Chapter 1 Floating-Gate Learning Systems

This thesis presents a floating-gate technology that we can use to build silicon systems that adapt and learn. **Learning** and **adaptation** have similar meanings and are often used interchangibly. Adaptation is the ability of a system to ignore baseline conditions; **adaptation** often refers to low-level processing. Examples of adaptation are gain control, habituation, and desensitization. **Learning** is the modification of system abilities in response to environmental changes; **learning** often refers to high-level processing. Examples of **learning** are aquiring motor skills, building internal representations of the external world, and so on.

Presently, biological systems are the most complex examples of both **adaptation** and **learning**; therefore, we have much to discover by examing how various engineering problems have been solved adaptively in biological systems. The biological and silicon media are similar; for example, potential barriers are established when the diffusion of charge carriers across a region of space balances the flow of carriers accelerated by the resulting electric field. The information-processing systems embodied in both silicon and biology are physical systems that are affected by the same nonidealities (mismatch, noise, etc.); biological information-processing systems perform astounding computations in spite of these nonidealities. It also seems that neurobiological systems have been optimized both for size and for power dissipation in their computational networks.

The challenge is that we cannot simply duplicate the biological solutions in the silicon media, because the constraints imposed by the biological and silicon media are not identical. It is likely that the differences in the media will lead to different optimal implementations of various information-processing systems. Mahowald's neuron circuit—an implementation of a detailed compartmental model similar to the

Hodgkin–Huxley neuron model—required 27 transistors, occupied $10^5 \mu\mathrm{m}^2$ of circuit area, and required 11 biases [1]. We might find denser implementations, but we probably could not build such dense implementations of realistic neurons in our silicon technology.

In addition to providing us with an efficient information-processing system, building a working system based on biological principles in the silicon medium can give us a deeper understanding of the underlying phenomena. One example that my colleagues and I developed is a silicon axon [2]. This circuit is based on the biological observation that pulse widths do not diffuse away, as they do in a passive RC cable. The axon circuit uses an active mechanism to restore the pulse width; it would be interesting to make a connection to the active mechanism in the biological system.

My goal in my thesis research was to develop an analog floating-gate technology to implement adaptation and learning in silicon. Over the past 10 years, I have been developing analog integrated circuits that adapt and learn [3, 4, 5]. The approach described in this dissertation begins with the constraints that the silicon medium imposes on the learning system; letting the silicon medium constrain the design of a system results in efficient methods of computation. Analog learning systems require a large number of interacting processors; this consideration tightly constrains the size and power dissipation of each processor. This design methodology has proved essential for building analog learning systems.

1.1 Historical Perspective on Analog Implementations of Neural Systems

Analog implementations of neural systems formed a recognizable field of study roughly 10 years ago. Neural networks became an active research topic, due to the pioneering work of Hopfield [6, 7], the PDP group [8], and other researchers [9]. Since that time, use of neural networks has become an accepted numerical technique for solving many nonlinear signal-processing, optimization, and control problems, although the neural network field still has many open questions to address. Expectations for analog implementations were high, and many researchers supposed that these expectations would be fulfilled quickly. Part of the research interest in analog VLSI implementations of neural systems was fueled by digital VLSI becoming a mature field in which few solvable problems remained.

After 6 years of research, several noteworthy neural systems emerged. Biologically inspired neural systems were built in three major categories, mostly as a result of the work of Mead's group and of other related laboratories. Image-processing systems, which were the first active-pixel chips, include silicon retinas [10, 11, 12, 13] optical-flow processors [11], and stereo processors [14]. Auditory-processing chips included many cochlea chips [15, 16], an auditory-localization chip [17], and a chip for binaural hearing [18]. These systems significantly advanced the state-of-the-art, but the majority did not include the additional complexity of adaptation or learning.

Connectionist neural systems are typically built as direct mappings of mathematical and computer models of neural networks into analog silicon hardware. Intel's ETANN chip was the first commercially available neural-network integrated circuit, and the ETANN used floating gates for weight storage [19]. The implementation of the Heuralt–Juetten algorithm by Andreou's group was one of the most successful large-scale adaptive neural systems, but it required a great deal of circuit complexity [20]. Other researchers implemented unsupervised learning and back-propagation algorithms with mixed success [21, 5, 22]. The successful analog implementations of connectionist networks included algorithmic modifications that facilitate its implementation in silicon [23, 5, 24]; history has shown that the success of an implementation is strongly correlated to the degree to which the algorithm is adapted to the silicon medium.

A substantial portion of the early work in neural-network implementations went into developing dense multiplier circuits. Currents are preferred for outputs, because the summation typically required for most connectionist models is easily performed on a single wire, and voltages are preferred for inputs because they are easy to broadcast. Since an input voltage should modulate an output current, most implementations

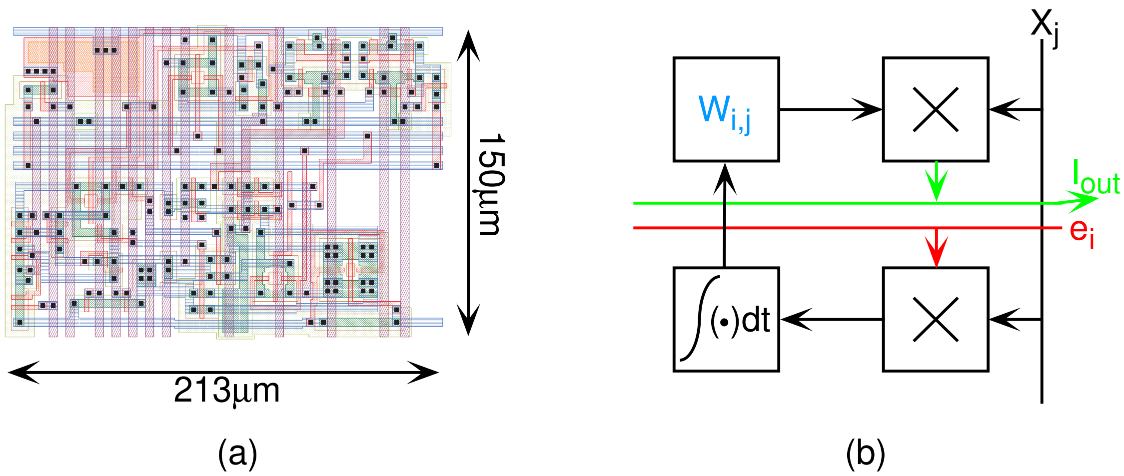

Figure 1.1: In previous neural network implementations, the adaptive elements (synapses) were complex and required a tremendous amount of area. (a) Synapse layout from [5]; the size of this synapse was typical of dense synapse designs of similar complexity. (b) Typical implementations used separate computation, memory, and adaptation blocks, because these networks were direct implementations of mathematical models.

employ a variable resistance or transconductance element. The approaches include synapses based on

- Fixed resistances, which were the earliest implementations [25],
- Switched-capacitor, charge-coupled devices (CCD), and related implementations [26, 3],
- Gilbert multiplier cells [27],
- Linearized conductance elements [28, 29, 30, 31].

Murray [32] and other researchers argued that pulse computation—which can include mean firing rate computation, pulse-width modulation, and related techniques—would make multipliers denser than analog-valued multipliers; they were wrong. Pulse computation may prove important in biologically inspired implementations that employ event-based processing, since biological systems communicate and compute with action potentials that encode events.

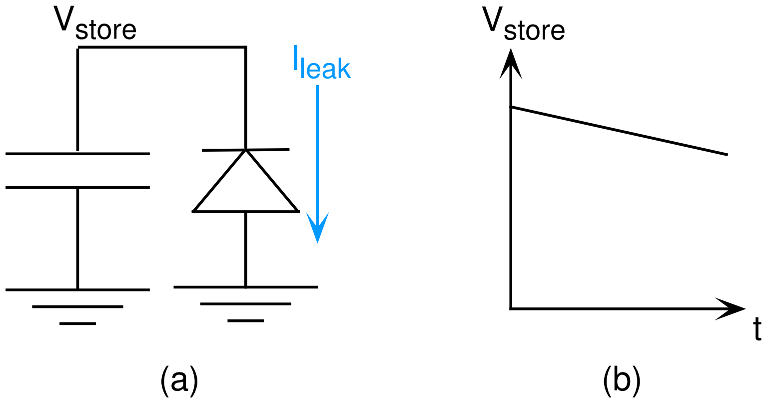

Figure 1.2: Transistor leakage currents limit adaptation time-constants. (a) Circuit schematic when storing a voltage on a capacitor. (b) Plot of the resulting stored voltage versus time. The constant leakage current decreases linearly the stored voltage over time.

Even though many researchers made progress over this 6-year period, the resulting analog neural-network implementations still had many shortcomings. Most networks were direct implementations of mathematical models, and therefore the synapses required large circuit complexity, as illustrated in Fig. 1.1. Not only does large circuit complexity consume tremendous circuit area and power, but also the chance of a network operating correctly decreases exponentially with cell size. One reason that the synapses were large is that the implementations used separate memory, computation, and adaptation blocks. In addition, few successful systems had been built that included adaptation, and even fewer such networks had been built elegantly.

The effect of p-n junction leakage currents, illustrated in Fig. 1.2, was the most difficult problem to overcome to build efficient adaptive circuits. Real-time adaptation and learning requires time constants in the range from 10ms to days; because of junction leakage currents, most integrated-circuit processes are restricted to time constants shorter than 1s unless they use prohibitively large capacitor areas. The adaptation rate must be much smaller than the input-signal rate; therefore, the addition of adaptation to a system constrains the minimum computation rate. A system that has several levels of adaptation requires that the slowest time constant be several orders of magnitude slower than the slowest input-signal rate. If the input-signal frequencies are relatively fast—measured in MHz frequencies—then adaptation can be employed directly in these systems. The adaptive photoreceptor circuit is one of the best examples of adaptive non–floating-gate circuits; this circuit amplifies only fast changes in the input (ms timescales), but does not amplify the background intensities (100ms timescales) [33, 34]. The *Tobi element*, used in the hysteretic differentiator, is the key component that performs the adaptation. Even this circuit shows the time-constant restrictions, and higher-order adaptation is not available.

Four years ago, Brad Minch, Chris Diorio, and I, with Carver Mead, set out to develop a natural silicon technology for learning and adaptation based on the constraints of the silicon medium. We collaborated closely to actualize the possibilities of this new technology. This dissertation is a detailed account of the work that I contributed to this collaboration. In addition, this collaboration produced several related publications [35, 36, 37, 39, 40, 41, 42, 43], as well as the dissertations of my two colleagues. Our collective work provides a good example of synergistic cooperation; none of us would have accomplished nearly as much without the assistance of the others.

1.2 Floating-Gate Technology

To use our medium efficiently, we must find its computational primitives. Digital MOS fabrication processes have been designed and optimized to produce high-quality MOSFET (Metal-Oxide-Semiconductor Field-Effect Transistor) devices with well-controlled gate oxides. The CMOS (Complementary Metal Oxide Semiconductor) process also yields several devices that CMOS process engineers did not intended explicitly; this includes the floating-gate MOSFET. Although floating-gate devices that are fabricated in highly specialized processes have long been used for nonvolatile memories, other potential applications of floating-gate devices have remained largely unexplored. Inspired by the biological systems, we have utilized physical characteristics of the silicon medium that have traditionally posed problems for engineers.

The ETANN chip—one of the few neural networks built with floating-gate technology—

used electron tunneling to program the stored weights from off chip (it had no on-line adaptation) [19]. Concerns about this implementation fueled several arguments against floating-gate implementations. First, most researchers assumed that specialized EEPROM (Electrically Erasable Programmable Read Only Memory) processes, which are not available to most researchers, were necessary for programming the synapses. This concern was partially dispelled by an implementation of interpoly-oxide electron tunneling in the 2μm Orbit process [44]; poor quality of interpoly oxide resulted in poor matching and limited repeatability of tunneling-current behavior. We developed our floating-gate technology from the same standard CMOS processes. The voltages required for electron tunneling (and hot-electron injection) exceed the rated supply voltage, and therefore would be dangerous engineering. Of course, CMOS process designers have considered electron tunneling (and hot-electron injection) to be the source of many digital VLSI reliability problems, and have not seen it as a potentially useful phenomenon. Floating-gate memories were difficult to program with high precision due to complicated programming schemes and nonlinear tunneling characteristics. Floating-gate memories typically require long programming times (when compared with digital clock rates); this slow behavior has been considered by many researchers as a serious limitation of this technology for neural systems. These floating-gate memories, like EEPROM devices, required that the floating gate be driven over a large voltage range to tunnel electrons onto the floating gate; synaptic computation must stop for this type of weight update.

Consider the potential of this floating-gate technology from an adaptive-systems perspective. The reason this technology is well suited for implementing adaptation is that we are not building analog EEPROMS, but rather circuit elements with important time-domain dynamics. In other words, floating-gate devices are not used just for memories anymore. Figure 1.3(a) shows a cross section of a floating gate, which is a polysilicon gate surrounded by SiO_2. Charge on the floating gate is almost permanently stored, because it is completely surrounded by a high-quality insulator. Consequently, once the adaptation is finished, the resulting network state is preserved nearly indefinitely. The floating gate can modulate a channel between a source and drain, as shown in Fig. 1.3(b). Since the floating-gate voltage can modulate a MOSFET's channel current, the floating gate not only serves as a memory, but also can be an integral part of a computation. In subthreshold, the channel current is proportional to the exponential of the floating-gate voltage; therefore, these computations are a direct function of the stored charge on the floating gate. We showed that we can use these floating-gate devices to compute generalized translinear functions by a particular choice of capacitive couplings between the floating gates [38]. The most well-known translinear circuits are multipliers, which date back to the Gilbert multiplier [47].

Fig. 1.3(c) shows electrons moving from the transistor channel to the floating gate by hot-electron injection. The floating-gate charge can be modified by electron tunneling, by hot-electron injection, and by ultra-violet photoinjection. The oxide currents can be arbitrarily smaller than the MOSFET's channel current; typically, the oxide currents are at least 10^4 times smaller than the channel currents. These small current levels gives us *slow* programming rates, which are precisely what we need to build adaptive systems, because the adaptation should integrate over many presentations of the inputs, or over several periods of the input. Continuous-time adaptation occurs because the gate current is a function of the device parameters; therefore, floating-gate adaptation is a direct function of the computations being performed. Hot-electron injection is an important adaptation mechanism, because floating-gate current is proportional to channel current and to an exponential function of drain voltage. In subthreshold MOS transistors, the adaptation time constants due to the gate currents well match the computation time constants due to the transistor channel currents.

A frequently asked question is "What is the resolution of a floating-gate device?" The question implies that we are still talking about a memory element. At a fundamental level, the precision is limited by the number of voltage levels on the floating gate, where the voltage level is determined by the voltage that a single electron generates on the floating-gate capacitor. The limiting factor, as it is in non–floating-gate neural memories [45], is the circuitry that converts the stored representation to the required circuit value. We have shown that it is the analog circuitry, rather than the floating gate, that limits analog resolution in our processes [41].

1.3 Thesis Overview: Electron Transport to Floating-Gate Circuits

This dissertation systematically presents a bottom-up progression from electron transport to floating-gate circuits. Unless otherwise stated, the data presented are experimental results that I measured from integrated circuits fabricated in either the 1.2μm or 2μm, double-poly $n$well Orbit CMOS process available through MOSIS. We will begin in Chapter 2 with my model of hot-electron injection and impact ionization derived from a first principles model of hot-electron transport. This analytical model of hot-electron transport in the depletion region between the channel and drain regions was derived from Boltzman transport. From a physics perspective, this model is important because it explains the underlying phenomena of both impact ionization and hot-electron injection, and complements the current approach of semiconductor modeling of Monte Carlo numerical calculations using full-bandstructure models. From a circuits perspective, we have the first realistic device model of hot-electron injection that we can use to develop, simulate, and verify floating-gate circuits.

Chapter 3 presents the single-transistor learning synapses, which are the fundamental circuit elements for adaptation and learning circuits. The single-transistor synapse, invented and developed by me and my collaborators, Diorio and Minch, comprises a hot-electron–injecting floating-gate transistor and a well-tunneling junction. This chapter describes the basic nFET and pFET synapses; other variations that we have invented will be presented in Diorio's dissertation. These synapses simultaneously store the analog weight, compute a product based on this weight and on the input signal, and adapt this weight, all in the same element. This chapter briefly discusses electron tunneling, a well-understood phenomenon that we use to remove electrons from the floating gate. This chapter also presents a simplier device-level model of hot-electron injection in nFET and pFETs. The single-transistor-learning synapse represents the transition from devices to circuits for this floating-gate technology. The small size and low-power operation of single-transistor synapses permits us to develop dense synaptic arrays. When the steady-state source current is used as the representation of the weight value, both the incrementing and the decrementing functions are proportional to a power of the source current.

Chapter 4 presents the possible dynamics of simple continuous-time floating-gate circuits. Circuits that have one nFET or pFET synapse show a range of stabilizing and destabilizing behaviors, because each circuit imposes a particular type of feedback to the floating gate. By providing feedback to the source, we get another synapse type with unique dynamics, because voltage changes in both the floating gate and drain stabilize the floating gate. Multiple floating-gate circuits show both competitive and cooperative behaviors between synapses, consistent with Hebbian and anti-Hebbian learning. Modeling the dynamics of a floating-gate circuit gives us intuition into developing more complex circuits, and shows the strengths of simple floating-gate current models.

Although the single-transistor synapse can serve as the fundamental adaptive element, and although its feedback properties are well characterized, we still need to demonstrate that an adaptive circuit based on our floating-gate technology could learn continuously. Chapter 5 presents the autozeroing floating-gate amplifier (AFGA), a floating-gate amplifier that adapts its output voltage back to the same steady-state value over a long timescale. The AFGA is the simplest practical form of a floating-gate circuit that adapts. This continuous-time bandpass amplifier uses capacitive feedback and capacitor ratios to set several important circuit properties. The AFGA circuit is the basic building block for a family of continuous-time amplifier and filtering circuits; it provides an attractive low-power alternative to switch-capacitor filters.

Chapter 2 starts by considering my first-principles model of hot-electron injection. The circuit-oriented reader might choose to skim read Chapter 2, and then jump to Chapter 3.

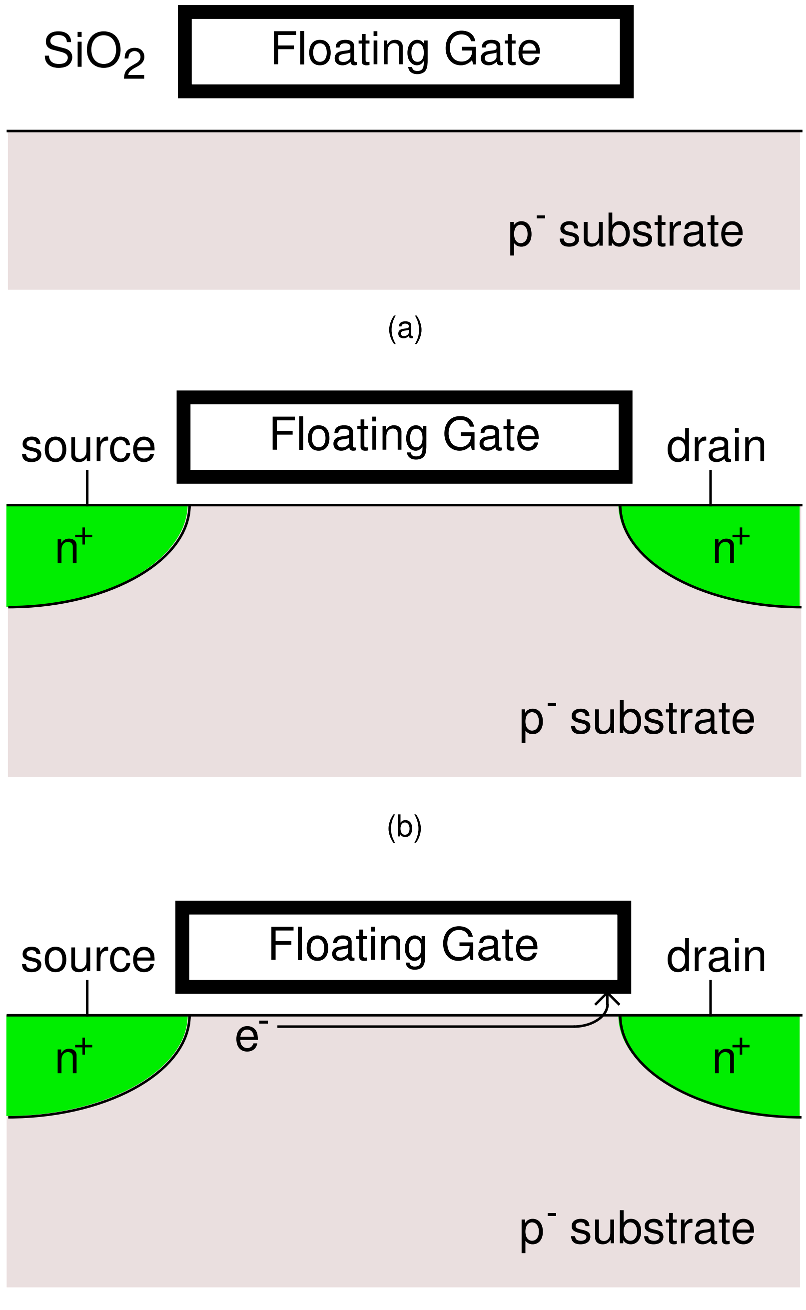

Figure 1.3: Illustration of our floating-gate CMOS technology. (a) Charge on the floating-gate is nearly permanently stored because electrons are surrounded by a high quality insulator. (b) Since the floating-gate voltage can modulate a MOSFET's channel current, the floating gate is not only a memory device, but also can be an integral part of a computation. (c) The floating-gate charge can be modified by electron tunneling and hot-electron injection. Continuous-time adaptation is possible if the floating-gate current is a function of other device parameters. In this respect, hot-electron injection is an important adaptation mechanism, because floating-gate current is proportional to channel current and an exponential function of drain voltage.