

**BIO-INSPIRED, BIO-COMPATIBLE,
RECONFIGURABLE ANALOG CMOS CIRCUITS**

A Dissertation
Presented to
The Academic Faculty

by

Christal Gordon

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
Electrical and Computer Engineering

Georgia Institute of Technology
December 2009

**BIO-INSPIRED, BIO-COMPATIBLE,
RECONFIGURABLE ANALOG CMOS CIRCUITS**

Approved by:

Professor David V. Anderson,
Committee Chair
Electrical and Computer Engineering
Georgia Institute of Technology

Professor Paul E. Hasler, Advisor
Electrical and Computer Engineering
Georgia Institute of Technology

Professor Robert J. Butera, Jr.
Electrical and Computer Engineering
Georgia Institute of Technology

Professor Gary May
Electrical and Computer Engineering
Georgia Institute of Technology

Professor Dieter Jaeger
Department of Biology
Emory University

Date Approved: May 18, 2009

I have been on long road to get to this point. This work is dedicated to all those who have helped me along the way. Unfortunately, not everyone was able to complete this journey with me.

Farewell

Carl Gordon

Pearl Jackson

Evelyn Ray

Elaine Campbell

Thankfully, a few more folks have joined me along the way.

Welcome

Kayla Appleton

Ahmaya Robinson

Shanaya Barrows

Amaya Allerdyce

Elijah Robinson

Jamere Barrows

ACKNOWLEDGEMENTS

Thank you to those who did without so that I could have something.

I heartily thank my mother for all of her love, sacrifice, and dedication.

I would like to thank my advisor and academic father Paul Hasler. He has supported me through all of the bright and rough patches in my academic career.

Thanks to my committee; their mentorship has provided beneficial insight.

My family and friends have been a source of unending support.

There have been many teachers who inspired me to reach further and know more.

Thank you all very much. Without you, I would not be where I am.

TABLE OF CONTENTS

DEDICATION	iii
ACKNOWLEDGEMENTS	iv
LIST OF FIGURES	vii
I OVERVIEW	1
II BIOLOGY PRIMER	7
2.1 Biological Model	7
2.2 Ion Channels	10
2.3 Spike-Timing Dependent Plasticity	11
III FLOATING GATE PRIMER	13
3.1 Basics	13
3.2 Injection	14
3.3 Tunneling	17
3.4 Array Programming	18
3.5 Indirect Programming	19
IV ARTIFICIAL NEURAL CIRCUITS	30
4.1 Models of Synapses	30
4.1.1 Excitatory Synapse	31
4.1.2 Inhibitory Synapse	35
4.1.3 NMDA Excitatory Synapse	37
4.1.4 Analysis of the Synapses	38
4.2 Field-Programmable Neural Array	41
4.2.1 Architecture	41
4.2.2 Applications	44
4.2.3 Improvements	44

V	METHODS OF LEARNING	46
5.1	Previous Work	46
5.2	Experiment	49
5.3	Data and Discussion	50
5.3.1	A Greatly Increasing Rule	51
5.3.2	A Tuned Rule	53
5.3.3	A Rule that can Increase and Decrease	54
5.3.4	Creating Additional Rules	54
5.4	Effect of Ca^{2+}	56
VI	NEURAL INTERFACING	64
6.1	Neural Amplifier	66
6.1.1	Design Considerations	66
6.1.2	Discussion	68
6.1.3	Resistive Element	69
6.2	Interfacing Experiment	74
6.2.1	Linking An Artificial Neuron to a Living Neuron	74
6.2.2	Linking Living Neurons Through Electronics	75
6.2.3	Effect of Different Synaptic Weights	75
6.3	Charge Balancing	76
6.3.1	Corrosion	76
6.3.2	pH	77
6.3.3	System	77
VII	CONCLUSION	89
7.1	Learning	90
7.2	Interfacing	90
7.3	Processing	91
7.4	Final Synopsis	92
VITA	94

LIST OF FIGURES

1	The biological synapse is the main communication port for the nervous system. Here, an electrical response in the first (pre-synaptic) cell causes chemicals to leave and be deposited into the second (post-synaptic) cell. These chemicals, called neurotransmitters, then cause an electrical response in the second cell. The silicon synapse has been highlighted in red to illustrate how it functionally relates to the biological synapse.	3
2	Other forms of biological connections. (a) A neuromuscular junction linking a motoneuron to a muscle fiber. The pre-synaptic cell and the post-synaptic cells are different types of cells. The neurotransmitter here is acetylcholine. Note the undulating surface of the muscle fibers. This surface creates a higher surface area that contains ion channels. (b) An electronic synapse greatly differs from typical cellular communication mechanisms since they use connexions instead of ion channels.	4
3	A system that can connect living neurons to each other using bio-inspired, bio-compatible circuits has been created. This system can be seen as a bio-inspired medium in which artificially or biologically created signals can be processed and then sent out to artificial or biological devices or cells. The system is represented as the IC, which contains three portions: an amplifier, a field programmable neural array (FPNA), and output synapses. This work looks at the zeroth and first-order cases of connecting an artificial neuron to a living neuron and connecting two living neurons, with no extra processing done by the FPNA.	5
4	Visual explanation of a learning rule.	6
5	Graphical representations of different learning rules. These rules can depend upon the time at which inputs and outputs occur, the type of neuron, and the position of the synapse on the neuron. (Adapted from Abbott and Nelson)	6
6	Examples of ion channels found in the nervous system.	10
7	Pedagogical illustration of a floating gate.	14
8	Injection in a $0.6\mu m$ process. Note that the rate of change in the current plateaus.	21

9	The rate of injection in a $0.6\mu\text{m}$ process. The injection rate changes depending upon the value of V_{sd} and the present current value. There are three main phases, increasing, maximum, and decreasing injection rate. Note that the noise in the decreasing phase is due to fluctuations the least significant digit in the picoammeter.	22
10	(a) Schematic of negative Dickson charge pump. The pFETs act as diodes. The clock signals (ϕ and ϕ') are complements. The waveforms illustrated are for the ideal case in which there is no droop due to the capacitor discharging. (b) Simulated data from the negative charge pump. V_{in} and the wells of the pFETs are at ground. The capacitors are 1 pF and the clock is 500 kHz. Note that at this capacitor size and clock frequency, there is no droop.	23
11	Illustration of band diagram of tunneling through a MOS capacitor. The shading at the gate indicates the probability of an electron to move through the barrier.	23
12	Tunneling in a $0.6\mu\text{m}$ process. Note that there is a delay in the onset of tunneling. This delay is inversely proportional to V_{tun} . The delay has not been modeled in the literature. Although this effect is not fully understood, it may be due to the behavior of the MOS capacitor at the tunneling junction. Since the capacitance value of MOS capacitors is dependent upon the voltage, the device may be moving through different regions of operation depending on V_{tun} . The tunneling voltages range from 13.5V to 14.5V in increments of 0.1V.	24
13	Tunneling in a $0.35\mu\text{m}$ process. Note that there is no delay in the onset of tunneling as in the $0.6\mu\text{m}$ process. The tunneling voltages range from 11 to 14V in 0.2V increments.	25
14	Change in tunneling rate for different pulse widths in a $0.35\mu\text{m}$ process at 13V. An tunneling input of 13V and pulse width of t_{tun} was presented. The length of t_{tun} varied from 10ms to 0.1ms. The current began at $2\mu\text{A}$ and was allowed to drop to at least 90nA or for 100 iterations. The current through the floating-gate pFET was measured after each pulse.	26
15	Natural log of the rate of change in tunneling in a $0.35\mu\text{m}$ process for different pulse widths. The natural log is taken since the rate of change is proportional to e^{ttun} where t_{tun} is the time that tunneling is on. Therefore, a line is created when the natural log of the rate is plotted. The tunneling voltage V_{tun} is 13V in this experiment. . . .	27

16	Programming of a matrix of floating gates. V_{DD} has been set to some value that would allow for a large enough V_{DS} for injection, such as 6 V. The floating gate at row 1, column 2 is isolated by putting all of the other drains (rows 0 and 2) to V_{DD} and the other gates (columns 0,1, and 3) to V_{DD} . The drain of row 1 is dropped to some desired level, such as ground, and the gate is sent to another level, such as 3 V. Figure originally from Kucic.	28
17	Figures of indirect programming structures. These structures allow for the adjustment of floating gate charge without the use of switches. (a) Indirect programming structure for a pFET. (b) Indirect programming structure for an nFET.	28
18	Data from indirectly programmed FETs as reported in the indirect programming paper.	29
19	Data from indirectly programmed FETs illustrating the effect of capacitance values as reported in the indirect programming paper. . . .	29
20	A typical PSP that would be found in biology modeled from Rall's alpha function on top with the log of the alpha function at the bottom. The right illustrates the steps taken in developing the synapse. Important functions, nodes, and waveforms are highlighted in red. . .	32
21	An increase in the frequency of the signal results in the aggregation of basic EPSPs, as seen in biological data.	33
22	1214.5=1212(a) The input of the basic excitatory circuit is an inverter with an output that has been slowed down by the capacitor to . The net synaptic strength is proportional to the charge on the floating gate. Like its biological analogue, this synaptic strength can be modified. (b) Changing in an excitatory synapse. In biology, decreasing the membrane voltage decreases the size of the basic EPSP. Decreasing the voltage across the synapse channel transistor decreases the size of the EPSP. In this case, =1.259 and =2.238. (c) The rate at which the initial, upgoing side of the basic EPSP increases by increasing . Note that the inset, a plot of the voltage on the input of the STLS, illustrates only the the beginning slope changing with a change in voltage. Here, varies and =2.080.(d) The rate at which second, downgoing side of the basic EPSP increases by increasing . Note that the inset, a plot of the voltage on the input of the STLS, illustrates only the the second slope changing with a change in voltage. Here, =1.994 and varies.	34

23	1214.5=1212(a) The inhibitory synapse is similar to the basic excitatory, with the exception of the location of the output terminal. The floating gate source draws current, rather than having the drain provide current. Thus, we have an inhibiting effect on any subcircuit which this synapse is connected. (b) As expected, these inhibitory results are basically the mirror of the excitatory synapses, where this IPSP decreases for decreasing voltage across the STLS in this voltage clamp experiment. Here, $\tau = 1.867$ and $\tau = 2.109$	35
24	1214.5=1212(a) Like the biological NMDA synapse, this circuit's response depends on the membrane voltage as well as the carrier flow. Here, we use a β amplifier to mimic the effect of β . (b) As expected, the NMDA voltage clamp experiment results are similar to the basic excitatory synapse where the EPSP decreases, then goes negative for decreasing voltage across the STLS. Here $\tau = 1.259$ and $\tau = 2.238$. (c) The rate at which initial, upgoing side of the NMDA EPSP increases by increasing V_n . Note the inset, a plot of the voltage on the input of the STLS, illustrates only the the beginning slope changing with a change in voltage. Here τ varies and $\tau = 2.080$. (d) The rate at which second, downgoing side of the NMDA EPSP increases by increasing V_p . Note the inset, a plot of the voltage on the input of the STLS, illustrates only the the second slope changing with a change in voltage. Here $\tau = 1.777$ and τ varies.	36
25	Schematic of the Field Programmable Neural Array (FPNA).	42
26	1214.5=1212(a) Layout for the first version of the FPNA. This was fabricated by MOSIS on a AMI 0.5 μ m process. This chip is 1.5 mm x 1.5 mm and has 40 pins. (b) Layout for the second version of the FPNA. This was fabricated by MOSIS on a TSMC 0.35 μ m process. This chip is 3 mm x 3 mm and has 64 pins.	43
27	Networks that can be created using the FPNA. (a) One complex cell. (b) Two less complex cells forming a simple CPG (half-coupled oscillator). (c) Many simple cells connected in a complex network.	44

28	1214.5=1212(a) State-space analysis of learning rule. Two cases of simultaneous tunneling and injection are explored. As illustrated, equilibrium points can be shifted to desired locations by varying the tunneling and injection voltages. Furthermore, the trajectories of the weights can be determined for a given initial condition. The arcsin function is used to magnify the detail of the data. (b) Experimental measurements from a 3 synapse system, where one synaptic input (Synapse 2) is visited significantly more than the remaining 2 inputs. Weight adaptation at a particular synapse is dependent upon the occurrence of input pulses with output pulses. With an increase in synaptic activity, the probability of input and output pulse-overlap also increases. The overall network adaptation rate is dependent upon the tunneling and injection voltages. We illustrate the case where we have a combination of LTP and LTD occurring in the system.	47
29	Basic floating-gate circuit for an action-potential learning system. Our synapses are a single floating-gate pFET device with a DIBL transistor for stability. The neuron element is an integrate and fire neuron that can be easily replaced with a more biologically accurate model. The feedback circuitry moves the source (V_s) and tunneling (V_{tun}) voltages after the onset of an action potential. The current from each synapse is summed along a wire; extending this circuit to biological type dendrites would be straightforward.	48
30	Plot of the inputs to the single transistor learning synapse.	50
31	A learning rule that increases the weight most efficiently due to the current value through the floating-gate. This form of the rule takes place when there is no tunneling at all. Note that the peak change in synaptic weight takes place for a current that is near the threshold.	52
32	Change in synaptic weight per iteration. The timing differences from $\Delta t = -1$ to 4ms are represented with different shapes. The rate of change in current is large for a difference of -1, 0, and 4ms. Thus, these timing differences produce a relatively large synaptic weight change every iteration. Timing differences of 1, 2, and 3ms result in negligible changes in current, therefore producing a slow synaptic weight change. Note that timing differences -1, 0, and 4ms produce currents that begin to plateau since they have reached absolute values that are above threshold.	58
33	A learning rule that increases the weight most efficiently due to the current value through the floating-gate. This rule takes place if the tunneling pulse only occurs when the gate is off and the injection pulse is near the time that the current is near threshold. Note that the translucent red indicates when the tunneling pulse is on.	59

34	A learning rule with a tuned response. This response is tuned by the timing of the tunneling pulse. Note that the translucent red indicates when the tunneling pulse is on.	60
35	A rule that can change the synaptic weight positively and negatively. Note that the translucent red indicates when the tunneling pulse is on.	61
36	A rule that increases synaptic weight when there is no input pulse. Note that the translucent red indicates when the tunneling pulse is on.	62
37	Modified ACh synapse which uses indirect programming and the concentration of Ca^{2+} . The subcircuit which models the effect of Ca^{2+} on LTP is highlighted with a dotted box around it.	63
38	1214.5=1212(a) Neural amplifier based on the work of Harrison and Charles. The capacitor values are = 100 fF and = 10 pF. (b) The wide-range amplifier component of the neural amplifier.	66
39	1214.5=1212(a) Power spectral density plots of extracellular noise recordings from an Aplysia cell. The results are similar despite the great difference in size between the rack-mounted Brownlee amplifier and the amplifier IC. The integrated amplifier has an ideal gain of 100, while the Brownlee amplifier has a gain set to 750. In order to better compare typical readings with the amplifier IC, an external gain of 7.5 was added by connecting the output of the chip to another Brownlee input. (b) Superimposed spikes from the amplified extracellular recordings. The solid line is a spike amplified using the rack-mounted Brownlee amplifier and the dotted line is a spike amplified by the IC. The spikes are taken from different recordings and are not the same signal being amplified.	67
40	Examples of different multiplexing schemes. Note that each figure is one cell out of an array of cells.	69
41	CMOS based resistive elements.	70
42	CMOS based resistive element with programmable MOS and BJT ends.	70
43	MOS versus BJT operation of the simple CMOS based resistive element.	72
44	I-V curve for the simple resistive element near the operation point. .	73

45	I-V curve for the simple resistive element. Note that this data set is limited to the range of the picoammeter used. The curve fit here illustrates what the resistive element current would be if it were operating in subthreshold and there were no experimental limitations. The normal operation range is limited to $\pm 0.2V$. Using another resistive elements in series helps to expand the range to $\pm 0.4V$. The exponential rate of change for the BJT side is calculated as 37.715 while the rate for the MOS side is only -26.455.	80
46	I-V relationship through each node the simple resistive element. Note that the current through the substrate was calculated since there was no way to measure the substrate current in this process. For empirical data, a triple-well process is needed to measure the current due to the resistive element that is flowing through the p-substrate.	81
47	R-V curve for the simple resistive element. The normal operation range is limited to $\pm 0.2V$. Using another resistive elements in series helps to expand the range to $\pm 0.4V$. Even with this limited operating range, the effective resistance is in the gigaohm range. The effective resistance was calculated using Ohm's Law on the original data collected from the picoammeter.	82
48	R-V curve for the resistive element with a modification to the MOS side of the device. Note that this data set is limited to the range of the picoammeter used. The maximum that the picoammeter could output is $2\mu A$, so the values above $2/\mu A$ should be regarded as well above $2\mu A$. The slight slope is due to using Ohm's Law with a constant current and changing voltage.	83
49	R-V curve for the resistive element with a modification to the BJT side of the device.	84
50	R-V curve for the fully modified resistive element.	85
51	The full setup uses electronics to link two cells that are not normally connected. Cell 1 is the presynaptic cell and Cell 2 is the postsynaptic cell. Ideally, the amplifiers and converters would not be necessary. However, we have initially included them. Note the switch between the I-V converter and the V-I converter that can be used to isolate the circuitry from Cell 2. All of the necessary processing is done by the circuitry. Other test have used an on-chip amplifier to amplify the signal from Cell 1.	85
52	Effect of linking an artificial neuron to a living neuron with an artificial synapse. Note that the switch is disconnected from time 0 to 2.5 s and connected from 2.5 s onward.	86

53	Effect of linking two living neurons through an artificial neuron and an artificial synapse.	87
54	1214.5=1212(a) Output of silicon synapse for various synaptic weights. (b) Output of silicon synapse in logarithmic scale. Note that the slopes for the curves are all the same, yet the maximum values differ. (c) Output of Cell 2 (postsynaptic cell) for given synaptic weights. Note that the third and smallest input did not elicit an output response since the input did not reach the threshold of the cell.	87
55	Block diagram of charge balancing system.	88

CHAPTER I

OVERVIEW

Biological systems are often compact, robust, efficient, and can solve many common engineering problems. This work seeks to emulate the power of these systems for use in solving such problems and modeling the neurobiology of the brain. Two living neurons have been successfully connected using electronics that are based on biological systems. The key circuit, an electronic synapse, is compact in size, can easily operate in parallel, and provides signals that match the biological data [1], [2]. Figure 1 illustrates the inspiration for and the schematic of the electronic synapse. Furthermore, this artificial synapse can also be adapted to reproduce other types of biological signaling mechanisms. For instance, a neuromuscular junction can easily be emulated in the same way as a chemical synapse, as seen in Figure 2(a). With adjustments to the main biases, this same structure can be used to emulate an electronic synapse, as seen in 2(b).

Figure 3 is a representation of the ideal system that can link various types of biological and artificial systems with a neural version of a field-programmable gate array (FPGA) called a field-programmable neuron array (FPNA). Rather than connecting digital gates to create digital systems with digital switching networks, the FPNA connects electronic models of ion channels, dendrite sections and axons to create neurons of various levels of complexity with analog synapses. Besides connecting neurons to inputs and each other in the FPNA, there are synapses that connect the internal connections to the external world. The inputs will ideally come from a variety of systems; however, the most difficult ones to receive are extracellular voltage recordings. An amplifier of gain A can be used to increase the signal to levels that

can be distinguished by the FPNA. After processing in the FPNA, the output signal would then be sent to the next system. On-chip signal conditioning for computers and actuators is fairly straight-forward; however, bio-compatible output signals are not. The electronic synapse was created to generate signals that are of the same size and shape as biological electrical signals.

Hebbian learning is an obvious extension of these experiments. The output of the living neurons would be used to change the synaptic strengths. Using Cell 1, the presynaptic cell, as an input to the synapse and the architecture described by Gordon and Hasler [1], the weight between the input and output cells can be modified. The firing of Cell 1 would select a particular floating gate for a weight change, while Cell 2, the postsynaptic cell, provides the input for the feedback circuitry, which pulses the value of V_{ds} and V_{tun} , as described in [1]. This form of learning is dependent upon the timing of the inputs and outputs. This learning is called spike-timing dependent plasticity (STDP). STDP has been found in systems ranging from the hippocampus, barrel cortex, and visual cortex [3], [4], [5], [6], [7], [8], [9], [10], [11]. Synapses that are modified by STDP are not limited to Hebbian learning rules. Other learning rules have been found in neurons that show only a decrease in weight, anti-hebbian behavior, and a change in the learning rule at different parts along a dendritic branch [12], [13], [4]. Many descriptions of how the synaptic weight changes exists. This description of how the synaptic weight can change is known as a learning rule. In order to understand learning rules found in STDP systems, simple graphs as shown in Figure 4 are created. Furthermore, as shown in Figure 5, many methods of synaptic weight change exist in biology.

The signal compatibility between this system and real cells would allow for better prosthetics. For instance, extracellular signals can be read from motor cortex, processed on an FPNA, and then output to motoneurons. The replacement of function, particularly signal transduction, can easily be done even without an FPNA. For

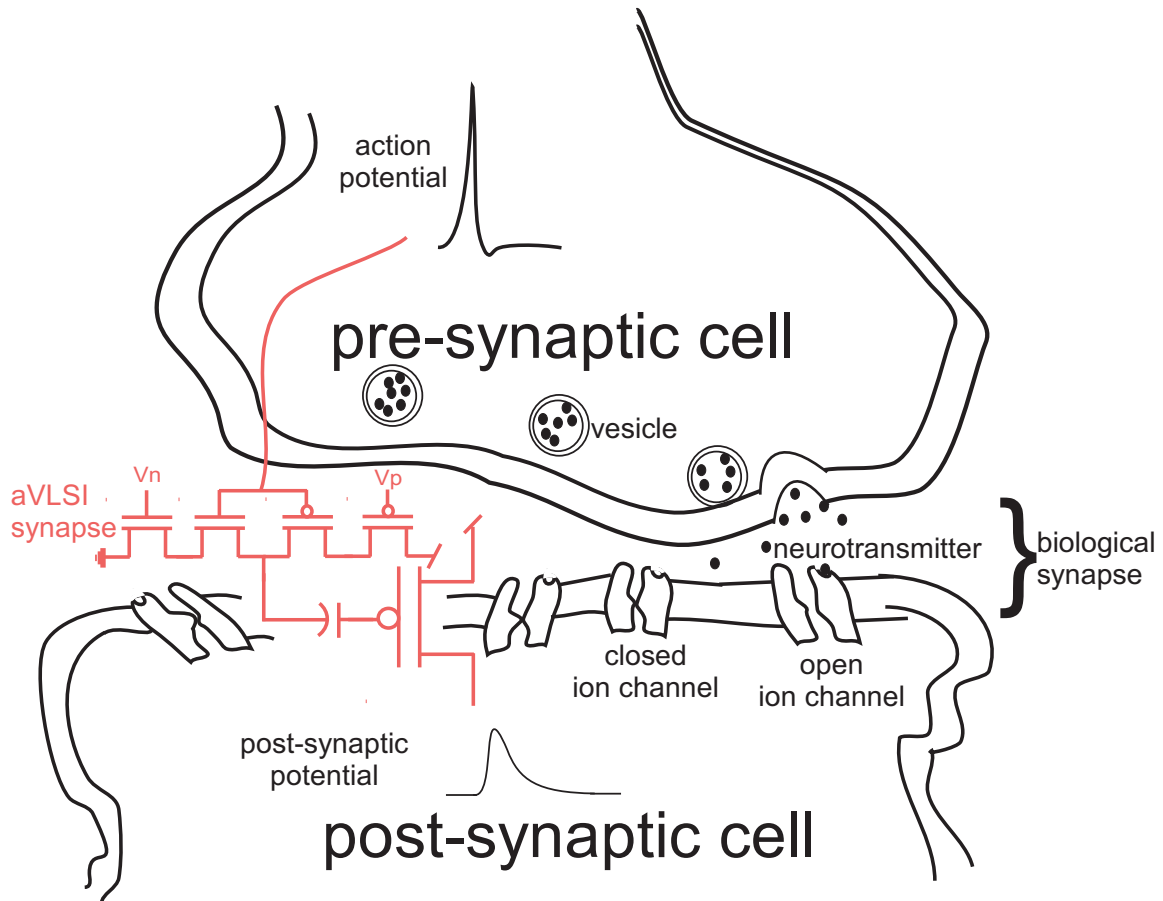


Figure 1: The biological synapse is the main communication port for the nervous system. Here, an electrical response in the first (pre-synaptic) cell causes chemicals to leave and be deposited into the second (post-synaptic) cell. These chemicals, called neurotransmitters, then cause an electrical response in the second cell. The silicon synapse has been highlighted in red to illustrate how it functionally relates to the biological synapse.

instance, a damaged section of the auditory nerve can be reconnected by using this small, low-power, bio-compatible system. Multiplexing circuitry can be added to help ameliorate pin-limiting applications such as receiving dozens of inputs that are then mapped to dozens of outputs.

Future work will incorporate more complex processing, which will allow the FPNA to create systems ranging from large networks of simple cells to a complex model of a single neuron. This system will be effective whether the systems are programmed

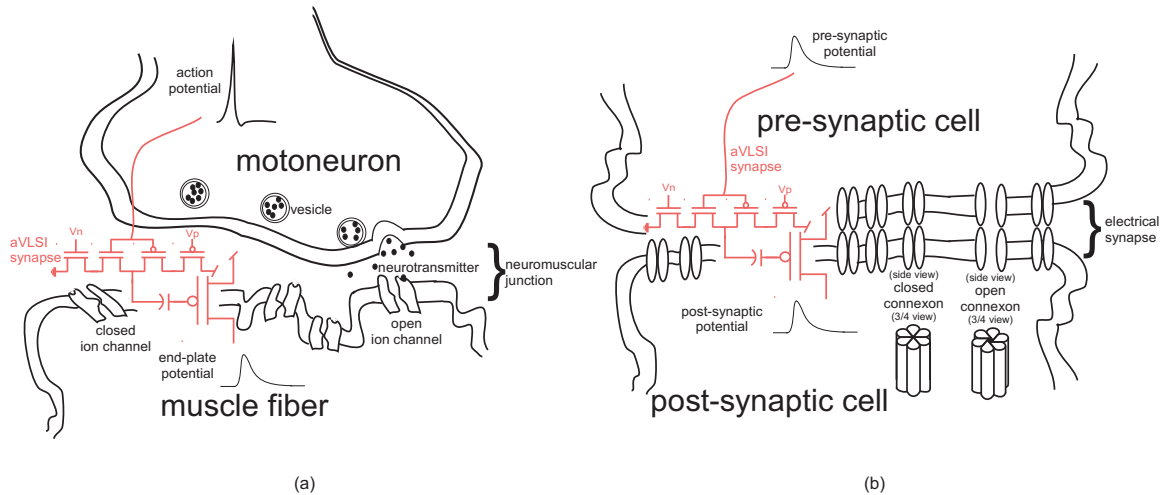


Figure 2: Other forms of biological connections. (a) A neuromuscular junction linking a motoneuron to a muscle fiber. The pre-synaptic cell and the post-synaptic cells are different types of cells. The neurotransmitter here is acetylcholine. Note the undulating surface of the muscle fibers. This surface creates a higher surface area that contains ion channels. (b) An electronic synapse greatly differs from typical cellular communication mechanisms since they use connexions instead of ion channels.

or adapt their synaptic weights, use biologically inspired reconfigurable circuitry or just a simple neuron model, or connect two or 200 cells.

These electronic synapses are an effective representation of biological synapses. As in biology, there is a high density of unique connections that can be made. This differs greatly from other models which simplify realistic multi-valued synapses to binary synapses [14] or create the multiple values for synapses through large, power-hungry look-up tables and routing [15]. The electrical signals that these artificial synapses produce are the same as the electrical signals produced by biological synapses. This is quite different from other models which produce outputs that are similar, but do not match biological data [16]. The utility of these synapses is greatly increased by incorporating them in a system that allows for a true connection to living cells. Thus, a link that can effectively be modulated by changing the charge on the floating-gate has been created, a true electronic synapse. Presently, no other form of electronic synapse can emulate the function of biological synapses as well as the ones described

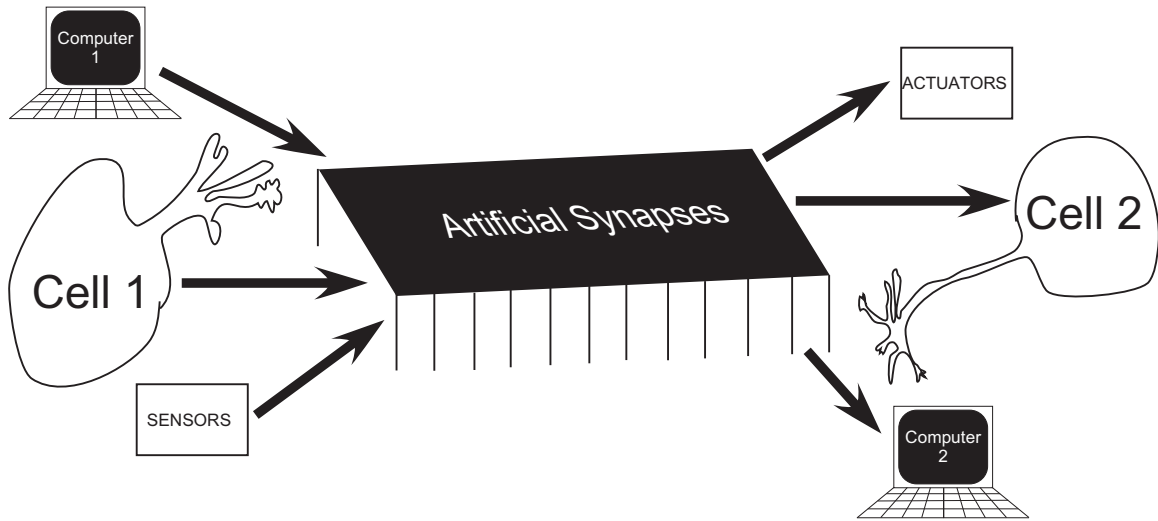


Figure 3: A system that can connect living neurons to each other using bio-inspired, bio-compatible circuits has been created. This system can be seen as a bio-inspired medium in which artificially or biologically created signals can be processed and then sent out to artificial or biological devices or cells. The system is represented as the IC, which contains three portions: an amplifier, a field programmable neural array (FPNA), and output synapses. This work looks at the zeroth and first-order cases of connecting an artificial neuron to a living neuron and connecting two living neurons, with no extra processing done by the FPNA.

in this work.

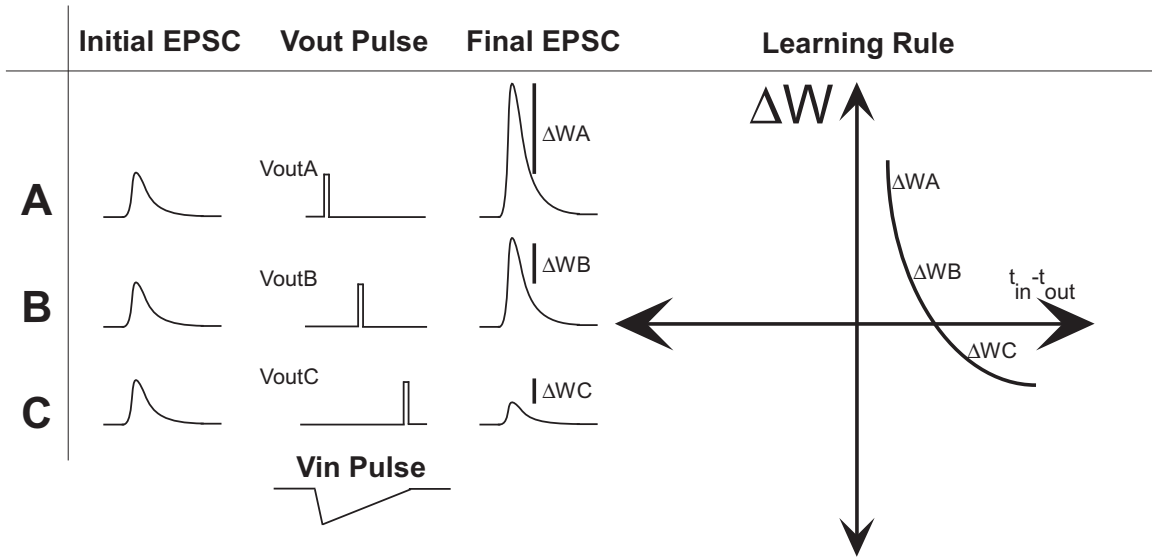


Figure 4: Visual explanation of a learning rule.

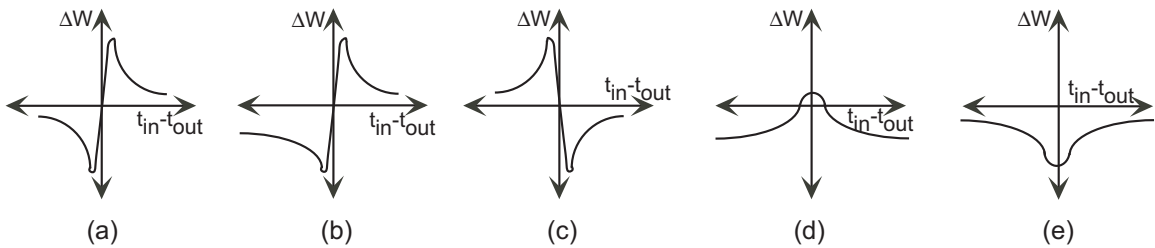


Figure 5: Graphical representations of different learning rules. These rules can depend upon the time at which inputs and outputs occur, the type of neuron, and the position of the synapse on the neuron. (Adapted from Abbott and Nelson)

CHAPTER II

BIOLOGY PRIMER

Nature has provided a plethora of solutions to common engineering problems. One of these solutions, the neuron, is the foundation of phenomena such as controlled movement, intra-body communication, and learning. Nature implements phenomena like this through communication between neurons. The fundamental unit of neuronal communication is the action potential, a biological digital pulse. The action potential plays a key role in learning and memory. Memory is stored information. Learning is the act of storing information. In biology, learning and memory stem, in part, from the strength of the links between neurons. As input action potentials create action potentials at the outputs, these links (called synapses) become stronger. This process, called long-term potentiation (LTP), is a foundation of learning [17, 18]. Furthermore, a synapse can be weakened if this ideal input - output relationship is not maintained [18]. This is known as long-term depression (LTD). These modulatory mechanisms have been used to describe a method of learning known as spike-timing dependent plasticity (STDP). We have created action-potential networks in analog VLSI that attempt to mimic the effects of STDP.

2.1 Biological Model

In a biological synapse, one cell communicates with another cell through an electrochemical process. Rather than controlling the flow of electrons (as in electronic devices), these biological synapses control the flow of ions. Figure 1 is a ligand-gated chemical synapse.

The biological synapse links neurons to one another, as seen in Figure 1. Cellular

signaling takes place when the first neuron fires, causing a chemical (neurotransmitter) to be sent from the first cell (pre-synaptic) to the second (post-synaptic). This chemical diffuses through a gap between the cells (chemical synapse) and binds to receptor sites on the post-synaptic cell. This binding causes the ion channels to open up much like how a key (the neurotransmitter) can open a locked door (the ion channel). Ionic currents through the post-synaptic cell flow because the neurotransmitters bound on these receptor sites and allowed the ion channels to open. These currents cause an electrical response, and the post-synaptic cell has successfully received the pre-synaptic cell's signal. Although the pre-synaptic cell's electrical signal (action potential) is basically binary in nature, it is essential that the post-synaptic potential (PSP), the second cell's electrical signal, remain analog in nature [17], [19], [20]. This post-synaptic potential can be either excitatory (EPSP) or inhibitory (IPSP) and can be modified by various pharmacological and morphological factors [21], [17], [19], [20], [22],[23]. For instance, the number and type of open ion channels allow more or less of an ion of a polarity to flow in or out of the cell [24]. The PSP is often modelled with Rall's alpha function:

$$V_m = V_o \frac{t}{\alpha} e^{-\alpha t} = V_{max} \alpha t e^{(1-\alpha t)},$$

where V_{max} is the peak of the EPSP (or IPSP minima) [19].

The type of potential produced depends on the type of ions that flow through the cell when the neurotransmitter has binded to the cell. Ion channels, the site of ion flow through the cell and neurotransmitter reception, are akin to doors with locks. The door can open only when the proper key is used. GABA_A and non-NMDA ligand-gated synapses employ ion channels that are keyed by the type of neurotransmitter present, while NMDA ligand-gated synapses are keyed by the type

of neurotransmitter and the voltage across the cellular membrane. This ligand-gated NMDA ion channel is much like a more secure door requiring both a physical key and the opening of an electronic lock with keycard. Figure 6 illustrates a few of these ion channels. Modeling of these ion channels has implications beyond chemical synapses. Ion channels are found other places such as plant, pancreatic beta, and muscle cells. Although this work specifically mimics synapses, it can be modified for these other ion channels to expand the application base from just neuroscience to range from botany, diabetes medications, heart disease, and beyond [25],[26].

The chemical synapses are one broad type of cellular connection. Another, similar type of connection is the neuromuscular junction (NMJ) [23], [17]. In an NMJ, the link is between a specific type of neuron (motoneuron) and a muscle fiber. The output of an NMJ is essentially a broadcast of a neurotransmitter from the motoneuron to a large surface area on a muscle fiber. Figure 2(a) shows how the basic electronic synapse can also be used to emulate the input/output response of the NMJ. Unlike the chemical synapses described above, NMJs produce end-plate potentials or smaller versions, mini-end-plate potentials. Note that the end-plate potential is essentially the same as an EPSP. The key difference is that the muscular fiber does not need to process the resulting membrane voltage in the complex way neuron does. Although the areas of learning in the brain and facilitation in muscles have not been fully explored, the current research implies that there are more complex connections in the central nervous system than in the peripheral nervous system.

Another more simple of cellular communication mechanism is the electrical synapse. This synapse uses a physical exchange of ions to create a link between two neurons. Note that this is not the same as an electronic synapse. Electronics are artificial devices which control the flow of electrons (or the absence of electrons known as holes). The flow of ions in an electrical synapse is controlled by connexons, as seen in Figure 2(b). Electrical synapses are the fastest type of synapses since the intracellular fluid,

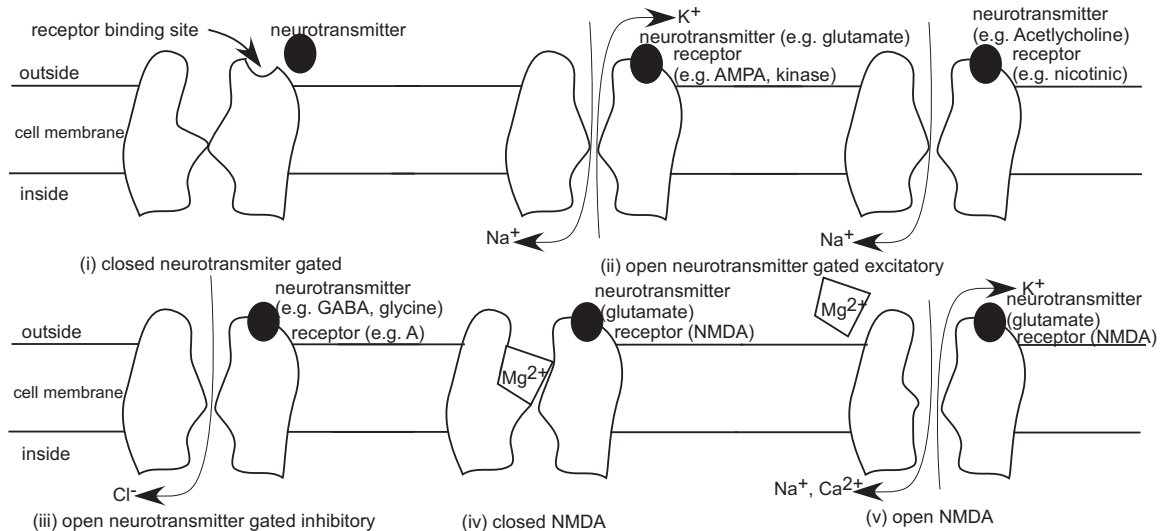


Figure 6: Examples of ion channels found in the nervous system.

and the ions within, can flow from one cell to another. Furthermore, although it is possible to have a unidirectional electronic synapse, the free flow of intracellular fluid allows for bidirectional communications. The type of electrical synapses shown here with connexons are found in vertebrates. These connexons and ion channels in chemical synapses and NMJs can be modeled with FETs as well. This work emulates the effect of chemical synapses. Electrical synapses are formed when these circuits are connected to living systems since there is no transmission of chemicals from the circuit.

2.2 Ion Channels

The dynamics of the ion channels found in neurons have been modeled as equations having the same form as the Fermi function [17], [19]. The equations that describe the way charge is distributed through FETs are also modeled with Fermi functions [27]. This is due to the fact that these phenomena, both biological and electronic, are governed by the same physics equations. The movement of both the ions and carriers is determined from Fick's Laws [19], [27]. Therefore, rather than using unrealizable potentiometers as suggested by Hodgkin and Huxley [17],[19], a model using FETs to

represent a population of ion channels has been created [28]. This FET-based model provides an accurate representation of ion channels that requires a much smaller die area compared to previous models [28].

2.3 Spike-Timing Dependent Plasticity

Spike-timing dependent plasticity (STDP) has been shown as a biological method used to modify synaptic weight [3, 29, 4, 5, 6, 7, 8, 9, 10, 11]. Although the full underlying cause of STDP is still being debated in the neuroscience community, a few details are understood [30, 31, 32, 33, 34, 11]

1. synaptic weights can change with the type of neuron, position of synaptic input, and, of course, the timing between inputs and outputs,
2. the levels of Ca^{2+} are important to the synaptic weight.

There are many contentious, yet crucial, points that are not understood. Much like what is found in physics, there is no "grand unified theory" to explain how learning works. We just see the effects as changes in post synaptic currents or voltages. Since these points are not fully understood, any future model that depends upon them must be proved or disproved with new biological evidence. We still need to understand

1. the exact role of NMDA receptors,
2. the effect of nitric oxide,
3. the mechanism behind the dendritic effect upon learning rules.

As seen in Figure 5, the change in synaptic weight will change depending upon the timing between the input and output. Since it is not precisely known, the timing of possible mechanisms behind the weight change (such as $[Ca]$ or $[NO]$) is not presented. When the input occurs before the input, Hebbian learning rules exhibit an

increase in the synaptic weight (Figure 5a). The opposite case, although not mentioned specifically in Hebb's law, stipulates that outputs that occur before inputs should lead to a decrease in synaptic weight. This decrease is also illustrated in Figure 5a. Even more peculiar is the fact that neurons from the same region (such as hippocampus) can exhibit changes to the learning rule as seen in 5b and d. Other strange cases, such as the complete opposite of Hebbian learning is found in electric fish (Figure 5c) and a strictly decreasing case (5e) have been found. This work attempts to create a method for explaining various bio-inspired learning rules that are found in biologically plausible CMOS circuits.

CHAPTER III

FLOATING GATE PRIMER

To create a biologically plausible electronic circuit, we have used floating-gate transistors. These circuit elements can be cost-effectively manufactured in most CMOS processes, are well suited to storing analog information, and can operate using low power levels on the order of pW to μW . These factors lead floating-gates as an excellent fundamental component in circuits that can emulate the electronic aspects of synapses.

3.1 Basics

The core of the synapse is the floating-gate transistor (Figure 7). Floating gate transistors are basically EEPROMs. This compact, non-volatile memory cell operates by storing electrons in an isolated portion of polysilicon called the floating gate. The charge is trapped and must use one of three methods to move the electrons: hot-electron injection, Fowler-Nordheim tunneling, or ultraviolet light. Ultraviolet light requires extra equipment and will not be covered in this document. Injection and tunneling are used to approximate the biological functions of long-term potentiation (LTP) and long-term depression (LTD).

There are two main methods of deciding how to adjust charge: programming and adaptation. Programming entails moving charge to a specific value that is determined by the user, while adaptation involves the system itself deciding on where the charge should be. These processes are akin to supervised and unsupervised learning. There are computationally interesting types of learning which include both supervised and unsupervised methods. Biological learning on the cellular level is unsupervised.

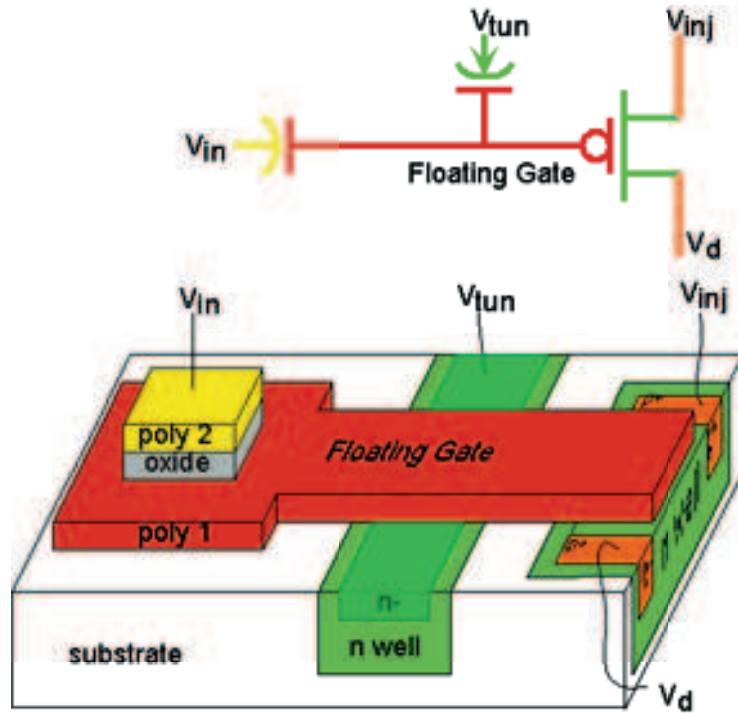


Figure 7: Pedagogical illustration of a floating gate.

3.2 Injection

Injection is used to add electrons to the floating gate. Two criteria must be met for injection to occur:

1. large voltage across the FET,
2. current through the FET.

In the case of a pFET floating gate, hot-electron injection takes place when a hole has so much energy as it speeds through the channel that when it collides within the lattice, it creates an electron-hole pair. Having sufficient energy to move into the conduction band and nowhere else to go, the stray electron moves into the gate oxide. Present CMOS technologies attempt to prevent injection by including spacers that prevent injection in nFETs. These spacers act as a barrier that prevent electrons from traveling into the gate. However, there are no spacers for pFETs. Thus, pFETs

are used to inject charge onto floating gates. The process voltage is determined by the minimum voltage needed to cause injection.

Injection has a built-in AND function since it requires both the flow of carriers through the channel and a large enough electric field to accelerate the carriers. This natural AND is used to isolate single floating gates within a matrix when injecting. To take advantage of the AND, the input gates are connected along columns and the sources and drains are connected along rows. The columns determine the current through the FET and rows establish the voltage across the FET. The following steps are used to inject:

1. Raise all voltages (drain, gate, source, and well) to a voltage suitable to initiate injection. This prepares the matrix for injection without actually injecting.
2. Select the desired column by dropping the gate voltage to a suitable level for current to flow. This prepares the column without injecting since the voltage across the pFET is still 0 V.
3. Drop the drain voltage of the desired row so that the voltage across the selected FET is large enough to accelerate the carriers.

As seen in Figure 8, V_{sd} easily modifies the rate at which the current changes. Figure 8 also shows that the current does not increase without bound. The rate of injection is illustrated in Figure 9. Every injection iteration is most efficient at the boundary between subthreshold and threshold. This is shown as the peak in each of the curves in Figure 9. Furthermore, the overall rates are quite different. The three phases allow for very different injection dynamics depending upon the present value of the current through the floating gate.

The first phase of injection is the increasing portion seen in Figure 9. The currents are low enough to still be in threshold. Here the rate of injection is increasing in the fastest manner. So, a two changes in current from two injection iterations within this

regime will cause the second change to be larger than the first change. This allows for exponential growth of a current.

The second phase of injection is the peak seen in Figure 9. The rate of change is at a maximum, which still allows for exponential changes in current. However, each injection iteration will be about the same within this mode of operation. The peak occurs right before threshold.

The downward slope in Figure 9 illustrates the third phase of injection. Here, two consecutive injection iterations result in consecutively decreasing current changes. The current is still increasing, but at a very slow rate. This slow rate can easily be seen in the plateaued currents in Figure 8.

These three phases of injection efficiency allow for greatly different rates of changing the current through the floating gate. This is used as an advantage in crafting a learning rule since injection allows for multiple ways of increasing a synaptic weight without adding more circuitry.

Note that the source can remain at V_{DD} and the drain can drop to a negative voltage to provide a suitable V_{ds} . A negative charge pump can be used to achieve negative voltages on chip. A negative Dickson charge pump is shown in Figure 10a. The diodes (diode-connected pFETs here) are in the opposite direction compared to the original Dickson charge pump [35], [36]. Changing the direction of the pFETs allows this circuit to act as a negative charge pump (dropping the output voltage), rather than a positive charge pump (raising the output voltage).

This charge pump works like a bucket brigade removing water from a boat. The task of this circuit is to remove charge from the output node (inside of the boat) to the input node (outside of the boat). The clock and its complement are like the members of the bucket brigade passing buckets from one to the other. Instead of buckets of water, the circuit has capacitors that hold charge. Each stage can provide a drop of V_{DD} ; however, the diode-connected pFETs cause a threshold drop (U_T). Ideally,

there is no additional drop because of the capacitors (exponentially leaky buckets). However, there will be a drop if the capacitor size or clock frequency are inadequate. Figure 10b illustrates the result of simulating this circuit.

The charge pump has been fabricated and has been found to work. Charge pumps with floating-gates were fabricated as well. The ability to modify the effective V_T is a useful tool that can be used for more precise output voltages [37]. However, as we use CMOS processes with smaller feature sizes, we do not require as large of a voltage across the floating-gate FET. The creation of this negative can be used to program floating-gates, but it was found to be unnecessary.

3.3 Tunneling

Tunneling removes charge from the floating gate. Like injection, a large field must be produced to yield successful results. However, the energy needed for tunneling is not used for accelerating electrons. Charge is trapped in the oxide by an electron barrier. Charge can leave the oxide if the electrons overcome the barrier. However, because of quantum mechanics, there is a minuscule probability that the electrons can go through the barrier. This probability can be increased by decreasing the thickness of the barrier. With a narrow enough barrier, the almost impossible task of moving an electron through the oxide becomes a certainty. The barrier is thinned by increasing the voltage drop between the tunneling node and the gate as seen in Figure 11.

In a $0.6\mu m$ process, the tunneling effect is easily seen at voltages above 12 V. This voltage is currently provided off chip. In smaller processes, the necessary voltage is much less and can be switched on and off using the high voltage FETs available in these processes. Therefore, future designs, should incorporate amplifiers and charge pumps to provide the proper tunneling voltage.

A further limitation seen in the $0.6\mu m$ process is the delay before tunneling. This delay is clearly marked in Figure 12. Rather beginning the downward progression of

current after the first iteration of a tunneling pulse, the current remains the same. The current begins the expected exponential drop after a few iterations. The length of this delay (number of iterations) is inversely dependent upon the current. This effect is not widely seen across all chips in a $0.6\mu m$ process and is currently under investigation. Furthermore, this effect is not seen in smaller processes. Figure 13 shows that tunneling begins at the first iteration in a $0.35\mu m$ process. Note that the rate of tunneling is faster (the current decreases more for the same number of iterations) for the same value of V_{tun} .

The stable behavior of tunneling in a $0.35\mu m$ process can be used to predict the change in current using mathematical methods. The equation describing tunneling is not as effective in a $0.6\mu m$ process since the delay causes an unexpected effect. The effect of the time that the tunneling pulse is on (t_{tun}) is illustrated in Figure 14. Intuitively, the current drops more for longer pulse widths. Further analysis (Figure 15) clearly shows that this rate of change is exponential.

3.4 Array Programming

A method to quickly and precisely program large numbers of floating gates has been developed by Serrano and others [38]. Tunneling is used as a global erase. Injection is used to adjust the output current of a floating-gate pFET to a specified value. Figure 16 illustrates the process for injection. The injection process is controlled by an embedded system. This system uses an FPGA, a custom PCB, and on-chip support circuitry to read currents and provide the proper voltages for injection [38]. The system can program a floating gate pFET current to within 0.5 percent error in 350 ms [38]. Targets can be hit reasonably well within one injection pulse in as little as $10\ \mu s$.

First, tests are performed to characterize an example pFET. Next, the parameters

extracted from the characterization are used to estimate the proper V_{ds} for programming the entire array. To implement the fastest method of programming, a single, short pulse with varying V_{ds} values for every floating gate. The more accurate method uses multiple, short pulses that are modified to better approach the given target.

The method of array programming illustrates the fact that no matter the size of the array, floating-gate pFETs can be isolated and individually modified. Typically, one can intentionally set the output current through each floating-gate pFET to a specific value. However, the output current can also be changed without intentionally setting the value to a specific target. This work modifies the current without using a specific, a priori target.

3.5 Indirect Programming

The synapses have been revolutionized due to the ability to keep the operating pFET in the loop while changing the synaptic weight. Previously, switches were used to isolate the STLS from the rest of the operating circuit while adjusting the synaptic weight. As illustrated in Figure 17, a second pFET has been added to the STLS. The second pFET is used for programming, while the first remains part of the regular operating circuit. The same methods of injecting can be used with the second pFET and the resulting change in charge is mirrored in the first pFET. This method is known as indirect programming [39]. As a result of indirect programming, the synaptic weight can be changed during regular operation instead of entering a program state. Moreover, many other types of circuits become feasible because of indirect programming. A few examples include the floating-gate tunable charge pumps mentioned above in the Injection section and large networks of floating gates such as Field Programmable Analog Arrays.

Initial data shows that indirectly-programmed FETs work, as seen in Figure 18. Figure 18 shows that the currents match well for when the floating-gate is programmed

to low currents, however do not exhibit a linear relationship for currents that are programmed above the nA range. When designing the prototype devices, we did not take the effects of W/L and input capacitor sizing into account. As seen in Figure19, the size of the input capacitance greatly affects the behavior of the indirectly programmable floating-gates. Designs made since have rectified this oversight and indirect programming works more effectively. The collaborative efforts of David Graham, Ethan Farquhar, Brian Degnan, and I resulted in the improvement of this new approach to the floating-gate [40].

The true benefit of indirect programming can be illustrated when programming multiple floating-gate pFETs. This method has been improved so that floating-gate pFETs can be programmed while running [41]. This allows a system with many floating-gates to be modified as a function of the value of other output currents in the system. The ability to modify the output current of the floating-gate while running is essential to this work.

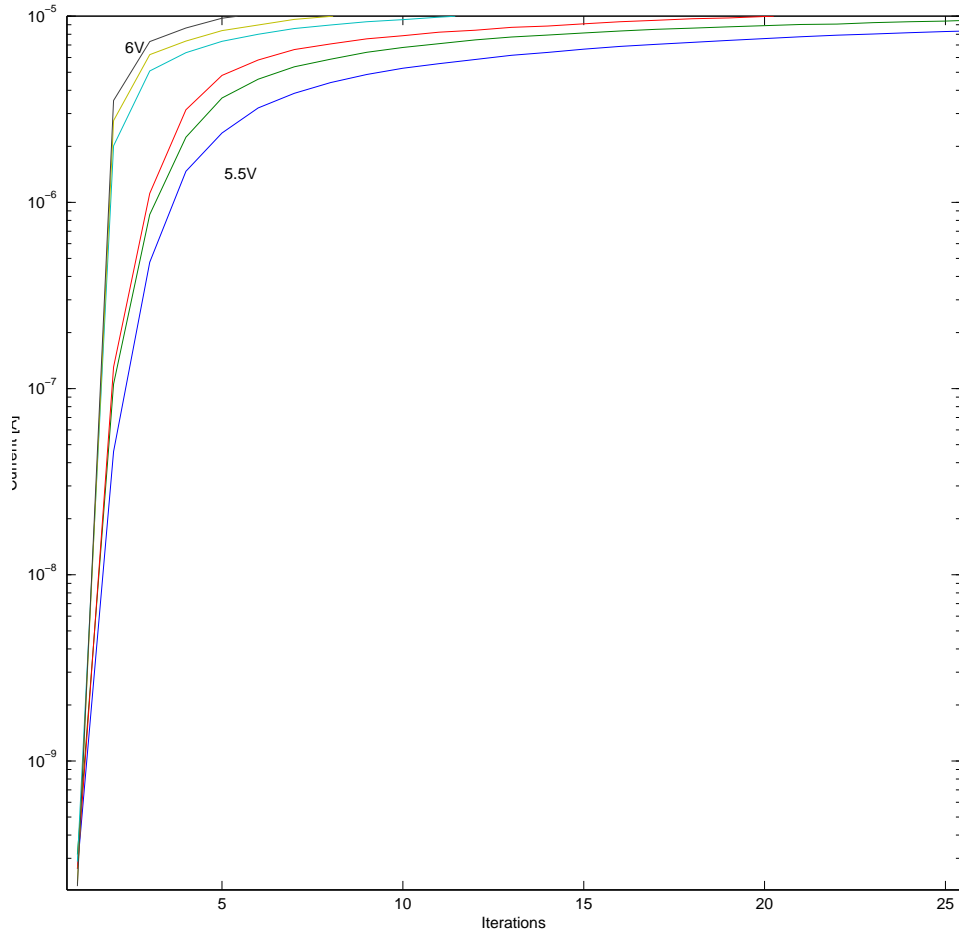


Figure 8: Injection in a $0.6\mu m$ process. Note that the rate of change in the current plateaus.

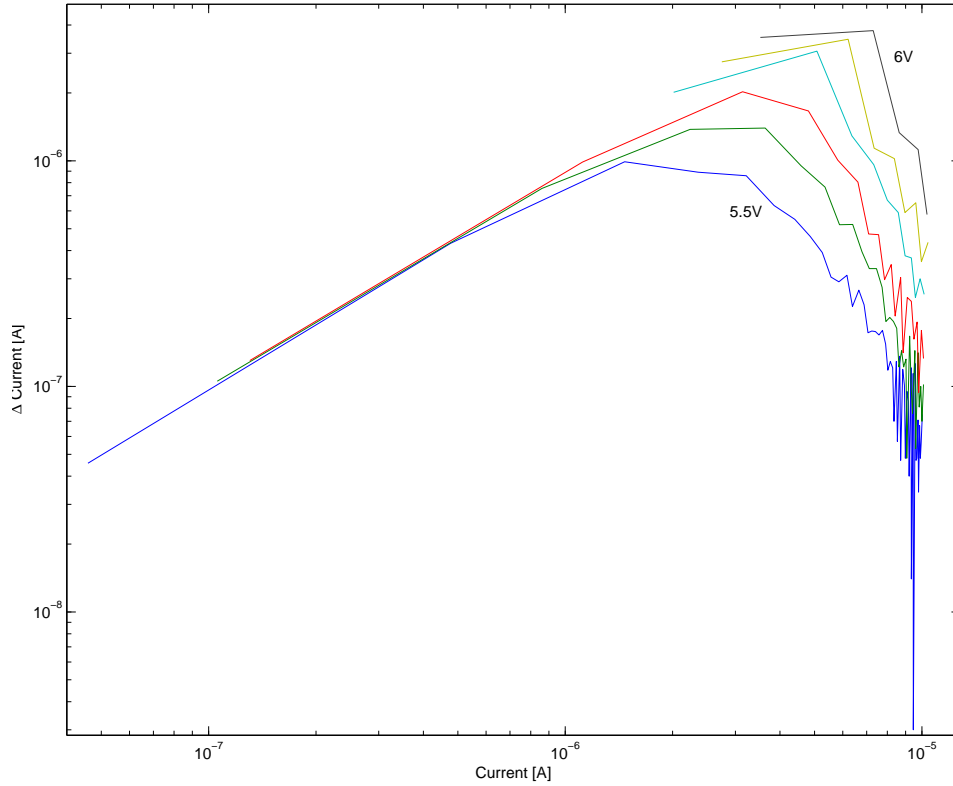


Figure 9: The rate of injection in a $0.6\mu m$ process. The injection rate changes depending upon the value of V_{sd} and the present current value. There are three main phases, increasing, maximum, and decreasing injection rate. Note that the noise in the decreasing phase is due to fluctuations the least significant digit in the picoammeter.

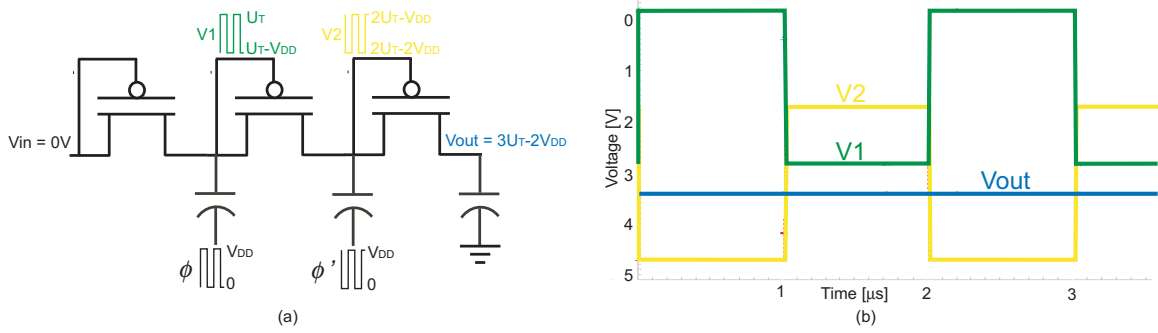


Figure 10: (a) Schematic of negative Dickson charge pump. The pFETs act as diodes. The clock signals (ϕ and ϕ') are complements. The waveforms illustrated are for the ideal case in which there is no droop due to the capacitor discharging. (b) Simulated data from the negative charge pump. V_{in} and the wells of the pFETs are at ground. The capacitors are 1 pF and the clock is 500 kHz. Note that at this capacitor size and clock frequency, there is no droop.

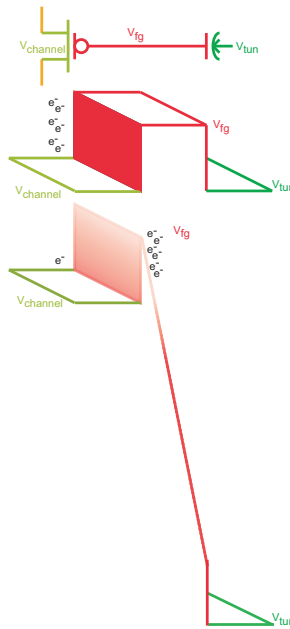


Figure 11: Illustration of band diagram of tunneling through a MOS capacitor. The shading at the gate indicates the probability of an electron to move through the barrier.

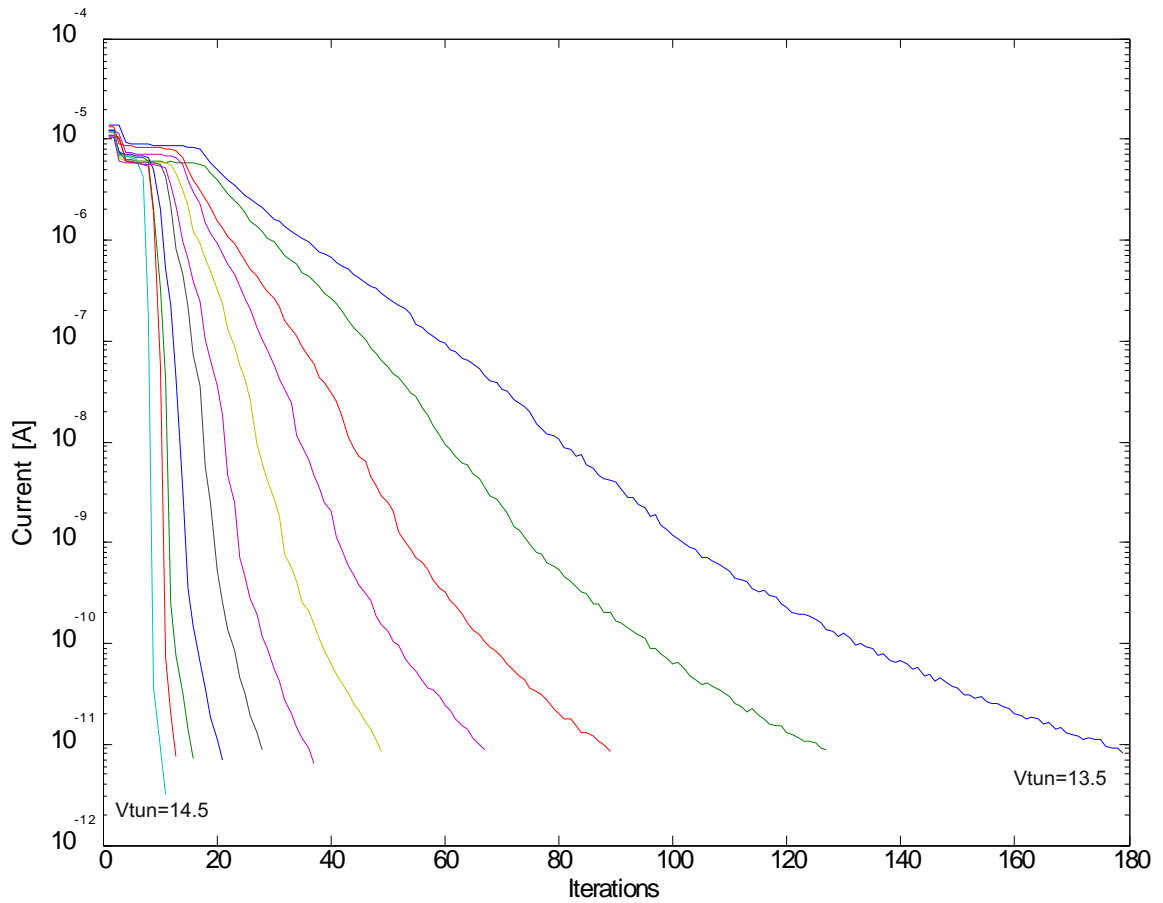


Figure 12: Tunneling in a $0.6\mu\text{m}$ process. Note that there is a delay in the onset of tunneling. This delay is inversely proportional to V_{tun} . The delay has not been modeled in the literature. Although this effect is not fully understood, it may be due to the behavior of the MOS capacitor at the tunneling junction. Since the capacitance value of MOS capacitors is dependent upon the voltage, the device may be moving through different regions of operation depending on V_{tun} . The tunneling voltages range from 13.5V to 14.5V in increments of 0.1V.

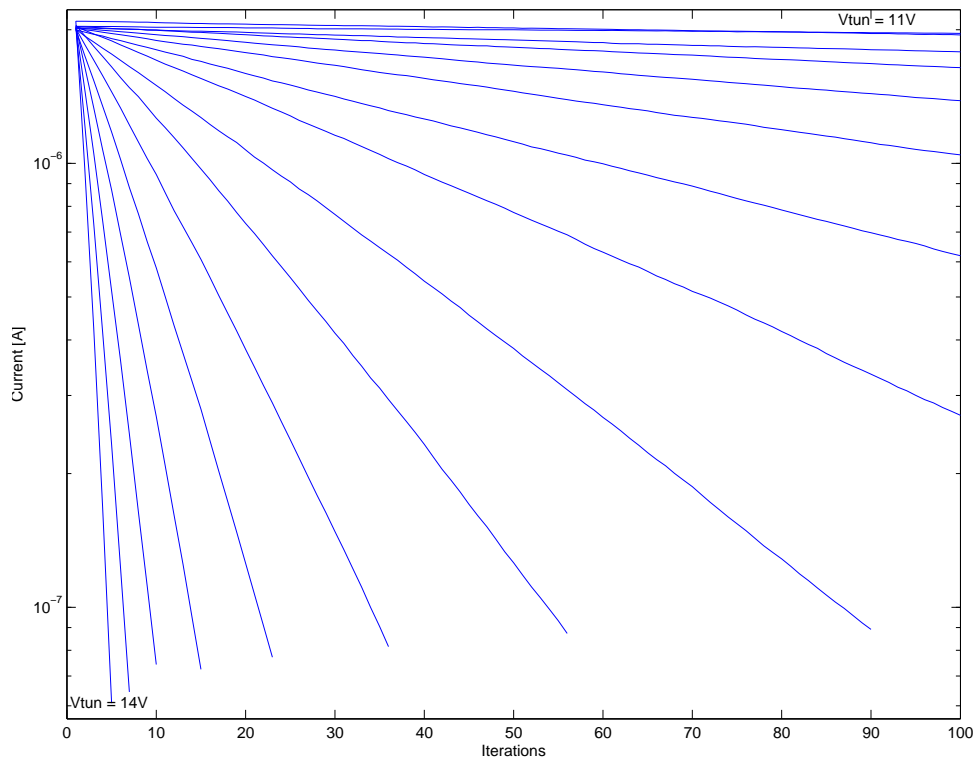


Figure 13: Tunneling in a $0.35 \mu\text{m}$ process. Note that there is no delay in the onset of tunneling as in the $0.6 \mu\text{m}$ process. The tunneling voltages range from 11 to 14V in 0.2V increments.

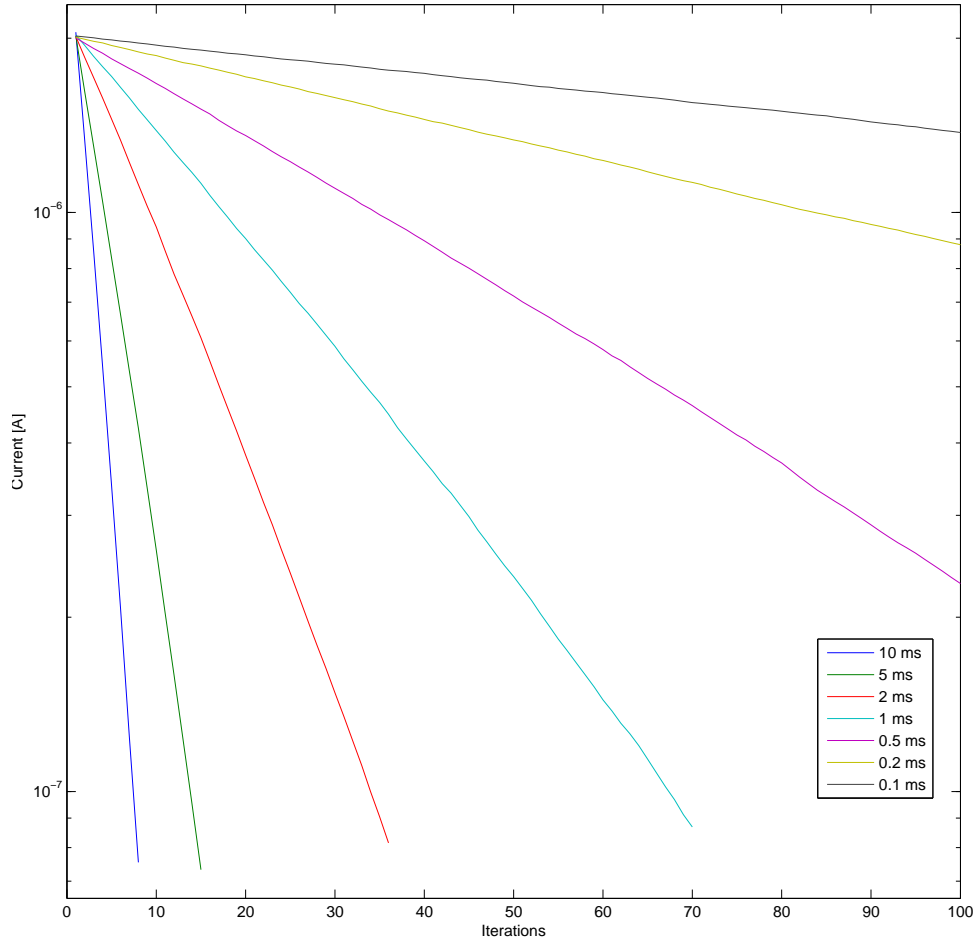


Figure 14: Change in tunneling rate for different pulse widths in a $0.35 \mu\text{m}$ process at 13V. An tunneling input of 13V and pulse width of t_{tun} was presented. The length of t_{tun} varied from 10ms to 0.1ms. The current began at $2\mu\text{A}$ and was allowed to drop to at least 90nA or for 100 iterations. The current through the floating-gate pFET was measured after each pulse.

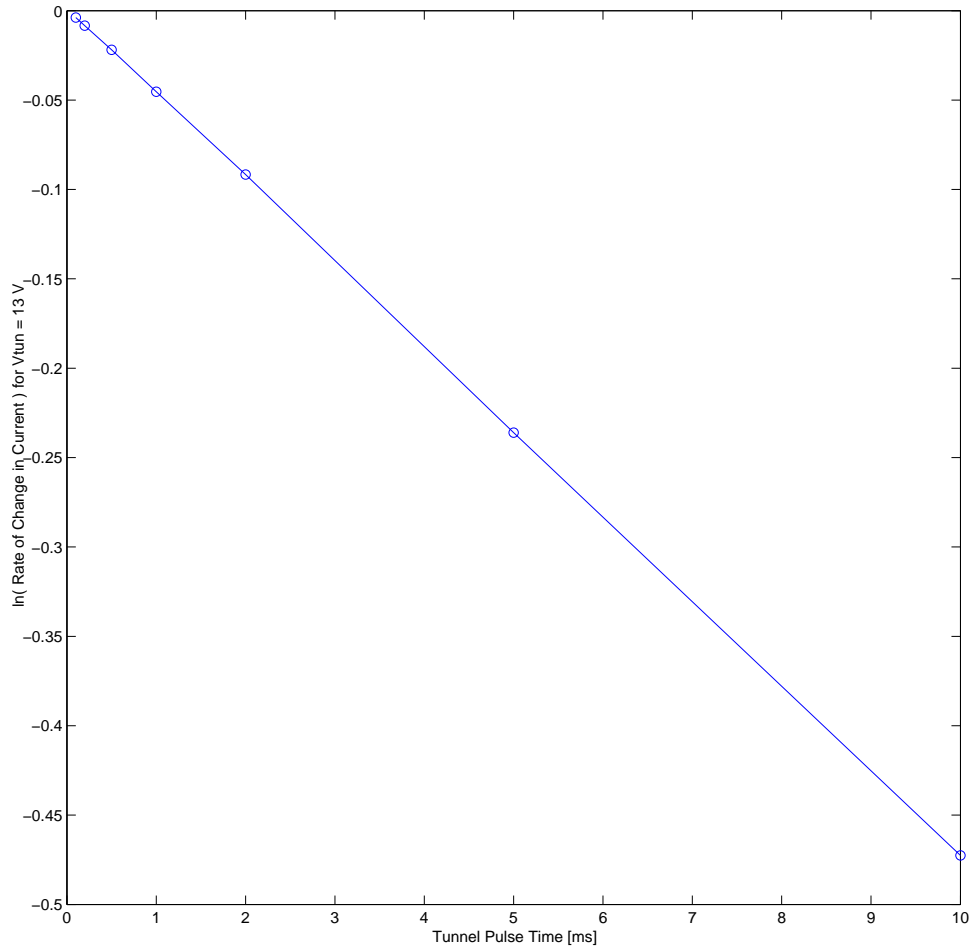


Figure 15: Natural log of the rate of change in tunneling in a $0.35 \mu\text{m}$ process for different pulse widths. The natural log is taken since the rate of change is proportional to e^{ttun} where $ttun$ is the time that tunneling is on. Therefore, a line is created when the natural log of the rate is plotted. The tunneling voltage V_{tun} is 13V in this experiment.

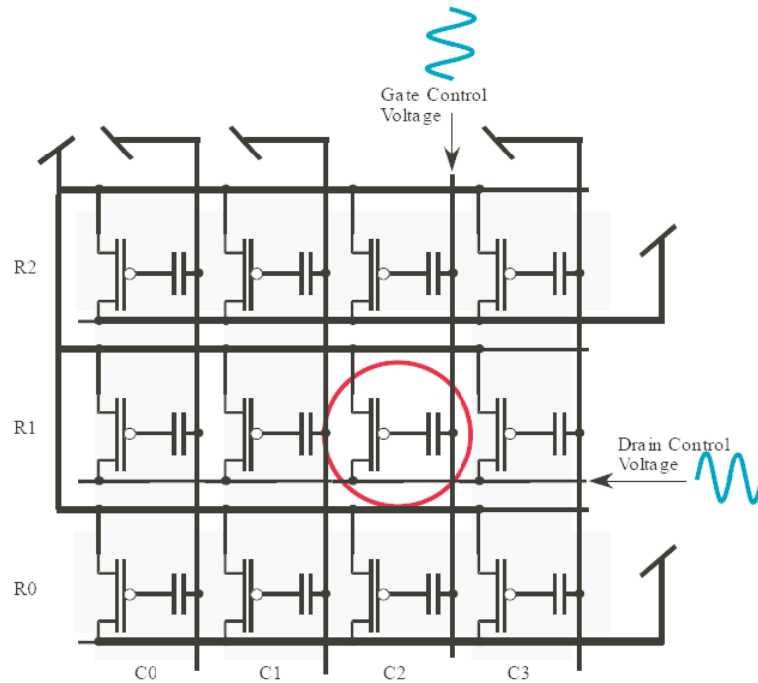


Figure 16: Programming of a matrix of floating gates. V_{DD} has been set to some value that would allow for a large enough V_{DS} for injection, such as 6 V. The floating gate at row 1, column 2 is isolated by putting all of the other drains (rows 0 and 2) to V_{DD} and the other gates (columns 0,1, and 3) to V_{DD} . The drain of row 1 is dropped to some desired level, such as ground, and the gate is sent to another level, such as 3 V. Figure originally from Kucic.

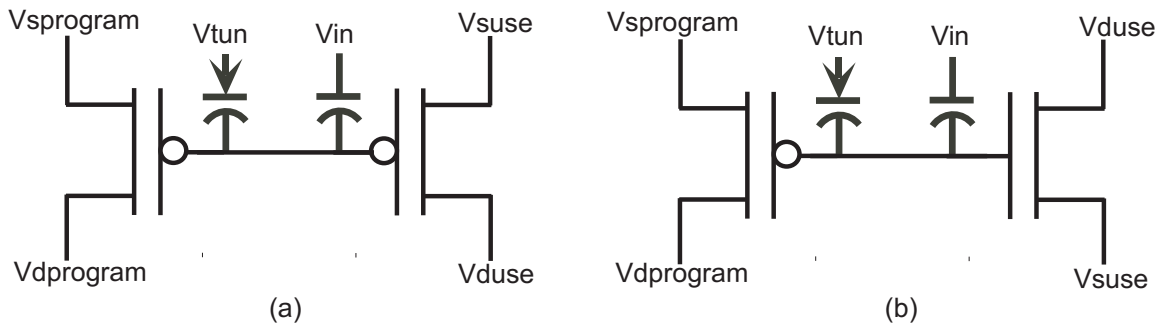


Figure 17: Figures of indirect programming structures. These structures allow for the adjustment of floating gate charge without the use of switches. (a) Indirect programming structure for a pFET. (b) Indirect programming structure for an nFET.

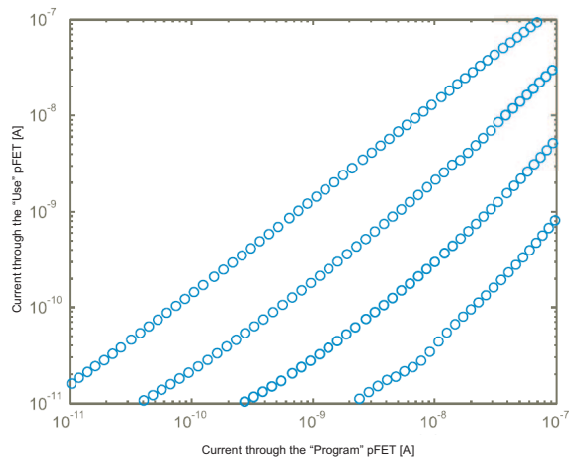


Figure 18: Data from indirectly programmed FETs as reported in the indirect programming paper.

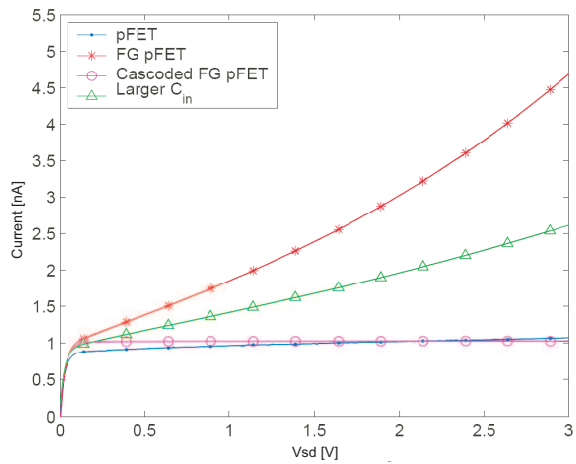


Figure 19: Data from indirectly programmed FETs illustrating the effect of capacitance values as reported in the indirect programming paper.

CHAPTER IV

ARTIFICIAL NEURAL CIRCUITS

I have developed a family of bio-physically inspired, bio-compatible, reconfigurable artificial synapses in CMOS. These synapses operate in subthreshold and draw power on the order of pW to μW . Therefore, an array of these elements has a power draw that is comparable to a biological system. An array of these artificial synapses were combined with artificial dendrites and ion channels designed by Ethan Farquhar. The collaborative effort with Farquhar led to the development of a reconfigurable system with neurally inspired core components: the Field Programmable Neural Array.

4.1 Models of Synapses

A single-transistor learning synapse (STLS) has been modified to create different synaptic types that are based on the types of ion channels present at the synapse. The STLS is a floating gate pFET [42]. This non-volatile method of charge storage allows for the easy change of the threshold voltage of the pFET through hot-electron injection and Fowler-Nordheim tunneling. The STLS is used as a memory device that will change its output current depending on the amount of charge on the floating gate, thus the charge on the floating gate is equivalent to the synaptic strength.

Furthermore, the forces that drive electric currents in a subthreshold MOSFET, diffusion and drift, are the same forces that drive ionic currents through ion channels in neurons [43], [44], [28]. Thus, subthreshold MOSFETs are ideal structures to model ion channel behavior. These low subthreshold currents allow us to have thousands of synapses on chip that collectively use little power. Although our present implementation has not been optimized for size, its area is less than $850 \mu m^2$ and would allow for more than 600 synapses on a MOSIS TinyChip (1.5 x 1.5 mm die) with room for

40 pads, testing circuitry, peripheral circuitry, and wiring.

To create accurate synaptic circuits with variable strength, we first investigate an EPSP that is produced by an action potential-like signal. By inspection, a typical EPSP is an exponential signal with the upgoing side's time constant being much faster time constant than the downgoing side, as illustrated in Figure 20. The log of that signal is basically a triangle wave with an upgoing slope that is much faster than the downgoing slope. Since the output current of a subthreshold MOSFET is exponentially dependent on the input gate voltage, an asymmetric triangular signal placed on the gate should look like an EPSP. Since, pFETs invert the input signal instead of a fast upgoing slope and a slow downgoing slope, we should invert this signal so that it is a fast downgoing slope followed by a slow upgoing slope. A simple method of producing such an asymmetric triangular signal is to have a subcircuit that can charge a capacitor faster than it discharges it. An inverter modified with pull-up and pull-down biases can easily accomplish this task by setting the pull-up bias such that it draws current more slowly than the pull-down bias. Furthermore, this signal can be created with a digital pulse, which is an action potential-like signal. Thus, we have accomplished our goal to have an action potential-like input with an EPSP like output that can be weighted according to synaptic strength. All of the other synapse types discussed here are built on this foundation.

4.1.1 Excitatory Synapse

There are many varieties of excitatory channels in the nervous system, as seen in Figure 6. Acetylcholine channels allow Ca^{2+} to rush into the cell in the presence of acetylcholine, while glutamate channels bring Na^+ in and drive K^+ out. This circuit topology can emulate many types of excitatory synapses by simply changing the relative values of the constant voltages (Figure 22). For instance, while E_{Ca} and ground are used to represent a collection of ACh channels, they could be replaced

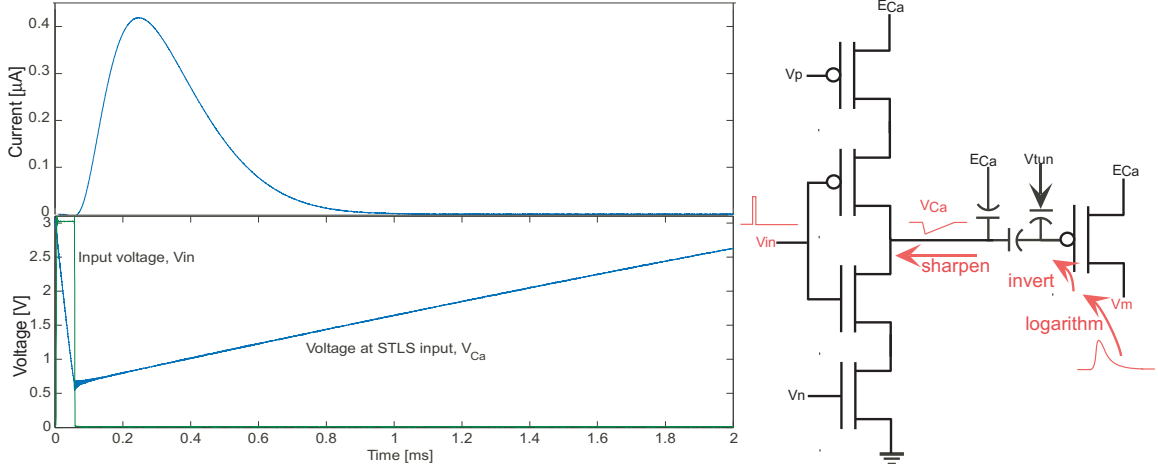


Figure 20: A typical PSP that would be found in biology modeled from Rall's alpha function on top with the log of the alpha function at the bottom. The right illustrates the steps taken in developing the synapse. Important functions, nodes, and waveforms are highlighted in red.

by E_{Na} and E_K , respectively, to emulate a family of glutamate channels. Of course, the digital input to the synapse should be scaled accordingly to maintain proper operation.

Figures 22c and 22d illustrate how we can easily create a variety of EPSPs by simply modifying our pull-down (V_n) and pull-up (V_p) biases. As expected, V_n changes the EPSP's rising time constant, while V_p changes the falling time constant. Therefore, by changing our location in the V_n, V_p parameter space, we can emulate many of the natural stimuli that modify EPSP shape and size [17], [19], [21]. Although the voltage levels given are accurate to three decimal places, the operating point within the V_n, V_p parameter space need not be so accurate. Our experiments were repeatable and stable using 12-bit DACs and hand-turned potentiometers.

One natural stimuli that greatly affects the PSP is the driving force that a particular ion has to propel it through the channel. The larger this driving force, the easier it is for the ion to move through the channel. This driving force is dependent on the membrane voltage (V_m) and the ion's concentration gradient between the inside and

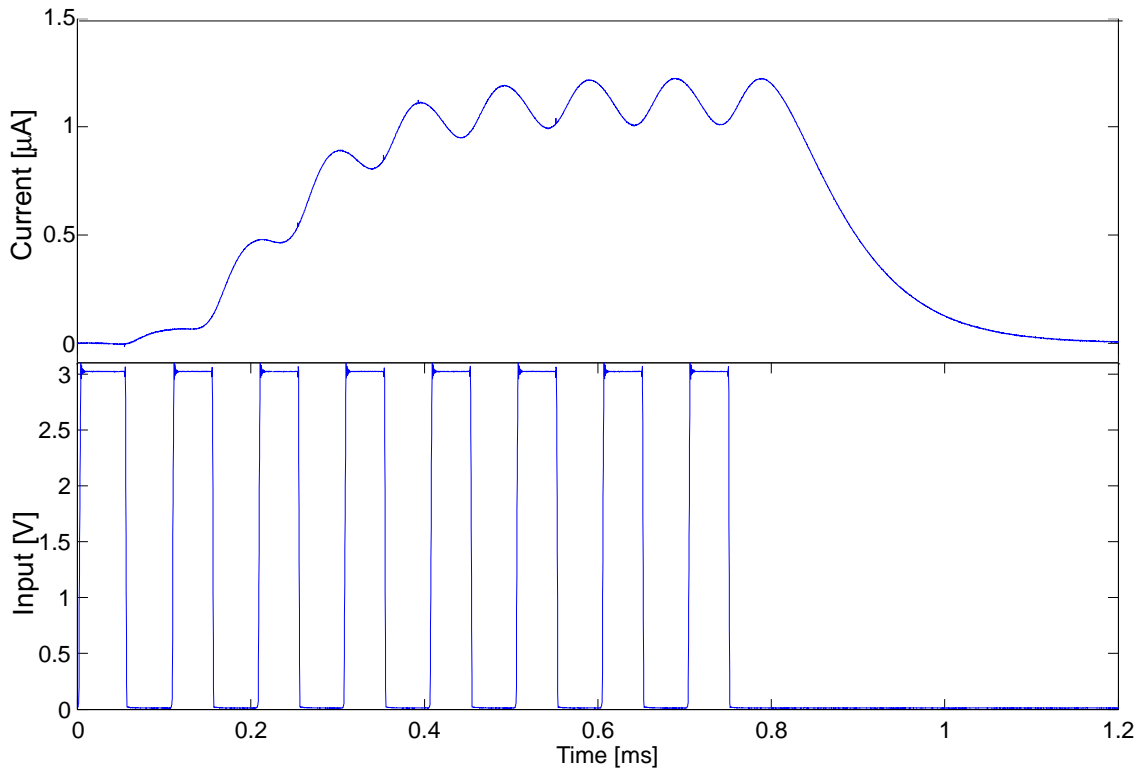


Figure 21: An increase in the frequency of the signal results in the aggregation of basic EPSPs, as seen in biological data.

outside of the cell. This gradient can be represented as a voltage such as E_{Ca} for calcium. Biological experiments to explore this biological phenomena are known as voltage clamp measurements. Figure 22b shows how we can modify our voltage levels to re-create biological voltage clamp data [17], [19], [43].

Another biological phenomenon is that of temporal summation. EPSPs combine to form one large aggregate signal. This effect is much like charging a capacitor with a series of small pulses. Each individual pulse is not enough to charge the capacitor, but if the period is small enough, the capacitor does not have enough time to completely discharge. Over time, the capacitor will be completely charged with some ripple.

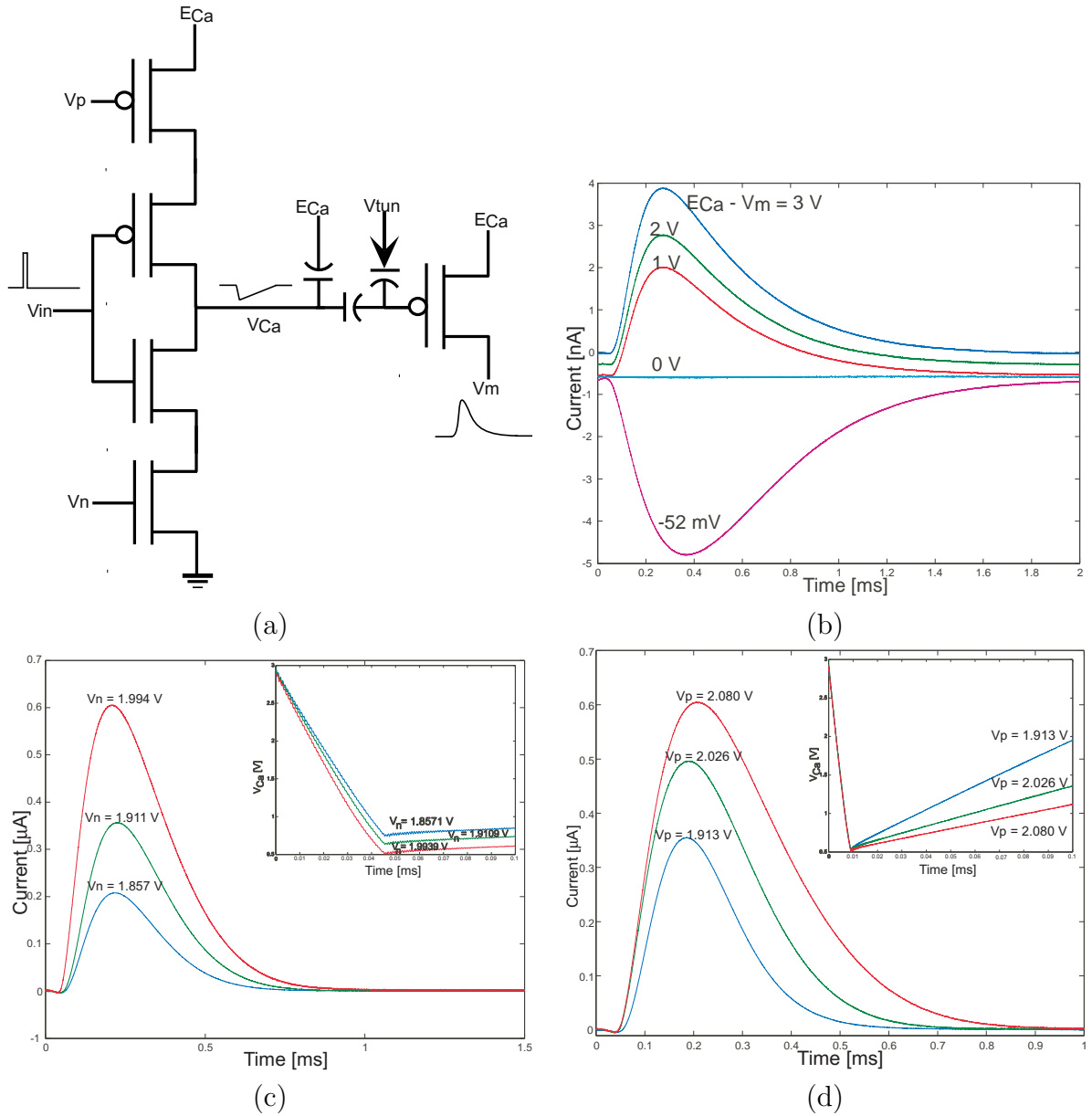


Figure 22: (a) The input of the basic excitatory circuit is an inverter with an output that has been slowed down by the capacitor to E_{Ca} . The net synaptic strength is proportional to the charge on the floating gate. Like its biological analogue, this synaptic strength can be modified. (b) Changing V_d in an excitatory synapse. In biology, decreasing the membrane voltage decreases the size of the basic EPSP. Decreasing the voltage across the synapse channel transistor decreases the size of the EPSP. In this case, $V_n=1.259$ and $V_p=2.238$. (c) The rate at which the initial, upgoing side of the basic EPSP increases by increasing V_n . Note that the inset, a plot of the voltage on the input of the STLS, illustrates only the the beginning slope changing with a change in voltage. Here, V_n varies and $V_p=2.080$.(d) The rate at which second, downgoing side of the basic EPSP increases by increasing V_p . Note that the inset, a plot of the voltage on the input of the STLS, illustrates only the the second slope changing with a change in voltage. Here, $V_n=1.994$ and V_p varies.

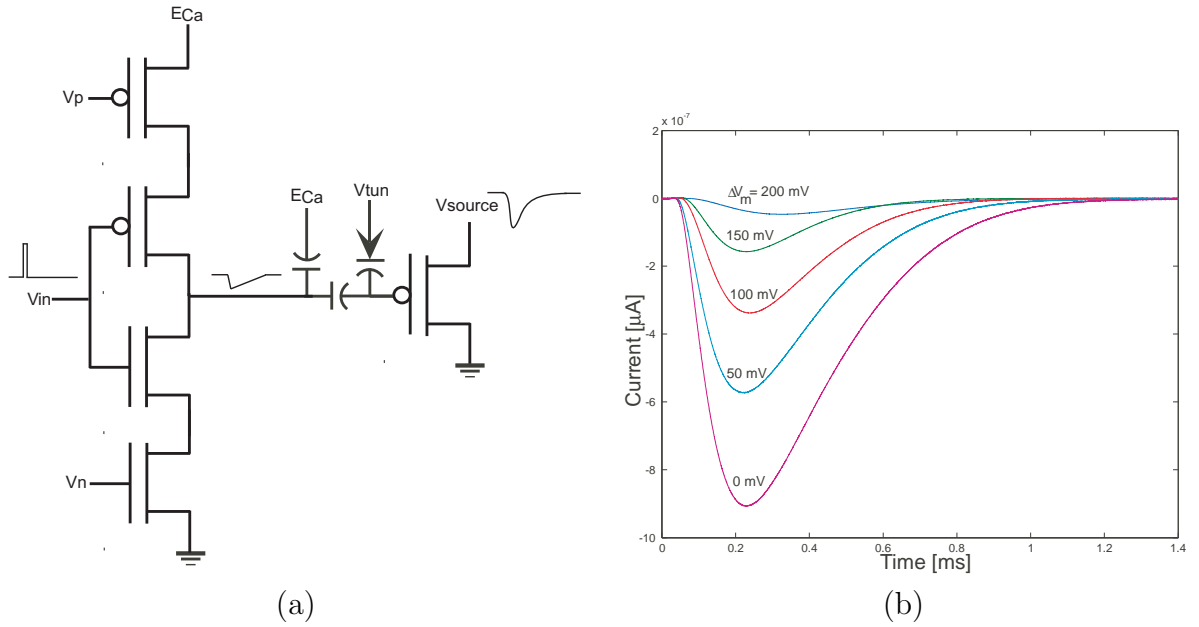


Figure 23: (a) The inhibitory synapse is similar to the basic excitatory, with the exception of the location of the output terminal. The floating gate source draws current, rather than having the drain provide current. Thus, we have an inhibiting effect on any subcircuit which this synapse is connected. (b) As expected, these $GABA_A$ inhibitory results are basically the mirror of the excitatory synapses, where this IPSP decreases for decreasing voltage across the STLS in this voltage clamp experiment. Here, $V_n=1.867$ and $V_p=2.109$.

Figure 21 shows that we can achieve results similar to what is found in biology.

4.1.2 Inhibitory Synapse

The inhibitory synapse produces a decrease in membrane voltage in response to an action potential in the pre-synaptic cell. Thus, the IPSP is effectively the mirror image of the EPSP. This IPSP is caused by a flow of Cl^- ions through $GABA_A$ gated channels. Glycine gated channels operate in a similar fashion and these silicon synapses can be considered to be similar to glycine biological synapses as well.

Figure 23a shows that we simply take a basic excitatory synapse and use a different node for the output. Since we want a net flow of negative current, we use the STLS source node for output rather than the STLS drain node. Current is now drawn rather than provided. The resulting effect is shown in Figure 23b. Figure 23b is a repetition of the voltage clamp experiment done above in the excitatory circuit section where we have effectively changed the driving force of the ions through the channel in an

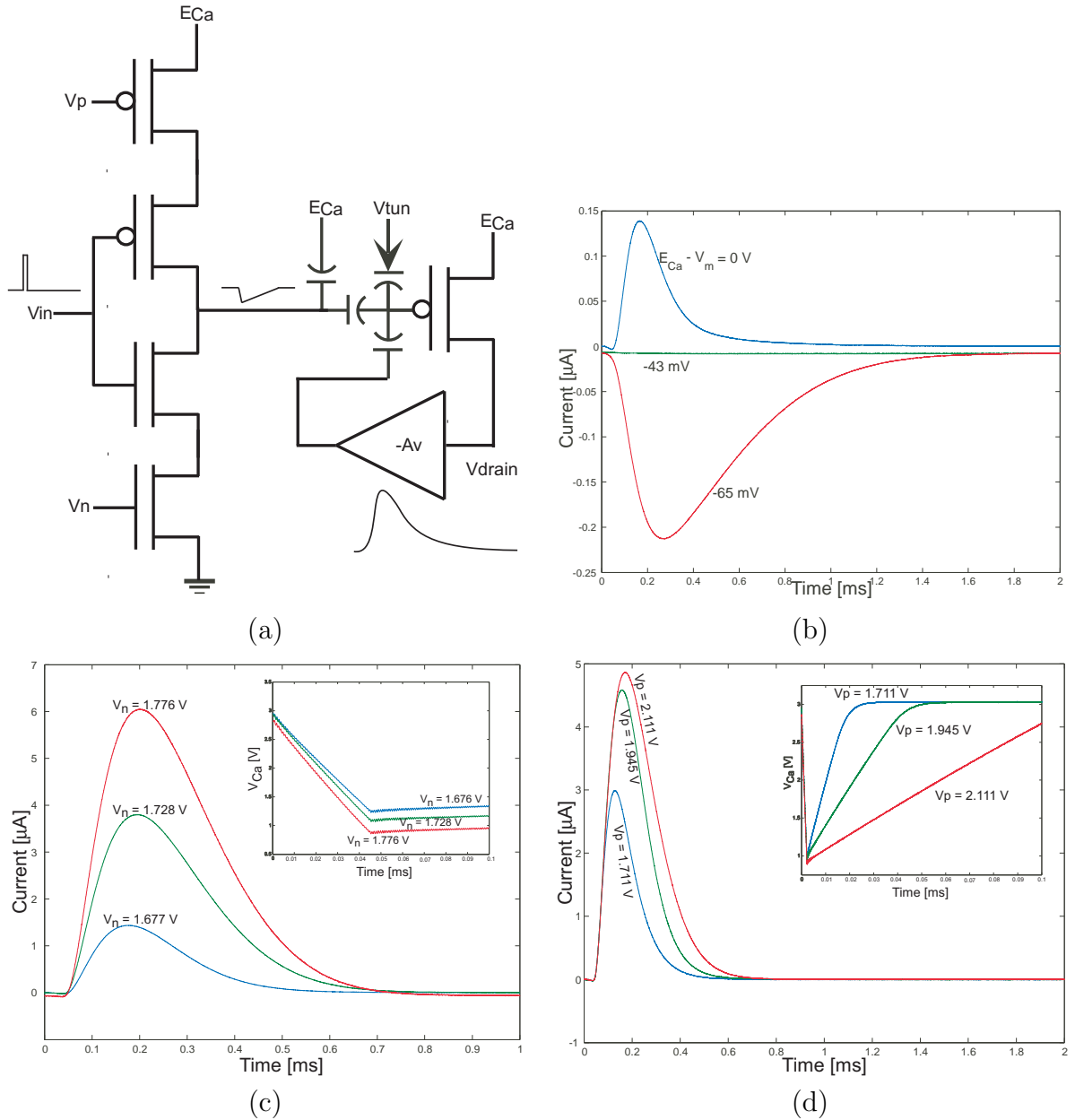


Figure 24: (a) Like the biological NMDA synapse, this circuit's response depends on the membrane voltage as well as the carrier flow. Here, we use a C^4 amplifier to mimic the effect of Mg^{2+} . (b) As expected, the NMDA voltage clamp experiment results are similar to the basic excitatory synapse where the EPSP decreases, then goes negative for decreasing voltage across the STLS. Here $V_n=1.259$ and $V_p=2.238$. (c) The rate at which initial, upgoing side of the NMDA EPSP increases by increasing V_n . Note the inset, a plot of the voltage on the input of the STLS, illustrates only the the beginning slope changing with a change in voltage. Here V_n varies and $V_p=2.080$. (d) The rate at which second, downgoing side of the NMDA EPSP increases by increasing V_p . Note the inset, a plot of the voltage on the input of the STLS, illustrates only the the second slope changing with a change in voltage. Here $V_n=1.777$ and V_p varies.

inhibitory synapse.

4.1.3 NMDA Excitatory Synapse

Biological NMDA synapses are doubly gated by the type of a neurotransmitter present and the voltage across the cell membrane. We have reproduced this gating mechanism by adding a feedback mechanism to the basic excitatory synapse. As illustrated in Figure 24a, the feedback is a C^4 amplifier with a gain of -1 to a second input to the STLS. The C^4 is a tunable bandpass filter in which each corner can be adjusted with a bias [45]. We pushed the corners as far out as possible, thus allowing the C^4 to act like an all-pass amplifier. As the output of the synapse increases, the output of the feedback decreases, thus decreasing the voltage at the input of the pFET floating gate. A decrease in the pFET input effectively increases the output, thus repeating this cycle. In the biological NMDA synapse, the presence of neurotransmitter initially opens other non-NMDA channels, which begins to depolarize the cell. Once the cell has depolarized enough, a Mg^{2+} ion is forced out of the NMDA channel and the channel fully opens. The initial depolarization of the cell is like the initial voltage change on the output of the circuit, while the feedback that augments this change on the circuit effectively like the Mg^{2+} ion being forced from the channel.

Figures 24c and 24d illustrate that we can easily create a variety of EPSPs by simply changing V_n and V_p . Similar to the basic excitatory synapse data shown in Figures 22c and 22d we can easily choose V_n and V_p to create an EPSP to suit a variety of applications. Figure 24b illustrates that we can drive the circuit from a positive to negative EPSP with a voltage clamp experiment. This is effectively the same as initially driving K^+ out and Na^+ in, not moving the ions much at all, and finally driving K^+ in and Na^+ out.

4.1.4 Analysis of the Synapses

The input stage of the synapse is a modified inverter. Current flows only through the pFETs when the input signal is low and only through the nFETs when the input is high. We can modulate the size of the current with the biases V_p and V_n . Since we have a capacitor at the output of the inverter, i_p does not need to equal i_n .

Current flows through all FETs when the input is between the rails. Since the input is a pulse, we can simplify our analysis by viewing the input as being either high or low and ignoring the insignificant transition time. This transition time is no longer than $10 \mu s$ for a digital pulse. For input pulses on the order of biological timescales, this transition time is less than 0.1% of the digital action potential. The inputs are not limited to strictly digital inputs. Here, the analysis is of the common, simplified input of a digital action potential.

$$v = v_c - E_{Ca}$$

$$i = i_n - i_p$$

$$i = C \frac{dv}{dt} \tag{1}$$

$$\frac{1}{C} i dt = dv \tag{2}$$

$$\int_{t_{start}}^{t_{end}} \frac{1}{C} i dt = \int_{t_{start}}^{t_{end}} dv \tag{3}$$

$$v = \frac{1}{C} \int_{t_{start}}^{t_{end}} i dt \tag{4}$$

Since i is a pulse, we can think of it as a constant at two different times. The current will be $i_p = I_p$ when the current is low and $i_n = I_n$ when the current is high. So, the solution to the integral is simply a line of slope $-I_p$ or I_n that changes with

time.

$$v = \frac{1}{C} \left[[-I_p t]_{t_{pstart}}^{t_{pend}} + [I_n t]_{t_{nstart}}^{t_{nend}} \right] \quad (5)$$

$$t_p = t_{pstart} - t_{pend}, t_n = t_{nstart} - t_{nend} \quad (6)$$

$$v_c = \frac{1}{C} [-I_p t_p + I_n t_n] + E_{Ca} \quad (7)$$

Therefore, we have a square, digital pulse that is transformed to an asymmetric triangle wave. The slopes of this triangle wave are controlled by the currents flowing through the FETs. These currents are controlled by the biases V_p and V_n . To determine the relationship between the biases and v_c , we have two methods. The FETs can operate in either subthreshold or above-threshold. Subthreshold operation uses lower currents and therefore a lower power operation. Ideally, we would operate in subthreshold so that we can minimize power consumption.

First, we will assume that the currents through the FETs of the modified inverter do not depend upon the drain voltages of the FETs. To ignore the drain voltages, we assume that the FETs are in saturation when current is flowing (the voltage across each FET is at least $4U_T$). We will check this assumption at the end of this analysis. Since the FETs are in saturation, the currents are:

$$I_n = I_{n0} e^{(\kappa V_n - gnd)/U_T}, I_p = I_{p0} e^{\kappa(V_{dd} - V_p)/U_T}$$

So, the triangular waveform is now:

$$v_c = V_{dd} - \frac{1}{C} \left[-t_p I_{p0} e^{\kappa(V_{dd} - V_p)/U_T} + t_n I_{n0} e^{\kappa V_n/U_T} \right] \quad (8)$$

Therefore, we have a voltage v_c that has a slope of I_p when input is low and $-I_n$ when the input is high. As equation shows, we can easily modify these slopes with

the biases V_p and V_n respectively.

Now, we can check our initial assumption that we can ignore the dependence of the current upon the drain voltages of the FETs. The current I_p effectively is only on when the digital input is low. When the input is low, the nFET M2 is off and pFET M1 is on. No current can flow through the nFETs and all of the current must flow through the pFETs to discharge capacitor C. The vice versa case occurs when the input is high and the current flows through the nFETs to charge capacitor C. As the capacitor discharges, the voltage across the cap, $V_{dd} - v_c$, decreases. If the voltage across C is V_{dd} , then v_c is 0 V. If $v_c = 0$, the voltage across the pFET leg of the modified inverter is large enough for each pFET to be in saturation, provided that $V_{dd} \geq 2V$ at room temperature. This value of V_{dd} is sufficient until a process size of 180 nm. However, we see that if $V_{dd} = 3V$ and $v_c = 1V$, the pFETs slip out of saturation. As the drain voltage V_{midp} approaches V_{dd} (since the headroom is shrinking), the terms dependent upon this drain voltage go to 1 and can be ignored. Therefore, the drain voltages can be ignored and the value of v_c approaches a value dependent upon the difference $V_p - Vin$ instead of simply V_p . As we shrink the headroom down to 0 ($v_c = V_{dd}$), the pFETs are off. The value of v_c remains at V_{dd} until the input goes high and the process is controlled by the nFETs. The nFET case is the analogue of the pFET case.

The next stage is the floating-gate pFET. The current through a floating-gate is

$$I_{fg} = I_{0p} e^{\frac{\kappa V_{fg}}{U_T}} \quad (9)$$

$$I_{fg} = I_{0p} e^{\kappa \frac{Q_{fg} + C_{in} V_{in}}{C_T U_T}} \quad (10)$$

$$I_{fg} = I_{0p} e^{\frac{Q_{fg}}{C_T U_T}} e^{\kappa \frac{C_{in} V_{in}}{C_T U_T}} \quad (11)$$

$$I_{fg} = W e^{\kappa \frac{C_{in} V_{in}}{C_T U_T}} \quad (12)$$

where W is a weighting factor that represents the synaptic weight that can be altered through tunneling and/or injection [46]. Therefore, we have a method to produce a weighted exponential curve that has characteristics of a PSP [19], [17]. The analysis for the other synapses is similar, except that the output current for the inhibitory synapse is in the opposite direction and the feedback from the NMDA type adds a term which increases the current.

4.2 Field-Programmable Neural Array

The majority of the benefit in computing comes from its reconfigurability. Microprocessors are useful because software can be written for them to suit the applications desired by the user. Hardware reconfigurability is a relatively recent phenomenon spearheaded by the popularity of EPROMs, PLDs, and FPGAs. Now, field-programmable analog arrays (FPAAs) are beginning to grow as a field. FPAAs allow the user to connect analog circuit blocks, much like FPGAs allow a user to connect digital circuit blocks. Example blocks include amplifiers, multipliers, and single transistors [47]. These analog blocks can be arranged to form more complex circuits, such as matrix multipliers and filters, in a manner that is similar to building adders and processors with FPGAs. Reconfigurability in hardware greatly reduces the design cycle and allows for fast, low-cost systems. We have added a form of FPAA, called the field-programmable neural array (FPNA), which is comprised of analog blocks that are geared toward building neuro-inspired and neuro-mimetic systems. Much like how DSPs are specialized microprocessors, our FPNAs are specialized reconfigurable devices. Like DSPs, FPNAs are a tool that users can use to create more powerful applications than what could be realized with a more generalized device [48], [49].

4.2.1 Architecture

As seen in Figure 25, there is a full-crossbar connected matrix of elements. Thus, any element can be connected to any other element. Each element consists of synapses,

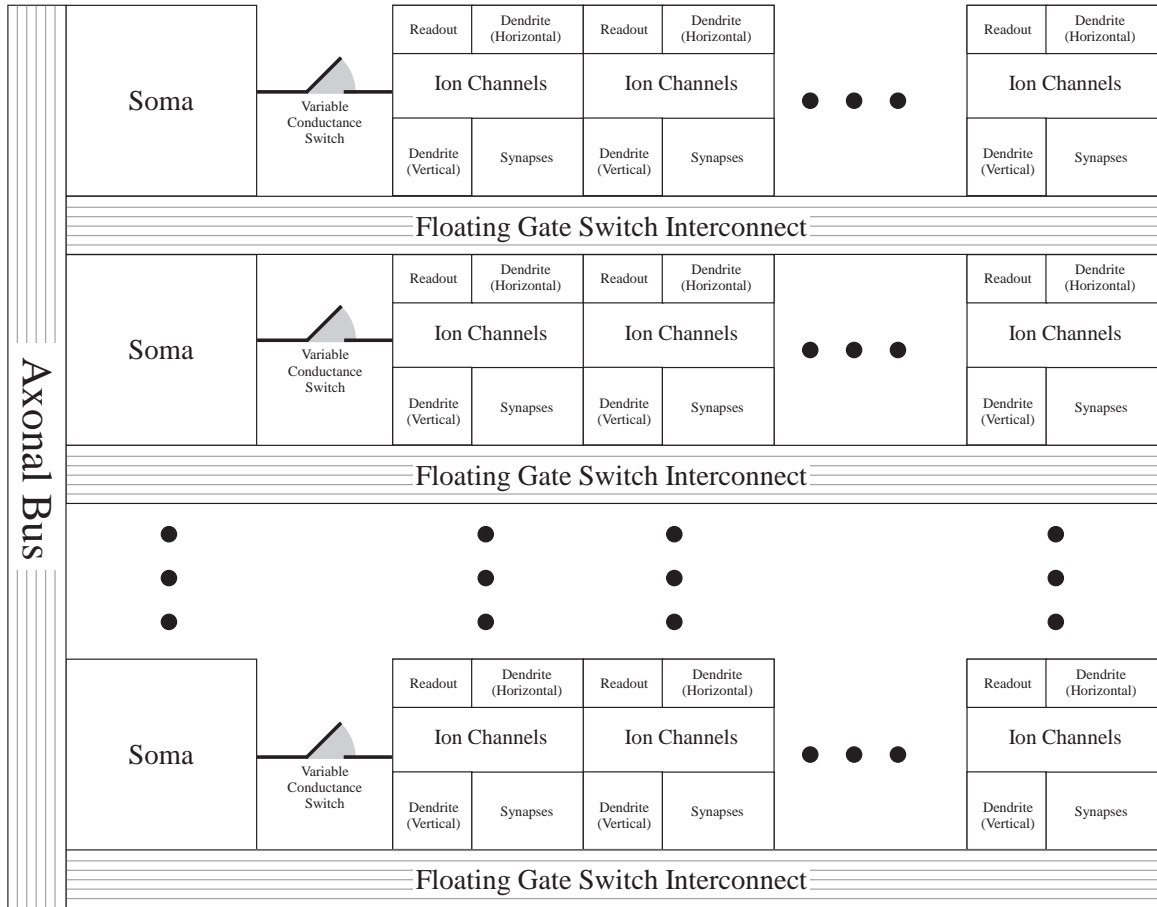


Figure 25: Schematic of the Field Programmable Neural Array (FPNA).

ion channels, dendritic sections, and readout circuitry. The latest version has two ion channels (Na^+ and K^+), two synapses (excitatory and inhibitory), two dendritic sections (horizontal connection and vertical connection), and four wide-range amplifier buffers for reading output voltages. Any of these parameters can be adjusted in future versions. These numbers were chosen to fit the following criteria in order of importance:

1. at least enough ion channels to create an action potential (Na^+ and K^+),
2. at least two synapses, preferably one excitatory and one inhibitory,
3. pitch-matched to the smallest area possible,

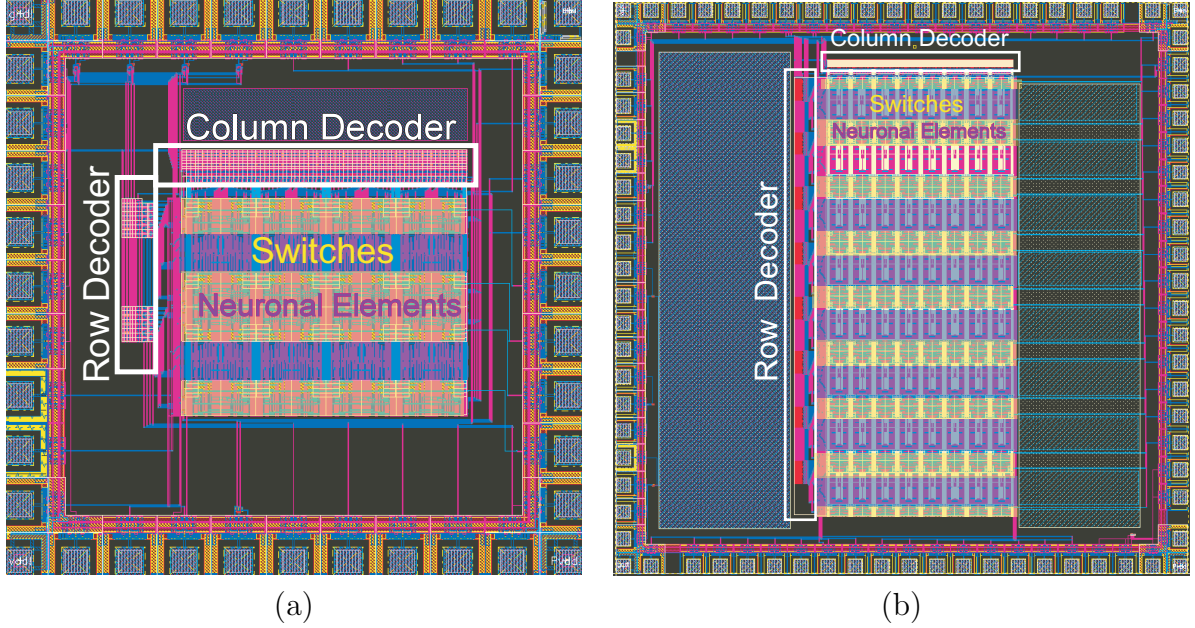


Figure 26: (a) Layout for the first version of the FPNA. This was fabricated by MOSIS on a AMI $0.5 \mu\text{m}$ process. This chip is $1.5 \text{ mm} \times 1.5 \text{ mm}$ and has 40 pins. (b) Layout for the second version of the FPNA. This was fabricated by MOSIS on a TSMC $0.35 \mu\text{m}$ process. This chip is $3 \text{ mm} \times 3 \text{ mm}$ and has 64 pins.

4. dendritic connections to connect in both dimensions of the matrix,
5. at least one circuit to read voltage output.

Criterion 1 required six floating gates to program and a large area (to create signals on the same timescale as biology). Criteria 2 and 4 raised the number of floating gates to 10. The final area was limited to 8×8 decoder-bit sections since the area of the ion channels was so large. The leftover space was used to fit the readout circuitry. A pitch-matched wide-range amplifier mentioned in the was used to create a voltage-follower buffer. This provided the best voltage response compared to other circuits such as operational transconductance or two-transistor amplifiers. There was enough room to fit four of these structures. The output of each row is sent to the soma block. The soma creates the triangle-shaped waveform used in the synapse circuit and is similar to biological data [17]. This output and external inputs are sent to other areas of the circuit depending on the arrangement of switches in the floating gate

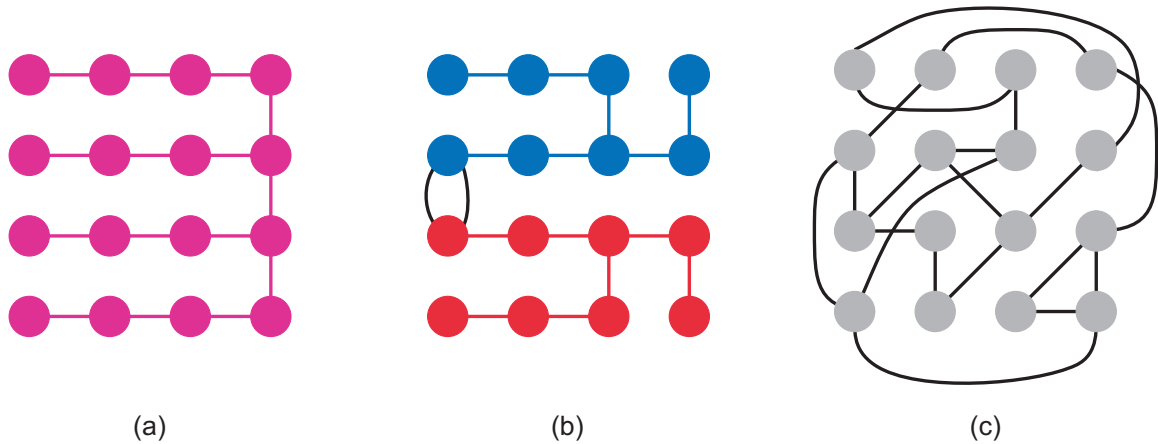


Figure 27: Networks that can be created using the FPNA. (a) One complex cell. (b) Two less complex cells forming a simple CPG (half-coupled oscillator). (c) Many simple cells connected in a complex network.

switch network. This switching scheme allows for full connectivity throughout the matrix. On the initial implementation, 8192 connections can be made. The second-generation FPNA can achieve more than 12K connections. The layout for this chip is shown in Figure 26b; the first version is illustrated in Figure 26a.

4.2.2 Applications

This structure is used for a range of applications. Complex models of cells can be made by modeling small sections of the cell in each subblock of the FPNA. Small central pattern generator networks can be created by connecting a few approximate models of neurons. Large neuronal networks can be created by connecting hundreds of the individual blocks. Figure 27 illustrates the variety of networks that can be achieved with this chip.

4.2.3 Improvements

The initial prototypes of the FPNA heavily relied upon indirectly programmed floating-gates. Unfortunately, this dependence made programming impossible. Furthermore, analyzing currents within the FPNA to determine the operation of the synapses is

not possible since the outputs are buffered voltages. The next generation chips, called biological FPAA's or BioFPAA's, were made by Basu and Ramakrishnan [50]. Data collected by Basu and others showed that neurons and synapses could be made to work individually. The next generation chip was designed by Basu and others to make the indirectly programmed floating-gates more viable.

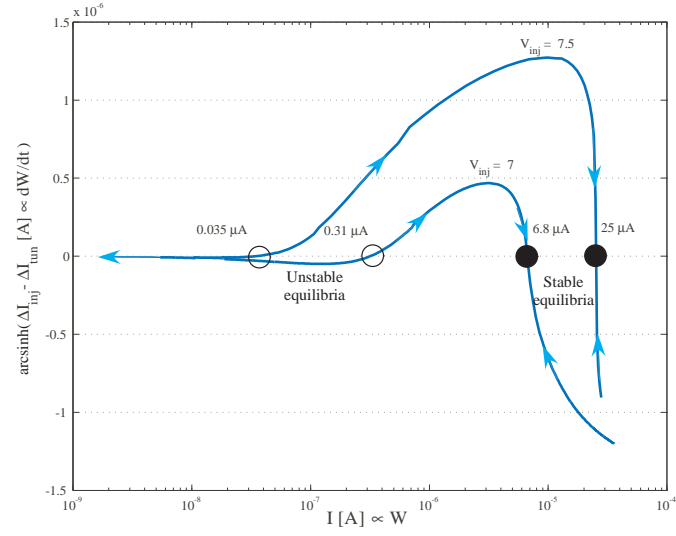
CHAPTER V

METHODS OF LEARNING

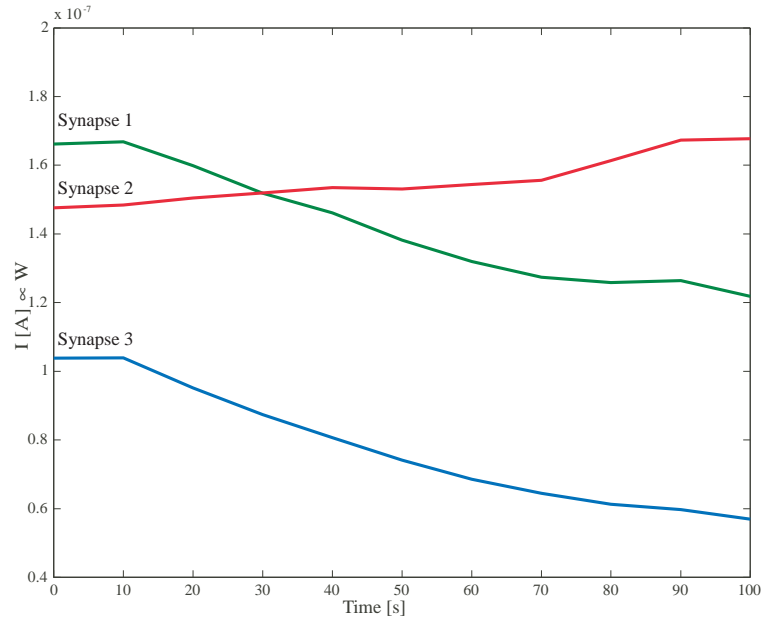
Learning rules to explain how inputs and outputs can change the synaptic weights within a system. Although learning rules were first used to describe and implement artificial neural networks, biology has adapted to using them as well. Understanding learning and memory in biology has resulted in the discovery of many, but not all learning rules. This chapter describes how learning rules have been explored in the previously described synapses. I initially explored the effects of implementing learning with a simple floating-gate called a single transistor learning synapse (STLS). Taking advantage of the rates of injection and tunneling, I found that Hebbian learning can be implemented. This previous work involved providing a rectangular, digital pulse directly to the input of the floating gate. However, the bio-mimetic synapses receive this digital pulse at the overall input; the input at the gate is a triangular pulse. So, experiments with this synapse were performed to explore the types of learning that could occur.

5.1 Previous Work

Using a simple synaptic circuit, networks with Hebbian-type adaptation rules have been realized. With increased synaptic activity, the synaptic weights are increased or decreased. That increase or decrease continues with subsequent synaptic activity. This work explores the relationship between synaptic activity and weight for various inputs. Figure 29 is the schematic of the circuit that realizes this form of LTP and LTD. As seen in Figure 28, the system will reach an equilibrium point; either it will be pruned away to zero, or it will move toward some stable synaptic weight. The movement of three synaptic weights is illustrated in Figure 28. The importance of



(a)



(b)

Figure 28: (a) State-space analysis of learning rule. Two cases of simultaneous tunneling and injection are explored. As illustrated, equilibrium points can be shifted to desired locations by varying the tunneling and injection voltages. Furthermore, the trajectories of the weights can be determined for a given initial condition. The arcsin function is used to magnify the detail of the data. (b) Experimental measurements from a 3 synapse system, where one synaptic input (Synapse 2) is visited significantly more than the remaining 2 inputs. Weight adaptation at a particular synapse is dependent upon the occurrence of input pulses with output pulses. With an increase in synaptic activity, the probability of input and output pulse-overlap also increases. The overall network adaptation rate is dependent upon the tunneling and injection voltages. We illustrate the case where we have a combination of LTP and LTD occurring in the system.

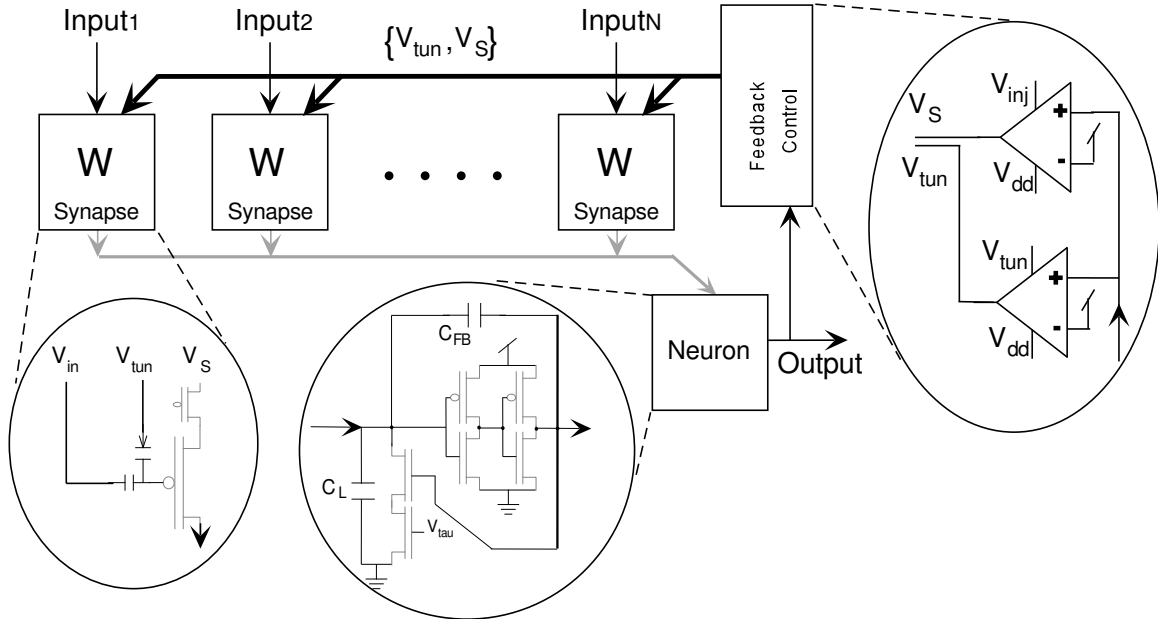


Figure 29: Basic floating-gate circuit for an action-potential learning system. Our synapses are a single floating-gate pFET device with a DIBL transistor for stability. The neuron element is an integrate and fire neuron that can be easily replaced with a more biologically accurate model. The feedback circuitry moves the source (V_s) and tunneling (V_{tun}) voltages after the onset of an action potential. The current from each synapse is summed along a wire; extending this circuit to biological type dendrites would be straightforward.

the dynamical nature of neuronal systems has been used in creating software [51], [52] and hardware models of neurons [53] as well as synapses based in software [54]. This work was the first to illustrate a synapse with a dynamical nature that could also be used for interfacing to living cells.

I have already created simpler systems that can modify synaptic strength in a biological fashion [1]. Biologically, synapses strengthen through long-term potentiation (LTP) in which pharmacological and morphological changes are made to improve signal transduction from the pre-synaptic to the post-synaptic cell. After LTP, the PSP is much stronger than it was before LTP. Conversely, long-term depression (LDP) decreases synaptic strength such that the PSP becomes weaker. We will incorporate our previous work with these circuits to move from programmable synapses to

self-adapting synapses.

5.2 *Experiment*

In order to explore the intrinsic learning rules that can be found with a floating gate, I used a STLS with variable inputs to the input, tunneling node, and V_{sd} . All of the essential inputs to the whole system are spikes. Recall that the input to the synapse is a spike, while the input to the STLS within the synapse is a triangular waveform. I inputted a triangular waveform to the STLS to simplify the full effect of inputting a pulse to the synapse. This simplification allows analysis that is independent of the biasing values of V_n and V_p . By varying the input values in size and time, spike-timing dependent plasticity (STDP) can be explored in its basic form. An example input sequence is illustrated in Figure 30. Although the known learning rules found in biology exhibited in Figure 5 are an example of what could be possible, all learning rules have not been discovered.

Inspired by the experiments of Bliss and Lømo and Bi and Poo, the initial synaptic weight was measured, then a particular input sequence was repeated several times, and the final synaptic weight was noted [55], [29]. Every new experiment began with re-initializing the synaptic weight to the same value. If the initial synaptic weight were changed, equivalent curves would occur since the the peak injection rate would occur for an equivalent current. The curves would be more flat for a larger initial current since the rate of injection decreases for above-threshold currents. However, curves would have the same overall shape for these large currents. Similarly, initial currents that are very small (on the order of 100pA) provide injection rates that are less than optimal, but those rates are still larger than above-threshold injection. So, initial values that were close to the optimal range were chosen.

The learning rules discovered represent the forms of unsupervised learning that found in the STLS. Since floating-gates can be programmed, it is possible to use an

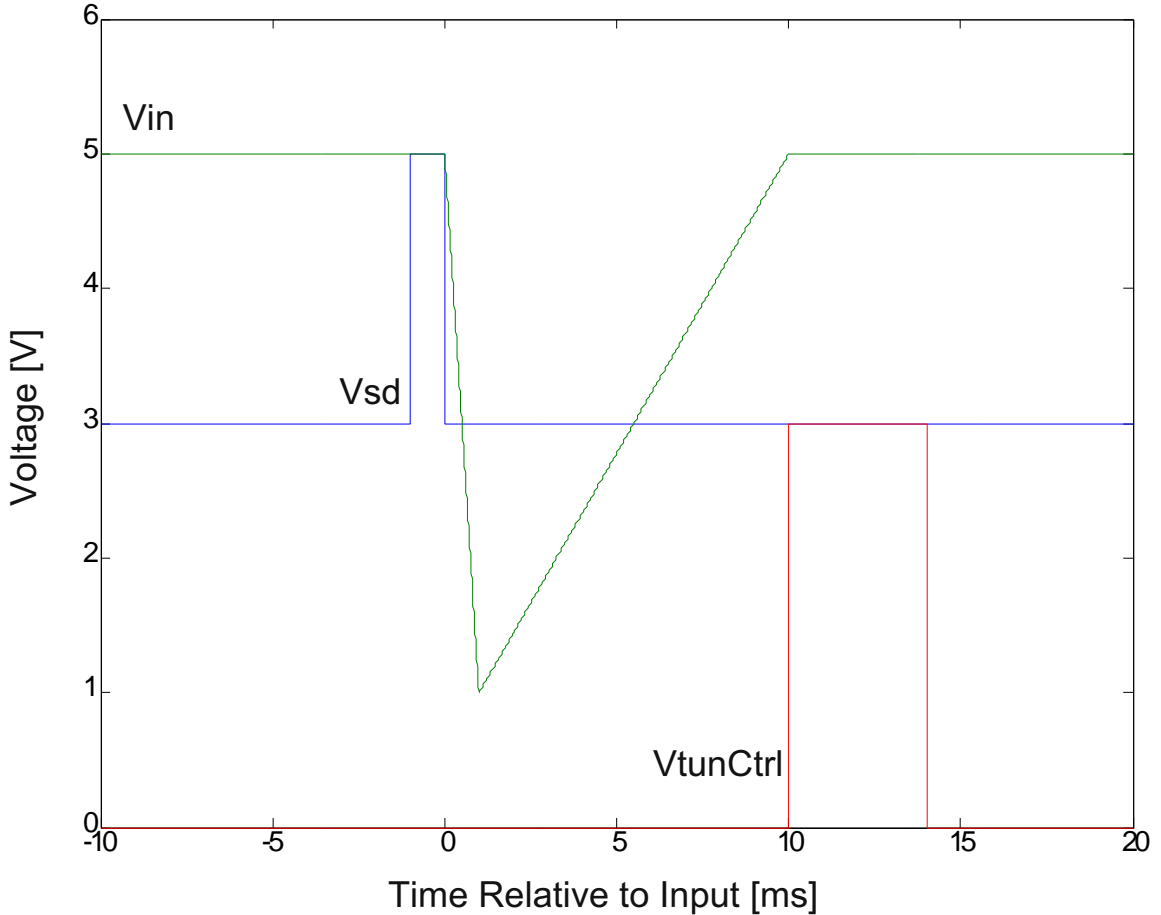


Figure 30: Plot of the inputs to the single transistor learning synapse.

external system to force the synaptic weight to arbitrary values. Rather than using hardware to set injection and tunneling voltages, a computer can be used to program the weight based upon software implementing a desired learning rule. This is a trivial exercise and is not implemented in this work.

5.3 Data and Discussion

The benefits of using an artificial system were obvious by the second experiment. Bliss and Lømo and Bi and Poo needed to prepare another rabbit or rat in order to conduct a new experiment. I simply programmed the synaptic weight to the initial value and began the experiment with a different timing difference. Repeating experiments A,

B, and C stereotyped in Figure 4 simply took a few minutes and not days.

Software yields a similar benefit, with some detriments. The detriments of software include, but are not limited to, the inability to produce real-world signals that are of biological levels, the inability to scale easily, and the need for a high-power device such as a processor. These are not detriments for most modelers. However these disadvantages are crippling when seen in the greater context of attempting to create a fully integrated system that can be implanted in vivo. Therefore, using artificial devices to implement these learning rules provides fruitful results that are most beneficial in the long-term.

The data collected in the artificial system yields three distinct learning rules in a STLS created in a $0.6 \mu m$ process. The data shows that the weight can change greatly, can be tuned for a specific timing, or can increase or decrease. The rules are shown within a window around the input. Empirically, the synaptic weight was not altered noticeably outside of the area shown in this work.

Outside of the time in which the input is on, there is no injection. Injection only occurs when current is flowing. Since there is no current when the gate is at Vdd, no injection can take place. Therefore, the synaptic weight cannot change if an injection pulse takes place outside a prescribed window. Conversely, the delay in tunneling causes the window for decreasing the synaptic weight to be more difficult to judge. Tunneling can take place regardless of the value of the gate voltage. However, the delay causes interesting effects described here.

5.3.1 A Greatly Increasing Rule

The synaptic weight obviously increases greatly if only injection is employed and tunneling is left off, as seen in Figure 31. Figure 31 shows a single, mountainous peak with negligible change outside of the presence of an input. Here, the peak synaptic weight change occurs when the gate is about 3.5V (near threshold). However, the

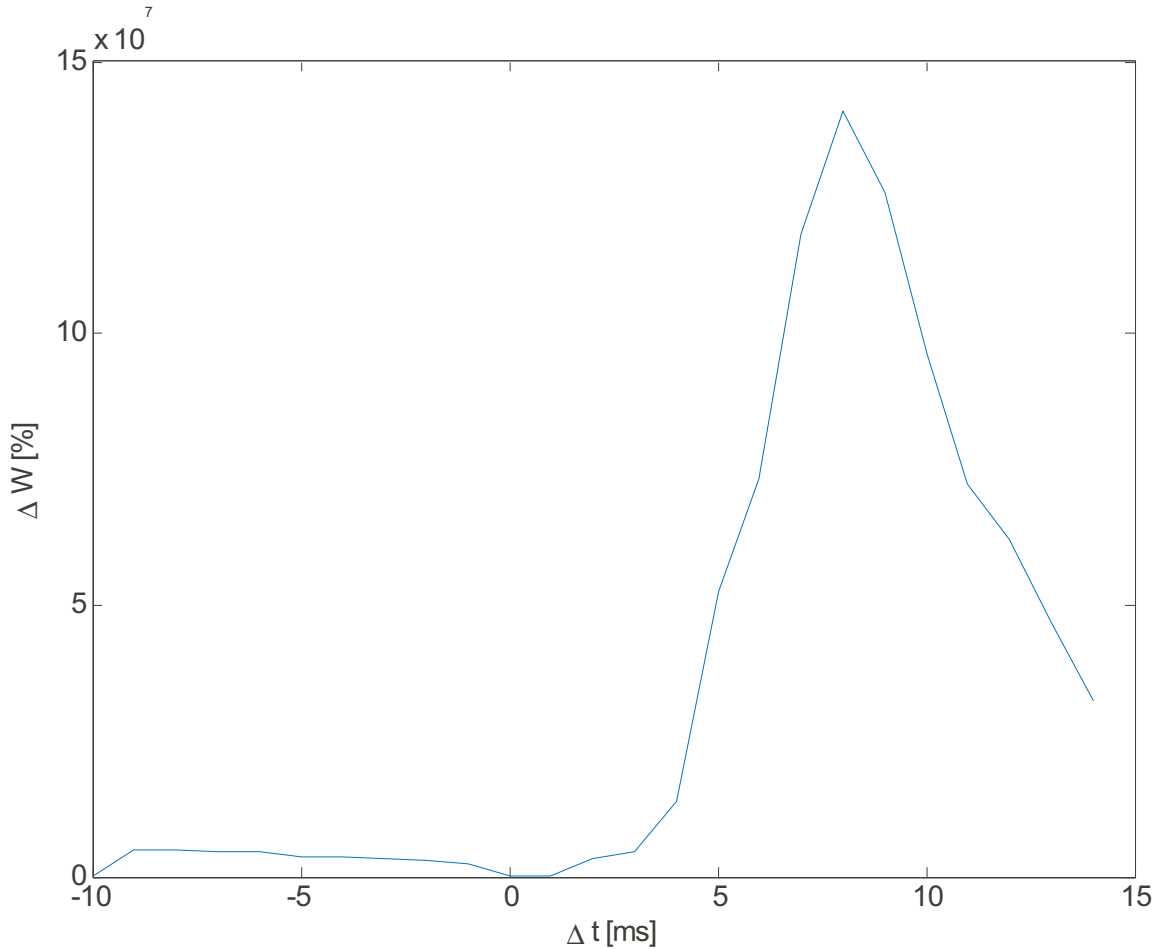


Figure 31: A learning rule that increases the weight most efficiently due to the current value through the floating-gate. This form of the rule takes place when there is no tunneling at all. Note that the peak change in synaptic weight takes place for a current that is near the threshold.

maximum change in weight does not occur for the lowest input level (and thus the largest current through the floating gate). This is due to the fact that the injection rate is more efficient right at threshold as seen in Figure 32. So, the maximum change in current occurs when an output is activated when the gate voltage provides the current closest to this optimal injection value. The peak represents a change in current that went from the nA range to the mA range in one step. This huge change in synaptic weight is due to the fact that the current began in an optimal range and that there was no tunneling.

The data in Figure 33 show that the tunneling pulse is insignificant. Here, the position and magnitude of the peak is altered by the tunneling pulse. When tunneling is on, the current through the STLS can go down to zero. Therefore, injection can be prevented by the presence of a large V_{tun} value. Figure 33 shows the effect of muting the effects seen in 31 with tunneling. This effect produces a five-fold increase from the original synaptic weight rather than the increasing in many orders of magnitude. Tunneling takes place within the window of injection. The actual control pulse is administered before the input, but due to the delay in tunneling, the effect is not experienced until later.

The optimal injection point occurs at a timing difference of 4ms, as seen in Figure 33. Figure 30 show that a time of 4ms, the input value is 2V. The other time that the input is at that value, thus producing an equivalent current, between 0 and 1ms. Since the pulses are 1ms long, any pulse that occurs from 0 to 1ms will not have the same effect as a pulse between 4 and 5ms. This is due to the fact that the equivalent gate voltage is optimal for a small fraction of the pulse. Whereas, the pulse overlaps a longer portion of an optimal gate voltage from 4 to 5ms. Therefore, the change in current is greater at 4ms than at approximately 0.5ms.

5.3.2 A Tuned Rule

The second rule shows that the efficacy of weight change can be mitigated by the presence of the tunneling pulse (Figure 34). This result was found for $V_{tun} \geq 14V$ and when the tunneling pulse takes place within the first 4 ms of the input. Larger values of V_{sd} did not produce a different curve. Higher values of V_{tun} did not produce a curve that was noticeably different either. For lower values of V_{tun} , the distinct double peak turns into one peak and results in a variation of the first rule.

Also like the first rule, the presence of tunneling decreases the the maximum synaptic weight change possible. The double peaks are do not even double the initial

synaptic weight. The slight dip at a time difference of 2ms is misleading. This is less than a 10 percent change and it is within the error of the system. This setup does not produce a true decrease in weight. Further proof is in the fact that higher values of V_{tun} did not make this effect more pronounced.

5.3.3 A Rule that can Increase and Decrease

The third rule, seen in Figure 35, can both increase and decrease the synaptic weight. This rule is interesting since it allows for optimal "remembering" for a specific tuning between input and output, while providing optimal "forgetting" for another tuning. The timing between these tunings allows for even a sharper discrimination of the timing difference. If the timing difference is 4ms, then the synaptic weight is increased greatly. However, if the timing is stretched by just one millisecond, then the synaptic weight is decreased. Therefore, the relationship between input and output must be exact. Connections between inputs and outputs that occur at the right time are greatly strengthened while slightly tardy timing differences result in the diminishing of connections.

This rule only applies when the the value and timing of the tunneling pulse is on as illustrated. For lower values of V_{tun} , the decreasing weight simply goes to zero weight change or to a slightly positive weight change. For values of $V_{sd} \geq 5V$, the negative weight change becomes zero or a slightly positive weight change.

5.3.4 Creating Additional Rules

As mentioned previously, V_{tun} and V_{sd} have been limited to pulse inputs within certain voltages. If V_{tun} and V_{sd} were different, then different learning rules can be manifested.

Altering the fundamental expectations of when injection is on will also result in different learning rules. For the previous experiments, injection was on only when there was an input. Injection was optimized for certain input values. Injection was

off when the input was off because the gate was set to a value to provide no current. Hence, only one of the two necessary components of hot-electron injection occurred. However, if the input is altered so that injection can occur at anytime, the number of possible learning rules is expanded.

An example of a learning rule in this expanded set is created when a input pulse is provided that does not turn all the way off. The input to the STLS was set to a maximum of 3V instead of $V_{dd}=5V$. If V_{sd} is pulsed at 5V when the input is only 3V, injection can occur since there will be a current. So, timing differences that are outside of the normal realm can create increases to the synaptic weight. This results in a learning rule as seen in 36. Here, the maximum changes to the synaptic weight occur outside of the time that the input pulse is present. This is because the optimal injection rate takes place for the subthreshold currents that occur when the gate is 3V. The currents when the input is on are much higher and result in a less efficient injection rate. Therefore, another learning rule can easily be created by modifying our initial expectation of being able to turn injection off. Altering the requirement of using pulses at the main inputs creates an even larger number learning rules.

Since there is a potentially unlimited number of input possibilities with unlimited timing differences, the entire space of possible learning rules are too large to explore through brute force. So, further explorations of the possible learning should take a targeted approach. An example is to attempt to reproduce the learning rules found in biology (Figure 5). For instance, learning rules (a) and (b) can be created. To change tunneling so that it is stronger for negative timing differences, one simply needs to change the tunneling waveform. An example waveform is to use a sawtooth, rather than a rectangular, pulse. The maximum of the pulse occurs first and the minimum occurs last. Changing the pulse length alters the slope of sawtooth. This waveform should result in learning rules (a) and (b) seen in Figure 5 depending upon the slope of the waveform.

5.4 *Effect of Ca^{2+}*

Calcium is a common ion used in many cells to regulate various processes. In neurons, calcium has been shown to be involved in the mechanisms for learning [17], [19]. The presence of Ca^{2+} augments the effect of the action potential by making the post-synaptic cell depolarize. The depolarized neuron can be brought to threshold more easily and produce an action potential. The time scale of the influence of Ca^{2+} is from the ion channel (voltage-gated calcium ion channels) level to the second messenger level (calcium-calmodulin-dependent kinase II) [17], [19]. This work will be restricted to the same time scale as the operation of ion channels in order to emulate basic learning functions.

Cellular learning stems from the presence of an input and an output action potential [17]. When an action potential occurs, the entire cell depolarizes. When an excitatory input is received, the area near that synapse depolarizes. The combination of both signals produces a stronger synaptic weight [17]. Calcium has the effect of modulating the strength of the synaptic weight change [19].

The basic excitatory synapse has been modified to include the the effect of Ca^{2+} . As seen in Figure 37, a second input has been added to the floating gate. This second input is present on every synapse. This input is controlled by circuitry to emulate the dynamics of Ca^{2+} . The controlling circuitry can be shared for every row, column, or matrix of electronic synapses. The effect of Ca^{2+} is enabled with a digital signal SelectLearn. When SelectLearn is low, the secondary input is off and the synapse is effectively the same as the previous version. When SelectLearn is high, VinCa goes to a voltage that is inversely proportional to the concentration of Ca^{2+} in the area around the synapse. Thus, the secondary input can adjust the output of the electronic synapse. The lower the $[Ca^{2+}]$ pool voltage, the larger the resulting EPSP. A larger EPSP produces a stronger depolarization and when coupled with the presence of injection can result in a greater change in weight. Here, an indirect

programming structure is used for injection. Injection pulses are introduced on the programming pFET. These pulses can be due to either adaptation or programming. In biology, the initiation of learning depends on coincidences that take place on the order of milliseconds. This system is able to meet this criterion and more. Although precise values are not necessary to emulate biology, pulses on the order of $10 \mu s$ are attainable when programming [38]. Moreover, the effective weight change can be achieved within one pulse [38].

This circuit was fabricated through MOSIS in a $0.6\mu m$ process. The delay in V_{tun} was experienced in this chip and created problems with estimating the time at which tunneling should occur. This circuit, and similar circuits, will be reinvestigated by Shubha Ramakrishnan and Arindam Basu in their biological Field Programmable Analog Array chip. That chip was created in a $0.35\mu m$ process and the tunneling delay should not be an issue.

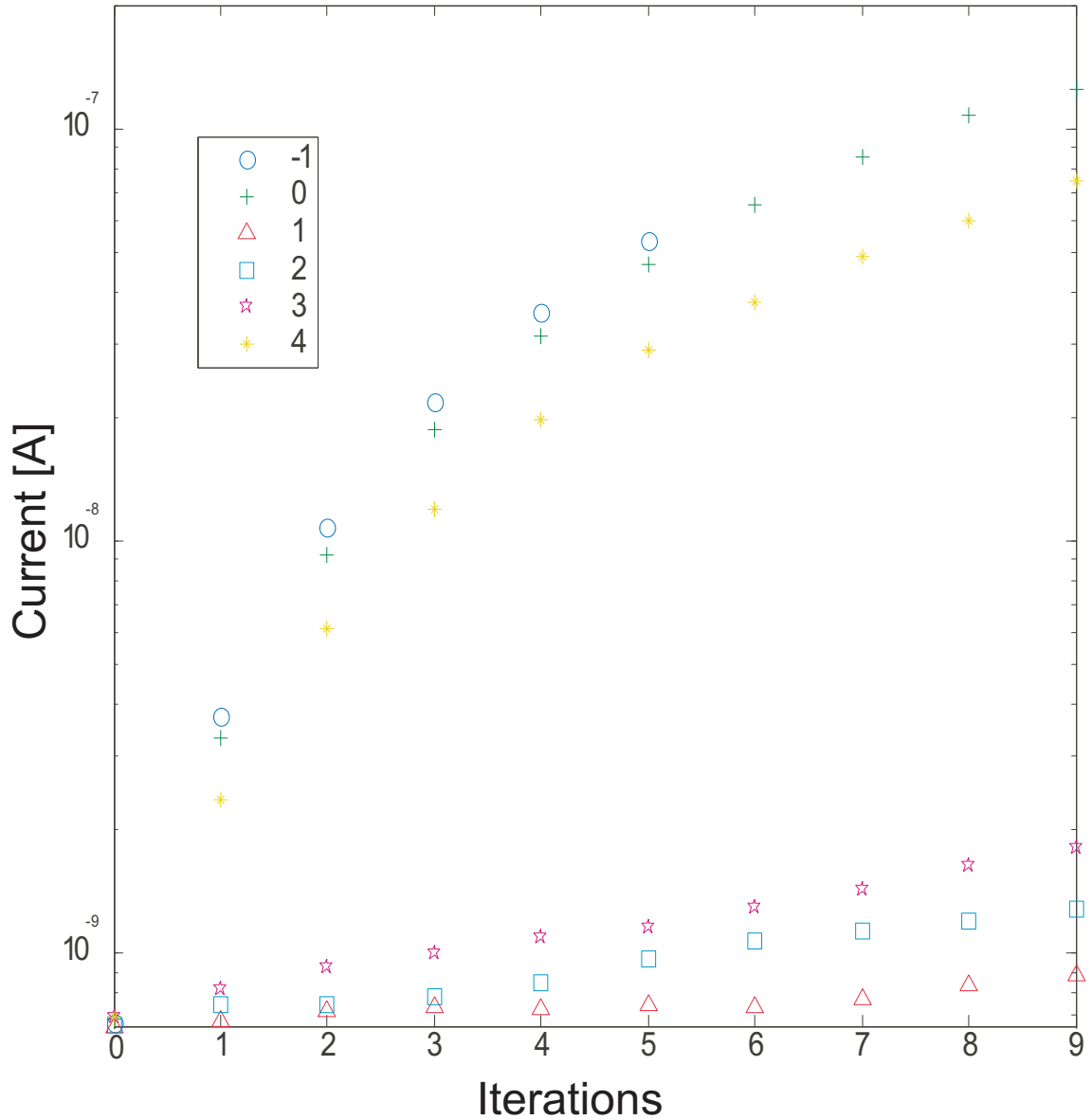


Figure 32: Change in synaptic weight per iteration. The timing differences from $\Delta t = -1$ to 4ms are represented with different shapes. The rate of change in current is large for a difference of -1, 0, and 4ms. Thus, these timing differences produce a relatively large synaptic weight change every iteration. Timing differences of 1, 2, and 3ms result in negligible changes in current, therefore producing a slow synaptic weight change. Note that timing differences -1, 0, and 4ms produce currents that begin to plateau since they have reached absolute values that are above threshold.

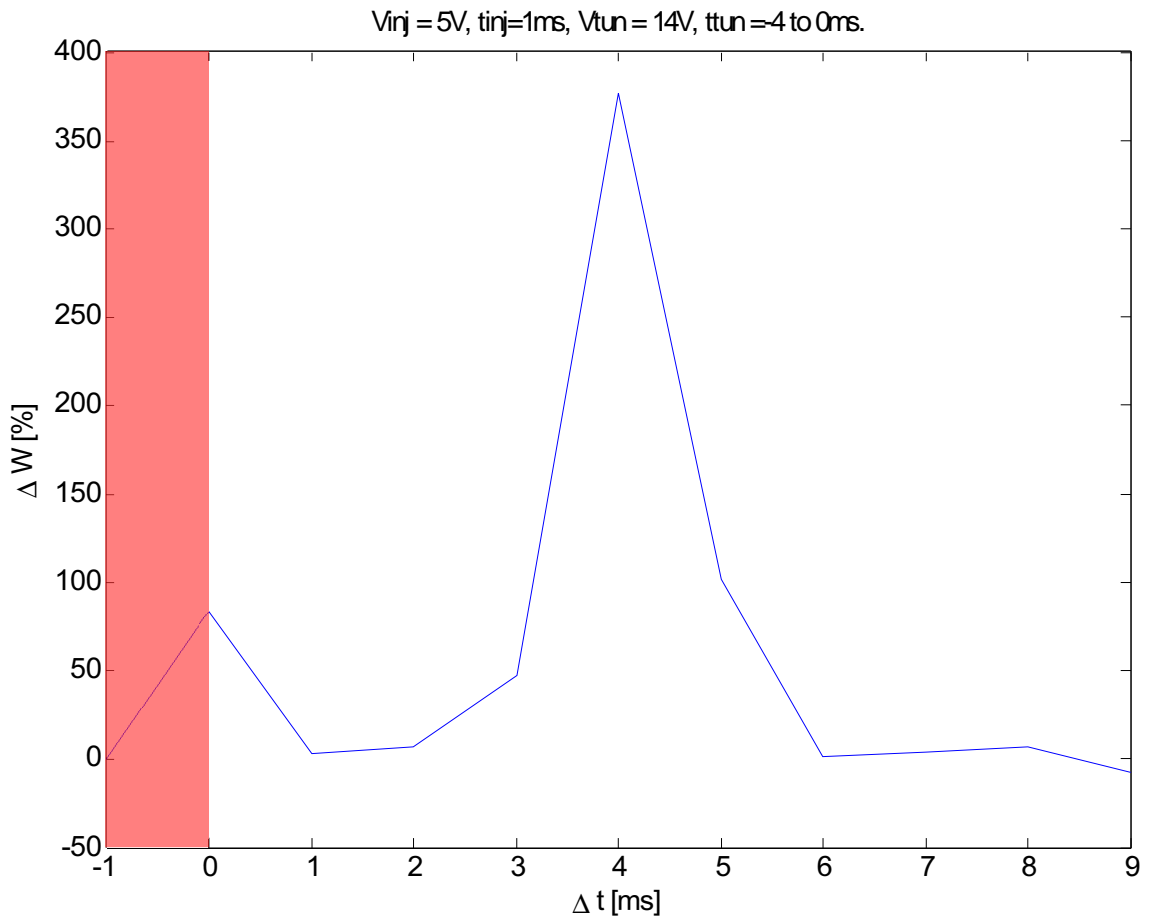


Figure 33: A learning rule that increases the weight most efficiently due to the current value through the floating-gate. This rule takes place if the tunneling pulse only occurs when the gate is off and the injection pulse is near the time that the current is near threshold. Note that the translucent red indicates when the tunneling pulse is on.

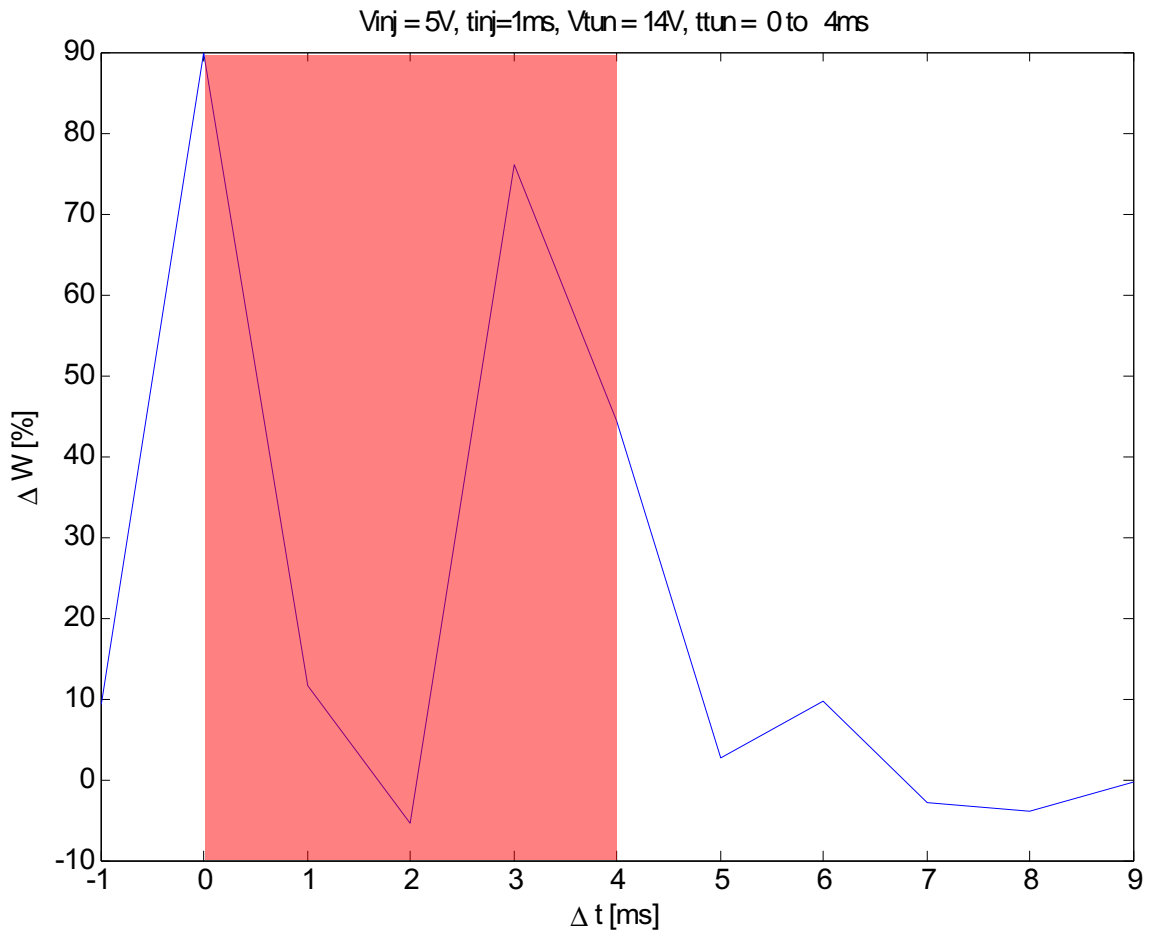


Figure 34: A learning rule with a tuned response. This response is tuned by the timing of the tunneling pulse. Note that the translucent red indicates when the tunneling pulse is on.

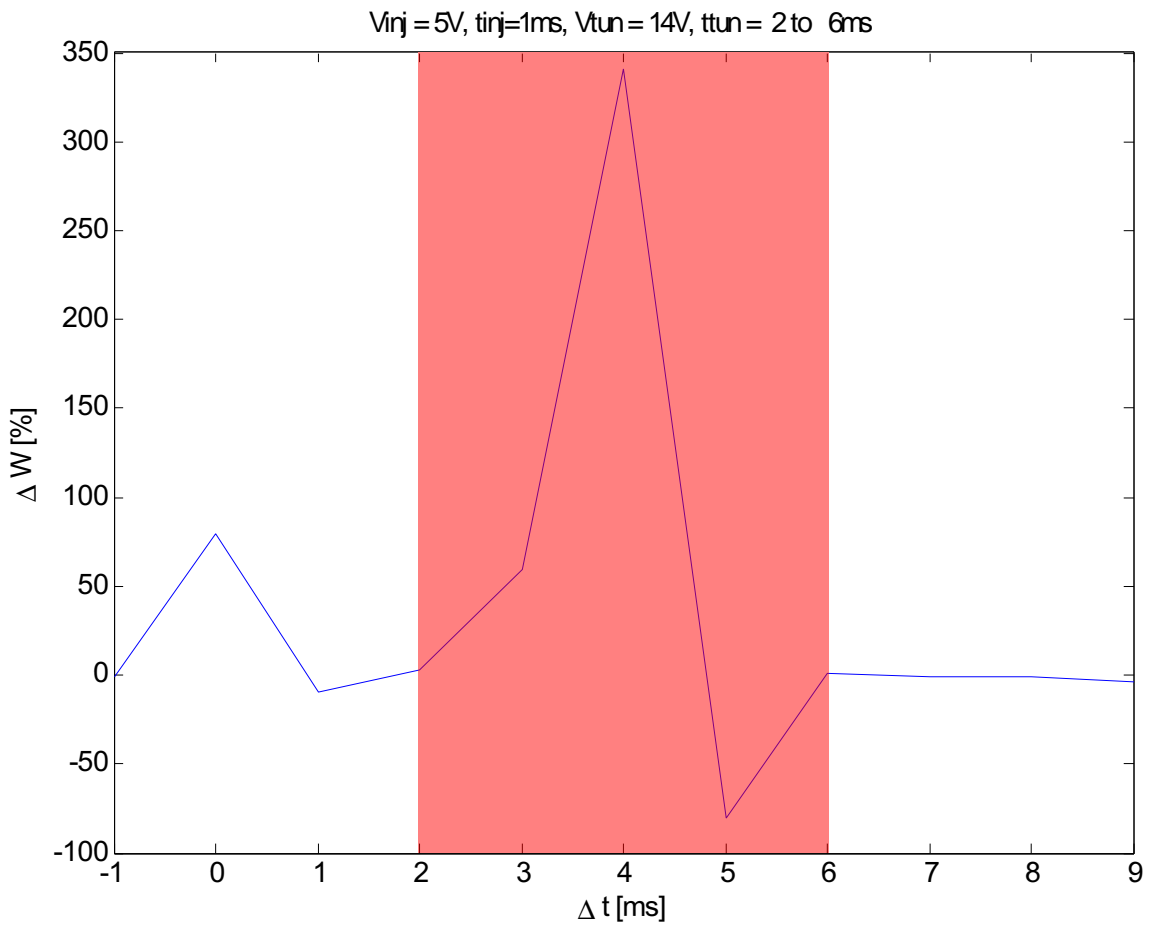


Figure 35: A rule that can change the synaptic weight positively and negatively. Note that the translucent red indicates when the tunneling pulse is on.

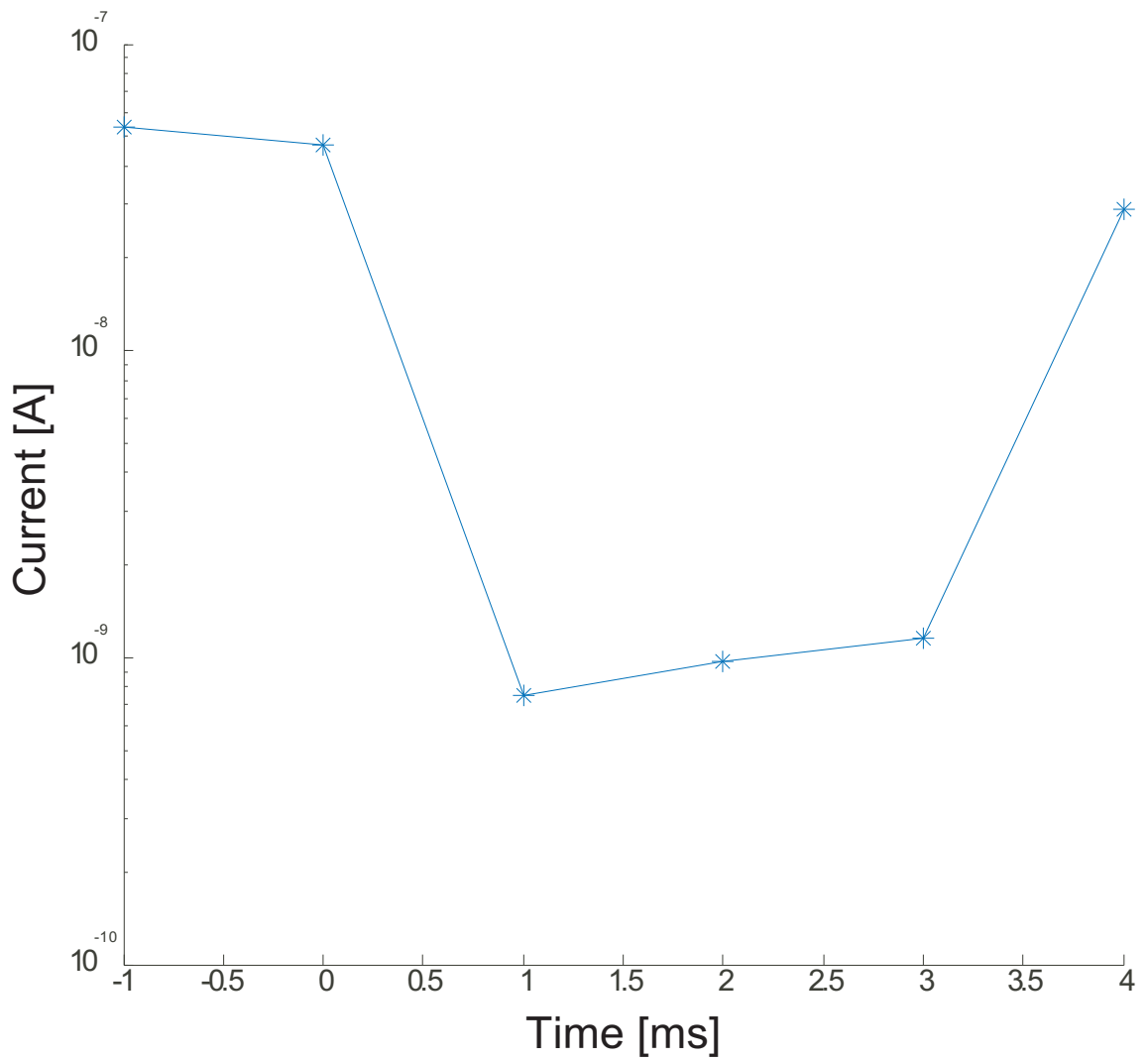


Figure 36: A rule that increases synaptic weight when there is no input pulse. Note that the translucent red indicates when the tunneling pulse is on.

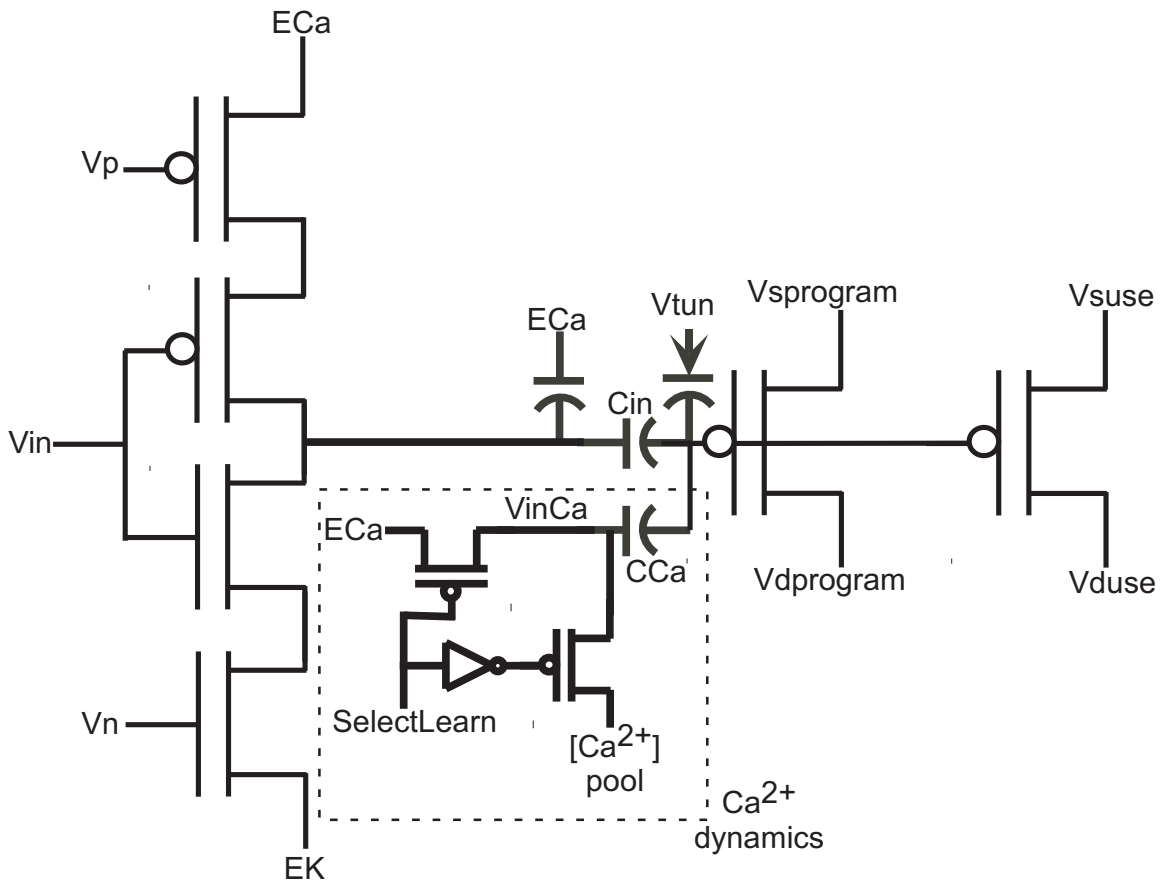


Figure 37: Modified ACh synapse which uses indirect programming and the concentration of Ca^{2+} . The subcircuit which models the effect of Ca^{2+} on LTP is highlighted with a dotted box around it.

CHAPTER VI

NEURAL INTERFACING

Interfacing to living, neural tissue provides many interesting engineering challenges. These issues include: reading signals of a small value, low frequency, and from many locations. These issues are typically encountered by analog circuit designers when creating systems for applications such as cellular telephone service miles from the nearest tower to detecting a few photons in an orbiting telescope. However, designing for biological interfacing has the added danger of destroying the specimen under test. A typical designer does not need to worry about destroying a cell tower whenever a user speaks into a cell phone. Therefore, the challenges in designing a neural interfacing system are more precarious than typical analog circuit design.

The low-power, low-noise circuits described in this chapter provide a foundation that allow a neural interfacing system to be developed. Intracellular neural signals are on the order of tens of mV. Extracellular signals are far more difficult to read at levels of tens of μV . Local field potentials (LFPs) are an even greater challenge. LFPs are extracellular signals for a general area of the brain with frequencies that dip into sub-hertz levels. A greater understanding of brain areas such as cortex, thalamus, and hippocampus has been gleaned from studying this aspect of electrophysiology [56], [57], [58], [59], [60]. These low frequencies are easily drowned out by flicker noise. A key component of the neural amplifier to help with flicker noise is a resistive element. This element allows for resistances on the order of gigaohms over a limited operating range. This high resistance allows for sub-hertz operation so that the system can be used to measure LFPs. I have made two critical improvements to a previously revolutionary circuit by Harrison and Charles [61]. The first improvement

is a reduction of the physical size of the circuit so that a higher density of amplifiers can be put on a die. The more amplifiers, the more readings can be taken from the neural tissue. The second improvement is an expansion of the operating range of the resistive element. I have produced a small, programmable resistive element that does not increase the footprint of the amplifier. This improved resistive element allows the amplifier to operate in a nearly rail-to-rail operation while providing gigaohms of resistance.

The amplifier system is used to read in signals and cannot provide outputs. A stimulator is needed to send signals to neural tissue. The values provided by the aforementioned excitatory synapse were fed to living neurons in a series of experiments. This synapse was used to successfully interface with a pair of living neurons. Although the system worked well, it used bench-top lab equipment to help provide the proper current and voltage levels. The proper electrical interface must be maintained to prevent damage to the electrodes or the cells.

A charge balancing system was developed to replace the extra lab equipment. This charge balancing system works for any arbitrary input current in a range from 100 pA to 37 μA . This operating range stems from the fact that this system was designed to work with the synapses mentioned above.

Details of the amplifier, resistive elements, synapse, and charge balancing system are described in this chapter. The amplifier, resistive elements, and synapse were all empirically tested [62]. The charge balancing system was simulated [63], [64].

In all, these circuits are shown to provide a useful toolkit for interfacing to neural tissue. Depending on the values programmed in the resistive element and the size of capacitances, these tools can be used for measuring LPFs and other extracellular signals and intracellular signals (action potentials or post-synaptic potentials). This toolkit can be useful for single-cell, slice, and in-vivo recordings.

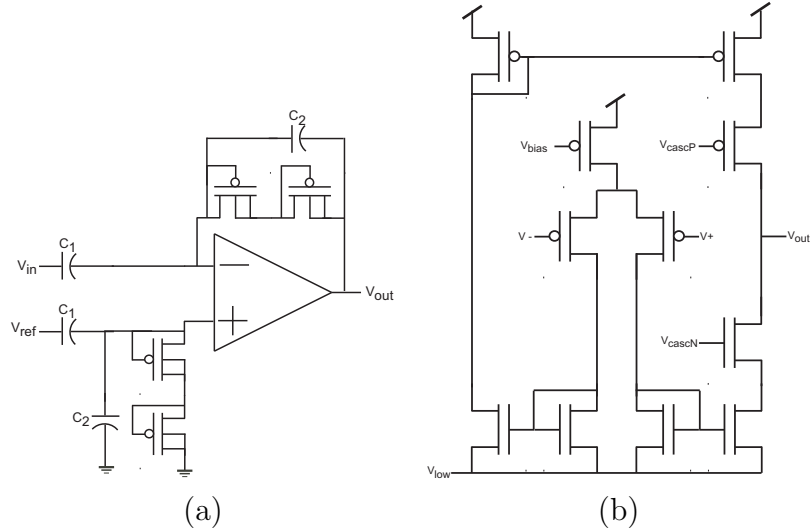


Figure 38: (a) Neural amplifier based on the work of Harrison and Charles. The capacitor values are $C_1 = 100$ fF and $C_2 = 10$ pF. (b) The wide-range amplifier component of the neural amplifier.

6.1 Neural Amplifier

The interface from biological cells to on-chip systems is made with an amplifier, illustrated in Figure 38. The amplifier is similar to the low-noise, low-power wide-range amplifier described by Harrison and Charles [61]. This amplifier exhibits excellent noise and power performance. This circuit uses microwatts of power while attaining a gain of nearly 40 dB [61]. Thus, a die with hundreds of these elements will still use little power. Multiplexing hundreds of inputs and outputs can be achieved using flip-chip technology and electrode arrays (such as the Utah or Michigan arrays) or through a multiplexing circuit [65],[66]. The first method was not available to me since I did not have access to micromachined arrays. The second method was explored and provided promising results detailed in this work.

6.1.1 Design Considerations

This implementation uses smaller capacitors for C_1 and C_2 (100 fF and 10 pF, respectively). The large input capacitors help reject input noise and the capacitor ratio C_1/C_2 sets the gain. The smaller capacitor C_2 is the dominant determination for the

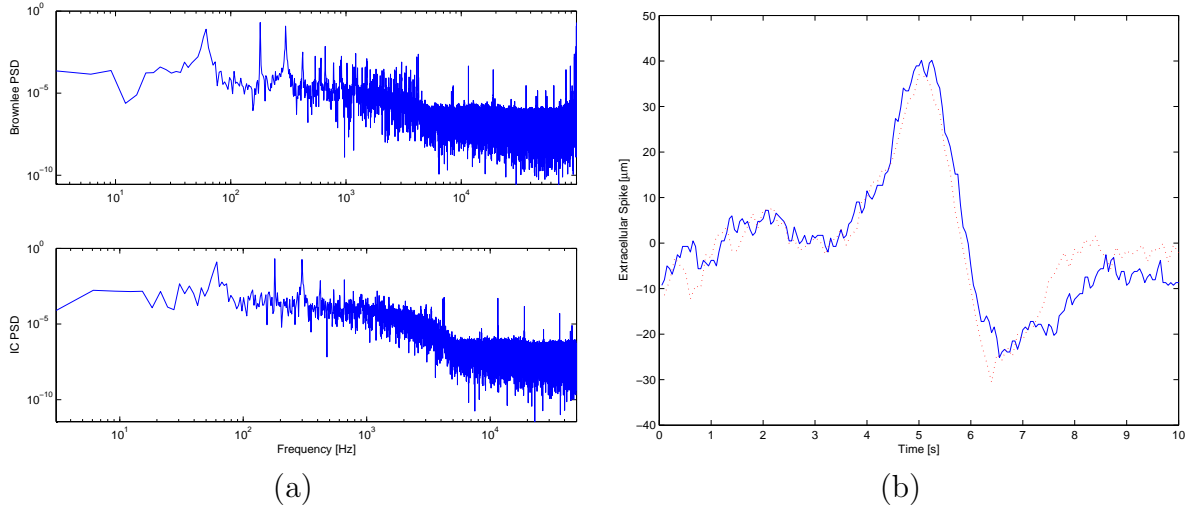


Figure 39: (a) Power spectral density plots of extracellular noise recordings from an Aplysia cell. The results are similar despite the great difference in size between the rack-mounted Brownlee amplifier and the amplifier IC. The integrated amplifier has an ideal gain of 100, while the Brownlee amplifier has a gain set to 750. In order to better compare typical readings with the amplifier IC, an external gain of 7.5 was added by connecting the output of the chip to another Brownlee input. (b) Superimposed spikes from the amplified extracellular recordings. The solid line is a spike amplified using the rack-mounted Brownlee amplifier and the dotted line is a spike amplified by the IC. The spikes are taken from different recordings and are not the same signal being amplified.

1/f noise corner. Although these large input capacitors also have a large footprint, they are still not much larger than the pads needed to connect the chip inputs to the external interface. Using smaller capacitors allows for a much smaller die footprint, thus allowing more amplifiers per chip. Unfortunately, small C_2 values expose the amplifiers to gain mismatch and noise. The optimal trade-off between die area and capacitor size is currently being explored. Figure 39 illustrates that the noise profile for the IC and a rack-mounted amplifier are similar. As seen in Figure 39b, the output is still easily distinguishable for an equivalent gain. The noise is comparable and the action potential can easily be sorted using various spike-sorting techniques. The extra gain that was added to compare the signals can easily be added with an on-chip gain stage. Future tests will use another amplifier IC that has an extra gain stage. Although the gain of the on-chip amplifier is more susceptible to mismatch, the footprint is now about 30 percent smaller than the equivalent circuit with $C_1 =$

200 fF and $C_2 = 20$ pF. Here, better matching has been sacrificed for the benefit of increasing the number of amplifiers that can be placed on a single die. Precise gain values are not necessary since the following stage needs information on whether a spike has been produced and not an exact value for the size of that spike. Future versions of this amplifier will compensate for the mismatch caused by smaller capacitors for applications that require better matching.

Amplifiers of varying capacitor size have already been fabricated. By varying the size of the capacitors used, it can be shown that it is unnecessary to have such large capacitors to achieve a satisfactory noise performance. Furthermore, having a second amplification stage allows for smaller capacitors as well. This is important for systems composed of multiple amplifiers. By multiplexing the outputs of multiple amplifiers, precious die area can be saved by having a single second gain stage.

In addition to reducing the capacitor size, other improvements will include adding floating gates to the wide-range amplifier portion [67] and resistive element. The floating gates will provide offset removal in the wide-range amplifier and should increase the operating range of the resistive element.

The following multiplexing schemes were explored: scanned T-gates, addressable and scanned T-gates, and switched buffers (Figure 40). The prototype multiplexer needed to switch among 32 signals to read each signal at a rate of at least 40 kHz. Therefore, the overall switching rate needed to be at least 1.28 MHz.

6.1.2 Discussion

The modified amplifiers successfully amplified voltages akin to both intracellular and extracellular neural signals (Figure 39). Therefore, these amplifiers can be used for single-cell, slice, and in-vivo recordings. The improved amplifier has been useful for our collaborative efforts with Neural Signals, Inc. However, the multiplexer designs have not been as fast as the required MHz scale switching speed. The latest version

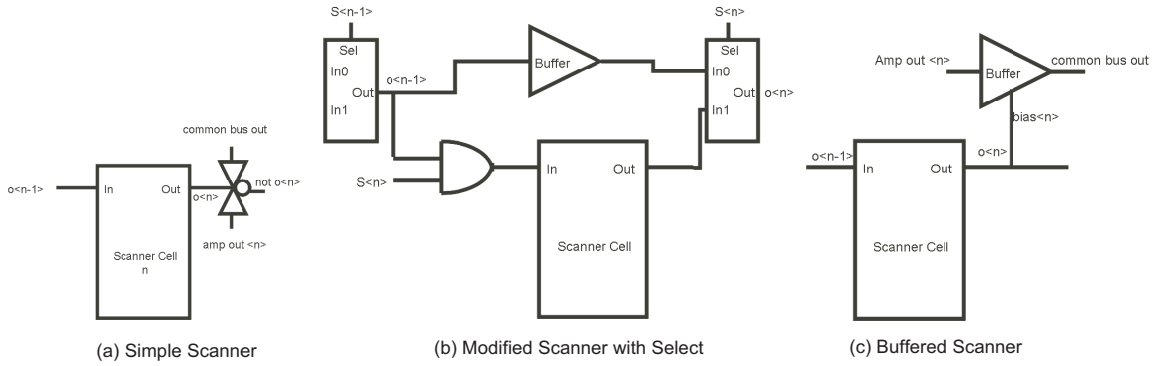


Figure 40: Examples of different multiplexing schemes. Note that each figure is one cell out of an array of cells.

of the multiplexer can switch at over 1 MHz when more power is applied. If the power is increased by 130%, the output can effectively switch at the proper rate. The multiplexer with the option to skip values was deemed unnecessary since programming the control circuitry would not be desired once the chip was implanted to interface with the cortex.

6.1.3 Resistive Element

A major improvement to the above amplifier can be made by reducing the lower range of frequencies that the circuit can operate. The design consideration to use large input pFETs helps with reducing flicker noise of the amplifier. However, the lower frequency corner is also determined by the size of resistances used in the filter.

Most CMOS processes do not include a high resistance layer. If a high resistance layer is available, it is usually on the order of 300 to 1500 ohms/square. Resistance of over 1 MΩ are rarely used in designs due to size limitations. However, these resistances are linear across voltage ranges. This type of resistance cannot be used in this system since resistances of at least GΩ range are needed. Furthermore, each amplifier requires two, separate resistance values (series elements are effectively one element in layout). Therefore, therefore each resistance would be on the about the same size as each 10 pF capacitor. The area of each amplifier would balloon by a

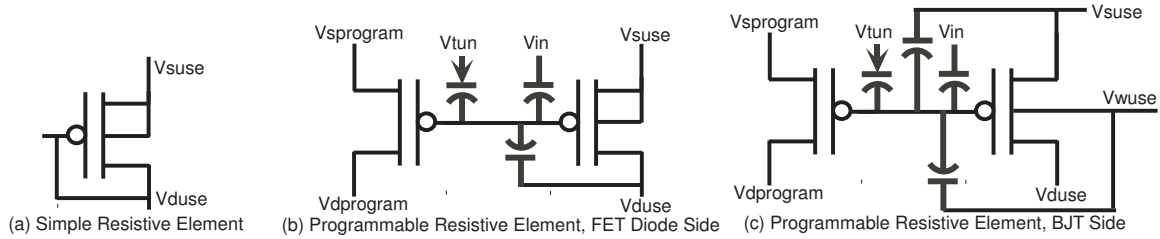


Figure 41: CMOS based resistive elements.

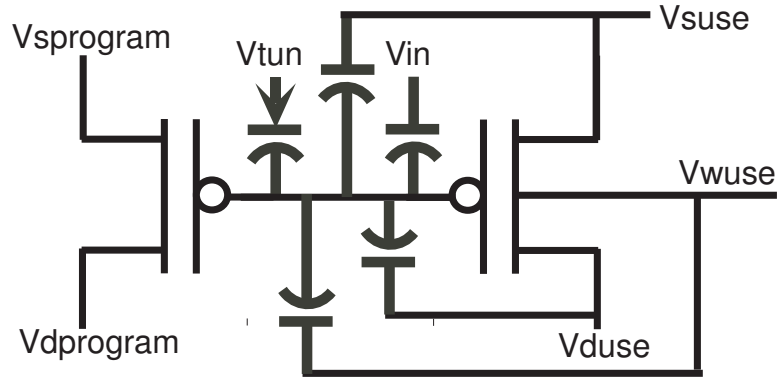


Figure 42: CMOS based resistive element with programmable MOS and BJT ends.

third. This is an unacceptable design tradeoff.

The original resistive element can provide effective resistances of over $1 \text{ G}\Omega$, but the operating range is limited to $\pm 0.2\text{V}$. These devices are an attempt to provide high resistance in a compact area in a CMOS process and across a larger voltage range. These elements would not be realizable without using indirectly programmed floating gates. Resistances of this size are practically impossible using high resistance layers in current CMOS technology.

Four resistive element designs are presented: one is the original design by Delbruck and the other three are my modified designs [68]. These designs are shown in Figures 41 and 42. The elements provide a higher resistance, however, they have not been incorporated into an amplifier design yet. So, the full benefits have not been reaped in the new amplifiers yet.

Note that the experiment was limited by the operating range of the D/A converter used, so the voltages were only measured from -3V to 3V. Furthermore, the picoammeter used could only read currents as low as around 30 pA and the data also reflects that limitation.

The simple resistive element can be considered as two diodes in parallel as seen in Figure 43. The diodes are in opposite directions and only one diode will operate depending upon the voltage across the device. One diode is formed by the connection to the body of the transistor. This is a diode-connected BJT. The other diode is formed by the connection to the gate: a diode-connected MOS. The current when the voltage is negative is mostly due to the element operating like a diode connected BJT. The element behaves like a diode connected MOSFET when the voltage across it is positive. The current-voltage relationship for the simple resistive element is shown in Figure 44. Note that the exponential rate of change of the current is far larger for the BJT side than the MOS side. This is due to the fact that the rate of the BJT operation is proportional to $e^{U_T V}$, while the rate of the MOS operation is proportional to $e^{U_T V/\kappa}$.

Figure 45 shows the absolute value of the current in logarithmic scale. This view clearly illustrates the exponential relationship between the current through the device and the voltage across it. Although this relationship is not linear, for the purposes of decreasing a low-frequency corner, the resistance simply need not drop below a certain value (such as 1 M Ω). Here, the useful operating range is about $\pm 0.2V$.

Some current escapes through the substrate as the device operates as a BJT. This is due to current flowing from the n-well to the p-substrate. This substrate current was calculated by measuring the current source from a voltage source to each of the two accessible nodes. These opposite currents are summed to provided the net difference. This substrate current cannot be measured in this CMOS process since the substrate is shared among the entire chip. The substrate current is large when the

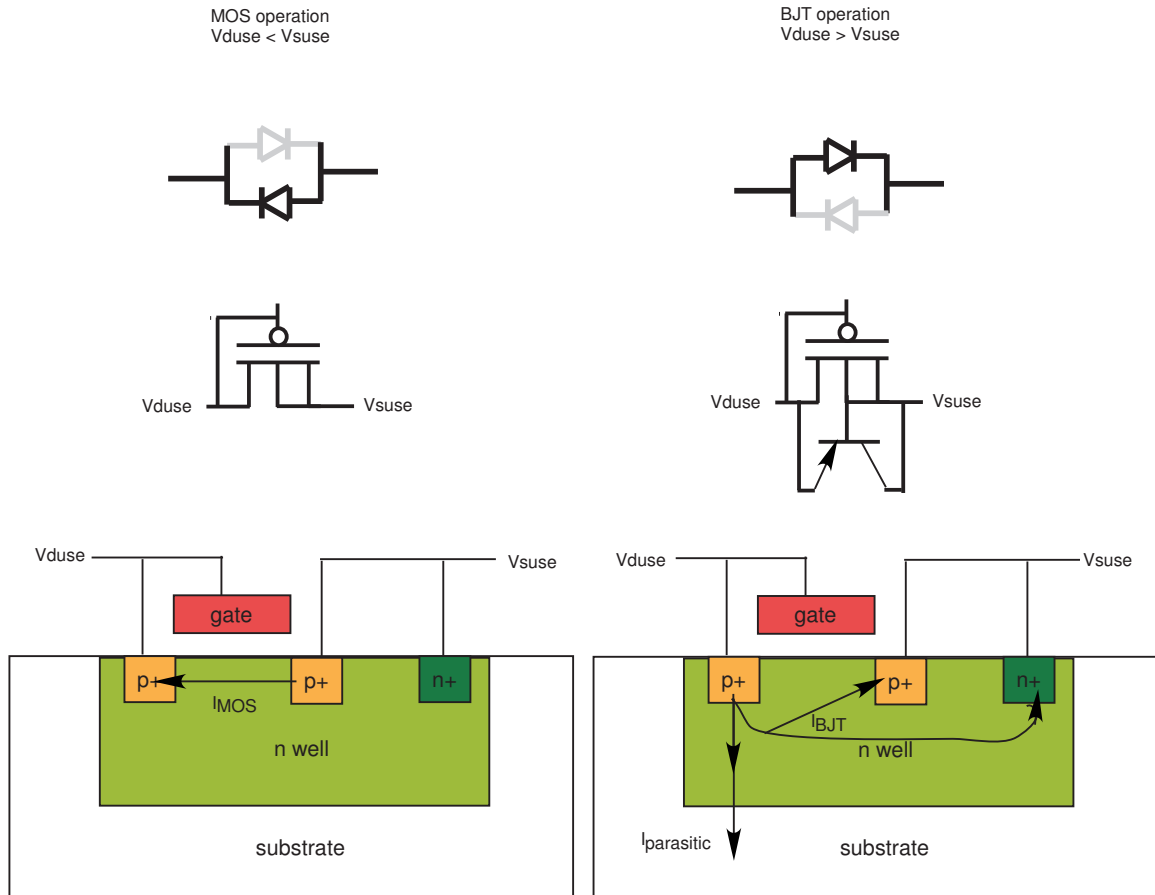


Figure 43: MOS versus BJT operation of the simple CMOS based resistive element.

device is operating as a BJT and negligible during MOS operation. In BJT operation, the total current must include the substrate current, while the total current can be approximated to be the current measured at either terminal. Figure 46 shows this relationship between the currents.

The effective resistance was calculated using Ohm's Law, the given voltage, and the measured current. The effective resistance of the simple resistive element is shown in Figure 47.

The first design is shown in 41b. Here, the MOSFET side of the device can be programmed indirectly. Figure 48 shows the data from this element. Initially, the modified design had a particular floating gate charge. After programming that charge so that the current through the floating gate decreased, the current improved

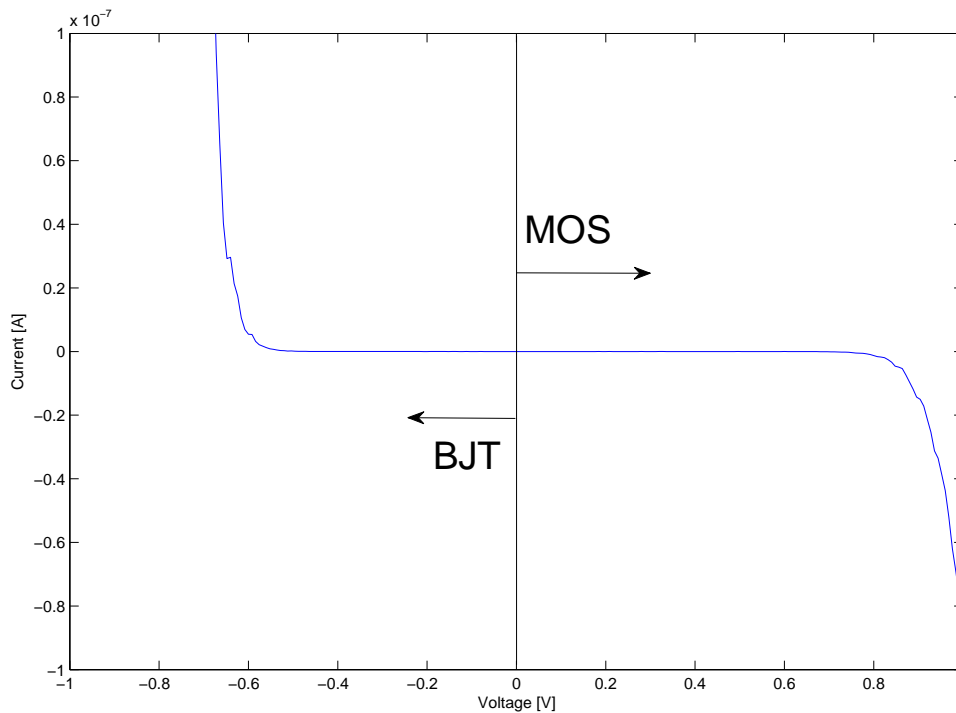


Figure 44: I-V curve for the simple resistive element near the operation point.

to provide an effective resistance in the teraohm range. The programmed current widens the operating range on the MOS current to at least 3 V. Furthermore, this resistance is relatively linear in the MOS operating regime. This modification provides a linear effective resistance from 0 to 3V in a value that is over a million times the largest value that is usually attempted in a CMOS process while being only a fraction of the size.

The second design is shown in Figure 41c. The BJT side of the device has been programmed to expand the operating range. This improved device operates for a range of at least -3V as seen in Figure 49. The maximum resistance dropped, but is still on the order of megaohms from -3V to 0.2V. The MOS operating range is essentially the same as the original design since that part of the circuit is unaffected by the floating gate value.

The third design is a combination of the first and second (Figures 41b and c). Figure 50 shows that this improved device operates for a range of at least -3V to 1.25V. The maximum resistance dropped, but is still on the order of megaohms within the expanded operating range. The MOS operating range could be improved by programming that floating gate to a lower current. Unfortunately, at the time that this device was created, indirectly programmed floating-gates were brand new and design techniques had not been developed. To improve the operation this device, the FETs and capacitors need to be sized more precisely.

6.2 Interfacing Experiment

The true test of the electronic synapse is whether or not it can actually act as the interface between two living neurons. As seen in Figure 51, the full system is comprised of living cells, portions of an electrophysiology rig, and ICs. Ideally, the interfaced system would look more like Figure 3. Unfortunately, because the change in impedance introduced by the microelectrodes, the electrophysiology setup must be used as part of the interface. This external equipment also helps maintain the proper amount of charge needed in the system to prevent the destruction of the electrodes and cell death. A system to provide this functionality is described here as well. Therefore, a full system containing all necessary components can be produced in an affordable CMOS process.

6.2.1 Linking An Artificial Neuron to a Living Neuron

Figure 52 illustrates the ability to link an artificial integrate-and-fire neuron with a living neuron. First, a spike train is produced by providing a constant input current to the artificial neuron. This current is provided by a pFET current source, such that the resulting frequency reaches some desired value. Here, we set the frequency to be approximately triple the natural frequency of the neuron. Next, the output of the artificial neuron is fed to the artificial synapse. The shape and size of the synapse

were chosen to approximate similar EPSPs that the neuron would normally receive. Note that the electronic synaptic output is already a biologically accurate shape and size and does not need any further signal conditioning. The output of the electronic synapse then goes through two converters to prevent any possible damage to the testing equipment. The signal is finally sent to the cell intracellularly. Biologically, synapses are electrochemical devices. Instead of applying neurotransmitters to the postsynaptic cell, we inject a current similar to the one created by the biological synapse. The resulting change in membrane voltage is read, digitized, and stored on a computer. Figure 52 shows that we can produce these signals and effectively change the output of the cell.

6.2.2 Linking Living Neurons Through Electronics

A true electronic synapse has been created using circuitry to link two living cells. Figure 53 shows the signal flow of the system. Cell 1, the presynaptic cell, is stimulated to produce action potentials. The output of Cell 1 receives no amplification and remains on the order of tens of millivolts. If amplification is needed, the rack-mounted amplifier can be replaced with a neural amplifier IC. The intracellular voltage is then sent to the integrate-and-fire neuron. Here, the neuron acts as a spike identifier that sends an output spike when an input spike is received. This digital spike is then sent to the artificial synapse, which creates the EPSP. As in the previous section, the signal goes through converters to protect the equipment for these preliminary tests and are not needed for signal conditioning for the cell. Cell 2, the post-synaptic cell, now begins to produce output when it would normally remain at the resting potential.

6.2.3 Effect of Different Synaptic Weights

The expected result is that the larger the synaptic weight, the larger the output of the postsynaptic cell until the action potential threshold is reached. If threshold is reached, then an action potential is produced. Figure 54 illustrates three EPSPs for

different synaptic weights. By taking the logarithm of these waveforms, as shown in Figure 54b, it is clear that the rising and falling rates are exponential and do not change. The rates do not change because the biases V_n and V_p have not changed. Only the charge on the floating gate has changed, thus, only changing the overall weight of the signal by shifting it upward. The tail end of the signal is dominated by instrumentation noise. The slight 60 Hz noise seen on the largest signal is due to the fact that a portion of the setup was unshielded during that experiment. The 60 Hz noise is more apparent in Figure 54 and is due to the fact that the microelectrodes are unshielded. As expected, the smallest EPSP produced the smallest response. The response is so small that threshold is not reached. However, the largest EPSP does not produce the largest signal since both of the larger EPSPs are large enough to elicit an action potential. The discrepancy in size is still within normal parameters.

6.3 Charge Balancing

A major limitation of neural interfacing systems is the amount of time that the electrodes can be coupled with living tissue. The interface time is primarily due to two interactions: corrosion and cell death.

6.3.1 Corrosion

Electrodes corrode due to the presence of ions. This corrosion is effectively a layer of rust around the electrode. Unlike verdigris that improves the appearance of copper over time, the black patina that is formed around a silver electrode degrades the interface and will cause it stop operating. Therefore, the presence of the essential component which causes electrical responses in cells, ions, also undermine the behavior of the equipment measuring those responses.

6.3.2 pH

The presence of ions also changes the pH level of the area around electrode. In small amounts (such as typical cellular behavior), this effect is negligible. However, the external addition or removal of charge can cause the area around the electrode to become more alkaline or acidic. The cells have no way to compensate for the ions that are introduced or removed from the system over time. Therefore, the ionic charge (whether negative or positive) will build up over time. When the pH level exceeds the tolerance of the cells, they will die. Therefore, greatly altering the balance of charge in a neural interfacing system will cell death. Cell death is expected in the laboratory setting, but this effect is life-threatening when experienced in vivo (such as a deep brain stimulator).

6.3.3 System

The simple solution to this problem is to provide a method to balance the charge in the system. Presently, that is done by using a biphasic signal. Stimulation signals must have a positive and negative phase. This solution greatly limits the type of stimulation that is possible. Collaborating with Jingzhen Hu of North Carolina State University, I have implemented a system that can balance the charge of an arbitrary waveform within a current range of over six orders of magnitude. The system allows a series of output signals and in between each signal, the system will discharge the electrode.

The proposed charge balancing system is illustrated in Figure 55. This system allows an arbitrary current waveform in the range of 100pA to $37\mu\text{A}$ to be balanced. The amount of current needed to be discharged is dependent upon the present output at the electrode and how much current has been built up. The rate of discharge can be modified with a scaling factor V_{ref} . The electrode is connected to the output node.

A current comes into the system and is detected by the first block. This current

detector produces a digital pulse to indicate when the current is on and when it is off. The input node also provides a voltage proportional to the current directly to the output block. This on-off semaphore allows the system to know when to allow the signal through and when to discharge.

The detection pulse stream is then converted to a current to indicate how long the current is on. This time is then modified by a scaling factor V_{ref} . The value of V_{ref} helps to adjust the operating range of the current. The scaled time is then integrated. These calculations are needed to operate a charge pump that modifies the discharge current. The discharge is controlled by the present output value, the desired discharge rate, and how much charge has built up. The resulting summation of parameters yields the expression

$$V_{dis} \propto V_{out} - (V_{ref} - V_{ref}dt/T)$$

. The discharge control signal is sent to the discharge block. The discharge block is comprised of standard discharge circuit. This standard circuit is the same as the input stage of the synapse: a modified inverter. The top-most pFET allows the input to pass through. The bottom-most nFET controls the discharge current. The inverter section controls which operation takes place, input or discharge.

A system tailored for providing PSC-like outputs [63] and the general system described above [64] were simulated in Cadence by Jingzhen Hu. The initial simulation results showed that the upper range of operation can be as high as $37\mu\text{A}$. The bottleneck for lower range of 100pA is caused by the leakage current of the FETs in the modified inverter. The range can easily be shifted upwards to allow for the larger stimulation values used in visual, intramuscular, and deep brain stimulation systems. These stimulators often use currents on the order of $1\mu\text{A}$ to 11mA . The lower current range was chosen to operate with my low current synapse circuits. Hu also created a

prototype system in a $0.6\mu\text{m}$ process through MOSIS.

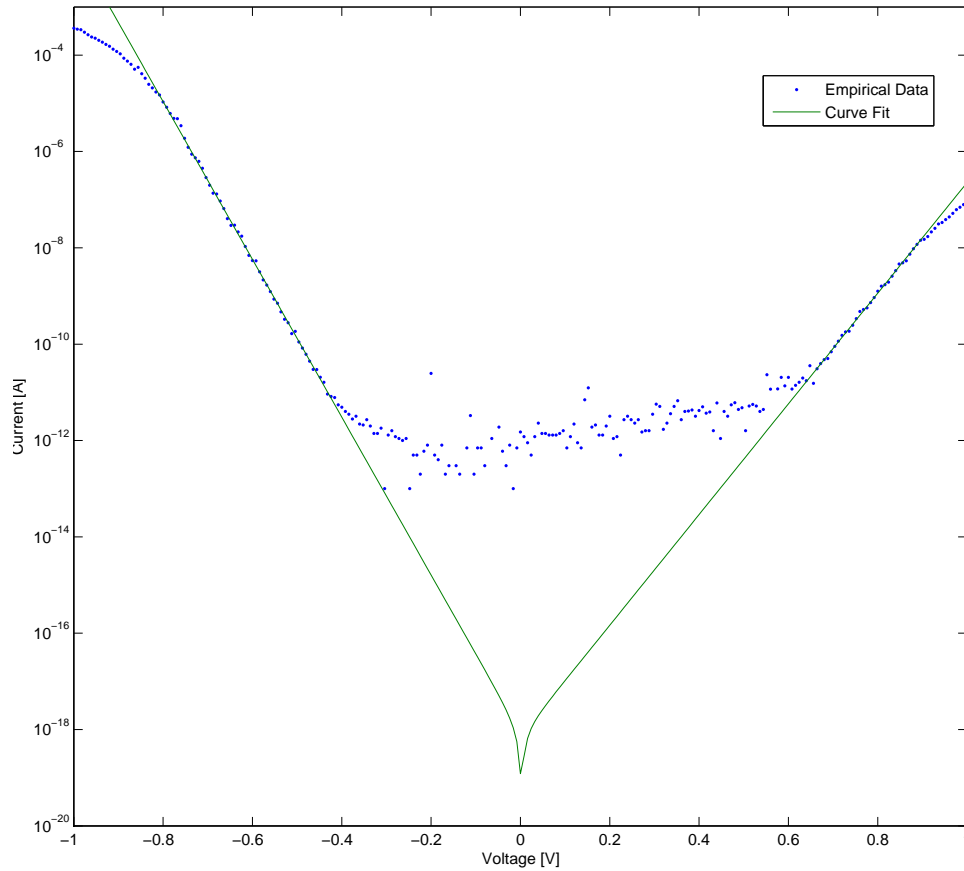


Figure 45: I-V curve for the simple resistive element. Note that this data set is limited to the range of the picoammeter used. The curve fit here illustrates what the resistive element current would be if it were operating in subthreshold and there were no experimental limitations. The normal operation range is limited to $\pm 0.2\text{V}$. Using another resistive elements in series helps to expand the range to $\pm 0.4\text{V}$. The exponential rate of change for the BJT side is calculated as 37.715 while the rate for the MOS side is only -26.455.

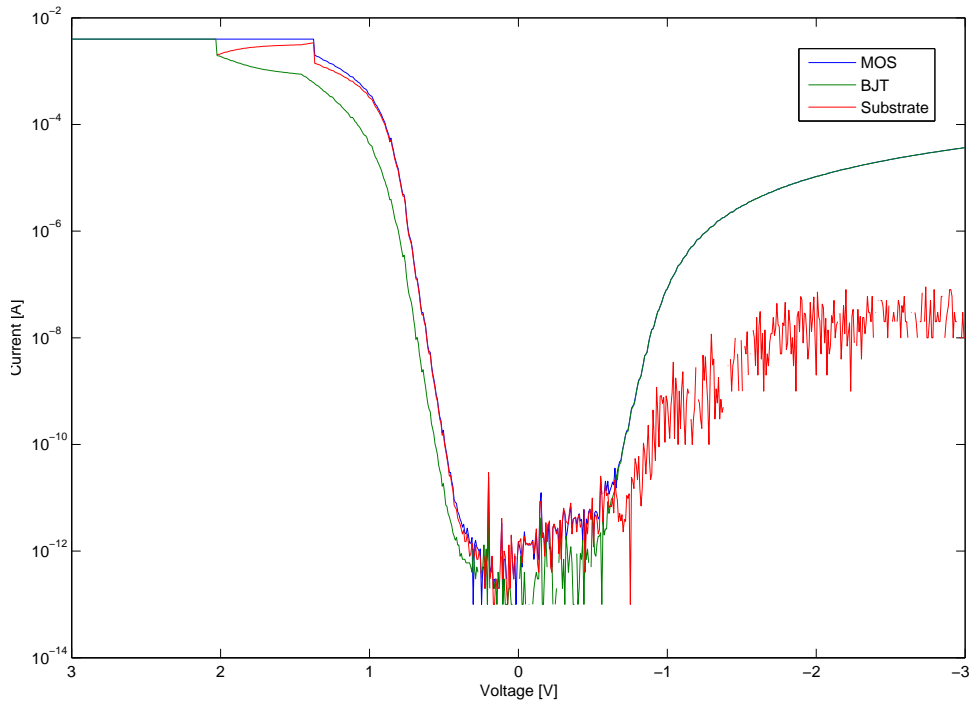


Figure 46: I-V relationship through each node the simple resistive element. Note that the current through the substrate was calculated since there was no way to measure the substrate current in this process. For empirical data, a triple-well process is needed to measure the current due to the resistive element that is flowing through the p-substrate.

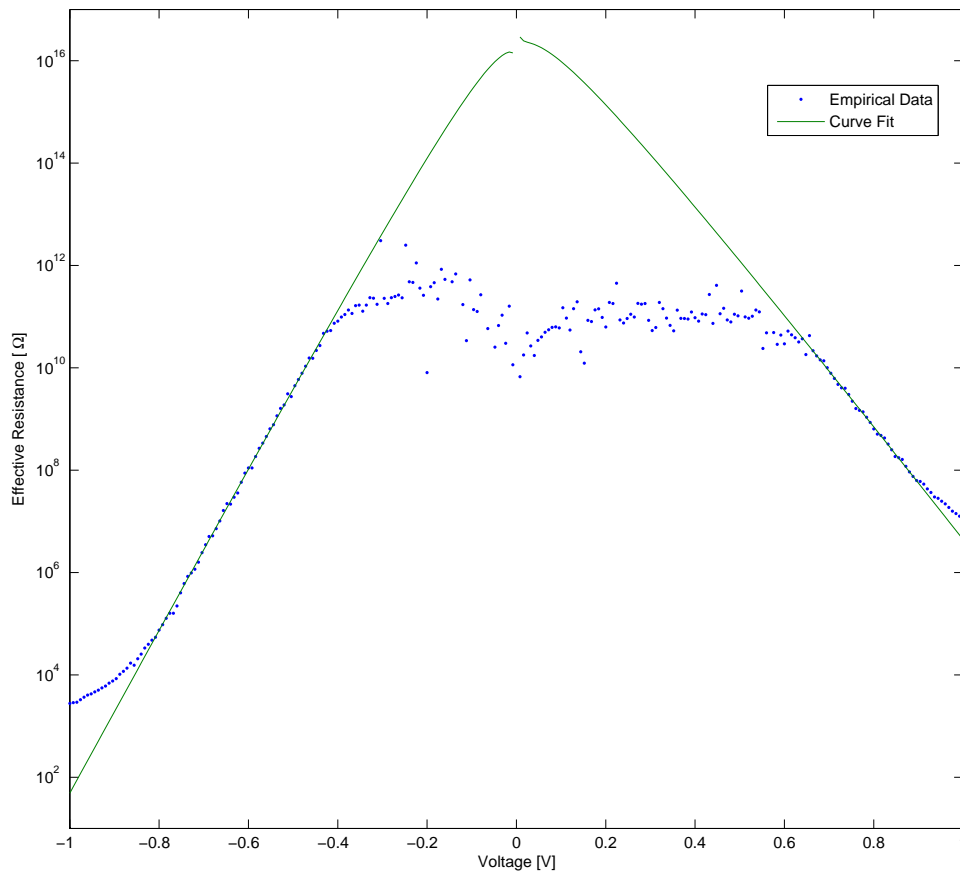


Figure 47: R-V curve for the simple resistive element. The normal operation range is limited to $\pm 0.2\text{V}$. Using another resistive elements in series helps to expand the range to $\pm 0.4\text{V}$. Even with this limited operating range, the effective resistance is in the gigaohm range. The effective resistance was calculated using Ohm's Law on the original data collected from the picoammeter.

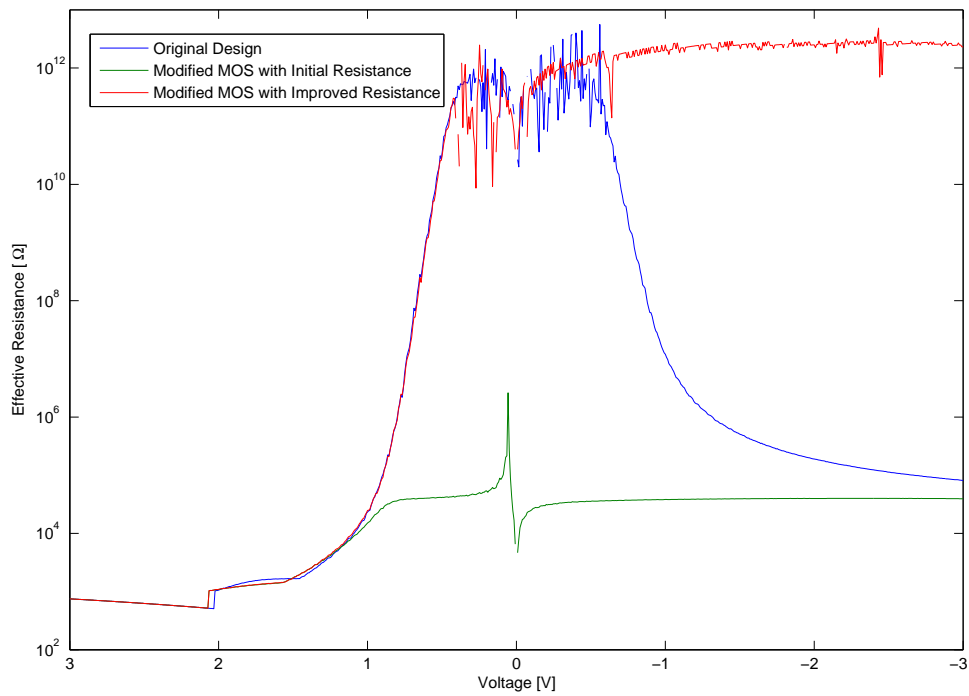


Figure 48: R-V curve for the resistive element with a modification to the MOS side of the device. Note that this data set is limited to the range of the picoammeter used. The maximum that the picoammeter could output is $2\mu\text{A}$, so the values above $2/mu\text{A}$ should be regarded as well above $2\mu\text{A}$. The slight slope is due to using Ohm's Law with a constant current and changing voltage.

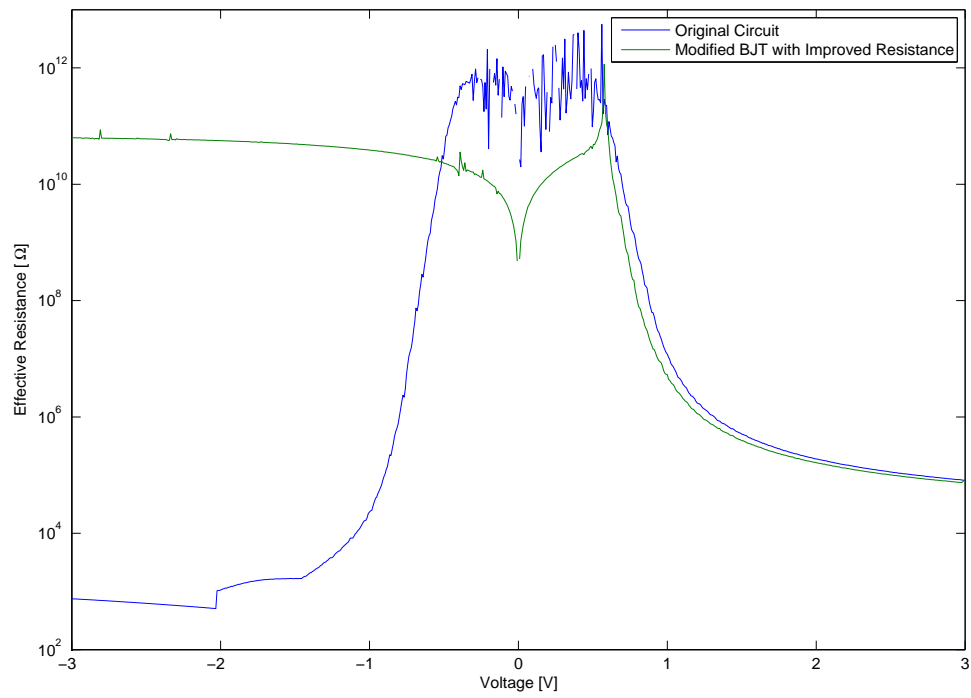


Figure 49: R-V curve for the resistive element with a modification to the BJT side of the device.

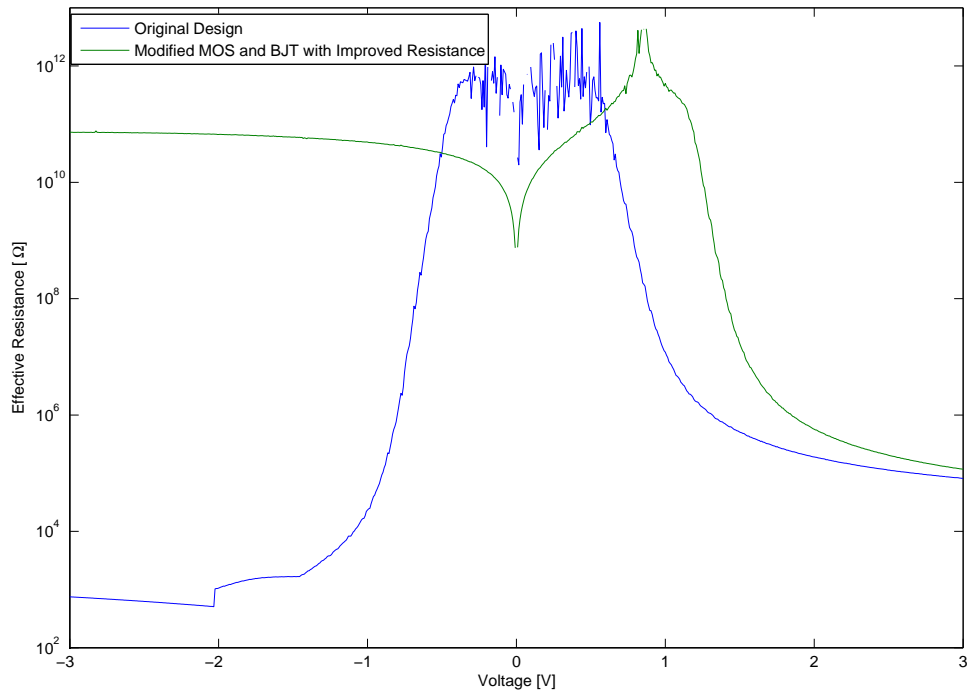


Figure 50: R-V curve for the fully modified resistive element.

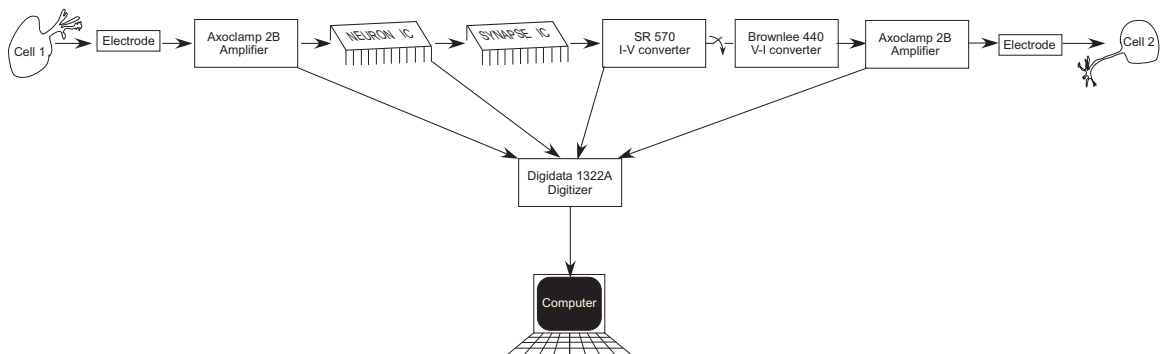


Figure 51: The full setup uses electronics to link two cells that are not normally connected. Cell 1 is the presynaptic cell and Cell 2 is the postsynaptic cell. Ideally, the amplifiers and converters would not be necessary. However, we have initially included them. Note the switch between the I-V converter and the V-I converter that can be used to isolate the circuitry from Cell 2. All of the necessary processing is done by the circuitry. Other test have used an on-chip amplifier to amplify the signal from Cell 1.

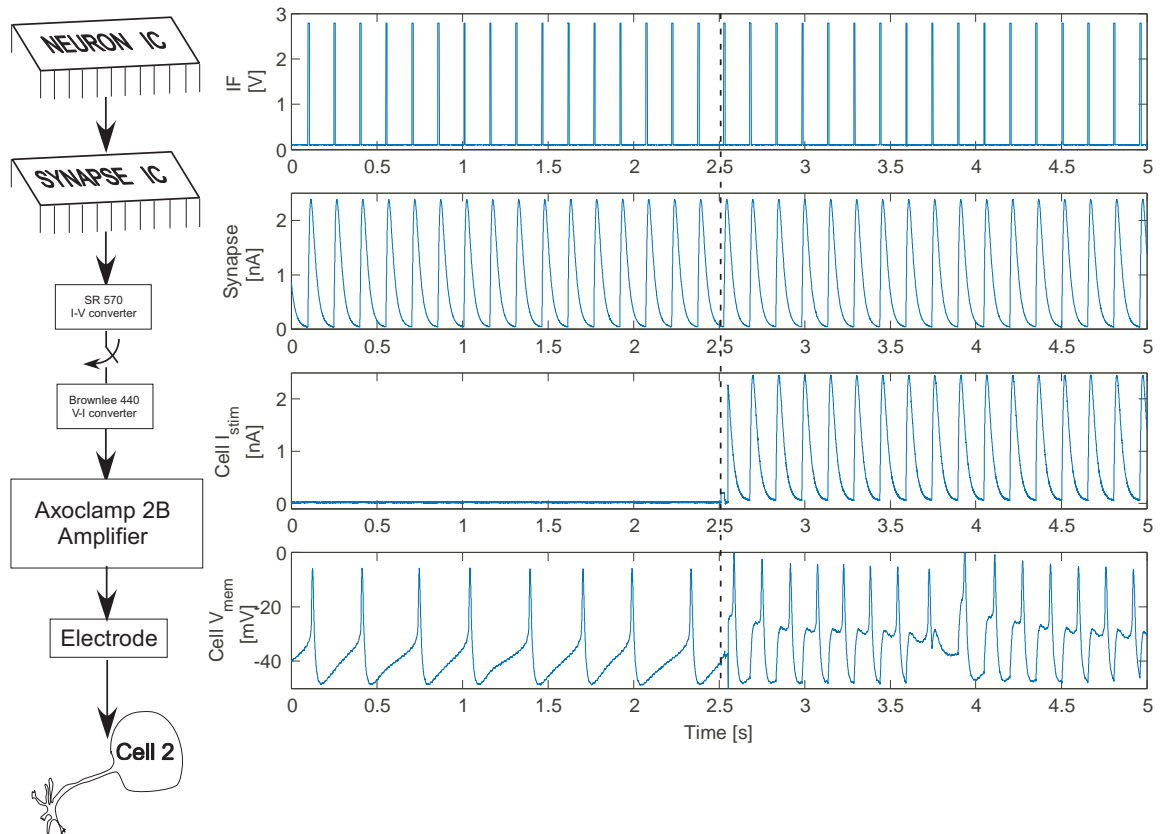


Figure 52: Effect of linking an artificial neuron to a living neuron with an artificial synapse. Note that the switch is disconnected from time 0 to 2.5 s and connected from 2.5 s onward.

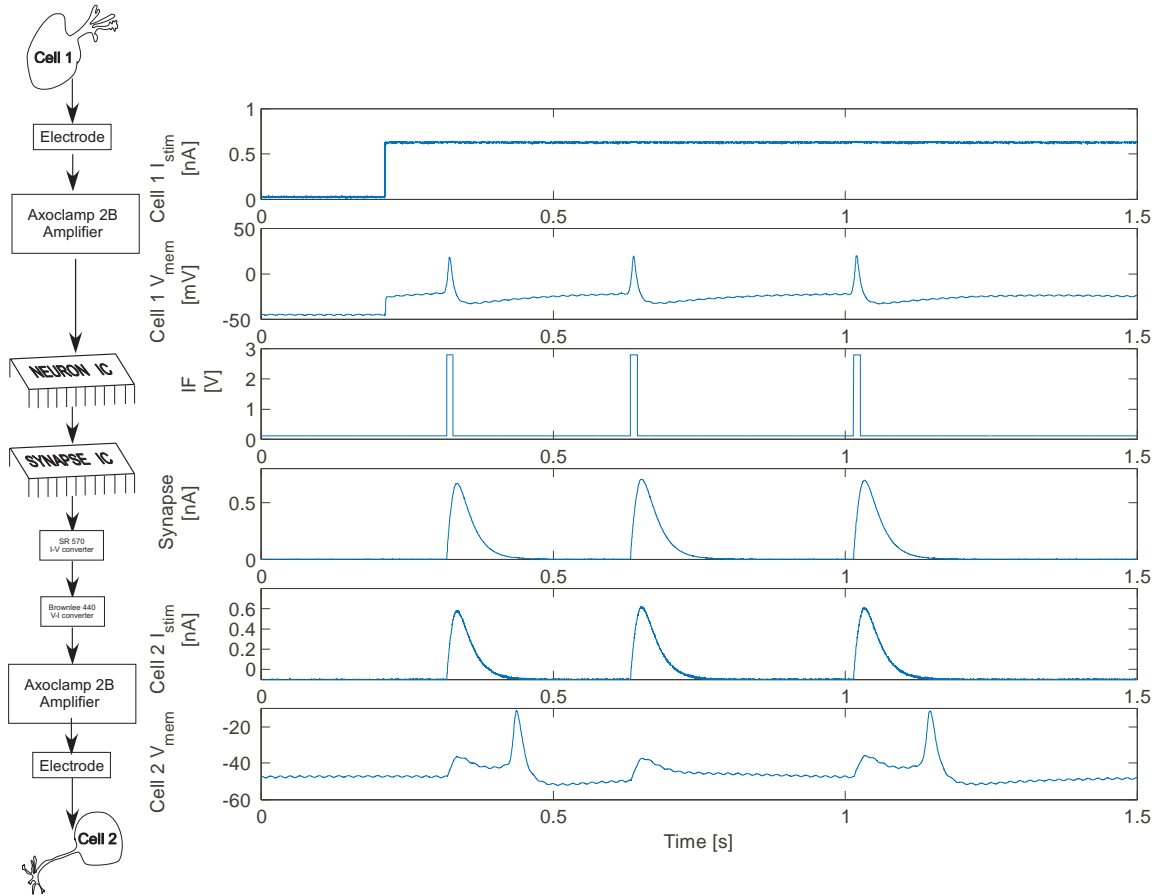


Figure 53: Effect of linking two living neurons through an artificial neuron and an artificial synapse.

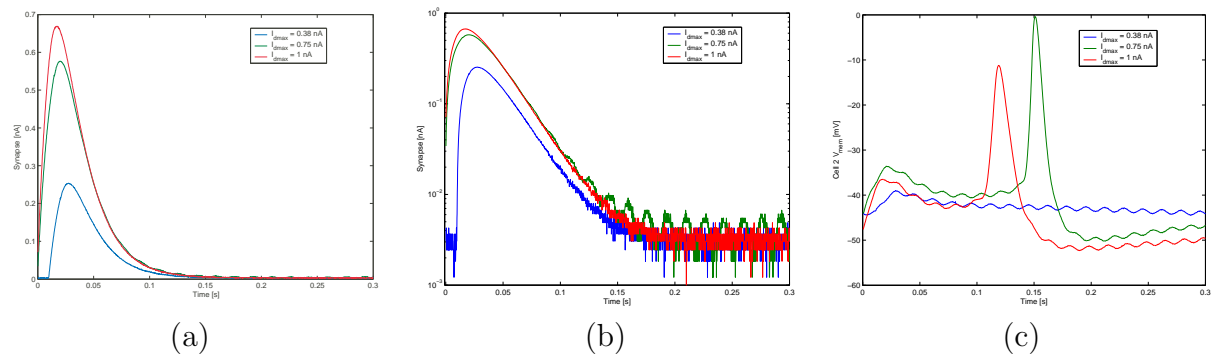


Figure 54: (a) Output of silicon synapse for various synaptic weights. (b) Output of silicon synapse in logarithmic scale. Note that the slopes for the curves are all the same, yet the maximum values differ. (c) Output of Cell 2 (postsynaptic cell) for given synaptic weights. Note that the third and smallest input did not elicit an output response since the input did not reach the threshold of the cell.

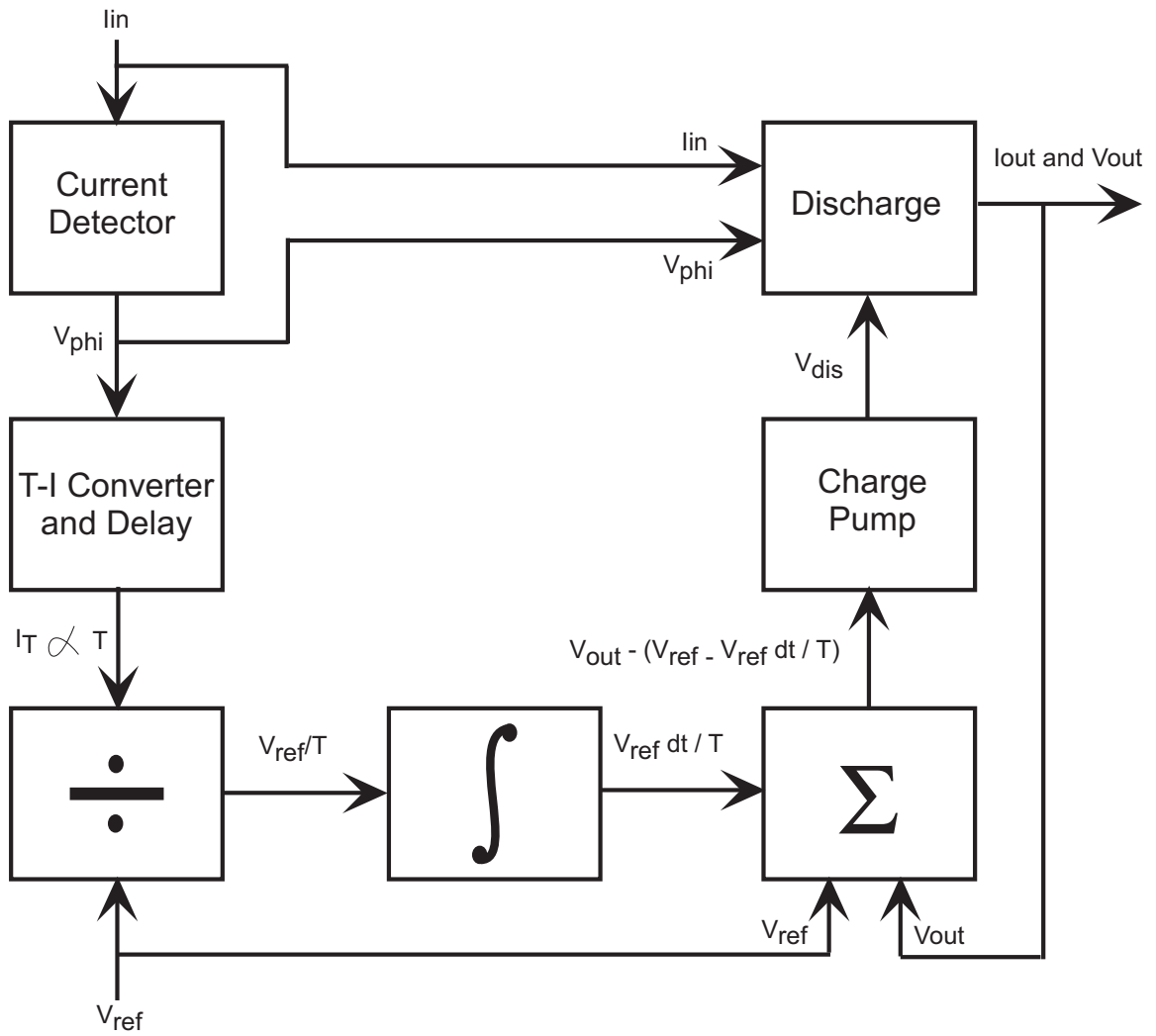


Figure 55: Block diagram of charge balancing system.

CHAPTER VII

CONCLUSION

This work outlines a collection of circuits that are used to form a system that has biologically-plausible learning rules, can interface with neural tissue, and can operate in parallel. This collection forms a toolbox that allows for the development of advanced neural interfaces. These neural interfaces fall into two camps, intracellular and extracellular. Intracellular interfaces can be used for tools used by neuroscientists such as dynamic clamp setups. I have performed an intracellular connection from an artificial neuron to a biological neuron and between two biological neurons. Extracellular interfaces are useful for slice preparations in the lab and neural prosthetics. The current version of the neural amplifier is well suited for reading from slices. However, in vivo readings are difficult since the input electrodes may move. Such movements introduce an unacceptable level of noise and are currently a problem for existing neural prosthetics. Cortical prosthetics are often affixed to the skull to minimize electrode migration. Improvements to the physical structure of electrodes and electrode arrays are needed and will not be addressed here.

I have successfully created a family of biologically plausible synapse circuits that produce electrical outputs and implement learning rules. Support circuitry for these synapses has provided a framework for successful experiments proving that I can perform intracellular recordings, can read signals on the order of extracellular neural outputs, and perform basic analysis on these neural signals. Furthermore, this circuitry has shown that it is plausible to have a system on one chip that can read neural signals, analyze these signals in a bio-mimetic fashion, and produce outputs that are electrically equal to biological signals.

The continuing goals of this work strive toward implementing tools for neuroscientists and engineers. Although this work has provided important advances in our ability to re-create biological systems, more work can be done to make these systems easier to use, more efficient, and a better fit to biology.

7.1 *Learning*

Scientists, mathematicians, and engineers have made great strides in our understanding of learning over the past 50 years. The work of pioneers as disparate as Minsky and Kandel have helped us explain or reproduce some of what is found in biology. However, before this work, no tool had been created that could produce individually plastic synapses with electrically compatible signals that can produce a variety of learning rules in a core area that is less than $200\mu^2$. Improvements upon our biologically feasible learning rules include:

1. porting our designs to a smaller process to more easily implement high voltage amplifiers and to use lower voltages (can be done by a M.S. student),
2. using multiple tunneling fingers to implement multiple tunneling rates,
3. incorporating feedback for multiple synapses.

All of these improvements can be accomplished by an M.S. student in a special topics course. The first improvement may be able to be completed by an advanced undergraduate student.

7.2 *Interfacing*

The toolbox that I created is flexible enough to use in many intracellular and extracellular applications. To maximize potential funding opportunities, I have decided to explore both avenues. The following improvements can be made with the interfacing circuits:

1. porting our designs to a smaller process to increase the density of components,
2. perform extracellular readings with our neural amplifier,
3. analyze extracellular readings with our hys diff circuit,
4. design, fabricate, and test an on-chip voltage clamp system,
5. design, fabricate, and test charge-balancing circuits.

An advanced undergraduate can perform the first task. The rest of the work can be accomplished by an M.S. student. Items 4 and 5 would be done as part of an M.S. thesis while items 2 and 3 could be done as a special topics project.

7.3 Processing

Although the initial version of the FPNA had some programming problems, improvements for another system can easily be made. Since the fabrication of initial version by Farquhar and I, we now have a better understanding of how to effectively use indirect programming and to make large networks. The level of expertise has grown due to the implementation of several FPAA chips, including FPAA with biological elements such as our synapses. Improvements for our neural processor are:

1. implementing a new version of the FPNA with the improved indirectly programmable components,
2. using a traditional switch network rather than floating-gate switches for faster operation,
3. implementing a feedback system that can change connections on the fly using the switch network.
4. implementing a hybrid FPNA-FPGA analog/digital system that can take advantage of the biomimetic properties of the FPNA and traditional computing properties of FPGAs

The first two improvements can easily be done by an M.S. student taking a special topics course. Thoroughly exploring the biologically plausible and computationally interesting aspects of the third improvement would require the time of a Ph.D. student. Initial steps toward the fourth improvement have already been made with a student at North Carolina State University, Gary Charles [69].

Ting Zhu, a student at North Carolina State University and I have begun work on the analysis of rebound signals. Rebounds are a specific type of neural signal that has been found in the cerebellum and retina [70], [20]. We have created a method that can eventually automatically analyze these signals and produce database entries that describe the characteristics of these signals. Even though the processing of these signals are currently done in Matlab, it can be implemented on chip. Therefore, the following improvements can be made:

1. design, fabricate, and test a real-time hardware gaussian classifier,
2. design, fabricate, and test a method of converting the classifier information to a digital form in real-time,
3. test the full system with intracellular recordings,
4. use the system to predict the behavior of cells that produce rebounds.

The first task can be accomplished as part of an M.S. thesis. The second item can be done as a semester-long project by an M.S. student if the first item is done. The rest can be done as part of a Ph.D. project.

7.4 Final Synopsis

I created a toolkit that can be used to bridge the link between artificial and natural neural systems. This toolkit includes the following components:

1. a family of artificial synapses that

- (a) is inspired by biological synapses,
 - (b) interfaces with living systems, and
 - (c) implements a form of learning
2. a compact, low-power, low-noise neural amplifier
 3. a dense, wide-range, high resistance element

Several Masters and Doctoral works can easily branch from the work described here. Future work that stems from this can be used to create fully integrated, scalable, low-power systems for the fields of engineering and neuroscience.

VITA

Christal Gordon

TSRB, Georgia Tech

85 Fifth Street, NW

Atlanta, GA 30308

RESEARCH EXPERIENCE

North Carolina State University Raleigh, NC Aug 05 – Dec 08

Lecturer Electrical and Computer Engineering

Projects included the design and fabrication of bio-inspired circuits for neural modeling and implants. Targeted users of these systems were engineers and neuroscientists.

Data Converters Advanced Graduate Developed from scratch

Students designed and fabricated sigma – delta and pipelined **A/D converters**

Bioinstrumentation Circuits Advanced Graduate Retooled

Students designed and fabricated various circuits for **interfacing to biological systems**

Intro to Bioinformation Senior Developed from scratch

Students used Matlab to design software to **analyze signals in a biomimetic fashion**

Neural Signals Atlanta, GA May 04 - Present

Contractor Movement Restoration Project

The movement restoration project seeks to free patients who have functioning brain activity, but are unable to move their body due to a spinal cord injury or disease. The project involves reading in neural signals which is then sent wirelessly to the next stage of the system. Design **IC amplifiers for recording brain signals**. Develop **multiplexing circuitry for amplifiers**. Test **integrated system**. Improve architecture to allow for **on-chip processing**.

Georgia Institute of Technology Atlanta, GA Aug 99 – present

Research Assistant Integrated Computational Electronics Lab /
Neuroengineering Lab

Fabricate biologically inspired circuits that solve engineering problems. **Design and test novel analog VLSI designs**. **Develop** innovative **architecture and systems**. **Model biological systems** in hardware. Use software such as **Cadence, Quartus**, and various **SPICE** packages for design. Use **Matlab, VHDL, FPGAs**, various **DAQ** boards, and custom **PCBs** in order to interface systems. **Train and advise** several undergraduates, graduate students, and high school teachers.

Motorola Labs Schaumburg, IL May 00 - Aug 00, May 01 - Aug 01

Engineering Intern Integrated Circuit Design Lab

The **smart-card** project involved testing existing **analog front-end wireless receiver designs**. This project was eventually sold to Atmel. The **digital amplifier** project involved designing a **power amplifier for audio applications**. Both projects involved **modeling** real-world effects in

software and designing new, more **cost-effective** analog circuits.

MIT Lincoln Laboratories Lexington, MA Jun 97 - Aug 97, Jun
98 - Aug 98

Technical Assistant High Speed Electronics Group

Tested and analyzed **photonic crystals**. Developed a **real-time electromechanical system** for **data retrieval** and **analysis**. **Designed** and **built** system using a 15 W CO2 laser, optics equipment, a sensor, a computer, and a custom computer-controlled mechanism that moved the sensor. Used Visual Basic to create software to control the entire system.

TEACHING EXPERIENCE

Rochester Institute of Technology Rochester, NY May 04 -
Aug 04

Adjunct Faculty Computer Engineering Department

Developed new **biologically based neural networks** course. The project based course involved students implementing traditional neural networks and biological processing in **Matlab**. Thirty percent of the covered traditional neural networks and seventy percent of the course was on **neuroscience** topics. Projects used traditional and **bio-inspired algorithms** and included **optimizing image file size using PCA** and **speaker ID**.

Georgia Institute of Technology Atlanta, GA Aug 04 - Present
Instructor Electrical and Computer Engineering Department

Team teach **traditional** and **neuromorphic analog VLSI** circuits. **Lecture** on diverse topics including amplifiers, photodetectors, and bio-mimetic circuits. Co-manage **laboratory**. **Created integrated chip** used in lab exercises. Develop and grade quizzes and lab assignments.

Tri-Cities HS / Georgia Tech Atlanta, GA Aug 02 - May 03
Teacher / STEP Fellow Science Department

Responsibilities involved **classroom** and **laboratory** instruction of **Biology** and **Physical Science** to **at-risk** students and **Earth Science** to **at-level** students. **Developed teaching aids** for classroom and stand-alone use. Developed new and **expanded** existing **curriculum**. **Created** a **chapter** of the high school level of the National Society of Black Engineers (**NSBE Jr.**) in order to **mentor** and **nurture** students through engineering activities.

Polytechnic University Brooklyn, NY Sep 98 - Jun 99
Teaching Assistant Electrical Engineering Department

Taught **recitation** for the first two electrical engineering courses (**DC and AC circuits**). **Tutored** students and **graded** homework assignments.

INDUSTRY EXPERIENCE

Ford Motor Company Dearborn, MI Jun 99 - Aug 99

Intern Product Development, Ranger Electric Vehicle

Created and maintained web based information hub in order to maintain information on all of Ford's Ranger Electric Vehicles. **Visual Basic** for Applications (VBA) was used to develop this custom analysis software. This hub was used to **analyze financial** and **logistical data**.

Cox and Company New York, NY Jun 96 - Sep 96

Intern MIS Department

Maintained computer network for an aerospace engineering firm. **Repaired computer systems**, installed software, and provided **technical support**.

EDUCATION

Georgia Institute of Technology Atlanta, GA

Ph.D. Electrical Engineering Defense date: May 09

Georgia Institute of Technology Atlanta, GA

M.S. Electrical Engineering Graduation date: Dec 00

Polytechnic University Brooklyn, NY

B.S. Dual Electrical and Computer Engineering Graduation date: Jun

99

PUBLICATIONS

1. Charles, G.; **Gordon, C.**; Alexander, W.; “An Implementation of a Biological Neural Model using Analog-Digital Integrated Circuits”; BMAS 2008. IEEE International Behavioral Modeling and Simulation Conference, 2008.
1. Hu, J.; **Gordon, C.**; “A General Adaptive Charge-Balancing Stimulator”; MWSCAS 2008. Annual International Conference of the IEEE Engineering in Medicine and Biology Society, 2008. *The 2008 Midwest Symposium on Circuits and Systems* 2008.
1. Hu, J.; **Gordon, C.**; “A Wide Range Charge-Balancing Circuit using Floating-Gate Transistors”; EMBC 2007. Annual International Conference of the IEEE Engineering in Medicine and Biology Society, 2007.
1. D. Graham, E. Farquhar, B. Degnan, **C. Gordon**, and P. Hasler, ”Indirect programming of floating-gate transistors,” *IEEE Transaction on Circuits and Systems II*, In Press.
1. Hasler, P.; Farquhar, E.; **Gordon, C.**; “Building Large Networks of Biological Neurons”; EMBC 2006. Annual International Conference of the IEEE Engineering in Medicine and Biology Society, 2006.
1. Farquhar, E.; **Gordon, C.**; Hasler, P.; “A Field Programmable Neuron Array”; Accepted in Invited Session of *ISCAS 2006. IEEE International Symposium on Circuits and Systems, 2006.*
1. **Gordon, C.**; Preyer, A.; Babalola, K.; Butera, R.; Hasler, P.: “An Artificial Synapse for Interfacing to Biological Neurons”; *ISCAS 2006. IEEE International Symposium on Circuits and Systems, 2006.*

1. **Gordon, C.**; McCoy, M.; Reid, P.; Taylor, K.: “Reaching Back to Look Ahead”; *2005 ASEE Southeast Section Conference*.
1. Graham, D.; Farquhar, E.; Degnan, B.; **Gordon, C.**; Hasler, P.: “Indirect Programming of Floating-Gate Transistors”; *Circuits and Systems, 2005. ISCAS 2005. IEEE International Symposium on*.
1. **Gordon, C.**; Farquhar, E. Hasler, P.; “A Family of Floating-Gate Adapting Synapses Based Upon Transistor Channel Models”; *Circuits and Systems, 2004. ISCAS 2004. IEEE International Symposium on*,. 2004.
1. Woessner, D.; Jangha S.; **Gordon, C.**; et al; “Value Added: Integrating NSBE Jr. Chapters Into High School Mathematics and Science Curricula”, *Proceedings of 2003 American Society for Engineering Education Annual Conference & Exposition*; 2003.
1. Hasler, P.; Smith, P.; Duffy, C.; **Gordon, C.**; Dugger, J.; Anderson, D.; “A Floating Gate Vector Quantizer”; *Circuits and Systems 2002. MWSCAS-2002. The 2002 45th Midwest Symposium on, Volume: 1, 2002 Pages: 196-199*. 2002.
1. **Gordon, C.**; Hasler, P.; “Biological learning modeled in an adaptive floating-gate system”; *Circuits and Systems, 2002. ISCAS 2002. IEEE International Symposium on, Volume: 5, 2002 Page(s): V-609 -V-612 vol 5*. 2002.

PATENT

“Systems and methods for programming floating-gate transistors”; #7269046;

Issued Sep 2007.

COURSES TAUGHT

1. Bioinstrumentation Circuits advanced graduate **retooled**

North Carolina State University Raleigh, NC Fall 2007

1. Intro to Bioinformation undergraduate elective **created**

North Carolina State University Raleigh, NC Spring 2007, 2008

1. Data Converters advanced graduate **created**

North Carolina State University Raleigh, NC Fall 2006, 2007

1. Intro to Electronics required undergraduate

North Carolina State University Raleigh, NC Spring 2006

1. Neuromorphic Analog VLSI beginning graduate

Georgia Institute of Technology Atlanta, GA Fall 2004

1. Neural Networks grad elective **created**

Rochester Institute of Technology Rochester, NY Summer 2004

1. "CMOS Circuits for Neuronal Interfacing"

Boston University Boston, MA Apr 2005

Purdue University West Lafayette, IN Apr 2005

North Carolina State University Raleigh, NC Apr 2005

Rochester Institute of Technology Rochester, NY Mar 2005

1. "Reaching Back to Look Ahead"

ASEE-SE

Chattanooga, TN Apr 2005

1. "A Family of Floating-Gate Adapting Synapses Based Upon Transistor Channel Models"

ISCAS

Vancouver, BC May 2004

1. "Value Added: Integrating NSBE Jr. Chapters Into High School Mathematics and Science Curricula"

ASEE

Nashville, TN Jun 2003

1. "Real-world challenges to teaching low-achieving students"

POD Network in Higher Education

Atlanta, GA

Oct 2002

1. "Biological learning modeled in an adaptive floating-gate system"

ISCAS

Phoenix, AZ

May 2002

1. "Analog Electronic Implementation of Learning"

SENN

Atlanta, GA

Mar 2002

POSTERS

1. "Human Speech Cortex Recordings: IC Development of Implantable Electronics"

(Presenter: Dinal Andresen)

Society for Neuroscience

San Diego, CA Nov 2007

1. "Reconfigurable, Biophysically Based Analog VLSI Models Of Neurons"

Society for Neuroscience

Washington, DC Nov 2005

1. "A biophysically based IC model of biological channels and synapses"

GTAC Atlanta, GA Oct 2003

1. "When Graduate School Isn't Enough - How K-12 Partnerships Enhance Graduate Education"

AAAS Denver, CO Feb 2003

1. "Augmenting the Engineering Pipeline through NSBE Jr."

AAAS Denver, CO Feb 2003

1. "Cooperative Analog/Digital Signal Processing Using Floating-Gate Arrays"

GTAC Atlanta, GA Oct 2000

1. "An Adaptive Floating - Gate Network Using Action Potential Signaling"

GTAC Atlanta, GA Oct 2000

HONORS AND ACTIVITIES

Reviewer for IEEE TCAS, IEEE ISCAS, IEEE MWSCAS, and Neural Computation

Managed undergraduate project which won President's Undergraduate Research Award

NSF STEP Fellowship

GEM PhD Fellowship

AT&T ALFP Finalist

GEM MS Fellowship

Promise Scholarship

REFERENCES

- [1] C. Gordon and P. E. Hasler, “Biological learning modeled in an adaptive floating-gate system,” *Proceedings of the International Symposium on Circuits and Systems*, vol. 5, pp. 609–612, May 2002.
- [2] C. Gordon, E. Farquhar, and P. E. Hasler, “A family of floating-gate adapting synapses based upon transistor channel models,” *Proceedings of the International Symposium on Circuits and Systems*, vol. 1, pp. 317–320, May 2004.
- [3] H. Markram, J. Lubke, M. Frotscher, and B. Sakmann, “Regulation of synaptic efficacy by coincidence of postsynaptic APs and EPSPs,” *Science*, vol. 275, no. 5297, pp. 213–215, 1997.
- [4] L. Abbott and S. Nelson, “Synaptic plasticity: taming the beast,” *Nature Neuroscience*, vol. 3, pp. 1178–1183, 2000.
- [5] M. Tsodyks, “Spike-timing-dependent synaptic plasticity - the long road towards understanding neuronal mechanisms of learning and memory,” *Trends in Neurosciences*, vol. 25, no. 12, pp. 599–600, Dec. 2002.
- [6] S. Song, K. D. Miller, and L. F. Abbott, “Competitive hebbian learning through spike-timing-dependent synaptic plasticity,” *Nat Neurosci*, vol. 3, no. 9, pp. 919–926, 2000.
- [7] C. C. Bell, V. Z. Han, Y. Sugawara, and K. Grant, “Synaptic plasticity in a cerebellum-like structure depends on temporal order,” *Nature*, vol. 387, no. 6630, pp. 278–281, May 1997.

- [8] L. I. Zhang, H. W. Tao, C. E. Holt, W. A. Harris, and M. ming Poo, “A critical window for cooperation and competition among developing retinotectal synapses,” *Nature*, vol. 395, no. 6697, pp. 37–44, 1998.
- [9] D. Debanne, B. H. Gähwiler, and S. M. Thompson, “Long-term synaptic plasticity between pairs of individual CA3 pyramidal cells in rat hippocampal slice cultures,” *The Journal of Physiology*, vol. 507, no. 1, pp. 237–247, 1998.
- [10] M. C. W. van Rossum, G. Q. Bi, and G. G. Turrigiano, “Stable hebbian learning from spike Timing-Dependent plasticity,” *J. Neurosci.*, vol. 20, no. 23, pp. 8812–8821, Dec. 2000.
- [11] N. Caporale and Y. Dan, “Spike TimingDependent plasticity: A hebbian learning rule,” *Annual Review of Neuroscience*, June 2008.
- [12] W. C. Abraham and M. F. Bear, “Metaplasticity: the plasticity of synaptic plasticity,” *Trends in Neurosciences*, vol. 19, no. 4, pp. 126–130, Apr. 1996.
- [13] G. C. Castellani, E. M. Quinlan, L. N. Cooper, and H. Z. Shouval, “A biophysical model of bidirectional synaptic plasticity: Dependence on AMPA and NMDA receptors,” *Proceedings of the National Academy of Sciences of the United States of America*, vol. 98, no. 22, pp. 12 772–12 777, Oct. 2001.
- [14] G. Indiveri and S. Fusi, “Spike-based learning in VLSI networks of integrate-and-fire neurons,” in *Proc. IEEE International Symposium on Circuits and Systems, ISCAS*, vol. 2007, 2007, pp. 3371–3374.
- [15] D. Sridharan, B. Percival, J. Arthur, and K. Boahen, “An in-silico neural model of dynamic routing through neuronal coherence,” in *Advances in Neural Information Processing Systems 20*, J. Platt, D. Koller, Y. Singer, and S. Roweis, Eds. Cambridge, MA: MIT Press, 2008, pp. 1401–1408.

- [16] C. Bartolozzi and G. Indiveri, "Synaptic dynamics in analog VLSI," *Neural Computation*, vol. 19, no. 10, pp. 2581–2603, 2007.
- [17] E. R. Kandel, J. H. Schwartz, and T. M. Jessel, Eds., *Principles of Neural Science*, 4th ed. McGraw-Hill Medical, 2000.
- [18] P. S. Churchland and T. J. Sejnowski, *The Computational Brain*. Cambridge, MA: MIT Press, 1992.
- [19] D. Johnston, S. M. sin Wu, and R. Gray, *Foundations of Cellular Neurophysiology*. MIT Press, 1994.
- [20] G. Shepherd, *The synaptic organization of the brain*. Oxford University Press, USA, 2004.
- [21] M. J. Wall, A. Robert, J. Howe, and M. Usowicz, "The speeding of epsc kinetics during maturation of a central synapse," *Eur. J. Neurosci.*, vol. 15, pp. 785–797, 2002.
- [22] R. Plonsey and R. Barr, *Bioelectricity: a quantitative approach*. Springer, 2000.
- [23] W. Cowan, T. Südhof, and C. Stevens, *Synapses*. Johns Hopkins University Press, 2003.
- [24] B. Hille, *Ion Channels of Excitable Membranes*, 3rd ed. Sinauer Associates, July 2001.
- [25] R. Hedrich and E. Neher, "Cytoplasmic calcium regulates voltage-dependent ion channels in plant vacuoles," *Nature*, 1987.
- [26] H. Yokoshiki, M. Sunagawa, T. Seki, and N. Sperelakis, "ATP-sensitive K⁺ channels in pancreatic, cardiac, and vascular smooth muscle cells," *American Journal of Physiology- Cell Physiology*, vol. 274, no. 1, p. C25, 1998.

- [27] R. F. Pierret, *Semiconductor Device Fundamentals*. Prentice Hall, 1995.
- [28] E. Farquhar and P. Hasler, “A bio-physically inspired silicon neuron,” *IEEE Transactions on Circuits and Systems I*, In Press.
- [29] G.-Q. Bi and M.-M. Poo, “Synaptic modifications in cultured hippocampal neurons: Dependence on spike timing, synaptic strength, and postsynaptic cell type,” *J. Neurosci.*, vol. 18, no. 24, pp. 10 464–10 472, Dec. 1998.
- [30] J. Lisman, “A mechanism for the hebb and the anti-Hebb processes underlying learning and memory,” *Proceedings of the National Academy of Sciences of the United States of America*, vol. 86, no. 23, pp. 9574–9578, Dec. 1989.
- [31] T. V. P. Bliss and G. L. Collingridge, “A synaptic model of memory: long-term potentiation in the hippocampus,” *Nature*, vol. 361, no. 6407, pp. 31–39, 1993.
- [32] L. F. Abbott, J. A. Varela, K. Sen, and S. B. Nelson, “Synaptic depression and cortical gain control,” *Science*, vol. 275, no. 5297, pp. 221–224, 1997.
- [33] J. C. Magee and D. Johnston, “A synaptically controlled, associative signal for hebbian plasticity in hippocampal neurons,” *Science*, vol. 275, no. 5297, pp. 209–213, 1997.
- [34] G. Castellani, E. Quinlan, F. Bersani, L. Cooper, and H. Shouval, “A model of bidirectional synaptic plasticity: From signaling network to channel conductance,” *Learning and Memory*, vol. 12, no. 4, p. 423, 2005.
- [35] J. F. Dickson, “On-chip high-voltage generation in mnos integrated circuits using an improved voltage multiplier technique,” *IEEE Journal of Solid-State Circuits*, vol. 3, pp. 374 – 378, June 1976.
- [36] F. Pan and T. Samaddar, *Charge pump circuit design*. McGraw-Hill Professional, 2006.

- [37] E. Sánchez-Sinencio, A. Andreou, and I. S.-S. C. Society, *Low-voltage/low-power integrated circuits and systems: low-voltage mixed-signal circuits*. IEEE Press, 1999.
- [38] G. Serrano, P. Smith, H. Lo, R. Chawla, T. Hall, C. Twigg, and P. Hasler, “Automated Rapid Programming of Large Arrays of Floating-gate Elements,” in *Proceedings of the International Symposium on Circuits and Systems*, vol. I, May 2004, pp. 373–376.
- [39] D. W. Graham, E. Farquahr, B. Degnan, C. Gordon, and P. E. Hasler, “Indirect programming of floating-gate transistors,” *Proceedings of the International Symposium on Circuits and Systems*, May 2005.
- [40] D. Graham, E. Farquhar, B. Degnan, C. Gordon, and P. Hasler, “Indirect programming of floating-gate transistors,” *Circuits and Systems I: Regular Papers, IEEE Transactions on*, vol. 54, no. 5, pp. 951–963, 2007.
- [41] D. Graham and P. Hasler, “Run-Time Programming of Analog Circuits Using Floating-Gate Transistors,” in *Circuits and Systems, 2007. ISCAS 2007. IEEE International Symposium on*. IEEE, 2007, pp. 3816–3819.
- [42] P. Hasler, C. Diorio, B. A. Minch, and C. A. Mead, *Advances in Neural Information Processing Systems 7*. Cambridge, MA: MIT Press, 1995, ch. Single transistor learning synapses, pp. 817–824.
- [43] R. Plonsky and R. Barr, *Bioelectricity, A Quantitative Approach*. Kluwer Academic Press, 2000.
- [44] C. Mead, *Analog VLSI and Neural Systems*. Boston, MA: Addison Wesley, 1989.

- [45] P. Hasler, M. Kucic, and B. A. Minch, “A transistor-only circuit model of the autozeroing floating-gate amplifier,” in *Midwest Conference on Circuits and Systems*, Las Cruces, NM, 1999.
- [46] S. Liu, J. Kramer, G. Indiveri, T. Delbruck, and R. Douglas, *Analog VLSI: Circuits and Principles*. The MIT Press, Nov. 2002.
- [47] T. S. Hall, C. M. Twigg, P. Hasler, and D. V. Anderson, “Developing large-scale field-programmable analog arrays for rapid prototyping,” *International Journal of Embedded Systems*, 2004.
- [48] E. Farquhar, C. Gordon, and P. Hasler, “A field programmable neural array,” in *Proceedings of the International Symposium on Circuits and Systems*, 2006, pp. 4114–4117.
- [49] P. Hasler, E. Farquhar, and C. Gordon, “Building large networks of biological neurons.” in *Conference proceedings:... Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Conference*, 2006, p. 6548.
- [50] A. Basu and S. Ramakrishnan, “Personal communication,” March 2009, conversation.
- [51] C. Koch, *Biophysics of Computation: Information Processing in Single Neurons*, 1st ed. Oxford University Press, USA, Oct. 2004.
- [52] E. Izhikevich, *Dynamical systems in neuroscience: The geometry of excitability and bursting*. The MIT Press, 2007.
- [53] A. Basu, C. Petre, and P. Hasler, “Bifurcations in a silicon neuron,” in *Circuits and Systems, 2008. ISCAS 2008. IEEE International Symposium on*. IEEE, 2008, pp. 428–431.

- [54] M. Tsodyks, K. Pawelzik, and H. Markram, “Neural networks with dynamic synapses,” *Neural computation*, vol. 10, no. 4, pp. 821–835, 1998.
- [55] T. V. P. Bliss and T. Lømo, “Long-lasting potentiation of synaptic transmission in the dentate area of the anaesthetized rabbit following stimulation of the perforant path,” *J Physiol*, vol. 232, no. 2, pp. 331–356, July 1973.
- [56] H. Scherberger, M. Jarvis, and R. Andersen, “Cortical local field potential encodes movement intentions in the posterior parietal cortex,” *Neuron*, vol. 46, no. 2, pp. 347–354, 2005.
- [57] V. Murthy and E. Fetz, “Synchronization of neurons during local field potential oscillations in sensorimotor cortex of awake monkeys,” *Journal of Neurophysiology*, vol. 76, no. 6, p. 3968, 1996.
- [58] P. Brown and D. Williams, “Basal ganglia local field potential activity: character and functional significance in the human,” *Clinical Neurophysiology*, vol. 116, no. 11, pp. 2510–2519, 2005.
- [59] A. Kuhn, T. Trottenberg, A. Kivi, A. Kupsch, G. Schneider, and P. Brown, “The relationship between local field potential and neuronal discharge in the subthalamic nucleus of patients with Parkinson’s disease,” *Experimental Neurology*, vol. 194, no. 1, pp. 212–220, 2005.
- [60] Y. Izaki, M. Takita, M. Nomura, and T. Akema, “Effects of ventral hippocampal long-term potentiation and depression on the gamma-band local field potential in anesthetized rats,” *Experimental Brain Research*, vol. 157, no. 2, pp. 147–151, 2004.
- [61] R. Harrison and C. Charles, “A low-power low-noise CMOS amplifier for neural recording applications,” *IEEE Journal of Solid-State Circuits*, vol. 38, pp. 958–965, June 2003.

- [62] C. Gordon, A. Preyer, K. Babalola, R. Butera, and P. Hasler, “An artificial synapse for interfacing to biological neurons,” in *Circuits and Systems, 2006. ISCAS 2006. Proceedings. 2006 IEEE International Symposium on*. IEEE, 2006, pp. 4–1126.
- [63] J. Hu and C. Gordon, “A Wide Range Charge-Balancing Circuit using Floating-Gate Transistors,” in *Engineering in Medicine and Biology Society, 2007. EMBS 2007. 29th Annual International Conference of the IEEE*, 2007, pp. 5695–5698.
- [64] —, “A general adaptive charge-balancing stimulator,” in *Circuits and Systems, 2008. MWSCAS 2008. 51st Midwest Symposium on*, 2008, pp. 678–681.
- [65] E. Maynard, C. Nordhausen, and R. Normann, “The Utah intracortical electrode array: A recording structure for potential brain-computer interfaces,” *Electroencephalography and clinical Neurophysiology*, vol. 102, no. 3, pp. 228–239, 1997.
- [66] K. Wise, D. Anderson, J. Hetke, D. Kipke, and K. Najafi, “Wireless implantable microsystems: high-density electronic interfaces to the nervous system,” *Proceedings of the IEEE*, vol. 92, no. 1, pp. 76–97, 2004.
- [67] F. Adil, G. Serrano, and P. Hasler, “Offset removal using floating gate circuits for mixed-signal systems,” in *Southwest Symposium on Mixed-Signal Design*, Feb. 2003, pp. 190–195.
- [68] T. Delbruck and C. Mead, “Analog VLSI phototransduction by continuous-time, adaptive, logarithmic photoreceptor circuits,” *Dept. Computation Neural Syst., California Inst. Technol., Tech. Rep.*, vol. 30, 1993.
- [69] G. Charles, C. Gordon, and W. Alexander, “An Implementation of a Biological Neural Model using Analog-Digital Integrated Circuits,” in *Behavioral Modeling and Simulation Workshop, 2008. BMAS 2008. IEEE International*. IEEE, 2008, pp. 78–83.

- [70] C. Aizenman and D. Linden, “Regulation of the Rebound Depolarization and Spontaneous Firing Patterns of Deep Nuclear Neurons in Slices of Rat Cerebellum,” *Journal of Neurophysiology*, vol. 82, no. 4, pp. 1697–1709, 1999.