A Floating-Gate MOS Learning Array with Locally Computed Weight Updates

Chris Diorio, Student Member, IEEE, Paul Hasler, Student Member, IEEE, Bradley A. Minch, Student Member, IEEE, and Carver A. Mead, Fellow, IEEE

Abstract—We have demonstrated on-chip learning in an array of floating-gate MOS synapse transistors. The array comprises one synapse transistor at each node, and normalization circuitry at the row boundaries. The array computes the inner product of a column input vector and a stored weight matrix. The weights are stored as floating-gate charge; they are nonvolatile, but can increase when we apply a row-learn signal. The input and learn signals are digital pulses; column input pulses that are coincident with row-learn pulses cause weight increases at selected synapses. The normalization circuitry forces row synapses to compete for floating-gate charge, bounding the weight values. The array simultaneously exhibits fast computation and slow adaptation: The inner product computes in 10 μs, whereas the weight normalization takes minutes to hours.

I. INTRODUCTION

Our goal is to develop silicon learning systems. We believe that these systems must possess the following attributes: high device density; low power consumption; fast, parallel computation; and slow, local adaptation. We build our learning systems as integrated circuits, achieving high device density by using MOS IC technology, effecting low power consumption by using subthreshold channel currents, and performing the requisite computations and adaptation by using innate features of the silicon-MOS physics.

We began our investigations by building single-transistor silicon synapses [1]–[5] modeled loosely after biological synapses [6]. Our synapse transistors are floating-gate MOSFETs; they possess nonvolatile analog weight storage, compute locally from the product of their stored weight and an applied control-gate input, permit simultaneous computation and weight modification, and determine locally their own weight updates. We select source current as the synapse output, store the weights as floating-gate charge, and achieve bidirectional learning by using a combination of electron tunneling and hot-electron injection to modify the floating-gate charge.

Because our synapse transistors comprise a single device, and employ subthreshold channel currents, we can use them to build dense, low-power, silicon learning systems. Although a single transistor cannot model the complex behavior of a neural synapse completely, our synapse transistors can learn from an input signal without interrupting the ongoing computation.

In this paper, we demonstrate on-chip learning in a 4 × 4 array of our four-terminal nFET synapse transistors. We show the array block diagram in Fig. 1. The input vector comprises 10-μs pulses; the array computes the inner product of this input vector and the stored analog weight matrix. The weights are nonvolatile; column input pulses that are coincident with row-learn pulses cause weight increases at selected synapses. To prevent unbounded weight values, we enforce a constraint: The time-averaged sum of the row-synapse weights to be a constant, bounding the row weights by forcing the synapses to compete for weight value.

Fig. 1. The learning-array block diagram. For clarity, we show only 2 × 2 of the 4 × 4 synapses in the array. Each synapse multiplies its column input with its stored analog weight, and outputs a current to the row-output wire; the row wire sums the synapse-output currents along the row. The stored weights are nonvolatile; column inputs that are coincident with row-learn signals cause weight increases at selected synapses. The error signal constrains the time-averaged sum of the row-synapse weights to be a constant, bounding the row weights by forcing the synapses to compete for weight value.

II. THE nFET SYNAPSE TRANSISTOR

We begin by reviewing our four-terminal nFET synapse transistor. As we show in Fig. 2, this device is an n-type...
floating-gate MOSFET, to which we add a fourth terminal for gate-oxide tunneling. We operate the synapse from a single-polarity supply, use Fowler–Nordheim (FN) tunneling [7] to remove electrons from the floating gate, and use channel hot-electron injection (CHEI) [8] to add electrons to the floating gate. We fabricate the synapse in a 2-μm n-well CMOS process (with NPN option) available from MOSIS.

A. The Synapse Stores a Weight

We select source current as the synapse output. We apply signal inputs to the poly2 control gate, which, in turn, couples capacitively to the poly1 floating gate. We operate the MOSFET in the subthreshold regime [9], for three reasons. First, subthreshold channel currents ensure low power consumption. Second, because a subthreshold MOSFET’s source current increases exponentially with gate voltage, only small quantities of oxide charge are required for learning. Third, the synapse output is the product of the stored weight and the control-gate input, as we derive from the subthreshold MOSFET equation

\[
I_s = I_0 e^{\kappa V_{th}/V_t} = \frac{I_0 e^{\kappa Q_{fg}/C_{m}V_{in}}}{C_T U_{th}}
\]

where \( I_s \) is the source current, \( I_0 \) is the pre-exponential current, \( \kappa \) is the coupling coefficient from the floating gate to the channel, \( Q_{fg} \) is the floating-gate charge, \( C_T \) is the total capacitance seen by the floating gate, \( U_{th} \) is the thermal voltage \( kT/q \), \( C_m \) is the input (poly1 to poly2) coupling capacitance, \( V_{in} \) is the control-gate voltage, \( Q_T \equiv C_T U_{th}/\kappa \), and, for simplicity, we have assumed the source potential to be ground (\( V_s = 0 \)). The synapse weight \( W \) is the learned quantity: Its value derives from the floating-gate charge, which can change with synapse use. The synapse output is the product of \( W \) and the source current of an idealized MOSFET that has a control-gate input \( V_{in} \) and a coupling coefficient \( \kappa' \) from the control gate to the channel.

B. Electron Tunneling Increases the Weight

We increase \( W \) by tunneling electrons off the floating gate. In Fig. 3, we show the tunneling gate current (the oxide current) versus the reciprocal of the voltage across the tunneling oxide. We fit these data with an FN fit [7], [10]

\[
I_g = I_{to} e^{-V_{th}/V_{ox}}
\]

where \( I_g \) is the gate current; \( V_{th} \) is the oxide voltage; \( V_f = 984 \) V is consistent with a recent survey [11] of SiO\(_2\) tunneling, given the synapse transistor’s 400 Å gate oxide; and \( I_{to} \) is a pre-exponential current.

The present synapse requires large tunneling voltages, because the gate-oxide thickness is 400 Å. Synapses fabricated in more modern processes with thinner oxides have much lower tunneling voltages. In addition, at lower voltages, the well implant that we use for tunneling can be replaced with a graded implant, reducing the synapse size.

C. CHEI Decreases the Weight

We decrease \( W \) by injecting electrons onto the floating gate. To permit CHEI with subthreshold channel currents, we add a bulk p-type implant to the synapse transistor’s channel region. This implant serves two functions. First, it increases the peak drain-to-channel electric field, thereby increasing the hot-electron population in the drain-to-channel depletion region.
We define $V_{ox}$ to be the potential difference between the $n^+$ tunneling implant and the floating gate. We fit the data using a conventional Fowler–Nordheim expression. We normalized the data to the tunneling-junction gate-to-$n^+$ edge length, in lineal microns, because the floating gate induces a depletion region in the lightly doped n-well, reducing the effective oxide voltage and with it the tunneling current. Because the gate cannot deplete the $n^+$ well contact appreciably, the oxide field is higher where the self-aligned floating gate overlaps the $n^+$. Because $I_g$ increases exponentially with $V_{ox}$, gate-oxide tunneling in the synapse transistor is primarily an edge phenomenon.

When $V_{dc}$ is greater than 2.5 V, the CHEI gate current causes measurable changes in the synapse weight $W$. For reasons that we discuss in Section V, $V_{dc}$, in this application, typically is less than 3 V, and always is less than 3.5 V. Consequently, we approximate the data of Fig. 4 with a simple exponential

$$I_g = \beta I_s e^{V_{dc}/V_{inj}}$$  \hspace{1cm} (4)

where $I_g$ is the gate current; $I_s$ is the source current; $V_{dc}$ is the drain-to-channel potential; and $\beta$, $V_{inj}$ are fit constants.

As a consequence of the synapse transistor’s 6 V threshold, the floating-gate voltage usually exceeds 5 V, and the drain-to-gate oxide electric field strongly favors the transport of injected electrons to the floating gate. The CHEI efficiency therefore is, to first order, independent of the gate-to-channel potential, and we model the CHEI process using only (4).

### D. Synapse Weight Updates Follow a Power Law

A synapse’s weight updates derive from the tunneling and CHEI oxide currents that alter the floating-gate charge. Because these oxide currents vary with the synapse’s terminal voltages and source current, $W$ varies with the terminal voltages, which are imposed on the device, and with the source current, which is the synapse output. Consequently, the synapse learns: Its future output depends on both the applied input and the present output.

In Fig. 5, we show the temporal derivative of the source current versus the source current, for a synapse transistor with (part A) a set of fixed tunneling voltages, and (part B) a set of fixed drain voltages. In both experiments, we held the control-gate input $V_{in}$ fixed; consequently, these data show the synapse weight updates $\partial W/\partial t$, as can be seen by differentiating (2).

In Appendix A, we show that the tunneling-induced weight increments follow a power law

$$\frac{\partial W}{\partial t} = \frac{1}{\tau_{tun}} W^{(1-\sigma)}$$  \hspace{1cm} (5)

where we define $\sigma$ and $\tau_{tun}$ in (16) and (17), respectively. In Appendix B, we show that the CHEI-induced weight decrements also follow a power law

$$\frac{\partial W}{\partial t} = -\frac{1}{\tau_{inj}} W^{(2-\varepsilon)}$$  \hspace{1cm} (6)

where we define $\varepsilon$ and $\tau_{inj}$ in (26) and (27), respectively.

### III. THE LEARNING ARRAY

In Fig. 6, we show one row of the learning array, comprising a synapse transistor at each array node and a normalization circuit at the row boundary. The column inputs $X_i$ and the row-learn signals $Y_j$ are 10 μs digital pulses. Each synapse multiplies its binary-valued input $X_i$ with its stored weight $W_{ij}$, and outputs a source current $I_{sij}$ whose magnitude is given by (2). The total row current $I_{out}$ is the sum of the source currents from all the synapses in the row. Synapses ordinarily are on; low-true gate inputs $X_i$ turn off selected synapses, decreasing the current $I_{out}$ transiently. This decrease in $I_{out}$, in response to an input vector $X$, is the row computation.
and decreases, and is true (\( \text{where} \), are true. To see why, we first s input pulses is low; is large, . In equilibrium, the is high; as follows: As-
is , when both is time invariant and we assume that times longer than the in (2) with its temporal into capacitor is low, causing forces . Consequently, is small, the . If a low-
se, causing is true, then to rise; is false ( is too small to cause appreciable also falls, and the row returns to equilibrium. The typically causes little or no CHEI in the to fall. As rises, all the row synapses undergo ;
i(9) follows . When
Fig. 5. Synapse-transistor (A) tunneling and (B) CHEI weight updates. In both experiments, we measured the synapse’s source current \( I_s \) versus time, and plotted \( \partial I_s / \partial t \) versus \( I_s \). We fixed the synapse’s terminal voltages; consequently, the change in \( I_s \) is a result of changes in the synapse’s weight \( W \). In part A, we applied \( V_{in} = 5 \text{ V, } V_c = 0 \text{ V, } V_{in} = 2 \text{ V, and stepped } V_{tun} \text{ from 29 to 35 } \text{ V in 1 V increments}; \) in part B, we applied \( V_{in} = 5 \text{ V, } V_c = 0 \text{ V, } V_{in} = 20 \text{ V, and stepped } V_{out} \text{ from 2.9 to 3.5 } \text{ V in 0.1 V increments. We turned off the tunneling and CHEI at regular intervals, to measure } I_s \). Because, for a fixed \( V_{in} \), the synapse’s weight updates \( \partial W / \partial t \) are proportional to \( \partial I_s / \partial t \) [see (2)], these data show that the weight updates follow a power law. The mean values of \( (\sigma) \) and \( (\varepsilon) \) are 0.17 and 0.24, respectively.

Synapse-weight increases occur only when both the row and column inputs, \( Y_j \) and \( X_i \), are true. To see why, we first consider the case when the row learn signal \( Y_j \) is false (\( V_{tun} \) is low). Because \( V_{ox} \equiv V_{tun} - V_{fg} \), when \( V_{tun} \) is low, \( V_{ox} \) is small for every synapse in the row. When \( V_{ox} \) is small, the tunneling currents are small, and there is no weight increase at any row synapse.

Now we consider the case when \( Y_j \) is true (\( V_{tun} \) is high). \( V_{ox} \) increases as \( V_{fg} \) decreases, and \( V_{fg} \) follows \( X_i \). If a low-
true column input \( X_i \) is true, then \( V_{fg} \) is low; \( V_{ox} \) is large, and electron tunneling causes a weight increase at the selected synapse. If, on the other hand, the low-true column input \( X_i \) is false, then \( V_{fg} \) is high; \( V_{ox} \) is too small to cause appreciable tunneling, and there is little change in the synapse’s weight.

Tunneling increases the weight value of a row-column selected synapse. Because this weight update is single quadrant, tunneling allows unbounded weight increases. To constrain the array-weight values, we renormalize the weights in each row of the array. Our array affords unsupervised learning [14], with the following constraint: The sum of the row-synapse weights, averaged over time, is a constant. The array error metric is a weight normalization; we use CHEI feedback along each row of the array to enforce the constraint.

IV. WEIGHT NORMALIZATION

The weight-normalization circuit (see Fig. 6) compares \( I_{tun} \), the sum of the synapse drain currents in a row, with \( I_b \), the bias current in transistor \( M_3 \); if \( I_{tun} > I_b \), then the circuit uses CHEI to renormalize the weights. To explain the renormalization, we begin by defining row equilibrium: A row is in equilibrium when \( I_{tun} = I_b \). In equilibrium, the drain voltage \( V_d \) typically causes little or no CHEI in the row synapses.

The normalization circuit constrains \( I_{tun} \) as follows: Assume that the row initially is in equilibrium, and that tunneling then raises the weight values of selected synapses, increasing \( I_{tun} \). The excess drain current \( (I_{tun} - I_b) \) is mirrored by \( M_2 \) and \( M_3 \) into capacitor \( C_{tun} \), causing \( V_c \) to rise; \( Q_d \) forces \( V_d \) to follow \( V_c \). When \( V_d \) rises, all the row synapses undergo CHEI, decreasing all the weights, causing \( I_{tun} \) to fall. As \( I_{tun} \) falls, \( V_d \) also falls, and the row returns to equilibrium. The drain-current constraint requires that, over time, \( I_{tun} = I_b \). The normalization circuit creates a negative resistance at the synapses’ common drain node, causing \( V_d \) to rise when \( I_{tun} \) increases.

We now show how the drain-current constraint renormalizes the synapse weights. We begin with the constraint

\[
\sum_i I_{si} \approx \sum_i I_{di} \equiv I_{sum} = I_b.
\]  

In Section V, we show that the renormalization time constant \( \tau_{ren} \) exceeds 10 s; this value is 10⁶ times longer than the 10-μs input pulses \( X_i \) (where \( V_{in} = X_i \)). Consequently, for renormalization, we replace \( V_{in} \) in (2) with its temporal average \( \overline{V}_{in} \) and we assume that \( V_{in} \) both is time invariant and has the same value for all the row synapses. Substituting (2) into (7), we have

\[
\sum_i W_i I_o e^{t \overline{V}_{in}/U_i} = I_o e^{t \overline{V}_{in}/U_i} \sum_i W_i = I_b
\]  

\[
\Rightarrow \sum_i W_i = \frac{I_b}{I_o} e^{t \overline{V}_{in}/U_i} \equiv W_{sum} = \text{constant}.
\]  

The drain-current and weight-value constraints are equivalent; consequently, row feedback renormalizes the synapse weights.

Renormalization forces the row synapses to compete for floating-gate charge; when one synapse’s weight value increases, the sum of the weight values of its row neighbors must decrease by the same amount. However, when a selected synapse tunnels, increasing its weight, renormalization forces all the row synapses to undergo CHEI, decreasing all the row-synapse weights. The selected synapse undergoes both tunneling and CHEI; because the exponent in the CHEI weight-update rule is larger than that in the tunneling rule [see
Fig. 6. One row of the learning array. The column input vector $X$ comprises low-true, 5 V, 10-μs digital pulses; the row input vector $Y$ comprises high-true, 12 V, 10-μs digital pulses. Because the 2-μm CMOS process that we use has 400 Å gate oxides, the tunneling voltages are high; to cause measurable tunneling, we superimpose the row inputs onto a 25 Vdc bias. The voltage coupling between a synapse’s control and floating gates is about 0.8. Consequently, a 5 V (low-true) input on column wire $X_1$ causes a 4 V decrease in syn1’s floating-gate voltage, which, in turn, causes a 4 V increase in syn1’s tunneling-oxide voltage. A column input $X_1$ that is coincident with a row-learn pulse $Y_1$ causes a 16 V increase in the tunneling-oxide voltage at syn1, but only a 12 V increase at the other synapses. Because electron tunneling increases exponentially with tunneling-oxide voltage (see Fig. 3), syn1’s floating gate receives about 100 times more charge than do the other synapses’ floating gates; because charge increases exponentially with floating-gate charge [see (2)], syn1’s weight increases much more than do the other synapses’ weights. The weight increase causes $I_{\text{sum}}$ to rise, which, in turn, causes the normalization circuit to raise $V_d$. Because the CHEI efficiency increases with $V_{\text{cm}}$ (see Fig. 4), a higher $V_d$ causes CHEI in all the synapses, decreasing all the weights. The array eventually settles back to equilibrium, with $I_{\text{sum}}$ equal to $I_0$, but syn1 now takes a larger share of the total row current, and the other synapses each take a smaller share. The inverting amplifier in the weight-normalization circuit enhances loop stability, for reasons that we discuss in Section V.

(5) and (6)], renormalization constrains a synapse’s weight-update rate, in addition to its weight value.

Tunneling and CHEI effectively redistribute a fixed quantity of floating-gate charge among the row synapse transistors. In Appendix C, we derive the array learning rule, for coincident $(x, y)$ pulse inputs to synapse $j$

$$W_i, j(n + 1) = W_i, j(n) - f_{\text{learn}}W_i(n)(2^{-\epsilon})$$

$$W_j(n + 1) = W_j(n) + f_{\text{learn}} \sum_i W_i(n)(2^{-\epsilon})$$

where we define $\epsilon$ and $f_{\text{learn}}$ in (26) and (36), respectively. In Figs. 7 and 8, we show unsupervised learning in one row of our $4 \times 4$ array; these data highlight both the synapse weight and the update-rate constraints. We fit the data by applying (10) and (11), recursively; the only inputs to the fit equations are the synapse weights at $n = 0$ and the fit constants $\tau_{\text{turn}}, I_{\text{pre}}, \sigma$, and $\epsilon$.

V. NORMALIZATION-CIRCUIT STABILITY

The normalization circuit creates a negative resistance at the synapses’ common drain node: When $I_{\text{sum}}$ increases, $V_d$ rises. The loop output is $V_d$, and the loop feedback comprises CHEI oxide currents: When $V_d$ rises, CHEI decreases the synapse weights, causing $I_{\text{sum}}$ to fall. Because the CHEI oxide currents increase exponentially with $V_d$, the loop dynamics are highly nonlinear. We therefore describe qualitative, rather than quantitative, loop-stability criteria.

The normalization circuit employs positive feedback; to ensure stability, we must make the loop gain less than unity for all frequencies. This requirement implies that the small-signal impedance $z_d$, looking into the synapse drain terminals, must be greater than the total impedance $z_c$, at capacitor $C_{\text{int}}$.

To see why, we assume instead that $z_c > z_d$. A rising $V_d$ induces a small-signal current $i_d = z_c V_d$. CHEI is mirrored by $M_2$ and $M_3$ into $C_{\text{int}}$, causing $V_c$ to rise by an amount $u_c = i_{\text{sum}} z_c$. Because $V_d$ follows $V_c$, if $z_c > z_d$, then $u_c > u_d$; $i_{\text{sum}}$ will increase rapidly, causing $V_c$ to rise toward $V_{\text{dc}}$.

The impedance $z_d$ is limited by interconnect capacitances, and by synapse-transistor channel-length modulation, floating-gate-to-drain overlap capacitance, and drain-current impact ionization. We consider each of these limitations in turn.

A. Interconnect Capacitance

Interconnect capacitance at the synapses’ common drain node causes $z_d$ to decrease with frequency. We choose $C_{\text{pre}}$ to be much larger than this parasitic capacitance, so the reactive impedance ratio, $z_c/z_d$, favors loop stability for all frequencies.

B. Channel-Length Modulation

Channel-length modulation reduces a synapse’s drain impedance, limiting $z_d$. Fortunately, the synapse transistor’s Early voltage exceeds 100 V, as a result of both the 10 μm channel length and the p-type channel implant; consequently, the channel-length modulation is small.

C. Floating-Gate-to-Drain Overlap Capacitance

$V_d$ couples to a synapse transistor’s floating gate, by means of the floating-gate-to-drain overlap capacitance $C_{\text{df}}$. The coupling coefficient is $C_{\text{df}}/C_T$, where $C_T$ is the total floating-gate
Fig. 7. Array learning behavior, with fits. We initialized all synapses to the same source-current value prior to starting the experiment. We first applied a train of coincident \( (x, y) 10-\mu s \) pulses to synapse 1, causing its weight value and source current to increase. Renormalization caused the weight values and source currents of the other synapses to decrease. Once synapse 1 had acquired 90% of the total row current, we removed the pulse-train stimulus and instead applied it to synapse 2, and then, in turn, to synapses 3 and 4. We measured the synapse source currents after every 10-\( \mu s \) input pulses. In the lower half of the figure, we highlight the first 1600 data points, and fit these data by applying (10) and (11), recursively. The inputs to the fit equations are the initial synapse source-current values (at \( n = 0 \)), the pulsewidth \( t_{pw} = 10 \mu s \); and the empirical constants \( \gamma_{syn}, \sigma, \) and \( \epsilon \). These data show that we can address individual synapses with good selectivity, and can achieve wide separation in the weight values of selected versus deselected synapses.

capacitance. Because \( I_{sh} \) increases exponentially with \( V_{fg} \), \( C_{lg} \) causes \( I_{sum} \) to increase exponentially with \( V_d \), limiting \( z_d \). To minimize the effect, we use a large interpoly capacitor \((C_T = 1 \text{ pF})\); we also apply inverting feedback from \( V_d \) to the floating gate, increasing \( z_d \) (see Fig. 6). We use an off-chip amplifier to generate this inverting feedback; in future arrays, we will use instead our on-chip adaptive floating-gate amplifier [15].

D. Drain-Current Impact Ionization

Channel electrons that posses sufficient energy for CHEI also posses sufficient energy for impact ionization [16], [17]. In the synapse transistor, a drain-to-channel electric field that causes CHEI also creates additional electron-hole pairs, causing \( I_d \) to increase exponentially with \( V_{lg} \). As a result, \( I_{sum} \) increases exponentially with \( V_d \), limiting \( z_d \). If \( V_d \) becomes greater than about 4 V, the rate of drain-current increase causes loop instability, and \( V_d \) rises rapidly. As \( V_d \) rises, CHEI decreases all the synapse-transistor weights; as \( V_d \) saturates near \( V_{dl} \), CHEI causes \( I_{sum} \) to fall below \( I_{sh} \), causing \( V_d \) to fall, and the loop to return to a stable operating regime. Loop instability causes \( V_d \) to undergo a single brief \((\sim 10 \mu s)\) voltage spike, and reduces all the synapse weights substantially. Fortunately, because the synapse CHEI efficiency is high, weight renormalization rarely causes \( V_d \) to exceed 3.5 V; consequently, the loop is stable.

In Fig. 9, we show the normalization-circuit impedance versus frequency; in Fig. 10, we show the circuit’s impulse response. Although the low-frequency time constant \( \tau_a \) (the adaptation time constant) decreases as \( V_d \) increases, \( \tau_a \) typically exceeds 10 s. The loop impulse response shows that, for short timescales, the total drain current \( I_{sum} \) can exceed \( I_{sh} \), violating the normalization constraint; for long timescales, \( I_{sum} = I_{sh} \).

The parasitic coupling between a synapse’s tunneling junction and its floating gate is about 5 fF. With \( C_T = 1 \text{ pF} \), a 12 V row-learn pulse \( Y_j \) increases the floating-gate voltage of every row synapse by about 60 mV. This coupling does not affect the row computation significantly, for two reasons. First, 5 V low-true column inputs \( X_i \) always turn off selected synapses, regardless of \( Y_j \). Second, because row-learn pulses \( Y_j \) increase the floating-gate voltage of every deselected synapse by a fixed 60 mV, we can calculate the corresponding source-current increase using (1), and can adjust \( I_{out} \) accordingly.

VI. CONCLUSION

We have shown simultaneous computation and unsupervised learning in a \( 4 \times 4 \) array of nFET synapse transistors. The array computes the inner product of an input vector and a stored analog weight matrix. The array weights are nonvolatile; coincident row and column input pulses cause weight increases at selected synapses. We constrain the time-averaged sum of the row-synapse weights to be constant, forcing row synapses to compete for weight value.
Fig. 9. Normalization-circuit impedance magnitude versus frequency. We applied a small-signal sinusoidal current $i_{in}$ to the synapses’ row-drain node (see Fig. 6), measured the resulting small-signal voltage $v_{dr}$, and plotted $z_d = v_{dr} / i_{in}$. Because the loop feedback comprises CHEI oxide currents, which increase exponentially with $V_d$, the low-frequency corner increases with $V_d$. To hold this corner at a single frequency, we applied a constant $V_{lin} = 37$ V to all the row-synapse transistors, causing continuous tunneling. The normalization loop re-established equilibrium by setting $3.3$ V, inducing continuous CHEI to compensate the continuous tunneling. For these (artificial) operating conditions, the low-frequency corner comprises a single pole at about 0.03 Hz. The high-frequency rolloff comprises two poles: The first is the normalization-loop response, set by $C_{int}$; the second is a consequence of interconnect capacitance at the synapses’ common drain node, attenuating our injected signal $i_{in}$.

The array computation and synapse-weight modification occur locally and in parallel. The array achieves our goals of fast, single-transistor analog computation and of slow, locally computed weight adaptation. We describe the array computation and learning behavior using rules derived directly from the silicon-MOS and silicon-oxide physics.

SiO$_2$ trapping is a well-known issue in floating-gate transistor reliability [18]; in the synapse, oxide trapping decreases the weight-update rates. Fortunately, because our synapses require only small quantities of charge for their weight updates, we can ignore oxide trapping in the learning array safely.

Finally, although our array affords unsupervised learning, it uses a feedback error signal to constrain the weight values. Feedback error signals typically are used in supervised neural networks, to adjust the array weights according to the network learning rule. In future floating-gate arrays, rather than using unsupervised learning, we intend to use CHEI to adjust the synapse weights in a supervised fashion, using either pulsed, or continuously valued analog [19], inputs and row-error signals.

### APPENDIX A

#### The Tunneling Weight-Increment Rule

We begin by taking the temporal derivative of the synapse weight $W$, where $W \equiv \exp(Q_{Gf}/kT)$:

$$\frac{\partial W}{\partial t} = \frac{W}{Q_T} \frac{\partial Q_{Gf}}{\partial t} = \frac{W}{Q_T} I_g.$$  \hspace{1cm} (12)

We substitute (3) for the gate current $I_g$:

$$\frac{\partial W}{\partial t} = \frac{I_{to}}{Q_T} W e^{-V_f/V_{ox}}.$$  \hspace{1cm} (13)

We substitute $V_{ox} = V_{lin} - V_{Gf}$ (where $V_{lin}$ and $V_{Gf}$ are the tunneling-implant and floating-gate voltages, respectively), assume that $V_{lin} \gg V_{Gf}$, expand the exponent using $(1 - x)^{-1} \approx 1 + x$, and solve

$$\frac{\partial W}{\partial t} \approx \frac{I_{to}}{Q_T} W e^{-(V_f/V_{lin})-(V_f/V_{lin})}.$$  \hspace{1cm} (14)

We substitute $V_{Gf} = U_i Q_{Gf}/kQ_T$, and solve for the tunneling weight-increment rule

$$\frac{\partial W}{\partial t} \approx \frac{1}{\tau_{lin}} W(1-\sigma)$$  \hspace{1cm} (15)

where

$$\sigma \equiv \frac{V_f U_i}{k V_{lin}^2}$$ \hspace{1cm} (16)

and

$$\tau_{lin} \equiv \frac{Q_T}{I_{to}} e^{V_f/V_{lin}}.$$  \hspace{1cm} (17)

The parameters $\sigma$ and $\tau_{lin}$ vary with the tunneling voltage $V_{lin}$. 

---

Fig. 10. Normalization-circuit impulse response. At time $t = 10$ min, we applied $2 \times 10^5$ coincident $(x, y)$ 10 µs pulses, over a 10 s period, to synapse 1. We plotted (A) the synapse source currents, and (B) the drain voltage $V_d$, for a period of about 5 h following the stimulus. Because the low-frequency loop time constant $\tau_a$ changes with $V_d$, the loop settling does not follow an RC decay; rather, the decay is closer to $1/t$. After 2 weeks, $V_d$ was about 2.4 V. At time $t = 0$, $V_d$ initially was decaying, because we had just finished resetting the synapse source currents to identical values.

We substitute $V_{ox} = V_{lin} - V_{Gf}$ (where $V_{lin}$ and $V_{Gf}$ are the tunneling-implant and floating-gate voltages, respectively), assume that $V_{lin} \gg V_{Gf}$, expand the exponent using $(1 - x)^{-1} \approx 1 + x$, and solve

$$\frac{\partial W}{\partial t} \approx \frac{I_{to}}{Q_T} W e^{-V_f/V_{lin}}.$$  \hspace{1cm} (14)

We substitute $V_{Gf} = U_i Q_{Gf}/kQ_T$, and solve for the tunneling weight-increment rule

$$\frac{\partial W}{\partial t} \approx \frac{1}{\tau_{lin}} W(1-\sigma)$$  \hspace{1cm} (15)

where

$$\sigma \equiv \frac{V_f U_i}{k V_{lin}^2}$$ \hspace{1cm} (16)

and

$$\tau_{lin} \equiv \frac{Q_T}{I_{to}} e^{V_f/V_{lin}}.$$  \hspace{1cm} (17)

The parameters $\sigma$ and $\tau_{lin}$ vary with the tunneling voltage $V_{lin}$.
APPENDIX B

The CHEI Weight-Decrement Rule

We begin by defining a synapse transistor’s drain-to-channel potential, $V_{dc}$, in terms of $V_{ds}$ and $I_s$. In a subthreshold floating-gate MOSFET, the source current is related to the floating-gate and source voltages [9] by

$$I_s = I_0 e^{(\kappa V_{fg} - V_s)/U_t}$$  \hspace{1cm} (18)

and the channel-surface potential, $\Psi$, is related to the floating-gate voltage, $V_{fg}$ [12], [13] by

$$\Psi \approx \kappa V_{fg} + \Psi_0$$  \hspace{1cm} (19)

where $\kappa$ is the coupling coefficient from the floating gate to the channel, and $\Psi_0$ derives from the MOS process parameters.

Using (18) and (19), we solve for the surface potential $\Psi$ in terms of $I_s$ and $V_s$

$$\Psi = V_s + \Psi_0 + U_t \ln \left( \frac{I_s}{I_0} \right).$$  \hspace{1cm} (20)

We now solve for $V_{dc}$

$$V_{dc} = V_d - \Psi = V_d - \Psi_0 - U_t \ln \left( \frac{I_s}{I_0} \right).$$  \hspace{1cm} (21)

The CHEI gate current $I_g$ is given by (4). We add a minus sign to $I_g$, because CHEI decreases the floating-gate charge, and substitute for $V_{dc}$ using (21)

$$I_g = -\beta I_0 e^{(V_{inj} - \Psi_0)/U_t} \frac{\ln(I_s/I_0)}{V_{inj}}$$

$$= -\beta I_0 e^{(V_{inj} - \Psi_0)/V_{inj}} \left[ 1 - \left( \frac{I_s}{I_0} \right) \right].$$  \hspace{1cm} (22)

We substitute for $I_s$ using (2), and solve

$$I_g = -\beta I_0 e^{(V_{inj} - \Psi_0)/V_{inj}} \left( V_{inj} - \Psi_0 \right) \left( 1 - \left( \frac{I_s}{I_0} \right) \right).$$  \hspace{1cm} (23)

We substitute (23) into $\partial W/\partial t$, (12)

$$\frac{\partial W}{\partial t} = -\frac{\beta I_0}{Q_T} e^{(V_{inj} - \Psi_0)/V_{inj}} \left( 1 - \left( \frac{I_s}{I_0} \right) \right) W(2 - \left( \frac{I_s}{I_0} \right))$$

to get the final weight-decrement rule

$$\frac{\partial W}{\partial t} = -\frac{1}{\tau_{inj}} W(2 - \varepsilon)$$  \hspace{1cm} (24)

where

$$\varepsilon \equiv \frac{U_t}{V_{inj}}$$  \hspace{1cm} (25)

and

$$\tau_{inj} \equiv \frac{Q_T}{\beta I_0} e^{-(V_{inj} - \Psi_0)/V_{inj}}.$$  \hspace{1cm} (26)

The low-true column-input ($X_d$) pulse duty cycle typically is small, so $V_{inj}$ normally is high ($V_{inj} = X_d$). We therefore assume that $V_{inj}$ is a constant ($V_{inj} = S V$) in (27).

APPENDIX C

The Array Learning Rule

We consider the row-synapse weights at discrete time intervals $t \equiv nT$, where $n$ is the step number and $T$ is the timestep, and derive the row-learning rule for a single coincident ($x, y$) input to a single row synapse. We begin with the equilibrium condition for the row-weight normalization

$$\sum_i W_i(n) = W_{sum}.$$  \hspace{1cm} (28)

We assume that the normalization time constant $\tau_o$ is fixed, for the following reason: Coincident ($x, y$) input pulses cause a weight increase at a synapse; the normalization circuit responds by establishing a drain voltage $V_d$ for which the total weight decay, summed over all the row synapses, balances the weight increase at the single synapse. If we assume that the mean density of the coincident input pulses is time-invariant, then $V_d$’s mean value, $\overline{V_d}$, is constant, and therefore the low-frequency loop time constant, $\tau_o$, also is constant.

We assume that $\tau_o \ll T$. The synapse weight values can violate (28) for times $t \ll \tau_o \ll T$, but we require that they satisfy (28) at our measurement time intervals $t = nT$. We permit array inputs at times $(t + \delta t) \equiv (n + \delta) T$; immediately after we measure the synapse weight values at $t = nT$. The array inputs comprise a pulsed column vector $X(n + \delta)$, where $X_i \in [0, 1] \equiv [\bar{3} \ V, 0 \ V]$, and a pulsed row vector $Y(n + \delta)$, where $Y_j \in [0, 1] \equiv [0 \ V, 12 \ V]$. Without loss of generality, we assume that at time $t = nT$; the circuit is in equilibrium, and that $(t + \delta t + t_{pw}) \equiv (n + \delta_{pw}) T$; coincident row and column inputs, of duration $t_{pw}$, have caused synapse $j$’s weight to increase

$$W_j(n + \delta_{pw}) \approx W_j(n) + \frac{\partial W_j(n)}{\partial t} t_{pw}$$

$$\approx W_j(n) + \frac{t_{pw}}{\tau_{inj}} W_j(n)^{(1 - \varepsilon)}$$  \hspace{1cm} (29)

where in (29) we have made the first-order approximation that $\partial W/\partial t$ is constant over $t_{pw}$, and in (30) we have substituted for $\partial W/\partial t$ using (5). Because $t_{pw} \ll \tau_o$, at time $(n + \delta_{pw}) T$ the circuit no longer is in equilibrium

$$\sum_i W_i(n + \delta_{pw}) > W_{sum}$$  \hspace{1cm} (30)

and the synapse weights inject down to reestablish equilibrium.

We wish to find the synapse weights at $(n + 1)$, when the row again satisfies (28). Using (25) and (30), we write weight-decrement expressions for the row synapses

$$\Delta W_i, i \neq j(n + 1)$$

$$= -\frac{T}{\tau_{inj}} W_i, i \neq j(n)^{(2 - \varepsilon)}$$  \hspace{1cm} (31)

$$\Delta W_j(n + 1)$$

$$\approx -\frac{T}{\tau_{inj}} \left[ W_j(n) + \frac{t_{pw}}{\tau_{inj}} W_j(n)^{(1 - \varepsilon)} \right]^{(2 - \varepsilon)}$$  \hspace{1cm} (32)

where, because the row drain voltage $V_d$ settles during renormalization, $\tau_{inj}$ may vary over $T$ (recall that $T \gg \tau_o \gg t_{pw}$). For reasonable values of $\tau_{inj}$ and $t_{pw}$, the weight increment from a single coincident ($x, y$) input is small; consequently, we can simplify (33) using $(1 + \varepsilon)^n \approx 1 + n \varepsilon$

$$\Delta W_j(n + 1)$$

$$\approx -\frac{T}{\tau_{inj}} W_j(n)^{(2 - \varepsilon)} \left[ 1 + (2 - \varepsilon) \frac{t_{pw}}{\tau_{inj}} W_j(n)^{-\varepsilon} \right].$$  \hspace{1cm} (33)

Because $\tau_{inj}$ varies over $T$, we now re-express $T/\tau_{inj}$ in terms of quantities that we know at $n$. We equate the weight
increment at synapse $j$ [see (30)] to the sum of the weight decrements at synapses $i, i \neq j$ (32) and $j$ (34)

$$
\frac{f_{\text{pw}}}{\tau_{\text{lin}}} W_i(n)(1-\varepsilon)
= \frac{T}{\tau_{\text{lin}}} \sum_{i, i \neq j} W_i(n)(2-\varepsilon)
+ \frac{T}{\tau_{\text{lin}}} W_j(n)(2-\varepsilon) \left[ 1 + (2-\varepsilon) \frac{f_{\text{pw}}}{\tau_{\text{lin}}} W_j(n)^{-\varepsilon} \right]
$$

(35)

and we solve for $T/\tau_{\text{lin}}$:

$$
T = \frac{(2-\varepsilon) f_{\text{pw}}}{\tau_{\text{lin}}} W_j(n)(2-\varepsilon) + \sum_i W_i(n)(2-\varepsilon).
$$

(36)

We define $f_{\text{learn}} \equiv T/\tau_{\text{lin}}$; substitute $f_{\text{learn}}$ into (32), and use (28) to solve for the row-learning rule

$$
W_i(\tau(n)+1) = W_i(\tau(n)) + f_{\text{learn}} \sum_{i, i \neq j} W_i(n)/(2-\varepsilon)
$$

(10)

$$
W_j(\tau(n)+1) = W_j(n) + f_{\text{learn}} \sum_{i, i \neq j} W_i(n)/(2-\varepsilon).
$$

(11)

Equations (10) and (11) describe the row weight-update rule for a single coincident $(x, y)$ pulse input to synapse $j$.

**REFERENCES**


