potential E and the membrane potential V_m increases, ΔV_m increases. When the synaptic reversal potential is close to the rest potential, the nonlinear interaction of neural activity gated by such a synapse is referred to as "shunting inhibition," an effect unique to the conductance-based model and missing in both standard and leaky I&F models (Section VII). The ability to implement a form of shunting inhibition is one of the many consequences of our design.

To quantify the neurons' dependence on the synaptic parameters W_i and E, we performed three experiments. First, to determine the effect of the weight capacitors $C0$–$C2$ in the binary expansion W_0–W_2 (Fig. 2), each neuron's membrane potential was reset to a fixed voltage and a series of excitatory events at a fixed reversal potential and synaptic weight were sent to the cell. To simplify the experiment, no inhibitory or leak events were implemented, so the effects are independent of input frequency. The number of events required to elicit a spike was recorded and after ten measurements at each synaptic weight, another cell in the array was tested. The results over all 2400 cells in one chip are summarized by plotting the average ratio of output events to input events versus synaptic weight (Fig. 7). In the absence of parasitics, we would expect a greater slope for the lines in Fig. 7 and a y-intercept of zero. In practice, bottom plate and line capacitance accounts for a relatively large parasitic capacitance that is active even when all weight capacitors are in the "off" state (which accounts for the nonzero y-intercept). At the maximum on-chip synaptic weight and an excitatory reversal

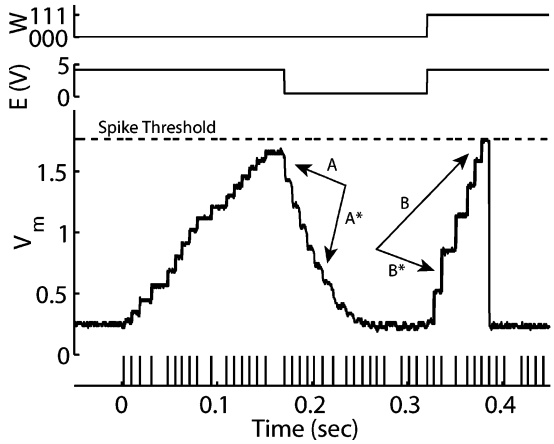


Fig. 6. Data captured from an oscilloscope during operation of the chip. The lower trace illustrates the membrane potential (V_m) of a single neuron in the array as a series of events are sent at times marked by vertical lines at the bottom of the figure. The synaptic reversal potential (E) and synaptic weight (W) are drawn in the top two traces. Arrow labeled A points to the effects of a "shunting" inhibitory input. See text for description of other symbols.

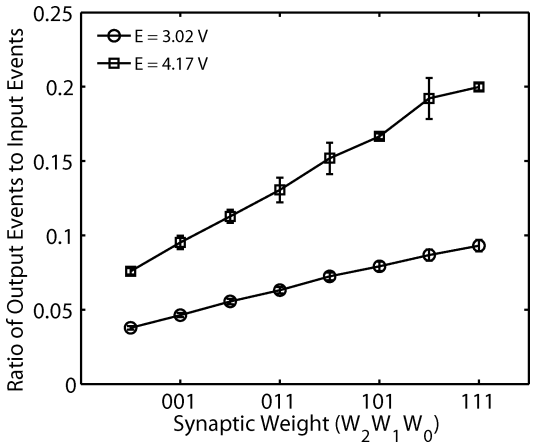


Fig. 7. Average ratio of output events to input events versus synaptic weight. Data are averaged across ten trials per cell at each value of synaptic weight, and then averaged over all 2400 cells. Error bars represent ± 1 standard deviation.

potential of 4.17 V, approximately five input events are required to generate an output event. However, this can be decreased by increasing E or, at the expense of bandwidth, by increasing n as described in Section III-B.

The second experiment quantifies the effect of the synaptic reversal potential (E). Here, each neuron's membrane potential was reset to a fixed voltage and a series of events at a given excitatory synaptic reversal potential and synaptic weight were sent to the cell. Again, no inhibitory or leak events were implemented, and the number of events required to elicit a spike was recorded. The results over all 2400 cells were summarized by plotting the average ratio of output events to input events versus synaptic reversal potential using two different values of V_{thresh} (Fig. 8).

To quantify mismatch between neurons on the same chip, we conducted a third experiment wherein all of the event parameters (synaptic weight, synaptic reversal potential, spike threshold voltage, and resting potential) were held constant at values that were barely suprathreshold. The average ratio

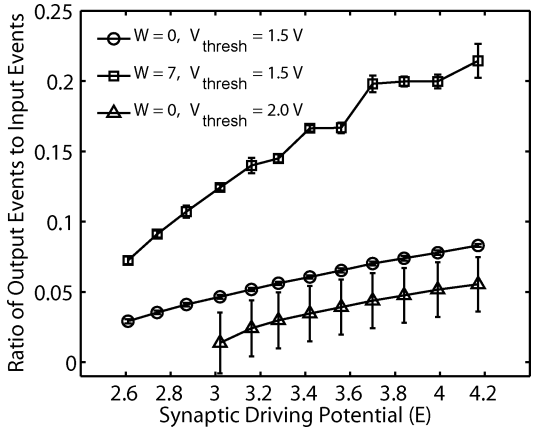


Fig. 8. Average ratio of output events to input events versus synaptic reversal potential E, measured in volts. Data are averaged across ten trials per cell at each value of synaptic weight, and then averaged across all 2400 cells. Error bars represent ± 1 standard deviation.

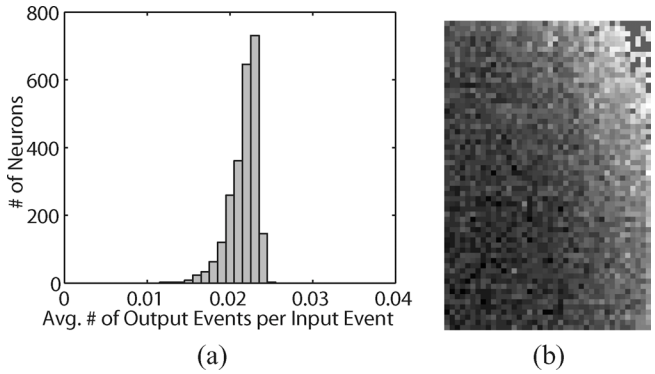


Fig. 9. (a) Histogram showing distribution of the average ratio of output events to input events. (b) Map illustrating spatial trends in the variation of the average ratio of output events to input events across the 60×40 array. Darker regions indicate larger values on a range-normalized scale.

of the number of output events to input events for neurons was measured as before, and the distribution was plotted as a histogram [Fig. 9(a)] with a standard deviation of 0.0017 around the mean of 0.0210. Spatial patterns in the distribution are indicated in the bitmap of Fig. 9(b), where darker pixels correspond to a larger number of output events per input event, normalized to the range of the histogram [Fig. 9(a)]. Evidently, there is a gradient in the upper-right quadrant of this array.

VI. System Example: Combinatorial Attractor Network

To demonstrate the utility of the IFAT system, we used it to implement a combinatorial attractor (CA) network, a model that exhibits some of the neural dynamics observed in hippocampal "place cells" [58]. As shown, the dynamics depend on conductance-based synapses in the network model, which are naturally and efficiently implemented on the IFAT system. The biological basis and predictions of this model, as well as a complete analysis of the network, will be explained in detail elsewhere [59]; here, we will only briefly describe the motivation behind this formulation and illustrate some salient functional properties of the network, in the context of the IFAT system functionality.

A. Biological Motivation

Hippocampal place cells are pyramidal neurons found in layers CA1 and CA3 of the rat hippocampus. They are notable for being active only when the rat is in a few particular locations in space [60], [61]. In experimental settings, it has been shown that as a rat traverses a closed room, its instantaneous position can be uniquely identified by decoding the activity vector from an ensemble of place cells [62]. Moreover, this activity vector persists even after all sensory and motor clues are extinguished, as long as the rat remains in a fixed location [63], [64]. This has led to the notion that the rat hippocampus stores a virtual map of space as an *attractor network*, wherein the present location is represented as a stable "bump" of neural activity [65], [66].

Rat hippocampal activity within a closed room is similar to that predicted by a two-dimensional (2-D) "plane attractor," where a stable activity bump can move smoothly between locations on the plane [67]. Exposing the rat to multiple different rooms leads to the formation of different planar maps. The CA network model allows for continuity between all states (similar to multidimensional attractors), and allows each neuron to participate in multiple nonadjacent attractors (similar to the multichart model [67]). It also provides for smooth transitions between stable states in different maps, as well as for "partial remapping."

The primary biological constraints on the CA network are that it functions with spiking excitation and inhibition, and that neurons must fire at realistic rates, i.e., neurons within a stable bump should not be maximally activated. In Section VI-B, the construction and operation of a CA network are described quantitatively, and its operation is illustrated by examples of a CA network implemented on the IFAT hardware in Section VI-C.

B. Network Architecture

We implemented a CA network representing 400 distinct places in a one-dimensional (1-D) ring using 200 excitatory neurons and 20 inhibitory neurons. In this model, each excitatory neuron was randomly assigned to represent two spatial locations along the ring, which were typically nonadjacent (Fig. 10). The network was fully interconnected, so that every neuron received input from every other neuron, and the synaptic weights between excitatory neurons were determined as a function of the distance between the spatial locations each neuron represented. Specifically, for two excitatory neurons N_i and N_j representing locations (d_{i1}, d_{i2}) and (d_{j1}, d_{j2}), the synaptic weights $W_{i,j}$ and $W_{j,i}$ were given by

$$\begin{aligned} W_{i,j} = W_{j,i} &= F(d_{i1}, d_{i2}, d_{j1}, d_{j2}) \\ &= e^{\frac{-(d_{i1}-d_{j1})^2}{2\sigma^2}} + e^{\frac{-(d_{i1}-d_{j2})^2}{2\sigma^2}} \\ &\quad + e^{\frac{-(d_{i2}-d_{j1})^2}{2\sigma^2}} + e^{\frac{-(d_{i2}-d_{j2})^2}{2\sigma^2}} \end{aligned} \tag{4}$$

where σ was a free parameter. In software simulations, these values were directly translated into maximum synaptic conductances $g_i(t)$ with a fixed reversal potential (2), whereas the IFAT implementation used a discrete (nonlinear) mapping from W to a parameter set (E, n, q) [see (1); we fixed $p = 1$ to simplify analysis in both hardware and software]. This mapping ensured that for each value of conductance, the number of input events per output event was roughly equivalent in both hardware and software.

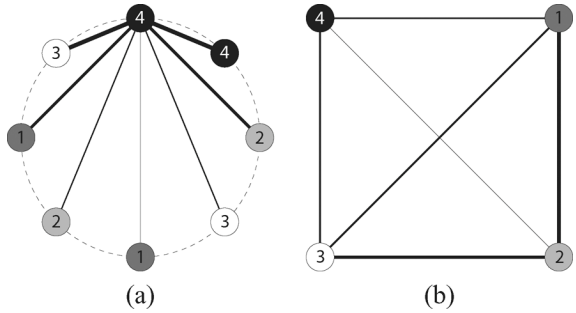


Fig. 10. Example of a simple CA network with four neurons. (a) Spatial representation of the CA network. Each neuron (shaded circles) represents two locations along the perimeter of a large circular area (dashed line). For clarity, only synaptic connections from neuron 4 are shown (solid lines), but the network is fully interconnected. Distance between neurons along the perimeter is represented by line width. (b) Neural representation of the same neuron 4 CA network. Solid lines between neurons represent bidirectional synapses, and line width is proportional to synaptic strength; see (4).

In contrast to the excitatory cells in the CA network, the inhibitory neurons do not represent any particular location, but rather act as a moderating force allowing stable states to emerge without overtaking the entire network. Thus, as the net excitation in the network increases during formation of a bump, the net inhibition must increase faster to dampen the effects and ensure that the bump does not spread and create epileptic activity. However, the inhibition must not increase too quickly or it will quench the bump and force the network to a quiescent state. In the IFAT implementation, this balance was achieved by allowing every excitatory neuron to connect to the inhibitory neuron with synaptic parameters $E = 4.28$ V, $n = 1$, and $q = 0.125$, and by fixing synaptic parameters from the inhibitory neuron to every excitatory neuron at $E = 60$ mV, $n = 1$, and $q = 0.25$, with a global resting potential of $E = 500$ mV (these values were chosen for convenience; other combinations of synaptic parameters were also effective).

Although one might expect to find only a small parameter space in which the CA network can function properly, software simulations show that the use of conductance-based synapses greatly increases this range. For example, the maximum conductance for excitatory connections could vary significantly using conductance-based synapses, whereas a grid search (with three digits of precision) for acceptable values of conductance using standard "linear" synapses (with ΔV independent of V_m) failed to produce any results. The conductance-based synapses described in Section III-B were, therefore, very useful (if not critical) for implementing this network on the IFAT, because the IFAT has a limited range for synaptic parameters (as compared to software). Furthermore, the qualitative experience of configuring the IFAT to implement a CA was relatively simple, with a stable operating point located after only a few attempts.

One advantage of implementing the CA network on the IFAT hardware instead of in software is a significant decrease in simulation time. Because of the extensive connectivity between cells (over 40 000 synapses in a 200-neuron network), the system of coupled differential equations (2) takes over 40 min to simulate approximately 4000 output spikes (approximately 5 s of neural

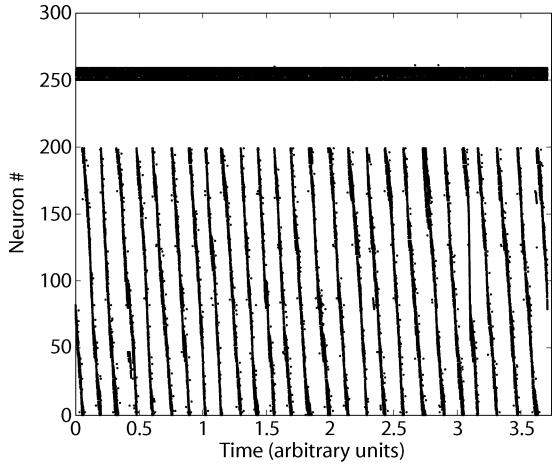


Fig. 11. Traveling wave of neural activity. Approximately 40 000 events from excitatory and inhibitory neurons (numbered 0–199 and 250–269, respectively) are represented as individual points and plotted as a function of time.

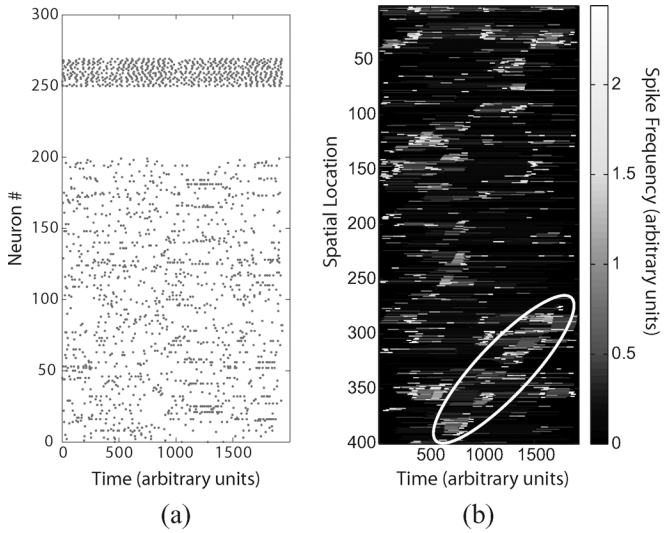


Fig. 12. Discontinuous traveling waves similar to those observed during rat sleep cycles. (a) IFAT output events from excitatory neurons 0–199 and inhibitory neurons 250–269 plotted as neuron number versus time. (b) IFAT output events plotted according to their representation of spatial location versus time. Although there is no obvious structure to the events in neural space, a discontinuous traveling wave (circled) is evident in the neural representation of physical space.

activity) on a 3-GHz Intel computer with 2 GB of RAM. In contrast, the same network executes in just a few seconds on the IFAT, even with a slow (kilohertz) link to a computer for data logging. Of course, additional measures could be taken to improve the software model's performance [68]–[70], but the same holds for the hardware model.

C. Functional Properties

The basic operation of the CA network was first tested by implementing a "tilted" circular network. Here, neuron N_i was assigned to represent location i, and instead of the symmetric weight function given by (4), $W_{i,i+1}$ was set slightly greater than $W_{i-1,i}$. When a few suprathreshold input events were applied to one neuron, the resulting activity generated a persistent traveling wave that propagated clockwise around the ring (Fig. 11). In software simulations, this can be a noiseless process, but the intrinsic circuit noise in hardware generates a few spurious and out-of-order events.

The continuous traveling wave in Fig. 11 is different than those observed during rat sleep cycles, as the latter typically contain discontinuous traveling waves interposed with periods of quasi-stable activity bumps and spontaneous jumps between states. However, these kinds of activity patterns can be created by the CA model if the 400 different spatial locations are randomly assigned to neurons in the network. An example of this behavior is shown in Fig. 12. Because synaptic weights are assigned based on the spatial representations of the neurons (4), there is little obvious structure to the events generated by adjacent neurons in the network [Fig. 12(a)]. However, when the same activity is plotted as a function of the physical locations each neuron represents [Fig. 12(b)], the activity is observed to move discontinuously through space.

An important feature of the neural outputs in Fig. 12 is that the spike trains appear aperiodic. Aperiodicity typically arises only when neural parameters are specifically chosen for this purpose, but it seems to be an emergent property of the CA network (in both software and hardware), possibly because of the interactions between excitatory and inhibitory neurons, and because conductance-based synapses add charge to neurons nonlinearly. One useful side effect of spike aperiodicity is that it allows for the aforementioned quasi-stability, wherein activity bumps form spontaneously, persist for an arbitrary length of time, and then collapse, sending the system into a different quasi-stable state or a traveling wave. This does not usually occur during regular, periodic spiking, but the chaotic combination of neurons firing at any given time during aperiodic spiking is occasionally sufficient to shift the center of activity away from a stable bump.

By removing the asymmetry in the weight matrix and implementing (4) exactly, the CA network can generate stable bumps of activity instead of traveling waves. In this configuration, it can also represent "partial remapping," which is a phenomenon observed in the hippocampus when a rat enters a room that is similar to another room it has visited—the new room "imports" attractors corresponding to similar locations in other rooms in addition to manifesting novel attractors. An example of partial remapping is shown in Fig. 13, where the network was briefly excited by two simultaneous inputs. Each of these inputs individually represents a particular set of sensory clues and should induce a unique stable activity bump. The parallel combination of both inputs represents indeterminate, intermediate, or otherwise novel sensory information, to which the network responds with a unique persistent stable state that is a nonlinear combination of the two independent inputs. In Fig. 13(b), the new state is manifested as four stable bumps, two of which represent the individual inputs (unfilled arrowheads) and two that emerge from the nonlinear interactions between neurons (filled arrowheads).

Because linear combinations of independent inputs result in nonlinear combinations of the outputs, the capacity of a CA network is potentially very large. Although a theoretical result may be computed mathematically, an empirical result requires testing all combinations of inputs. These experiments are currently underway, and a full analysis of the capacity of the CA network will be reported in a future manuscript. The IFAT is

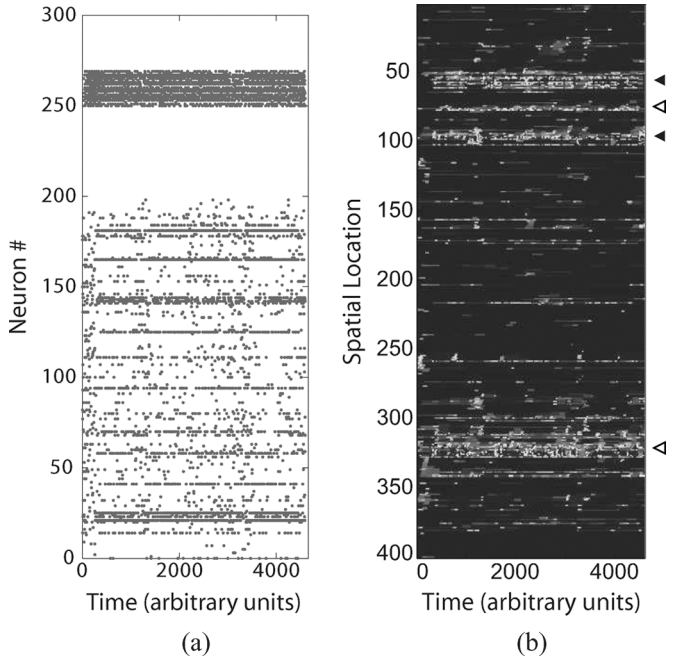


Fig. 13. Example of partial remapping and stable activity bumps. (a) IFAT output events from excitatory neurons 0–199 and inhibitory neurons 250–269 plotted as neuron number versus time. (b) IFAT output events plotted as spatial location versus time. Color scale as in Fig. 12. Arrowheads point to activity bumps; see text for details.

well suited to this type of large scale simulation, and offers a significant speed advantage over an equivalent software simulation (by a factor of a few hundred in our tests).

VII. Discussion

A. Comparison With Other AER Transceivers

There are some important differences between the implemented AER architecture and previous AER-based neural systems (e.g., [15] and [16]). First, the implemented system allows for emulation of a large number of synapses on every cell (limited only by the capacity of the LUT), as each incoming event can be assigned a unique weight and reversal potential. Second, it emulates conductance-based synapses, which can have important implications in neural models (c.f. Sections III and VI). For example, shunting inhibition by conductance-based synapses is key to some known complex nonlinear interactions between synaptic inputs [71]–[74]. Unlike standard I&F models, the order of input events in a conductance-based model is an essential factor in determining the neural output (Fig. 6). Even so-called "leaky" I&F models, which include a static conductance to the rest potential, fall short of modeling such nonlinear interaction effects because this conductance is activity independent. In contrast, shunting inhibition by a conductance-based synapse is both level dependent and activity dependent [32]. Third, charge-based circuits exploit better matching between capacitors than between MOS transistors due to threshold variations, which results in greater uniformity of operation across the chip when compared to a current-based implementation (Fig. 9). Finally, there is very little charge leakage off the membrane capacitor, allowing for the implementation of neural dynamics on various time scales. Because neural integration is implemented in discrete time, it is also possible to decouple event timing from emulated time, and dynamically warp the time axis [31].

In biology, neurons are subject to leakage currents through g_{leak} (Fig. 3), so excitatory events must occur in rapid succession to bring the membrane potential above threshold. In the silicon neuron, leakage current is implemented by repeatedly sending discrete events at a low synaptic weight and with synaptic reversal potential E_{leak}. In both cases, the rate of leakage essentially defines the time constant of the system; without leakage, the neuron would be insensitive to absolute time. Dynamic time warping allows for modulating the speed at which the system operates by electrically changing the time constant depending on the computational load. That is, by varying the frequency of leakage events, we can accelerate or decelerate the effective time constant and run small networks faster than their biological counterparts, while larger or more active networks may run slower than real-time, as limited by bandwidth of the AE bus and any peripheral interfaces. Of course, when our system interacts with a neuromorphic sensor [75], operation is constrained to real-time.

B. Comparison With Digital Simulators

The event-driven approach to emulating spiking neural networks with dynamic synapses also lends itself to efficient implementation in purely digital hardware or in software on digital computers [76]. There are two fundamental differences in the mixed-signal implementation described here. The first is parallelization: When the network architecture is configured so that postsynaptic targets with identical synaptic weights and equilibrium potentials are arranged in rows, columns, or whole chips, the IFAT can perform up to 7.68 billion operations per second [77]. The second difference between this mixed-signal system and a purely digital system is that while the synaptic routing is digital and therefore deterministic (other than stochastic gating of events with the synaptic parameter p in the model of quantal release in Section II), the analog implementation of membrane dynamics (3) contributes imprecision due to component mismatch and noise. Though undesirable in many computational tasks, these sources of imprecision account for variability naturally present in biological neural networks. Computational artifacts of perfectly matched or noise-free network components often give rise to biologically unrealistic results of simulated network response.

One important implication of noise in the membrane dynamics is possible decoupling between timing of input and output events. In particular, additive noise in membrane current and in the membrane threshold potential V_{thresh} may trigger an action potential event even in the absence of a coincident synaptic input event. Random neural firing patterns are accurately registered by the event generation and arbitration circuits in the AER architecture, yet are harder to emulate in digital hardware or software with discrete-time steps.

While spiking patterns of individual biological neurons are highly random and variable [52], coherent patterns of synchronous firing activity in large neuronal populations arise from complex synaptic interactions [78] and are possibly

modulated by attention-driven mechanisms [79]. Externally supplied additive noise on the V_{thresh} terminal could emulate some important functionality in modulating activity-dependent synchronicity otherwise lost in a deterministic implementation of the conductance-based I&F model.

Even though the event-driven approach to neural emulation is inherently sequential, it can be extended to a parallel architecture in which multiple neurons and/or multiple synapses are serviced at any given time. Partitioning of the neural network architecture over multiple processors in a computer network has been shown to deliver a linear increase in throughput, up to 2×10^6 synaptic events per second per processor node in a modern PC cluster [70]. Similar improvements in throughput can be attained, in principle, in the mixed-signal architecture by partitioning the neural network over multiple AER processing nodes, each with local IFAT transceivers and LUT memory, and with asynchronously pipelined message parsing between processing nodes [55]. Extending pipelining at all levels of the architecture could allow throughputs on the order of 10^8 events per second per AER node, limited by current read access rates of DRAM memory.

C. Address-Domain Learning

The IFAT architecture is well suited for implementing synaptic plasticity in the "address-domain" because the MCU monitors both the incoming and outgoing synaptic events and has read/write access to the synaptic parameters stored in RAM. In this context, plasticity only requires modifying the LUT in response to particular sequences of events. An additional component of learning in biological systems is the creation or elimination of synapses—on the IFAT, this is achieved by inserting or deleting entries in RAM. Like address-domain connectivity, the advantage of address-domain plasticity is that the constituents of the implemented learning rule are not constrained to be local in space or time. Various forms of learning algorithms can be mapped onto the same architecture by reconfiguring the MCU interfacing the IFAT and the LUT.

The IFAT can implement basic forms of Hebbian learning, including spike-timing dependent plasticity (STDP) [80], with very little processing overhead [31]. STDP-based learning rules specify changes in synaptic strength depending on the time interval between each pair of presynaptic and postsynaptic events. An STDP synaptic modification rule can be implemented in the address-domain by augmenting the AER architecture with two event queues, one each for presynaptic and postsynaptic events, and with a counter that is incremented every time a global decay event occurs (this sets the time constant of the system). For every presynaptic event, the sender's address is entered into a queue with an associated timestamp. A postsynaptic event then triggers a sequence of synaptic updates by iterating backward through the queue to find the causal spikes, locating the appropriate synaptic weights in the LUT, and increasing those values by an amount based on the timestamps. Anticausal events require an equivalent set of operations, matching each incoming presynaptic spike with a second queue of postsynaptic events. Depending on the size of the network and the rate of learning, the weight updates can occupy a large fraction of the MCU's resources, but because these operations are independent of the silicon neurons and AER bus, they can be executed more quickly with faster components.

VIII. Conclusion

The conductance-based I&F array chip provides a solution for creating large scale networks of silicon neurons with arbitrary connectivity and reprogrammability. When combined with the other hardware in the IFAT system, it is a powerful tool for simulating cortical circuits. In general, the IFAT has been designed for reconfigurability and ease of use (e.g., implementing a new network only requires downloading a table of synaptic values to the RAM) rather than speed or throughput. To date, the system has been used to detect salience in a visual image [57], spatially modulate a visual image [81], detect spatially oriented changes in visual contrast in real time [75], compute a nonlinear pooling function similar to winner-take-all [77], and implement a model of attractor dynamics and waves of neural activity in rat hippocampus (Section VI). Future work will focus on increasing the breadth of applications to include sensory modalities other than vision and more complex cortical models.

Appendix

Derivation of Discrete-Time Membrane Equation

Equation (3) is derived using simple switched-capacitor analysis techniques. At the end of a previous update $(i-1)$, switches $X1$ and $X2$ (Fig. 2) are open and charge Q_m is stored on the membrane capacitor C_m in proportion to the membrane potential V_m

$$Q_m^{i-1} = C_m V_m^{i-1}. \tag{5}$$

When the next input event (i) arrives, switch $X1$ is closed first, and charge Q_w is stored across the active weight capacitors $C0$–$C2$ in proportion to the synaptic reversal potential E

$$Q_w^i = C_w^i E^i. \tag{6}$$

Switch $X1$ is then opened, and switch $X2$ is closed, sharing charge between the weight and membrane capacitors. The total charge (Q_{tot}) is simply the sum of the two stored charges, and the resulting voltage is proportional to the total charge and the total capacitance

$$Q_{\mathrm{tot}}^i = Q_w^i + Q_m^{i-1} \tag{7}$$

$$V_m^i = \frac{Q_{\mathrm{tot}}^i}{C_m + C_w^i}. \tag{8}$$

Rearranging terms, and using (5)–(7) in (8), we obtain

$$\left(C_m + C_w^i\right) V_m^i - C_m V_m^{i-1} = C_w^i E^i \tag{9}$$

which yields (3) with the following definition of terms:

$$V_m = V_m^{i-1}$$

$$\Delta V_m = V_m^i - V_m^{i-i}$$

$$g_i = \frac{C_m C_w^i}{C_m + C_w^i} \frac{1}{\Delta T}.$$

The membrane potential thus updated, switch $X2$ is opened, and the synapse is ready to process another event.

ACKNOWLEDGMENT

The authors would like to thank K. Zhang for supervising J. T. Vogelstein's work on the CA model.

REFERENCES

[1] C. Mead, *Analog VLSI and Neural Systems*. Reading, MA: Addison-Wesley, 1989.

[2] T. Lande, Ed., *Neuromorphic Systems Engineering—Neural Networks in Silicon*. Norwell, MA: Kluwer, 1998.

[3] C. Koch and H. Li, Eds., *Vision Chips: Implementing Vision Algorithms with Analog VLSI Circuits*. Los Alamitos, CA: IEEE Computer Press, 1995.

[4] A. van Schaik, E. Fragniere, and E. Vittoz, "Improved silicon cochlea using compatible lateral bipolar transistors," in *Advances in Neural Information Processing Systems*, D. Touretzky, M. Mozer, and M. Hasselmo, Eds. Cambridge, MA: MIT Press, 1996, vol. 8, pp. 671–677.

[5] N. Kumar, W. Himmelbauer, G. Cauwenberghs, and A. Andreou, "An analog VLSI chip with asynchronous interface for auditory feature extraction," *IEEE Trans. Circuits Syst. II*, vol. 45, no. 5, pp. 600–606, May 1998.

[6] K. A. Zaghloul and K. Boahen, "Optic nerve signals in a neuromorphic chip II: Testing and results," *IEEE Trans. Biomed. Eng.*, vol. 51, no. 4, pp. 667–675, Apr. 2004.

[7] S. Kameda and T. Yagi, "An analog silicon retina with multichip configuration," *IEEE Trans. Neural Netw.*, vol. 17, no. 1, pp. 197–210, Jan. 2006.

[8] T. Y. W. Choi, B. E. Shi, and K. A. Boahen, "An ON-OFF orientation selective address event representation image transceiver chip," *IEEE Trans. Circuits Syst. I: Reg. Papers*, vol. 51, no. 2, pp. 342–353, Feb. 2004.

[9] M. Mahowald, *An Analog VLSI System for Stereoscopic Vision*. Boston, MA: Kluwer, 1994.

[10] M. Sivilotti, "Wiring considerations in analog VLSI systems with application to field-programmable networks," Ph.D. dissertation, California Inst. Technol., Pasadena, CA, 1991.

[11] J. Lazzaro, J. Wawrzynek, M. Mahowald, M. Sivilotti, and D. Gillespie, "Silicon auditory processors as computer peripherals," *IEEE Trans. Neural Netw.*, vol. 4, no. 3, pp. 523–528, May 1993.

[12] K. A. Boahen, "Point-to-point connectivity between neuromorphic chips using address events," *IEEE Trans. Circuits Syst.s II*, vol. 47, no. 5, pp. 416–434, May 2000.

[13] C. M. Higgins and C. Koch, "Multi-chip neuromorphic motion processing," in *Proc. 20th Anniversary Conf. Adv. Res. VLSI*, D. S. Wills and S. P. DeWeerth, Eds., Los Alamitos, CA, 1999, pp. 309–323.

[14] S.-C. Liu, J. Kramer, G. Indiveri, T. Delbrück, and R. Douglas, "Orientation-selective aVLSI spiking neurons," in *Advances in Neural Information Processing Systems*, T. G. Dietterich, S. Becker, and Z. Ghahramani, Eds. Cambridge, MA: MIT Press, 2002, vol. 14.

[15] G. Indiveri, E. Chicca, and R. J. Douglas, "A VLSI reconfigurable network of integrate-and-fire neurons with spike-based learning synapses," in *Proc. Euro. Symp. Artif. Neural Netw.*, 2004, pp. 405–410.

[16] D. H. Goldberg, G. Cauwenberghs, and A. G. Andreou, "Probabilistic synaptic weighting in a reconfigurable network of VLSI integrate-and-fire neurons," *Neural Netw.*, vol. 14, no. 6–7, pp. 781–793, 2001.

[17] S. R. Deiss, R. J. Douglas, and A. M. Whatley, "A pulse-coded communications infrastructure for neuromorphic systems," in *Pulsed Neural Networks*, W. Maass and C. M. Bishop, Eds. Cambridge, MA: MIT Press, 1999, ch. 6, pp. 157–178.

[18] P. Häfliger, "Asynchronous event redirecting in bio-inspired communication," in *Proc. IEEE Int. Conf. Electr., Circuits Syst.*, Sep. 2001, pp. 87–90.

[19] R. J. Vogelstein, U. Mallik, and G. Cauwenberghs, "Silicon spike-based synaptic array and address-event transceiver," in *Proc. IEEE Int. Symp. Circuits Syst.*, Vancouver, BC, Canada, 2004, vol. 5, pp. 385–388.

[20] W. Maass and C. M. Bishop, Eds., *Pulsed Neural Networks*. Cambridge, MA: MIT Press, 1999.

[21] A. Bofill-i-Petit and A. F. Murray, "Synchrony detection by analogue VLSI neurons with bimodal STDP synapses," in *Advances in Neural Information Processing Systems*, S. Thrun, L. Saul, and B. Schölkopf, Eds. Cambridge, MA: MIT Press, 2004, vol. 16.

[22] J. Arthur and K. Boahen, "Learning in silicon: timing is everything," in *Advances in Neural Information Processing Systems*, Y. Weiss, B. Schölkopf, and J. Platt, Eds. Cambridge, MA: MIT Press, 2006, vol. 18, pp. 75–82.

[23] G. Indiveri, "Neuromorphic bistable VLSI synapses with spike-timing-dependent plasticity," in *Advances in Neural Information Processing Systems*, S. Becker, S. Thrun, and K. Obermayer, Eds. Cambridge, MA: MIT Press, 2003, vol. 15, pp. 1115–1122.

[24] B. Taba and K. A. Boahen, "Topographic map formation by silicon growth cones," in *Advances in Neural Information Processing Systems*, S. T. S. Becker and K. Obermayer, Eds. Cambridge, MA: MIT Press, 2003, vol. 15, pp. 1139–1146.

[25] J. V. Arthur and K. A. Boahen, "Recurrently connected silicon neurons with active dendrites for one-shot learning," in *Proc. IEEE Int. Joint Conf. Neural Netw.*, 2004, vol. 3, pp. 1699–1704.

[26] T. Horiuchi and K. Hynna, "Spike-based VLSI modeling of the ILD system in the echolocating bat," *Neural Netw.*, vol. 14, pp. 755–762, 2001.

[27] R. Paz, F. Gomez-Rodriguez, M. Rodriguez, A. Linares-Barranco, G. Jimenez, and A. Civit, "Test infrastructure for address-event-representation communications," in *Lecture Notes in Computer Science*. Berlin, Germany: Springer-Verlag, 2005, vol. 3512, pp. 518–526.

[28] H. K. Riis and P. Häfliger, "An asynchronous 4-to-4 AER mapper," *Lecture Notes Comput. Sci.*, vol. 3512, pp. 494–501, 2005.

[29] Q. Zou, Y. Bornat, J. Tomas, S. Renaud, and A. Destexhe, "Real-time simulations of networks of Hodgkin-Huxley neurons using analog circuits," *Neurocomput.*, vol. 69, pp. 1137–1140, 2006.

[30] E. Ros, E. M. Ortigosa, R. Agis, R. Carrillo, and M. Arnold, "Real-time computing platform for spiking neurons (RT-spike)," *IEEE Trans. Neural Netw.*, vol. 17, no. 4, pp. 1050–1063, Jul. 2006.

[31] R. J. Vogelstein, F. Tenore, R. Philipp, M. S. Adlerstein, D. H. Goldberg, and G. Cauwenberghs, "Spike timing-dependent plasticity in the address domain," in *Advances in Neural Information Processing Systems*, S. Becker, S. Thrun, and K. Obermayer, Eds. Cambridge, MA: MIT Press, 2003, vol. 5.

[32] C. Koch, *Biophysics of Computation: Information Processing in Single Neurons*. New York: Oxford Univ. Press, 1999.

[33] M. Mahowald and R. Douglas, "A silicon neuron," *Nature*, vol. 354, pp. 515–518, 1991.

[34] S. Saighi, J. Tomas, Y. Bornat, and S. Renaud, "A conductance-based silicon neuron with dynamically tunable model parameters," in *Proc. 2nd Int. IEEE/EMBS Conf. Neural Eng.*, 2005, pp. 285–288.

[35] A. Bofill-i-Petit and A. F. Murray, "Learning temporal correlations in biologically-inspired aVLSI," in *Proc. IEEE Int. Symp. Circuits Syst.*, 2003, vol. 5, pp. 817–820.

[36] A. van Schaik, E. Fragniere, and E. Vittoz, "A silicon model of amplitude modulation detection in the auditory brainstem," in *Advances in Neural Information Processing Systems*, M. C. Mozer, M. I. Jordan, and T. Petsche, Eds. Cambridge, MA: MIT Press, 1997, vol. 9, pp. 741–750.

[37] M. F. Simoni, G. S. Cymbalyuk, M. E. Sorensen, R. L. Calabrese, and S. P. DeWeerth, "A multiconductance silicon neuron with biologically matched dynamics," *IEEE Trans. Biomed. Eng.*, vol. 51, no. 2, pp. 342–354, Feb. 2004.

[38] P. Häfliger and C. Rasche, "Floating gate analog memory for parameter and variable storage in a learning silicon neuron," in *Proc. IEEE Int. Symp. Circuits Syst.*, 1999, vol. 2, pp. 416–419.

[39] J. V. Arthur and K. Boahen, "Silicon neurons that inhibit to synchronize," in *Proc. IEEE Int. Symp. Circuits Syst.*, 2006, pp. 4807–4810.

[40] G. Indiveri, E. Chicca, and R. Douglas, "A VLSI array of low-power spiking neurons and bistable synapses with spike-timing dependent plasticity," *IEEE Trans. Neural Netw.*, vol. 17, no. 1, pp. 211–221, Jan. 2006.

[41] P. H. E. Tiesinga, J. V. Jose, and T. J. Sejnowski, "Comparison of current-driven and conductance-driven neocortical model neurons with Hodgkin-Huxley voltage-gated channels," *Phys. Rev. E*, vol. 62, no. 6, pp. 8413–8419, 2000.

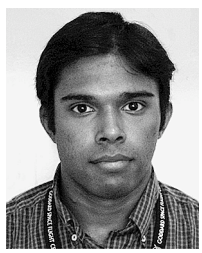
[42] G. L. Camera, W. Senn, and S. Fusi, "Comparison between networks of conductance- and current-driven neurons: Stationary spike rates and subthreshold depolarization," *Neurocomput.*, vol. 58–60, pp. 253–258, 2004.

[43] R. Jolivet, A. Rauch, H.-R. Lüscher, and W. Gerstner, "Predicting spike timing of neocortical pyramidal neurons by simple threshold models," *J. Comput. Neurosci.*, vol. 21, no. 1, pp. 35–39, 2006.
[44] A. L. Hodgkin and A. F. Huxley, "A quantitative description of membrane current and its application to conduction and excitation in nerve," *J. Physiol.*, vol. 117, pp. 500–544, 1952.
[45] R. Mirollo and S. Strogatz, "Synchronization of pulse-coupled biological oscillators," *SIAM J. Appl. Math.*, vol. 50, no. 6, pp. 1645–1662, 2000.
[46] F. Tenore, R. Etienne-Cummings, and M. A. Lewis, "Entrainment of silicon central pattern generators for legged locomotory control," in *Advances in Neural Information Processing Systems*, S. Thrun, L. Saul, and B. Schölkopf, Eds. Cambridge, MA: MIT Press, 2004, vol. 16.
[47] A. Kumar, S. Rotter, and A. Aertsen, "Propagation of synfire activity in locally connected networks with conductance-based synapses," presented at the Conf. Comput. Syst. Neurosci. (COSYNE), Salt Lake City, UT, Mar. 5–8, 2006.
[48] Z. F. Mainen and T. J. Sejnowski, "Reliability of spike timing in neocortical neurons," *Science*, vol. 268, no. 5216, pp. 1503–1506, 1995.
[49] W. M. Usrey, "The role of spike timing for thalamocortical processing," *Current Opinion Neurobiol.*, vol. 12, no. 4, pp. 411–417, 2002.
[50] D. Fricker and R. Miles, "Interneurons, spike timing, and perception," *Neuron*, vol. 32, no. 5, pp. 771–774, 2001.
[51] M. Abeles, *Corticonics: Neural Circuits of the Cerebral Cortex*. London, U.K.: Cambridge Univ. Press, 1991.
[52] F. Rieke, D. Warland, R. de Ruyter van Stevenick, and W. Bialek, *Spikes: Exploring the Neural Code*, 1st ed. Cambridge, MA: MIT Press, 1999.
[53] J. G. Elias, "Artificial dendritic trees," *Neural Comput.*, vol. 5, pp. 648–663, 1993.
[54] R. Gregorian and G. C. Temes, *Analog MOS Integrated Circuits for Signal Processing*. New York: Wiley, 1986.
[55] K. A. Boahen, "A burst-mode word-serial address-event channel—I: transmitter design," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 51, no. 7, pp. 1281–1291, Jul. 2004.
[56] R. J. Vogelstein, U. Mallik, and G. Cauwenberghs, manuscript in preparation.
[57] R. J. Vogelstein, U. Mallik, G. Cauwenberghs, E. Culurciello, and R. Etienne-Cummings, "Saliency-driven image acuity modulation on a reconfigurable silicon array of spiking neurons," in *Advances in Neural Information Processing Systems*, L. K. Saul, Y. Weiss, and L. Bottou, Eds. Cambridge, MA: MIT Press, 2005, vol. 17, pp. 1457–1464.
[58] J. T. Vogelstein and K. Zhang, "A novel theory for simultaneous representation of multiple dynamic states in hippocampus," in *2004 Abstract Viewer/Itinerary Planner*. Washington, DC: Society for Neuroscience, 2004, Program No. 710.1.
[59] J. T. Vogelstein and K. Zhang, manuscript in preparation.
[60] J. O'Keefe and J. Dostrovsky, "The hippocampus as a spatial map. Preliminary evidence from unit activity in the freely-moving rat," *Brain Res.*, vol. 34, no. 1, pp. 171–175, 1971.
[61] J. O'Keefe and L. Nadel, *The Hippocampus as a Cognitive Map*. New York: Oxford Univ. Press, 1978.
[62] K. Zhang, I. Ginzburg, B. L. McNaughton, and T. J. Sejnowski, "Interpreting neuronal population activity by reconstruction: Unified framework with application to hippocampal place cells," *J. Neurophysiol.*, vol. 79, pp. 1017–1044, 1998.
[63] E. J. Markus, C. A. Barnes, B. L. McNaughton, V. L. Gladden, and W. E. Skaggs, "Spatial information content and reliability of hippocampal CA1 neurons: Effects of visual input," *Hippocampus*, vol. 4, no. 4, pp. 410–421, 1994.
[64] E. Save, L. Nerad, and B. Poucet, "Contribution of multiple sensory information to place field stability in hippocampal place cells," *Hippocampus*, vol. 10, no. 1, pp. 64–76, 2000.
[65] M. Tsodyks and T. Sejnowski, "Associative memory and hippocampal place cells," *Int. J. Neural Syst.*, vol. 6, pp. 81–86, 1995.
[66] M. Tsodyks, "Attractor neural network models of spatial maps in hippocampus," *Hippocampus*, vol. 9, no. 4, pp. 481–489, 1999.
[67] A. Samsonovich and B. L. McNaughton, "Path integration and cognitive mapping in a continuous attractor neural network model," *J. Neurosci.*, vol. 17, no. 15, pp. 5900–5920, 1997.
[68] M. Mattia and P. Del Giudice, "Efficient event-driven simulation of large networks of spiking neurons and dynamical synapses," *Neural Comput.*, vol. 12, no. 10, pp. 2305–2329, 2000.
[69] M. Rudolph and A. Destexhe, "Analytical integrate-and-fire neuron models with conductance-based dynamics for event-driven simulation strategies," *Neural Comput.*, vol. 18, no. 9, pp. 2146–2210, 2006.
[70] A. Morrison, C. Mehring, T. Geisel, A. D. Aertsen, and M. Diesmann, "Advancing the boundaries of high-connectivity network simulation with distributed computing," *Neural Comput.*, vol. 17, no. 8, pp. 1776–1801, 2005.
[71] V. Torre and T. Poggio, "A synaptic mechanism possibly underlying directional selectivity to motion," *Proc. Roy. Soc. London. Ser. B*, vol. 202, pp. 409–416, 1978.
[72] C. Koch, T. Poggio, and V. Torre, "Retinal ganglion cells: A functional interpretation of dendritic morphology," *Proc. Roy. Soc. London. Ser. B*, vol. 298, pp. 227–264, 1982.
[73] M. Carandini and D. J. Heeger, "Summation and division by neurons in primate visual cortex," *Science*, vol. 264, pp. 1333–1336, 1994.
[74] M. E. Nelson, "A mechanism for neuronal gain control by descending pathways," *Neural Comput.*, vol. 6, pp. 242–254, 1994.
[75] U. Mallik, R. J. Vogelstein, E. Culurciello, R. Etienne-Cummings, and G. Cauwenberghs, "A real-time spike-domain sensory information processing system," in *Proc. IEEE Int. Symp. Circuits Syst.*, Kobe, Japan, 2005, pp. 1919–1922.
[76] M. Mattia and P. D. Giudice, "Efficient event-driven simulation of large networks of spiking neurons and dynamical synapses," *Neural Comput.*, vol. 12, no. 10, pp. 2305–2329, 2000.
[77] R. J. Vogelstein, U. Mallik, E. Culurciello, G. Cauwenberghs, and R. Etienne-Cummings, "A multi-chip neuromorphic system for spike-based visual information processing ," , 2006, submitted for publication.
[78] E. Salinas and T. J. Sejnowski, "Integrate-and-fire neurons driven by correlated stochastic input," *Neural Comput.*, vol. 14, no. 9, pp. 2111–2155, 2002.
[79] A. K. Engel, P. Fries, and W. Singer, "Dynamic predictions: oscillations and synchrony in top-down processing," *Nature Rev. Neurosci.*, vol. 2, pp. 704–716, 2001.
[80] S. Song, K. D. Miller, and L. F. Abbott, "Competitive Hebbian learning through spike-timing-dependent synaptic plasticity," *Nature Neurosci.*, vol. 3, no. 9, pp. 919–926, 2000.
[81] R. J. Vogelstein, U. Mallik, and G. Cauwenberghs, "Spatial acuity modulation of an address-event imager," in *Proc. IEEE Int. Conf. Electron., Circuits Syst.*, Tel Aviv, Israel, 2004, pp. 207–210.

R. Jacob Vogelstein (S'99) received the Sc.B. degree in bioelectrical engineering from Brown University, Providence, RI, in 2000 and he is currently working towards the Ph.D. degree in biomedical engineering at The Johns Hopkins University, Baltimore, MD.

His research focuses on neuromorphic and neuroprosthetic devices, and his papers have been finalists in the student paper competitions at the 2004 IEEE International Conference of the Engineering in Medicine and Biology Society and the 2004 IEEE International Conference on Electronics, Circuits and Systems.

Mr. Vogelstein was inducted into the Tau Beta Pi and Sigma Xi honor societies in 1999, won the Brown University Engineering Department's Outstanding Student Award in 2000, and received a National Science Foundation Graduate Research Fellowship in 2002.



Udayan Mallik received the B.S. degree in electrical engineering from The Johns Hopkins University, Baltimore, MD, in 2003.

He is presently with Goddard Space Flight Center, NASA, Greenbelt, MD.

Joshua T. Vogelstein was born in Baltimore, MD, on February 16, 1980. He received the B.S. degree in biomedical engineering from Washington University, St. Louis, MO, in 2002 and is currently working towards the Ph.D. degree at the Department of Neuroscience, The Johns Hopkins University, Baltimore, MD.

He is the author of several peer-reviewed scientific publications, and has presented his work at several conferences. His current research interests include two-photon microscopy, models of memory, neural systems identificaton, sensory coding, and neural plasticity.

Gert Cauwenberghs (S'89–M'94–SM'04) received the Ph.D. degree in electrical engineering from California Institute of Technology, Pasadena, in 1994.

He was a Professor of Electrical and Computer Engineering at The Johns Hopkins University, Baltimore, MD. He joined the University of California at San Diego, La Jolla, as a Professor of Neurobiology, in 2005. His research interests are in advancing silicon adaptive microsystems to the understanding of biological neural systems, and development of sensory and neural prostheses and brain–machine interfaces.

Dr. Cauwenberghs received the National Science Foundation (NSF) Career Award in 1997, the Office of Naval Research (ONR) Young Investigator Award in 1999, and Presidential Early Career Award for Scientists and Engineers in 2000. He is an Associate Editor of IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS I, IEEE TRANSACTIONS ON NEURAL SYSTEMS AND REHABILITATION ENGINEERING, and IEEE SENSORS JOURNAL.