PROGRAMMABLE ANALOG TECHNIQUES FOR PRECISION ANALOG CIRCUITS, LOW-POWER SIGNAL PROCESSING AND ON-CHIP LEARNING

A Dissertation
Presented to
The Academic Faculty

By

Venkatesh Srinivasan

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

School of Electrical and Computer Engineering
Georgia Institute of Technology
August 2006

PROGRAMMABLE ANALOG TECHNIQUES FOR PRECISION ANALOG CIRCUITS, LOW-POWER SIGNAL PROCESSING AND ON-CHIP LEARNING

Approved by:

Dr. Paul Hasler, Advisor
Assoc. Professor, School of ECE
Georgia Institute of Technology
Atlanta, GA

Dr. Alan Doolittle
Asst. Professor, School of ECE
Georgia Institute of Technology
Atlanta, GA

Dr. Farrokh Ayazzi
Assoc. Professor, School of ECE
Georgia Institute of Technology
Atlanta, GA

Dr. Mark Smith
Professor, School of ICT
Royal Institute of Technology
Kista, Sweden

Dr. David Anderson
Assoc. Professor, School of ECE
Georgia Institute of Technology
Atlanta, GA

Date Approved: June 12, 2006

DEDICATION

To

Amma, Appa, Viji and Poornima

ACKNOWLEDGEMENTS

I would like to thank Dr. Paul Hasler for his guidance and numerous stimulating conversations during the course of my doctoral studies. Thanks are due to the members of my committee for all the insightful comments that have helped shape this dissertation.

Many thanks to members of ICE lab for making my stay at GaTech a fun-filled experience. Guillermo Serrano requires a special mention for being both a great friend and an excellent work partner. It has been a pleasure working with you! I have had the opportunity to collaborate with many ICE labbies, both past and present: Jeff Dugger, Abhishek Bandyopadyay, Angelo Pereira, Ravi Chawla, Chris Twigg, Jordan Gray, Gail Rosen, David Graham, David Abramson, Ryan Robucci, Thomas and Erhan Ozalevli. It has been wonderful working with you all. I am indebted to Krishnakumar Sundaresan and Srinivasan Seetharaman for accommodating my countless requests for help during my stay here in Atlanta!

This work would not have been possible without the love, support and encouragement of my family. Amma, Appa, Viji and Poornima, I owe it to you all.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

SUMMARY

The transistor is one of the key components that has made possible the plethora of electronic gadgets that one finds in use today. Investigating the possibilities of providing an additional degree of design freedom to this fundamental element is the subject of this research. This is achieved using a floating-gate transistor that provides programmability in circuits and thereby positively impacts a wide range of applications from traditional analog circuits to systems that learn on-chip.

Using a programmable analog framework, precision analog circuits have been developed that are compact and power efficient. Floating-gate transistors form an inherent part of the circuits of interest. Candidate circuits demonstrated include programmable references and low-offset amplifiers. Lack of programmability in analog circuits has been the biggest stumbling block in implementing key signal processing operations such as multiplication and addition in an area and power efficient manner in the analog domain. Using floating-gate transistors, an analog current-mode multiply-accumulate unit has been developed. Experimental results show significant power savings when compared with digital implementations.

Programmable analog sets the stage for on-chip learning and adaptation as well. An analog architecture has been presented that implements an adaptive filter with on-chip learning of the necessary weights such that the error between the system output and a target signal is minimized. The fundamental building block of this system is a floating-gate synapse that modifies the charge stored on its floating-gate using a least-mean-square learning algorithm. A simulation model for the floating-gate synapse has also been developed in order to help design large-scale adaptive filters. In summary, this research involves developing techniques for improving analog circuit performance and in developing power-efficient circuits for signal processing and on-chip learning.

CHAPTER 1
PROGRAMMABLE TECHNIQUES

Electronic devices today are small, smart and flaunt an array of previously unimaginable features. This has been the result of several years of creativity at both the circuit and system levels and has been further enabled by the aggressive scaling of transistors following Moore's Law. However, from a circuit designer's standpoint, such a scaling has given rise to several impediments. Designers now have to deal with mismatch and noise under conditions of reduced signal swings caused by lower power supplies and yet, deliver the high accuracy and low-power demands of consumer electronics. At this juncture, what could the possibilities be if an additional degree of design freedom was available to circuit designers? This research aims to explore this question from the context of a floating-gate transistor as a programmable element in analog circuit design and demonstrates applications beyond traditional analog circuit designs.

A floating-gate transistor is a transistor whose gate is completely surrounded by a high quality oxide and hence has no DC path to ground. This provides a floating-gate transistor with non-volatile charge storage capability. The ability to alter the stored charge leads to programmable analog circuits, the implications of which are enormous. The ability to alter the transistor's characteristic provides an alternative to addressing the effects of mismatch in analog circuits, thereby resulting in precision analog circuits with minimal additional overhead. Programmable analog, apart from solving analog design issues, opens the door to a paradigm shift in signal processing. A co-operative analog-digital signal processing is now possible with computationally intensive tasks such as multiplication being performed with extremely low power dissipation in the analog domain. Achieving programmability coupled with a power efficient analog implementation of signal processing sets the stage for more complicated tasks such as adaptive filtering. This work develops a programmable analog framework using

floating-gate transistors and demonstrates the advantages of such a technique in the above areas by way of example circuits such as amplifiers, references, multipliers and synapse circuits that achieve adaptation and on-chip learning.

1.1 Precision Analog Circuits

Precision analog circuit design has been limited primarily due to mismatch and variations in parameter values in integrated circuit components such as transistors, resistors and capacitors. Mismatch, by way of offsets in amplifiers, limits the available input signal dynamic range. Offsets in comparators place a lower limit on available signal resolution. Continuous-time g_m-C filters require design effort in realizing tuning schemes to correct for variations in both the capacitance and the transconductance. Matching between transistors directly impact the achievable accuracy in current-mode digital-to-analog converters. Also, mismatch is the key issue when designing high-accuracy analog-to-digital converters and precision references.

Manufacturing imperfections and tolerances result in parameter variations. For instance, the random variations in the number of dopant atoms under the gates of identical transistors result in random variations in their device characteristics and such variations are classified as device mismatch [3]. Device mismatch between two geometrically identical transistors has been studied extensively by several researchers, notably [4, 5]. It has been observed experimentally that the threshold voltage difference between the transistors (ΔV_T) and the difference between their transconductance parameter ($\Delta\beta$) are the dominant sources of device mismatch [3]. These result in differences in drain currents between devices for identical bias conditions when used as current mirrors or gate-source voltage difference that result in an offset voltage when used as a differential pair.

These random variations in threshold voltage difference and transconductance parameter difference have been modeled as a gaussian distribution with zero mean

and a variance that is given by [4],

$$\sigma(\Delta V_T) = \frac{A_{VT}}{\sqrt{WL}} \tag{1}$$

and,

$$\frac{\sigma(\Delta\beta)}{\beta} = \frac{A_\beta}{\sqrt{WL}} \tag{2}$$

where, A_{VT} and A_β are process dependent constants. Also, experimental data indicate a low correlation between these two random variations. It can be observed from the above equations that mismatch can be countered by increasing the area of the device resulting in a minimum area requirement for a given accuracy specification. Such an approach increases the parasitic capacitance of the device, thereby, increasing the power dissipation required in order to achieve a given bandwidth. In summary, device mismatch results in a fixed bandwidth-accuracy-power tradeoff that is set by process parameters [3].

From an architecture perspective, offsets in amplifiers have been addressed using schemes such as autozeroing, correlated double sampling and chopper stabilization [6]. In analog-to-digital converters (ADCs), it is common to use digital calibration to correct for errors due to mismatch [7]. Continuous time filters employ elaborate tuning schemes to account for variations in transconductance and capacitance [8]. Current-mode digital-to-analog converters (DACs) use segmented architectures with current sources designed based on the area requirements of (1) and (2) along with complicated switching schemes to further improve accuracy [9].

On the physical layout level, a number of techniques can be used to result in matched devices. These include, the use of dummy devices, common centroid layout techniques to cancel the effects of process gradients, maintaining orientation between devices and avoiding metal coverages [10]. For passive components such as resistors, post-fabrication trimming techniques using laser and poly-fuses are commonly employed to correct mismatch and process induced variations. It can be observed

that be it the device level, physical layout level or the architecture level, mismatch is corrected at the cost of area, power and design complexity.

Programmable analog presents a viable solution to addressing mismatch at the transistor level. The design methodology uses floating-gate transistors as programmable elements that correct mismatch in analog circuitry. Rather than using floating-gate MOSFETs as seperate trimming elements, transistors that are an inherent part of the circuit architecture are designed to be floating-gates such that circuit imperfections due to mismatch can be accounted for through programming. Such an approach is scalable with process, reduces design overhead and results in a compact architecture with minimal extra power dissipation. Also, the non-volatile charge retention of floating-gates obviates the need for constant refresh cycles.

In this dissertation, the technique of using floating-gate transistors as both programmable elements and as an inherent part of the circuit of interest has been applied to demonstrate two high performance analog circuits, namely, offset cancellation in amplifiers and a programmable reference. A single stage folded cascode amplifier has been implemented in a $0.5\mu m$ CMOS process using floating-gate transistors to cancel the offset voltage of the amplifier to $\pm 25\mu V$ with the offset voltage displaying a total variation of $130\mu V$ over a $170^{\circ}C$ temperature range. The programmable reference has been implemented in a $0.35\mu m$ CMOS process and displays a temperature sensitivity of $53\mu V/^{\circ}C$ for a $0.4V$ reference with a programming accuracy of $\pm 40\mu V$. These circuits demonstrate the practical feasibility of a floating-gate transistor based programmable approach. The circuits display comparable performance with other approaches with a significant area and power advantage while not compromising other performance metrics of the circuits.

1.2 Co-operative Analog-Digital Signal Processing

Traditionally, signal processing has been performed in the digital domain with analog circuits handling the interface with the outside world. There are a number of reasons that explain such a partitioning of responsibilities. The most important of these include the benefits of programmability and accuracy that digital designs offer. Analog designs, although can be designed to be precise, suffer from a lack of programmability. However, such a partitioning is not the most optimum in terms of overall power consumption. A power optimized partitioning of tasks between the analog and digital domains is the subject of co-operative analog-digital signal processing.

Consider a commonly used digital signal processing task, namely, finite impulse response (FIR) filtering. This involves repeated multiplication and addition operations. These operations can be implemented in both an area and power efficient manner in the analog domain. The key to the popularity of the digital approach over an analog implementation, as mentioned earlier is programmability. In the case of the FIR filter, in a digital implementation, the weights of the filter can be re-programmed such that a fixed filter architecture can be made to realize not only different frequency responses but also different filter types such as low-pass, high-pass, band-pass etc.

Programmable analog by way of using floating-gate transistors opens the door for a paradigm shift in signal processing. Programmability that has been the biggest stumbling block in analog designs is addressed using floating-gate transistors. Exploiting the current-voltage relationship of a floating-gate transistor, analog multiplication can be achieved by using the floating-gate device as both memory and computational element. The resulting circuits are compact and power efficient.

In this research, an analog implementation of basic signal processing primitives such as programmable multiply-and-accumulate units are analyzed from an accuracy

and power standpoint. A current-mode approach has been adopted in this work as addition can be implemented very effectively with virtually no additional power dissipation by invoking Kirchoff's current law. Programmable, current-mode analog multiply and accumulate units have been designed that achieve a million multiply-accumulate operations by dissipating $0.27\mu W$ of power. This is three orders of magnitude lower than those achievable by commercially available chips.

1.3 On-Chip Adaptation and Learning

Adaptive systems are inherently non-linear and time varying in nature. The key advantages to such systems are that these are self-optimizing and can be trained to perform specific filtering or decision-making tasks. Adaptive filters find use in a variety of applications such as adaptive equalization, prediction, system identification and interference cancellation.

Adaptive filters comprise of a collection of nodes interconnected through a number of synapses. Synapses provide the computation and adaptation in an adaptive system. They multiply the input signals by gain parameters called weights, store the weights in a local memory and change the weights according to some learning algorithm. Implementation of adaptive filters in the analog domain is motivated by the benefits of low-power multiplication and addition operations and the use of floating-gate transistors provide a compact non-volatile memory that is used to store weights.

The synapse that is the fundamental element of adaptive filters is implemented by using a floating-gate transistor for both weight storage and for weight adaptation. Adaptation is achieved by continuously enabling Fowler-Nordheim tuneling and hot-electron injection that are used to program floating-gate transistors. Exploiting the non-linearities inherent in these mechanisms yield a Least-Mean-Square learning rule. The resulting floating-gate based synapse circuits are compact, low-power and offer

the benefits of non-volatile weight storage.

In this work, a large scale 16×4 adaptive node has been designed using the floating-gate synapse along with all other supporting circuitry on-chip. Simple experiments that demonstrate adaptation such as a Fourier decomposition have been performed. In the Fourier decomposition experiment, the chip is presented with a target square wave and the first five harmonics at its inputs. The system adapts such that it learns the appropriate weights necessary to output a square wave thereby demonstrating learning. A simulation model (All Transistor Synapse (ATS)) for the floating-gate synapse has been developed as well. The developed simulation model, that models the physical phenomenon of hot-electron injection and tunneling using transistors is particularly advantageous for simulating large-scale adaptive filters and for circuits that require faster adaptation rates.

This dissertation begins with chapter 2 presenting a brief review of MOSFET equations that are used extensively in later chapters. Chapter 3 introduces floating-gate transistors, their programming techniques, design techniques for improving programming precision and retention in these devices along with experimental results from $0.5\mu m$ and $0.35\mu m$ CMOS technologies. Chapter 4 explains offset cancellation using floating-gate transistors and demonstrates the technique using a single stage folded cascode amplifier with measured results that show an offset voltage of $25\mu V$ being achieved. A programmable floating-gate based CMOS reference is discussed in chapter 5. Experimental results indicate references programmed to within $\pm 40\mu V$ with a temperature sensitivity of $130ppm/^{\circ}C$ for a $0.4V$ reference. Chapter 6 discusses the programmable floating-gate based multiplier along with results from a 128×32 vector matrix multiplier that achieves a 3 order power improvement over commercially available digital DSPs. Chapter 7 describes an adaptive system using a floating-gate synapse circuit along with experimental results. Chapter 8 discusses the simulation model for the floating-gate adaptive system and demonstrates adaptation by way of

a fourier decomposition experiment. Finally, chapter 9 concludes by presenting the impact of the work along with applications and future work.

CHAPTER 2

REVIEW OF MOSFET MODELING

In this chapter, a brief review of the key equations that model the operation of the MOS transistor is presented. Equations from the bulk referenced model [11, 12] are summarized. The charge sheet model is first presented followed by a derivation of the bulk referenced model. Only key equations that are necessary to understand the analyses that follow in the rest of dissertation are presented. The interested reader is referred to [11, 12] for a more detailed and thorough analysis of the MOSFET operation from weak to strong inversion.

2.1 Charge Sheet Model

Assuming that the inversion layer charge is of infinitesimal thickness and noting that the channel current is composed of both drift and diffusion components, the charge sheet model that is valid in all regions of inversion results. The channel current from drain to the source of a transistor is given by,

$$I_{DS} = I_{DRIFT} + I_{DIFF} \tag{3}$$

where, I_{DRIFT} is the component of the current due to drift and I_{DIFF} is the component of the current due to diffusion.

Consider an nMOS transistor of width W and length L with the length measured with respect to the source end of the transistor. The surface potential is denoted by ψ_s with ψ_{s0} and ψ_{sL} denoting its value at the source and drain ends respectively. Assuming that the surface mobility (μ) is constant across the length of the channel, the drift and diffusion currents can be expressed as,

$$I_{DRIFT} = \frac{W}{L}\mu\int_{\psi_{s0}}^{\psi_{sL}}(-Q_I^{'})d\psi_s \tag{4}$$

and,

$$I_{DIFF} = \frac{W}{L}\mu U_T(Q_{IL}^{'} - Q_{I0}^{'}) \tag{5}$$

where, $Q_I^{'}$ is the inversion layer charge per unit area and $Q_{I0}^{'}$ and $Q_{IL}^{'}$ represent the inversion layer charge at the source and drain ends respectively with U_T being the thermal voltage. An approximate expression for the inversion layer charge is given by,

$$Q_I^{'} = -C_{ox}^{'}\Big(V_{GB} - V_{FB} - \psi_s + \frac{Q_B^{'}}{C_{ox}^{'}}\Big) \tag{6}$$

where, V_{FB} is the flatband voltage, V_{GB} is the gate-bulk voltage, $C_{ox}^{'}$ is the oxide capacitance per unit area, ψ_s is the surface potential and $Q_B^{'}$ is the bulk depletion charge per unit area that is given by,

$$Q_B^{'} = -\gamma C_{ox}^{'}\sqrt{\psi_s} \tag{7}$$

where, γ is the body-effect co-efficient given by,

$$\gamma = \frac{\sqrt{2q\epsilon_s N_{sub}}}{C_{ox}^{'}} \tag{8}$$

with N_{sub} being the subtrate doping concentration, ϵ_s is the dielectric constant of silicon and q is the electronic charge. Substituting the expression for the bulk depletion layer charge in (6) results in the inversion layer charge as,

$$Q_I^{'} = -C_{ox}^{'}\Big(V_{GB} - V_{FB} - \psi_s - \gamma\sqrt{\psi_s}\Big) \tag{9}$$

Using the above equation for the inversion layer charge, the drift and diffusion current components can be expressed as,

$$I_{DRIFT} = \mu C_{ox}^{'}\frac{W}{L}\Big[(V_{GB} - V_{FB})(\psi_{sL} - \psi_{s0}) - \frac{1}{2}(\psi_{sL}^2 - \psi_{s0}^2) - \frac{2}{3}\gamma(\psi_{sL}^{3/2} - \psi_{s0}^{3/2})\Big] \tag{10}$$

and,

$$I_{DIFF} = \mu C_{ox}^{'}\frac{W}{L}\Big[U_T(\psi_{sL} - \psi_{s0}) + \gamma U_T(\sqrt{\psi_{sL}} - \sqrt{\psi_{s0}})\Big] \tag{11}$$

The source and drain end surface potentials can be expressed in terms of the externally applied terminal voltages as,

$$\psi_{s0} = V_{GB} - V_{FB} - \gamma\sqrt{\psi_{s0} + U_T exp\Big(\frac{\psi_{s0} - 2\phi_F - V_{SB}}{U_T}\Big)} \tag{12}$$

and,

$$\psi_{sL} = V_{GB} - V_{FB} - \gamma\sqrt{\psi_{sL} + U_T exp\Big(\frac{\psi_{sL} - 2\phi_F - V_{DB}}{U_T}\Big)} \tag{13}$$

where, ϕ_F is the bulk Fermi potential that is given by,

$$\phi_F = \frac{kT}{q} ln\Big(\frac{N_{sub}}{n_i}\Big) \tag{14}$$

with k being the Boltzmann's constant, T being the temperature in kelvins and n_i being the intrinsic carrier concentration. Solving the above set of equations results in the channel current through the device.

Although, the charge sheet model is accurate, it is complicated on account of the presence of powers of 1/2 and 3/2 in the diffusion and drift equations respectively. Tracing back, it is clear that these terms arise from the $\sqrt{\psi_s}$ term that is present in the expression for the bulk depletion layer charge and thereby in the inversion layer charge as well. Linearizing the expression for $Q_B^{'}$ around a convenient point results in a simplified expression for the channel current without the 1/2 and 3/2 power terms. Based on the choice of the point around which $Q_B^{'}$ is linearized, two modeling approaches have appeared in the literature over the years. Linearizing $Q_B^{'}$ around ψ_{s0} results in the source-referenced model while linearizing around the point ψ_{sa} (surface potential in the absence of an inversion layer) results in the bulk-referenced model. The bulk-referenced model handles weak inversion better and preserves the symmetry inherent in a simple MOS transistor [11]. For these reasons, the bulk-referenced model has been used extensively in this dissertation and these model equations will be discussed next.

2.2 Bulk-Referenced Model

In the bulk referenced model, the bulk depletion layer charge is linearized around the point $\psi_s = \psi_{sa}$. This results in,

$$\frac{-Q_B^{'}}{C_{ox}^{'}} = \gamma\sqrt{\psi_{sa}} + (n-1)(\psi_s - \psi_{sa}) \tag{15}$$

where, n is given by,

$$n = 1 + \frac{\gamma}{2\sqrt{\psi_{sa}}} = \frac{1}{\kappa} \tag{16}$$

and ψ_{sa} is given by,

$$\psi_{sa} = \Big[-\frac{\gamma}{2} + \sqrt{\frac{\gamma^2}{4} + V_{GB} - V_{FB}} \Big]^2 \tag{17}$$

The above linearization results in a close approximation of the depletion layer charge in weak inversion. It should be noted that the depletion layer charge is dominant in weak inversion. The errors in the approximation increase in moving from the weak inversion to strong inversion operation. However, in strong inversion, the inversion layer charge dominates over the bulk depletion layer charge, with the result that errors in the depletion layer charge do not result in appreciable errors in the drain current. The above approximation models $Q_B^{'}$ accurately in regions where $Q_B^{'}$ is dominant and the errors occur in regions where the contribution of $Q_B^{'}$ to the total semiconductor charge is negligible.

Using the linearized expression for the bulk depletion layer charge, the inversion layer charge is now given by,

$$Q_I^{'} = -C_{ox}^{'}[V_{GB} - V_{FB} - \psi_{sa} - \gamma\psi_{sa} - n(\psi_s - \psi_{sa})] \tag{18}$$

Notice that in comparison with (9), the above equation does not contain a $\sqrt{\psi_s}$ term. As will be shown below, this results in a simple expression for the drain current in both the weak and strong inversion regions of operation.

2.2.1 Strong Inversion

In strong inversion, the drift component of the channel current dominates over the diffusion component. Therefore, the drain-source current can be approximated to be composed only of the drift current. Using this approximation along with the linearized expression for the inversion layer charge and integrating (4) after making a change of variables results in,

$$I_{DS} = \frac{W}{L}\frac{\mu}{2nC'_{ox}}(Q'^2_{I0} - Q'^2_{IL}) \tag{19}$$

In strong inversion, the inversion layer charge can be approximated as,

$$Q'_I = -nC'_{ox}(V_P - V_{CB}) \tag{20}$$

where, V_{CB} is the channel to bulk potential and V_P is called the pinch-off voltage and is given by,

$$V_P = \psi_{sa} - \phi_o \approx \frac{V_{GB} - V_{To}}{n} \tag{21}$$

where, $\psi_o = 2\phi_F + several\ U_T$ and V_{To} is the bulk referenced threshold voltage of the device that is given by,

$$V_{To} = V_{FB} + \psi_o + \gamma\sqrt{\psi_o} \tag{22}$$

In non-saturation, both the source and the drain ends are strongly inverted. Therefore, using the above set of equations, the drift current is given by,

$$I_{DS} = \mu C'_{ox}\left(\frac{W}{L}\right)\left[(V_{GB} - V_{To})(V_{DB} - V_{SB}) - \frac{1}{2\kappa}(V^2_{DB} - V^2_{SB})\right] \tag{23}$$

It is easy to show that forward saturation occurs at $V_{DB} = V_P$ and therefore, the drain-source current at saturation is given by,

$$I_{DS} = \mu C'_{ox}\left(\frac{W}{2\kappa L}\right)\left[\kappa(V_{GB} - V_{To}) - V_{SB}\right]^2 \tag{24}$$

The above set of equations model the transistor operation in strong inversion with (23) modeling the linear region of operation and (24) modeling strong inversion saturation.

2.2.2 Weak Inversion

In weak inversion, the inversion layer charge is negligible and therefore, diffusion is the primary current transport mechanism. On account of this, the drain-source current is approximated to be consisting entirely of the diffusion current component and is given by (5). In weak inversion, the inversion layer charge is given by,

$$Q_I^{'} = -\frac{\gamma C_{ox}^{'}}{2\sqrt{\psi_{sa}}} U_T exp\Big(\frac{\psi_{sa} - 2\phi_F}{U_T}\Big) exp\Big(\frac{-V_{CB}}{U_T}\Big) \tag{25}$$

Using the above expression, the inversion layer charge per unit area at the source and drain ends of the channel can be calculated. Using this along with (5) and the expression for the pinch-off voltage and its relationship with ψ_{sa}, the drain-source current in weak inversion is given by,

$$I_{DS} = I_o^{'} \exp\Big(\frac{\kappa(V_{GB} - V_{To})}{U_T}\Big) exp\Big(\frac{-V_{SB}}{U_T}\Big)\Big(1 - exp\Big(\frac{-V_{DS}}{U_T}\Big)\Big) \tag{26}$$

where, the pre-exponential constant $I_o^{'}$ is given by,

$$I_o^{'} = \Big(\frac{1-\kappa}{\kappa}\Big)\mu_n C_{ox} \frac{W}{L} {U_T}^2 \exp\Big(\frac{\psi_o - 2\phi_F}{U_T}\Big) \tag{27}$$

In weak inversion, saturation occurs for values for drain-source voltage higher than $100 - 150mV$. In this case, the last term in (26) becomes equal to 1 and the drain-source current in weak inversion saturation is given by,

$$I_{DS} = I_o^{'} \exp\Big(\frac{\kappa(V_{GB} - V_{To})}{U_T}\Big) exp\Big(\frac{-V_{SB}}{U_T}\Big) \tag{28}$$

Note that the currents in the weak inversion region are exponential functions of the terminal voltages.

2.2.3 Moderate Inversion

In moderate inversion, the drain current is composed of both drift and diffusion components thereby complicating the development of a simple analytical expression for the drain current in this region. The popular approach to modeling in this region is to use semi-empirical expressions that provide an acceptable level of accuracy.

2.2.4 Complete Current Expression

The expressions developed using the charge sheet model are valid in all regions of operation from weak to strong inversion. In the sections following the treatment of the charge sheet model (sections 2.2.1 & 2.2.2), simplified expressions were given that model transistor operation in individual regions of operation. In order to unify the modeling in the various regions of operation to result in a complete continuous model that is valid in all regions of operation, the popular EKV model [12] approximates the drain current as,

$$I_{DS} = \mu C'_{ox}(2n)U_T^2\left(\frac{W}{L}\right)\left\{\left[ln\left(1+e^{(V_P-V_{SB})/2U_T}\right)\right]^2-\left[ln\left(1+e^{(V_P-V_{DB})/2U_T}\right)\right]^2\right\} \quad (29)$$

The above semi-empirical expression is valid in all regions of operation in both saturation and non-saturation modes.

In weak inversion, the exponential term inside the natural logarithm expression is small and therefore, one can invoke the expression $ln(1 + x)\approx x$ to simplify the above expression. This results in the non-saturation weak inversion expression given in (26) with the term $2n$ in place of $(n - 1)e^{(\phi_o-2\phi_F)/2U_T}$. This causes an error in the drain current I_{DS} that is usually adjusted for by varying the value of some other model parameter [11]. In strong inversion, the exponential term inside the natural logarithm expression dominates and the drain current expression reduces to the one given in (23). No such simplification is possible for moderate inversion with the result that the complete drain currrent expression has to be used.

2.3 Summary

Key equations that represent the operation of the MOS transistor have been presented to aid in the analyses that follow in the later chapters. The charge sheet model that is valid in all regions of operation has been discussed. In order to reduce the computational complexity involved in the charge sheet model, the bulk depletion layer charge is linearized around ψ_{sa}. This results in the bulk referenced model. The bulk

referenced model equations that model transistor operation in the weak and strong inversion regions have been summarized. The EKV model that uses a semi-empirical expression to give a continuous model valid in all regions of operation has been briefly discussed for the sake of completeness. The equations presented in this chapter have been extensively used in this dissertation. Wherever needed, the model equations are repeated for the convenience of the reader. Also, the equations presented here are valid for an nMOS transistor. Corresponding equations for a pMOS transistor can be obtained by making appropriate sign changes.

CHAPTER 3

FLOATING-GATE TRANSISTORS

The concept of a floating-gate device was proposed in 1967 by Kahng and Sze [13]. The first commercially available floating-gate based memory, the floating-gate avalanche-injection MOS device (FAMOS) was developed in 1970 [1]. Since then floating-gate transistors have been used extensively as non-volatile memory elements in EPROMs, EEPROMs and Flash memories [14]. Apart from being memory elements, floating-gate transistors can be used as computational elements as well. Exploiting this property of a floating-gate transistor, one can design high performance analog circuits, power-efficient signal processing primitives and synapses for building adaptive filters and neural networks for on-chip learning.

The successful use of floating-gate transistors in analog circuits depends on understanding certain key aspects of floating-gate transistors. These include, (1) Techniques that are used to transfer charge onto and from the floating-gate, thereby programming the device, (2) Translating system level specifications to programming accuracy of floating-gate transistors and understanding the design aspects that impact programming precision and (3) Charge retention capabilities of floating-gate transistors. In this chapter, a detailed discussion of these topics along with experimental results are presented.

3.1 Basics of Floating-Gate Transistors

A floating-gate MOS transistor is a transistor whose poly-silicon gate is completely surrounded by SiO_2, a high quality insulator. This creates a potential barrier that prevents charge stored on the floating-gate from leaking from the floating node. Figure 1 shows the circuit schematic and layout of a single-poly floating-gate pMOS transistor. In order to maintain the non-volatile charge storage of the floating-gate,

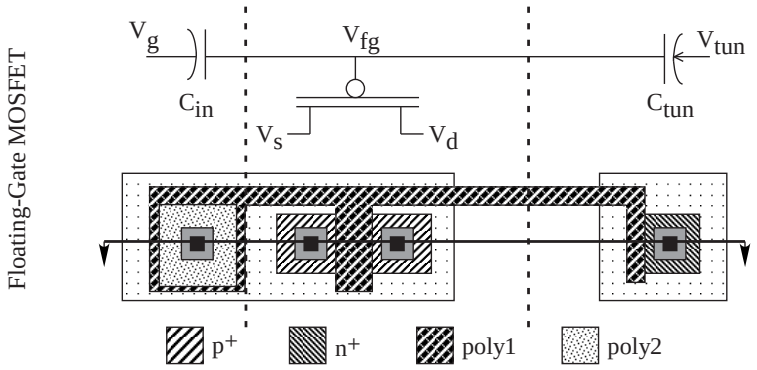


Figure 1. Circuit schematic and layout of a pFET floating-gate transistor: The floating-gate node V_{fg} is completely surrounded by SiO_2 and external inputs are coupled onto the floating node through an input capacitor C_{in}. The capacitor C_{tun} is used for Fowler-Nordheim tunneling. The input capacitor is implemented using a poly-poly capacitor while the tunneling capacitor is implemented using a MOS capacitor.

external inputs are capacitively coupled through an input capacitor C_{in}. It should be noted that the second polysilicon layer shown in Figure 1 is used primarily to implement the input capacitor. The tunneling capacitor C_{tun} is implemented using the gate oxide between the gate poly-silicon and n-well. A key advantage of the floating-gate device is that it provides programmability in a standard digital CMOS process.

Figure 1 shows the circuit diagram of a pFET floating-gate transistor. As can be observed, a floating-gate transistor is very similar to a normal transistor with a critical difference being that inputs are capacitively coupled onto the floating-gate. Consider, a floating-gate transistor in the strong inversion region of operation. Neglecting Early effects and assuming saturation, the drain current through the device is given by [11, 15, 12],

$$I_{sd} = \frac{\mu_p C_{ox} W}{2\kappa L}(V_s - \kappa V_{fg} - \kappa \left|V_{To}\right| + (\kappa - 1)V_b)^2 \tag{30}$$

where, I_{sd} represents the source-drain current, μ_p represents the effective hole mobility, C_{ox} is the oxide capacitance per unit area, W is the effective width of the device, L is the effective length of the device, V_{fg} is the floating gate voltage, V_s is the source voltage, V_b is the bulk voltage, V_{To} is the threshold voltage of the device referred to

the bulk, and κ is given by,

$$\kappa \approx \frac{C_{ox}}{C_{ox} + C_{dep}} \tag{31}$$

where, C_{dep} is the depletion capacitance per unit area. The above set of equations are part of what is commonly referred to as the *bulk referenced model* for MOS transistors as described in Chapter 2.

The floating-gate voltage can be expressed in terms of the terminal voltages and the charge stored on the floating-gate. Using these, the floating-gate voltage is given by,

$$V_{fg} = \frac{C_{in}}{C_T} V_g + \frac{Q}{C_T} \tag{32}$$

where, Q is the charge stored on the device and C_T is the total capacitance of the floating-node. Substituting (32) in (30) results in the source-drain current expression being modified as,

$$I_{sd} = \frac{\mu_p C_{ox} W}{2\kappa L} \Big(V_s - \kappa_{eff} V_g - \kappa(Q/C_T + |V_{To}|) + (\kappa - 1)V_b) \Big)^2 \tag{33}$$

where, κ_{eff} is the effective κ of the floating-gate device and is given by,

$$\kappa_{eff} = \kappa \frac{C_{in}}{C_T} \tag{34}$$

From (33), it is clear that the charge stored on the floating-gate can be viewed as modifying the threshold voltage of the device. Thus, by removing electrons from the floating-gate of a pFET, the effective threshold voltage of the device can be increased and the threshold voltage can be decreased by adding negative charge. The opposite will be true in the case of an nFET floating-gate transistor.

Now consider, a pFET floating-gate transistor operating in the weak inversion region of operation. The source-drain current through the device, ignoring Early effects and following the analysis as before is given by,

$$I_{sd} = I_o^{'} \exp\left(\frac{-\kappa_{eff} V_g}{U_T}\right) \exp\left(\frac{V_s}{U_T}\right) \exp\left(\frac{-\kappa(|V_{To}| + Q/C_T)}{U_T}\right) \exp\left(\frac{(\kappa - 1)V_b}{U_T}\right) \tag{35}$$

where, $U_T = kT/q$ is the thermal voltage, $I_o^{'}$ is a pre-exponential constant made up of fundamental device parameters and the other terms are as defined earlier in Chapter 2. As in the case of strong inversion, the floating-gate charge can be lumped along with the threshold voltage of the device. In the case of a pFET, a net positive charge on the floating-gate results in a lower current through the device and shifts the weak inversion $I-V$ characteristic to the left and a net negative charge on the floating-gate results in a higher current through the device and shifts the $I-V$ characteristic to the right. Similar results can be derived for an nFET taking care to reverse the polarities of all terminal voltages.

3.2 Floating-Gate Charge Modification Techniques

Programming a floating-gate transistor involves adding or removing charge from the floating-gate thereby modulating the threshold voltage of the device as shown earlier. There are a number of techniques that can be used to modify the charge on a floating-gate and the three commonly used techniques include (1) Exposure to ultra-violet radiation, (2) Fowler-Nordheim Tunneling and (3) Hot-electron injection. The following paragraphs discuss these in detail.

3.2.1 Ultra-Violet Radiation

Exposure of floating-gate transistors to ultra-violet (UV) radiation is a commonly used technique in EPROM devices to "erase" the information stored on the floating-gate [1]. In EPROM devices, information is stored by transferring electrons onto the floating-gate. Exposing the floating-gate transistor to high energy UV rays imparts sufficient kinetic energy to the electrons stored on the floating-gate to surmount the $Si - SiO_2$ barrier, thereby "erasing" the floating-gate. The obvious disadvantage of this technique is the requirement of a seperate set-up that generates UV radiation. Also, the exact amount of charge transferred to/from the floating-gate is not

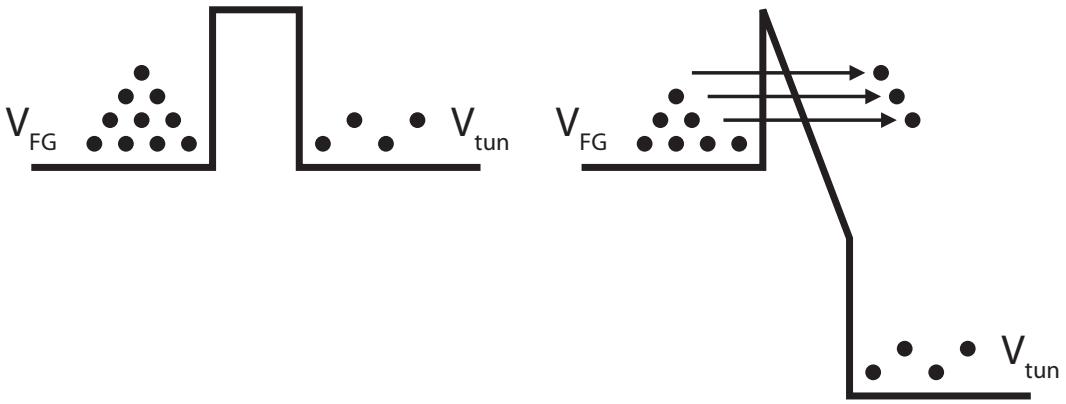


Figure 2. Fowler-Nordheim Tunneling Process: (a) $Si - SiO_2$ energy band diagram for the case of no applied electric field. The average thermal energy of electrons in the conduction band of Si is such that the probability of electrons surpassing the $Si - SiO_2$ barrier is very low. (b) The distortion of the bands when a high enough electric field is applied across the $Si - SiO_2$ interface. Under these conditions, there is a finite probability that electrons quantum-mechanically tunnel across the $Si - SO_2$ barrier.

controlled. This makes the technique unsuitable for accurate programming of floating-gate transistors. Besides, the technique is inefficient and time-consuming on account of significant absorption of UV rays by silicon and poly-silicon [1].

3.2.2 Fowler-Nordheim Tunneling

Fowler-Nordheim (FN) tunneling, named after Fowler and Nordheim, was first described in a $Si - SiO_2$ system by Leslinger and Snow in 1969 [16]. Using this phenomenon, electrons tunnel across the $Si - SiO_2$ barrier as a consequence of the high electric field that is applied across the $Si - SiO_2$ interface. Figure 2 shows the band diagram at the $Si - SiO_2$ interface both in the presence and absence of an electric field. In the absence of an electric field, shown in Figure 2(a), the difference between the conduction bands of Si and SiO_2 is approximately $3.2eV$. It should be noted that Si has an energy band-gap of about $1.1eV$ while SiO_2 has an energy band-gap of around $9eV$. At room temperature, the average electrons have an energy of around $0.025eV$, thereby, leading to a very low probability that the electron would surmount

the $3.2eV$ $Si - SiO_2$ barrier [1]. However, when a high enough electric field is applied at the $Si - SiO_2$ interface, the bands are distorted as shown in Figure 2(b). Under these conditions, there is a finite probability that an electron can quantum-mechanically tunnel through the barrier and make it to the conduction band of SiO_2. In Figure 2(b), the electric field was applied in the direction that caused electrons to tunnel from Si to SiO_2. Applying an electric field in the opposite direction causes electrons to tunnel from SiO_2 to Si.

This phenomenon is exploited in a floating-gate system to both add and remove electrons. Figure 1 shows the layout and schematic of the tunneling junction used to remove electrons from the floating-gate. The tunneling junction is nothing but a capacitive connection to the floating-gate. The floating-gate forms one terminal of the capacitor while the other terminal is called the tunneling voltage, V_{tun}. The tunneling capacitor is fabricated as a MOS capacitor formed on an n-well. The tunneling voltage makes an ohmic contact to the n-well using an n+ diffussion layer. It should be noted that the poly-silicon floating-gate is directly connected to the poly-silicon gate of the tunneling MOS capacitor. Using the above scheme, applying a high tunneling voltage ($> 15V$ for a $0.5\mu m$ CMOS process), high electric fields are generated at the $Si-SiO_2$ interface and electrons tunnel across and are collected at the n-well. The tunneling current can be modelled as a function of the terminal voltages as [17, 18],

$$I_{tun} = I_{tun0} exp\Big(\frac{V_{tun} - V_{fg}}{V_x}\Big) \tag{36}$$

where, V_{fg} is the floating-gate voltage, V_x is a process dependent parameter and I_{tun0} is an equilibrium tunneling current.

Figure 3 shows the practical implementation of the FN tunneling process in floating-gate transistors. Initially, the tunneling voltage is held low such that electric fields across the $Si - SiO_2$ interface is low enough that the probability of electrons tunneling across the barrier is negligible. Next, the tunneling voltage is increased to a high enough voltage that causes conditions conducive for tunneling to occur.

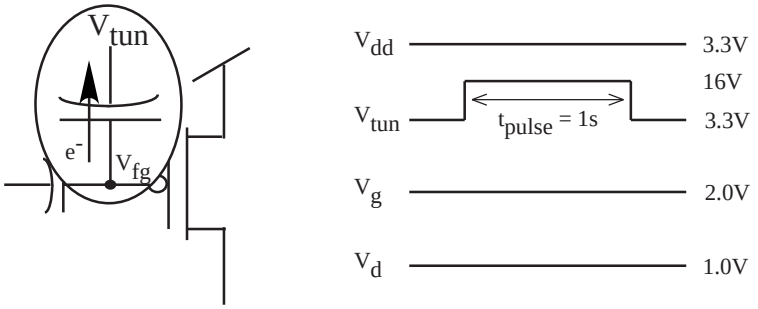


Figure 3. Performing Fowler-Nordheim tunneling in floating-gate transistors: Pictorial representation of the Fowler-Nordheim tunneling process in floating-gate device. Electrons tunnel from the floating-gate when the tunneling voltage is increased to a high enough value $> 15V$ for a $0.5\mu m$ CMOS process.

The amount of charge transferred due to tunneling depends on the tunneling voltage and the amount of time the high electric field is sustained across the $Si - SiO_2$ barrier. Since, the tunneling currents depend exponentially on the tunneling voltage, care should be taken when increasing the tunneling voltage to prevent accidental complete erasure of the floating-gate.

3.2.3 Hot-Electron Injection

Figure 4 schematically represents hot-electron injection process in pFETs. Hot-electron injection occurs in pFETs when carriers are accelerated to a high enough energy level to surmount the $Si - SiO_2$ barrier. At high electric fields and in the presence of drain currents, electrons are created at the drain edge of the drain-to-channel depletion region via hot-hole impact ionization. These electrons travel back into the channel region, gain sufficient kinetic energy such that they cross the $Si-SiO_2$ barrier and are injected onto the floating-gate [19]. Electrons that do not cross the $Si - SiO_2$ barrier are swept away towards the bulk and flow as bulk currents. Conditions conducive for hot-electron injection are created when the transistor experiences a high source-drain potential and when there is channel current flowing through the device. In order for hot-electron injection to occur, the electric fields in the channel must be greater than $10V/\mu m$ and the electric field across the oxide must be in a

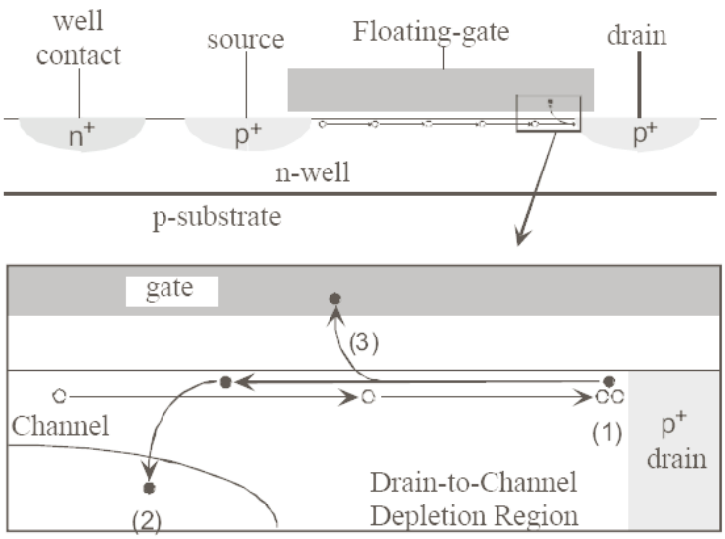


Figure 4. Hot-Electron Injection Process: Under high electric fields caused by a high source-drain voltage, impact ionization occurs creating electron-hole pairs. Electrons that have a high enough kinetic energy surmount the $Si - SiO_2$ barrier and are injected onto the floating-gate. The other electrons flow out through the bulk terminal while the holes flow through the drain terminal.

direction that aids transport of the electrons across the barrier.

A simplified model for hot-electron injection is given in detail in [19]. The model relates the injection current (I_{inj}) to terminal voltages and to the channel current as,

$$I_{inj} = I_{inj0}\left(\frac{I_s}{I_{s0}}\right)^{\alpha} exp\left(\frac{-\Delta V_{ds}}{V_{inj}}\right) \tag{37}$$

where, I_s is the channel current, I_{inj0} is the injection current corresponding to a channel current of I_{s0}, ΔV_{ds} is the incremental drain-source voltage across the device, $\alpha = U_T/V_{inj}$ and V_{inj} is a process and bias-dependent parameter. Typical values for V_{inj} range from $100mV - 250mV$ for a $0.5\mu m$ CMOS process. As can be observed, hot-electron injection depends on the presence of a channel current and high electric fields made possible through a large value of source-drain voltage.

Figure 5 shows the practical implementation of the hot-electron injection process in a floating-gate device. To perform hot-electron injection on a floating-gate transistor, the chip is ramped up such that V_{DD} is increased to $6.5V$ with all other voltages increased with respect to V_{DD} as well. Next, the high fields necessary for injection

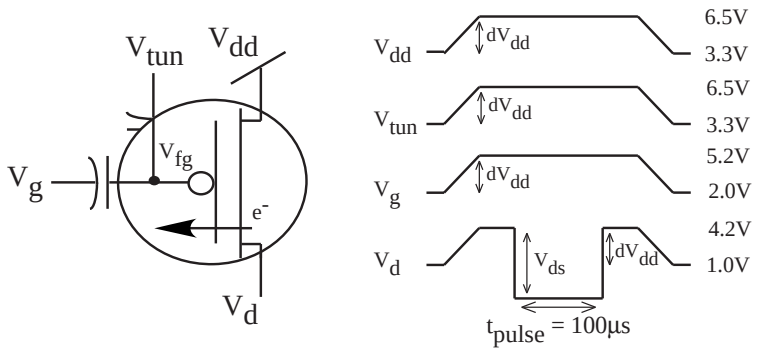


Figure 5. Performing hot-electron injection in floating-gate transistors: Pictorial representation of the hot-electron injection process. Hot-electron injection occurs when a high enough source-drain voltage is applied across the device while there is drain current flowing through the device.

are created by pulsing down the drain voltage (V_D) for a certain amount of time t_{pulse} such that a high source-drain voltage appears across the device. Typical V_{SD} voltages used for hot-electron injection range from $4V$ - $6.5V$ for a $0.5\mu m$ CMOS process while for a $0.35\mu m$ process, typical values for V_{SD} range from $3V - 5.5V$. After injection is completed, the chip is ramped down such that all voltages are restored to their original values. The number of electrons injected and hence the change in the drain current is a function of the source-drain potential V_{SD} and the time interval t_{pulse} for which V_{SD} is held high enough.

3.3 Automatic Programming of Floating-Gate Transistors

Given the techniques used for modifying charge on a floating-gate, tunneling and hot-electron injection are more suited for precision programming than using UV radiation. To demonstrate programming using hot-electron injection and tunneling, a floating-gate pFET transistor has been programmed to different threshold voltages with their magnitudes ranging from $0.75V - 2.75V$ as demonstrated in Figure 6. It should be noted that the absolute value of the threshold voltage of a pFET device that is not a floating-gate in the $0.5\mu m$ CMOS process used is $0.9V$. Figure 6(b) demonstrates programming in a $0.35\mu m$ CMOS process. As can be observed, a floating-gate pFET

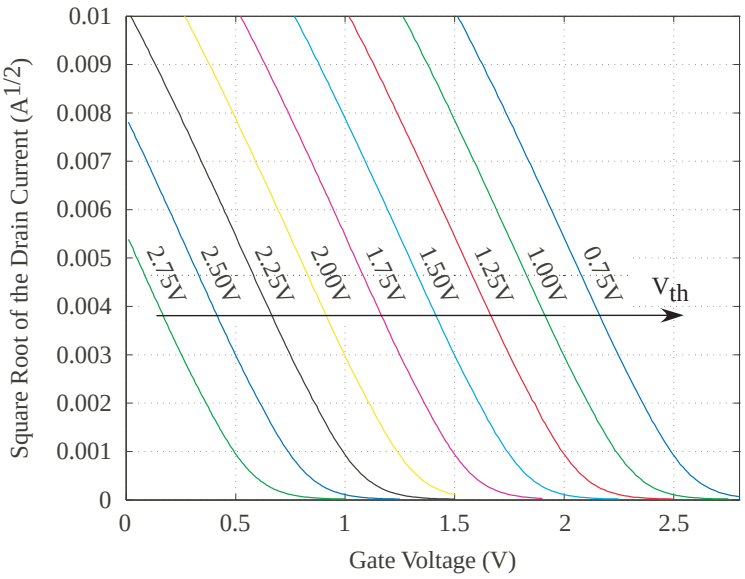


Figure 6. Programming a floating-gate pFET transistor in a $0.5\mu m$ CMOS process: A floating-gate pFET has been programmed using a combination of hot-electron injection and Fowler-Nordheim tunneling where the threshold voltage of a vanilla pFET in the process is $0.9V$.

has been programmed to threshold voltages ranging from $0.3V - 2.3V$ while the threshold voltage of a non floating-gate pFET in the process is $0.7V$. These figures clearly demonstrate scalability along with a wide range in programming capabilities of the floating-gate device.

As can be observed from (36), the relationship between the tunneling current and the tunneling voltage is logarithmic in nature and this makes precision programming time-consuming. Techniques have been proposed to improve the speed and precision of tunneling based programming [20, 21]. However, in this work, tunneling is used primarily as a global erase and precision programming is achieved through hot-electron injection. Such a scheme has a number of advantages over a tunneling based programming scheme as in [22, 23]. These include avoiding special processing steps such as ultra-thin tunneling oxides that are needed to increase the speed of tunneling and constant application of high voltages of both positive and negative polarities.

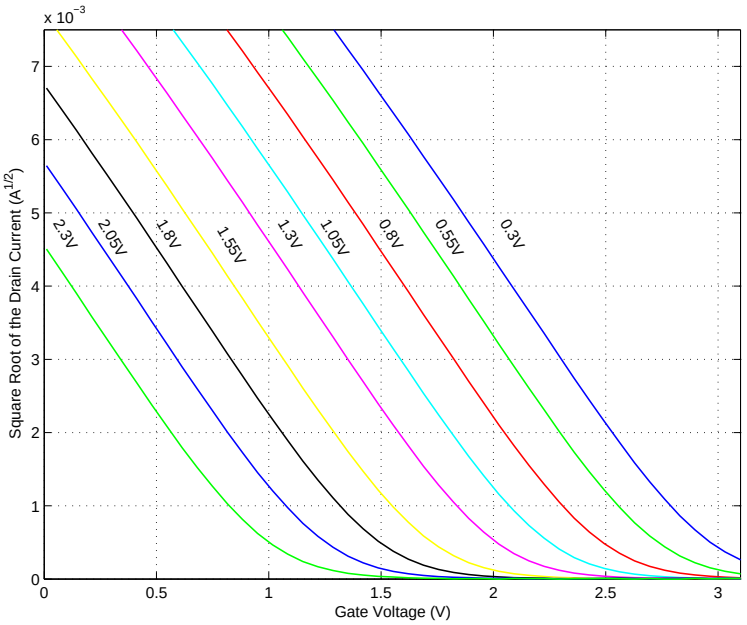


Figure 7. Programming a floating-gate pFET transistor in a $0.35\mu m$ CMOS process: A floating-gate pFET has been programmed using a combination of hot-electron injection and Fowler-Nordheim tunneling where the threshold voltage of a vanilla pFET in the process is $0.7V$.

Automatic programming of floating-gate transistors is achieved using a programming algorithm that calculates the value of the source-drain potential and the number of pulses required (with t_{pulse} held fixed) such that a target current is reached without any overshoot. The value of V_{SD} for a given pulse interval is estimated from the relationship between the initial drain current and the desired target current. This is obtained from a first-principles model for hot-electron injection and is described in detail in [19, 2]. A single initial calibration step that characterizes the hot-electron injection rates for a given process is performed such that the algorithm can predict an optimal value of V_{SD} during injection. Programming proceeds by first measuring the initial drain current of the device. This is used by the programming algorithm along with the target drain current to calculate the optimal value of V_{SD} for a fixed pulse interval of about $100\mu s$. The chip is then ramped up and the calculated V_{SD} is applied. The chip is then ramped down and the drain current is measured again. If

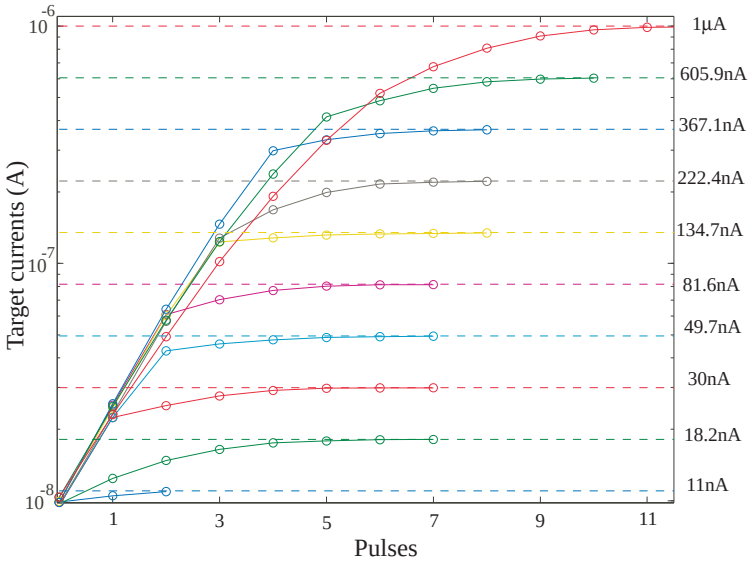


Figure 8. Convergence of the programmaing algorithm[2]: The convergence of the programming algorithm for different target currents from an initial starting current of $10nA$ is shown. The algorithm convergences to within 0.1% of the target current in all cases with the pulse width being $100\mu s$.

the measured current is different from the target current, the algorithm calculates a new V_{SD} value and injection is performed again. The above steps are repeated until the drain current of the device reaches the target value within a pre-defined tolerance. Since the injection rates vary between devices, programming is performed asymptotically such that overshoot is avoided. Typical convergence to target to within a 0.1% accuracy takes about $7-12$ pulses.

Figure 8 [2] demonstrates the programming algorithm when the drain current of the device is programmed to different target currents (logarithmically spaced) from an initial current of $10nA$. As can be observed, typically around 7 pulses are required to reach the target with the worst-case of 11 pulses for a programming change of 2 orders of magnitude. A pulse width of $100\mu s$ has been used in this case.

3.4 Floating-Gate Programming Precision

The accuracy to which one can program floating-gate transistors to a target current depends on the smallest drain current change that can be programmed onto a floating-gate device. In order to estimate the design choices available to improve programming precision, a figure of merit (FOM) is defined as,

$$FOM = -\log_2\Big(\frac{\Delta I}{I}\Big) \tag{38}$$

where, ΔI is the minimum programmable change in drain current that is necessary to meet a system level accuracy specification and I is the bias current of the floating-gate transistor. It should be noted that such a definition results in the FOM being represented in the familiar binary system, as number of bits of accuracy achievable. In the discussion below, the FOM is related to floating-gate circuit parameters for operation in both the weak and strong inversion regimes such that the floating-gate transistor can be designed to achieve the required bits of precision.

3.4.1 FOM in Weak Inversion

Consider a floating-gate pFET operating in the weak inversion regime. The source-drain current of the device ignoring Early effect and using (35) is given by,

$$I = I_o \exp\Big(\frac{-\kappa V_{fg}}{U_T}\Big)\exp\Big(\frac{V_s}{U_T}\Big) \tag{39}$$

where, I_o is the pre-exponential constant that includes $I_o^{'}$ and the threshold voltage and bulk voltage terms given in (35).

Now, for a ΔV_{fg} change in the floating-gate voltage, a ΔI change in drain current results. The net programmed drain current of the device is given by,

$$I + \Delta I = I_o \exp\Big(\frac{-\kappa(V_{fg} + \Delta V_{fg})}{U_T}\Big)\exp\Big(\frac{V_s}{U_T}\Big) \tag{40}$$

Dividing (40) by (39) and noting that $\Delta V_{fg} = \Delta Q / C_T$, the achievable change in drain current due to programming relative to the initial drain current is given by,

$$\frac{\Delta I}{I} = \exp\Big(\frac{-\kappa \Delta Q}{U_T C_T}\Big) - 1 \tag{41}$$

where, C_T is the total capacitance at the floating-gate and ΔQ is the programmed charge.

Note that for most cases, the term inside the exponential is much less than one, and therefore, the Taylor series approximation for the exponential can be used to arrive at the simplified expression shown below,

$$\frac{\Delta I}{I} \approx \frac{-\kappa \Delta Q}{U_T C_T} \tag{42}$$

It is clear from (42) that the achievable precision is inversely proportional to the charge that can be reliably transferred onto the floating-gate and directly proportional to the total floating-gate capacitance.

3.4.2 FOM in Strong Inversion

Consider a floating-gate nFET operating in the strong inversion region and programmed using an indirect programming scheme as outlined in [24]. Ignoring Early effects, the drain current of the device in saturation is given by [11, 12],

$$I = \frac{\mu_n C_{ox} W}{2\kappa L}(\kappa V_{fg} - V_s - \kappa V_{To})^2 \tag{43}$$

where, μ_n is the effective mobility of electrons and all other variables are as defined earlier in Chapter 2.

Programming the device such that a charge transfer of ΔQ results in a change in the floating-gate voltage of ΔV_{fg}, modifies the drain current to be,

$$I + \Delta I = \frac{\mu_n C_{ox} W}{2\kappa L}(\kappa(V_{fg} + \Delta V_{fg}) - V_s - \kappa V_{To})^2 \tag{44}$$

Dividing (44) by (43) and manipulating the algebra with the assumption that (ΔV_{fg}) is much smaller than the overdrive voltage $V_{od} = \kappa V_{fg} - V_s - \kappa V_{To}$ results in,

$$\frac{\Delta I}{I} = \frac{2\kappa \Delta V_{fg}}{\kappa V_{fg} - V_s - \kappa V_{To}} = \frac{2\kappa \Delta Q}{V_{od} C_T} \tag{45}$$

As can be observed from (45), the achievable precision is inversely proportional to the charge that can be transferred onto the floating-gate and directly proportional to the overdrive voltage of the device and the total floating-gate capacitance.

Comparing the above expression for programming precision in strong inversion with that in weak inversion, it should be noted that with exactly the same total floating-gate capacitance and equal amounts of charge transferred, the achievable precision is higher in the strong inversion region than in the weak inversion region. This can be verified by analyzing (42) and (45). Apart from a factor of 2, the equations differ in the denominator with the thermal voltage (U_T) being replaced by the overdrive voltage (V_{od}) in strong inversion. Typically, the thermal voltage is $\approx 26mV$ at room temperature ($298K$) while the overdrive voltage is a design parameter in analog circuits with values in the range of $150mV - 300mV$. This results in the FOM being a lot higher in strong inversion than in weak inversion.

3.4.3 FOM Experimental Results

In order to verify the theory presented above, a test chip was fabricated in a $0.5\mu m$ standard CMOS process that consisted of an array of floating-gate pFET transistors with the same aspect ratio but with varying input capacitors such that C_T varied from one transistor to another. According to (42), for a given charge transfer ΔQ onto the floating-gate transistor, the FOM varies inversely with capacitance. This was verified by injecting a constant charge ΔQ onto floating-gate transistors with different total gate capacitance (C_T) and measuring the change in the drain current to the initial drain current before injection. In order to ensure a constant charge transfer on all devices, the initial drain current for all devices was kept constant and a source-drain voltage (V_{sd}) of $6V$ was applied for a time period of $1mS$ for all devices. Figure 9 shows a plot of the measured $\Delta I/I$ vs. $1/C_T$. The plot is linear as expected from theory.

Noting that for a transistor operating in the strong inversion regime the overdrive voltage can be expressed as $V_{od} = \sqrt{\frac{2\kappa I}{\beta}}$, it can be inferred from (45) that the FOM is inversely proportional to the square-root of drain current. Also, from (42), it is clear that the FOM is independent of drain current in the weak inversion regime. Therefore,

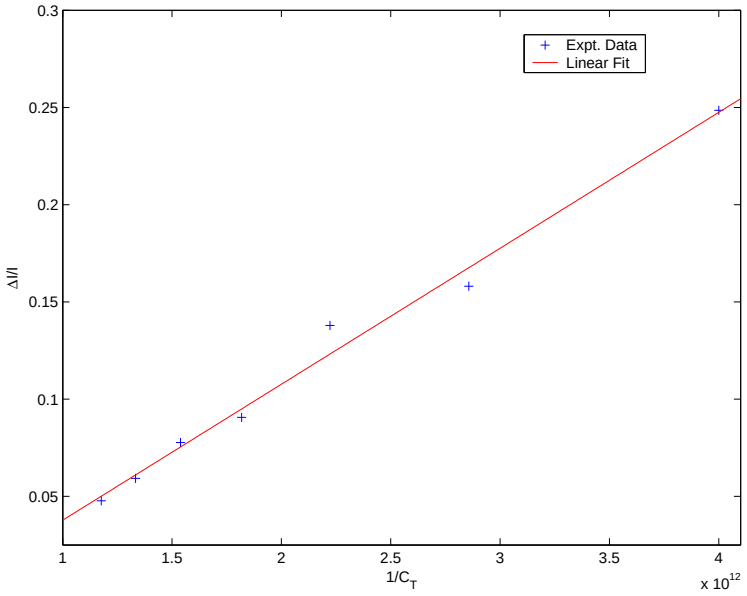


Figure 9. Plot of $\Delta I/I$ vs. floating-gate capacitance:Measured plot of $\Delta I/I$ against the total floating-gate capacitance (C_T) for a constant charge injected at a drain current of $100nA$ with a source-drain voltage of $6V$ for a time period of $1mS$. The plot is linear as expected from theory.

one would expect that the plot of FOM vs. drain current would be constant in the weak inversion regime and would increase in the strong inversion regime. This was verified by injecting a constant charge onto a floating-gate transistor and by measuring the $I - V$ characteristic both before and after injection. Calculating the difference in currents between the $I - V$ sweeps and plotting against the initial set of currents results in the plot shown in Figure 10. As can be observed, the plot is constant in the weak inversion regime and increases in the strong inversion regime thereby verifying the theory.

Table 1 presents quantitative numbers for the FOM for both the weak inversion and strong inversion regions based on the theory developed above. The FOM has been calculated for different values of charge transfer and C_T for a κ of 0.7, U_T of $26mV$ and an overdrive voltage of $250mV$. Figure 11 presents experimental data from programming an array of floating-gate transistors such that a sinusoid with a DC offset of $1\mu A$ and an amplitude of $20nA$ results. Also, shown is the percentage error in

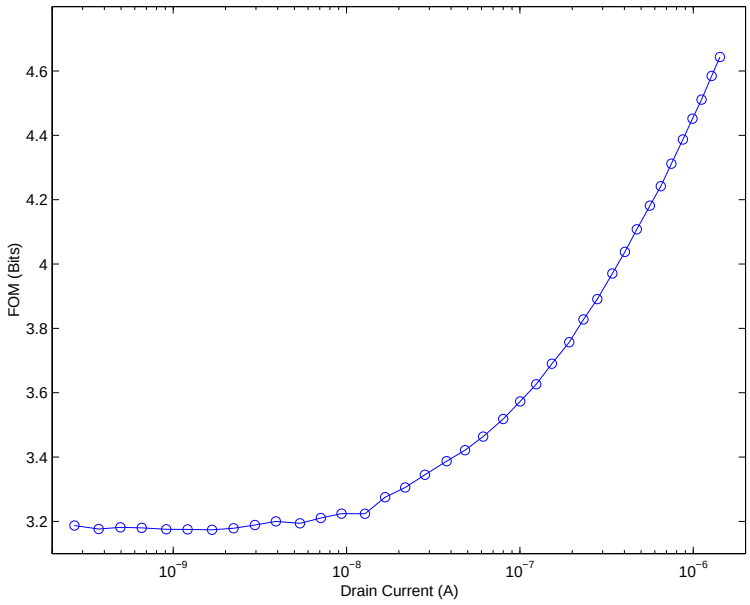


Figure 10. The FOM plotted against the drain current of a floating-gate transistor: The FOM is independent of the drain current in the weak inversion region of operation and increases as the transistor enters the strong inversion regime. The experimental results are consistent with the theoretical prediction.

programmed value of the sinewave to the ideal value [2]. The error is within $\pm 0.05\%$ indicating an FOM of approximately 11 bits. The total floating-gate capacitance for these transistors is approximately $100fF$. With the devices operating in strong inversion, it can be inferred from Table 1 that a little over $100e^-$ worth of charge is responsible for the measured precision.

Using the above developed theory and depending on the region of operation of the floating-gate transistor, one can design a floating-gate transistor (C_T) such that a target accuracy specification is met. For the sake of clarity, further discussions on designing to meet a system-level specification is deferred until Chapter 4.

3.5 Retention in Floating-Gate Transistors

Floating-gate transistors inherently have good charge retention capabilities on account of the gate being surrounded by a high quality insulator. Initial investigations of

Table 1. Summary of the achievable bits of accuracy (FOM)

	Weak Inversion			**Strong Inversion**		
$\Downarrow C_T\ \Delta Q \Rightarrow$	$1e^-$	$10e^-$	$100e^-$	$1e^-$	$10e^-$	$100e^-$
10fF	11.18	7.85	4.83	13.44	10.12	6.8
100fF	14.5	11.18	7.85	16.76	13.44	10.12
1pF	17.82	14.5	11.18	20.09	16.76	13.44

floating-gate retention were carried out by observing the drain current of a floating-gate device for long periods of time. Figure 12(a) shows the drain current of a floating-gate pFET measured over a period of 380 hours. The drain current was programmed to an initial value of $30\mu A$ and displayed a mean value of $29.93\mu A$ with a standard deviation of $28nA$. The current exhibits a short-term drift in the beginning beyond which no significant drift can be observed. This short-term drift is on account of the interface trap sites settling to a new equilibrium after programming [23]. Similar results have been observed in a $1.5\mu m$ CMOS process [25]. Although this is a good indicator of the charge retention capabilities of floating-gate transistors, accurate estimates of the long-term charge retention can be made through accelerated life time tests.

Long-term charge loss in floating-gates occur due to the phenomenon of thermionic emission [26, 27, 22, 23]. The amount of charge lost is a function of both temperature and time and is given by,

$$\frac{Q(t)}{Q(0)} = exp\Big[-t\upsilon \cdot exp\Big(\frac{-\phi_B}{kT}\Big)\Big] \tag{46}$$

where, $Q(0)$ is the initial charge on the floating-gate, $Q(t)$ is the floating-gate charge at time t, υ is the relaxation frequency of electrons in poly-silicon, ϕ_B is the effective $Si - SiO_2$ barrier potential in electron-volts, k is the Boltzmann's constant and T is the temperature. As expected from (46), charge loss in floating-gates is a slow process that is accelerated at high temperatures.

Floating-gate charge loss is measured indirectly by measuring the change in the

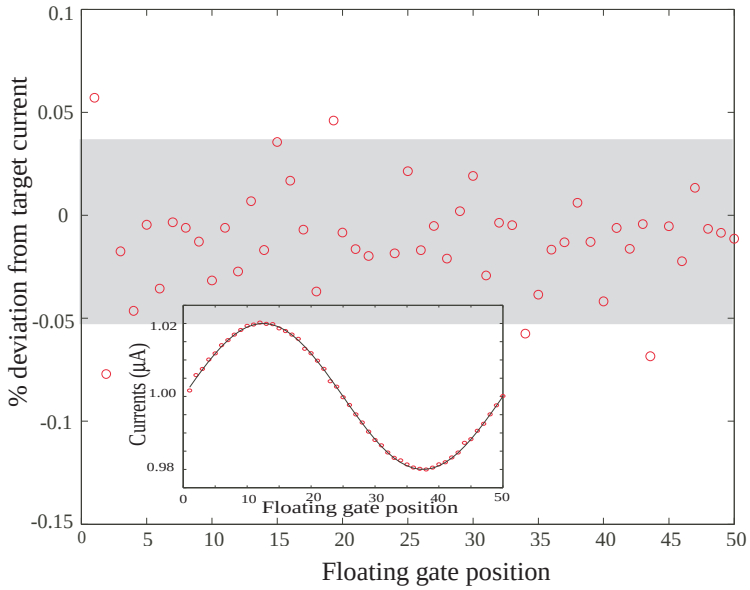


Figure 11. Programming precision [2]: Programming a $20nA$ sinusoid riding on a DC value of $1\mu A$ is shown along with the percentage error between the programmed current and the desired target. As can be observed, an error of $\pm 0.05\%$ has been achieved.

transistor's threshold voltage. Programming floating-gates by adding/removing charge modifies the threshold voltage of the device. The effective threshold voltage of the device, $V_{To}^{'}$, is given by,

$$V_{To}^{'} = V_{To} + \frac{Q}{C_T} \tag{47}$$

where, Q is the floating-gate charge, V_{To} is the threshold voltage of the transistor with zero floating-gate charge or that of a non floating-gate device and C_T is the total capacitance at the gate node. Using the above approximation for the threshold voltage of a floating-gate device the charge loss in a floating-gate can be rewritten as,

$$\frac{Q(t)}{Q(0)} = \frac{V_{To}^{'}(t) - V_{To}}{V_{To}^{'}(0) - V_{To}} \tag{48}$$

where, $V_{To}^{'}(t)$ indicates the effective threshold voltage of the device after time t and $V_{To}^{'}(0)$ represents the initial programmed threshold voltage.

Estimating the amount of charge loss in floating-gate transistors using (46) requires the estimation of the parameters υ and ϕ_B as these parameters exhibit a wide

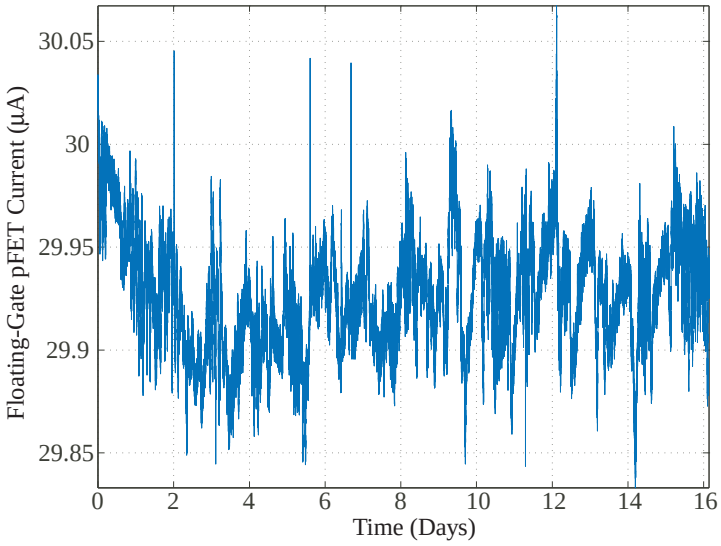


Figure 12. Drain Current of a Floating-Gate pFET: The drain current of a floating-gate pFET measured over 16 days. The floating-gate transistor was programmed to an initial value of $30\mu A$.

spread in their values and therefore need to be extracted for each process. For the $0.5\mu m$ process used in the design, floating-gate pFETs were programmed to a threshold voltage of $-0.5V$ and stored at high temperatures for a predefined time period. The change in threshold voltage is measured and using (48) the charge loss is estimated. Using (46), (48) and the measured data points, υ and ϕ_B can be extracted using,

$$\phi_B = \frac{kT_1T_2}{T_1 - T_2} ln\left[\frac{t_2}{t_1} \cdot \frac{ln(x_1)}{ln(x_2)}\right] \tag{49}$$

and,

$$\upsilon = \frac{-ln(x_1)}{t_1 \cdot exp(\frac{\phi_B}{kT_1})} \tag{50}$$

where, x denotes the ratio of the floating-gate charge at time t to the initial floating-gate charge and the subscripts denote two different data points measured at two different temperatures (T_1, T_2) and times (t_1, t_2). Using the above procedure, the values for the barrier potential and the relaxation frequency were extracted to be $0.9eV$ and $60s^{-1}$ for the $0.5\mu m$ CMOS process used in the experiments. For a $0.35\mu m$

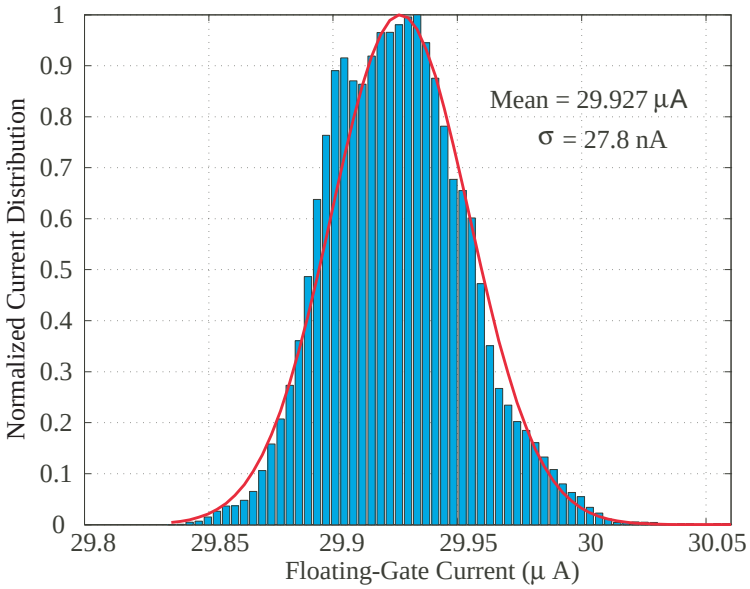


Figure 13. Current Distribution of a Floating-Gate pFET: The drain current distribution indicates a mean of $29.927\mu A$ with a standard deviation of $27.8nA$. The gaussian nature of distribution indicates the presence of thermal noise on the measured data.

CMOS process, similar experiments have been conducted and the values of barrier potential and relaxation frequency have been extracted to be $0.618eV$ and $55ms^{-1}$ respectively.

Figure 14 shows the measured floating-gate charge loss along with a theoretical extrapolated fit using the estimated model parameters. The measured data ($0.5\mu m$ CMOS process) agrees well with the theoretical prediction and the trends observed in Figure 14 have been observed across many floating-gate devices and across processes as well.

3.5.1 Retention in Floating-Gate Transistor Pair

In analog applications using floating-gate transistors, a structure that commonly appears is the "differential floating-gate pair (DFGP)". The DFGP is nothing but two identical floating-gate transistors that have been programmed such that a difference in current of ΔI exists between them. The DFGP is used in circuits such that the

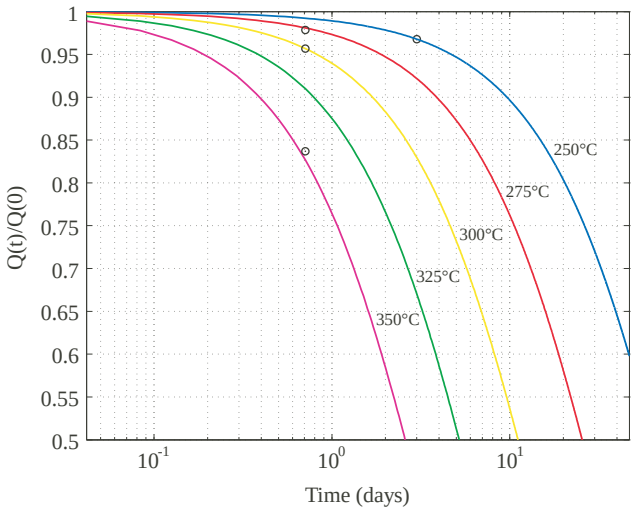


Figure 14. Charge loss in floating-gate transistors plotted versus temperature and time: Charge loss measured at different temperatures and time periods as estimated from threshold voltage changes is plotted using ∘'s. Parameters for a thermionic emission model were extracted using the measured data and the model is then used to calculate charge loss at different temperatures and time periods. This extrapolated theoretical fit is plotted using solid lines.

difference current ΔI plays a key role in influencing the circuit's performance. Therefore, it is worthwhile to evaluate charge retention in such a structure.

Consider the DFGP and assume weak inversion operation. Note that the analysis is similar to that in section 3.4 and so the difference in charge between the two floating-gate transistors is given by,

$$\Delta Q = C_T \frac{U_T}{\kappa} ln\left(1 + \frac{\Delta I}{I}\right) = C_T \Delta V_{fg} \tag{51}$$

where, all the variables have their usual meaning. Now, using (46) and the extracted values of ϕ_B and υ, the difference in charge at time t, namely, $\Delta Q(t)$ can be estimated. From this, the difference in floating-gate voltage can be calculated, based on which and using (51), the value of the programmed difference current at time t ($\Delta I(t)$) can be estimated. Table 2 summarizes the data retention numbers for the $0.5\mu m$ process for two different cases of programmed difference currents, namely, a 10% change and a 50% change for a time period of 10 yrs for different temperatures. A total floating-gate capacitance of $100fF$ and a κ of 0.7 has been assumed for these calculations.

Table 2. Summary of floating-gate parameter change in 10 years

	10% Programmed Change			**50% Programmed Change**		
Temperature	$\Delta Q/Q$	ΔV_{fg}	$\Delta I/I$	$\Delta Q/Q$	ΔV_{fg}	$\Delta I/I$
$25°C$	$10^{-3}\%$	$36.7nV$	$2\times10^{-4}\%$	$10^{-3}\%$	$156nV$	$9\times10^{-4}\%$
$90°C$	0.62%	$16.4\mu V$	0.06%	0.62%	$65\mu V$	0.47%
$140°C$	18.2%	$0.45mV$	1.8%	18.2%	$1.92mV$	10.7%

As can be observed from Table 2, the percentage change in charge over time at different temperatures is the same, irrespective of the programmed current difference between devices. However, the change in the floating-gate voltage with time for different temperatures depends on the programmed current difference. This is on account of the fact that larger the current difference, larger is the difference in charge and so larger is the absolute change in the charge with time for a given temperature. This results in the change in the floating-gate voltage being larger for a larger current difference. For the same reasons, the percentage change in programmed currents is larger for the case of the 50% programmed current difference as against the 10% programmed current difference.

3.6 Floating-Gate Transistors in Analog Circuitry

As has been demonstrated, floating-gate transistors provide programmability along with non-volatile memory capability. These features can be exploited to enhance performance in traditional analog circuits and to develop novel circuits built around the framework of programmability. In order to effectively use floating-gate transistors as part of a larger system, it is important that there is a way to individually isolate each floating-gate device such that it can be programmed and that each floating-gate device be designed such that the achievable programming precision is sufficient to meet the specifications of the system that the floating-gate transistor is a part of.

Figure 15 shows the use of floating-gate transistors as part of analog circuitry.

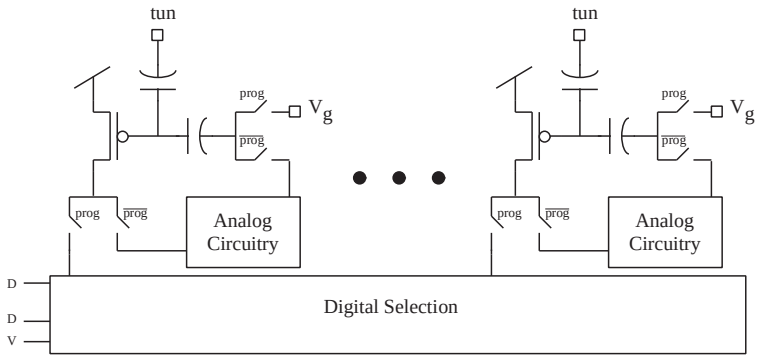


Figure 15. Floating-gate transistors in Analog Circuitry: The use of multiple floating-gate transistors as part of analog circuitry is shown. Applying a digital *High* to *prog* switches the floating-gate transistors into program mode. The floating-gate transistor of interest is then selected using the digital selection circuitry.

During normal operation, a digital *Low* is applied to *prog* thereby switching the floating-gate transistors into the circuit of interest. The operating V_{DD} is $3.3V$ during normal operation for a $0.5\mu m$ process. Programming is achieved by first isolating the floating-gate transistor from the rest of the circuitry such that one can access the gate and drain terminals of the device. This is achieved by applying a digital *High* to the *prog* terminal. The drain of the floating-gate transistor of interest that needs to be programmed is then switched to the external drain terminal V_d using the digital selection circuitry shown. The drains of the unselected devices are tied to V_{DD}. It should be noted that all floating-gate transistors share the same gate terminal during program mode. The tunneling terminal is shared amongst all floating-gate devices as well.

3.7 Summary

Floating-gate transistors are similar to normal transistors with the key difference being that their gate has no DC path to ground and all inputs are capacitively coupled onto the floating-gate. Floating-gate transistors can be fabricated on standard digital CMOS processes and require no special processing steps. The non-volatile charge retention when combined with programmability, makes floating-gate transistors well

suited for use in precision analog circuits and signal processing. In this work, precision programming is achieved using hot-electron injection while tunneling is used primarily for a global erase operation. Designing a floating-gate transistor to achieve a target programming precision is critical to its successful implementation as a part of a larger system. Towards this end, design equations have been developed in both the strong and weak inversion regions of operations that can be used in designing the aspect ratio and C_T of the floating-gate transistors. Charge loss in floating-gate transistors has been studied using the framework of a thermionic emission model. Model parameters have been extracted for two different processes, using which, extrapolations can be made for charge loss in floating-gate transistors. In summary, floating-gate transistors provide programmability with excellent charge retention. Exploiting these characteristics for achieving high performance in analog circuits and signal processing will be presented in the following chapters.

CHAPTER 4
PRECISION CMOS AMPLIFIER

Mismatches between MOS transistors pose a serious challenge to analog circuit designers and most commonly manifest themselves as an offset voltage in operational amplifiers. Techniques commonly used to reduce the offset voltage include auto-zeroing, correlated double sampling and chopper stabilization [6]. Auto-zeroing and correlated double sampling are techniques applicable to sampled data systems while chopper stabilization allows continuous-time operation of the amplifier. Resistor trimming using laser trims is another popular approach. This, however, is usually expensive. Another technique includes using current-mode digital-to-analog converters (DAC) to compensate for amplifier offsets by adjusing amplifier load currents [28].

In this chapter, a floating-gate based offset cancellation scheme is presented that results in a continuous-time operation of the amplifier with long-term offset cancellation that obviates the need for any refresh circuitry. A prototype amplifier has been fabricated with its offset voltage reduced to $25\mu V$. The use of floating-gate transistors for correcting mismatches in analog circuitry is particularly advantageous as it offers programmability, long-term retention and can be fabricated in a standard digital CMOS process. This approach involves no sampling and hence avoids such issues as charge injection, clock feedthrough and undersampled wideband noise that are serious limitations to autozeroing and correlated double sampling [6, 29]. Also, unlike chopper stabilization [6], the proposed scheme is not limited to low-bandwidth applications, while, at the same time offering continuous-time operation with comparable offset reduction.

The proposed scheme involves using floating-gate transistors as both an integral part of the circuit of interest and as programmable elements. Figure 16 shows a conceptual representation of the proposed scheme applied towards offset cancellation in

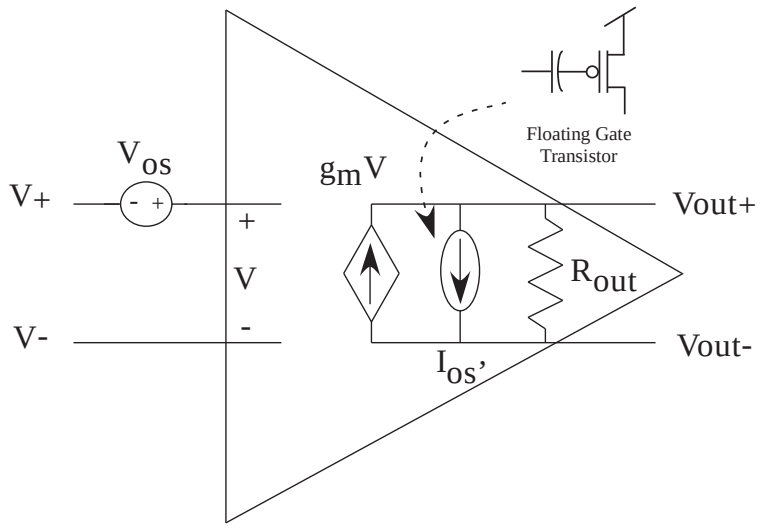


Figure 16. Offset Cancellation Macromodel: The offset voltage of the amplifier V_{os} is cancelled by programming an offset current $I_{os'}$ in the opposite direction on floating-gate transistors.

an operational amplifier. Floating-gate transistors are used as programmable current sources ($I_{os'}$) that provide offset compensation while being a part of the amplifier of interest during normal operation. Such an approach results in a compact architecture with a simple design strategy that avoids the overhead of using floating-gate transistors as separate trimming elements as in [22, 23] or current-mode DACs as trimming elements [28]. Also, the proposed offset cancellation scheme is independent of other amplifier parameters unlike other approaches [6, 30] and the offset cancellation by itself dissipates no additional power.

4.1 Amplifier Architecture

A single stage folded cascode amplifier shown in Figure 17 demonstrates a practical implementation of the proposed approach shown pictorially in Figure 16. The currents through the floating-gate transistor pair $M3$ and $M4$ are programmed such that they cancel the offset arising from mismatches in the input differential pair ($M1$, $M2$) and the cascoded current mirrors ($M5 - M8$). During normal operation, the multiplexors $S1$ and $S2$ are set such that the floating-gate transistors are a part of the

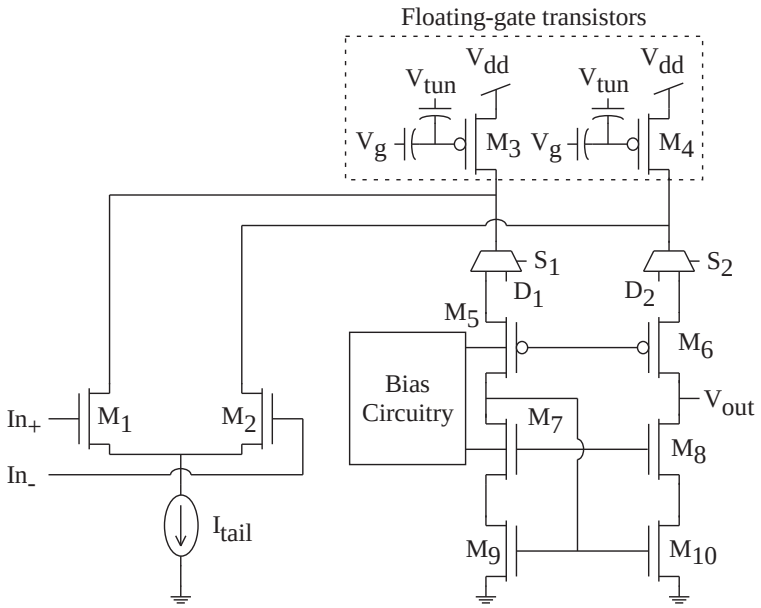


Figure 17. Operational Amplifier Circuit Schematic: A single stage folded cascode amplifier that uses floating-gate transistors as trimming elements is shown. During normal operation switches S_1 and S_2 are set such that floating-gate transistors M_3 and $M4$ are a part of the operational amplifier. Offset voltage cancellation is achieved by programming a current difference between $M3$ and $M4$. Using floating-gates both as a part of the amplifier and as trimming elements makes the architecture compact and easy to design.

operational amplifier. During programming, the floating-gate transistors are isolated from the amplifier such that the drains of the transistors are externally available. Note that both the transistors share the same tunneling voltage and the same gate voltage. During programming the gate voltage is externally available as well. Using the programming techniques described in chapter 3, the floating-gate transistors are programmed to exhibit a difference current $\Delta I(I3 - I4)$ such that the offset voltage is nullified.

A key advantage of this architecture is that the programming transistors are an integral part of the amplifier thereby simplifying the design process. Initially, all transistors including $M3$ and $M4$ are made non floating-gate transistors and are designed to meet the amplifier's specifications. Next, these transistors $M3$ and $M4$ are made floating-gate transistors and based on the offset requirement of the amplifier,

an estimate can be made of the programming precision required. In other words, an approximate value of the difference current (ΔI) that needs to be programmed can be estimated from which the FOM is calculated. Next, depending on the region of operation of the transistors $M3$ and $M4$, appropriate design equations developed in section 3.4 can be used to estimate the total floating-gate capacitance needed. With the aspect ratio of the transistors set during the amplifier's design stage, the input capacitance and the tunneling capacitance can be sized to either meet or exceed the C_T requirement. Appropriate switches are then added to isolate the floating-gate transistors during programming. For this design, the floating-gate current sources were set to be $10\mu A$ nominally and the total floating-gate capacitance was designed to be around $200fF$. From Table 1 it can be seen that a programming precision greater than 10 bits can be achieved for a charge transfer of around $100e-$ in strong inversion operation which is sufficient for the design.

4.2 Input Referred Offset Voltage

In this section, the input referred offset voltage of the amplifier is analyzed both in the small-signal domain and the large signal domain. In the large signal analysis, both weak inversion operation and strong inversion operations are considered for the sake of completeness.

4.2.1 Small Signal Analysis

The amplifier exhibits zero offset voltage when all currents are balanced at its output. Assume that the amplifier has an uncompensated offset voltage given by $V_{off}^{'}$. Let a current difference of ΔI_{fg} be programmed onto the pFET floating-gate transistors such that this difference current creates a voltage at the output equal to $\Delta I_{fg} r_o$, where, r_o represents the effective output impedance at the output of the amplifier.

The input referred offset voltage of the amplifier therefore becomes,

$$V_{off} = V_{off}{}' + \frac{\Delta I_{fg}}{g_{m1}} \tag{52}$$

where, g_{m1} is the transconductance of the input differential pair. Based on (52), one would expect the input referred offset voltage of the amplifier to exhibit a linear dependence with the programmed floating-gate difference current. Note that the above expression has been derived without assuming any specific region of device operation. By varying the polarity of the difference current ΔI_{fg} and choosing an appropriate value for $\Delta I_{fg}/g_{m1}$, the original offset voltage of the amplifier can be cancelled.

4.2.2 Large Signal Analysis - Weak Inversion

As before, the condition for zero offset is that all currents must be balanced at the output. Assume intially that all transistor pairs except $M9/M10$ are matched. Further assume that all transistors are in saturation and ignore Early effects. The current through transistor $M9$ is given by,

$$I_9 = I_o exp\Big(\frac{\kappa(V_g - V_{To})}{U_T}\Big) exp\Big(\frac{V_s}{U_T}\Big) \tag{53}$$

where, all variables have their usual meaning as defined earlier. Now assume that the threshold voltage of $M10$ is different from that of $M9$ by $-\Delta V_{th3}$, the current through $M10$ now becomes,

$$I_{10} = I_o exp\Big(\frac{\kappa(V_g - V_{To} + \Delta V_{th3})}{U_T}\Big) exp\Big(\frac{V_s}{U_T}\Big) = I_9 + \Delta I \tag{54}$$

Dividing (54) by (53) and re-arranging the terms results in,

$$\Delta I = I_9\Big[exp\Big(\frac{\kappa \Delta V_{th3}}{U_T}\Big) - 1\Big] \tag{55}$$

The current of $M10$ is larger than that of $M9$ by ΔI. The objective is to translate this mismatch to the input referred offset voltage. This can be achieved by noting

that making the current of $M2$ smaller by ΔI results in the currents at the output being balanced. In order to do this, one has to apply an offset voltage at the gate of $M2$ in the negative direction, or a voltage of $-V_{os3}$ with respect to the gate of $M1$. Therefore, the offset voltage contribution due to the mismatch in the transistor pairs $M9/M10$ is the value $-V_{os3}$ that needs to be applied to the gate of $M2$ such that the currents are balanced at the output.

Considering the currents of $M1/M2$ and assuming that the gate of $M2$ is offset from that of $M1$ by $-V_{os3}$ resulting in its current being smaller than that of $M1$ by ΔI and following the steps outlined earlier, one can express V_{os3} as,

$$V_{os3} = \frac{U_T}{\kappa} ln\Big(1 - \frac{\Delta I}{I_1}\Big) \tag{56}$$

Substituting (55) in the above expression, one gets,

$$V_{os3} = \frac{U_T}{\kappa} ln\Big(1 - \frac{I_9}{I_1}\Big[exp\Big(\frac{\kappa \Delta V_{th3}}{U_T}\Big) - 1\Big]\Big) \tag{57}$$

For typical values of threshold voltage mismatch, the term inside the exponential is less than 1. Therefore, the exponential can be expanded using a Taylor series with the second and higher order terms neglected. Also, designing such that I_9 is less than I_1, the Taylor series expansion of the natural logarithm can be performed and ignoring the higher order terms, the offset voltage due to mismatch in the $M9/M10$ transistor pair can be expressed as,

$$V_{os3} = -\frac{I_9}{I_1} \Delta V_{th3} \tag{58}$$

Next, consider the pFET floating-gate pair $M3/M4$. Assume that a difference in their floating-gate voltage $-\Delta V_{fg}$ exists between them such that the current of transistor $M4$ is greater than that of $M3$ by ΔI. It should be noted that the threshold voltage mismatches between the transistors and the differences in the programmed charge can be accounted for by ΔV_{fg}. Proceeding, as in the case of the $M9/M10$

pair, the difference current is given by,

$$\Delta I = I_3\Big[exp\Big(\frac{\kappa\Delta V_{fg}}{U_T}\Big) - 1\Big] \tag{59}$$

Now to balance this excess current, a positive offset needs to be applied such that the current of $M2$ is increased by ΔI. Performing the analysis as before, the offset voltage contribution by the floating-gate pair (ΔV_{os2}) is given by,

$$V_{os2} = \frac{U_T}{\kappa} ln\Big(1 + \frac{\Delta I}{I_1}\Big) \tag{60}$$

Substituting (59) in the above expression results in,

$$V_{os2} = \frac{U_T}{\kappa} ln\Big(1 + \frac{I_3}{I_1}\Big[exp\Big(\frac{\kappa\Delta V_{fg}}{U_T}\Big) - 1\Big]\Big) \tag{61}$$

In this case, one can still apply the Taylor series expansion of the exponential and ignore all the higher order terms. However, for typical designs, $I3$ is greater than $I1$ and so the higher order terms in the Taylor series expansion of the logarithm needs to be included as well. Therefore, including the second order term while ignoring third and higher order terms results in the offset voltage being,

$$V_{os2} = \frac{I_3}{I_1}\Delta V_{fg} - \frac{{I_3}^2}{I_1}\frac{\kappa}{U_T}\Delta V_{fg}^2 \tag{62}$$

Finally, consider mismatch in the input differential pair. Let the threshold voltage of $M2$ be higher than $M1$ by ΔV_{th1}. To compensate this mismatch, one needs to apply a differential voltage of $-\Delta V_{th1}$ at the gate of $M2$.

Thus far, the mismatch in each of the transistor pairs $M1/M2$, $M3/M4$ and $M9/M10$ has been analyzed individually with their contributions to the input referred offset voltage estimated. The input referred offset voltage collectively due to all the mismatch effects (V_{os}) can be estimated by applying superposition and noting that $V_{os} = V_{os1} + V_{os2} + V_{os3}$. The input referred offset voltage is now given by,

$$V_{off} = \Delta V_{th1} + \frac{I_3}{I_1}\Delta V_{fg} - \frac{I_9}{I_1}\Delta V_{th3} - \Big(\frac{I_3}{I_1}\Big)^2\frac{\kappa}{U_T}\Delta V_{fg}^2 \tag{63}$$

Although in arriving at the above expression, specific signs have been assumed for the threshold voltage offsets, there has been no loss in generality. The signs of the various terms in the expression can be modified accordingly to take into account the actual signs of the mismatch terms.

4.2.3 Large Signal Analysis - Strong Inversion

Consider saturation in strong inversion and ignore Early effects. The drain current through the transistor $M9$ is given by,

$$I_9 = \frac{\mu_n C_{ox} W}{2\kappa L}\Big[\kappa(V_g - V_{To}) - V_s\Big]^2 \tag{64}$$

where, all variables have their usual meaning. As before, assume a threshold voltage difference of ΔV_{th3} between the transistors $M9/M10$ such that the current of $M10$ is higher than that of $M9$ by ΔI. Taking into consideration the threshold voltage mismatch, the drain current of $M10$ can be written as,

$$I_{10} = I_9 + \Delta I = \frac{\mu_n C_{ox} W}{2\kappa L}[\kappa(V_g - V_{To}) - V_s]^2\Big[1 + \frac{\kappa\Delta V_{th3}}{\kappa(V_g - V_{To}) - V_s}\Big]^2 \tag{65}$$

For typical designs, the value of the over-drive voltage $(\kappa(V_g - V_{To}) - V_s)$ is on the order of $100-200mV$. This results in $\kappa\Delta V_{th3}/(\kappa(V_g - V_{To}) - V_s)$ being much less than 1 for typical values of threshold voltage mismatch. Therefore, performing a Taylor series expansion and ignoring higher order terms, the drain current of $M10$ can be rewritten as,

$$I_{10} = I_9\Big[1 + \frac{2\kappa\Delta V_{th}}{\kappa(V_g - V_{To}) - V_s}\Big] \tag{66}$$

Dividing, (66) by (64) and re-writing the over-drive voltage in terms of the drain current, the difference in currents ΔI becomes,

$$\Delta I = \sqrt{2\kappa I_9 \beta_9}\Delta V_{th3} \tag{67}$$

Now, in order to balance the currents at the output, an offset voltage of $-V_{os3}$ needs to be applied to the gate of $M2$ such that the current of $M2$ is smaller than that of

$M1$ by ΔI. Expressing this difference current in terms of V_{os3} and $M1/M2$ transistor parameters results in,

$$\Delta I = -\sqrt{2\kappa I_1 \beta_1} V_{os3} \tag{68}$$

As before, defining the input referred offset voltage contribution due to mismatch in the transistor pairs $M9/M10$ to be the voltage that needs to be applied to the gate of $M2$ in order to balance the currents at the output, it is clear that the difference current in the above expression has to be equal to that in (67). Using the above relationship, V_{os3} is given by,

$$V_{os3} = -\sqrt{\frac{I_9 \beta_9}{I_1 \beta_1}} \Delta V_{th3} \tag{69}$$

Similarly, the offset voltage contribution due to the floating-gates of $M3/M4$ being different by $-\Delta V_{fg}$ such that the current of $M4$ is greater than that of $M3$ by ΔI is given by,

$$V_{os} = \sqrt{\frac{I_3 \beta_3}{I_1 \beta_1}} \Delta V_{fg} \tag{70}$$

Now, as before, assume that the threshold voltage of $M2$ is higher than that of $M1$ by ΔV_{th1}. This is compensated by assuming that a differential voltage of $-\Delta V_{th1}$ is applied to the gate of $M2$. Performing superposition and taking into consideration the mismatch in all the transistor pairs comprising the amplifier, the offset voltage for operation in strong inversion becomes,

$$V_{off} = \Delta V_{th1} + \sqrt{\frac{I_3 \beta_3}{I_1 \beta_1}} \Delta V_{fg} - \sqrt{\frac{I_9 \beta_9}{I_1 \beta_1}} \Delta V_{th3} \tag{71}$$

Note that both equations (63) and (71) simplifies to (52) when the appropriate values for ΔV_{fg} is expressed as $\Delta I_{fg}/g_{m3}$. It is clear from the above equations that the offset voltage of the amplifer can be reduced to a very small value by appropriately programming a difference current between the floating-gate transistors. This can be accomplished by programming a difference between their floating-gate voltages. Also, the above technique is well suited for applications that demand a particular offset voltage, such as comparators used in a flash analog-to-digial converter (ADC)[31].

In the large-signal analysis presented thus far, the effect of mismatch in the cascode transistors $M7/M8$ and $M11/M12$ on the offset voltage has been neglected. A threshold voltage mismatch in the cascode transistors results in a difference in the drain voltage of the current mirror transistors. This difference in the drain voltages leads to a difference in the current on account of the Early effect. Also, the finite output impedance of the cascode current mirrors leads to an offset voltage as well. For well designed circuits, these errors can be minimized and the offsets arising out of these errors are usually lesser than those arising out mismatches between the transistor pairs considered earlier. Also, to a first approximation, these errors can be lumped along with the threshold mismatch in the transistor pairs $M3/M4$ and $M9/M10$, if needed.

4.3 Temperature Sensitivity of Input Referred Offset Voltage

Analyzing the temperature sensitivity of the input referred offset voltage is important in order to gain insights into the parameters that influence the temperature behavior of the offset voltage. Gaining such insights is critical in ensuring that the offset voltage is fairly independent of temperature. Observing (63) and (71), it is clear that the temperature sensitivity of the offset voltage can be estimated based on the sensitivities of the threshold voltage mismatch and ratios of transistor currents and β's. Note that ΔV_{fg} is temperature independent, as for a typical operating temperature range, the charge loss on the floating-gate is negligible and therefore assumed constant, and to a first-order, the total floating-gate capacitance is independent of temperature as well.

The temperature dependence of the threshold voltage is given by [11],

$$V_{th}(T) = V_{th}(T_o) + \alpha(T - T_o) \tag{72}$$

where, T is the temperature in kelvin, $V_{th}(T_o)$ represents the threshold voltage at a temperature T_o and α represents the linear temperature co-efficient of the threshold voltage. Now, the temperature dependence of the threshold mismatch between two

devices can be written as,

$$\Delta V_{th} = \Delta V_{th}(T_o) + \Delta\alpha(T - T_o) \tag{73}$$

where, $\Delta V_{th}(T_o)$ represents the threshold mismatch at temperature T_o and $\Delta\alpha$ is the difference in their temperature co-efficients.

The temperature dependence of κ has been analyzed by incorporating the temperature dependence of the terms describing κ and simulating using MATLAB. Assuming an n-channel transistor with a threshold voltage of $0.7V$ with a temperature co-efficient of $-2mV/°C$, a substrate doping of $1\times10^{17}cm^{-3}$, a γ of 0.5 and a gate-bulk voltage of $1V$ results in a κ of 0.8049 at room temperature ($300K$). The variation of κ with temperature over a range of $-40°C$ to $140°C$ was found to be $\approx 27ppm/°C$. Therefore, it was decided to assume κ to be constant with temperature to simplify the temperature analysis of the amplifier offset voltage.

4.3.1 Temperature Sensitivity - Strong Inversion

The expression for the input offset voltage, (71) contains a threshold voltage difference term and a $\sqrt{I\beta}$ term. The temperature dependence of the threshold voltage difference term is as given in (73). Next, consider the term $\sqrt{I\beta}$ that appears in the expression for the input offset voltage. This term can be rewritten as,

$$\sqrt{2I\beta} = \mu_n C_{ox}\frac{W}{L}(\kappa(V_g - V_{To}) - V_s) = g_m \tag{74}$$

where, all the terms are as defined earlier. Assuming fixed terminal voltages, the only terms that have a temperature dependence in the above equation are the threshold voltage and mobility.

The temperature dependence of mobility is modeled as [11],

$$\mu_n(T) = \mu_n(T_o)(T/T_o)^{\alpha_n} \tag{75}$$

where, $\mu_n(T)$ is the mobility of electrons at a temperature T, $\mu_n(T_o)$ is the mobility at a reference temperature T_o and α_n is the temperature co-efficient of mobility. The

expression for the hole mobility is similar to the above expression with the difference being that the temperature co-efficient is modeled differently by α_p. For doping levels less than $10^{12}cm^{-3}$, α_n equals -2.42 while that of α_p is -2.2. However, for doping levels above $10^{17}cm^{-3}$, the value of α_n becomes equal to -1.2 while that of α_p equals -1.9 [32]. Therefore, the ratio of mobilities for two similar type devices is independent of temperature while that of two dissimilar type devices is slightly temperature dependent. However, for ease of analysis, the temperature co-efficients of electron and hole mobilities will be assumed equal.

With the above observations, and (73), the third term in (71) can be written as,

$$\sqrt{\frac{I_9\beta_9(T)}{I_1\beta_1(T)}} = \Big(\frac{\beta_9(T_o)\kappa V_{od9}(T_o)}{\beta_1(T_o)\kappa V_{od1}(T_o)}\Big)\Big(\frac{1-\frac{\kappa\alpha_9\Delta T}{V_{od9}(T_o)}}{1-\frac{\kappa\alpha_1\Delta T}{V_{od1}(T_o)}}\Big) \tag{76}$$

where, $\Delta T = T - T_o$. A similar expression can be arrived at for the second term in (71). Denoting $\kappa\alpha_1/V_{od1}(T_o)$ as a, $\kappa\alpha_9/V_{od9}(T_o)$ as b, $\kappa\alpha_3/V_{od3}(T_o)$ as c and using (73) in (71) results in,

$$\begin{aligned} V_{off}(T) = \Delta V_{th1} + \Delta\alpha_1\Delta T + \tfrac{gm_3(T_o)}{gm_1(T_o)}\tfrac{(1-b\Delta T)}{(1-a\Delta T)}\Delta V_g \\ -\tfrac{gm_9(T_o)}{gm_1(T_o)}\tfrac{(1+c\Delta T)}{(1-a\Delta T)}(\Delta V_{th3} + \Delta\alpha_3\Delta T) \end{aligned} \tag{77}$$

As can be observed from (77), the offset voltage varies with the temperature and the variation can be approximated to be quadratic in nature. Also, it is clear that the offset voltage depends on threshold voltage mismatch multiplied by a ratio of quantities (transconductance). Since, the threshold voltage mismatch by itself has a weak temperature dependence, designing the ratio of transconductances to be fairly temperature independent can result in an overall offset voltage that is temperature independent. This can be achieved by either biasing the transistors to their zero-temperature co-efficient transconductances [33, 34] or by designing such that their overdrive voltages are close to each other making the terms a,b and c equal such that temperature sensitivity is minimized.

4.3.2 Temperature Sensitivity - Weak Inversion

Consider the expression for input offset voltage in weak inversion given by (63). Apart from the difference in threshold voltage terms and the difference in the floating-gate voltage, the term to analyze in terms of temperature dependence is the ratio of currents term. Consider the drain current of an nFET in weak inversion as given below,

$$I = I_o exp\Big(\frac{\kappa(V_g - V_{To})}{U_T}\Big) exp\Big(\frac{V_s}{U_T}\Big) \tag{78}$$

The pre-exponential constant, I_o is given by [11],

$$I_o^{'} = \Big(\frac{1-\kappa}{\kappa}\Big)\mu_n C_{ox}\frac{W}{L}{U_T}^2 \exp\Big(\frac{\psi_o - 2\phi_F}{U_T}\Big) \tag{79}$$

where, ϕ_F is the Fermi potential of the bulk and $\psi_o = 2\phi_F + n{\cdot}U_T$.

The pre-exponential constant I_o is temperature dependent on account of the temperature dependence of the physical constants that are present in its expression. However, in this case, one is interested in the ratio of two pre-exponential constants. Assume κ is independent of temperature and consider two similar type devices (2 nFETs or 2 pFETs). It is clear from (79) that the ratio of pre-exponential constants will be independent of temperature. Now consider two dissimilar devices, an nFET and a pFET. As before, κ is assumed independent of temperature and the temperature co-efficient of electron and hole mobilities are treated to be the same. The bulk Fermi potential for the n-type and p-type devices will be different. However, approximating the ψ_o term to be equal to the sum of twice the Fermi potential and several thermal voltages [11], the ratio of the pre-exponential constants becomes temperature independent to a first approximation.

Consider the drain current of an nFET in weak inversion saturation and ignore Early effects. The drain current is given by,

$$I = I_o exp\Big(\frac{\kappa(V_g - V_{To})}{U_T}\Big) exp\Big(\frac{-V_s}{U_T}\Big) \tag{80}$$

Using the temperature dependence of the threshold voltage and expanding the thermal voltage around a reference temperature T, the above expression can be re-written as,

$$I = I_o exp\Big(\frac{\kappa(V_g - V_{To}(To)) - V_s}{U_{To}}\Big) exp\Big(\frac{\Delta T}{U_{To}To}(-\kappa(V_g - V_{To}(To)) + V_s + \alpha(T_o + \Delta T)\Big) \tag{81}$$

In the above expression, the pre-exponential constant is temperature dependent and so is the last exponential term. The exponential term in the middle consists of quanitities that are temperature independent. Here it has been assumed that the terminal voltages are independent of temperature.

Next, consider the term, I_9/I_1. As mentioned before, the ratio of pre-exponential constants is independent of temperature. Also, observing (81), the second term is independent of temperature and so is its ratio. With these observations, the ratio of current I_9/I_1 can be written as,

$$\frac{I_9(T)}{I_1(T)} = k(To) \cdot exp\Big(\frac{\Delta T}{U_{To}(To)}(-\kappa(\Delta V_g - \Delta V_{To}(To)) + \Delta V_s + \Delta\alpha(T_o + \Delta T))\Big) \tag{82}$$

where, $\Delta V_g = V_{g9} - V_{g1}$, $\Delta V_{To} = V_{To9} - V_{To1}$, $\Delta V_s = V_{s9} - V_{s1}$ and $k(To)$ represents all the temperature independent terms. A similar expression can be derived for the current ratio term $I_3(T)/I_1(T)$. Substituting these into (63) results in a complete expression for the first-order temperature dependence of the offset voltage in weak inversion.

It should be noted that in the above discussions on temperature sensitivity, κ was assumed to be independent of temperature. This assumption is valid for certain operating conditions. Also, the terminal voltages such as gate voltage, source voltage etc. were assumed to be temperature independent as well. These assumptions were made in order to simplify the analytical derivations. In a real circuit, the terminal voltages of various devices are usually set by some biasing circuitry and therefore exhibit changes with temperatures. Owing to these, the above expressions for temperature

sensitivity should be treated as a initial starting point for design. A complete circuit simulation is required to estimate the temperature sensitivity more accurately.

4.4 Offset Voltage Measurement

It is evident that the parameter of utmost importance in this chapter is the input referred offset voltage. It is therefore critical that the test methodology used for measuring the input offset voltage be described. In this section, a technique that is well suited for accurately measuring very small offset voltages is discussed in detail. Accurate measurements of the offset voltage is made by using the amplifier under test along with a second amplifier configured as a nulling amplifier forming a servo loop [35]. Figure 18 shows the schematic of the test setup with A_{DUT} being the amplifer under test and A_{NULL} being the nulling amplifier. Assume that the nulling amplifier has an offset voltage of $V_{os,null}$. The capacitor C is used for stabilizing the measurement setup and will be ignored for DC considerations. Applying Kirchoff's current law (KCL),

$$\frac{V_{o,dut} - (V_{os,null} + V_{mid})}{R_3} = \frac{(V_{os,null} + V_{mid}) - V_1}{R_3} \tag{83}$$

Solving the above expression for $V_{o,dut}$ results in,

$$V_{o,dut} = 2(V_{os,null} + V_{mid}) - V_1 \tag{84}$$

Owing to negative feedback, the output of the nulling amplifier is such that the differential input of the A_{DUT} is at a value that results in the output of the amplifier being equal to V_{od} as given above. Since the amplifier under test is a CMOS amplifier, no current flows into its input and so the negative terminal is equal to V_{mid}. Now in order to determine the voltage at the positive terminal of the amplifier, one needs to equate the currents through resistors R_1 and R_2. Equating the currents and making algebraic manipulations results in,

$$V_{in}^{+} = \frac{R_1 V_{on} + R_2 V_{mid}}{R_1 + R_2} \tag{85}$$

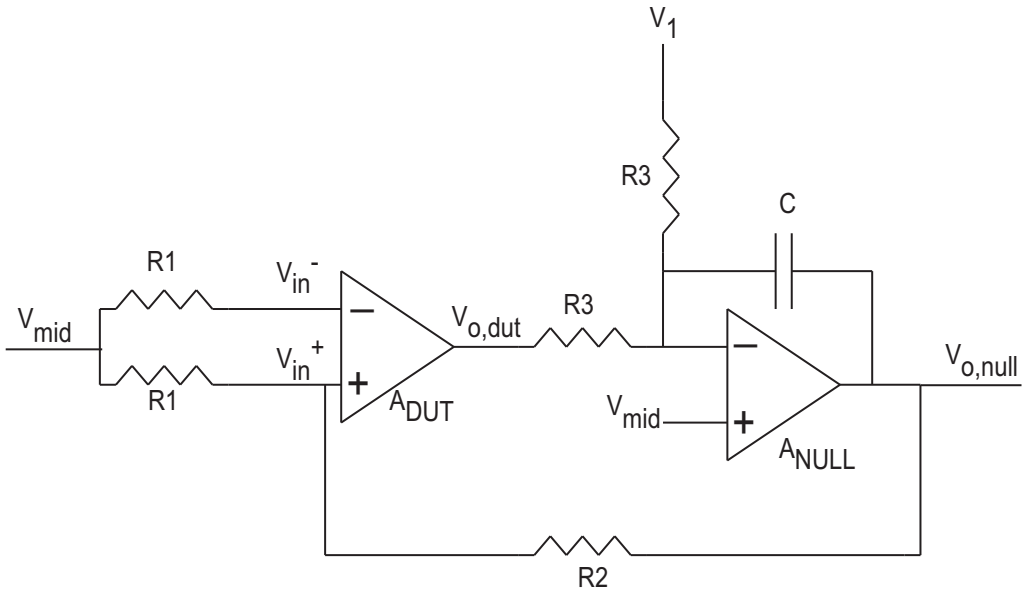


Figure 18. Test Setup For Measuring Input Offset Voltage: The test setup for measuring the input offset voltage of the amplifier (A_{DUT}) using a nulling amplifier (A_{NULL}) is shown.

Using the above expression along with the negative terminal being at V_{mid}, the differential input of A_{DUT} is given by,

$$V_{in}^{+} - V_{in}^{-} = \frac{R_1}{R_1 + R_2}\Big[V_{on} - V_{mid}\Big] = V_{in,d} \tag{86}$$

In dual-supply amplifiers, the input offset voltage is the amount of differential voltage that must be applied at the inputs of the amplifier in order to make the output equal to zero volts. Since in this case, the amplifier operates on a single supply voltage ($3.3V$), the offset voltage is defined to be equal to the differential voltage that results in the output of the amplifier equalling mid-supply voltage or V_{mid}. Now, when V_1 is set to V_{mid}, the output of A_{DUT} goes to V_{mid} as the nulling amplifier forces the output of A_{DUT} to equal V_{mid} by servoing its input differential voltage to the appropriate value. As per the definition for input offset voltage, this value equals the offset voltage of the amplifier. Therefore, the input offset voltage can be measured by setting V_1 equal to V_{mid} and using (86).

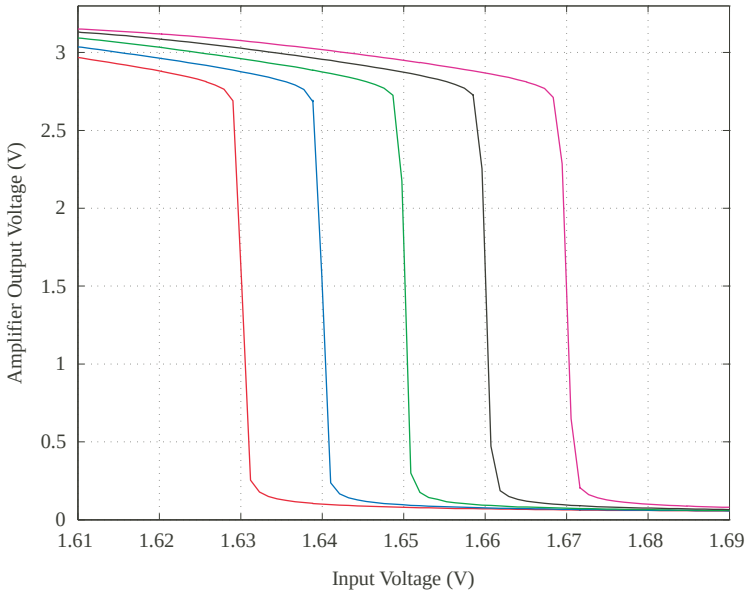


Figure 19. Open Loop DC Transfer Characteristics: The input offset voltage of the amplifier was programmed to five different values in steps of $10mV$. The non-inverting terminal of the amplifier was set at $1.65V$ and the inverting terminal was swept from $0 - 3.3V$. The DC transfer curves show the switching points ranging from $-20mV$ to $+20mV$ with a $10mV$ spacing as programmed.

4.5 Amplifier Experimental Results

In this section, measured results from a prototype amplifier designed and fabricated in a $0.5\mu m$ CMOS process is presented. The amplifier was designed to operate in the strong inversion region and was tested with a $3.3V$ power supply.

Coarse measurements of the offset voltage can be performed by configuring the amplifier as an open-loop comparator and measuring the switching point of the device. Using such a setup, the drain currents of transistors $M3$ and $M4$ were programmed to result in five different offset voltages for the amplifier. The offsets were programmed in steps of $10mV$ ranging from $-20mV - +20mV$. Figure 19 shows the DC transfer characteristics of the amplifier configured as a comparator with the non-inverting terminal held at $1.65V$. As can be observed the comparator trip points are evenly spaced $10mV$ apart as programmed, clearly demonstrating the feasibility of the approach and the range of programming that is possible.

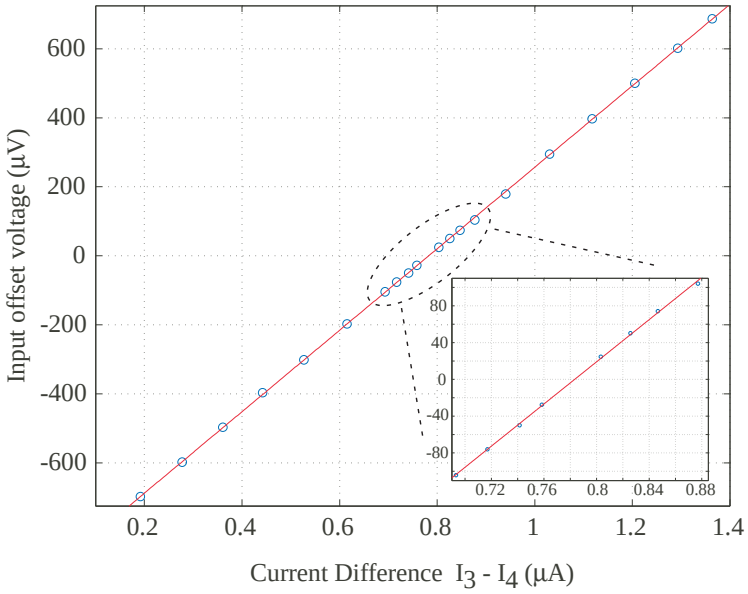


Figure 20. Input Offset Voltage vs. Floating-gate Difference Current: The input offset voltage of the amplifier was measured by programming different current differences between the floating-gate trimming transistors. The input offset voltage changes linearly with the difference current as expected from theory. The inset zooms into the region of very low offset voltages. It is clear from the inset that offset voltages in the $10's$ of micro-volts are achievable with the lowest being $25\mu V$.

Figure 20 shows the measured input referred offset voltage of the amplifier plotted against the various programmed floating-gate difference currents using the technique for measuring offset voltages as outlined above. The measured data shows a linear dependence of the offset voltage with the programmed difference currents as expected from (52). As can be observed in the inset in Figure 20, the offset voltage of the prototype amplifier has been reduced to $25\mu V$. Also, it can be seen that the amplifier can be programmed to display different offset voltages with both positive and negative polarities. This clearly demonstrates the programmable nature of the approach, a feature that could be exploited when designing, for instance, comparators. Experimentally, it is possible to program current increments as low as $0.1nA$. Theoretically, this indicates that offset voltages in the $100's$ of nano-volts range are possible to achieve. At present however, the primary limitation has been the internal noise of the amplifier itself.

Figure 21 shows the sensitivity of the input offset voltage with temperature. The offset voltage was measured for temperatures ranging from $-40\ ^{\circ}C$ to $130\ ^{\circ}C$ after programming at $25^{\circ}C$. A maximum change of $130\mu V$ was observed over the full temperature range of $170\ ^{\circ}C$. Since, the transistors in the amplifier were biased in a region close to strong inversion, the temperature dependence was modeled according to (77). Shown in the figure is a theoretical fit of the data using (77). Since, the exact values of the threshold voltage mismatch of the various transistor pairs are unknown, the fit was performed using a reasonable set of parameter values. It should be noted that the exact shape of the temperature characteristic depends on the transistor operating regions, biasing conditions and the mismatch between threshold voltages.

The offset voltage drift with time can be estimated from the charge retention experiments conducted on floating-gate transistors. Knowing the drift in the stored charge of a floating-gate device, the drift in the difference current can be estimated, based on which the offset voltage drift can be calculated. For a $25\mu V$ offset voltage, the drift has been calculated to be approximately less than $0.5\mu V$ in 10years at a storage temperature of $55^{\circ}C$. Table 3 summarizes the performance of the amplifier and the chip micrograph is shown in Figure 22. The total area of the amplifier excluding the buffer is $115\mu m \times 45\mu m$ and the additional area occupied by the input capacitors and the switches on account of using floating-gate transistors is $45\mu m \times 45\mu m$. As can be observed, using floating-gate transistors as a part of the amplifier and also as a programming element leads to a compact architecture. Also, the proposed cancellation scheme is independent of other amplifier parameters.

Automatic programming of the floating-gate transistor makes the approach attractive from a commercial standpoint. Unlike wafer trimming that is susceptible to offset drifts on account of packaging stress, the proposed scheme involves offset cancellation at the package level. Extra pins (V_{tun}, V_g, V_{drain}) and digital pins for

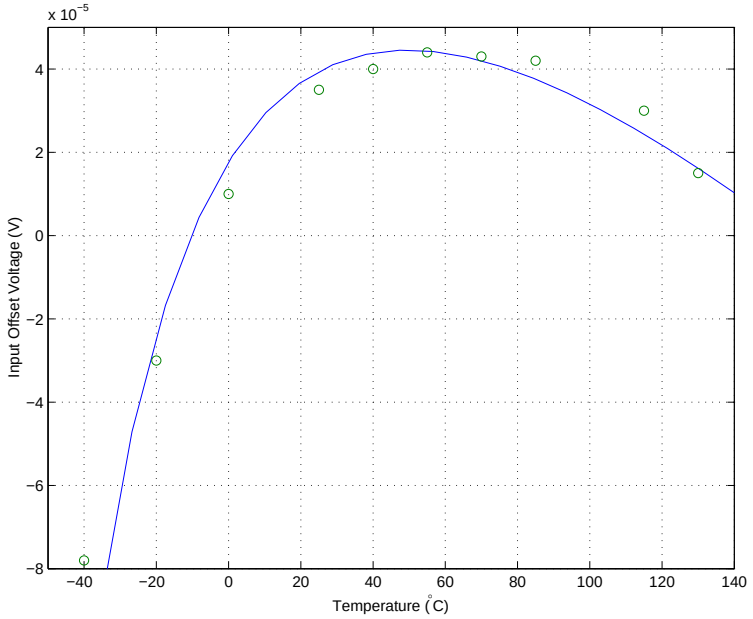


Figure 21. Input offset voltage vs. Temperature: The input offset voltage of the amplifier was measured across a temperature range of $-40\ ^{\circ}C$ to $130\ ^{\circ}C$. The offset voltage displayed a maximum change of $130\mu V$ across the entire temperature range. The $\circ$'s represent the measured data points while the solid line represents the theoretical fit based on (77)

the drain selection circuitry are needed for programming multiple floating-gate transistors. The programming infrastructure allows the gate, drain and tunnel voltages to be shared amongst different floating-gate transistors. This keeps the extra pins required constant even when using multiple floating-gate transistors, a scenario that is typical while using multiple amplifiers on the same chip. The pin count can be reduced if the gate voltage is supplied by the biasing structure during programming as well and by using a serial digital interface for the digital pins.

4.6 Comparisons to Alternate Techniques

The use of floating-gate transistors to correct for mismatch in analog circuitry has been investigated by other authors as well [22, 23]. The approach in [22] results in an uni-directional offset cancellation. This requires an intentional offset creation of the correct polarity during the design phase of the amplifier for proper operation. This

Table 3. Operational Amplifier Summary of Performance

Parameter	Value
Supply Voltage	$3.3V$
Technology	$0.5\mu m$ CMOS
Input Common Mode Range	$1.2V - 3.1V$
Output Voltage Swing	$0.2V - 3.1V$
Input Offset Voltage	$\pm 25\mu V$
Offset Voltage Drift with Temperature	$130\mu V/170°C$
Offset Voltage Drift @ $55°C$ for 10 yrs	$< 0.5\mu V$
Open Loop Gain	$63dB$
Unity Gain Bandwidth @ $C_L = 20pF$	$10MHz$
Phase Margin	$60°$
Common Mode Rejection Ratio	$73dB$ (Simulation)
Power Supply Rejection Ratio	$77dB$ (Simulation)
Input Referred Noise (rms)	$8.9\mu V$ (Simulation)
Slew Rate	$5V/\mu s$
Settling Time (10 Bit) for $100mV$ Step	$105ns$
Power Dissipation (Incl. Buffer)	$8.25mW$
Area (Excl. Buffer)	$115\mu m \times 45\mu m$

intentional offset creation has been cited as the reason for the degradation of the offset voltage temperature sensitivity [22]. The work in [23] introduces a trimming circuitry based on floating-gate transistors to produce a difference current which is then used as a building block to compensate for mismatch induced errors. The proposed approach in this paper is conceptually similar to that in [23] in that it uses a differential current to trim offsets. However, the difference current is created using just two floating-gate transistors which then form an integral part of the amplifier of interest. This results in an advantage in terms of both area and design overhead. Also, the proposed approach uses hot-electron injection to program floating-gate transistors while both [22] and [23] use Fowler-Nordheim tunneling as the primary programming mechanism. The advantages of an injection based programming scheme over a tunneling based programming has been highlighted earlier in section 3.3.

Table 4. Comparison of Offset Cancellation Schemes

	FGate	**Autozero**	**Chopper**	**Ping-Pong**	**R Trimming**	**DAC**
Mode	Continuous	Sampled	Continuous	Continuous	Continuous	Continuous
Offset (V_{os})	Low	Moderate	Low	Moderate	Low	Low
Bandwidth	High	High	Low	High	High	High
Complexity	Low	Moderate	High	Moderate	Moderate	Moderate
$1/f$ Noise	No effect	Reduced	Reduced	Reduced	No effect	No effect
Extra Power	Low	Moderate	Moderate	Moderate	Low	Moderate
Extra Area	Low	Moderate	Moderate	Moderate	Moderate	High
V_{os} Removal	Long-Term	Periodic	Continuous	Periodic	Long-Term	Long-Term
Field Programmability	Yes	No	No	No	No	Yes

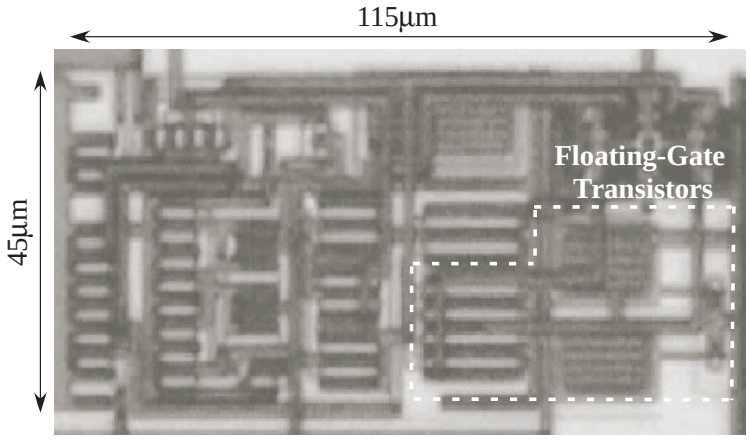


Figure 22. Amplifier die micrograph: The chip micrograph of the prototype operational amplifier excluding the output buffer is shown to occupy an area of $115\mu m \times 45\mu m$. The additional area on account of using floating-gate transistors is $45\mu m \times 45\mu m$.

Correcting analog circuit mismatch using resistor trimming is an alternate technique. Resistor trimming is usually performed using laser annealing, laser trims, poly fuses and zener zapping. Both laser annealing and laser trims are expensive and do not provide the flexibility of in-package trims. Trimming using poly fuses and zener zapping is discrete in nature and therefore accuracy is limited to the smallest resistor step used. Also, using a number of zener diodes and poly fuses involves an area penalty. All of the above resistor trimming techniques are one-time programmable. The approach described in this work is cost effective, field programmable and is a package level correction scheme.

The proposed approach involves lesser design overhead when compared to the technique of using current-mode digital-to-analog converters controlled using an EEPROM and serial interface [28, 34] to reduce amplifier offsets. Also, the proposed approach can provide a continuous range of offset voltages rather than discrete values offered by the DAC based scheme. This makes the approach well suited for other applications as well, such as, programming a chain of comparators to different trip points for use in say Flash analog-to-digital converters.

Auto-zeroing is primarily useful for sampled data systems and is limited by issues such as charge injection, clock feedthrough and wideband noise folding into the baseband on account of undersampling. For a continuous-time operation, chopper stabilization or continuous-time auto-zeroing such as a ping-pong amplifier [36] are the typical alternatives. The chopper amplifier is, however, limited in use to low-bandwidth applications [6]. The ping-pong approach involves the use of multiple amplifiers and multi-phase clocks that add additional overhead in terms of area and power. The proposed floating-gate approach involves none of the above tradeoffs and the offset cancellation by itself dissipates no additional power. The approach places minimal overhead on the amplifier design with non-volatile storage of offset reduction information. The primary limitation, however, is the lack of flicker noise reduction. Finally, Table 4 summarizes qualitatively the design tradeoffs of the proposed approach to the various offset cancellation schemes on the different design parameters of interest.

4.7 Summary

An amplifier topology has been presented that uses floating-gate programmable elements as an integral part of the actual amplifier. The approach places minimal overhead on the amplifier design with non-volatile storage of offset reduction information. A prototype amplifier has been fabricated in a $0.5\mu m$ standard digital CMOS process and trimmed to an offset voltage of $25\mu V$. The offset voltage exhibits a temperature sensitivity of $130\mu V$ over a temperature range of 170 $^{\circ}C$. Floating-gate transistors being surrounded completely by SiO_2, a high quality insulator, exhibit excellent charge retention capabilities. Accelerated life-time testing indicate an offset voltage drift of less than $0.5\mu V$ when stored at a temperature of $55^{\circ}C$ for 10 years. Direct tunneling through the gate oxide (gate leakage) is a limitation for charge retention in floating-gate transistors for oxide thicknesses less than $5nm$ that is typical

for finer line processes ($< 0.25\mu m$). However, the proposed approach is still scalable with process technologies. Floating-gate transistors have not been used in the signal path, therefore, in smaller dimension processes, floating-gate transistors can be implemented using the available thick oxide transistors with no impact on the speed of operation of the amplifier. Using thicker oxides preserves the charge retention capability of floating-gate devices thereby providing low long-term drifts in the amplifier offset voltage. Finally, programmability coupled with a negligible long-term drift and scalability makes this approach attractive for offset reduction in operational amplifiers.

CHAPTER 5
CMOS REFERENCE

Voltage references are critical components in both analog and digital systems. The accuracy, temperature sensitivity and drift of references impact the performance of many circuit blocks such as analog-to-digital converters, digital-to-analog converters and power management circuitry. With the recent trends in transistor scaling, the need for sub-1V reference voltages with low temperature sensitivity and high initial accuracy is growing. This need is addressed in this work by way of a temperature stable programmable voltage reference that uses floating-gate transistors to set the reference voltage.

In CMOS technology, the bandgap voltage reference [37] implemented using parasitic bipolar junction transistors (BJTs) is the popular choice for implementing a voltage reference. The bandgap reference provides a stable known reference voltage, namely, the energy bandgap of silicon. Typically, the bandgap reference is designed to achieve a first-order temperature cancellation that gives a zero temperature coefficient at a particular temperature. Mismatch between design components are corrected using a post-fabrication trim procedure while higher order temperature effects are reduced for by using schemes such as curvature correction [38]. Although the bandgap reference is attractive and provides temperature coefficients in the range of $25 - 50$ $ppm/^{\circ}C$ [39], it restricts the reference voltage to that of the energy bandgap of silicon ($\approx 1.205V$) which is undesirable from the viewpoint of a sub-1V reference as well as a programmable reference.

Several techniques have been proposed for modifying the bandgap reference voltage to provide voltages less than the bandgap voltage of silicon. The structure in [40] uses native nMOS transistors while those in [41] and [42] are architectures that avoid low-threshold voltage devices. A brief comparison of these techniques is given in [39].

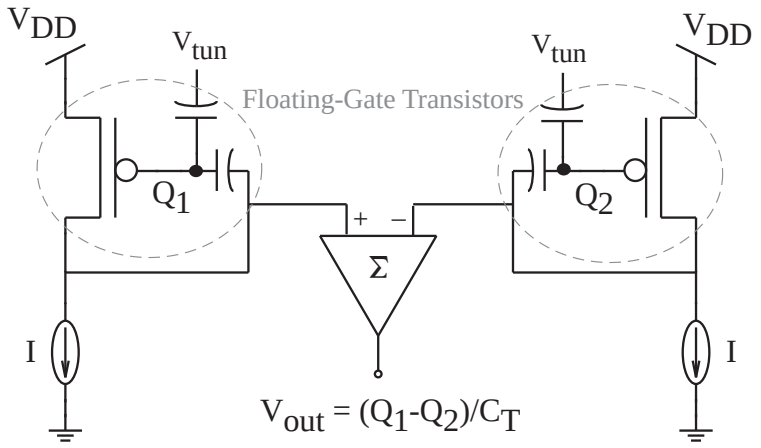


Figure 23. Conceptual representation of the proposed reference: The proposed reference is conceptually depicted. Charge is programmed onto floating-gate transistors, the difference of which forms the reference voltage. Such an approach gives a programmable reference that is temperature insensitive to a first order and displays negligible long term drift.

In all of these structures, the reference voltage is scaled using a ratio of resistors. These architectures require matched resistors with mismatch being addressed at the expense of area and costly post-fabrication schemes such as laser trimming. All of the above schemes restrict the output voltage to a single value that is set during the design phase thereby limiting the range of reference voltages.

A number of alternate techniques have been proposed to design voltage references wherein, the reference voltage is independent of the energy bandgap of silicon. The approach in [43] uses transistors fabricated with different threshold voltages to generate a voltage reference. A voltage reference has been demonstrated based on polysilicon gate work function difference in [44]. The approach in this work and in [25] uses floating-gate transistors.

In this chapter, a compact programmable architecture is proposed that implements the voltage reference based on the charge difference between two floating-gate transistors as depicted in Figure 23. The reference voltage is set by the programmed difference between the floating-gate charges. In Figure 23, the operation of finding

the difference between the charge on the floating-gates is shown explicitly using an operational amplifier for ease of understanding. In the practical implementation of the concept, the circuit architecture is such that the subtraction occurs without the need for a differencing amplifier. Programming the charge on the floating-gate transistors provides the flexibility of a programmable reference with the advantage of a single design providing multiple reference voltages. Programmability also results in a high initial accuracy for the reference. Owing to the non-volatile memory of floating-gate transistors, the reference displays low long-term drift in its output voltage, is well suited for low supply voltage operation and displays a low temperature co-efficient. Also, the proposed technique can provide a programmable current reference unlike other floating-gate transistor based approaches [45].

5.1 Reference Architecture

The practical implementation of the proposed concept in Figure 23 is shown in Figure 24. The proposed circuit is similar to the popular β-multiplier circuit [46] with the difference being that transistors $M1$ and $M2$ are designed to be floating-gate transistors. If $M1$ and $M2$ are identical and their currents match, the reference voltage (V_{ref}) is then given by

$$V_{ref} = V_{SG2} - V_{SG1} = \frac{\Delta Q}{C_T} \tag{87}$$

where ΔQ is the charge difference ($Q_2 - Q_1$) between the floating-gate transistors, C_T is the total capacitance at the floating-gate and κ is assumed to be equal to 1 in arriving at the above expression. These results are valid for both weak and strong inversion operation. This analysis ignores the Early effect and assumes that the input capacitance and the total floating-gate capacitance of the two transistors are matched.

The current through this circuit will be determined by the size of the resistor (R_1) and will be directly proportional to V_{ref} and is given by, V_{ref}/R_1. The resistor size can be used as a design parameter for a predetermined power consumption at a given

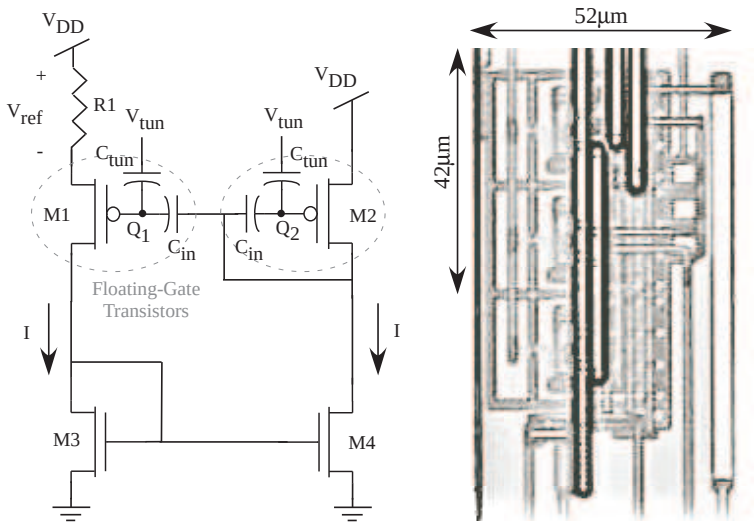


Figure 24. Simplified circuit schematic of the proposed reference and die photograph: Charge is programmed onto floating-gate transistors, the difference of which forms the reference voltage that appears across the resistor R_1. The die photograph of the reference fabricated in a $0.35\mu m$ CMOS process shows the compactness of the proposed approach.

reference voltage V_{ref}.

5.1.1 Reference Voltage

In this section, detailed expressions for the reference voltage will be developed in both the weak and strong inversion regions of operation to preserve generality.

5.1.1.1 Weak Inversion

Consider the current through transistor $M1$. Ignoring Early effects, the source-drain current through the device is given by,

$$I_1 = I_o exp\Big(\frac{-\kappa_{eff} V_g}{U_T}\Big) exp\Big(\frac{-\kappa \Delta Q}{U_T}\Big) exp\Big(\frac{V_{s1}}{U_T}\Big) exp\Big(\frac{(\kappa - 1)V_b}{U_T}\Big) \tag{88}$$

where, all the terminal voltages are referenced to ground and ΔQ is the charge difference between the floating-gate transistors $M1$ and $M2$. The current through $M2$ is given by,

$$I_2 = I_o exp\Big(\frac{-\kappa_{eff} V_g}{U_T}\Big) exp\Big(\frac{\kappa V_b}{U_T}\Big) \tag{89}$$

Dividing the above two equations, taking the natural logarithm on both sides and re-arranging the terms results in,

$$V_b - V_{s1} = -\Big[\frac{\kappa \Delta Q}{C_T} + U_T ln\Big(\frac{I_1}{I_2}\Big)\Big] = V_{ref} \tag{90}$$

It should be noted that if the currents I_1 and I_2 are exactly matched, the reference voltage is given by the charge difference between the floating-gate transistors. Also, for proper circuit operation, the total charge on $M1$ should be less than that of $M2$, thereby making ΔQ negative and therefore resulting in a positive reference voltage.

Now, owing to mismatch in the transistor parameters of $M3$ and $M4$, the currents in the two branches will not be exactly equal. As before, consider the dominant form of mismatch, namely, threshold voltage mismatch. Let the threshold voltage of $M3$ be different from that of $M1$ by ΔV_{th}. The current I_1 is given by,

$$I_1 = I_o exp\Big(\frac{\kappa V_{gb}}{U_T}\Big) exp\Big(\frac{-V_{sb}}{U_T}\Big) exp\Big(\frac{-\kappa V_{To}}{U_T}\Big) \tag{91}$$

and the current I_2 is given by,

$$I_2 = I_o exp\Big(\frac{\kappa V_{gb}}{U_T}\Big) exp\Big(\frac{-V_{sb}}{U_T}\Big) exp\Big(\frac{-\kappa (V_{To} + \Delta V_{th})}{U_T}\Big) \tag{92}$$

Dividing the above two equations results in,

$$\frac{I_1}{I_2} = exp\Big(\frac{\kappa \Delta V_{th}}{U_T}\Big) \tag{93}$$

Substituting the above expression in (90) gives,

$$V_{ref} = -\Big[\frac{\kappa \Delta Q}{C_T} + \kappa \Delta V_{th}\Big] \tag{94}$$

It is clear from the above expression that the threshold voltage mismatch of the nFET transistors appears directly as a term in the expression for the reference voltage and so should be minimized. It should also be pointed out that the presence of κ in the expression is on account of the bulk terminal of $M1$ tied to the power supply. It is easy to show that if the bulk terminal is tied to the source terminal the κ term drops out of the equation.

5.1.1.2 Strong Inversion

In order to analyze the ciruit behavior in strong inversion and to bring out the impact of body-effect in transistor $M1$ on the reference temperature sensitivity, the model popularly known as the source-referenced model has been adopted. Consider the drain current of an nFET in strong inversion. Ignoring Early effects, the current is given by,

$$I_1 = \frac{\mu_n C_{ox}}{2\alpha}\left(\frac{W}{L}\right)\Big[V_{gs} - V_T\Big]^2 \tag{95}$$

where, all variables are as defined earlier with V_T being the threshold voltage of the device given by,

$$V_T = V_{To} + \gamma\left(\sqrt{\phi_o + V_{sb}} - \sqrt{\phi_o}\right) \tag{96}$$

with ϕ_o being approximately equal to twice the Fermi potential of the bulk and α is given by,

$$\alpha = 1 + \frac{\gamma}{2\sqrt{\phi_o + V_{sb}}} \tag{97}$$

Using the above equations, the drain current of $M1$ can be written as,

$$I_1 = \frac{\mu_p C_{ox}}{2\alpha_1}\left(\frac{W}{L}\right)\Big[V_s - V_{fg} - |V_T|\Big]^2 \tag{98}$$

Taking the square-root on both sides of equation and noting that the charge difference between transistors $M1$ and $M2$ is ΔQ, the above equation can be re-written as,

$$\sqrt{\frac{I_1}{\beta_{p1}}} = V_{s1} - \frac{C_{in}}{C_T}V_g - \frac{\Delta Q}{C_T} - |V_{T1}| \tag{99}$$

Performing a similar analysis for the current through $M2$ results in,

$$\sqrt{\frac{I_2}{\beta_{p2}}} = V_{DD} - \frac{C_{in}}{C_T}V_g - |V_{T2}| \tag{100}$$

Subtracting the above two equations results in an expression for the reference voltage as,

$$V_{ref} = -\frac{\Delta Q}{C_T} + |V_{T2}| - |V_{T1}| + \sqrt{\frac{I_1}{\beta_{p2}}} - \sqrt{\frac{I_2}{\beta_{p1}}} \tag{101}$$

As before, the floating-gate of $M1$ has to be more negative than that of $M2$ for proper operation.

Next, consider mismatch in the current mirror pair $M3/M4$. Let the threshold voltage of $M4$ be higher than that of $M3$ by ΔV_{th}. Assuming strong inversion saturation and ignoring Early effects, the threshold voltage mismatch can be expressed as,

$$\Delta V_{th} = -\Big[\sqrt{\frac{I_2}{\beta_n}} - \sqrt{\frac{I_1}{\beta_n}}\Big] \tag{102}$$

Substituting the above expression in (101) and manipulating the algebra assuming that $\beta_{p1} = \beta_{p2} = \beta_p$ results in a complete expression for the reference voltage as given below.

$$V_{ref} = -\frac{\Delta Q}{C_T} + |V_{T2}| - |V_{T1}| - \Delta V_{th}\sqrt{\frac{\beta_p}{\beta_n}} \tag{103}$$

In deriving the above expression, it was assumed that the β's of $M1$ and $M2$ are equal. Although, this assumption is not valid, the error it introduces in the calculating the reference voltage is very small. This can be readily ascertained by noting that β_p term is multiplied by the threshold voltage mismatch, which, in a good design is on the order of $1 - 10mV$. Therefore, in assuming that the β's are the same, one has traded higher accuracy for a simple expression for the reference voltage.

A key design issue in strong inversion operation is the sizing of the input capacitance C_{in} of the floating-gate transistor. The capacitive division caused by C_{in} (see Figure 24) needs to be large enough to keep $M2$ in saturation. The bias current and the capacitive ratio should be designed such that the gate voltage of $M2$ obeys the following condition.

$$V_g < \frac{V_T}{(1 - C_{in}/C_T)} \tag{104}$$

Ideally a capacitive ratio of 1 ensures that $M2$ is in saturation for all values of V_g. However, designing according to the above equation will minimize circuit area.

5.1.2 Minimum Power Supply

The proposed architecture is advantageous in that it is well suited for low power supply operation. Notice that since the reference is essentially a circuit that operates at DC, long channel devices can be used and therefore cascoding can be avoided. For the circuit in Figure 24, a general expression can be written for the minimum power supply requirement. This is as given below.

$$V_{DD_{min}} = V_{ref} + V_{dsat1} + V_{gs3} \tag{105}$$

where, V_{ref} is the reference voltage, V_{dsat1} is the minimum source-drain voltage that is required across $M1$ to maintain it in saturation and V_{gs3} is the gate-source voltage of $M3$. Notice that the above expression is general in the sense that no specific region of operation has been assumed.

For weak inversion, a source-drain voltage of $100mV$ is sufficient to ensure saturation at room temperature and since the current is an exponential function of the terminal voltages, it can be assumed that V_{gs3} is approximately equal to the threshold voltage of the device (V_{T3}). Using the above assumptions, the minimum supply voltage required for weak inversion operation is given by,

$$V_{DD_{min}} = V_{ref} + 100mV + V_{T3} \tag{106}$$

For a $0.35\mu m$ process, the threshold voltage of an nFET is $0.5V$. Using this, a reference voltage of $0.4V$ can be achieved using a $1V$ power supply.

Assuming that the transistors are operating in the strong inversion regime, the minimum power supply for the reference is given by

$$V_{DD_{min}} = V_{ref} + V_{T3} + V_{dsat3} + V_{dsat1} \tag{107}$$

where, all variables are as defined earlier. Typical numbers for a $0.35\mu m$ CMOS process include, $V_{T3} = 0.5V$ and $V_{dsat3} = 0.3V = V_{dsat1}$. Using these, a $V_{DD_{min}} = 1.8V$ can be used to obtain a maximum reference voltage of $0.7V$. Modifications to the

reference circuit such as using a DC level-shifting current mirror [47] can result in lower supply voltage operation even in the strong inversion region.

5.1.3 Reference Output Noise

The total noise at the output of the reference circuitry is an important parameter that must be optimized. The noise source of a resistor is modeled using a current source in parallel with it. The mean-square thermal noise current spectral density of the resistor R_1 (i_n^2) is given by [48],

$$i_n^2/\Delta f = 4kT/R_1 \tag{108}$$

where, k is the Boltzmann's constant and T is the temperature in Kelvin. Similarly, noise for a transistor is modeled using a noise current source between its drain and source. The thermal noise mean-square current spectral density for a transistor is given by [48],

$$i_n^2/\Delta f = 4kT\gamma g_m \tag{109}$$

where, γ is a constant that varies from 2/3 to 2 and g_m is the transconductance of the transistor.

Assume that the effective input resistance looking into the source of $M1$ to be equal to R_{in}. The noise current sources of the resistor R_1 and all the transistors $M1 - M4$ flow into the impedance R_{in} to create a voltage noise at the source of $M1$. Considering each noise current source individually and applying superposition leads to the total voltage noise at the output to be equal to,

$$v_{out}^2/\Delta f = \frac{4kT}{R_1}R_{in}^2 + 4kT\gamma[g_{m1} + g_{m2} + g_{m3} + g_{m4}]R_{in}^2 \tag{110}$$

The above equation gives the total output noise spectral density. This when multiplied by the bandwidth of interest followed by taking a square-root of the result gives the total rms noise at the output.

The next step is to determine the input impedance at the source of $M1$. Refer to Figure 24. Let the input impedance looking into the source of $M1$ be Z_{in}. This is equal to the parallel combination of the input resistance R_{in} and a drawn capacitance of C. The input resistance R_{in} is composed of the parallel combination of the resistor R_1 and a resistance of R_s that is equal to the resistance looking into the source of $M1$ without the resistor R_1.

Inorder to estimate the resistance R_s, remove R_1 from the circuit and connect a test current source i_{test} to the source of $M1$. Using a small signal approximation, assuming infinite output impedance for the transistors and neglecting all parasitic capacitances, the current i_{test} flowing through $M1$ can be expressed as,

$$i_{test} = g_{m1}(v_s - v_{fg1}) \tag{111}$$

Assuming that the current mirror $M3/M4$ is a perfect $1:1$ current mirror, the test current i_{test} flows through the transistor $M2$ as well and can be expressed as,

$$i_{test} = -g_{m2}v_{fg2} \tag{112}$$

From the above expression, the small signal change in the floating-gate voltage of $M2$ can be determined. Now since the floating-gate transistor pair $M1/M2$ share the same gate voltage and assuming that their input and total floating-gate capacitors match, the change in the floating-gate voltage of $M1$ is equal to that of $M2$. With this observation one can use (112) in (111) to express the input resistance as,

$$R_s = \frac{1}{g_{m1}}\left(1 - \frac{g_{m1}}{g_{m2}}\right) \tag{113}$$

By using expressions for the transconductance of the two transistors, the above expression can be simplified. Assuming strong inversion saturation, the transconductance of $M2$ is given by,

$$g_{m2} = \beta[V_s - V_{fg} - V_T] \tag{114}$$

and the transconductance of $M1$ can be written as,

$$g_{m1} = \beta[V_s - V_{fg} + \frac{\Delta Q}{C_T} - V_T] = g_{m2}(1+m) \tag{115}$$

where, $m = \beta\Delta Q/(g_{m2}C_T)$ and it has been assumed that the threshold voltages of the two transistors match. Using the above expressions for the transconductance, the resistance R_s can be written as,

$$R_s = \Big(\frac{-m}{1+m}\Big)\frac{1}{g_{m2}} \tag{116}$$

Assuming κ is equal to 1, the resistor R_1 can be expressed as,

$$R_1 = \frac{2\beta\Delta Q}{g_{m2}^2 C_T} = \frac{2m}{g_{m2}} \tag{117}$$

Using the above two expressions, the overall input resistance at the source of $M1$ can be written as,

$$R_{in} = \Big(\frac{-2m}{m+1}\Big)\frac{1}{g_{m2}} = \frac{-\alpha}{g_{m2}} \tag{118}$$

Note that for proper circuit operation, m is positive and therefore making the input impedance negative! This negative impedance is on account of the circuit exhibiting positive feedback. The circuit, however, is stable, as for the configuration shown, the positive feedback gain is less than one. If the diode connections in the circuit were reversed with $M1$ and $M4$ being diode connected rather than the configuration shown in Figure 24, the positive feedback gain will be greater than one making the circuit unstable.

The effective input impedance is the parallel combination of the input resistance in (118) and a drawn capacitance C. It is easy to show that the noise currents are multiplied by a single pole transfer function. Therefore, the total voltage noise at the output is given by multiplying the noise spectral density given in (110) by the effective noise bandwidth. The effective noise bandwidth is given by,

$$NoiseBandwidth = \frac{\pi}{2} f_{-3dB} = \frac{g_{m2}}{4\alpha C} \tag{119}$$

The total output voltage noise is now given by,

$$v_n^2 = \frac{kT}{R}\frac{\alpha}{g_{m2}C} + \frac{kT}{C}\gamma\alpha\left(1 + \frac{g_{m1}}{g_{m2}} + \frac{g_{m3}}{g_{m2}} + \frac{g_{m4}}{g_{m2}}\right) \tag{120}$$

The above expression can be simplfied further to result in,

$$v_n^2 = \frac{kT}{C(1+m)}\left[1 + 2m\gamma\left(2 + m + \frac{g_{m3}}{g_{m2}} + \frac{g_{m4}}{g_{m2}}\right)\right] \tag{121}$$

It is clear from the above expression that the noise can be reduced by increasing the value of the capacitance C and by increasing the transconductance of transistor $M2$. Increasing the transconductance of $M2$ involves increasing the power dissipation and hence there exists a noise vs. power dissipation tradeoff.

5.2 Reference Temperature Sensitivity

The temperature sensitivity of the proposed reference voltage can be analyzed by considering each of the terms in (103) individually. To first order, the floating-gate capacitance and the charge difference display almost zero temperature dependence thus making the first term in (103) insensitive to temperature. With regards to the ratio of β's in the third term, it is important to consider the temperature dependence of the electron and hole mobilities. Typically, the electron and hole mobility temperature coefficient is modelled to be equal to -1.5. However, for doping concentrations greater than $10^{17}/cm^3$, the temperature coefficient of electron mobility is given by -1.2 while that of the hole mobility is given by -1.9 [32]. Taking these into account, the third term in (103) exhibits a temperature dependence. This can be mitigated by ensuring that the nMOS transistor pair $M3/M4$ match very well.

Finally consider the second term in (103), the threshold voltage difference between transistor pairs $M1/M2$. The threshold voltage of a MOS transistor, ignoring body effect, exhibits a linear temperature dependence with a temperature coefficient of $-2mV/^\circ C$ to $-4mV/^\circ C$ [11]. It is reasonable to assume that the temperature coefficients match, thereby resulting in $V_{th2} - V_{th1}$ being temperature independent.

However, if the bulk terminal of $M1$ is not tied to the source terminal the threshold voltage difference between $M1/M2$ becomes temperature sensitive. Assuming that this is the case, noting that the bulk-source potential of $M1$ is equal to the reference voltage V_{ref} and taking into consideration the differences in the temperature coefficients of electron and hole mobilities, (103) can be rewritten as,

$$V_{ref} = V_0 + \gamma\sqrt{2\phi_F + V_{ref}} - \gamma\sqrt{2\phi_F} - \Delta V_{thn}\sqrt{\frac{\beta_n}{\beta_p}} \tag{122}$$

where ϕ_F is the Fermi potential of the bulk, γ is the body effect coefficient which is constant and independent of temperature, and V_0 comprises all of the temperature independent contributions to the reference voltage. Note that ϕ_F, β_p and β_n are temperature dependent. The first order temperature dependance of V_{ref} is then given by,

$$\frac{\delta V_{ref}}{\delta T} \approx \frac{-\gamma\phi_F}{T}\left[\frac{1}{\sqrt{2\phi_F}} - \frac{1}{\sqrt{2\phi_F + V_{ref}}}\right] - \frac{\alpha\Delta V_{thn}}{2T}\sqrt{\frac{\beta_n}{\beta_p}} \tag{123}$$

where α is the difference in temperature coefficients of the electron and hole mobilities. It is clear from the above expression that the temperature sensitivity of the reference voltage is a function of the reference voltage itself.

5.3 Reference Drift With Time

The ideal reference voltage for the proposed circuit was shown in (87) to be directly proportional to ΔQ, the charge difference between the floating-gate transistors. Assuming that C_T remains constant with time, it is easy to see that the fractional change in the reference voltage $\left(\frac{V_{ref}(t)}{V_{ref}(0)}\right)$ will be equal to the fractional change in charge $(\frac{\Delta Q(t)}{\Delta Q(0)})$ of the floating-gate. The fractional change in charge occurs due to thermionic emission [27] and is modelled as a function of both temperature and time and is repeated here for convenience.

$$\frac{Q(t)}{Q(0)} = \frac{V_{ref}(t)}{V_{ref}(0)} = e^{-tv\cdot e^{\frac{-\phi_B}{kT}}} \tag{124}$$

where, $Q(0)$ is the initial charge on the floating-gate, $Q(t)$ is the floating-gate charge at time t in seconds, υ is the relaxation frequency of electrons in poly-silicon, ϕ_B is the $Si - SiO_2$ barrier potential, k is Boltzmann's constant and T is the temperature in Kelvins.

Assume that the two floating-gate transistors $M1$ and $M2$ are matched such that the charge loss in both the devices occur at an equal rate. For a given reference voltage at time $t = 0$, namely, $V_{ref}(0)$, the charge difference between the floating-gate transistors is $\Delta Q(0)$. For a given temperature, the charge difference at a time t, namely, $\Delta Q(t)$ can be estimated as $\alpha \Delta Q(0)$ using (124), where α is the fractional change in charge due to thermionic emission. Relating the charge difference to the reference voltage using (103) and assuming that all other parameters in (103) remain constant with time, the drift of the reference voltage with time is given by,

$$V_{ref}(t) - V_{ref}(0) = \frac{\alpha}{C_T} \Delta Q(0) \tag{125}$$

Estimating the parameters in the thermionic emission equation for a given process and using the above equation, the reference voltage drift in time can be estimated. It should be noted that on account of the non-volatile charge storage capability of floating-gate transistors, a low drift in the reference voltage can be expected.

5.4 Experimental Results

A prototype reference has been fabricated in a $0.35\mu m$ CMOS process. It is evident from the circuit architecture that for a given reference voltage, a trade-off results between the size of the resistor used and the power consumption and hence the region of operation of the circuit. Operation in weak inversion results in an extremely low power dissipation whereas a high resistance value will be required for a given reference voltage. However, operation in strong inversion requires smaller resistor values at the expense of increased power dissipation. In the design of the prototype circuit, a $50K\Omega$ resistor was chosen to limit the bias currents in the mico-ampere range. The

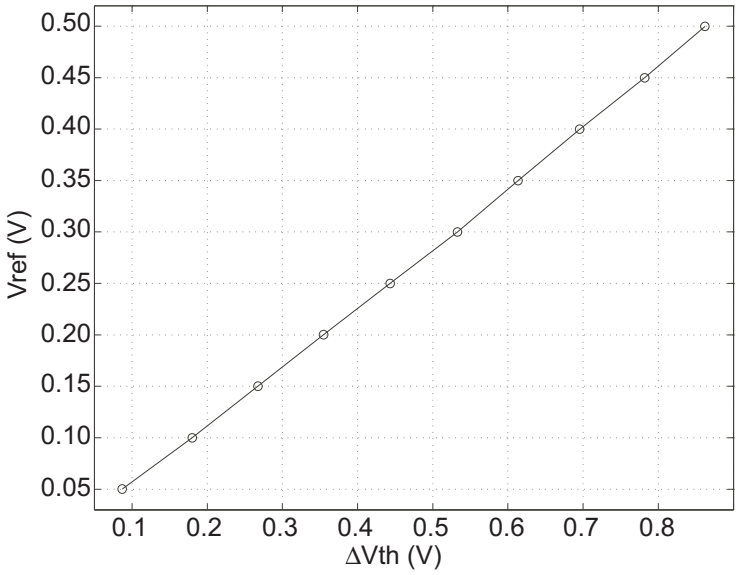


Figure 25. Reference Voltage vs. Threshold Voltage Difference: Plot of reference voltage plotted against the threshold voltage difference between transistors $M2$ and $M1$. The slope of the plot is equal to the capacitive division from the external gate of the transistors to the floating-gate.

circuit operates in the strong inversion region. Measured results that demonstrate the proposed reference's performance are presented in this section.

Figure 25 shows a plot of the programmed reference voltage as a function of the threshold voltage difference between transistors $M2$ and $M1$. The plot is linear as is implicitly conveyed in the theoretical equation (103). The experimental results can be better interpreted by recalling that programming a floating-gate transistor can be treated as modifying the threshold voltage of the device. In (103), the programmed charge can be lumped along with the threshold voltage terms. This results in the view that the reference voltage is linearly dependent on the difference in threshold voltages of transistors $M1$ and $M2$ as confirmed by measured results. The above representation is convenient from a measurement standpoint as modifications in the floating-gate charge can be quantified by means of a threshold voltage change.

It can be observed from Figure 25 that the slope of the plot is not equal to one as theoretically expected. Instead, the slope of the plot is equal to the capacitive

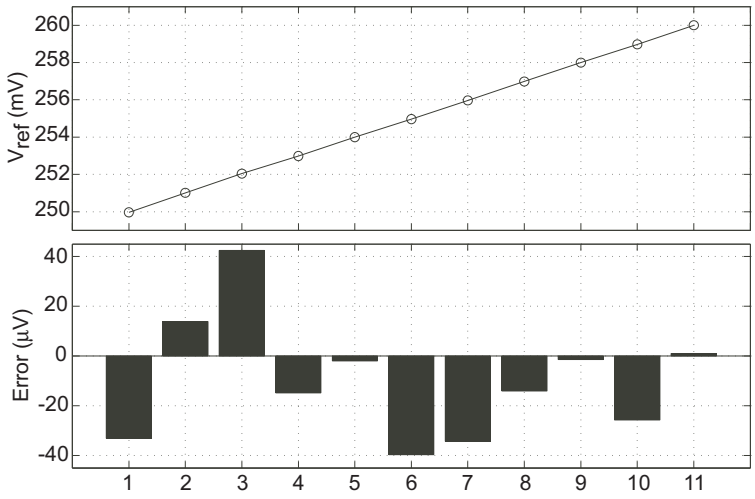


Figure 26. Fine programming of the reference voltage: (a) The reference voltage is programmed in steps of $1mV$ from $0.25V$ to $0.26V$. (b) The error voltage of the programmed reference to the ideal target value. The maximum error is between $\pm 40\mu V$ indicating a good programming accuracy on the reference voltage.

division that occurs from the external gate V_g of these devices to the floating-gate of $M1/M2$. This is on account of the fact that the threshold voltage of the transistors was measured by sweeping the external gate and hence the measured threshold voltage is scaled by a factor equal to the capacitive division at the input.

The programming capability of the reference is clearly demonstrated in Figure 25. A key parameter in designing references is the accuracy to which a particular reference value can be guaranteed. Typical accuracy numbers for popular schemes such as band-gap voltages is $1 - 2mV$. In order to estimate the accuracy achievable with the proposed scheme, the reference voltage was programmed in steps of $1mV$ from a value of $0.25V$ to $0.26V$. Figure 26 shows the measured curve. Figure 26 shows the deviation of the programmed reference from the target value. As can be observed, the average error due to programming is within $\pm 40\mu V$. This clearly demonstrates the high accuracy that is possible on account of the programmable nature of the reference voltage.

The above accuracy of $\pm 40\mu V$ has been achieved at the package level. This is a significant advantage over other schemes such as laser trimming that are techniques

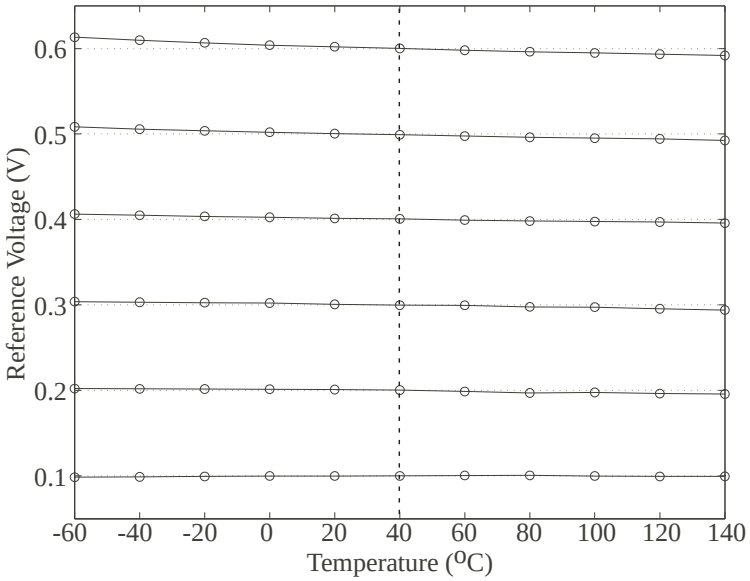


Figure 27. Reference Voltage vs. Temperature: The reference voltage as a function of temperature is plotted over a temperature range of $-60°C$ to $140°C$ for five different reference voltages of $0.1V$ to $0.5V$.

applied at the wafer level. A wafer level technique suffers from the drawback of the trimmed value changing drastically on account of the stresses induced by the packaging process and the package itself.

The temperature sensitivity of a reference voltage is a critical parameter that is of interest. In typical reference circuits, the reference voltage is a single value that is usually chosen to be a value that exhibits the minimum temperature sensitivity. In the proposed circuit, however, programmability provides the flexibility of multiple reference voltages. It is therefore important to analyze the temperature sensitivity for different reference voltages.

Figure 27 shows experimental results for the temperature dependence of the prototype chip. The reference voltage was programmed to five different values ranging from $100mV$ to $500mV$ at room temperature and measured across temperature from $-60°C$ to $140°C$. A plot of a single reference voltage ($0.4V$) is shown in order to provide a more detailed view of the temperature dependence of the reference voltage.

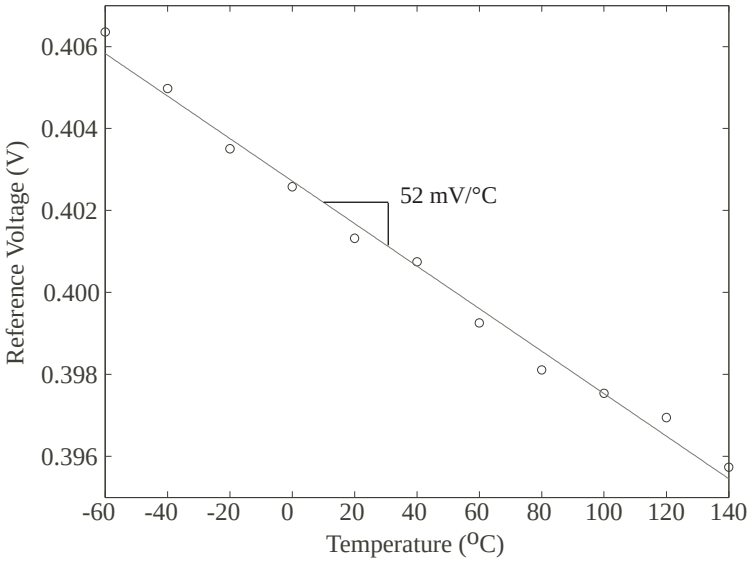


Figure 28. Reference Voltage vs. Temperature: The reference voltage as a function of temperature is plotted for a reference voltage of $0.4V$ over a temperature range of $-60°C$ to $140°C$. The reference voltage displays a temperature co-efficient of $53\mu V/°C$.

The reference voltage displays a linear dependence with temperature. This is primarily because of the temperature dependence of the Fermi level of the bulk terms that play a role in determining the reference voltage.

The temperature sensitivity of the reference voltage as a function of the reference voltage is shown in Figure 29. As expected from (123), the temperature sensitivity increases as a function of V_{ref}. A maximum sensitivity of $110\mu V/^oC$ was obtained for $V_{ref} = 0.6V$, while a minimum sensitivity of $10\mu V/^oC$ was obtained for $V_{ref} = 0.1V$. These results were corroborated by solving (122) via numerical analysis. As can be observed in Fig. 29, the measured data and theoretical predictions match closely. A maximum temperature sensitivity of $25\mu V/^oC$ is expected (assuming $\Delta V_{thn} \approx 15mV$) across all values of V_{ref} for the case were the bulk terminal of $M1$ is tied to the source terminal.

Accelerated life-time tests, as outlined in [49] for a $0.5\mu m$ CMOS process, were used to extract the parameters of (124) for a $0.35\mu m$ CMOS process. Table 5 shows

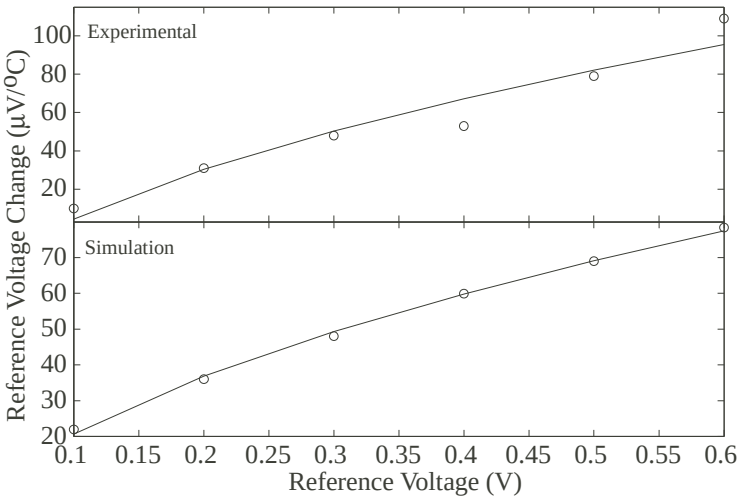


Figure 29. Reference Temperature Co-efficient vs. Reference Voltage: The temperature coefficients of the reference circuit for different values of the reference voltage is plotted. The variation of the temperature co-efficient is similar for both measured results and theoretical simulations.

Table 5. Reference Voltage Drift

Temperature (^oC)	325	325	125
Time (hrs)	24	48	400
$\frac{V_{ref}(T)}{V_{ref}(0)}$	0.967	0.953	0.998

the data obtained; values for υ and ϕ_B were found to be $55ms^{-1}$ and $0.618eV$ respectively. A $400\mu V$ drift over a period of 10 years at 25^oC was extrapolated from (124). Figure 30 shows V_{ref} at 125^oC for a period of approximately 450hrs; a net change of $400\mu V$ was recorded. The inset shows the same data on a log scale. A small jump of approximately $-5mV$ occurs, as expected from Fig. 29, due to the increase in temperature from 25^oC to 125^oC. Figure 31 shows V_{ref} at 25^oC for a period of approximately 100hrs. A negligible change in the reference voltage was obtained. It has been observed that the charge drift greatly reduces after a burn in period of around 1 day at temperatures above 300^oC. The chip shows $20\mu V_{rms}$ of noise over approximately a 10kHz bandwidth.

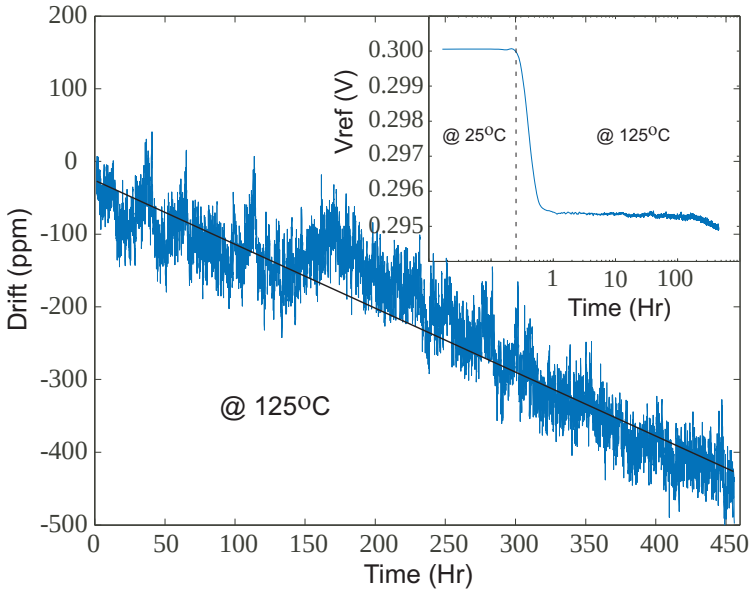


Figure 30. Reference Voltage Drift at $125°C$: Measured reference voltage drift against time at 125^oC for a reference programmed to $0.3V$ at $25°C$. The inset shows the transient behavior of the reference voltage as the temperature is increased from $25°C$ to $125°C$.

5.5 Comparisons To Alternate Techniques

Overall chip performance is summarized and compared against a select set of reported voltage references in Table 6. In arriving at Table 6 it was decided to only include techniques that were feasible in a standard CMOS process. The work in [47] was included as it implements a sub-$1V$ bandgap reference and reports a very low temperature co-efficient of $15ppm/°C$. The technique in [50] merits mention on account

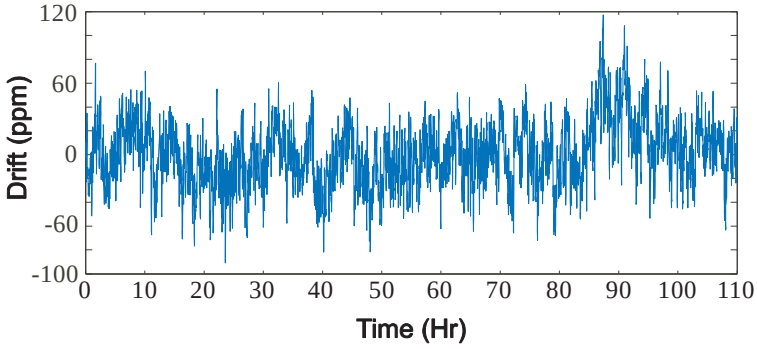


Figure 31. Reference Voltage Drift at $25°C$: Measured reference voltage drift against time at 25^oC for a $0.3V$ reference voltage observed over a period of $100hrs$.

of being a novel implementation of a bandgap reference with no resistors and without loss of performance. The approach in [51] is an alternative to the bandgap voltage reference that is implementable in a standard CMOS process. For the sake of completeness, this work is compared with an alternate floating-gate based technique [25] as well.

The technique in [47] implements a reference voltage that is a scaled version of the silicon bandgap voltage. The scaling is achieved using a ratio of resistors and occurs internally with the result that the reference operates with a $0.98V$ supply voltage. Two sets of resistors are used, one for scaling the bandgap voltage and the other for trimming the bandgap to achieve a zero temperature co-efficient at a reference temperature. By varying the resistor ratios, different reference voltages can be achieved. However, the circuit is not programmable in the true sense as only discrete voltage levels can be achieved with higher resolutions resulting in an area penalty.

In [50], transistors and diodes are used to design a bandgap voltage reference without using any resistors. The main advantage of this technique is that it can be implemented in any standard digital CMOS process. Like any bandgap reference, trimming is needed to achieve a good performance across the temperature range of operation. This is performed by using a bank of transistors of various aspect ratios and digitally switching in the transistor of the appropriate ratio that gives optimum performance. The main drawback of this approach is that the reference voltage is fixed at a value close to the silicon bandgap and is therefore not programmable.

The technique in [51] is different from the conventional bandgap voltage references in that it is based on the difference between a weighted nFET gate-source voltage from that of a pFET source-gate voltage. The nFET gate-source voltage is scaled by a ratio of resistors with the resistor ratio set such that temperature effects arising out of threshold voltage temperature dependencies are cancelled. Similarly, the nFET

Table 6. Comparison of Voltage References

Parameter	This Work	Leung [51]	Leung [47]	Buck [50]	Ahuja [25]
Technique	Floating-Gate	Weighted ΔV_{GS}	Bandgap	Bandgap	Floating-Gate
Technology	$0.35\mu m$ CMOS	$0.6\mu m$ CMOS	$0.6\mu m$ CMOS	$0.6\mu m$ CMOS	$1.5\mu m$ E^2PROM CMOS
Min. Supply Voltage	$2.5V$	$1.4V$	$0.98V$	$3.7V$	$2.7V$
V_{ref} **Range**	$[50mV - 0.6V]$	$309.31mV$	$603mV$	$1.1195V$	$[0.5V - 5V]$
Temperature Coefficient	$130ppm/°C$ (V_{ref}=$0.4V$)	$36.9ppm/°C$ (trim)	$15ppm/°C$ ($4bit$ R trim)	$134ppm/°C$ (trim)	$< 1ppm/°C$
Temperature Range	-60 to $140°C$	0 to $100°C$	0 to $100°C$	0 to $70°C$	-40 to $85°C$
Voltage Drift @ 10 years	$400ppm$	NA	NA	NA	$24ppm$
Initial Accuracy	$\pm 40\mu V$	$\pm 19.26mV$	NA	$\pm 0.5mV$	$\pm 200\mu V$
Power Dissipation	$40\mu W$	$13.6\mu W$	$17.6\mu W$	$1.4mW$	$1.3\mu W$
Area	$0.0022mm^2$	$0.055mm^2$	$0.24mm^2$	$0.40mm^2$	$1.6mm^2$

and pFET W/L ratios are set to a value that cancels mobility related non-linear temperature effects. This technique, just like the bandgap reference technique, can be designed to provide a zero temperature co-efficient at a single temperature. Achieving this requires resistor trimming which in turn changes the reference voltage thereby resulting in a poor initial accuracy for the reference. Also, the reference voltage is fixed to a particular value set during the design phase.

A programmable reference based on charge storage is reported in [25]. The reference has been fabricated in a $1.5\mu m$ E^2PROM process, although, the proposed technique can be implemented in a standard CMOS process as well. The reported temperature co-efficinet is better than the proposed work but suffers from a poorer initial accuracy. It is hypothesized that this is on account of better programming control that is possible using a hot-electron injection based programming than a completely tunneling based programming as in [25]. The proposed work is a more compact implementation which is advantageous when a number of on-chip programmable references are required as in a large scale reconfigurable framework as described in [52]. Also, the circuitry in this work very easily serves as a current reference as well.

5.6 Summary

A simple compact programable reference that only occupies $0.0022mm^2$ of area (excluding buffers) in a $0.35\mu m$ standard digital CMOS technology has been described. The proposed reference has been programmed to output voltages from $50mV$ to $600mV$ with a $\pm 40\mu V$ accuracy. Reference voltages beyond the demonstrated range are possible as well. A temperature coefficient of $130ppm/^{\circ}C$, for a $0.4V$ reference, has been obtained for the prototype chip. The temperature co-efficient is directly proportional to the reference voltage for the implementation shown. Both the temperature co-efficient and its dependence on the reference voltage can be reduced by

eliminating body-effect in transistor $M1$ in Figure 24. Reliability experiments performed for a reference programmed to $0.3V$ and baked at $125°C$ indicates a reference drift of $\approx 450\mu V$ over $450hrs$. A worst case voltage drift of $400\mu V$ over a period of 10 years at 25^oC has been extrapolated from the accelerated life-time tests.

CHAPTER 6

VECTOR MATRIX MULTIPLIER

Multiplication and addition are two operations that are performed repeatedly in signal processing. These multiply-and-accumulate blocks are the fundamental building blocks in a number of signal processing applications such as 2-D block transforms for image processing, FIR filtering, convolution and correlation [53, 54]. In signal processing, the multiplication operation involves multiplying an input signal or a discrete-time sample of an input signal by a quantity that is typically referred to as the weight. The result of this operation is usually added with the results of a number of other multipliers to produce the final system output.

A vector-matrix multiplier (VMM) is a system that contains a number of such multiply-accumulate units and performs the multiplication of a number of input signals (an input signal vector) with an array of weights (weight vector). This operation is quite common in applications such as neural networks and adaptive filters. The basic VMM operation is defined as,

$$Y_j = \Sigma_i W_{ji} I_i \tag{126}$$

where, Y_j is the output signal vector, I_i is the input signal vector and W_{ji} is a matrix of application-dependent coefficients.

Addition and multiplication that are key to the VMM operation are both area and power intensive in a digital realization thereby making it impractical for large VLSI systems [53]. An analog implementation on the other hand is compact, low power and can eliminate the need for data converters in the case of analog interfaces. Also, the computation can be done in parallel and faster in analog since the weights are stored at each multiplier site and thereby saves fetch time [55, 56, 57].

Analog multiplication of a signal with a weight co-efficient can be performed using either a voltage-mode approach or a current-mode approach. A voltage-mode

approach suffers from a limited linear range. The linear range is usually extended at the expense of both power dissipation and circuit complexity. A current-mode approach, on the other hand, inherently offers a larger linear range as can be observed from a simple current mirror that offers decades of linearity.

In this work, a current-mode approach for the VMM architecture has been adopted. The multiplier cell is biased in the weak inversion regime thereby achieving low-power operation. Also, the addition operation is performed using KCL and hence, does not dissipate any additional power when compared to digital approaches. The exponential I-V relationship of transistors operating in the weak inversion region provides a logarithmic compaction that increases the linearity of the multiplier as compared to a voltage-mode technique. Also, the proposed architecture provides for programmable, non-volatile weight storage through the use of floating-gate MOSFETs.

6.1 Programmable Multiplier

Figure 32(a) shows a simple pMOS current mirror composed of transistors $M1$ and $M2$. Transistor $M2$ is designed such that its aspect ratio is w times the aspect ratio of $M1$. Assuming infinite output impedance and that both transistors are matched, it is easy to show that the output current is equal to,

$$I_{out} = w \cdot I_{in} \tag{127}$$

It should be noted that the above expression is valid for a large range of input currents and is valid for transistor operation from weak inversion through moderate inversion to strong inversion.

A simple current mirror therefore produces an output that is a multiplication of the input current by a weight w and is a potential candidate for signal processing applications. The main disadvantage to such an approach is that the weight w cannot be altered after chip fabrication. Several researchers have attempted to provide programmability to the simple current mirror. One such approach is to build the

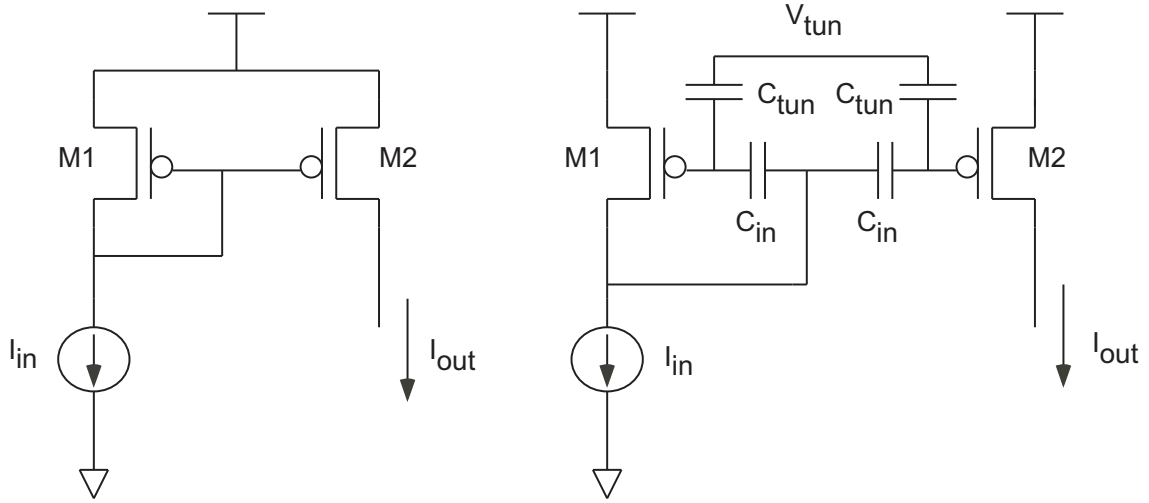


Figure 32. Current-mode multipliers: (a) Circuit schematic of a simple pMOS current mirror. (b) Circuit Schematic of a floating-gate pMOS current mirror.

transistor $M2$ using a number of unit size transistors and switch in a required number using a digital-to-analog converter. This creates a variable gain current mirror with finer weight resolutions obtained at the cost of both area and power dissipation.

Figure 32(b) shows a floating-gate current mirror that will be shown to implement a programmable multiplier in a compact and low-power fashion. The key difference of this circuit from a conventional current mirror is that transistors $M1$ and $M2$ are made floating-gate transistors. The operation of this floating-gate current mirror as a programmable multiplier will be described in both the weak and strong inversion regions of operation. Also, theoretical analysis of other performance metrics such as bandwidth and noise will be performed.

6.1.1 Multiplier operation

The operation of the multiplier in the weak inversion region is considered first. The drain current of floating-gate transistor $M1$ in weak inversion saturation, neglecting Early effects, is given by,

$$I_{in} = I_o exp\Big(\frac{-\kappa_{eff} V_g}{U_T}\Big) exp\Big(\frac{-\kappa Q_1}{C_T}\Big) exp\Big(\frac{V_s}{U_T}\Big) \tag{128}$$

where, all variables are as defined earlier, Q_1 is the charge on the floating-gate of $M1$ and $\kappa_{eff} = \kappa C_{in}/C_T$.

Similarly, the drain current of $M2$ is given by,

$$I_{out} = I_o exp\Big(\frac{-\kappa_{eff} V_g}{U_T}\Big) exp\Big(\frac{-\kappa Q_2}{C_T}\Big) exp\Big(\frac{V_s}{U_T}\Big) \tag{129}$$

where, Q_2 is the charge stored on the floating-gate of $M2$.

Dividing the above two equations results in the transfer function from the input to the output being,

$$\frac{I_{out}}{I_{in}} = exp\Big(\frac{-\kappa(Q_2 - Q_1)}{C_T U_T}\Big) = w \tag{130}$$

It is clear from the above equation that different multiplication weights can be implemented by programming the difference in the floating-gate charge of transistors $M1$ and $M2$. Theoretically, the above weight equation translates to decades of linearity as long as the two transistors remain in the weak inversion region of operation. However, the above equation has been derived under the assumption that κ does not vary with surface potential and hence the programmed floating-gate charge. Incorporating this second order effect, the weight is now given by,

$$w = exp\Big(\frac{-(\kappa_2 Q_2 - \kappa_1 Q_1)}{C_T U_T}\Big) exp\Big((\kappa_2 - \kappa_1)\frac{C_{in} V_g}{C_T U_T}\Big) \tag{131}$$

The dependence of the weight on the gate voltage limits the linearity of the multiplier structure. A possible solution to increase the linearity would be to program the two floating-gate transistors relatively close to each other such that their κ's are almost equal. This approach will yield fractional weights that can easily be amplified in later stages.

Next, consider the operation of the floating-gate current mirror in the strong inversion region of operation. The drain current of $M1$ in saturation, ignoring Early effects is given by,

$$I_{in} = \frac{\mu_p C_{ox} W}{2\kappa L}(V_s - \kappa_{eff} V_g - \frac{\kappa Q_1}{C_T} - |V_{To}|)^2 \tag{132}$$

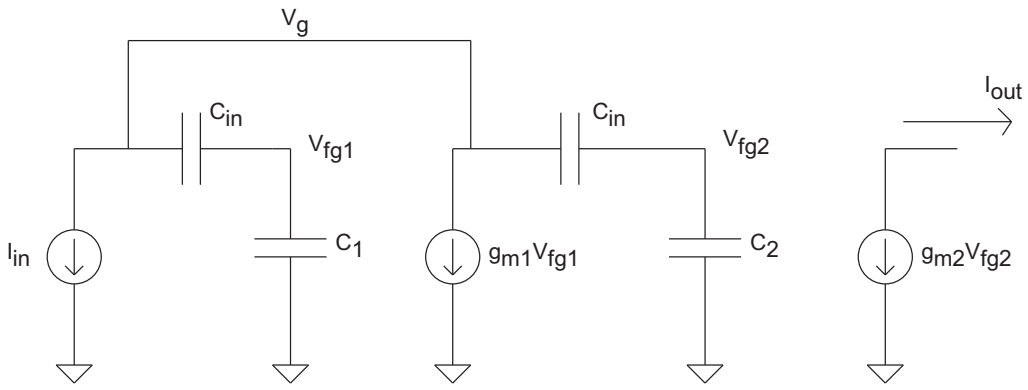


Figure 33. Floating-gate multiplier small-signal model: Circuit schematic showing the small-signal model for the proposed floating-gate current mirror.

and similarly, the drain current of $M2$ is given by,

$$I_{out} = \frac{\mu_p C_{ox} W}{2\kappa L}(V_s - \kappa_{eff} V_g - \frac{\kappa Q_2}{C_T} - |V_{To}|)^2 \tag{133}$$

Dividing the above two expressions results in the value of the weight as,

$$\frac{I_{out}}{I_{in}} = \frac{(V_s - \kappa_{eff} V_g - \frac{\kappa Q_2}{C_T} - |V_{To}|)^2}{(V_s - \kappa_{eff} V_g - \frac{\kappa Q_1}{C_T} - |V_{To}|)^2} = \frac{(V_{od1} - \frac{\kappa(Q_2 - Q_1)}{C_T})^2}{(V_{od1})^2} \tag{134}$$

where, V_{od1} is the overdrive voltage of transistor $M1$. Noting that the overdrive voltage is typically on the order of $200mV$, the above expression can be simplified as,

$$\frac{I_{out}}{I_{in}} = 1 - \frac{2\kappa(Q_2 - Q_1)}{C_T V_{od1}} = w \tag{135}$$

From the above expression it is clear that programming a difference in charge between the floating-gates of $M1$ and $M2$ results in a programmable current-mode multiplier in the strong inversion region as well. However, the dependence of the weight on changes in the gate voltage is much stronger in strong inversion than in weak inversion. This limits the linearity of the multiplier and therefore, both from a linearity and power dissipation standpoint, it is advantageous to operate the multiplier in weak inversion.

6.1.2 Frequency Performance

Figure 33 shows the simplified small-signal equivalent model of the floating-gate multiplier element. The floating-gate to drain capacitance and output impedance have

been neglected for ease of analysis. The capacitor C_1 shown in the figure is a combination of a number of parasitics and is given by,

$$C_1 = C_{gs1} + C_{gb1} + C_{tun} \tag{136}$$

where, C_{gs1} represents the floating gate to source capacitance of transistor M1, C_{gb1} represents the floating gate to bulk capacitance of $M1$ and C_{tun} represents the tunneling capacitance. It should be noted that to a first approximation, C_1 is dominated by the floating gate to source capacitance C_{gs1}. Also, the capacitance C_2 is the analogous lumped capacitance at the floating gate of $M2$.

The capacitors $C1$ and $C2$ can be assumed equal if the two floating-gate transistors $M1$ and $M2$ are equal. With this assumption, the floating-gate voltages can be written as,

$$V_{fg1} = V_{fg2} = \frac{C_{in}}{C_{in} + C_1} = \frac{C_{in}}{C_T} \tag{137}$$

Now, applying KCL at the input node results in,

$$I_{in} = -V_g\left(g_{m1}\frac{C_{in}}{C_T} + s\frac{C_{in}C_1}{C_T}\right) \tag{138}$$

Note that the output current I_{out} is equal to $-g_{m2}V_{fg2}$. Expressing the gate voltage V_g in terms of the floating-gate of $M2$ and using the expression for I_{out} in (138) and re-arranging results in,

$$\frac{I_{out}}{I_{in}} = \frac{g_{m2}}{g_{m1}}\left[\frac{1}{1 + sC_1/g_{m1}}\right] \tag{139}$$

The above expression is an approximate first-order response of the circuit. The bandwidth of the circuit depends on the transconductance of the input transistor $M1$ and the capacitance C_1.

6.1.3 Signal-to-Noise Ratio of Multiplier Cell

The f_{-3dB} frequency of the multiplier is given by,

$$f_{-3dB} = \frac{1}{2\pi} \cdot \frac{g_{m1}}{C_{gs1}} \tag{140}$$

Since, the input to output transfer function for the floating-gate current mirror is approximately a single-pole response, the noise bandwidth of the system is given by,

$$NoiseBandwidth = \frac{\pi}{2} \cdot f_{-3dB} = \frac{1}{4} \cdot \frac{g_{m1}}{C_{gs1}} \tag{141}$$

The total noise current spectral density at the output of the floating-gate current mirror is equal to the sum of the noise contributions of each of the two transistors $M1$ and $M2$.

$$\frac{i_o^2}{\Delta f} = 4kT\gamma\left(g_{m1}\frac{g_{m2}^2}{g_{m1}^2} + g_{m2}\right) \tag{142}$$

Referring the noise back to the input by dividing by the DC gain gives,

$$\frac{i_{in}^2}{\Delta f} = 4kT\gamma g_{m1}\left(1 + \frac{g_{m2}}{g_{m1}}\right) \tag{143}$$

Using the above expression and the expression for the noise bandwidth, the total input referred rms current noise is given by,

$$i_{in,rms} = g_{m1}\sqrt{\frac{\gamma kT}{C_{gs}}\left(1 + \frac{g_{m1}}{g_{m2}}\right)} \tag{144}$$

At this point, the SNR of the floating-gate current mirror can be calculated by assuming that the given current mirror has a bias current of I_{bias} flowing through it. The rms value of the full-scale input signal then becomes,

$$i_{sig,rms} = \frac{I_{bias}}{2\sqrt{2}} \tag{145}$$

Assuming weak inversion operation, the transconductance of a transistor is given by,

$$g_m = \frac{\kappa I}{U_T} \tag{146}$$

Using the above expression, the signal-to-noise ratio (SNR) of the floating-gate current mirror is given by,

$$SNR = \frac{U_T}{2\sqrt{2}\kappa}\sqrt{\frac{C_{gs}g_{m2}}{\gamma kT(g_{m1} + g_{m2})}} \tag{147}$$

Now, if the floating-gate current mirror is designed to have a weight of w, $g_{m2} = w \cdot g_{m1}$. Substituting this in the above expression simplifies the result as,

$$SNR = \frac{U_T}{2\sqrt{2}\kappa}\sqrt{\frac{C_{gs}w}{\gamma kT(1+w)}} \tag{148}$$

For a given value of weight w, the SNR of the floating-gate current mirror can be improved by increasing the floating-gate to source capacitance. Increasing C_{gs} to decrease the noise, decreases the frequency bandwidth and therefore, inorder to maintain a given frequency bandwidth, one has to increase the power dissipation. Thus, the floating-gate current mirror displays a noise vs. power dissipation tradeoff.

6.1.4 Multiplier Weight - Long Term Drift

The charge loss mechanism in floating-gate transistors can be modeled using a thermionic emission model as detailed in section 3.5. The thermionic emission model is repeated here for convenience. For a given initial charge $Q(0)$, the charge at time t ($Q(t)$) is given by,

$$\frac{Q(t)}{Q(0)} = exp\Big[-tvexp\Big(\frac{-\phi_B}{kT}\Big)\Big] = \alpha \tag{149}$$

where, ϕ_B is the effective barrier potential and υ is the relaxation frequency of electrons. Using the above expression the long term drift of the weight w over time can be estimated.

Using the expression for the weight in (130) along with the thermionic emission model, the long-term drift in the programmed weight can be estimated. Mathematically, this can be expressed as,

$$w(t) = w(0)^{\alpha} \tag{150}$$

where, $w(0)$ represents the initial weight at time $t = 0$ and $w(t)$ represents the weight at time t. The parameters υ and ϕ_B need to be extracted for each process using an accelerated life-time measurement technique as outlined in section 3.5. Using these extracted values, the drift in the weight can be estimated for different times for various operating and storage temperatures. For a $0.5\mu m$ CMOS process, the value of υ has

Table 7. Multiplier Weight Percentage Drift With Time

Temperature	$w = 0.25$	$w = 0.5$	$w = 0.75$
$25°C$	0.0017%	8.27×10^{-4}%	3.43×10^{-4}%
$90°C$	0.8730%	0.4355%	0.1805%
$140°C$	29.13%	13.63%	5.446%

been extracted to be $60s^{-1}$ and that of ϕ_B has been extracted to be $0.9eV$. Using these measured values and (150) the long-term percentage drift in weights over a 10 year period has been computed for three different weights (0.25, 0.5 and 0.75) over three different storage temperatures ($25°C$, $90°C$ and $140°C$). Table 7 summarizes the result and as can be observed from Table 7, for normal operating conditions, the floating-gate transistor and therefore the programmed weight exhibits minimal drift and therefore presents a viable technique for implementing programmable multipliers.

6.1.5 Multiplier Weight - Temperature Sensitivity

Observing (130), it is clear that the temperature sensitivity of the multiplier weight depends on the temperature sensitivity of κ, the floating-gate charge difference, the total floating-gate capacitance and the thermal voltage U_T. As will be shown later, for normal operating temperatures, the floating-gate charge loss is negligible and therefore the floating-gate charge difference can be assumed independent of temperature. To a first order, the total floating-gate capacitance can be assumed temperature independent. In comparison to the thermal voltage, the temperature variation of κ is small and can therefore be assumed independent of temperature as well.

Since the dominant mechanism for the temperature sensitivity of the weight is the thermal voltage and the variation of the thermal voltage with temperature is well known, the weight can be calibrated with temperature if needed. An alternate technique to nullifying the temperature dependence would be to create the input signal using a proportional to absolute temperature (PTAT) current source. Such an

approach would ensure that the multiplication is temperature independent to a first order.

6.2 Vector Matrix Multiplier Implementation

The floating-gate current mirror shown in Figure 32(b) implements a one-quadrant current-mode multiplier. Two floating-gate current mirrors are used in a differential fashion to implement the two-quadrant multiplier. Each floating-gate current mirror is programmed to implement a certain weight, w^+ and w^- in this case and the output of the two-quadrant multiplier is given by the input signal multiplied by the difference between these two weights. The output of the two-quadrant multiplier is therefore given by,

$$I_{out}^{+} = I_{in}^{+}(w^+ - w^-) \tag{151}$$

where, I_{in}^{+} is the input signal and I_{out} is the multiplier output.

A four-quadrant multiplier can be implemented by using a pair of two-quadrant multipliers. Assume that a two-quadrant multiplier is implemented with the difference that the input current is now given by I_{in}^{-}. The output of this multiplier is given by,

$$I_{out}^{-} = I_{in}^{-}(w^+ - w^-) \tag{152}$$

Subtracting the above two equations results in a four-quadrant multiplier operation. The output is given by,

$$I_{out} = (I_{in}^{+} - I_{in}^{-})(w^+ - w^-) \tag{153}$$

Figure 34 shows a simplified circuit schematic of rows of four-quadrant multipliers. A symbol for a floating-gate transistor has been used for convenience.

In a floating-gate device, the output impedance is degraded primarily due to the drain voltage (V_d) variation coupling onto the floating-gate node through C_{gd} rather than channel length modulation. Therefore, cascoding helps in reducing the floating-gate to drain parasitic capacitor-coupling effect by keeping the drain terminal fixed.

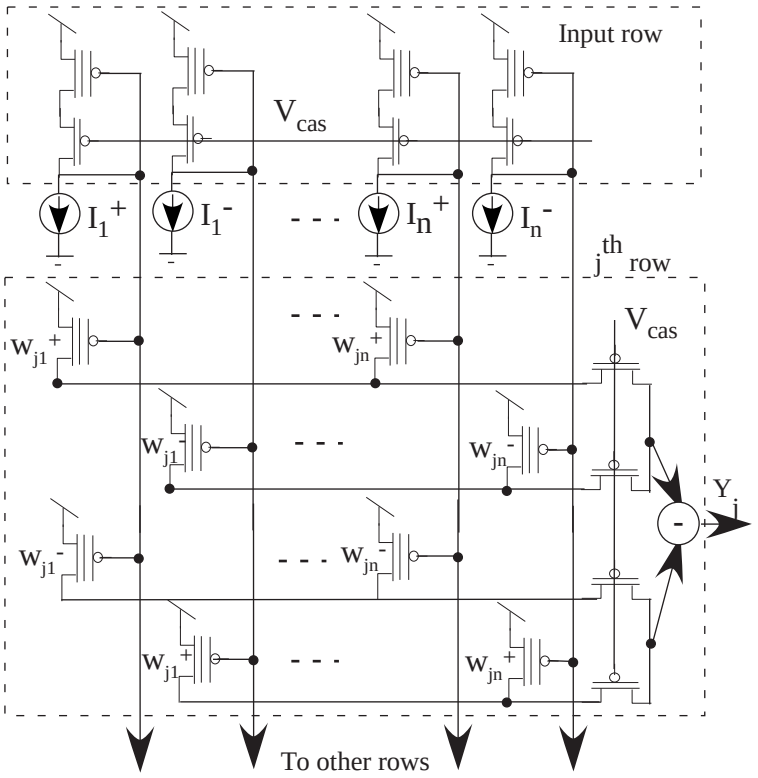


Figure 34. Current-mode multiplier schematic: Circuit schematic showing the j^{th} row for a fully–differential current–mode vector–matrix multiplier.

Also, cascoding creates a high impedance at the output as well. On account of these reasons, a cascoded version of the multiplier was implemented.

6.3 Experimental Results

A prototype vector matrix multiplier has been fabricated in a $0.5\mu m$ CMOS process using the four-quadrant multiplier presented earlier. An array of 128×32 floating-gate transistor elements has been used for the implementation. A block diagram of the VMM system is shown in Figure 35. The floating-gate transistors and the digital circuitry required for programming the floating-gate transistors have been fabricated on-chip while the current subtraction circuitry and $I - V$ converters have been implemented off-chip.

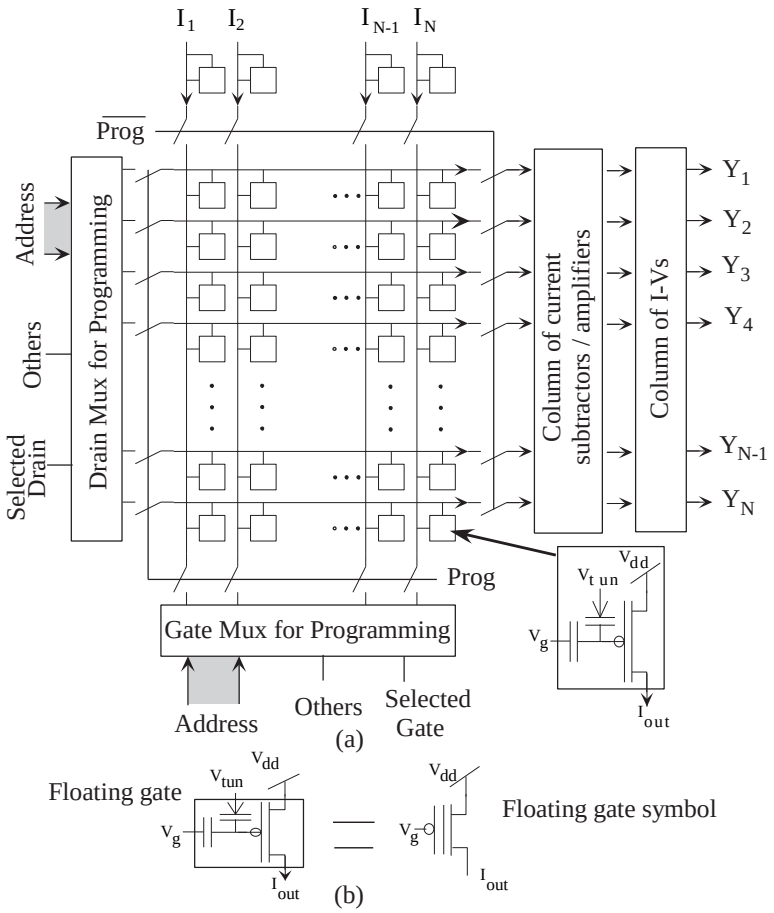


Figure 35. Block diagram of chip: (a) The chip consists of a 128x32 array of floating–gate vector matrix multiplier elements, peripheral digital control for isolation of floating–gate elements during programming, and current amplifiers; (b) Symbol used for a floating–gate (FG) device.

The VMM chip affords the flexibility of configuring the system as either a two-quadrant or a four-quadrant multiplier for both positive and negative weights. Different rows were programmed to different weights and all the weights in one particular row were programmed identical. Figure 36 demonstrates two-quadrant multiplication while Figure 37 demonstrates four-quadrant operation in the multiplier.

The linear range of the multiplier can be estimated from Figure 38 that shows the differential output current vs. the input current for various positive weights. The linearity is measured to be greater than two decades, beyond which the multiplier

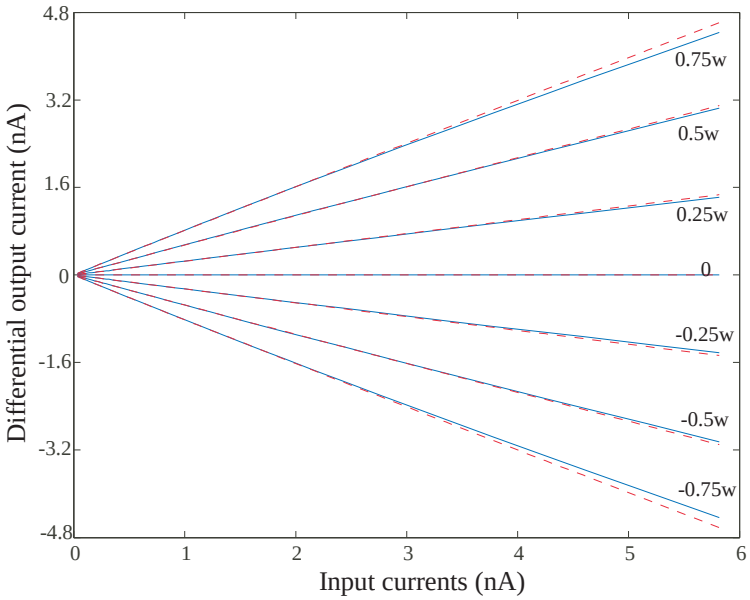


Figure 36. Measured Results For Two-Quadrant Multiplier: Plot of measured differential output current vs. input current on a linear scale, for two-quadrant configuration.

deviates from the ideal linear curve with an error that is higher than 2.5%. As explained earlier, this linearity limitation is partly due to the difference in κ between identical transistors programmed to different currents and the variation of κ with the gate voltage. This effect can be alleviated by programming the elements relatively close to each other. Figure 38 also emphasizes the point that a current-mode implementation gives decades of linearity in signal swing that is especially hard to obtain in voltage-mode circuits without consuming more power.

A custom PCB was fabricated to perform speed measurements for low input currents. Figure 39 shows the measured and simulated frequency response for different DC input currents. The measured corner frequencies (f_{-3dB}) match closely to the simulated results. The plot shows that the VMM would easily operate up to 10MHz if it was not limited by the frequency response of the I-to-V converter (Bandwidth = 5MHz) at the output. Figure 40 shows a plot of measured corner frequencies with the input DC bias current on a log-log scale. The data points follow a straight line with a slope of 1 as expected in weak inversion. The deviation for higher current levels

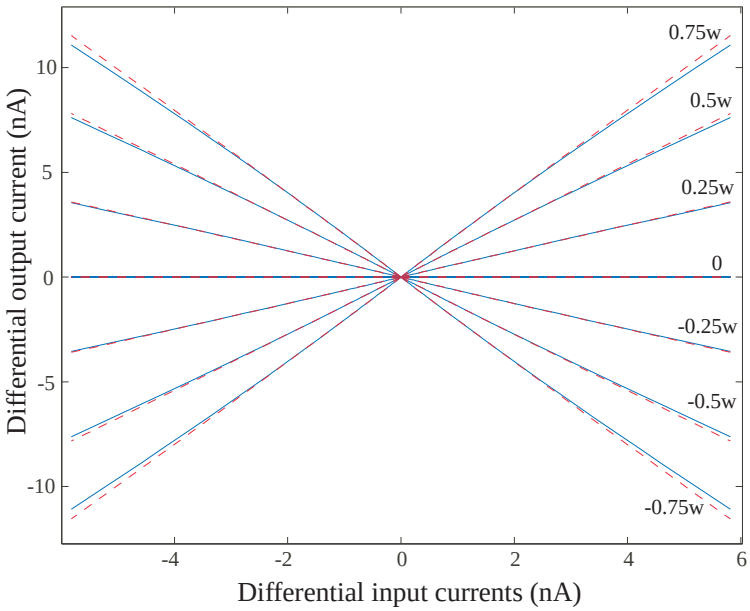


Figure 37. Measured Results For Four-Quadrant Multiplier: Measured differential output current vs. differential input current for four-quadrant configuration.

is due to the transistor moving from weak inversion regime to the strong inversion region. The bias currents required for a bandwidth of 1MHz and 10MHz are 40nA (measured) and 512nA (simulated), respectively for each Floating-gate device. The VMM chip required 531nW/MHz (from Figure 40) for each differential cell clearly demonstrating the speed vs. power tradeoff. The DC bias current however can be set solely on the basis of speed requirements as the Signal-to-Noise Ratio (SNR) is independent of the input DC bias level. The SNR however is directly proportional to the Gate-Source Capacitance (C_{gs}) and can be increased at the expense of chip area.

The VMM chip can be used for applications like audio and video processing. The VMM architecture was configured to perform real–time block matrix transforms of input images in a row–parallel manner. The weights were programmed to be the DCT kernel. Figure 41(a) shows the image that was placed as an input to the chip. To estimate the performance of the VMM, the programmed weights were first measured and the block DCT (8×8) was performed off-chip. Figure 41(b) shows the image

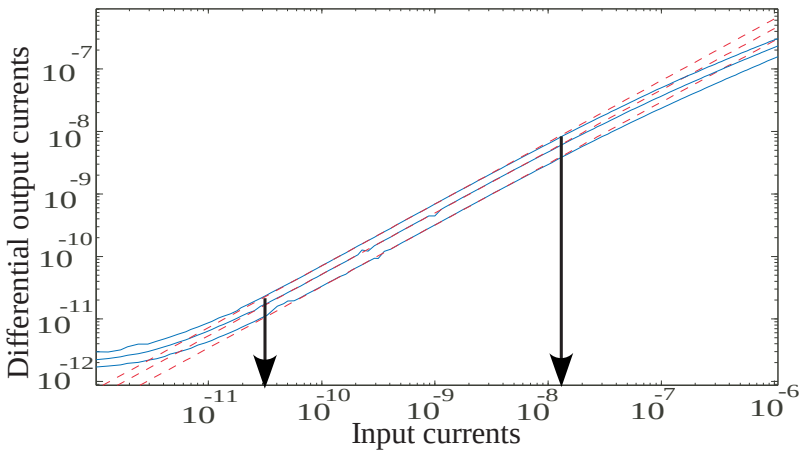


Figure 38. Multiplier Linearity: Plot showing the limits of linearity for the current-mode configuration for the two-quadrant configuration.

obtained after inverse transformation. Next, the block transform was performed on-chip and the result is shown in 41(c). It can be observed that the results for part (b) and (c) are similar thereby demonstrating the usefulness of the VMM architecture. The distortion observed in both the images are due to the programming accuracy limitations (0.2% error).

6.4 Comparisons To Alternate Techniques

Owing to the power savings when the VMM is operated in the analog domain, several researchers have attempted VMM implementation in the analog domain. Multiplication implemented in the voltage-mode has been the popular approach. In [58], multiplication is achieved by using MOS transistors in the triode region, thereby making them sensitive to drain-source variations. An alternate approach has been to use MOS transistors operating in saturation based on 'quarter-square algebraic identity'. The design comprises of at least 12 transistors [59], dual-input floating-gate MOS that requires two capacitors per cell and has offsets in the final results that have to be corrected for off-chip [55]. Moreover, all of the above operated at slower speeds, had a high power dissipation due to limitations in linearity on account of

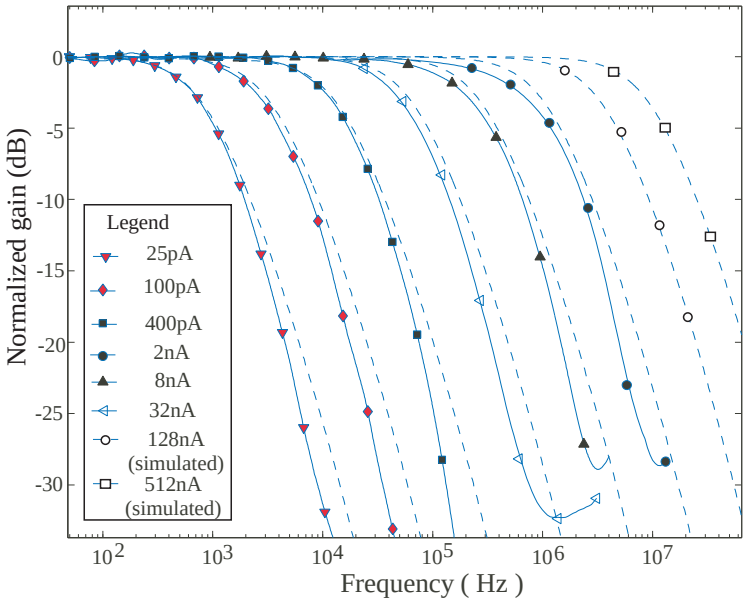
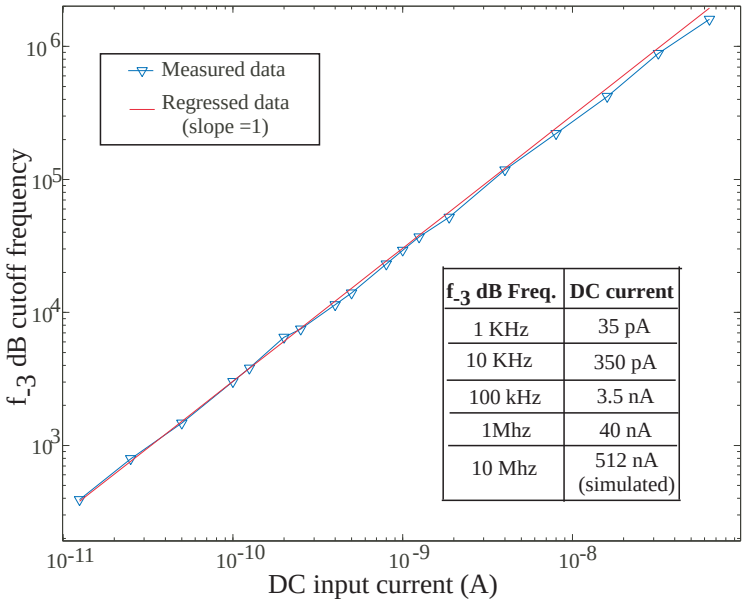


Figure 39. Frequency response: Plot of frequency response of current mode multipliers. The solid lines represent measured data while dashed lines represent simulation results.

being a voltage-mode implementation.

Aside from multiplication, weight storage is another key issue. Previous implementations have used some modification of EEPROM cells [55] or some variation of multiple-input floating-gate transistor for analog storage [60]. The programming schemes used were primarily tunneling based. On the other hand, in this work, the weight storage and multiplication occur at the same site and hence leads to a very compact implementation. Also, the programming technique allows for fast and accurate programming [2].

Table 6.4 summarizes the performance of the VMM along with that of [55]. The chip operates on a $3.3V$ power supply and displays a power dissipation of $531nW/MHz$ when operated in the weak inversion region. Operating at $10MHz$ results in the multiplier cell operating in the strong inversion region and dissipates $7.2\mu W$. The approach in [55] on the other hand dissipates $0.39mW$ of power for a $60KHz$ operation. The bulk of the power is expended in an effort to increase the linear range. The linear range for the multiplier in this work is on the order of decades



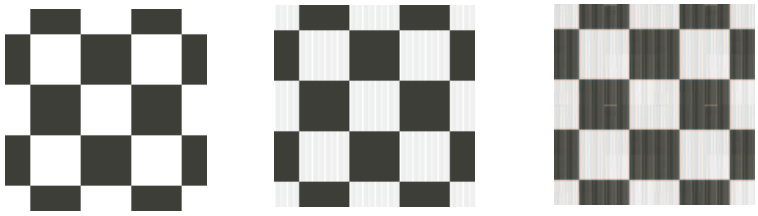
f_{-3} dB Freq.	DC current
1 KHz	35 pA
10 KHz	350 pA
100 kHz	3.5 nA
1Mhz	40 nA
10 Mhz	512 nA (simulated)

Figure 40. Frequency response: Variation of f_{-3dB} cut–off frequency vs. DC input current (per FG device) is plotted. For subthreshold currents a linear relationship is observed, as expected. The table shows the measured DC input current (per FG device) required for various f_{-3dB} cut–off frequency.

while a linear range of $3V$ is achieved in [55]. Figure 42 shows the micrograph of the VMM chip that was fabricated in a $0.5\mu m$ N-well CMOS process. The chip area is $0.83mm^2$ for an array size of 128×32 floating-gate elements. As can be observed, the proposed architecture is both power and area efficient.

6.5 Summary

A programmable fully-differential current-mode VMM architecture has been described in this chapter. The use of floating-gate transistors provide programmability to the multiplier with the result that a continuous set of weights can be implemeted in the multiplier. A prototype chip has been fabricated in a $0.5\mu m$ standard CMOS process with an array of 128×32 floating-gate elements. The architecture is suitable for low voltage, low power applications and has a bandwidth-to-frequency ratio of 531nW/MHz per differential multiplier cell. A linearity of over two decades has been reported for the proposed multiplier. For a bandwidth of less than 10MHz,



(a) (b) (c)

Figure 41. 8x8 block DCT of a 128x128 image: (a) Original input image; (b) Image after inverse DCT, when block matrix transformation was performed off–chip, using the measured weight matrix from the VMM chip. (c) Output of the VMM chip (after inverse DCT) for 8x8 block transform that was performed on–chip.

Table 8. VMM Summary of Performance

Parameter	**Proposed VMM**	**VMM in [55]**
Technology	0.5μm N-Well CMOS	1.5μm single poly CMOS/EEPROM
Power Supply	3.3V	5V
FG Dim.(W/L)	18λ / 4λ	N/A
Array size	128×32	16×16
Chip area	0.83mm^2	1mm^2
Programming % error	$< \pm 0.2\%$	<10mV
BW/power per cell	531 nW/MHz	N/A
Linearity	$>$ 2 decades	3V
Power per cell	7.2 μW @10MHz	0.39mW @60KHz
Programming scheme	Injection & Tunneling	N/A
Programming Time per W_{ji}	1mS	100mS

this architecture is capable of performing 1 million Multiply-Accumulate (MAC) operations/0.27μW that is orders of magnitude lower power when compared to digital approaches. The approach is advantageous from an area standpoint as well. An array of 128×32 floating-gate transistors comprising the multiplier occupies an area of $0.83mm^2$. The use of floating-gate transistors result in a non-volatile storage of the weight information. Extrapolations from a thermionic emission model for charge loss indicate good performance in the $< 90°C$ temperature range over a 10 year time period for a range of weight values. Approaching the multiplication in the current domain has been advantageous from both power dissipation and linearity standpoint.

Figure 42. Chip Die Photograph: The die photograph of the chip containing an array of 128×32 floating-gate transistors implemented on a $0.5\mu m$ CMOS process showing the compactness of the proposed approach.

The VMM plays a key role in enabling adaptive signal processing and its role in an adaptive filter framework will be discussed in the next chapter.

CHAPTER 7
ADAPTIVE SIGNAL PROCESSING

Adaptive filters are highly non-linear, time varying systems that are useful in a variety of tasks that are self-optimizing in nature. Applications for adaptive filters include adaptive equalization, prediction, system identification and noise cancellation [61]. Traditional implementations of adaptive filters have been in the digital domain. Adaptive filter design in the analog domain is motivated by the benefits of a low-power and compact implementation. Aside from adaptation and addition, multiplication is another operation that is performed repetitively. And, as has been demonstrated earlier in chapter 6, multiplication is both power and area efficient using an analog approach. Also, as will be shown in this chapter, adaptation can be achieved in a compact and low-power fashion in the analog domain.

An analog approach to building synapses and adaptive filters have been attempted by a number of researchers [62, 63, 64, 65, 66, 67, 68]. In all of these approaches, except [65], the weight adaptation was performed off-chip followed by an operation that transferred the learnt weights on-chip. The weight storage was either on on-chip capacitors requiring constant refreshes [68, 67] or on floating-gate transistors [62, 66] or in the digital form requiring digital-to-analog converters for multiplication using an analog multiplier [65] or using charge coupled devices [64]. The approach in [65] implemented the learning algorithm on-chip with weights stored on capacitors. The approach used analog building blocks to directly implement the mathematics behind the learning algorithm with the result that the circuits were area intensive.

Exploiting the non-linearities inherent in hot-electron injection and Fowler-Nordheim tunneling, weight adaptation can be achieved using floating-gate transistors. Also, the use of floating-gate transistors provides a non-volatile storage capability for the

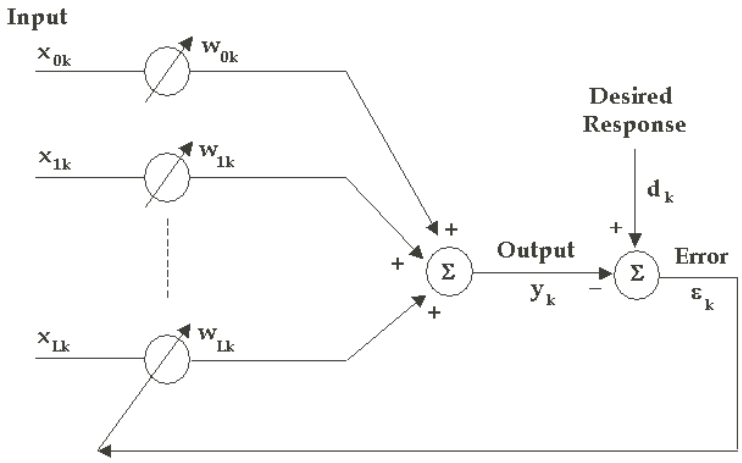


Figure 43. Adaptive Linear Combiner: Block diagram representation of an adaptive linear combiner that adapts its weights such that the error between its output and the target signal is minimized.

weights. Using floating-gate transistors for both weight adaptation and weight storage results in the synapse circuits being compact and low-power [69, 70]. This chapter demonstrates a fully integrated, compact, current-mode, floating-gate based analog approach towards implementing adaptive filters. The system learns using the Least-Mean-Square (LMS) learning algorithm.

7.1 Adaptive Filters and LMS Learning

Figure 43 shows the block diagram of a fundamental functional block, namely, an adaptive linear combiner, also known as an adaptive node. An adaptive node consists of a number of synapses with each synapse performing a multiplication of the input signal with a weight. The weight is learnt as a result of trying to minimize an error function. Weight adaptation is obtained by comparing the output of the adaptive node to a desired target signal and changing the weights of each synpase such that the error between the target and the system output is minimized.

An adaptive matrix is realized by utilizing a number of adaptive nodes. Each node has a unique output and a unique target signal while the inputs may or may

not be shared across the nodes. The output of the k^{th} node of such a system is given by,

$$y_k = \sum_{j=0}^{L} w_{jk} x_{jk} = {w_k}^T x_k \tag{154}$$

where, x_k is the input signal vector of the k^{th} node and w_k is weight vector of the k^{th} node. The above equation is referred to as feedforward computation with the error signal fedback to the synapses of a given node (k) given by,

$$e_k = d_k - y_k \tag{155}$$

where, d_k is the desired target for a given node.

The weight adaptation can be mathematically modelled as,

$$\tau \frac{dw_k}{dt} = f(w_k, x_k {e_k}^T) \tag{156}$$

where, τ is the adaptation rate. A number of learning algorithms fall under the above mathematical model with each differing in the choice of the function $f(\cdot)$ and are broadly classified into two categories, namely, supervised and unsupervised. Supervised algorithms adapt the weights based on the input signal and the error between the actual system response and a desired output signal. Unsupervised learning on the other hand depends entirely on the input signal.

The least-mean-square (LMS) learning rule is a supervised learning algorithm that results from the minimization of a least-square-error objective function. Some LMS algorithms intentionally incorporate weight decay, that is a form of 'forgetfulness' exhibited by learning systems. This is useful for better learning generalization and tracking of non-stationary signals [61]. Analytical modeling of learning rules yields the following weight dynamics and steady-state solutions for an LMS algorithm with and without weight decay [61]:

$$\begin{array}{rll} & \text{No Weight Decay} & \text{Weight Decay} \\ \text{Dynamics:} & \tau \frac{d\mathbf{w}}{dt} = \mathbf{x}e & \tau \frac{d\mathbf{w}}{dt} = \mathbf{x}e - \epsilon\mathbf{w} \\ \text{Steady-state:} & \mathbf{w}_{ss} = \mathbf{Q}^{-1}\mathbf{r} & \mathbf{w}_{ss} = (\mathbf{Q} - \mathbf{I}\epsilon)^{-1}\mathbf{r} \end{array} \tag{157}$$

The steady-state solution, $\mathbf{w}_{ss}$, depends on the input auto-correlation matrix, $\mathbf{Q} = E\left[\mathbf{x}\mathbf{x}^T\right]$ and the input-output cross-correlation vector $\mathbf{r} = E\left[\mathbf{x}y\right]$. The strength of the weight decay, ϵ, should be small ($\epsilon \ll 1$) relative to the input and learning-signal amplitudes to minimize deviation from the ideal LMS solution [71].

7.2 Analog Implementation of Adaptive Filters

The block diagram of the analog adaptive filter system is shown in Figure 44. The system implements the adaptive linear combiner as described earlier. Each adaptive node consists of a number of synapse elements and by combining a number of such nodes, an adaptive matrix can be realized. Figure 44 shows an adaptive matrix that consists of 4 adaptive nodes with each node consisting of 16 synapses. A total of 16 analog input signals can be provided with the signals being shared between all four adaptive nodes while the target signal is unique for each node. The output of each node is compared with the target signal and the error signal is fed back to each of the synapses in a given node. The weights of the synapse adapt in such a way that the error signal is minimized with the result being that the output of each node tracks the target signal. The system employs current-mode signalling with current-voltage (I-V) and voltage-current (V-I) converters forming the interface.

The key circuits that are essential to an adaptive filter can be identified from the block diagram representation shown in Figure 44. These include the synapse along with the post-distort circuitry, single-ended voltage to differential current converter, current-mode high pass filter, current-to-voltage converter and voltage-to-current converter. In this section, each of these circuits will be described in detail.

7.2.1 Floating-gate Synapse

The floating-gate synapse [71] that implements the least-mean-square learning rule is shown in Figure 45. The synapse is an extension of earlier work done in single transistor learning synapses that exploits the inherent physics of tunneling and hot-electron

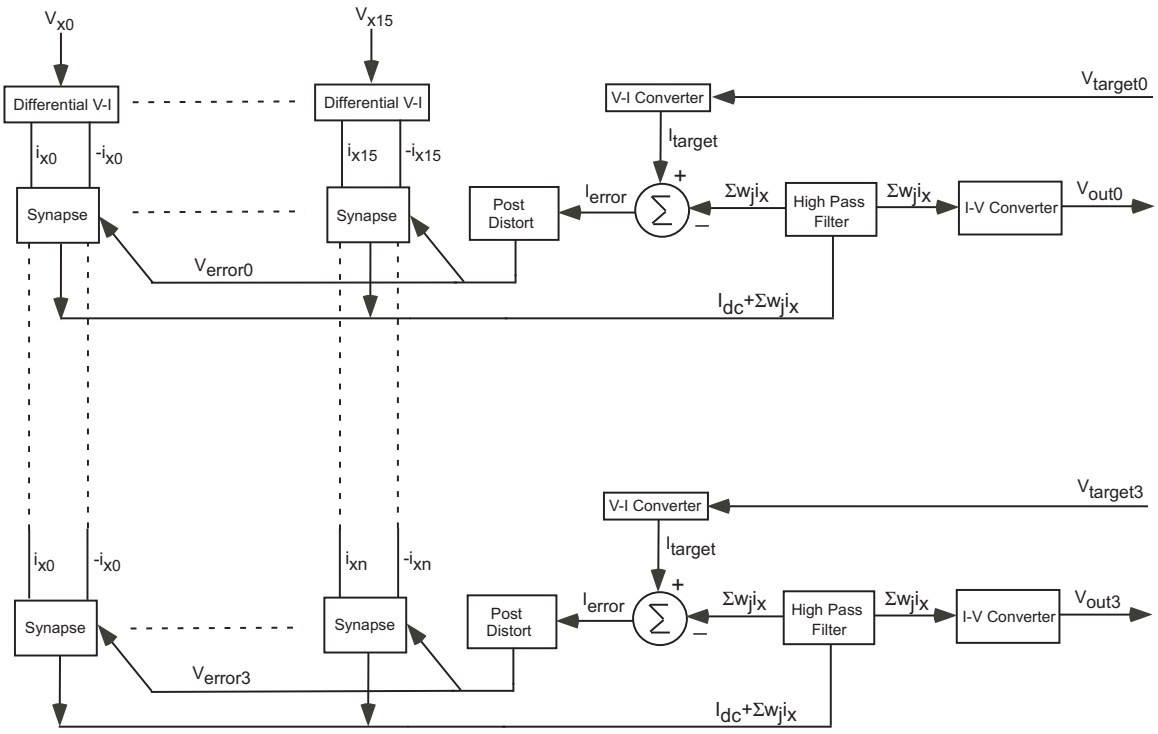


Figure 44. Adaptive Filter System block diagram: Block level representation of the analog implementation of adaptive filtering.

injection to achieve learning. This approach differs substantially from popular approaches of implementing learning algorithms using traditional circuit building blocks. The result is a compact and power-efficient synapse that is amenable to use in a large synaptic array.

Transistors $M1 - M7$ form the actual synapse while the post-distort circuitry plays a function that is different from that of the synapse. The post-distort circuitry is shared across a row of synapses as shown in Figure 44. Transistor pairs $M1/M2$ and $M6/M7$ form a floating-gate current mirror and implement a differential synapse such that both positive and negative weights can be realized. It should be noted that adaptation occurs only at the floating-gate of $M1$ while the floating-gate of $M6$ is programmed to an equilibrium weight that acts as a reference. The drain currents of $M1$ and $M6$ are summed together and become the synapse output.

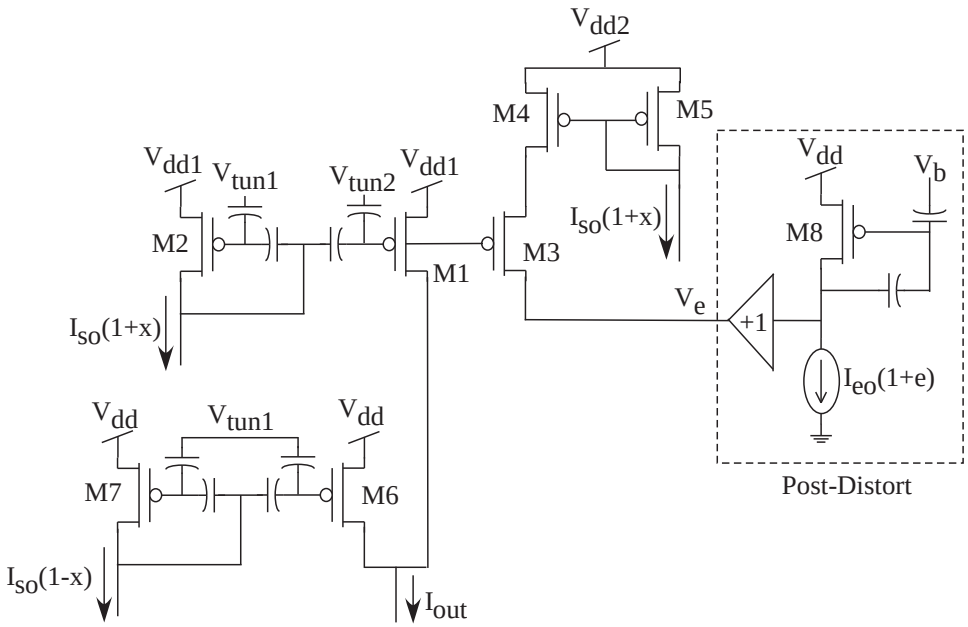


Figure 45. Floating-Gate Synapse: Circuit schematic of the floating-gate synapse circuitry. Transistors $M1-M7$ form the synapse element while the post-distort circuitry is common to each adaptive node (comprised of a number of synapses with their outputs summed together).

The synapse is designed such that two distinct operations are performed simultaneously. These include multiplication and weight adaptation. Transistor pair $M1/M2$ perform a current-mode multiplication of the input signal x with the multiplication co-efficient being set by the charge difference between their floating-gates. Similarly, transistors $M6$ and $M7$ multiply the inverse of the input signal $-x$ by a multiplication co-efficient set by the charge difference between their floating-gates. Transistors $M3 - M5$ along with the post-distort circuitry are responsible for weight adaptation. During normal operation, the tunneling voltage V_{tun2} is held high enough for tunneling to occur at the floating-gate of $M3$ and the chip is ramped up such that hot-electron injection occurs as well. The biasing for all other transistors is such that neither tunneling nor hot-electron injection occurs.

Before proceeding to analytically describe feedforward computation and weight adaptation in the synapse, it is worthwhile to establish the convention followed for

representing signals. The signals are split into a fixed component and a time-varying signal component. For example, the input signal is defined as,

$$I_s = I_{so}(1 + x) \tag{158}$$

where, I_{so} represents the DC component of the input signal and x is the dimensionless AC component.

7.2.1.1 Feedforward Computation

In order to mathematically express the feedforward behavior, consider transistor pair $M1/M2$. Using the above described convention for representing signals, the $I - V$ relationship of transistor $M2$, ignoring Early effects and assuming weak inversion can be expressed as,

$$I_{so}(1 + x) = I_o exp\Big(\frac{-\kappa_{eff} V_g}{U_T}\Big) exp\Big(\frac{-\kappa Q_2}{C_T U_T}\Big) exp\Big(\frac{V_s}{U_T}\Big) \tag{159}$$

where, Q_2 is the charge on the floating-gate. Similarly, the drain current of $M1$ is given by,

$$I_1 = I_o exp\Big(\frac{-\kappa_{eff} V_g}{U_T}\Big) exp\Big(\frac{-\kappa Q_1}{C_T U_T}\Big) exp\Big(\frac{V_s}{U_T}\Big) \tag{160}$$

It should be noted that both the transistors share the same gate voltage and the source voltage. Also, assume that the two transistors match. Now, substituting the expression for the gate voltage from (159) in (160), the drain current of $M1$ is given by,

$$I_1 = I_{so}(1 + x) exp\Big(\frac{-\kappa(Q_1 - Q_2)}{C_T U_T}\Big) \tag{161}$$

Now, defining the weight as,

$$w = exp\Big(\frac{-\kappa(Q_1 - Q_2)}{C_T U_T}\Big) - 1 \tag{162}$$

and letting the weight of transistor pair $M1/M2$ be the positive weight, the drain current of $M1$ is given by,

$$I_1 = I_{so}(1 + x)(1 + w^+) \tag{163}$$

Now, assume that the floating-gate of $M6$ is programmed to a charge of Q_2 as well and that the transistor pair is programmed such that a negative w^- results. Therefore, the drain current of $M7$ can be written as,

$$I_7 = I_{so}(1-x)(1+w^-) \tag{164}$$

Adding the above two equations results in the output of the synapse and is as given below.

$$I_{out} = I_{so}(2+w+wx) \tag{165}$$

where, the weight w is defined as $w^+ - w^-$. As can be observed from (154), the quantity of interest in the above equation is the term containing wx. This is obtained by high-pass filtering the output current. This is possible because of the seperation of time-scales between the weight adaptation and the input signal. The weight adaptation is on account of tunneling and injection currents charging and discharging the floating-gate capacitance. Typically, these currents are very small and so the weight adaptation is a slow process (on the order of a few hundred mHz). Therefore, using input signals that are atleast two orders of magnitude higher than weight adaptation results in a good seperation of timescales and therefore easier to high-pass filter.

7.2.1.2 Weight Adaptation

Analyzing weight adaptation begins by first applying KCL at the floating-gate node and noting that tunneling and hot-electron injection are continuously enabled. This results in the following,

$$C_T\frac{dV_{fg}}{dt} = C_1\frac{dV_g}{dt} + C_2\frac{dV_d}{dt} + I_{tun} - I_{inj} \tag{166}$$

where, C_1 is the input capacitance, C_2 is the parasitic floating-gate to drain capacitance, C_T is the total floating-gate capacitance, I_{tun} is the tunneling current and I_{inj} is the hot-electron injection current. In the above equation, it has been assumed that the tunneling voltage is held fixed.

At this point, the concept of seperation of timescales will be introduced. It should be noted that in an adapting system such as the one being described, signals with widely seperated timescales are encountered. The weight adaptation is a slow process while the input signals are relatively faster. With this in mind, signals will be split into its DC component and an AC component with the AC component being further split into a fast moving component and a slow moving component. For example, the floating-gate voltage can be written as,

$$V_{fg} = V_{fg0} + \tilde{V}_{fg} = V_{fg0} + \bar{V}_{fg} + \Delta V_{fg} \tag{167}$$

where, V_{fg0} is the DC component, $\bar{V}_{fg}$ is the component on the slower timescale, ΔV_{fg} is the component in the faster timescale and $\tilde{V}_{fg}$ is the combined AC component comprising of the slow and fast timescale components.

Equation (166) is a general equation valid in both the slow and fast timescales. As mentioned earlier, weight adaptation is a process that occurs on a timescale much slower than that of the signals applied to system. Assuming that the signals applied to the system have a zero mean, the changes in the floating-gate voltage are entirely on account of tunneling and injection currents. Therfore, considering a slow timescale, (166) can be represented as,

$$C_T \frac{d\bar{V}_{fg}}{dt} = I_{tun} - I_{inj} \tag{168}$$

It should be pointed out that considering (166) in fast timescale yields the feedforward computation equations presented earlier.

It is clear that in the class of synapse considered, weight adaptation depends entirely on Fowler-Nordheim Tunneling and Hot-electron injection currents. Therefore, it is important to consider the dependence of these currents on terminal voltages. These equations have been presented earlier in chapter 3 and are repeated here for convenience. An approximate expression for the tunneling current is given by,

$$I_{tun} = I_{tun0} exp\Big(\frac{V_{tun} - V_{fg}}{V_x}\Big) \tag{169}$$

where, I_{tun0} is an equilibrium tunneling current and V_x is a device-dependent parameter that is a function of the bias voltage across the oxide. Next, assume that the tunneling voltage is held constant. The tunneling current equation can be modified as,

$$I_{tun} = I'_{tun0} exp\Big(\frac{-\tilde{V}_{fg}}{V_x}\Big) \tag{170}$$

where, I'_{tun0} contains all the DC components. Decomposing the floating-gate voltage into its slow and fast timescale components results in,

$$I_{tun} = I'_{tun0}(1+w)^{\beta-1}(1+x)^{\beta-1} \tag{171}$$

where, β is given by,

$$\beta = 1 + \frac{U_T}{\kappa V_x} \tag{172}$$

A simplified model for hot-electron injection that is based on the channel current (I_{ch}) and the change in the drain-source voltage (ΔV_{ds}) is given by,

$$I_{inj} = I_{inj0}\Big(\frac{I_s}{I_{s0}}\Big)^{\alpha} exp\Big(\frac{-\Delta V_{ds}}{V_{inj}}\Big) \tag{173}$$

where, I_{inj0} is the injection current when the channel current is I_{s0}, V_{inj} is a device and bias dependent parameter and α is defined to be,

$$\alpha = 1 - \frac{U_T}{V_{inj}} \tag{174}$$

Noting that transistor $M1$ is a source follower, the above equation can be re-written as,

$$I_{inj} = I_{inj0}(1+x)^{\alpha} exp\Big(\frac{\kappa\tilde{V}_{fg}}{V_{inj}}\Big) exp\Big(\frac{-\Delta V_d}{V_{inj}}\Big) \tag{175}$$

As before, expanding the floating-gate voltage into its slow and fast timescale components and relating the slowly changing component to $(1+w)$ and the fast moving component to $(1+x)$ the above equation can be modified as,

$$I_{inj} = I_{inj0}(1+x)^{\gamma}(1+w)^{-U_T/V_{inj}} exp\Big(\frac{-\Delta V_d}{V_{inj}}\Big) \tag{176}$$

where, γ is equal to $\alpha - U_T/V_{inj}$.

Next, with regards to the term containing the change in the drain voltage (ΔV_d), consider the post-distort circuit shown in Figure 45. The drain voltage of the adapting floating-gate transistor is the output of the post-distort circuitry. The input to the post-distort circuitry is the error current ($I_{e0}(1+e)$). This input current is applied to a floating-gate transistor with an input capacitor C_1 that can be adjusted by digitally selecting from a capacitor bank. This is done to ensure that the exponential term containing ΔV_d can be linearized to equal $(1+e)$. Solving the $I-V$ relationship of the post-distort circuitry, one can write an expression for $-\Delta V_d/V_{inj}$ as,

$$\frac{-\Delta V_d}{V_{inj}} = \frac{U_T C_T}{C_1 \kappa V_{inj}} ln(1+e) \tag{177}$$

Assume that the capacitor ratio C_1/C_T is set to equal $U_T/\kappa V_{inj}$. Using the above assumption, (176) can be re-written as,

$$I_{inj} = I_{inj0}(1+x)^{\gamma}(1+w)^{-U_T/V_{inj}}(1+e) \tag{178}$$

Differentiating the weight as defined in (162) with respect to time and performing some algebraic manipulations, the above equation can be rewritten as,

$$\frac{U_T C_T}{\kappa}\frac{dw}{dt} = (1+w)(I_{inj} - I_{tun}) \tag{179}$$

Now using the expressions for tunneling and injection in the above expression, the weight update equation can be written as,

$$\tau\frac{dw}{dt} = (1+w)^{\alpha}(1+x)^{\gamma}(1+e) - (1+w)^{\beta}(1+x)^{\beta-1} \tag{180}$$

In order to express the above equation in a way that is consistent with the typical LMS weight update equation, assume that the input signals are ergodic. This assumption enables one to use time averages to calculate the expected value of the signal. The expected value of a signal $x(t)$ is given by,

$$E[x(t)] = \frac{1}{T}\int_0^T x(t)dt \tag{181}$$

It will be further assumed that the time interval T is much shorter than the timescale in which weight adaptation occurs but longer than the fast timescale. Using the above assumption in conjunction with a Taylor series approximation for the term containing x and evaluating $E[\cdot]$, the weight update equation can be re-written as,

$$\tau \frac{dw}{dt} = (1 + \gamma E[xe])(1 + w)^{\alpha} - (1 + \frac{1}{2} a_1 E[x^2])(1 + w)^{\beta} \tag{182}$$

Next, applying a Taylor series approximation to the terms containing w results in,

$$\tau \frac{dw}{dt} = \gamma E[xe] - a_1 E[x^2] + w(\gamma E[xe] - \epsilon - a_1 \beta E[x^2]) \tag{183}$$

where, $a_1 = \frac{U_T}{\kappa V_x}$ and $\epsilon = \frac{U_T}{\kappa V_x}$. In most cases, the input variance terms are small enough to be ignored and therefore, the above equation can be simplified to be,

$$\tau \frac{dw}{dt} = \gamma E[xe] + w(\gamma E[xe] - \epsilon) \tag{184}$$

The above equation represents the LMS learning rule. Therefore, theoretically, the floating-gate synapse implements the LMS learning rule.

7.2.2 Single-Ended Voltage to Differential Current Converter

It is clear from the discussion on the floating-gate synapse that there is a need for differential currents. From a system perspective, it will be advantageous to communicate with the external world in terms of voltages. With this in mind, Figure 46 shows the schematic of a circuit that converts a single-ended voltage signal into a differential current. These currents form the inputs to the floating-gate synapse described earlier. It is clear from Figure 46 that a simple differential pair is sufficient for the purpose of generating differential currents.

In order to quantitatively analyze the circuit behavior, consider small signal operation. Ignoring mismatch between the transistor pair $M1/M2$, the drain currents of $M1$ and $M2$ are both equal to $I_1/2$ when V_{in} is equal to V_{ref}. Now, assume that the voltage input V_{in} is such that an input signal is applied around the DC value of

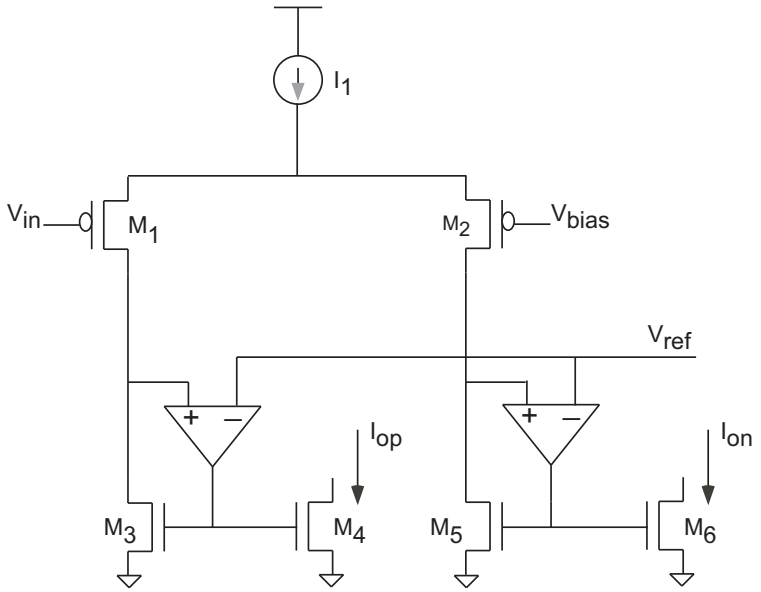


Figure 46. Single-ended Voltage to Differential Currents: Circuit schematic of the single-ended voltage to differential current converter.

V_{ref}. For this case, the voltage input V_{in} can be represented as,

$$V_{in} = V_{ref}(1 + x) \tag{185}$$

where, as before, x is a dimensionless AC signal component of V_{in}. Since small-signal behavior has been assumed, the drain current of $M1$ is given by,

$$I_1 = \frac{I_1}{2} + g_{m1} V_{ref} x = \frac{I_1}{2}(1 + y) \tag{186}$$

where, $g_{m1} = g_{m2} = g_m$ is the small-signal transconductance of transistors $M1/M2$ biased at a drain current of $I_1/2$ and y is the dimensionless signal component of the drain current I_1. Using the above expression for the drain current of $M1$ and applying KCL at the source of transistors $M1$ and $M2$, the drain current of $M2$ can be written as,

$$I_2 = \frac{I_1}{2}(1 - y) \tag{187}$$

From the above two equations, it is clear that a differential output current is generated using the circuit of Figure 46.

Thus far in the analysis, Early effects have been ignored. On account of the finite output impedances of the transistors $M1$ and $M2$, any difference in the drain voltages of these two transistors will result in distortion in the output currents. In order to alleviate this problem, the current mirror formed by transistor pairs $M3/M4$ and $M5/M6$ along with the amplifiers $A1$ and $A2$ are employed. Negative feedback is applied via the amplifiers to ensure that the drains of $M1$ and $M2$ are held fixed at V_{bias}. This ensures that distortion in the currents I_1 and I_2 is substantially reduced. Also, the current mirrors are designed to be $1:1$ mirrors such that I_{op} and I_{on} are equal to I_1 and I_2 respectively.

7.2.3 Current-Mode High-Pass Filter

The circuit schematic of the log-domain current-mode high-pass filter is shown in Figure 47. Qualitatively, the circuit operation can be explained by noting that a change in the input current changes the source voltage of transistor $M1$ on account of its gate voltage being fixed. Note that transistors $M1$ and $M3$ share the same source terminal. The current through transistor $M3$ is fixed at I_τ and so a change in the source voltage of $M3$ changes its gate voltage. The circuit is designed such that the current I_τ is a much smaller current compared to the input and an explicit capacitance C is connected to the gate of $M3$. This results in the change in the gate voltage of $M3$ to be a low-pass filtered version of the change at its source. Following the signal along the circuit, assuming that transistors $M3$ and $M4$ are matched and so are their bias currents, the low-pass filtered gate voltage results in a change in the source voltage of $M4$. Again, observing that $M4$ and $M6$ share the same source and noting that $M1$ and $M6$ are identical with identical gate voltages, the drain current of $M6$ is a low-pass filtered version of the input signal. Subtracting this from a copy of the input signal results in the high-pass filtered version of the input signal.

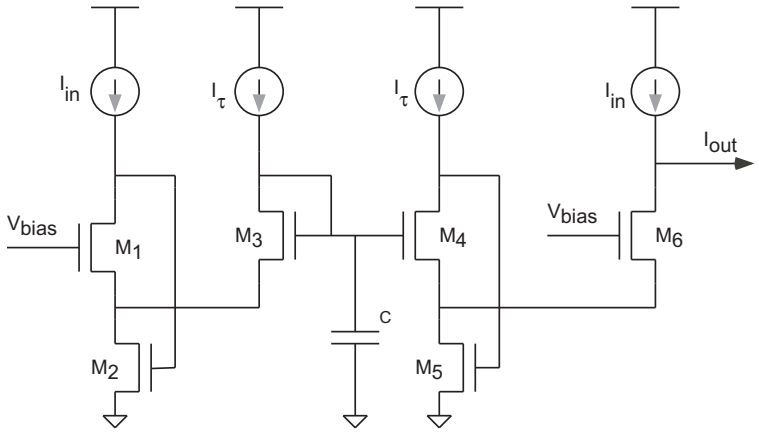


Figure 47. Current-mode High Pass Filter: Circuit schematic of the current-mode high pass filter is shown. The bias current I_τ sets the high pass filter corner frequency.

7.2.3.1 Small-Signal Behavior

In order to arrive at a simple mathematical expression quanitfying the behavior of the circuit in the small-signal domain, assume saturation and neglect all parasitic capacitances in the circuit. The current-to-voltage transfer function from the input to the source of $M1$ is given by,

$$-\frac{V_1}{I_{in}} \approx \frac{1}{g_{m1}} \tag{188}$$

Following the signal path, the gate voltage of $M3$ is modulated as,

$$V_2 = V_1\left(\frac{1}{1+s\tau}\right) = -\frac{I_{in}}{g_{m1}}\left(\frac{1}{1+s\tau}\right) \tag{189}$$

where, the time constant τ is given by,

$$\tau = \frac{C}{g_{m3} + 1/r_{o3}} \tag{190}$$

It is easy to see that the gate voltage of $M3$ is a low-pass filtered version of the input current I_{in}. The low-pass filter corner is set by a combination of the transconductance of $M3$ and an explicitly drawn capacitor C.

The current through transistor $M4$ is held fixed at I_τ and therefore forms a source follower. So, to a first order, one can assume that the transfer function from the gate terminal of $M4$ to its source is equal to 1. Next, consider the voltage-to-current

transfer function from the source of $M4$ to the drain current of $M6$. This transfer function is given by,

$$I_6 = -g_{m6}V_3 \approx -g_{m6}V_2 \tag{191}$$

where, g_{m6} is the transconductance of $M6$. Substituting for V_2 in the above equation results in the drain current of $M6$ being,

$$I_6 = \frac{I_{in}}{1 + s\tau} \tag{192}$$

where, it has been assumed that transistors $M1$ and $M6$ match and they have the same transconductance. Note that the drain current of $M6$ is a low-pass filtered version of the input current I_{in}. The final step in determining the overall circuit transfer function is to subtract the drain current of $M6$ from a copy of the input current. This results in the output current being,

$$I_{out} = I_{in}\frac{s\tau}{1 + s\tau} \tag{193}$$

which is a high-pass transfer function.

7.2.3.2 Effect of Mismatch

In deriving the transfer function for the high-pass filter, it was implicitly assumed that all transistors and current sources were matched. In a practical implementation, however, mismatch is an issue of concern and can be shown to set the upper limit of the noise floor. In the high-pass filter circuitry, errors can occur from a mismatch between the input signal and its copy, mismatch between the two current sources that set the filter corner frequency, device mismatch between transistor pairs $M1/M6$ and $M3/M4$. Note that mismatch between transistor pairs $M2/M5$ need not be considered seperately as they can be absorbed into the mismatch between the I_τ current sources and I_{in}.

When all devices and current sources are perfectly matched, the DC component of the output current is zero. In the presence of mismatch, the DC component of I_{out}

is non-zero and this residual current can be viewed as setting the upper limit for the noise floor. The objective of this analysis is to determine this residual component in terms of various mismatch components. Before proceeding further, assume that the copy of the input current differs from the input by ΔI_{in} and the two I_τ current sources are mismatched by ΔI_τ. Further, only threshold voltage mismatches will be considered.

As before, nodal voltages V_1 and V_2 are set given an input current I_{in} and a current I_τ flowing through the drain of $M3$. The drain current of $M4$ is $I_\tau + \Delta I_\tau$ and assume that its threshold voltage is different from that of $M3$ by ΔV_{th3}. Assuming weak inversion operation, the node voltage V_3 can be written in terms of V_1 as,

$$V_3 = V_1 + \kappa \Delta V_{th3} - U_T ln\Big(1 + \frac{\Delta I_\tau}{I_\tau}\Big) \tag{194}$$

Assuming that the threshold voltage of $M6$ is different from that of $M1$ by ΔV_{th1} and solving for the drain current of $M6$ gives,

$$I_6 = I_{in}\Big(1 + \frac{\Delta I_\tau}{I_\tau}\Big) exp\Big(\frac{\kappa(\Delta V_{th1} - \Delta V_{th3})}{U_T}\Big) \tag{195}$$

The above equation can be approximated as,

$$I_6 = I_{in}\Big(1 + \frac{\Delta I_\tau}{I_\tau}\Big)\Big(1 + \frac{\kappa(\Delta V_{th1} - \Delta V_{th3})}{U_T}\Big) \tag{196}$$

Using the above expression, the residual current (ΔI_{out}) is given by,

$$\Delta I_{out} = I_{in}\Big(1 - \Big(1 + \frac{\Delta I_\tau}{I_\tau}\Big)\Big(1 + \frac{\kappa(\Delta V_{th1} - \Delta V_{th3})}{U_T}\Big)\Big) + \Delta I_{in} \tag{197}$$

As can be observed, in comparison to threshold voltage mismatch between transistor pairs $M1/M6$ and $M3/M4$, the mismatch between the current mirrors play a dominant role in determining the residual output current and therefore should be minimized by proper design.

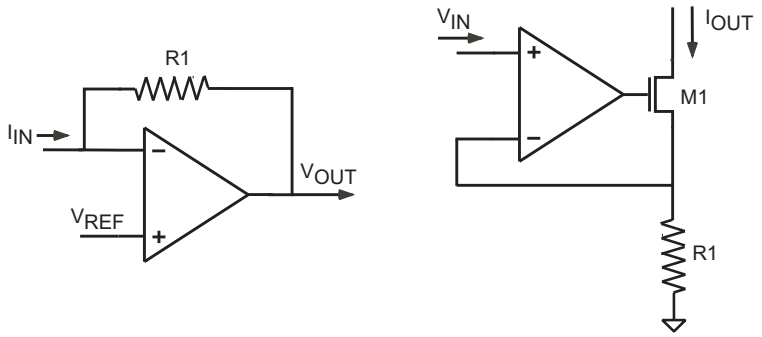


Figure 48. Popular Interface Circuits: (a) Transimpedance amplifier used for $I-V$ conversion. (b) Typical circuitry used for $V-I$ conversion.

7.2.4 Current-to-Voltage Converter

A popular approach to implementing $I-V$ converters is to configure an operational amplifier as a charge integrator. This approach, owing to sampling delays is limited to measuring low frequency currents. A transimpedance amplifier, as shown in Figure 48(a) provides continuous time $I-V$ conversion and is a viable alternative. This approach requires careful consideration to compensation to ensure good performance [72]. Also, measuring small currents on chip is prohibitive owing to the large values of resistors needed. Logarithmic converters using BJTs have a high dynamic range but implement a non-linear current conversion and are not suited for standard digital CMOS processes. The proposed $I-V$ converter described in this section, uses the output impedance of MOS transistors to perform the current conversion [73]. The key issue in using such an approach is the difficulty of biasing the high-gain output node. This is addressed through the use of negative feedback and replica biasing.

Figure 49 shows the circuit schematic of the proposed $I-V$ converter that consists of the core converter, the replica biasing scheme and the current multiplication block that provides current ranging capability. The $I-V$ conversion is performed using transistors $M1$ and $M2$ where transistor $M2$ is a common source amplifier with $M1$ being the active load. For zero signal input, the DC operating point for the high gain output voltage, V_{out} is designed to equal V_{ref} through the use of replica transistor

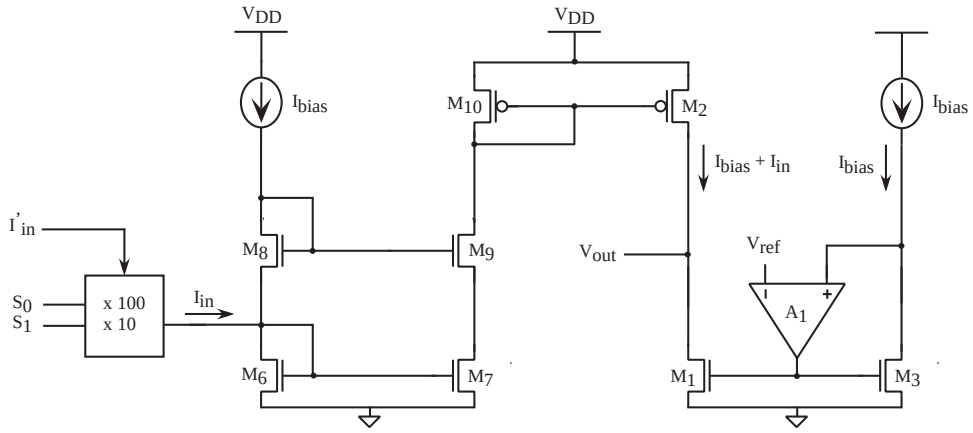


Figure 49. Circuit Schematic of the proposed $I-V$ converter: Transistors $M1-M2$ perform the core $I-V$ conversion while amplifier $A1$ serves to set the DC equilibrium for the high gain output voltage. Switches S_0 and S_1 implement input current multiplications of 100 and 10 respectively to increase the linear range.

$M3$, identical current source I_{bias} and the operational amplifier $A1$. On account of negative feedback, the amplifier $A1$ sets the gate of M3 such that at a drain voltage of V_{ref}, its drain current equals I_{bias}. This ensures that the drain voltage of $M1$ equals V_{ref} as well.

An input current I_{in}, is mirrored through current mirrors $M6/M7$ and $M10/M2$ such that a drain current of $I_{bias}+I_{in}$ flows through $M2$. Since, the current through $M1$ is set to equal I_{bias}, the difference current ΔI_{in} causes a change in the output voltage,(ΔV_{out}) given by,

$$\triangle V_{out} = (r_{o1} \| r_{o2}) \triangle I_{in} = r_o \triangle I_{in} \tag{198}$$

where r_{o1} and r_{o2} are the output impedances of transistors $M1$ and $M2$ respectively. It should be noted that the conversion gain is set by the output impedances of transistors $M1$ and $M2$ and can be designed to be quite large. Also, to a first approximation, the $I-V$ conversion given by (1) is linear.

The non-linearities and hence the distortion introduced can be estimated by utilizing the relationship between the drain current of a transistor and its output impedance. Assuming, a first order MOS model, the change in the output voltage, is

given by,

$$\triangle V_{out} = \left[\frac{1}{\lambda_1 I_{bias}} \| \frac{1}{\lambda_2 (I_{bias} + \triangle I_{in})}\right] \triangle I_{in} \tag{199}$$

Assuming, that the λ's of $M1$ and $M2$ are equal and further assuming that the signal current is much smaller than the bias current I_{bias}, (2) simplifies to,

$$\triangle V_{out} = \frac{\triangle I_{in}}{2\lambda I_{bias}} \left[1 - \frac{\triangle I_{in}}{2I_{bias}}\right] = \triangle I_{in} r_o - \frac{\triangle {I_{in}}^2 r_o}{2I_{bias}} \tag{200}$$

From (3) it is clear that the second harmonic term and hence the distortion is proportional to the input signal amplitude and is inversely proportional to the bias current. This brings about a direct tradeoff between distortion and power dissipation. A differential approach can help eliminate the even order harmonics and lead to lower levels of distortion.

The $I - V$ converter can be approximated to be a single pole system with the dominant pole being at the output node. The small signal bandwidth is given by,

$$f_{-3dB} = \frac{1}{2\pi r_o C_o} \tag{201}$$

where, C_o is the total capacitance at the output node. It must be noted that the bandwidth of the $I - V$ converter is inversely proportional to the gain. Therefore, for a given gain, minimizing the parasitic capacitance at the output node maximizes the bandwidth. For the same reason, the output of the $I - V$ converter must be followed by a voltage buffer.

7.2.5 Voltage-to-Current Converter

$V - I$ converters play a vital role at the input interface of current-mode systems. A common approach to current generation involves the use of an operational amplifier with a MOS transistor $M1$ and a resistor $R1$ as shown in Figure 48(b). Negative feedback ensures that the current through the transistor $M1$ is equal to the applied input voltage divided by the resistor $R1$. For a given size of $M1$ and resistor $R1$, the finite rail-to-rail output voltage swing of the amplifier poses the major limitation

to the achievable linear range of currents. Alternate approaches that have been proposed for $V-I$ converters [74], [75], [76], [77] suffer from limited linearity and/or susceptible to loading conditions affecting performance. The $V-I$ converter described in this section is compact, easy to design and uses a single external resistor to set its transconductance. The design adopted in this work is an improvement over that in [77]. This makes the performance of the $V-I$ converter immune to loading conditions and experimental results are presented as well.

Figure 50 shows the circuit schematic of the CMOS $V-I$ converter. The use of amplifier $A2$ helps fix the output node at a fixed voltage thereby nullifying the effect of the output capacitance leading to a high bandwidth. This also serves to isolate the output of the $V-I$ converter from external loading effects.

The use of a regulated cascode current mirror ensures that the drain of $M1$ is set to a well defined value of V_{ref}. Also, the regulated cascode increases the output impedance of the current mirror and the matching between the drain currents of $M1$ and $M2$. With the drain of $M1$ set to V_{ref}, the output current I_{out} of the $V-I$ is,

$$I_{in} = \frac{(V_{in} - V_{ref})}{R_{in}} = I_{out} \tag{202}$$

where, V_{in} is the applied input voltage and R_{in} is the value of the resistor used.

The small signal input impedance (r_{in}) at the drain of $M1$ is given by,

$$r_{in} = \frac{1}{g_{m1}[1 + g_{m6}r_{o6}(1 + A)]} \tag{203}$$

where A is the open loop gain of the feedback operational amplifier $A1$, and g_{m6} is the transconductance of the cascode transistor $M6$. The use of a regulated cascode, ensures a very low impedance at the drain of $M1$ that further ensures that the voltage at the drain remains at V_{ref} independent of the current flowing through $M1$. This ensures that (202) holds for a large range of currents.

With proper design and a correct choice of resistor R_{in}, the linear range of the $V-I$ converter will usually not be an issue. There are however two key factors that need

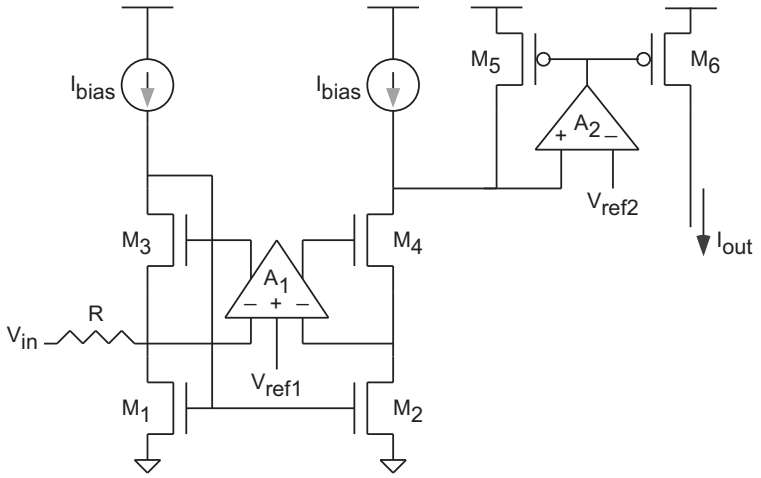


Figure 50. Voltage-Current Converter: Circuit scehmatic of the voltage-current converter that uses a single external resistor to perform the conversion.

to be considered. Assuming the feedback amplifier $A1$ to be ideal, the gate-source voltage of $M6$ can reach a value of $V_{dd} - V_{ref}$ at most and thereby places an upper bound on the output current. Also, the pFET current mirrors come out of saturation and lead to distortion when the gate-source voltage of $M1$ reaches $V_{dd} - 2V_{dsat,p}$. This leads to an an upper bound on the linear range as well. The input voltage swing for the $V - I$ converter is not limited by the power supply and can therefore exceed the positive supply voltage. When the input signal V_{in}, falls below V_{ref}, signal inversion occurs. The output current in this case is limited by the bias current of the PMOS transistors and should therefore be designed accordingly. The speed of the $V - I$ converter is dependent upon the DC bias current, I_{bias} and the parasitic capacitances at the output. The regulated cascode loop must be designed such that the loop bandwidth is greater than the input signal bandwidth and its stability must be ensured as well.

7.3 Adaptive Filter Experimental Results

The proposed analog architecture has been fabricated in a $0.35\mu m$ CMOS process. The experimental setup consists of a custom designed board for the chip that contains

the hardware infrastructure necessary for programming floating-gate transistors. The delay lines have been implemented off-chip using digital-to-analog converters (DACs) and smoothing filters (low-pass filters) to provide flexibility for testing different applications. The setup is controlled using a FPGA board with a computer in the loop resulting in a fully automated test fixture. This provides the flexibility of implementing a variety of learning scenarios as arbitrary waveforms can be generated in software and applied to the chip using DACs. Experimental results that have been measured using the test setup are presented in this section to demonstrate adaptation and learning. Characterization results for the various circuit blocks will be presented first, followed by results from system level experiments performed on the adaptive filter chip.

7.3.1 Interface Circuits And High-Pass Filter Characterization

Figure 51 shows the DC transfer characteristic of the voltage-to-current converter. The converter displays a linear range of about $1.6\mu A$ when biased at a current level of $5\mu A$. The slope of the conversion is equivalent to that of approximately $300K\Omega$ resistor. This is expected as the external resistor used for the conversion was a $300K\Omega$ resistor.

The DC transfer characteristic of the current-to-voltage converter is shown in Figure 52. The converter is designed such that the input current is sinking in nature and therefore the output of the converter displays a negative slope. The measured transimpedance gain of the $I-V$ converter is about $1.6M\Omega$. The converter displays a linear range of approximately $1\mu A$.

The system bandwidth is determined by the available bandwidth of the interface circuits. This has been determined by cascading the $V-I$ converter with the $I-V$ converter and determining the frequency response of the resulting voltage-in/voltage-out system. Figure 53 shows the measured frequency response of such a system. As can be observed from the figure, the available system bandwidth is approximately

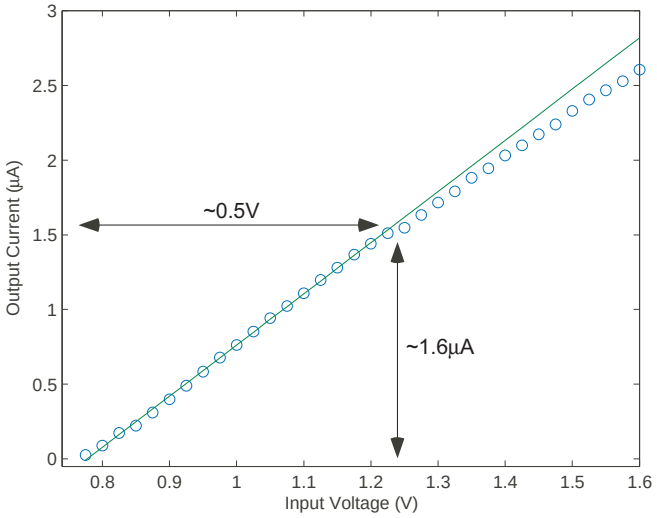


Figure 51. Measured DC Sweep of $V-I$ converter: Measured DC transfer characteristic of the voltage-to-current converter that displays an impedance of $\approx 300K\Omega$.

$110KHz$. The $I-V$ converter is the primary limitation to the available bandwidth and this is on account of biasing the $I-V$ converter to display a high transimpedance gain. However, a bandwidth of $110KHz$ is sufficient for adaptive filter applications pertaining to the audio domain.

Figure 54 shows the step response of the high-pass filter, where the input to the filter is applied using the $V-I$ converter and the current output is read using the $I-V$ converter. The filter displays a high-pass behavior as expected with an offset that is on account of the mismatch between transistor pairs $M1/M6$, $M3/M4$ and current mirror mismatches as has been analytically derived earlier.

7.3.2 Floating-Gate Synapse Step Responses

The equilibrium weight of the synapse is indirectly measured at the source of transistor $M1$ (refer Figure 45) and is at the point where tunneling and injection currents are balanced such that no change occurs at the floating-node. Figure 55 shows the source voltage when a square wave is provided as the input current of the synapse. The instantaneous increase in the bias current of the synapse causes an increase in

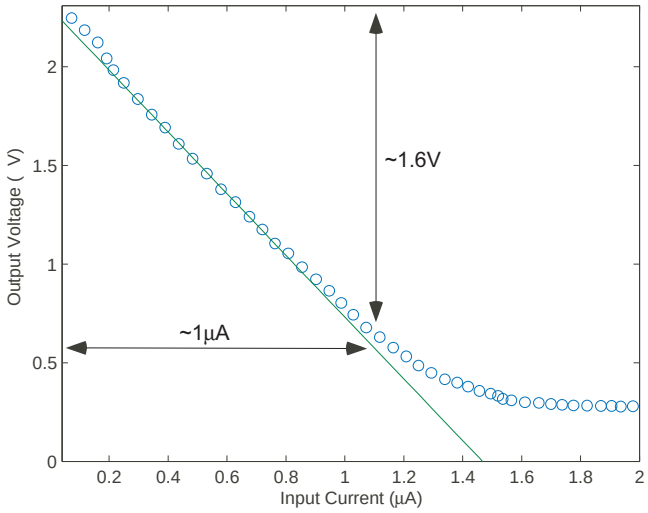


Figure 52. Measured DC Sweep of $I-V$ converter: Measured DC transfer characteristic of the current-to-voltage converter with a transimpedance gain of approximately $1.6M\Omega$.

the source voltage of the synapse. This increase in the source voltage increases the injection current that makes the floating-gate more negative. Since the transistor behaves like a source follower, the decrease in the floating-gate voltage decreases the source voltage and the source voltage reaches a new equilibrium that is again determined by a balance in the injection and tunneling currents.

Figure 56 shows the source voltage for a square wave provided as the error signal. It should be noted that the error signal propogates to the drain of the adapting transistor of the synapse through the post-distort circuitry. When the source-drain voltage across the device decreases on account of the applied step, the injection current decreases. This causes the floating-gate node to increase on account of a higher tunneling current and hence the source voltage increases as well. The rise in the source voltage increases the injection current and the source voltage reaches a new equilibrium value such that tunneling and injection currents are once again balanced.

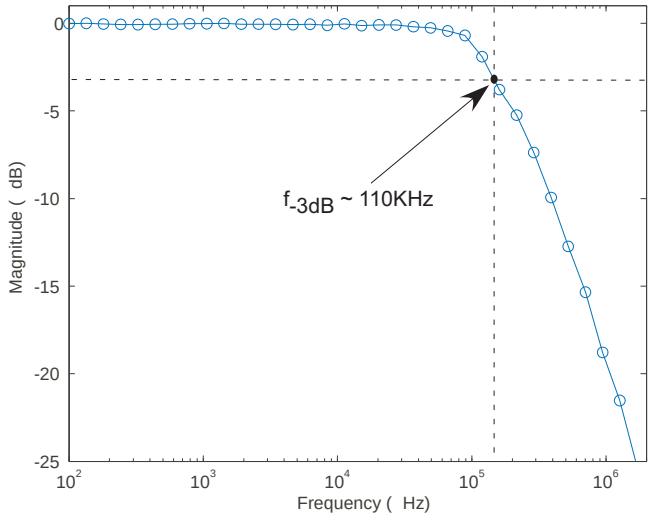


Figure 53. Interface Circuitry Frequency Response: Measured frequency response of the voltage-in voltage-out system formed by cascading a $V-I$ converter with an $I-V$ converter. The system bandwidth is approximately $110KHz$.

7.3.3 Phase Correlation Experiment

In order to demonstrate the correlation behavior of a single synapse as given in (157), a sinusoidal signal is applied to both the input and the error terminal of the synapse circuit. According to (157), the synapse computes the correlation between the two signals, the result being a change in the DC level of the floating-gate voltage. This change in the floating-gate voltage results in a DC change in the source voltage. For an LMS learning rule, with two sinusoidal signals applied to the input and the error voltage, the equilibrium weight is approximately given by,

$$w_{eq} = \approx \frac{A_i A_d}{2} cos\theta \tag{204}$$

where, A_i, A_d are the amplitudes of the applied sinusoidal inputs and θ is the phase difference between the two signals. Measuring the steady-state value of the source voltage for different sinusoidal inputs at the input and the error terminals of the synapse should result in a cosine function. Experimental results are shown in Figure 57 that confirm correlation learning in the synapse. Note that the 180° phase shift

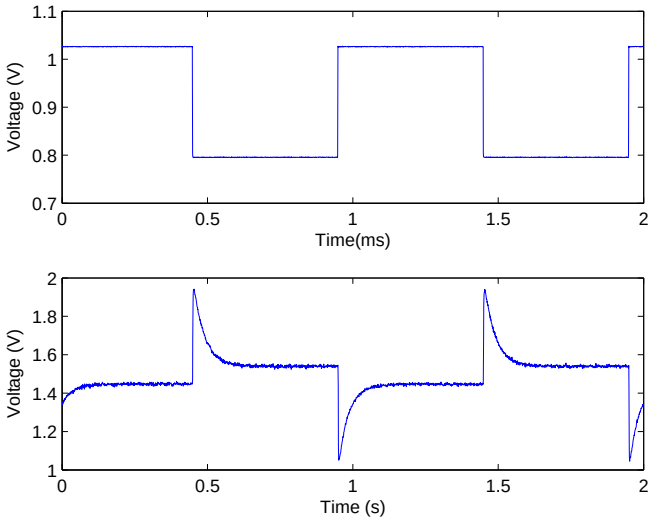


Figure 54. Step Response of High-Pass Filter: Measured step response of the current-mode high-pass filter.

in the cosine is on account of the signal inversion resulting through the post-distort circuitry.

7.3.4 Fourier Decomposition Experiment

A square-wave can be decomposed into a weighted sum of harmonic sinusoids. Therefore, an adaptive linear combiner can learn a square-wave when presented with sinusoids that are at integer multiples $(1, 2, 3...)$ of the square-wave frequency. The fourier decomposition experiment is represented pictorially in Figure 58. The weights adapt to the fourier co-efficients such that the output resembles a square-wave with the result that the error between the output and the target is minimized.

The chip was presented with a $1KHz$ square-wave target and the equilibrium weight was measured by providing the first five harmonics of the target square-wave. The top plot of Figure 59 shows the ideal square-wave that results when the first five harmonics are weighted with the ideal fourier co-efficient and combined together. The bottom plot of Figure 59 shows the resulting square-wave using the weights obtained from the chip. As can be observed, a square-wave results when the first five

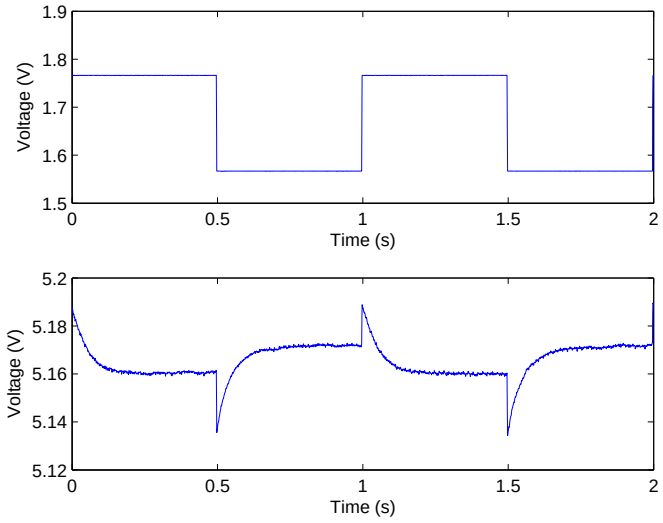


Figure 55. Synapse Weight Dynamics for an Input Current Step: Measured response of the source voltage of the synapse for an input step applied to its bias current. The top plot shows the voltage input to the $V-I$ converter that generates the input current of the synapse. The bottom plot shows the output of the synapse source voltage.

Table 9. Equilibrium weights for a Fourier Decomposition Experiment

Sine Frequency (KHz)	1	2	3	4	5
Meas. Weight	1	0.0445	0.3142	0.0469	0.1881
Fourier Co-efficients	1	0	0.33	0	0.2

harmonics are combined using the measured weights, thereby demonstrating learning in the chip. Table 9 presents the weights obtained experimentally by conducting the above experiment and compares them with the ideal expected value. As expected, the weights converge closely to the ideal values.

7.4 Comparisons To Alternate Techniques

It is clear that implementing adaptive filters requires four key operations: (1) Multiplication with a weight, (2) Addition, (3) Weight adaptation and (4) Weight storage. Several researchers have tackled implementing adaptive filters in the analog domain with the various approaches differing in the way in which the above operations have

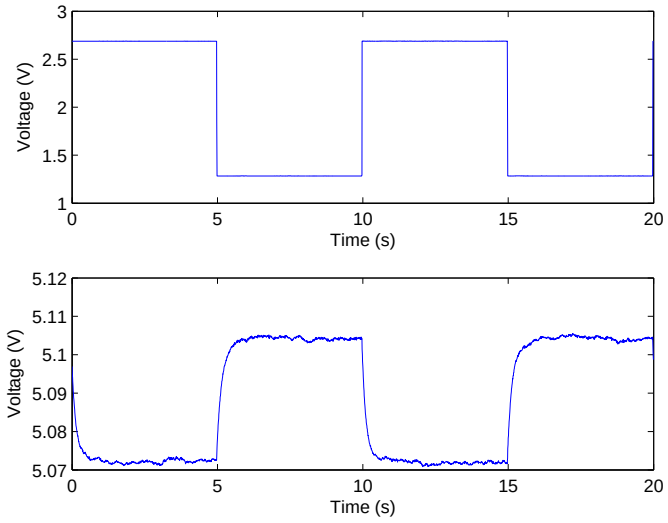


Figure 56. Synapse Weight Dynamics for an Input Current Step: Measured response of the source voltage of the synapse for an input step applied as its error signal. The top plot shows the voltage input to the $V-I$ converter that generates the error current to the post-distort circuitry. The bottom plot shows the output of the synapse source voltage.

been implemented.

The approaches in [66, 62] use floating-gate transistors for weight storage. In [62] a Gilbert-Cell Multiplier [48] was used. The Gilbert-cell multiplier implements a signal-by-signal multiplication with the output being a set of differential currents that were tied together to implement addition. The weight adaptation was performed off-chip followed by an operation that programmed the floating-gate transistors and thereby updated the memory with the correct set of weights. The approach offers long-term weight storage but is not a standalone implementation as demonstrated in this chapter.

A programmable analog synapse based on charge coupled device (CCD) was described in [64]. The weight storage and multiplication are performed using arrays of CCDs. The weight adaptation is performed off-chip followed by a weight transfer on-chip. Aside from the fact that the system requires a computer in the loop, the approach requires a process that supports fabrication of CCDs and hence is not suitable

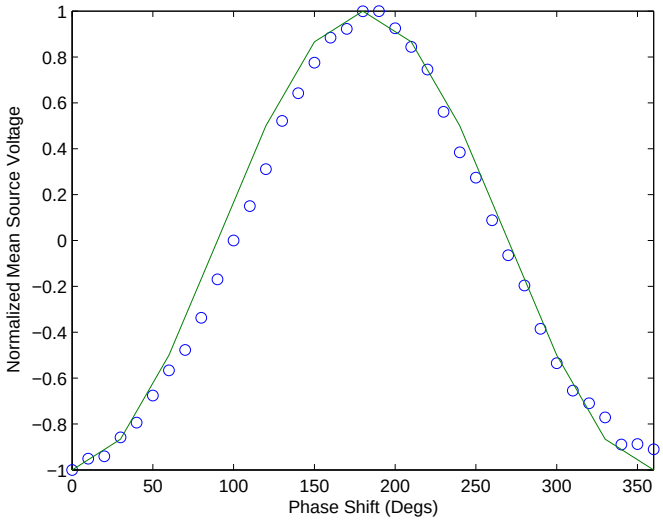


Figure 57. Measured Results For Phase Correlation Experiment: Plot of the normalized equilibrium weight vs. the phase difference between the sinusoidal synapse input and the sinusoidal error signal.

for standard CMOS processes.

The approaches in [68, 67, 63] all use non-floating-gate approaches to storing the weight with the weight adaptation being performed off-chip. The approach in [63] used a voltage-mode multiplying digital-to-analog converter to perform the multiplication of an input signal with the weight. The weight is stored on-chip in a digital form and is updated from off-chip. In [68], the weight is represented as an analog quantity with the weight being stored on capacitors. A transconductance multiplier

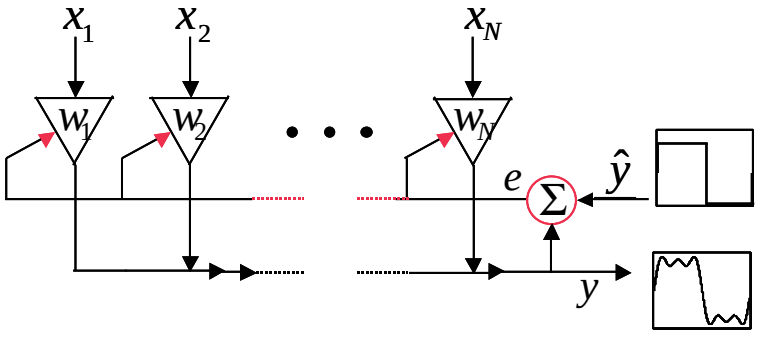


Figure 58. Fourier Decomposition Experiment: Learning a square-wave from harmonic sinusoids

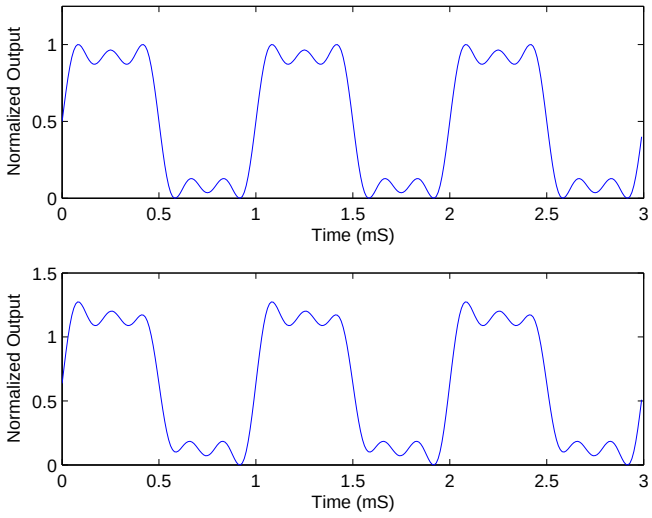


Figure 59. Learning a Square Wave: The top-plot shows an ideal normalized square-wave generated using the fourier co-efficients for the first five harmonics. The bottom plot shows the square-wave generated using equilibrium weights obtained from the analog adaptive filter chip.

is used such that the output of the multiplication is a current such that KCL can be invoked for addition. In [67], a hybrid analog-digital approach is used. The weights are stored in the digital format in a seperate memory. The weights are fetched from memory, converted into the analog form by a digital-to-analog converter and applied to an analog multiplier for multiplication with the input signal.

In [65], a standalone adaptive filter is demonstrated with all the filtering operation performed on-chip. Discrete-time adaptive filtering is demonstrated with operations like addition, subtraction and multiplication being performed using operational amplifiers as building blocks. Weights are stored on on-chip capacitors. The weight adaptation is performed on-chip by directly implementing the various operations required for discrete-time LMS using operational amplifier based functional blocks. On account of this, the implementation is both area and power intensive with a 4 synapse filter occupying an area of $4mm^2$ in a $2\mu m$ p-well process.

Table 10 summarizes the key performance parameters of the fabricated adaptive

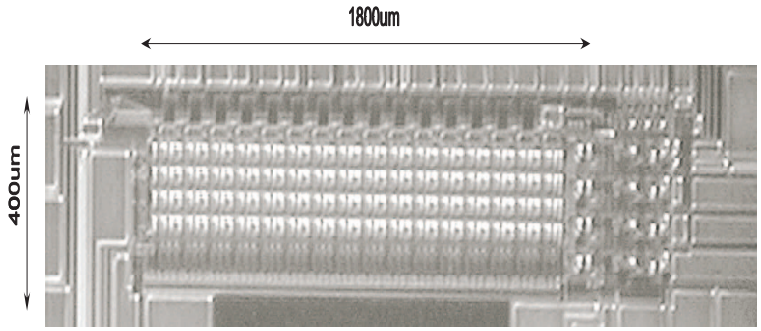


Figure 60. System Die Photograph: The die photograph of the system fabricated in a $0.35\mu m$ CMOS process. The system occupies an area of $1800\mu m \times 400\mu m$

Table 10. Adaptive Filter Summary of Performance

Parameter	Value
Power Supply	$3.3V$
Process	$0.35\mu m$ CMOS
Area	$1800\mu m \times 400\mu m$
Power Dissipation	$13.2mW$
Adaptation Mechanism	Hot-electron injection and Tunneling
Adaptation Time	$1mS - 10S$
Input Signal Bandwidth	$100KHz$

filter chip. The chip occupies an area of $1800\mu m \times 400\mu m$ and contains 4 adaptive nodes with 16 synpses each for a total of 64 synapses and associated circuitry. The entire chip dissipates a power of $13.2mW$ at an operating supply voltage of $3.3V$. The bulk of the power is dissipated in the amplifiers and buffers used in the interface circuitry to drive signals on and off the chip. The use of tunneling and injection as the mechanisms controlling adaptation enables adaptation time-constants in the range of $1mS - 10S$. The wide range of time-constants that are available is a key advantage in the proposed approach.

7.5 Summary

A fully integrated analog implementation of adaptive filters has been demonstrated in a $0.35\mu m$ CMOS process. The synapse is implemented using floating-gate transistors for multiplication, weight storage and weight adaptation. The weight adaptation is performed by exploiting the non-linearities inherent in the physical processes of Fowler-Nordheim tunneling and hot-electron injection. This results in an LMS learning rule for the synapse. Integrating the operations of multiplication, weight storage and weight adaptation in the synapse results in a compact, low-power implementation. Using a current-mode approach allows exploiting KCL for addition with the result that addition dissipates virtually no extra power dissipation.

The system has been designed as a voltage-in/voltage-out system with $V - I$ converters and $I - V$ converters designed to form the interface. Theoretical analysis of the various functional blocks of the system have been presented along with experimental results that characterize their performance. System characterization results indicate correlation behavior and adaptation based on an LMS learning rule. Phase correlation experiments have been performed that demonstrate correlation between the input signal and the error signal as expected. Fourier decomposition experiment using a $1KHz$ square wave target and measuring the equilibrium weight for sinusoids of different harmonic frequencies further demonstrate adaptation in the system.

The architecture is well-suited for implementing large arrays of synapses that are advantageous when envisioning scaling up the system to implement large scale neural networks. With the present design, a 40×75 matrix of synapses can be designed to fit in a chip area of $9mm^2$. With regards to power dissipation, a significant portion of the power dissipation is due to the supporting circuitry. On account of the supporting circuitry being outside the synapse matrix, a reduced power penalty results when scaling to larger systems. For example, doubling the number of synapses by implementing an adaptive filter with 4 nodes of 32 synapses each involves an additional

power dissipation of only around $2.5mW$. The system can be further optimized by performing a low-power design of the peripheral circuitry. In the present implementation, the synapses dissipate only 4.8% of the total power dissipation while 95.2% of the power is dissipated in the peripheral circuitry with 25% of the power being drawn by the output buffer amplifiers alone. In summary, an analog architecture is presented that adapts using the LMS learning rule and is well suited for implementing large-scale adaptive filters and neural networks with minimal area and power penalty.

CHAPTER 8

SIMULATION MODEL FOR FG SYNAPSE

In this chapter, a variation on the source follower floating-gate synapse (SFFG) that was described in Chapter 7 is presented. In this synapse, MOS transistors model the weight adaptation dynamics of injection and tunneling. This proposed synapse has a number of potential applications. First, it can be used as a simulation model for the floating-gate synapse as injection and tunneling cannot be simulated in a circuit level simulator such as SPICE. Second, this structure can be used for studying such phenomena as weight decay and the design tradeoffs that it entails in an adaptive system. Third, the synapse offers a direct control over the adaptation current thereby making faster adaptation rates possible.

8.1 Circuit Description

The proposed synapse structure that models the physical processes of hot-electron injection and tunneling in the SFFG synapse is shown in Figure 61. All signals are represented by fractional variation in currents around a bias value and all transistors operate in the weak inversion region of operation. Transistors M_{abias} and M_{corr} form a high gain amplifier, the gain of which controls the weight decay. This pair will henceforth be referred to as the correlation amplifier. The multiplication of the input signal with the weights is achieved through the use of capacitively coupled current mirrors M_{i+}, M_{s+}, M_{i-} and M_{s-}. The DC bias voltage of V_{fg0} is set by the correlation amplifier. The other floating-gates are programmed to the same DC equilibrium voltage as V_{fg0}.

In the correlation amplifier, the transistor M_{xcorr} mirrors the input into M_{corr} for multiplication, while transistor M_{e1} pre-distorts the error signal at the source of M_{corr}. Transistor M_{e2} guarantees a bias condition on the source of M_{xcorr} that

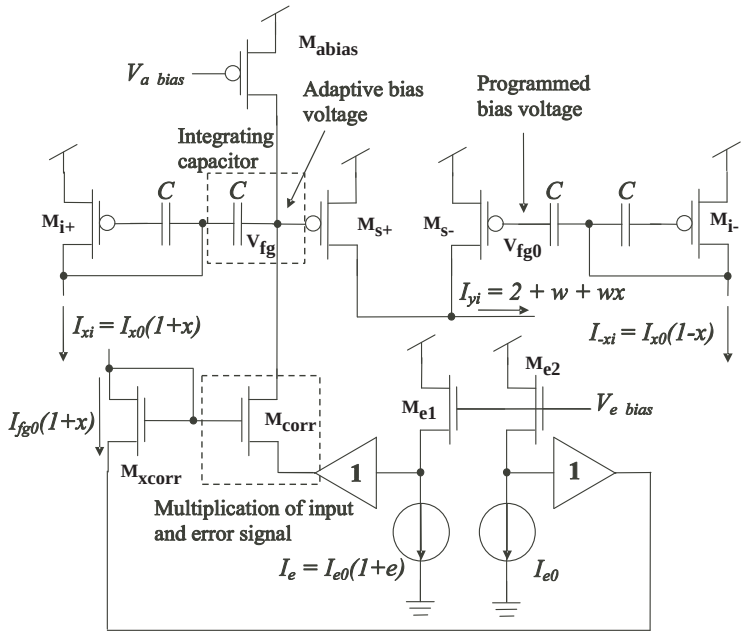


Figure 61. Circuit schematic of the simulation model of the floating-gate synapse: Transistors M_{a_bias} and M_{corr} model the Fowler-Nordheim tunneling and hot-electron injection and result in a weight update based on the LMS rule. The transistor, M_{corr} provides multiplication of the input signal by an error signal. The source of M_{corr} is driven by a buffered error signal voltage which is generated as a logarithmic transform of a linear current signal providing multiplication that is linear in the error signal. All signals are represented as variations in current around a bias point.

matches that on the source of M_{corr} set by M_{e1}. Unity gain buffers guarantee that the source voltages on M_{xcorr} and M_{corr} are independent of their currents. Finally, M_{abias} provides the equilibrium bias current I_{fg0}; current deviations caused by M_{corr} are integrated on the floating-gate yielding correlations, thereby performing weight updates.

The computation is performed by transistors M_{s+} and M_{s-}. The synapse is implemented as a differential structure with the differential inputs being applied to the input transistors M_{i+} and M_{i-}. The change in the drain current of M_{s+} due to the slow varying weight updates and the fast varying input current is given by,

$$I_y = I_{y0} e^{-\kappa(\bar{V}_{fg}+\delta V_{fg})/U_T} \tag{205}$$

where, I_{y0} is the equilibrium bias current,$\bar{V}_{fg}$ is the slow time-varying change in the

floating-gate voltage that denotes weight adaptation and δV_{fg} is the fast time-varying change in the floating-gate voltage due to the input signal. Given that the transistors M_{i+} and M_{s+} match and so do the two input capacitors, the change in the floating-gate voltage of M_{s+} due to the change in the input current signal is given by,

$$\delta V_{fg} = -\frac{U_T}{\kappa} ln(1+x) \tag{206}$$

Next, we define the weight in the adaptive circuit to be,

$$w = e^{-\kappa \bar{V}_{fg}/U_T} - 1 \tag{207}$$

Substituting the above result and the definition of the synapse weight into (4), the output drain current becomes,

$$I_y = I_{y0}(1+x)(1+w) \tag{208}$$

The currents of the two differential pair floating-gate transistors are summed together to give an output current I_y that is expressed as,

$$I_y = I_{y0}(2+w+wx) \tag{209}$$

The required output wx is extracted from this expression by high-pass filtering the output signal to remove the DC component and the slow-varying weight term.

Now to derive the weight adaptation, the floating-gate voltage is first expressed in terms of the weight as,

$$\bar{V}_{fg} = -\frac{U_T}{\kappa} ln(1+w) \tag{210}$$

Taking the first derivative of the above equation, we get,

$$\frac{d\bar{V}_{fg}}{dt} = -\frac{U_T}{\kappa} \frac{1}{(1+w)} \frac{dw}{dt} \tag{211}$$

The effective output impedance at output of the high-gain amplifier is given by,

$$R = \frac{V_{An} \| V_{Ap}}{I_{fg0}} \tag{212}$$

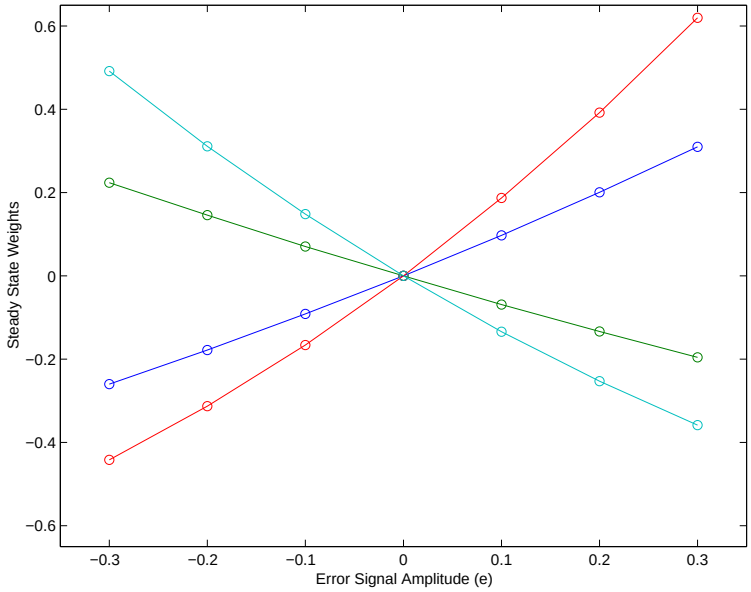


Figure 62. Amplitude Correlation Experiment For Synapse Simulation Model: Plot showing the amplitude correlation of the proposed synapse. For an input signal given by $A_sCos(wt)$ and an error signal given by $A_eCos(wt)$, the steady-state weight is proportional to the product of the error signal and input signal amplitudes. This results in the plot of the steady-state weight vs. error signal amplitude being linear as shown.

where, V_{An} and V_{Ap} are the early voltages of the nFETs and pFETs respectively. Performing nodal analysis at the gate node of transistor M_{s+} and using the linearization techniques as in (4),(5) and (7) we get,

$$C\frac{dV}{dt} = \frac{1}{R}V + I_{fg0}(1 - (1+x)(1+e)) \tag{213}$$

Defining, the time constant of adaption (τ) to be,

$$\tau = \frac{U_T C}{\kappa I_{fg0}} \tag{214}$$

and the weight decay (ϵ) to be,

$$\epsilon = \frac{U_T}{\kappa V_{An} \| V_{Ap}} \tag{215}$$

The synapse learning rule now becomes,

$$\tau\frac{dw}{dt} = -\epsilon w + E[xe] \tag{216}$$

where it has been assumed that w is small and the signal and error terms have been assumed to be zero mean. Thus, the simulation model proposed for the SFFG indeed displays a weight adaptation that is in accordance with the LMS learning rule.

8.2 Simulation Results

The proposed transistor model of the SFFG synapse has been simulated using a $0.5\mu m$ process. The implementation of the LMS learning rule by the synapse is demonstrated by means of two experiments: (a) Amplitude Correlation and (b) Fourier Series Decomposition. The effect of weight decay on the circuit's performance is illustrated as well.

8.2.1 Amplitude Correlation

Assume an input signal given by $A_sCos(wt)$ and an error signal given by $A_eCos(wt)$. Now, if we further assume that the weight decay is zero, the steady-state value of the weight equals,

$$w = E[xe] = \int_{-\infty}^{+\infty} A_sA_eCos^2(wt)dt = \frac{A_e}{2}A_s \tag{217}$$

The above equation implies that a plot of the steady-state value of the weight vs. the input signal amplitude should be linear with the slope being equal to half of the error signal amplitude. Figure 62 shows a plot of the steady-state weight vs. the input signal amplitude for different values of the error signal amplitude. As expected, the plot is linear thereby confirming correlation learning in the synapse.

8.2.2 Fourier Series Decomposition

The fourier series decomposition provides an excellent approach for examining the LMS learning rule in the proposed synapse. The experimental setup is similar to that in Figure 43. For the target signal, a square wave is chosen with the inputs being three harmonically related sinusoids. With this being the setup, each synapse adapts to the appropriate weight so as to reconstruct the desired target square wave

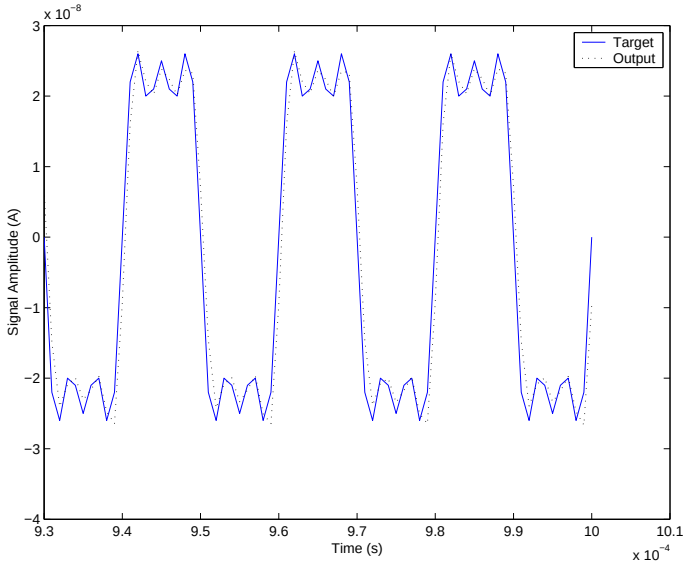


Figure 63. Fourier Decomposition Experiment on Synapse Simulation Model: Plot of the output of the adaptive linear combiner configured to learn a square wave. The input to the system consists of three sinusoids at the fundamental frequency, the third harmonic and the fifth harmonic. The weights adapt to the appropriate value so as to reconstruct the square wave. The solid line shows the target square wave while the dashed line indicates the system output. The design used a channel length of $1.2\mu m$ for the correlation amplifier.

at the output. Figure 63 shows the system output and the target square wave after convergence. The frequency spectrum of the output and the target square wave is shown in Figure 64. As can be observed, both the output and the target have their fundamental, 3^{rd} and the 5^{th} harmonic match closely. However, there are some undesirable even harmonics at the output. The even harmonic terms are related to the extent to which weight decay is minimized. In the particular implementation, the weight decay is related to the length of the transistors M_{a_bias} and M_{corr}. For the case of a channel length of $L = 1.2\mu m$ shown in Figure 64, the second harmonic is $-30dB$ below the fundamental. Decreasing the channel length to $L = 0.9\mu m$ causes the second harmonic to increase to $-7dB$ as shown in Figure 65. This clearly demonstrates the effect of weight decay on the performance of the adaptive system and further emphasizes the need for minimizing the weight decay.

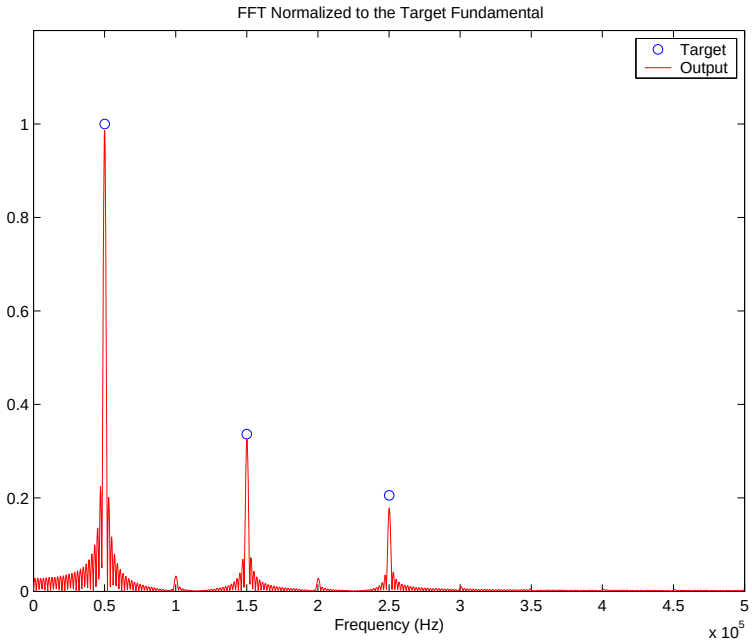


Figure 64. FFT Results For Fourier Decomposition Experiment on Synapse Simulation Model: The FFT of the output of the adaptive linear combiner and that of the target square wave. The output frequency spectrum matches closely with that of the target. There are however even order harmonics that are the result of non-zero weight decay. The design used a channel length of $L = 1.2\mu m$ for the transistors in the correlation amplifier.

8.3 Summary

A circuit model for the SFFG synapse circuit that uses MOS transistors to model the effects of hot-electron injection and Fowler-Nordheim tunneling has been proposed. The proposed model implements the LMS learning rule. Simulation results that demonstrate the performance of the synapse has been presented. The circuit when configured as an adaptive linear combiner with three inputs, adapts to the appropriate weights to learn a target square wave. The issue of weight decay in implementing adaptive systems has been discussed as well.

Two key application spaces exist for the circuit described. The circuit can be used in place of the floating-gate synapse described in chapter 7 in simulating large scale adaptive filters. This is because, adaptation occurs in the floating-gate synapse using the physical phenomenon of hot-electron injection and Fowler-Nordheim tunneling

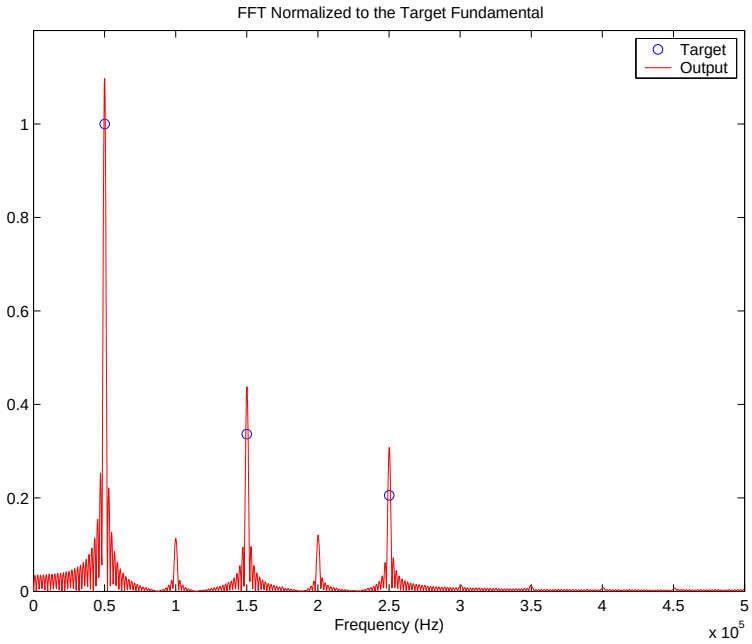


Figure 65. Demonstrating Weight Decay: The FFT of the output of the adaptive linear combiner and that of the target square wave. The design used a channel length of $L = 0.9\mu m$ in the correlation amplifier to illustrate the effect of weight decay. As can be seen, the output no longer matches the target closely and the even order harmonics are a lot higher than the case for $L = 1.2\mu m$.

and at present these cannot be simulated in a circuit simulator such as SPICE. Second, owing to the better control of adaptation currents in the proposed circuit model, this circuit can be used in conjunction with the floating-gate synapse for applications that require faster learning. Initial adaptation can be performed using the circuit model with the weights being transferred onto the floating-gate synapse after adaptation. Such a setup will be useful as it utilizes faster learning capabilities of the circuit model and the non-volatile weight storage offered by the floating-gate synapse.

CHAPTER 9

CONCLUSIONS

Floating-gate transistors provide the option of programmability in analog circuits. Programming the floating-gate transistor (adding electrons using hot-electron injection and removing electrons using Fowler-Nordheim Tunneling) can be viewed as modifying the threshold voltage of the device. This property of a floating-gate transistor can be exploited to correct for mismatch in analog circuits, perform power efficient signal processing and for on-chip adaptation and learning. In this chapter, key milestones that have been achieved in progressing towards this goal are summarized along with ideas for moving forward on this research path.

9.1 Research Summary

Mismatch in analog circuitry is a critical issue that most commonly manifests itself as offset voltages in operational amplifiers. A solution has been proposed in this work that cancels the offset voltage of amplifiers in a compact, low-power fashion. The technique has been experimentally demonstrated by way of a prototype chip fabricated in a $0.5\mu m$ CMOS process and reducing the offset voltage of the amplifier to $25\mu V$. The offset drift with temperature has been measured to be $130\mu V$ over a $170°C$ temperature range. Also, on account of storing the offset cancellation information using floating-gate transistors, the offset voltage drift with time is negligible. Overall, the proposed approach offers comparable offset cancellation with other techniques in a compact and low-power fashion while offering continuous-time amplifier operation [49].

The programmability of floating-gate transistors can be used to design reference circuits as well. In this work a programmable reference circuitry has been designed

that is compact ($0.0022mm^2$) and displays a low temperature co-efficient. The reference uses floating-gate transistors as an inherent part of the circuit and outputs the programmed difference in charge between them as the reference voltage. To a first order, the reference voltage depends only on the charge stored on floating-gate transistors thereby being insensitive to temperature variations. However, second-order effects lead to a temperature dependence. The reference has been fabricated in a $0.35\mu m$ CMOS process and experimental results indicate a temperature sensitivity of $110\mu V/^{\circ}C$ for a $0.6V$ reference voltage. The use of floating-gate transistors allow for a high initial accuracy for the reference. The reference has been programmed to a $\pm 40\mu V$ accuracy. Measured performance of the reference voltage indicate a negligible long-term drift as well. The key advantages of the proposed work include programmability, compactness, high initial accuracy, low temperature dependence and negligible long-term drift [78].

Programmable analog techniques open the door for power efficient signal processing. Multiplication is an operation that is performed repeatedly in signal processing and is both area and power intensive in the digital domain. In this work, a current-mode programmable multiplier has been demonstrated that is both compact and low-power. The multiplier uses floating-gate transistors in such a way that both the multiplication and weight storage operation occur at the same site. This results in a dense implementation that makes large arrays of multipliers (VMMs) possible. The use of current-mode signalling allows for an increased linearity with the multiplier demonstrating over 2 decades of linear range and addition being accomplished by just invoking KCL and tying the outputs together. Weak inversion operation results in a low power dissipation of $531nW/MHz$. Comparing the approach in this work to commercially available DSPs, the analog VMM dissipates $0.27\mu W$ of power for 1 million multiply-accumulate (MMAC) operations while commercial DSPs dissipate typically 3 orders higher power [79].

The charge modification process used in floating-gate transistors can be exploited to create a system that adapts and learns on-chip. A floating-gate synapse has been presented that has the ability to modify the charge stored on its floating-gate based on a LMS learning algorithm [80]. Such a synapse is the fundamental building block for designing adaptive filters and neural networks. In this work, an analog chip architecture has been presented that implements an adaptive filter with on-chip adaptation. Adaptation has been demonstrated in a prototype chip fabricated in a $0.35\mu m$ CMOS process along with characterization results from the basic building blocks such as current-mode high pass filters and interface circuitry [81] for handling current-mode signalling. The chip consists of 4 rows of 16 synapses and occupies an area of $1800\mu m \times 400\mu m$ and dissipates $13.2mW$ of power on a $3.3V$ power supply. The bulk of the power dissipation is on account of circuitry peripheral to the synapse matrix. This chip marks the first fully integrated adaptive filter system using the floating-gate synapse element and is a significant milestone in proceeding forward towards neural network based on-chip learning.

The floating-gate synapse adapts using the physical phenomenon of hot-electron injection and Fowler-Nordheim tunneling. At the time of writing this dissertation, these processes are not modelled in a circuit simulator such as SPICE. Therefore, it is not possible to simulate floating-gate adaptation based on tunneling and injection. Towards this end, a circuit based simulation model has been developed for the floating-gate synapse. This circuit uses transistors to model tunneling and injection and has been demonstrated to show adaptation based on the LMS learning rule, just like the floating-gate synapse. Experiments that demonstrate correlation behavior have been performed succesfully. A Fourier decomposition experiment has been performed as well where an adaptive filter built using the simulation model learns a square wave when presented with sinusoids of different harmonics of the fundamental [82].

9.2 Research Directions-Looking Forward

The work in this dissertation has demonstrated the feasibility of precision analog circuits, power efficient signal processing and on-chip learning using floating-gate programmable analog circuits. A number of research trajectories are possible that build on the foundation laid by this dissertation. Some of the research directions that can be extrapolated from this work are given below:

1. The voltage reference demonstrated in Chapter 5 can also be used as a current reference. The reference current, however, displays a temperature dependence on account of the temperature dependence of the resistor used to bias the circuitry. Theoretically, the transconductance of a transistor can be biased in a zero temperature co-efficient region. Using such a transconductance to implement a resistor and biasing the reference circuitry can result in a current reference that is relatively independent of temperature.

2. The voltage reference, being programmable can be used to implement digital-to-analog converters. On account of the compactness of the reference circuitry, the resulting DAC will be area efficient. Also, the low temperature sensitivity of the reference can be exploited to build DACs that maintain their accuracy (INL and DNL) over a large range of temperature.

3. The offset cancellation technique outlined in Chapter 4 can be extended for use in comparators. Comparators are essential building blocks in analog-to-digital converters, most notably, Flash type converters. These converters are limited in their accuracy on account of mismatch that gives rise to offsets in comparators. Offsets are corrected at the expense of area. Using a floating-gate based offset cancellation can result in both an area and accuracy advantage. Also, floating-gate transistors in general can be used for designing precision analog circuits [83].

4. The work in Chapters 4 & 5 have demonstrated the feasibility of achieving a low-sensitivity to temperature of parameters that are set by floating-gate transistors. Since for normal operating range of temperatures ($-55°C - 140°C$), the charge stored on the floating-gate can be assumed constant, a class of circuits can be built around this framework that display behavior that is independent of temperature. Pursuing this research direction can lead to circuits with parameters, set by floating-gate transistors, optimized for low temperature sensitivity.

5. As demonstrated in this work, programming a floating-gate transistor can be viewed as modifying the threshold voltage of the device. This property of the floating-gate transistor can be exploited by building circuits that operate on a reduced supply voltage. Investigating architecture modifications that make this possible can result in low-voltage operation on processes that require higher supply voltages.

6. The programmable multiplier in Chapter 6 finds application in other areas of signal processing such as FIR filtering and IIR filtering. Preliminary work done by the author in this area indicate power saving of about $7 - 10$ times when compared with digital techniques. The bulk of the power is dissipated in the delay lines that are critical to the filter operation [84]. Research in developing circuit architectures for low-power delay lines can further enhance the power savings.

7. Implementing delay lines along with the adaptive filter implementation in Chapter 7 opens the door for a number of adaptive filter applications. These include channel equalization, echo cancellation, prediction, system identifcation and adaptive beamforming.

8. Architectures should be investigated to integrate the simulation model, described in Chapter 8 in an adaptive filter framework along with the floating-gate

synapse presented in Chapter 7. The architecture should be such that the simulation model is utilized to achieve faster learning, followed by the learnt weights being transferred onto the floating-gate synapse for long-term retention.

9. Having established learning using the LMS algorithm, sets the stage for investigating circuit architectures for implementing large-scale neural networks on-chip. This opens the possibility of performing such complex tasks as pattern recognition and other computationally intensive tasks in a low-power, yet parallel fashion in the analog domain.

It is hoped that pursuing these research directions lead to exciting new frontiers thereby making the area of floating-gate based programmable analog techniques a challenging and rewarding research area.

VITA

Venkatesh Srinivasan was born in Chennai, India in 1978. After completing his high school in 1995, he joined Birla Institute of Technology and Science, Pilani, India where he pursued his Bachelor's in Electrical and Electronics Engineering. Upon completion of his Bachelor's in 1999, he was employed with Wipro Technologies, Bangalore, India as a VLSI/System Design Engineer. At Wipro, he was involved in the design of the Analog Physical Layer of the IEEE1394 Standard. In 2000, he enrolled in the Master's program in the Department of Electrical and Computer Engineering at The University of Tennessee, Knoxville. Graduating with a Master's in 2002, he joined the Integrated Computational Electronics Lab at Georgia Institute of Technology for his Ph.D. He completed his doctoral thesis in 2006 and is presently employed with Texas Instruments, Dallas, TX as an Analog Design Engineer.

REFERENCES

[1] "IEEE standard definitions and characterization of using floating-gate semiconductor arrays," *IEEE Std 1005-1998*, Feb. 1999.

[2] 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, pp. 373–376, May 2004.

[3] P. Kinget, "Device mismatch and tradeoffs in the design of analog circuits," *IEEE Journal of Solid-State Circuits*, pp. 1212–1224, June 2005.

[4] M. J. M. Pelgrom, A. C. J. Duimaijer, and A. P. J. Welbers, "Matching properties of MOS transistors," *IEEE Journal of Solid-State Circuits*, vol. 24, pp. 1433–1440, Oct. 1989.

[5] K. R. Lakshmikumar, R. A. Hadaway, and M. A. Copeland, "Characterization and modeling of mismatch in MOS transistors for precision analog design," *IEEE Journal of Solid-State Circuits*, vol. 21, pp. 1057–1066, Dec. 1986.

[6] C. C. Enz and G. C. Temes, "Circuit techniques for reducing the effects of op-amp imperfections: autozeroing, correlated double sampling and chopper stabilization," *Proceedings of the IEEE*, vol. 84, pp. 1584–1614, Nov. 1996.

[7] J. Ming and S. H. Lewis, "An 8-bit 80Msamples/s pipelined analog-to-digital converter with background calibration," *IEEE Journal of Solid-State Circuits*, pp. 1489–1497, Oct. 2001.

[8] F. Krummenacher and N. Joehl, "A 4-mhz continuous-time filter with on-chip automatic tuning," *IEEE Journal of Solid-State Circuits*, vol. 23, pp. 750–758, June 1998.

[9] Y. Nakamura, T. Miki, A. Maeda, H. Kondoh, and N. Yazawa, "A 10-bit 70MS/s CMOS D/A converter," *IEEE Journal of Solid-State Circuits*, vol. 26, pp. 637–642, Apr. 1991.

[10] A. Hastings, *The Art of Analog Layout*. Prentice-Hall, 1 ed., 2000.

[11] Y. Tsividis, *Operation and Modeling of the MOS Transistor*. McGraw Hill, New York, 1999.

[12] C. C. Enz, F. Krummenacher, and E. Vittoz, "Am analytical MOS transistor model valid in all regions of operation and dedicated to low-voltage and low-current applications," *Journal of Analog Integrated Circuits and Signal Processing*, pp. 83–114, July 1995.

[13] D. Kahng and S. M. Sze, "A floating-gate and its application to memory devices," *The Bell System Technical Journal*, vol. 46, no. 4, pp. 1288–1295, 1967.

[14] S. Lai, "Flash memories: where we were and where we are going," in *IEEE International Electron Devices Meeting*, (San Francisco), pp. 971–974, 1998.

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

[16] M. Lezlinger and E. Snow, "Fowler-Nordheim tunneling in thermally grown SiO_2," *Journal of Applied Physics*, vol. 40, pp. 278–283, Jan. 1969.

[17] P. Hasler, *Foundations of Learning in Analog VLSI*. PhD thesis, California Institute of Technology, February 1997.

[18] P. Hasler, B. A. Minch, J. Dugger, and C. Diorio, "Adaptive circuits and synapses using pfet floating-gate devices," in *Learning in Silicon* (G. Cauwenbergs, ed.), pp. 33–65, Kluwer Academic, 1999.

[19] C. Duffy and P. Hasler, "Modeling hot-electron injection in PFET's," *Journal of Computational Electronics*, vol. 2, pp. 317–322, Dec. 2003.

[20] S. Kinoshita, T. Morie, M. Nagata, and A. Iwata, "A PWM analog memory programming circuit for floating-gate MOSFETs with $75\mu s$ programming time and 11-Bit updating resolution," *IEEE Journal of Solid-State Circuits*, vol. 36, pp. 1286–1290, May 2003.

[21] W. Gao and W. M. Snelgrove, "Floating gate charge-sharing: a novel circuit for analog trimming," *Proceedings of the International Symposium on Circuits and Systems*, pp. 315–318, May 1994.

[22] L. R. Carley, "Trimming analog circuits using floating-gate analog MOS memory," *IEEE Journal of Solid-State Circuits*, vol. 24, pp. 1569–1574, Dec. 1989.

[23] E. Sackinger and W. Guggenbuhl, "An analog trimming circuit based on a floating-gate device," *IEEE Journal of Solid-State Circuits*, vol. 23, pp. 1437–1440, Dec. 1988.

[24] D. Graham, E. Farquhar, B. Degnan, C. Gordon, and P. Hasler, "Indirect programming of floating-gate transistors," *Proceedings of the International Symposium on Circuits and Systems*, vol. 1, pp. 2172–2175, May 2005.

[25] B. K. Ahuja, H. Vu, C. L. Aber, and W. Owen, "A $0.5\mu A$ precision CMOS floating-gate analog reference," *Proceedings of the International Solid State Circuits Conference*, vol. 48, pp. 286–287, Feb. 2005.

[26] C. Bleiker and H. Melchior, "A four-state EEPROM using floating-gate memory cell," *IEEE Journal of Solid-State Circuits*, vol. 22, pp. 460–463, June 1987.

[27] H. Nozama and S. Kokyama, "A thermionic electron emission model for charge retention in SAMOS structures," *Japanese Journal of Applied Physics*, vol. 21, pp. L111–L112, Feb. 1992.

[28] W. J. Kim, S. Sompur, and Y. B. Kim, "A novel digial controlled technique for operational amplifier compensation," *Proceedings of the Midwest Symposium on Circuits and Systems*, pp. 211–214, Aug. 2001.

[29] G. Erdi, "Never-mentioned OP AMP issues," *Proceedings of the Bipolar Circuits and Technology Meeting*, pp. 219–222, Sept. 1990.

[30] F. Adil, G. Serrano, and P. Hasler, "Offset removal using floating gate circuits for mixed-signal systems," in *Southwest Symposium on Mixed-Signal Design*, pp. 190–195, Feb. 2003.

[31] P. Brady and P. Hasler, "Investigations using floating-gate circuits for flash adcs," *Proceedings of the Midwest Symposium on Circuits and Systems*, vol. 2, pp. 83–86, August 2002.

[32] I. M. Filanovsky and A. Allam, "Mutual compensation of mobility and threshold voltage temperature effects with applications in CMOS circuits," *IEEE Transactions on Circuits and Systems I*, vol. 48, pp. 876–884, July 2001.

[33] A. A. Osman and M. A. Osman, "Investigation of high temperature effects on MOSFET transconductace (g_m)," *Proceedings of the IEEE High Temperature Electronics Conference*, pp. 301–304, June 1998.

[34] J. H. Artherton and H. T. Simmonds, "An offset reduction technique for use with CMOS integrated comparators and amplifiers," *IEEE Journal of Solid-State Circuits*, vol. 27, pp. 1168–1175, Aug. 1992.

[35] M. Burns and G. Roberts, eds., *An Introduction to mixed-signal IC test and measurement*. Oxford University Press, 2001.

[36] I. E. Opris and G. A. Kovacs, "A rail-to-rail ping-pong op-amp," *IEEE Journal of Solid-State Circuits*, vol. 31, pp. 1320–1324, Sept. 1996.

[37] A. P. Brokaw, "A Simple Three Terminal IC Bandgap Reference," *IEEE Journal of Solid-State Circuits*, pp. 388–393, Dec. 1974.

[38] B. S. Song and P. R. Gray, "A Precision Curvature-Compensated CMOS Bandgap Reference," *IEEE Journal of Solid-State Circuits*, pp. 634–643, Dec. 1983.

[39] P. K. T. Mok and K. N. Leung, "Design considerations of recent advanced low-voltage low-temperature-coefficient cmos bandgap voltage reference," in *IEEE Custom Integrated Circuits Conference*, pp. 635–642, Apr. 2004.

[40] H. Banba, H. Shiga, A. Umezawa, T. Tanzawa, S. Atsumi, and K. Sakui, "A CMOS Bandgap Reference Circuit with Sub-1-V Operation," *IEEE Journal of Solid-State Circuits*, vol. 34, pp. 670–674, May 1999.

[41] K. N. Leung and P. K. T. Mok, "A Sub-1-V 15-ppm/c CMOS Bandgap Voltage Reference without Requiring Low Threshold Voltage Devices," *IEEE Journal of Solid-State Circuits*, vol. 37, pp. 526–530, Apr. 2002.

[42] Y. Jiang and E. K. F. Lee, "Design of Low-Voltage Bandgap Reference Using Transimpedance Amplifier," *IEEE Transactions on Circuits and Systems II*, vol. 47, pp. 552–555, June 2000.

[43] R. A. Blauschild, P. A. Tucci, R. S. Muller, and R. G. Meyer, "A new NMOS temperature-stable voltage reference," *IEEE Journal of Solid-State Circuits*, vol. 13, pp. 767–774, Dec. 1978.

[44] H. J. Oguey and B. Gerber, "Mos voltage reference based on polysilicon work function," *IEEE Journal of Solid-State Circuits*, vol. 15, pp. 264–269, June 1980.

[45] R. Harrison, J. Bragg, P. Hasler, B. Minch, , and S. Deweerth, "A cmos programmable analog memory-cell array using floating-gate circuit," *IEEE Transactions on Circuits and Systems I*, vol. 48, p. 4, Jan 2001.

[46] R. J. Baker, H. W. Li, and D. E. Boyce, *CMOS Circuit Design, Layout and Simulation*. Prentice-Hall of India, New Delhi, 2000.

[47] K. N. Leung and P. T. Mok, "A sub-1-v $15 - ppm/^{\circ}C$ CMOS bandgap voltage reference without requiring low threshold voltage device," in *IEEE Journal of Solid-State Circuits*, vol. 37, pp. 526–530, Apr. 2002.

[48] P. Gray, P. J. Hurst, S. H. Lewis, and R. G. Meyer, *Analysis and Design of Analog Integrated Circuits*. Wiley, 4 ed., 2001.

[49] V. Srinivasan, G. Serrano, J. Gray, and P. Hasler, "A precision CMOS amplifier using floating-gates for offset cancellation," *Proceedings of the IEEE Custom Integrated Circuits Conference*, pp. 739–742, 2005.

[50] A. Buck, C. McDonald, S. Lewis, and T. R. Viswanathan, "A CMOS bandgap reference without resistors," in *IEEE International Solid-State Circuits Conference*, Feb. 2000.

[51] K. N. Leung and P. K. T. Mok, "A CMOS voltage reference based on weighted ΔV_{GS} for CMOS low-dropout linear regulators," in *IEEE Journal of Solid-State Circuits*, vol. 38, pp. 146–150, Jan. 2003.

[52] T. S. Hall, C. M. Twigg, J. D. Gray, P. Hasler, and D. V. Anderson, "Large-scale field-programmable analog arrays for analog signal processing," *IEEE Transactions on Circuits and Systems I*, vol. 52, pp. 2298–2307, Nov. 2005.

[53] H. Fujishima, Y. Takemoto, T. Onoye, and I. Shirakawa, "An architecture of a matrix-vector multiplier dedicated to video decoding and three-dimensional graphics," *IEEE Transactions on Circuits and Systems II*, vol. 9, pp. 306–314, Mar. 1999.

[54] C. Yee and A. Buchwald, "A sampled-data switched-current analog 16-tap fir filter with digitally programmable coefficients in 0.8μm cmos," *Proceedings of the International Solid State Circuits Conference*, vol. 33, pp. 54–54, Feb 1997.

[55] A. Aslam-Siddiqi, W. Brockherde, and B. Hosticka, "A 16 x 16 nonvolatile programmable analog vector-matrix multiplier," *IEEE Journal of Solid-State Circuits*, vol. 33, pp. 1502–1509, Oct. 1998.

[56] R. Genov and G. Cauwenberghs, "Charge-mode parallel architecture for vector-matrix multiplication," *IEEE Transactions on Circuits and Systems II*, vol. 48, pp. 930–936, Oct. 2001.

[57] F. Kub, K. Moon, I. Mack, and F. Long, "Programmable analog vector-matrix multipliers," *IEEE Journal of Solid-State Circuits*, vol. 25, pp. 207–214, Feb. 1990.

[58] N. Khachab and M. Ismail, "Mos multiplier/divider cell for analogue vlsi," *Electronics Letters*, vol. 23, pp. 1550–1552, Nov. 1989.

[59] H. Song and C. Kim, "An nmos four-quadrant analog multiplier using simple two-input squaring circuits with source followers," *IEEE Journal of Solid-State Circuits*, vol. 25, pp. 841–848, June 1990.

[60] H. Mehrvarz and C. Kwok, "A novel multi-input floating-gate mos four quadrant analog multiplier," *IEEE Journal of Solid-State Circuits*, vol. 31, pp. 1123–1131, Aug. 1996.

[61] S. Haykin, *Neural Networks: A Comprehensive Foundation*. Prentice-Hall, 1999.

[62] M. Holler, S. Tam, H. Castro, and R. Benson, "n electrically trainable artificial neural network with 10240 'floating gate' synapses," in *Proceedings of the International Joint Conference on Neural Networks*, vol. II, (Washington, D. C.), pp. 191–196, 1989.

[63] M. Al-Nsour and H. Abdel-Aty-Zohdy, "ANN digitally programmable analog synapse," *Proceedings of the Midwest Symposium on Circuits and Systems*, pp. 489–492, Aug. 1999.

[64] A. J. Agranat, C. F. Neugebauer, R. D. Nelson, and A. Yariv, "The CCD neural processor: A neural network integrated circuit with 65536 programmable analog synapses," *IEEE Transactions on Circuits and Systems*, vol. 37, pp. 1073–1075, Aug. 1990.

[65] G. Gomez and R. Siferd, "Single-chip FIR adaptive filter using CMOS analog circuits," *Proceedings of the IEEE International ASIC Conference and Exhibit*, pp. P3–5.1–P3.5.4, Sept. 1991.

[66] P. Hafliger and C. Rasche, "Floating-gate analog memory for parameter and variable storage in a learning silicon neuron," *Proceedings of the International Symposium on Circuits and Systems*, pp. 416–419, May 1999.

[67] I. A. Mack, F. Kub, K. K. Moon, and F. M. Long, "Programmable Analog Vector-Matrix Multiplier," *IEEE Journal of Solid-State Circuits*, vol. 25, pp. 207–214, Feb. 1990.

[68] Y. Tsividis and S. Satyanarayana, "Analogue circuits for variable synapse electronic neural networks," *Electronics Letters*, vol. 24, no. 2, pp. 1313–1314, 1987.

[69] P. Hasler and J. Dugger, "Correlation learning rule in floating-gate pFET synapses," *IEEE Transactions on Circuits and Systems II*, vol. 48, pp. 65–73, Jan. 2001.

[70] J. Dugger, *Adaptive analog VLSI signal processing and neural networks*. PhD thesis, Georgia Institute of Technology, November 2003.

[71] P. Hasler and J. Dugger, "An analog floating-gate node for supervised learning," *IEEE Transactions on Circuits and Systems I*, vol. 52, pp. 834–845, May 2005.

[72] R.N.Caffin, "On some aspects of the high-frequency performance of operational amplifiers and current-to-voltage converters," *IEEE Journal of Solid-State Circuits*, vol. 10, pp. 503–505, Dec. 1975.

[73] C.Wang and J.Wang, "Design of linear transimpedance amplifiers," *Proceedings of the 4th International Conference on ASIC*, pp. 232–235, Oct. 2001.

[74] E. B.Nauta and W.Kruiskamp, "A CMOS triode transconductor," *Proceedings of the International Symposium on Circuits and Systems*, vol. 4, pp. 2232–2235, June 1991.

[75] A. Nedungadi and T.R.Viswanathan, "Design of linear transconductance elements," *IEEE Transactions on Circuits and Systems I*, vol. 31, pp. 891–894, Oct. 1984.

[76] K.C.Kuo and A.Leuciuc, "A novel linear tunable MOS transconductance," *IEEE Journal of Solid-State Circuits*, vol. 36, pp. 147–151, Jan. 2001.

[77] A.-M. R.Shukla, J.Ramirez-Angulo and R.G.Carvajal, "A Low Voltge Rail to Rail V-I Conversion Scheme for Applications in Current Mode A/D converters," *Proceedings of the International Symposium on Circuits and Systems*, vol. 1, pp. 916–919, May 2004.

[78] V. Srinivasan, G. Serrano, C. M. Twigg, and P. Hasler, "A compact programmable CMOS reference with $\pm 40\mu$V accuracy," *Accepted to Proceedings of the IEEE Custom Integrated Circuits Conference*, Sept. 2006.

[79] R. Chawla, A. Bandyopadhyay, V. Srinivasan, and P. Hasler, "A 531nw/mhz, 128x32 current-mode vector matrix multiplier with over 2 decades of linear range," *Proceedings of the IEEE Custom Integrated Circuits Conference*, pp. 29–4–1 – 29–4–4, Oct. 2004.

[80] J. Dugger, V. Srinivasan, and P. Hasler, "A supervised neural network of continuously adapting analog floating-gate nodes," *Proceedings of the 37th Asilomar Conference of Signals, Systems and Computers*, pp. 2031–2035, Nov. 2003.

[81] V. Srinivasan, R. Chawla, and P. Hasler, "Linear current-voltage and voltage-current converters," *Proceedings of the Midwest Symposium on Circuits and Systems*, pp. 675–678, Aug. 2005.

[82] V. Srinivasan, J. Dugger, and P. Hasler, "A adaptive analog synapse circuit that implements the least-mean-square learning rule," *Proceedings of the International Symposium on Circuits and Systems*, pp. 4441–4444, May 2005.

[83] V. Srinivasan, D. Graham, and P. Hasler, "Floating-gates for precision analog circuits: An overview," *Proceedings of the Midwest Symposium on Circuits and Systems*, pp. 71–74, Aug. 2005.

[84] V. Srinivasan, G. Rosen, and P. Hasler, "Low-power implementation of FIR filters using current-mode analog design techniques," *Proceedings of the 38th Asilomar Conference of Signals, Systems and Computers*, pp. 2223–2227, Nov. 2004.