

Low Control Voltage Programming of Floating Gate MOSFETs and Applications

Axel Thomsen *Member, IEEE*, Martin A. Brooke *Member, IEEE*

Abstract—Programming of EEPROM devices fabricated in standard processes often requires voltages that exceed the bulk-drain breakdown voltages of integrated circuits, making the full integration of the programming circuitry difficult. This paper presents a modified EEPROM device with two tunneling injectors that allows bidirectional, accurately controlled programming with voltages in the range of normal supply voltages. Additionally, two static voltages are required. This scheme allows full integration of the programming circuitry on chip. Applications in the areas of offset reduction for OPAMPs in clocked and continuous time systems as well as in neural network learning are presented.

I. INTRODUCTION

FLOATING gate devices have been used as digital and analog memory [1], [2], as adjustable biasing device [3], [4], and for weight storage in neural networks [5]. The terminals of a floating gate device are the drain, source, bulk, and a control gate that is capacitively coupled to the floating gate. Depending on the programming mechanism, another terminal, the charge injector, may be needed. The design criteria for analog applications are significantly different from the criteria important for digital memory applications. For analog applications charge retention and accurate control of the floating gate charge are of primary concern. This can be accomplished best if well controlled bi-directional programming is possible. It is also desirable to use no special processing to achieve EEPROM operation to reduce the cost of the IC. Standard process EEPROMs require high voltages to initiate tunneling due to thicker insulating oxides. It is also preferable, and for continuous time operation required, that the terminals for programming, such as the control gate and the tunneling injector, are separated from the terminals needed for operation, such as drain and source.

For many possible applications of EEPROMs requirements exist that have not been met by existing standard process EEPROM designs. Applications employing several EEPROMs on one chip require a switching network for selectivity to direct the programming voltage to a certain device. This limits the size of the voltage that controls the programming to the power supply range. Analog applications of EEPROMs such as offset cancellation of amplifiers or weight storage in neural network implementations require bidirectional programming,

Manuscript received December 8, 1992. This work was supported by Analog Devices Inc. and NSF Grant MIP-90 11360. This paper was recommended by Associate Editor John Choma Jr.

A. Thomsen is with the School of Electrical and Computer Engineering at The University of Alabama in Huntsville, Huntsville, AL 35899 USA.

Martin A. Brooke is with the School of Electrical and Computer Engineering at the Georgia Institute of Technology, Atlanta, GA USA.
IEEE Log Number 9400336.

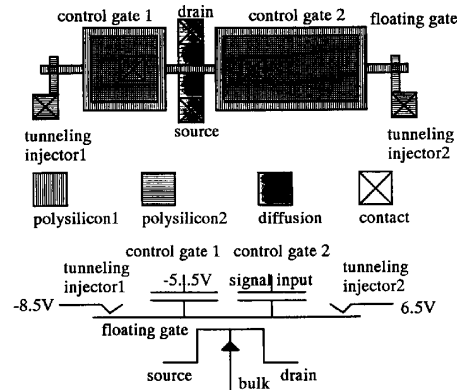


Fig. 1. Layout of the dual injector floating gate device.

either to achieve accelerated programming of the device to high accuracy, or because learning rules employed can result in weight changes of either direction. Applications such as on-chip learning of neural networks require that the signal which controls the weight change is generated and distributed on chip, again limiting this voltage to the power supply range.

For the fabrication process used in this work a tunneling injector design has been reported [6]. Voltages in the order of 12 to 20V are required to induce tunneling. These voltages exceed the breakdown voltage of the drain-bulk junction and thus make the handling of the programming voltage on chip a difficult issue. In other work in this area [3] the power supply voltage swing has been utilized to control the occurrence of tunneling in the charge injector. The main drawbacks in this work are that incremental charge transfer is only possible in one direction and that the control voltage could not easily be managed on chip since it assumes large positive and large negative values beyond the power supply rails. The system requires clocking since the floating gate FET is turned off or turned into the ohmic region during programming whereas it is in saturation during operation.

In a standard CMOS process it is very difficult to implement bidirectional programming on a chip with the usual device structure. If there are only two terminals for programming, the range of voltage differences between those two terminals has to be at least twice the minimum voltage required for tunneling.

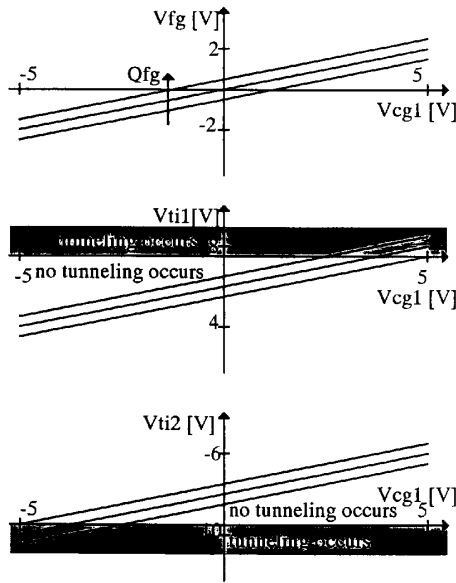


Fig. 2. Principle of operation: Proper biasing puts both injectors on the verge of tunneling, so that the small changes in control gate voltage are sufficient to control tunneling.

This is a voltage swing of more than 20V, which is very difficult to accommodate when the means of insulation are reverse biased p-n junctions with a limited breakdown voltage.

A new floating gate device structure that employs two tunneling injectors is presented. This feature allows low control voltage programming with the requirement of two static high voltages that can be generated and controlled on chip. It will be demonstrated that this device provides a solution for the applications listed above.

II. LAYOUT AND MODELING

The layout of the device is a modification to the layout published in [6]. It is shown in Fig. 1. It was fabricated in the $2\mu\text{m}$ p-well double polysilicon process at the ORBIT foundry through MOSIS. Instead of one tunneling injector two are used. The device under test also used two control gates for measurement purposes. This allowed a measurement of the floating gate charge or the threshold voltage as seen from the control gate without disconnecting the device. The capacitor sizes in this work were $8 \times 8 \mu\text{m}^2$ and $16 \times 8 \mu\text{m}^2$ for C_{cg1} and C_{cg2} respectively. The capacitance per area for a control gate capacitor is $0.6\text{fF}/\mu\text{m}^2$. The device was $W/L=4\mu\text{m}/2\mu\text{m}$ and the tunneling injectors $2\mu\text{m} \times 2\mu\text{m}$ in size. The characteristics of the tunneling injector are described in [6]. It requires voltages in the order of 12 to 22V between control gate and tunneling injector in either direction to transfer charge on or off the floating gate. Note that the voltages reported in [5]

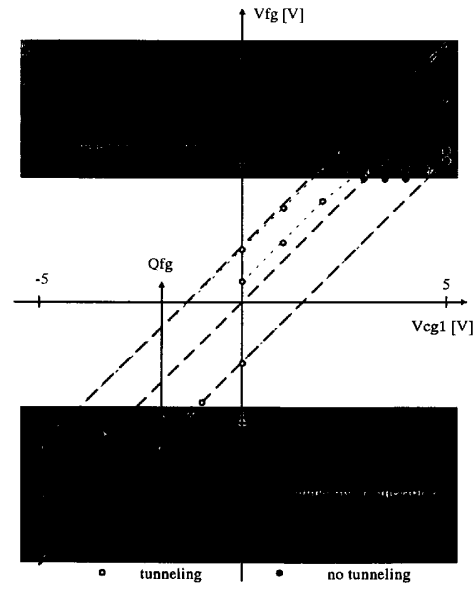


Fig. 3. Experimental results from a device test: a good match to the model.

are voltages between control gate and tunneling injector and thus lower due to the imperfect coupling from control gate to floating gate.

The potential of the floating gate V_{fg} can be described by a voltage summing equation

$$V_{fg} = \sum K_i V_i + Q_{fg}/C_{tot} \quad (1)$$

where the V_i are the voltages at the device terminals and the K_i their respective coupling coefficients given by

$$K_i = C_i/C_{tot}$$

where

$$C_{tot} = \sum C_i$$

Typically the value of K_{cg} is larger than all other K_i . When all voltages except the control gate voltage are constant this equation can be simplified to

$$V_{fg} = V_o + K_{cg} V_{cg} + Q_{fg}/C_{tot} \quad (2)$$

where the V_o represents the sum of all constant coupling voltages.

The drain current I_D of the device can be modeled by the equation

$$I_D = \beta(V_{fg} - V_s - V_T)^2$$

where β is the transconductance parameter, V_s is the source potential, and V_T the threshold voltage. This equation is valid for the saturation mode only.

BEST COPY AVAILABLE

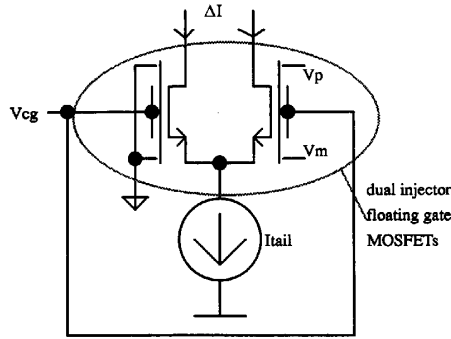


Fig. 4. An adjustable differential current source based on the new device. The output current is a function of the floating gate charge independent of the control gate voltage.

Tunneling is controlled by the voltage across the tunneling injector

$$I_{tunnel} = f(V_{ti} - V_{fg}).$$

Previous work in the area of tunneling through silicon oxide [7] has shown an exponential relationship between programming voltage and tunneling current. For long pulses of programming voltage a simplified description of the behavior of the tunneling injector can be given:

For programming voltages below a 'tunneling threshold' no noticeable charge transfer occurs due to the fact that in this region the exponential is practically zero. The effects are considered on the same level as leakage. This threshold will be denoted V_{hi} and V_{lo} for charging or discharging respectively.

For high programming voltages tunneling occurs until the voltage across the floating gate is at the 'tunneling threshold' V_{hi} or V_{lo} , if the programming pulses are long enough, that is $I_{tunnel} \times t_{pulse} \gg Q_{fg}$. For this case tunneling will happen until

$$V_{fg} - V_{ti} = V_{hi} \quad (3a)$$

or

$$V_{fg} - V_{ti} = V_{lo} \quad (3b)$$

For intermediate voltages incremental charge transfers occur according to the exponential relationship. In continuous time programming the pulse length is considered infinite and thus no intermediate voltages exist.

The accuracy limitation of this simple model is that in reality there is a small tunneling current flowing when the programming voltage is slightly below the tunneling threshold. For practical applications this current has to be considered similarly to charge leakage from the floating gate. It is of no consideration during programming but important for long-term accuracy expectations.

The underlying idea for operation is that although high voltages are required for tunneling it is actually a low voltage range required to control tunneling. A wide range of voltages

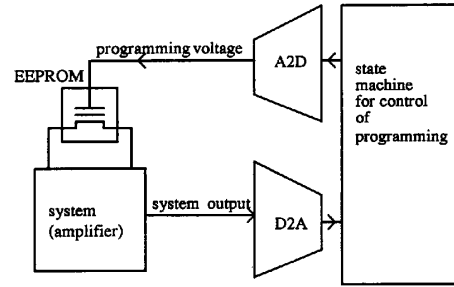


Fig. 5. Schematic of a possible full integration of an EEPROM programming algorithm using standard CMOS building blocks.

exists in which no tunneling occurs. This dead band can be described by

$$I_{tunnel} = 0 \quad \text{for } V_{hi} > V_{fg} - V_{ti} > V_{lo}.$$

In [3] unidirectional tunneling was controlled by a variable signal at the tunneling injector of size V_{dd} or V_{ss} with a fixed signal V_{cg} on the control gate. This voltage was chosen such that

$$V_{lo} > V_{dd} - V_{fg} \quad \text{and} \quad V_{lo} < V_{ss} - V_{fg}$$

where V_{fg} is given by (1). The idea is that a high static voltage on the control gate is used to put the injector on the verge of tunneling near V_{lo} and the additional voltage swing of the power supply is used to decrease the voltage across the tunneling injector $V_{fg} - V_{ti}$ beyond this threshold. This concept can be easily extended to accommodate bidirectional tunneling and accurate control of programming voltages.

By using two tunneling injectors for each floating gate one can be biased to operate around V_{hi} , whereas the other one can be biased to operate near V_{lo} . Thus a change in floating gate voltage in either direction will cause tunneling and transfer charge either on or off the floating gate. Fig. 2 shows the dependence of the floating gate voltage on the control gate voltage and the floating gate charge. It also shows the voltages across the tunneling injectors and how they relate to the tunneling thresholds.

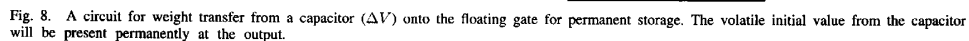
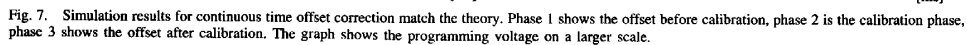
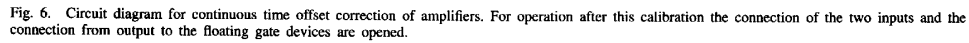
The device behavior can be described with the model given above. Tunneling does not occur in the range of $V_{hi} > V_{fg} - V_{ti} > V_{lo}$. The following conditions must hold for proper operation of the device: No tunneling occurs at the operating point $V_{cg} = 0$ for all possible values of Q_{fg} . In case the control gate is used also as a signal input, no tunneling must occur for all V_{cg} within the dynamic range of the signal. To account for mismatch in an array of devices, maximal and minimal values for the tunneling thresholds will be used.

$$V_{ti1} - V_{fg} < V_{hi}^{min} \quad (4a)$$

and

$$V_{ti2} - V_{fg} > V_{lo}^{max} \quad (4b)$$

where V_{fg} is given by (2) with $V_{cg} = 0$ and $Q_{fg}^{max} > Q_{fg} > Q_{fg}^{min}$.



be achievable for every device on chip.

$$Q_{fg}^{min} = C_{tot} \times (V_{ti2} - V_{lo}^{min} - V_o - K_{cg} \times V_{dd}) \quad (5b)$$

From these equations the size of the static voltages V_{ti1} and V_{ti2} that are applied to the two tunneling injectors can be derived. The results have to be tested for condition (4). By

A similar condition holds for the minimum. Typically a certain range of floating gate charge should be specified that has to

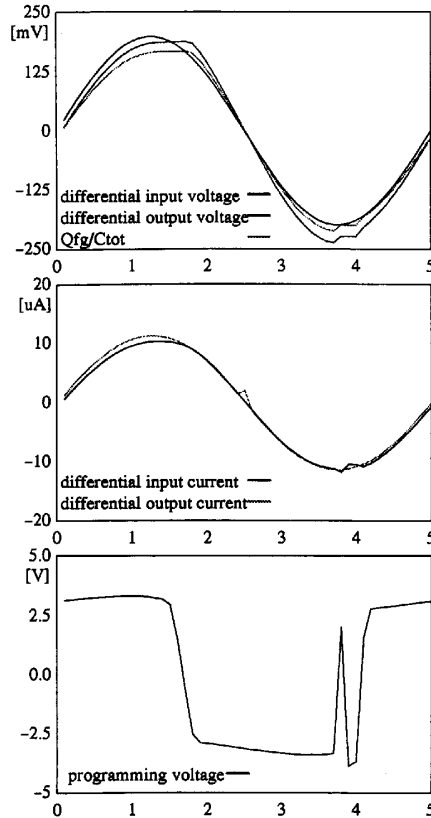


Fig. 9. Simulation results for weight storage on the floating gate. The output of the weight circuit tracks the target value. Errors are due to second-order effects and the presence of the deadband.

evaluating (4) and (5), the maximum achievable range of floating gate charges can be calculated as

$$Q_{fg}^{max} - Q_{fg}^{min} < K_{cg} \times V_{dd} - (V_{hi}^{max} - V_{hi}^{min}) - (V_{lo}^{min} - V_{lo}^{max}) \quad (5c)$$

which simplifies for a single device to

$$Q_{fg}^{max} - Q_{fg}^{min} < K_{cg} \times V_{dd}. \quad (5d)$$

III. EXPERIMENTAL RESULTS

Fig. 3 shows that the charge transfer is working according to the theory. The voltages applied to the tunneling injectors are -8.5V and 6.7V respectively. The coupling coefficient in this layout is $K_{cg1} = 0.5$. Measurement of the threshold was done with the second control gate terminal by recording V_{cg2} for a drain current of $1\mu A$. It should be noted here that in this experiment the programming pulses were usually long so

that the data points are on the border of the tunneling region near V_{hi} or V_{lo} , respectively. Shorter pulses will lead to data points within the tunneling region. In these cases the pulse length was shorter than the time required to perform complete tunneling, i.e. tunneling until $V_{fg} - V_{ti}$ is in the dead band between V_{hi} and V_{lo} .

A similar experiment was done in the setup shown in Fig. 4 which will be used in the application section shown below. It consists of two similar floating gate devices, one of which has high voltages applied to its tunneling injectors for tunneling whereas the other has its injectors grounded and thus keeps a constant floating gate charge. The output parameter here is the differential output current which is given in first-order approximation by

$$\Delta I = (\beta \times I_{tail}) 0.5 \times Q_{fg} / C_{tot}.$$

The device programming worked as expected, it allowed adjustment of ΔI in the range of $-I_{tail}$ to $+I_{tail}$ using voltages between -5 and 5 Volt. An important feature of this circuit is that the output current does not depend on the value of the control voltage but only the floating gate charge. This concept was experimentally verified. The suitability of this scheme for selectivity in a small array has also been tested. After careful choice of V_{ti1} and V_{ti2} the devices could be programmed independently. They have to be chosen so that no tunneling occurs in any device for $V_{cg} = 0$ yet programming is possible for all devices for $V_{ss} < V_{cg} < V_{dd}$. Strong nonuniformity of tunneling thresholds leads to a not fully programmable array of devices.

The tunneling injector voltages should be turned off during regular operation due to the following reasons: Leakage across a biased tunneling injector will be significantly larger than across one without biased close to the tunneling threshold since the energy barrier is reduced. Also it is expected that these voltages are noisy due to their dynamic on-chip generation and feed their noise into the circuit through the parasitic capacitance associated with the tunneling injector design. Ideally the tunneling injector voltages should be reduced by the same magnitude between programming and operation so that the parasitic coupling from the tunneling injector to the floating gate cancels.

IV. APPLICATIONS

The control of the floating gate charge with a low voltage offers the opportunity to integrate more or all of the circuitry required for accurate programming of floating gate devices on chip. Below are a few examples where this device and programming method lead to improvements over existing solutions.

Offset reduction of operational amplifiers is one important possible application area for floating gate MOSFETs. Examples of possible circuits can be found in [3], [4]. Smart programming algorithms that monitor the convergence of the offset value and adjust the programming voltage accordingly [4] converge faster than previously published work [3]. Using this device a more complex programming algorithm can be implemented in a state machine on chip. This can include

```

* the actual MOS device
meeprom d fg s b nfet
*coupling of the control gate voltage into the floating gate
ecg cgx 0 cg 0 0.8
*representing the floating gate charge as the difference of two capacitor charges
*see below
eqfg cg fg 100 101 1

*modeling of one tunneling injector current using the exponential characteristics
*of the diode, vos1 models the difference between vp and the tunneling threshold Vhi.
*r11 limits the current for convergence.
e1 110 0 fg 0 1
vos1 112 110 -1
r11 112 111 1
d1 111 0 diode

*integrating the charging current on capacitor C1 representing the floating gate
f1 100 0 vos1 1
c1 100 0 10u
r1 100 0 1T

*modeling of one tunneling injector current using the exponential characteristics
*of the diode, vos2 models the difference between vm and the tunneling threshold Vlo.
*r22 limits the current for convergence.
e2 120 0 0 fg 1
vos2 122 120 -1
r22 122 121 1
d2 121 0 diode

*integrating the discharging current on capacitor C2 representing the floating gate
f2 101 0 vos2 1
c2 101 0 10u
r2 101 0 1T
.model diode d
.model nfet nmos

```

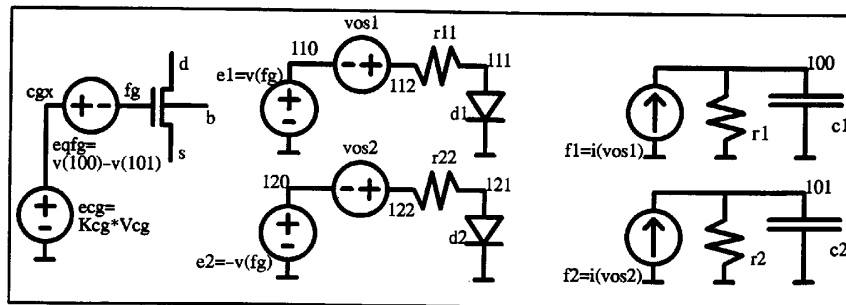


Fig. 10. A SPICE macromodel of the floating gate device.

monitoring of the error reduction, model based algorithms etc. The required control voltages in the power supply range can be easily generated and adjusted using standard digital to analog converters. A typical block diagram is shown in Fig. 5. A simple example is the algorithm shown in appendix B.

The above system and previously presented systems require a clock. The device presented allows also to construct a programming system for offset reduction that operates in continuous time. Aside from the high voltage handling problems stated above, the problem associated with a continuous time

```

variables:      vpm,vpp programming voltage amplitudes for charging and discharging
                dvp increment of programming voltage change
                a(i) records sign of previous pulses: H for vpp-pulse, L for vpm pulse
function
start
pulse(-5V)      *reset
vpp=0, vpm=0, a(0...n-1)=L *initialization
repeat vpp=vpp+dvp; pulse(vpp); *increase pulse size until target passed
until result<target-tolerance
repeat vpm=vpm-dvp; pulse(vpm) *decrease pulse size until target passed
until result>target+tolerance
mark1
*apply pulses according to deviation
if result>target+tolerance then *positive deviation-use vpp
*check for error changing sign twice in a row due to too small vpm
  if a(0)=L and a(1)=H then vpm=vpm+dvp
*check for result not changing over n pulses due to too small pulse vpp
  if a(0)*a(1)...*a(n-1)=H then vpp=vpp+dvp
  for i=n downto 1 do a(i)=a(i-1) *shift history register
  pulse(vpp); a(0)=H *set new history register entry
  goto mark1
elseif result<target-tolerance then *negative deviation-use vpm
*check for error changing sign twice in a row due to too large vpp
  if a(0)=H and a(1)=L then vpp=vpp-dvp
*check for result not changing over n pulses due to too large pulse vpm
  if a(0)+a(1)...+a(n-1)=L then vpm=vpm-dvp
  for i=n downto 1 do a(i)=a(i-1) *shift history register
  pulse(vpm); a(0)=L *set new history register entry
  goto mark1
else stop *stop when result within tolerance of target

```

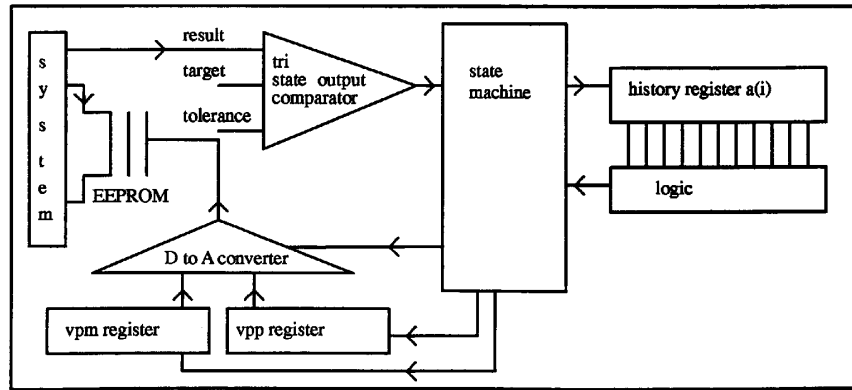


Fig. 11. A programming algorithm for smart bi-directional on-chip programming and its hardware requirements.

programming scheme is the following: Tunneling requires that the difference between floating gate voltage and tunneling injector voltage is large. Operation of the transistor is typically in saturation which requires that the floating gate potential is within normal power supply range. Programming control

requires that the effects of the floating gate charge change is not overridden by a change in potential at the control gate. Typically when programming by adjusting the voltage on the control gate this requirement cannot be met. Equation (1) shows the dependence of V_{fg} on both V_{cg} and Q_{fg} . An

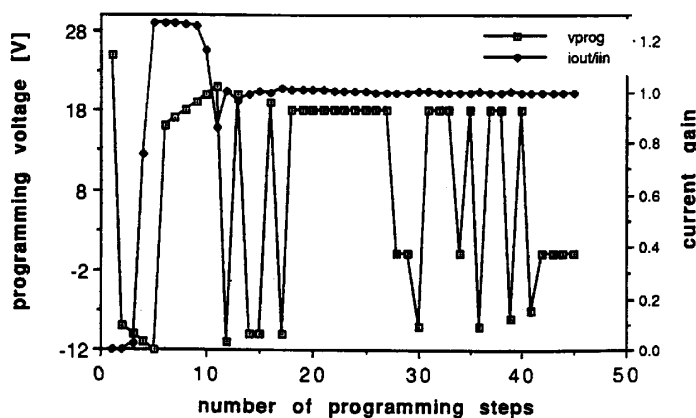


Fig. 12. Results obtained from a software implementation of the algorithm.

increase of potential due to V_{cg} is at least as large as the decrease due to the change in Q_{fg}/C_{tot} . The time constant associated with the change in Q_{fg} is much longer. As a result, the effect of V_{cg} dominates.

This problem can be overcome only if the full voltage is applied to the tunneling injector (which leads to high voltage breakdown problems as illustrated above) or in a differential setup where the voltage applied to the control gate is cancelled and only the charge difference affects the correction current. The circuit shown in Fig. 4 shows how common mode rejection of V_{cg} can be achieved and the differential output current ΔI remains only a function of Q_{fg} . It required for this setup that V_{cg} remain within the power supply range for the common mode rejection to work properly. The dual injector device is crucial to this scheme.

The circuit shown in Fig. 6 shows how the offset of an OPAMP can be reduced in continuous time by adding an adjustable differential current source. This source consists of a differential pair of two floating gate devices, one of which has active tunneling injectors, whereas the other serves only as a reference. Care has to be taken that no charge is stored on the floating gate of the reference device, for example by heating, UV erase, a discharge switch or application of V_{ti1} and V_{ti2} to its dummy tunneling injectors. Proper connection will insure negative feedback from the output voltage to the programming voltage. A positive programming voltage will decrease the floating gate charge, decrease the current through M1, which in turn should lead to a reduction in output voltage of the OPAMP, which is equivalent to the programming voltage. Due to the large time-constant associated with the tunneling effects it is expected that the system will be stable. The dominant pole of a folded cascode OPAMP is at the output/programming node where this effect is occurring. A simulation of this circuit was done using a qualitative SPICE macromodel of the floating gate device that is given in Appendix A. Fig. 7 shows the simulation result showing the offset voltage before and after trimming and also showing the change in

floating gate charge during the calibration phase. The offset is reduced to a remainder in the order of the deadband voltage divided by the OPAMP gain. In this example, the offset was reduced from 100mV to 7 mV by charging the floating gate to $Q_{fg}/C_{tot} = 220\text{mV}$. During the end of the calibration phase the output/programming voltage remains at a nonzero value that is at the border of the deadband. This voltage relates to the remaining input offset voltage by the inverse of the OPAMP gain.

Weight storage in neural networks is an important application for EEPROM devices due to their ability to store analog values permanently in a very compact area. While applications that require downloading of the weights from a simulation can rely on external programming control for the downloading period, complete handling of the programming control is required for on-chip learning of neural networks. This is another area where the properties of the presented device are advantageous because they allow to build fully integrated learning neural networks with semi-permanent weight storage. Since the presented learning schemes are not suitable for precise increments in floating gate charge, the device is not suitable to integrate the actual weight increment that is calculated in most learning algorithms. But if this accumulation is done elsewhere in the network, for example on a capacitor, a simple circuit (see Fig. 8) can be used to transfer this voltage into semi-permanent memory. The differential output current from the left differential pair has to match the differential output current from the right floating gate differential pair under equilibrium conditions. The differential output current of the right differential pair is replicated with an identical set of transistors connected to the same two floating gates. This current shows simulation results that verify the principle of operation. Fig. 9 shows simulation results that verify the principle of operation. The top graph shows the relation of input voltage, floating gate charge and output voltage, the middle graph shows the input and output differential currents and the third shows the value of the programming voltage

as created by the amplifier output. The visible errors are due to second order effects in the device models and can be reduced by using cascoding techniques in the current mirrors and differential pairs [9]. The stored weight on the floating gate will remain after the high static voltages are turned off at the end of the training period independent of the presence of signals or power supply. The circuit can be utilized with both global or local learning rules. It should be mentioned that a very robust learning algorithm is required for any analog implementation of learning due to the presence of nonidealities, such as uncertainty in device parameters and device mismatch.

V. CONCLUSION

The addition of a second tunneling injector to a standard process floating gate device can be used to overcome several problems for the application of EEPROMs in analog circuits. It is demonstrated that low control voltage programming of the device is possible, which is the key to integrated on-chip programming of EEPROMs. It is shown how the device can be used to implement smart programming algorithms for faster convergence and a fully integrated continuous time programming scheme. Applications in the area of amplifier calibration and neural network learning are demonstrated.

VI. APPENDIX

A. A Qualitative SPICE Macromodel

A qualitative model of the EEPROM device can be implemented in SPICE using the diode model to realize the exponential relationship of programming voltage and tunneling current. Below is a SPICE listing and circuit diagram for a qualitative model of the EEPROM (Fig. 10). The floating gate charge is modeled by a controlled voltage source EQFG which senses the voltage difference of two capacitors. The voltage between floating gate and ground is sensed by the controlled source E1. The voltage source Vos1 provides an offset voltage to represent the difference between tunneling threshold and tunneling injector voltage. It also serves as current sensing devices. The diode D1 provides the exponential relationship of voltage and current

$$I_d = I_s \times (\exp(q/kT(E1 + VOS1)) - 1).$$

The resistor R11 limits this current to provide better convergence of the simulator. This current is then integrated on capacitor C1. R1 is required by SPICE for operating point calculations and is large. This part represents the charging of the floating gate at tunneling injector 1. The discharging is represented by a similar circuit with reversed polarity of E2. The voltage difference between V_{c1} and V_{c2} represents the floating gate charge.

B. A Programming Algorithm for Floating Gate MOSFETs

This algorithm has small hardware requirements because it does not require as accurate modeling of the tunneling injector or storage of analog values. It is robust enough to converge in spite of nonuniformity of devices and trapup. Fig. 11 shows the listing of a program.

The floating gate is reset to a highly positive value by a pulse of -5V. Then the tunneling thresholds for charging and discharging of the floating gate are measured by stepping up the control voltage with step DVP and applying a pulse until the target value is passed and the output comparator changes state. The two programming voltages for negative and positive tunneling threshold, v_{pp} and v_{pm} are recorded in memory and applied iteratively according to the measured offset value until the result is within tolerance of the target. The polarity of the programming pulse is recorded in a shift register (history register) and kept there for 16 clock cycles. The content of the shift register is tested for two conditions: whether a polarity change occurred for two consecutive cycles, which requires a reduction of the programming voltage, and whether no polarity change occurred for 16 pulses, which requires an increase of programming voltage. The comparator has three output states, 'above', 'below', or 'within tolerance.' Extensive testing of this algorithm implemented in software has proven its robustness and convergence properties [8]. While it is not the fastest algorithm, it has small hardware requirements and reliable convergence. Fig. 12 shows results from a software implementation of the algorithm when applied to an adjustable current mirror. The different phases of the algorithm can be seen.

REFERENCES

- [1] D. Kahng and S. M. Sze, "A floating gate and its application to memory devices," *Bell Syst. Tech. J.*, vol. 46, p. 1283, 1967.
- [2] T. C. Ong, P. K. Ko, and C. Hu, "The EEPROM as an analog memory device," *IEEE Trans. Electron. Dev.*, vol. 36, pp. 1840-1841, 1989.
- [3] L. R. Carley, "Trimming analog circuits using floating-gate analog MOS-memory," *IEEE J. Solid-State Circuits*, vol. 24, no. 6, pp. 1569-1575, Dec. 1989.
- [4] E. Saeckinger and W. Guggenbuehl, "An analog trimming circuit based on a floating gate device," *IEEE J. Solid State Circuits*, vol. 24, pp. 1437-1440, Dec. 1988.
- [5] M. Holler, S. Tam, H. Castro, and R. Benson, "An electrically trainable artificial neural network (ETANN) with 10240 floating gate synapses," in *Proc. of Int. Joint Conf. Neural Networks*, vol. II, pp. 191-196, 1989.
- [6] A. Thomsen and M. A. Brooke, "A floating gate MOSFET with tunneling injector fabricated using a standard double polysilicon CMOS process," *IEEE Electron. Device Lett.*, vol. 12, no. 3, pp. 111-113, Mar. 1991.
- [7] M. Lenzlinger and E. Snow, "Fowler-Nordheim tunneling into thermally grown SiO_2 ," *J. Appl. Phys.*, vol. 40, pp. 278-283, 1969.
- [8] A. Thomsen, "High speed high accuracy signal processing with parallel analog circuits," Ph.D. dissertation, Georgia Institute of Technology, Dec. 1992.
- [9] P. Allen and D. Holberg, *CMOS Analog Circuit Design*. New York: Holt, Rinehart, and Winston, 1987.



Axel Thomsen (M'93) attended the Technische Universitaet Braunschweig from 1984 to 1988. He transferred to the Georgia Institute of Technology in 1988 and received his Ph. D. degree in Electrical Engineering in 1992.

He was a postdoctoral researcher at the Microelectronics Research Center from January to August 1993. Currently, he is Assistant Professor at the University of Alabama in Huntsville. His research interests are in the area of analog circuit design with poorly characterized devices, applications of floating gate devices in analog circuits, neural network implementations and applications, and on-chip trimming and learning of circuits.

In 1992, Dr. Thomsen won the IEEE MWCAS Myril B. Reed Best Paper Award (with Martin A. Brooke). He is a member of the IEEE.

Martin A. Brooke (S'85-M'86-S'86-M-87) received the B.E. (Elect.) degree (1st. Class Hons.) from Auckland University in New Zealand in 1981. He received the M.S. and Ph.D. in electrical engineering from the University of Southern California in 1984 and 1988, respectively. His doctoral research focused on reconfigurable analog and digital integrated circuit design.

He is currently an Analog Devices Inc. Career Development Assistant Professor at The Georgia Institute of Technology, where he teaches and performs research on analog circuits for high-speed communication and adaptive neural networks. He has published 24 refereed papers in journals and conference proceedings, and articles on his work have appeared in 15 different trade publications. He has one patent award and two pending. Professor Brooke has also developed software programs for analog CAD.

In 1990 Professor Brooke won the National Science Foundation Research Initiation Award, in 1992 the IEEE Midwest Symposium on Circuits and Systems Myril B. Reed Best Paper Award (with Axel Thomsen). He is a member of the IEEE and the INNS.