Real-time reconfigurable devices implemented in UV-light programmable floating-gate CMOS

by

Snorre Aunet

June 2002

A dissertation
submitted to the Norwegian University of Science and Technology
Faculty of Information Technology, Mathematics and
Electrical Engineering
in partial fulfillment of the requirements for the degree of
Doktor Ingeniør.

Abstract

This dissertation describes using theory, computer simulations and laboratory measurements a new class of real time reconfigurable UV-programmable floating-gate circuits operating with current levels typically in the pA to μA range, implemented in a standard double-poly CMOS technology. A new design method based on using the same basic two-MOSFET circuits extensively is proposed, meant for improving the opportunities to make larger FGUVMOS circuitry than previously reported. By using the same basic circuitry extensively, instead of different circuitry for basic digital functions, the goal is to ease UV-programming and test and save circuitry on chip and I/O-pads. Matching of circuitry should also be improved by using this approach.

Compact circuitry can be made, reducing wiring and active components. Compared to earlier FGUVMOS approaches the number of transistors for implementing the CARRY' of a FULL-ADDER is reduced from 22 to 2. A complete FULL-ADDER can be implemented using only 8 transistors. 2-MOSFET circuits able to implement CARRY', NOR, NAND and INVERT functions are demonstrated by measurements on chip, working with power supply voltages ranging from 800 mV down to 93 mV. An 8-transistor FULL-ADDER might use 2500 times less energy than a FULL-ADDER implemented using standard cells in the same 0.6 μm CMOS technology while running at 1 MHz. The circuits are also shown to be a new class of linear threshold elements, which is the basic building blocks of neural networks. Theory is developed as a help in the design of floating-gate circuits.

Preface

I have had the great pleasure of working with UV-programmable floating-gate ("FGUVMOS") circuits for a period. It has first and foremost been interesting and rewarding, but also hard work. Support from the surroundings has been of paramount importance:

I am very grateful to my supervisors, professors Trond Sæther, NTNU, and Yngvar Berg at the University of Oslo. They gave great support at various levels, including moral support, in seeking financial support, making various decisions, and writing and discussing technological issues.

Thanks also to associate professor Tormod Njølstad, Department of Physical Electronics, NTNU, for initiating my first meeting with FGUVMOS circuits by arranging a trip to the Department of Informatics, University of Oslo.

I would also like to thank the Microelectronics group at the Department of Informatics (Ifi), University of Oslo, for accepting me as an occasional coworker and guest during these years.

Dag T. Wisland (Ifi) deserves thanks for SW-support and laboratory support during my visits to Ifi. Shevana Bianca Aquariux was of great help in many discussions. Associate professor Nancy Lea Eik-Nes gave help with writing of the manuscript. Thanks to professor Steinar Andresen, NTNU, for financial support for producing chips, going to conferences and more, through the Telecom 2005 programme. Thanks to Sverre Vegard Pettersen for taking the chip photos.

Thanks also to colleagues at the Department of Physical Electronics, NTNU, for nice working conditions since May 1997, and lately the Department of Computer and Information Science, NTNU. Thanks also to my family and friends for help, support and patience.

Trondheim, June 2002

Snorre Aunet

Contents

List of Figures

Chapter 1

Introduction

1.1 The neuron-MOS and FGUVMOS circuit concepts

This work might be perceived as having it's startingpoint from two basic ideas, namely the multiple-input floating-gate transistor concept [ShOh91], and UV-programmable floating-gate circuits [LaWi96], [BeLa97], using the "FGUVMOS" UV-programming approach from [BeLa97].
A few important characteristics in a few words: In [ShOh91] multiple-input floating-gate circuits working in the classical above threshold regime are introduced. The device in [ShOh91] turns on when the weighted sum of all input signals exceeds a threshold. The number of transistors and interconnections can be drastically reduced by using multiple-input floating-gate devices in [ShOh91], a method that could entirely alter the way of constructing logic circuits, according to the authors of [ShOh91].
Dynamic power consumption depends linearly on the physical capacitance being switched [RaPe96]. Therefore, using fewer wires and fewer active elements for a given function may be attractive for minimizing power consumption.
FGUVMOS circuits utilize the multiple-input floating-gate transistor principle. A goal of FGUVMOS circuits has been to be able to reduce the supply voltage and adjust the effective threshold voltages of standard CMOS circuits at the same time [BeLa97], [BeWi98]. Voltage reduction offers the most direct and dramatic means of minimizing energy consumption [RaPe96]. By being able to adjust the current levels at the same time, the reduced performance following from a V_{dd} reduction can be counteracted. Supply voltages for FGUVMOS circuits have typically been in the 300 mV

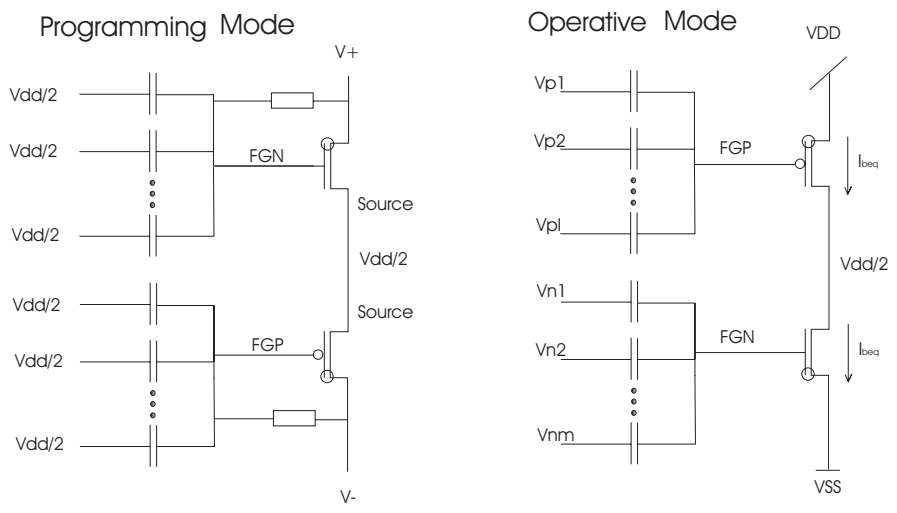

Figure 1.1: Schematic of the UV-programming mode and the normal operative mode, based on [BeLa97]. The extra circles in the MOSFET symbols indicate UV-conductances. In operative mode the equilibrium current is the drain current at the switching point, when all inputs and the output equal $V_{dd}/2$.

to 800 mV range. This, in combination with currents typically in the pA to μA range provide a significant low-power potential.
A shift in the effective threshold voltages, or current levels, is set during a UV-programming procedure. The UV-programming ensures a certain equilibrium current, I_{beq}, at the switching point of every basic FGUVMOS circuit element under normal operation. This switching point is ideal when the input voltages and output voltage are equal for all input voltages having a voltage of $V_{dd}/2$ under normal operation, shown to the right in figure 1.1. The equilibrium current, I_{beq}, might typically be in the nA to μA range for circuits implemented in an AMS 0.6 μm technology [AMS98]. UV-activated conductances between the power supply rails (V_{dd} and V_{ss}) are used to program the desired I_{beq} levels, as illustrated in figure 1.1. In a "reverse-biased" mode [BeLa97] during UV-programming the programming voltages, V_+ and V_-, on the power lines are used to control charge transport to and from the floating-gates, through the UV-activated conductances. The chosen pair of programming voltages (V_+, V_-) determines the equilibrium current level under normal operation, after the UV-light is

turned off. The shift in equilibrium currents means that a change in the effective threshold voltages, seen from the driving nodes, is done simultaneously. The circuits can be reprogrammed.
A future goal could be to program all transistors on a chip, or even a wafer, without using any additional programming circuitry [BeWi98].

Floating-gate digital circuitry like EXOR, NAND2, NOR2, INVERT,

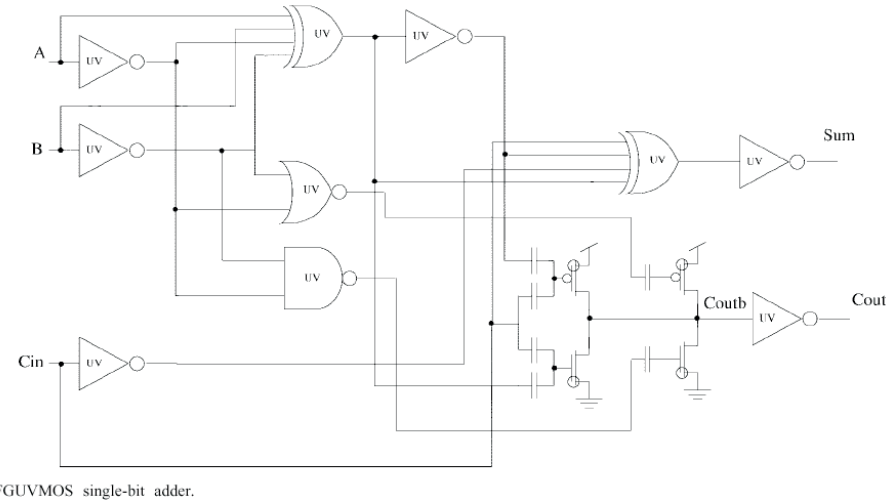

Figure 1.2: ©1999 IEEE. Reprinted, with permission, from IEEE Transactions on Circuits and Systems II, analog and digital signal processing, Vol. 46, Issue 7, July 1999.

a D-flip-flop, a FULL-ADDER and a pad-driver were published in 1999 [BeWi99]. Publications describing digital FGUVMOS circuits prior to work described in this thesis can be found in [LaWi96], [BeLa97b], [BeWi97], [BeWi98], [BeWi99]. The paper [BeWi99] demonstrated the ultra-low power potential of FGUVMOS circuits working with supply voltages far below 1 V and equilibrium currents in the 4 nA to 200 nA range. A FULL-ADDER presented is shown in figure 1.2.
FGUVMOS circuits are not restricted to digital use, and some recent research results can be found in [BeLa01b], [BeLa01a]. An introduction to

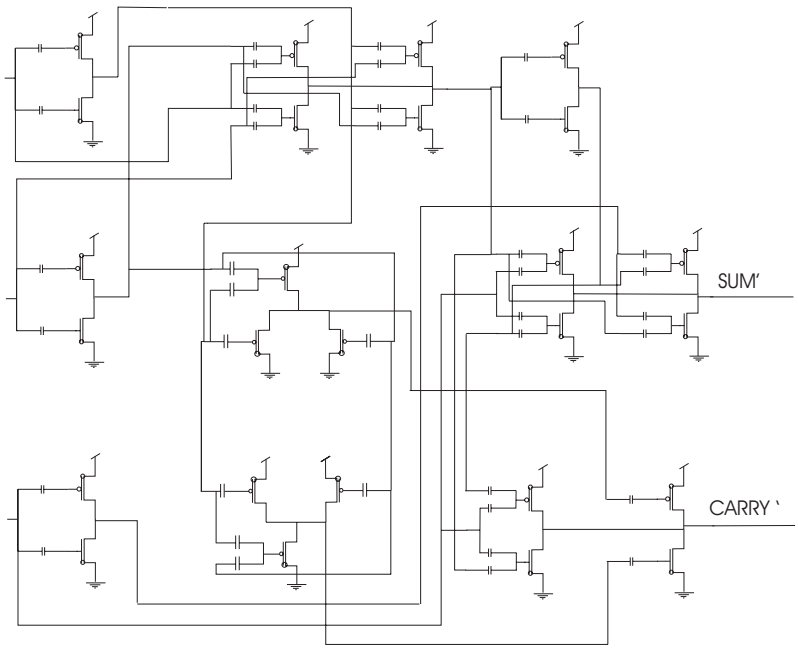

Figure 1.3: Schematics for FGUVMOS FULL-ADDER from [BeWi99] for producing SUM' and CARRY'.

the floating-gate research field can be found in a special issue of the IEEE Transactions on Circuits and Systems II, on floating gate devices, circuits and systems, from 2001 [HaLa01].

1.2 New real time reconfigurable floating-gate circuits

The digital functions in [BeWi99] used different basic circuitry for different functions like NAND, NOR and INVERT, resembling the standard concept of using the transistor as a switch only.
The real time reconfigurable circuit concepts presented in this thesis have undergone some evolution. It started with the idea of using three input circuits for the CARRY' function, which has the 2-input NAND, 2-input NOR and INVERT functionality embedded [AuBe01b]. This led to an attempt to utilize a FULL-ADDER concept from [KoSh92] to make a 4-transistor circuit able to generate SUM' and CARRY' for the FULL-ADDER function. This, together with inherent capabilities of the circuit to generate Boolean functions like NAND, NOR, INVERT, XOR and XNOR were discovered [AuBe01a]. The circuits in [AuBe01a] were both composed from two different basic building blocks.
Added to this is a universal threshold circuit concept [AuBe01d], which became used for a static memory and an adder in [AuBe01e]. The idea of building FGUVMOS circuitry using only one type of basic block was used here. Later measurements from chip confirmed the function of some basic building blocks.
It has been stated earlier that "previous experience with UV-programmable floating-gates suggested the only "proof of the pudding is the eating"" [LaWi96]. Even if the techniques and experience have been developed since that, some efforts have been done towards prototyping and laboratory measurements here as well. The AMS 0.6 μm technology have been used throughout this work, due to the experience gained from using it at the University of Oslo. Unless otherwise mentioned, the simulations are based on netlists extracted from layout. Test circuits in a 0.35 μm technology from another producer of integrated circuits have also been produced, but no laboratory measurements have been done yet.
The added functionality per active device compared to earlier approaches enables a reduction in the transistor count and amount of wiring, for implementation of many digital functions. The approach from [BeWi99] used the circuitry shown in figure 1.3 to compute SUM' and CARRY'. An implementation using one of the new circuits [AuBe01d] is shown in figure 1.4. The two schematics both implement the same functions. For example is the number of transistors used for implementation of the CARRY' function reduced from 22 to 2, or more than 90% [AuBe01b] by using the new approach. Also an 8-transistor FULL-ADDER is presented, built from 4

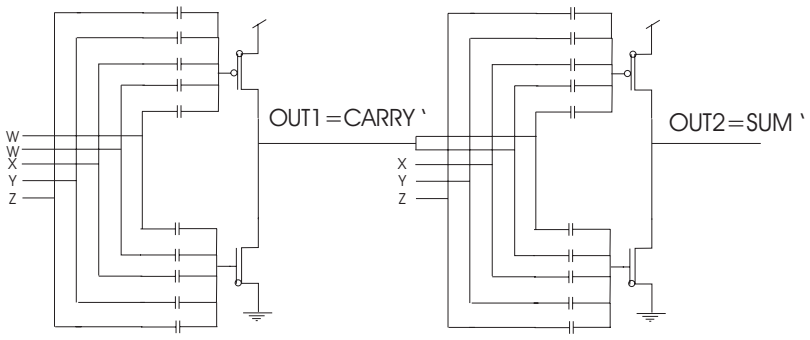

Figure 1.4: Schematic for circuit generating SUM' and CARRY' using two P5N5 elements.

of the elements shown in figure 1.4 [AuBe01d]. This is among the lowest numbers of active elements for FULL-ADDER circuits that are known from litterature. An even simpler version is shown in figure 1.5
A widespread use of identical basic building blocks on the gate level increases matching properties when circuits are implemented in CMOS, since MOS technology is well suited for fabrication of circuits where accuracy is determined by precision of ratio between elements, while absolute tolerances are much poorer [Sæth91].
Among contributions of the work presented here is that it is based on an approach using as little diversity among building blocks as possible. It is argued that it may ease UV-programming of FGUVMOS circuits a lot, as well as reduce wiring and transistor count. This might improve the low-power potential of digital FGUVMOS circuits.
Subthreshold circuits have been shown to consume orders of magnitude less power than the regular strong-inversion circuit at the same operating frequency [SoRo01]. FGUVMOS circuits using the standard programming scheme have always two transistors "stacked" between V_{ss}=0 V, and $V_{dd} \leq 0.8$ V. The current I_{beq} levels of the transistors can be set to the desired level, typically in the 1nA to 1μA range according to experience with the AMS 0.8 and 0.6 μm CMOS processes. This means that the circuits might work entirely in subthreshold in some cases, or in weak and moderate inversion. Moderate inversion means higher I_{beq} levels than weak inversion, and can be a useful region of operation to attain a relatively high operational speed of FGUVMOS circuits. For the FULL-ADDER function running at 1 MHz operational speed, it is argued that an 8-transistor

FGUVMOS FULL-ADDER uses about 2500 times less energy than a circuit based on standard library cells in the same technology.
The basic building blocks proposed all consist of 2 transistors and a number of drawn capacitances between inputs and the floating gates of the PMOS and the NMOS transistors. Chosen restrictions for the capacitances have been that all drawn capacitances were of equal size and shape, and that the sum of the drawn capacitances connected to the floating gate of the PMOS transistor equals the sum of the drawn capacitances connected to the floating gate of the NMOS transistor, of a basic circuit. The number of inputs to the PMOS and NMOS, respectively, are used for naming the circuits. P1N3 means a circuit with one capacitively weighted input to the PMOS, and three capacitively weigthed inputs to the NMOS.
For the inverter functionality is demonstrated by chip measurements with a V_{dd} of only 93 mV, which is amongst the lowest supply voltages reported in litterature.
Among recent work on subthreshold digital logic, work like [SvMa00] and [SoRo01] can be mentioned. Subthreshold digital logic in most cases rely on some substrate bias stabilization, as shown in [SvMa00], [SoRo01]. Subthreshold operation is common with most FGUVMOS circuits, but the multiple-input floating-gate MOSFET capabilities leading to real time reconfigurable logic functions might be unique.
As an example, the P1N3 element used twice in 1.5 may be used as a circuit producing each and every of the CARRY', NAND3, NAND2, NOR3, NOR2 and INVERT functions, depending on voltage on one or several control inputs.
A "P7N7" building block is also used in making a 3-bit analog-to-digital converter containg only 6 transistors.
In addition the new circuits were recognized as belonging to the class of circuits called linear threshold elements, the name of basic computational units in neural networks [AuBe02a].
Building blocks presented here are also currently under investigation as building blocks for systems based on evolvable hardware [Eske02] These are examples that the circuits proposed may find use in other areas than a pure digital context.

During the work on this thesis, several of the results have been published at international conferences with peer review: [AuBe01a], [AuBe01b], [AuBe01d], [AuBe01e], [AuBe02a], [AuBe01c], [BeAu01a], [BeAu01b], [BeAu01c], [BeAu01d], [BeAu01e], [YtAu02], [BeAu02], [BeNa02b], [BeNa02c], [BeNa02d]. The first five publications mentioned form the core of this thesis.

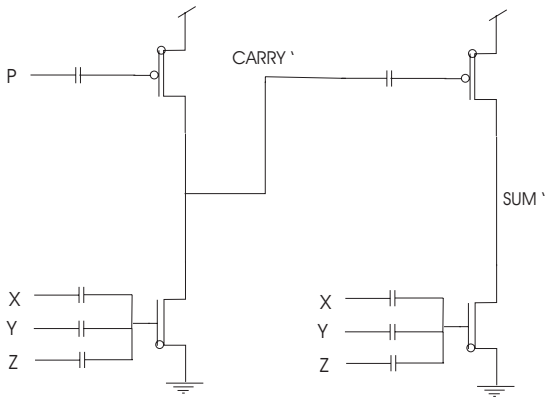

Figure 1.5: Two P1N3 elements used for implementing CARRY' and SUM'.

1.3 Thesis outline

In chapter 2.1 important properties of the multiple-input floating-gate MOSFET operating with current levels in the pA to μA range are explored, from an ordinary MOS transistor up to behavior in a circuit. At the end some parameters are pointed out that can provide insight in how transconductance and output resistance of these devices might be adjusted. Then some widely used equations for the universal FGUVMOS element is introduced with an inverter as example, in chapter 2.2. One of our circuit building blocks are briefly introduced in chapter 2.3.
In chapters 3, 4, 5, 6 and 7 there are descriptions regarding the function and implementation of the different circuitry in each chapter, and results and discussion following . This have been done to sum up the discussion in portions, and later add them up in a hierarchical manner.
Chapter 3 describes PMOS and NMOS transistors in the above mentioned manner.
In chapter 4 these elements are combined for inverters, with the chapter including simulations, IC layouts, UV- programming and test setup, results and discussion.
Chapter 5 deals with 2-MOSFET floating-gate elements with identically weighted inputs to both PMOS and NMOS, with a structure having much in common with chapter 4.
Chapter 6 is made basically like chapter 5, but this time the circuits have the signals used as ordinary signals, and not control inputs, connected to the NMOS floating gate only.
Chapter 7 introduces circuits made from the previously described building blocks. Another layer in the "discussion hierarchy" is added for each of them, making connections back to the basic inverter in chapter 4, and the transistors from chapter 3. FGUVMOS "Inverter-only" based logic is introduced. Linear threshold functions and implementation using FGUVMOS are discussed. New possibilities regarding UV-programming and testing are also discussed in 7.7.
Chapter 8 describes the overall results and major contributions, as well as suggestions for further work.

Chapter 2

Multiple-input UV-programmable MOSFETs and two-MOSFET circuits

2.1 MOSFETs

2.1.1 MOSFETs in weak inversion

For an NMOS transistor operating in subthreshold, or more precisely weak inversion, the drain current, I_{ds}, is given by [AnBo91]:

$$I_{ds,n} = I_0 exp\{\frac{\kappa V_{gs}}{U_t}\} exp\{\frac{(1-\kappa)V_{bs}}{U_t}\}(1 - exp\{\frac{-V_{ds}}{U_t}\} + \frac{V_{ds}}{V_0}). \qquad (2.1)$$

For gate voltages well below the threshold voltage, V_t, of the transistor, the transistor may be said to work in subthreshold [Mead89], instead of the classical above threshold region.

Other names for operating regions of the MOSFET are "leakage-affected region", "weak inversion", "moderate inversion" and "strong inversion" [Tsiv99]. If taken in the above mentioned order they mean different nonoverlapping regions with increasing drain currents, if they were to be plotted in figure 2.1, as can be seen from in [Tsiv99] p. 45. For practical purposes I_{ds} is exponentially related to V_{gs} in weak inversion. In weak inversion, diffusion contributes to the larger part of the drain current, while drift dominates

in strong inversion. In moderate inversion, both drift and diffusion contribute significantly to the value of the drain current. Weak and moderate inversion are the areas used for practical implementations of FGUVMOS circuits in this thesis.

I_0 is the zero-bias current for the current for the given device [AnBo91], a constant where all preexponential constants have been absorbed [Mead89]. This includes W and L, the channel width and length of the MOSFET structure, respectively. V_{gs} is the gate-to-source potential, V_{ds} is the drain-to-source potential and V_{bs} the substrate-to-source potential. V_0 is the Early voltage, which is proportional to the channel length. κ measures the effectiveness for which the gate potential is controlling the channel current. It is often around 0.7 - 0.75 [Mead89], [AnBo91]. Boltzmann's constant, $k = 1.38 \cdot 10^{-23}\ J/K$, and elementary charge, $q = 1.602 \cdot 10^{19}$ Coulomb. At room temperature $T = 300$ degrees Kelvin, and thus $U_t = 25.8mV$.

For devices in saturation within weak inversion, when $V_{ds} \geq 4U_t$, neglecting the early effect and the body effect [AnBo91]:

$$I_{ds,n} = I_0 exp\{\frac{\kappa V_{gs}}{U_t}\} \tag{2.2}$$

In figure 2.1 the drain current of an NMOS transistor as a function of the gate-source voltage for different drain-source voltages is simulated. For FGUVMOS circuits we have a maximum equilibrium current, I_{beq}, which often is in the microampere to nanoampere range, for the AMS 0.8 CMOS technology [BeWi99]. The voltage between drain and source, V_{ds}, adjusts the current level, but it is mainly dependent on V_{gs}. Sometimes only V_{gs} is taken into account, like in [KoGo01].

The factor $\kappa < 1$ reduces the impact of the changing of the gate voltage. If the exponential dependence of V_{gs} on the drain current is taken into account, some of the effect of κ can be shown. The exponential function, $exp(ax)$, for $a = 0.1, 0.2, ..., 1.0$, is shown in figure 2.2. When κ decreases the effect of a change in V_{gs} is reduced, analog to this figure. Simulating a transistor using for example the BSIM3V3 model displays a more complex behavior. The drain current depends on several additional parameters.

An approximated κ has been extracted from figure 2.1. In the case when I_{ds} varies between $1\mu A$ and 1 nA, V_{gs} must change in average about 89 mV to change the current by a factor of 10. κ is relatively constant in the 10 pA to 1 μA range. For the ideal exponential function this would only have to be 60 mV. If $U_t = 25.8$ mV is used as unity voltage, and an 89 mV change in V_{gs} for a 10-fold change in I_{ds} is needed:

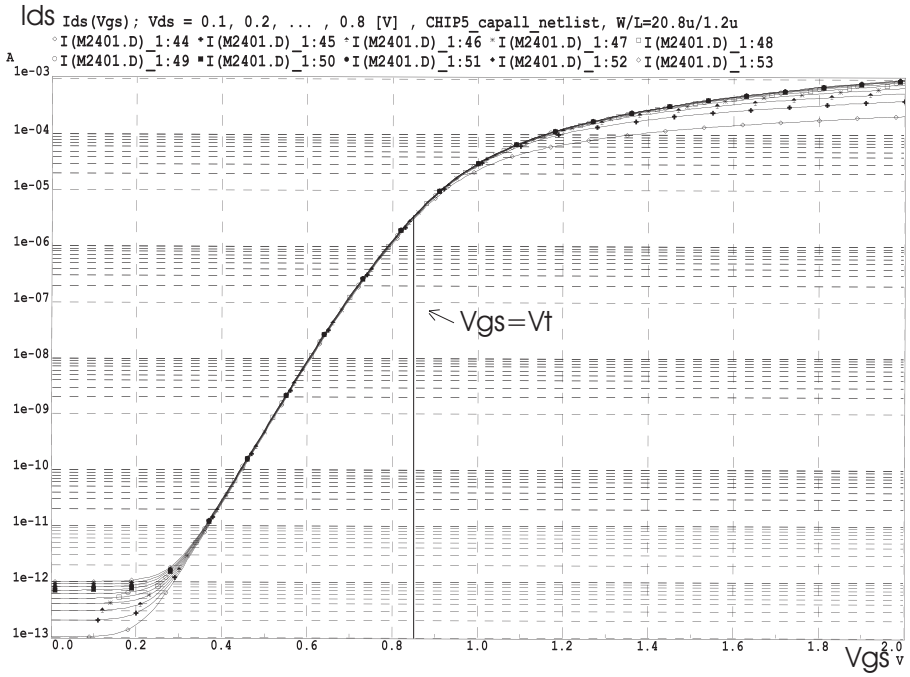

Figure 2.1: I_{ds} is shown as a function of V_{gs} for $V_{ds} = 0.1, 0.2, ..., 1.0[V]$. The size of the NMOS transistor is W/L=$20.8\mu m/1.2\mu m$. The threshold voltage is approximately 0.85 V [AMS98]. The actual W/L ratio is used for several of the circuit structures realized.

$$\kappa \frac{89}{25.8} = ln(10) \tag{2.3}$$

This gives $\kappa = 0.67$, compared to the κ value of 0.7 mentioned in [Mead89] p. 38. In figure 2.3 simulation results are shown where V_{gs} differs by 100 mV for each curve. When V_{ds} is above about 100 mV it is easy to see that the current level increases drastically for these increases in V_{gs}, while it is less dependent on the V_{ds} level. When $V_{ds} > 4kT/q$ the transistor is in saturation [AnBo91]. The small-signal transconductance in saturation is [AnBo91], [IsFi94]:

$$g_m = \frac{\Delta I_{ds}}{\Delta V_{gs}} = \frac{\kappa I_{ds}}{U_t} \tag{2.4}$$

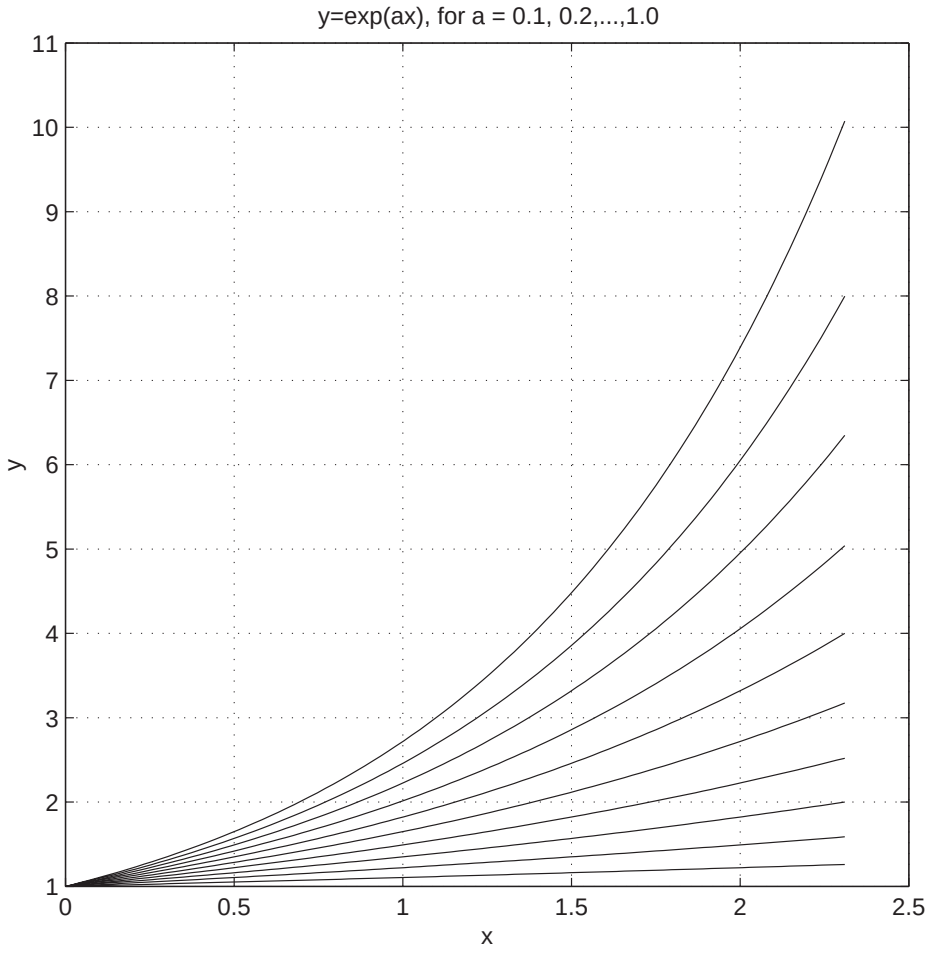

Figure 2.2: $exp(ax)$, for $a = 0.1, 0.2, ..., 1.0$. ln(10)=2.302 is how much the exponent must increase in order to increase the functional value by 10 times.

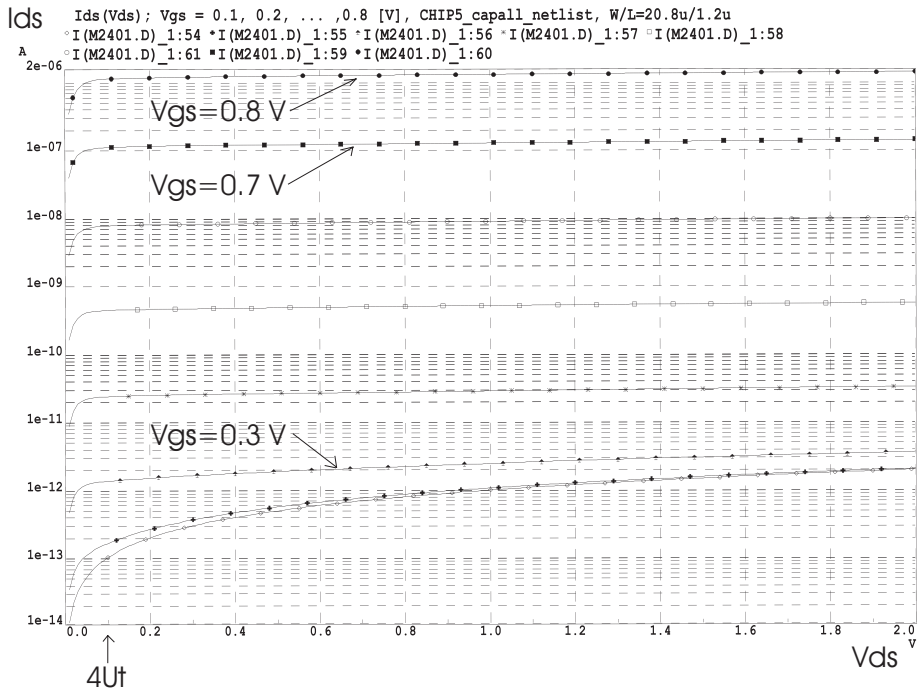

Figure 2.3: I_{ds} is shown as as a function of V_{ds}, for $V_{gs} = 0.1, 0.2, ..., 0.8[V]$. The current increases when V_{gs} increases. W/L = 20.8μm/1.2μm. The NMOS netlist used for simulation is extracted from layout.

Simulated transconductances for an NMOS transistor with $W/L = 20.8\mu m/1.2\mu$m are shown in figure 2.4. The output conductance is given by [AnBo91]:

$$g_{ds} = \frac{\Delta I_{ds}}{\Delta V_{ds}} \tag{2.5}$$

For V_{ds} greater than or equal to $4U_t$, the output conductance may be written [AnBo91]:

$$g_{ds} = \frac{I_{ds}}{V_o} \tag{2.6}$$

The conductance is proportional to the drain current, I_{ds}, for a given transistor.

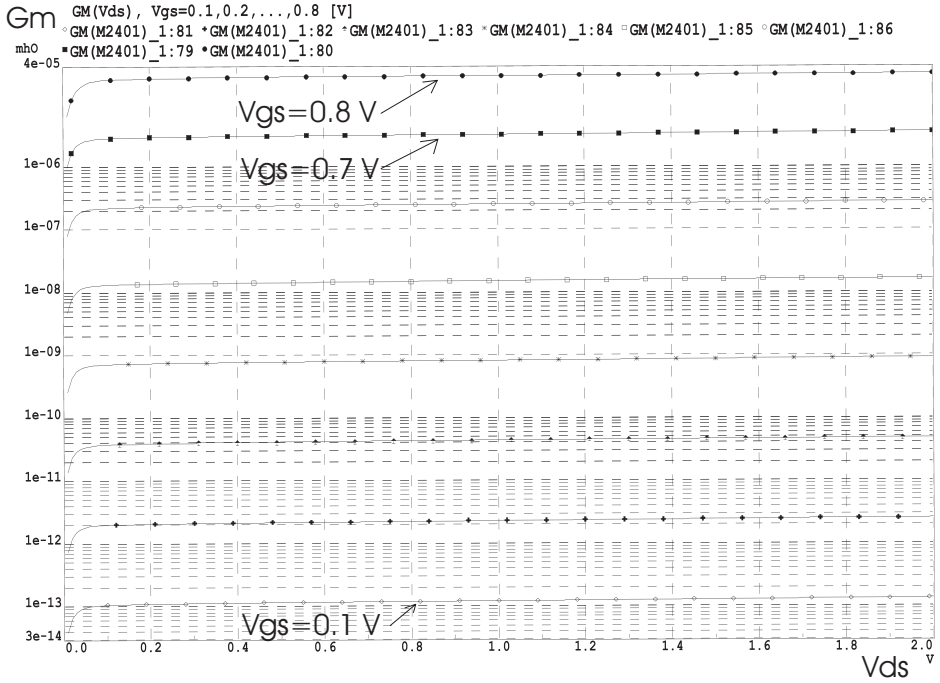

Figure 2.4: Transconductance, g_m, is simulated as a function of V_{ds} for $V_{gs} = 0.1, 0.2, ..., 0.8[V]$. In the saturation region, g_m increases proportionally to I_{ds}.

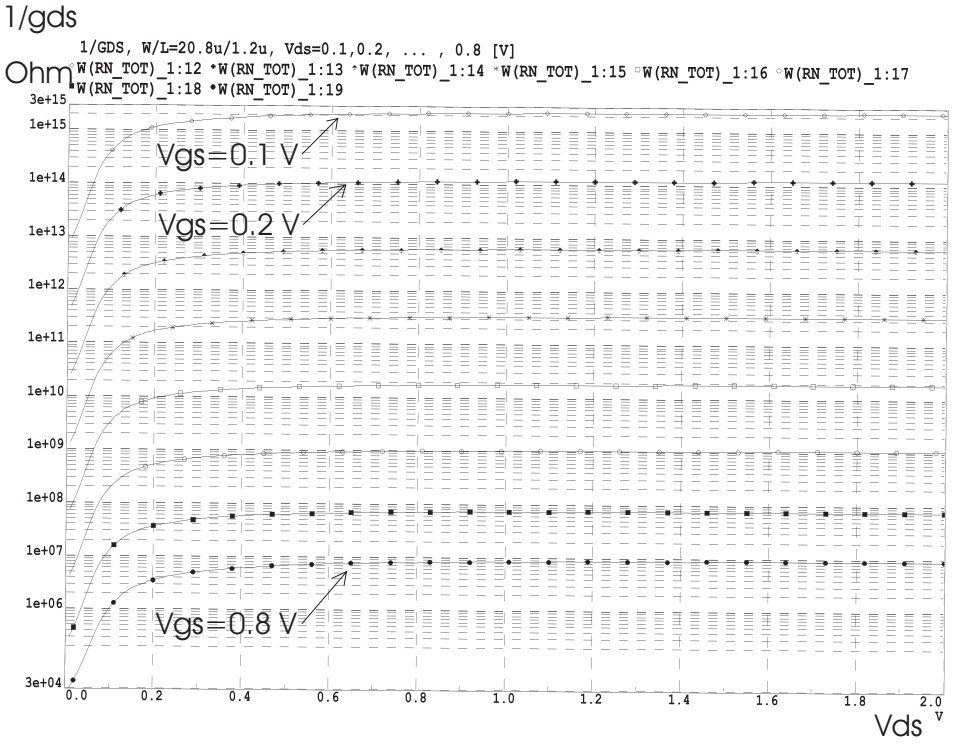

Figure 2.5: Output resistance is simulated as a function of V_{ds} for $V_{gs} = 0.1, 0.2, ..., 0.8[V]$. The output resistance decreases for an increase in V_{gs}.

$$r_{out} = \frac{1}{g_{ds}} \tag{2.7}$$

Output resistance is depicted in figure 2.5. The output resistance decreases proportionally with an increasing V_{gs}, which at the same time means a growing current level.

In the common-source mode the transistor can be used as an inverting amplifier with voltage gain

$$A = \frac{g_m}{g_{ds}} = \frac{\kappa V_o}{U_t} \tag{2.8}$$

which is constant for a given temperature, and for a V_o of 15 V gave a gain of 430 in [AnBo91].

2.1.2 Floating-gate CMOS transistors

From practical reasons the floating-gate UV-programmable circuits are most often modeled in the weak inversion region, even if the operation might be partly in moderate and strong inversion. This is also used throughout this thesis.
If an input signal to the gate of a MOSFET is connected via a capacitor, C_n, we have a floating-gate transistor, like in figure 2.6. For practical purposes, the gate has no DC path to ground. The charge on the floating gates, and thereby the effective threshold voltages, can be changed by Fowler-Nordheim tunneling, Hot Electron Injection or UV-light, in different ways [HaLa01], [KoGo01]. To illustrate the behavior we make the assumption that the NMOS floating gate "sees" a total load capacitance, C_l, between the floating gate and the source [LaWi96]. This C_l is not constant in reality, but is treated as such in this simple model, illustrated in figure 2.6. The poly layer forming the floating gate does contribute to C_l as well, so that increasing C_n leads to an increase in C_l. C_l also incorporates parasitic capacitances from the transistor, seen from the floating gate.

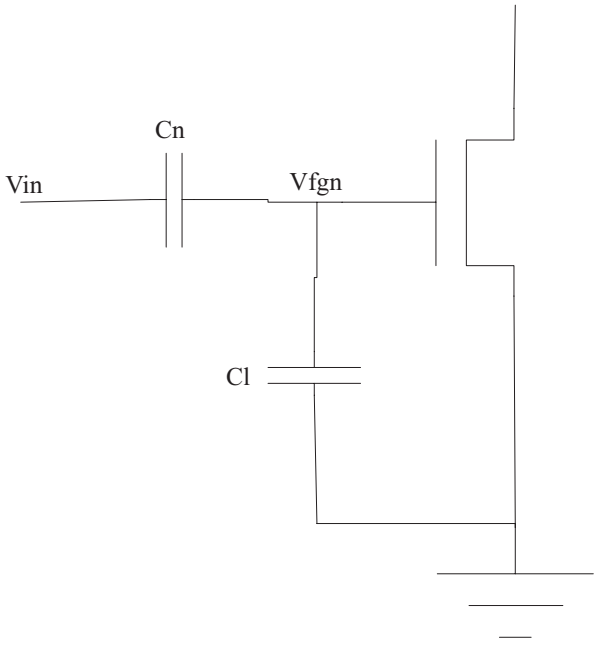

Figure 2.6: Simple model of floating-gate transistor.

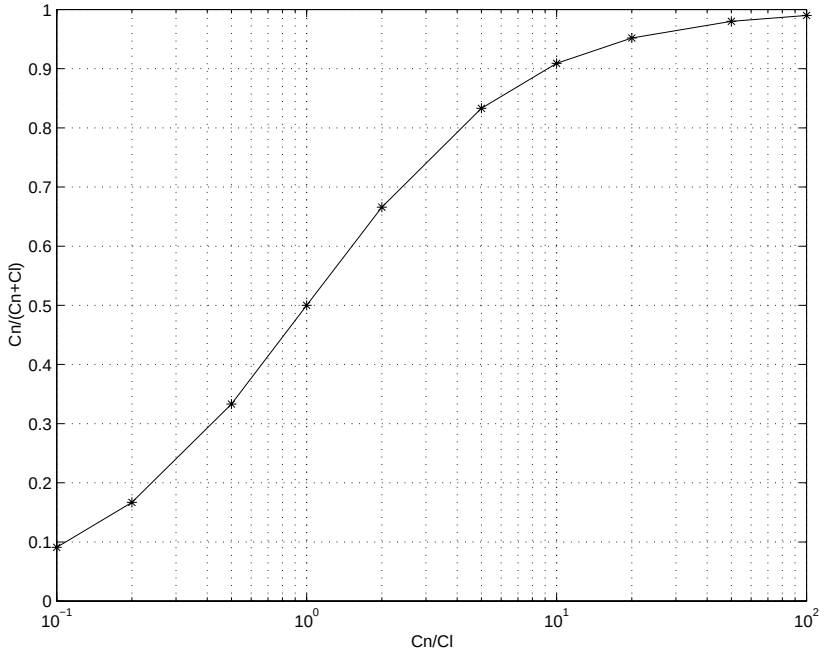

Figure 2.7: $\frac{C_n}{C_n+C_l}$, capacitive division involving the drawn capacitance between input and floating gate and parasitics.

$$\Delta V_{fgn} = \frac{C_n}{C_n + C_l} \Delta V_{in} \tag{2.9}$$

The amount of the signal V_{in} in figure 2.6 that slips through to the floating gate, V_{fgn}, depending on the ratio between the coupling capacitor and the load capacitor; $C_n/(C_n+C_l)$, is depicted in figure 2.7. In the AMS 0.8 CMOS process a coupling capacitance, C_n, of twice the size of the load capacitance, C_l was implementable without any area penalty [LaWi96]. As can be seen from figure 2.7 that case would allow approximately two thirds of the signal perturbation on the input through to the floating gate, while equally sized C_n and C_l would let 50% through.

In figure 2.8 it is demonstrated, with our $20.8\mu m/1.2\mu$m MOSFET and for a κ of 0.67, how much signal is needed on the floating gate to make the current change a certain amount, assuming that it follows a simple exponential function. The second curve from the top shows the function for $\kappa = 0.67$, and the uppermost curve the unrealistic case (for standard CMOS) when $\kappa = 1.0$. For equally sized C_n and C_l, letting 50% of the input

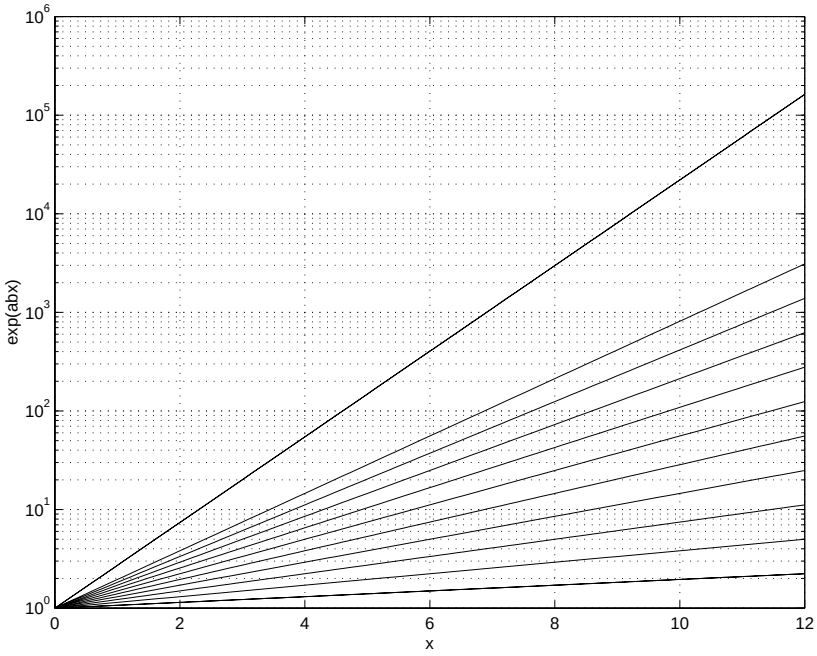

Figure 2.8: $exp(0.67ax)$, for $a = 0.1, 0.2, ..., 1.0$, and the case $exp(x)$

signal perturbation through to the floating gate, and a κ of approximately 0.67, the relevant line in figure 2.8 is the 5th one counted from the bottom (of 11). From that curve a change of 7 units along the horizontal axis would lead to a little bit more than 10 times change in the output value. If this is translated to an NMOS transistor with U_t =26 mV as unity, this would correspond to a ΔV_{gs} of 7 times 26 mV, or 182 mV, needed to change the magnitude of the drain-current, or output conductance of a single transistor roughly a factor of 10. In comparison an ideal MOSFET with $\kappa = 1.0$ and $C_n/(C_n + C_l) \approx 1$ would require a ΔV_{gs} of 60 mV to produce the same change in the output current under the same circumstances. Taking such damping of the input into account gives the following form of the equation for the current in saturation

$$I_{ds,n} = I_0 exp\{\frac{\kappa V_{gs}(\frac{C_n}{C_n+C_l})}{U_t}\} \tag{2.10}$$

Using more than one capacitively coupled input signal to the floating gate of a transistor adds properties to floating-gate transistors, as will be treated later in this thesis.

2.1.3 Single-input floating-gate MOSFETs

Since the sizes of the capacitances between one or several input signals relative to parasitics determines how much of the input signal that gets through to the floating gate, and thereby the transconductance and output resistance of a device, this is examined somewhat further. Firstly a very simple model is mentioned, thereafter one slightly more advanced. SPICE-based simulations using a basic floating-gate inverter are used to illustrate some of the behavior of the relevant capacitances associated with the floating gate. C_{ox} per area of a large area capacitor is calculated from [AlHo87]

$$C_{ox} = \frac{\epsilon_{ox}}{t_{ox}} \tag{2.11}$$

where ϵ_{ox}=34.515 pF/m, and the thickness of the gate insulator, t_{ox}, is typically 12.5 nm [AMS98]. The value of integrated circuit capacitors can be approximated by $C = C_{ox}A$, where A is the area of the capacitor [AlHo87]. Using a simple model for manual analysis, the gate capacitance, C_g, equals $C_{ox}WL$, and can be decomposed in a number of elements with different behavior, including parts solely dependent on the topological structure of the device, and other nonlinear capacitances depending on applied voltages on the MOSFET terminals [Raba96]. For a device with W=20.8μm and L=1.2μm, C_g is estimated to 68.9fF using our technology of choice.

In reality, both source and drain tend to extend somewhat below the oxide by an amount x_d, called the lateral diffusion. This means that the effective channel length of the produced transistor, L_{eff} becomes shorter than the drawn length, or the length the transistor originally was designed for, and also leads to parasitic capacitances between the gate and the source and drain terminals. These are called overlap capacitances [Raba96], and have have fixed values.

Some slightly more complex view on the parasitic capacitances are briefly mentioned here, in an attempt to provide a brief view of factors providing a little bit more realistic picture. Different models take into account different approaches or parameters influencing the floating gate, like in [WoLi92], [YaAn93], [ChKi94], [FuAr01]. Figure 2.9 is taken from [YaAn93], and it shows that the floating gate voltage, V_{fgn}, is dependent on the voltage on the input(s) as well as voltages coupled from source, substrate and drain. The subthreshold capacitance models used in [FuOm98], [YaAn93] take 4 parasitic capacitances into account. Capacitances C_{gs}, C_{gd}, and C_{gb} are parasitic capacitances from the (floating-) gate to the source, drain and bulk, respectively. C_{ox} is the oxide capacitance. If there is one drawn capacitance, C_n, between the only input, V_{in},

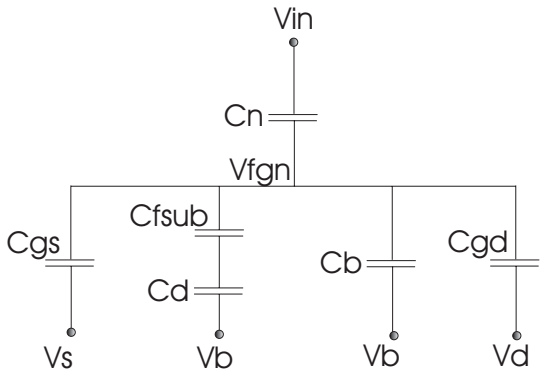

Figure 2.9: Capacitor model of a floating-gate MOSFET [YaAn93]. V_{cg}, V_s, V_b, V_d denote voltages on the input, source, bulk and drain, in that order.

and the floating gate, we can denote the sum of capacitances "seen from" the floating gate as:

$$C_{sum} = C_n + C_{gs} + C_{gd} + C_{gb} + C_{ox} \tag{2.12}$$

The weighted sum of voltages coupled via capacitances is in many cases not the only factor determining the voltage on a particular floating gate. A charge Q_g can be left after production of the chips, or it can be set by Fowler-Nordheim tunneling, hot-electron injection or UV-illumination. UV-illumination might also be used to remove all charge from the floating gate, so that $Q_{fg} = 0$ [FuOm98]. After production the floating gate voltages are somewhat random, even for identically drawn structures [BeWi99]. The voltage on the floating gate of the NMOS, V_{fgn}, can be written

$$V_{fgn} = \frac{C_n V_{in} + C_{gs} V_s + C_{gd} V_d + C_{gb} V_b + C_{ox} V_{sur} + Q_{fg}}{C_{sum}} \tag{2.13}$$

where V_{sur} is the surface potential [YaAn93], [FuOm98]. The parasitic capacitances may be nonlinearly dependent on applied voltages. Some parasitics in the subthreshold capacitance model [FuOm98], [WoLi92], [YaAn93] have been simulated using the Eldo simulator , for supply voltages of 200 mV and 800 mV, and I_{beq}-levels in the 1nA to 1000 nA range. The circuit is an inverter and the netlist extracted from layout ($CHIP5\_capall\_netlist$) [Aune02]. They are extracted from a transient simulation like the one in

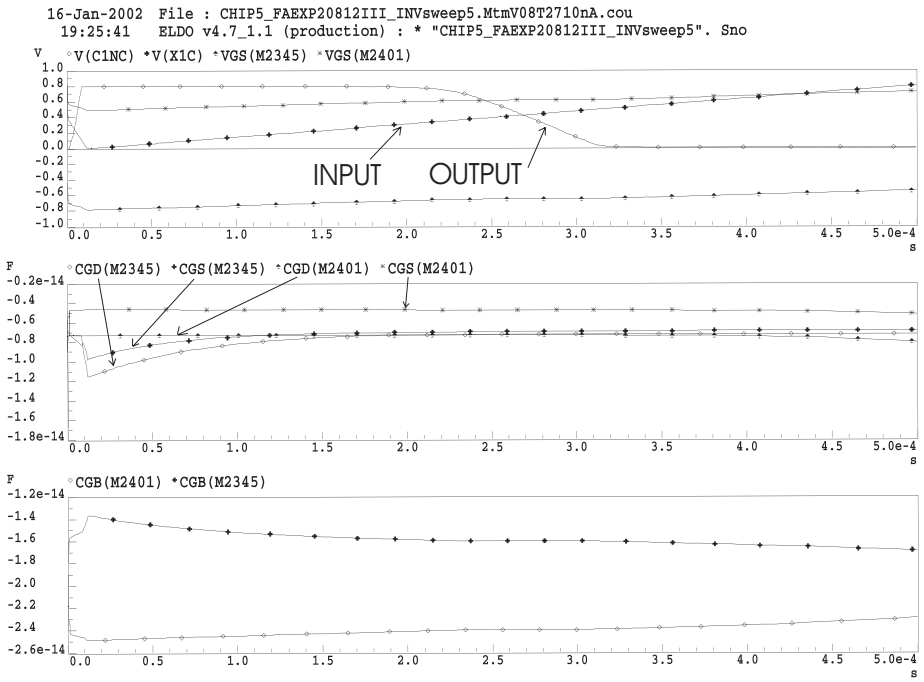

Figure 2.10: Capacitances related to the 20.8μm/1.2μm PMOS and NMOS are shown. "CGS(M2401)", for example, means the capacitance between gate and source for the NMOS in the inverter.

$V_{dd}[V]$	I_{beq}[nA]	C_{gsp} [fF]	C_{gdp}[fF]	C_{gsn}[fF]	C_{gdn}[fF]	C_{gbp}[fF]	C_{gbn} [fF]
0.2	1	6.9	7.3	4.7	7.3	16	24.5
0.2	10	7.0	7.3	4.7	7.3	15.9	24
0.2	100	8.5	7.3	5.0	7.3	15.1	23.3
0.2	1000	27.3	7.7	9.4	7.4	10.2	21.4
0.8	1	6.9	7.3	4.7	7.3	16.7	24
0.8	10	7.0	7.3	4.7	7.3	16	24
0.8	100	8.4	8.4	5.1	7.3	15	23.4
0.8	1000	27	6.7	9.6	6.7	10	21

Figure 2.11: MOSFET average channel capacitances for different operation regions have been simulated. The "n" in C_{gsn}, for example, means the capacitance between gate and the source of an NMOS transistor. The unexpected relation $C_{gd} > C_{gs}$ may come from a misbehaviour in the netlist extraction tool.

figure 2.10. The values used are taken from the time of the simulation where the output voltage, V(C1NC), changes from 0.8 V to 0 V (figure 2.10), around a time of 0.26-0.27 ms. V(X1C) means the input signal and VGS(M2345) and VGS(M2401) (figure 2.10) mean the gate-to-source voltages of the PMOS and NMOS respectively. Results can be found in figure 2.11. The parasitic capacitances are relatively constant until the I_{beq} changes from 100 nA to 1000 nA. The parasitic capacitances influence a range of important parameters such as voltage gain and transconductance. This makes them interesting to model, for example for the switching region of subcircuits, where voltage gain might be especially important.

To illustrate the effect of sizing of drawn capacitances between the input and a floating gate, some simulations relevant to circuitry integrated on chip have been done. The input to an inverter were changed abruptly in steps, and the resulting steps on each of the two floating gates were measured, close to the switching point. Simulations as depicted in figure 2.12 were done, using capacitances of 74.6 and 122.7 fF, and an I_{beq} of 10 nA, measured at about 0.4 ms. Results are shown in figure 2.13, demonstrating that the bigger capacitance out of the two let more of the changing input signals through to the floating gate than the smaller one.

In figure 2.14 the inverter function of a P5N5 element is both simulated and measured on chip. The total change of the drain current of the inverting element spans between two and three decades of magnitude both for the

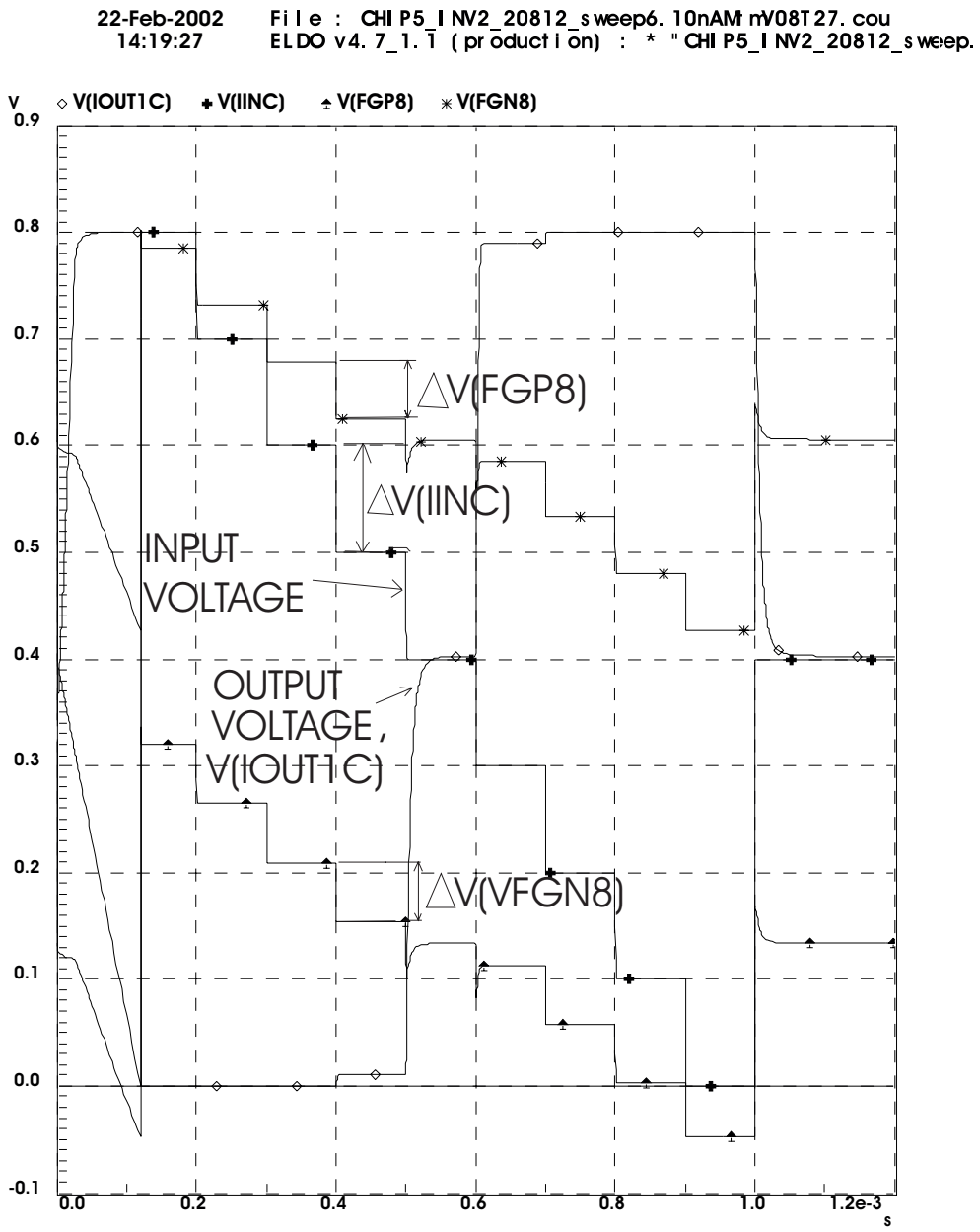

Figure 2.12: The input signal, V(IINC) is changed in 100 mV steps, and the floating gate voltages, V(FGN8) and V(FGP8), for NMOS and PMOS respectively, are shown. V(IOUTC) is the output voltage of the inverter.

$C_n[fF]$	$\Delta V_{fgp8}/\Delta V_{iinc}[\%]$	$\Delta V_{fgn8}/\Delta V_{iinc}[\%]$
73.6	54.1	52.5
122.7	59.4	57.9

Figure 2.13: Larger capacitances between the input and the floating gates let more of the input signal through to the floating gates than do the smaller ones. In this case the voltage change on the floating gate of the PMOS and NMOS increased by 5.3% and 5.4%, due to C_n increasing from 73.6 fF to 122.7 fF.

simulations and the measurements, and seem to be in reasonable agreement. The inverter function was made by using a multiple-input floating-gate circuit as an inverter, though the same type of building block could also be used as a generator of the inverted carry, 3-input NAND, 2-input NAND, 3-input NOR or 2-input NOR [AuBe01e].

Though some simple capacitance models are briefly mentioned in this chapter, computer tools have been widely used in this work to model different capacitances, since trying to take all relevant effects into account into a manual, first-order analysis results in intractable and opaque circuit models, according to p. 51 in [Raba96]. Computer simulations have been done for devices with similar dimensions as on later produced on chips, in an attempt to find out some of what could be expected from later chip measurements. In figure 2.14 there seems to be a reasonably good agreement between simulations and measurements, as in a similar comparison in [AuBe01c].

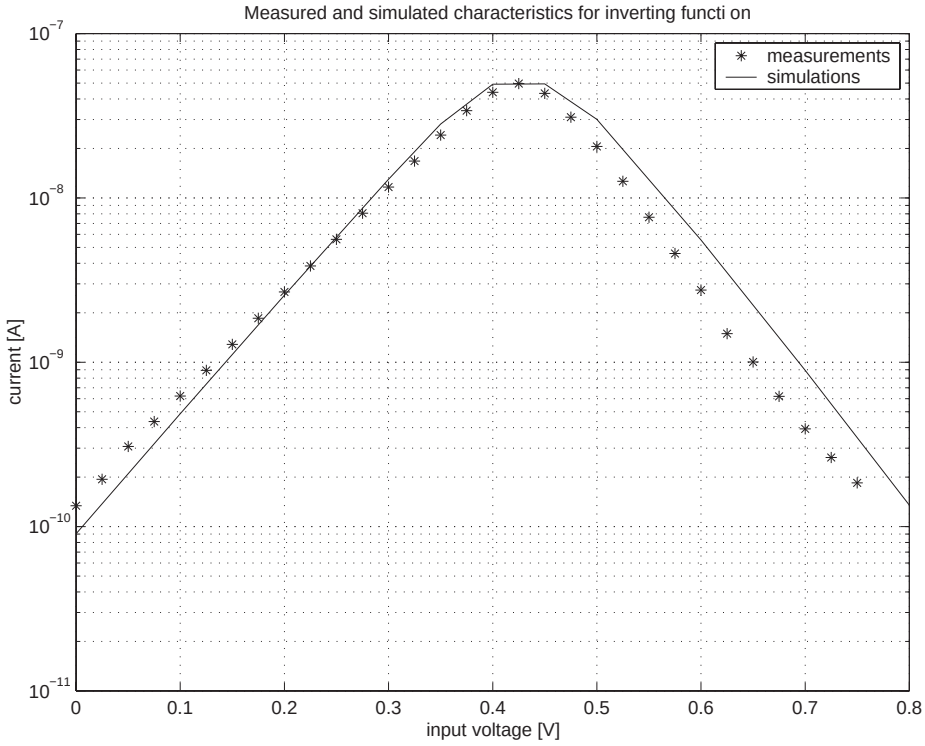

Figure 2.14: Measurements and simulations of current through the P5N5 circuit, as a function of input voltage, are demonstrated. The circuit is used as an inverter in this case.

2.1.4 Multiple-input floating-gate CMOS transistors

Each floating gate can have one or several capacitively weighted inputs, illustrated for two inputs in figure 2.15.

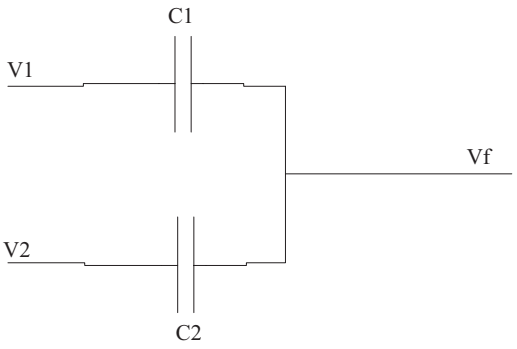

Figure 2.15: Capacitive voltage divider.

If voltages V_1, V_2 and V_f in figure 2.15 initially are zero, and V_1 and V_2 are applied afterwards, the voltage at node V_f becomes [KoGo01]

$$V_f = \frac{C_1 V_1 + C_2 V_2}{C_1 + C_2} \tag{2.14}$$

The voltage, V_f, is a linearly weighted sum of the input voltages. This is utilized. Parasitics can also be taken into account, in addition, as mentioned previously.

If equations from [YaAn93] are used, and there are n inputs, V_1, V_2,...,V_n, weighted by equally sized capacitances,C_n, the following equation results:

$$\begin{aligned} I_{ds,n} = I_0 exp\{\frac{\kappa C_n (V_1 + V_2 + ... + V_n)}{C'_{sum} U_t}\} \\ exp\{\frac{\kappa (C_{gs} V_{sb} + C_{gd} V_{db})}{C'_{sum} U_t}\} \\ (exp\{\frac{-V_{sb}}{U_t}\} - exp\{\frac{-V_{db}}{U_t}\} + \frac{V_{ds}}{V_0}) \end{aligned} \tag{2.15}$$

$$C'_{sum} = C_{sum} - \kappa C_{ox} = C_n + C_{gs} + C_{gd} + C_{gb} + C_{ox}(1 - \kappa) \tag{2.16}$$

I_0, the preexponential constant, is proportional to the W/L-ratio of the transistor. Some of the above terms are not included in the floating-gate

MOSFET model in [YaAn93]. When this fact is taken into account the equation can be rewritten, for source and substrate connected together:

$$I_{ds,n} = I_0 exp\{\frac{\kappa C_n(V_1 + V_2 + ... + V_n)}{C'_{sum}U_t}\}exp\{\frac{\kappa C_{gd}V_{ds}}{C'_{sum}U_t}\}(1 + \frac{V_{ds}}{V_0}) \quad (2.17)$$

g_m in saturation, for the multiple-input floating-gate MOSFET, is given by [YaAn93]:

$$g_m = \frac{\kappa' I_{ds}}{U_t} \quad (2.18)$$

The effective gate effeciency [YaAn93] of a floating-gate MOSFET is

$$\kappa' = \frac{\kappa C_n}{C'_{sum}} \quad (2.19)$$

, and the output conductance, g_{ds}, in saturation for the multiple-input floating-gate MOSFET is given by [YaAn93]:

$$g_{ds} \simeq \frac{I_{ds}}{V_o} + \frac{\kappa'' I_{ds}}{U_t} \quad (2.20)$$

Here

$$\kappa'' = \frac{\kappa C_{gd}}{C'_{sum}}. \quad (2.21)$$

The two components of the output conductance, g_{ds}, can be split into g_{ds1} and g_{ds2}:

$$g_{ds1} = \frac{I_{ds}}{V_o} \quad (2.22)$$

$$g_{ds2} = \frac{\kappa C_{gd} I_{ds}}{C_{sum'} U_t} \quad (2.23)$$

The conductances g_{d1} and g_{d2} are not linear, since I_{ds} increases exponentially with V_{ds}. The g_{d2} conductance is due to the capacitive coupling between the floating gate and the drain, and is usually much larger than g_{d1}, and the greater part the output conductance [YaAn93]. V_0, the Early voltage, is proportional to the channel length [YaAn93]. Therefore, an

increase in the gate langth of the device can decrease the g_{ds1} part of the output conductance, and increase the output resistance simultaneously.

If the goal is to reduce the output conductance while increasing the transconductance, the coupling capacitance(s) between the input and the floating gate should be increased. This increases the ratio of C_n/C_{sum} while decreasing the ratio C_{gd}/C_{sum}.

If the saturation region is considered, we might find the gain in common-source mode: A=g_m/g_{dsat} [AnBo91]. Then

$$A = \frac{\frac{\kappa C_n I_{ds}}{C_{sum'} U_t}}{\frac{I_{ds}}{V_o} + \frac{\kappa C_{gd} I_{ds}}{C_{sum'} U_t}} = \frac{\kappa C_n V_0}{C_{sum'} U_t + \kappa C_{gd} V_0} \tag{2.24}$$

According to this simple 1st order model, the voltage gain does not depend on the drain current level. Experience with simulations as well as practical measurements contradict this. Since there is a continuous transistion between the regions where the FGUVMOS circuits operate, the inverse slope factor κ (=1/n) will decrease in the transition between weak inversion and moderate inversion. Therefore a dependency on the current level might be observed. Some simulations illustrating this can be found in figure 2.20. For the current range where we want our circuits to operate, the dependency of voltage gain from input to output is rather limited compared to sizing of capacitances between the input(s) and the floating gate(s).

The impacts on important parameters, like g_m and r_{out} for floating-gate transistors, from other dimensions, are depicted in the table in figure 2.16. It can be read like, for example: To increase r_{out} one could decrease C_{gd}. Also, an increase in the dimension of C_n will increase both g_m and r_{out}.

	C_n	C'_{sum}	C_{gd}
$g_m \uparrow$	$\Uparrow$	$\Downarrow$	$\Downarrow$
$g_{ds} \downarrow$	$\Downarrow$	$\Downarrow$	$\Uparrow$
$r_{out} \uparrow$	$\Uparrow$	$\Uparrow$	$\Downarrow$
A $\uparrow$	$\Uparrow$	$\Downarrow$	$\Downarrow$

Figure 2.16: Parameters such as transconductance and output resistance for a floating-gate transistor as a function of some other parameters are shown.

2.2 UV-programmable inverters

For a simple analysis, of behavior of digital FGUVMOS circuits, less complicated equations than the current-voltage relationships for multiple-input floating-gate transistors given in [YaAn93] are used. From the form of the equations in [BeWi99] we can use the following for the drain currents of PMOS and NMOS transistors:

$$I_{ds,p} = I_{beq} \prod_{j=1}^{l} exp\{\frac{1}{nU_t}(V_{dd}/2 - V_j)k_j\} \tag{2.25}$$

$$I_{ds,n} = I_{beq} \prod_{i=1}^{m} exp\{\frac{1}{nU_t}(V_i - V_{dd}/2)k_i\} \tag{2.26}$$

It is assumed that the sum of capacitances between the input and the floating gate of the PMOS equals the sum of capacitances between the input and the floating gate of the NMOS, and that the total capacitance seen from the floating gate of the PMOS equals the total capacitance seen from the floating gate of the NMOS. Then $k_j = k_i$, and $k_i = C_n/C_{tot}$.

For simplicity, the assumption is made that both the PMOS and the NMOS transistors have the same intrinsic slope factors, $n = 1/\kappa$. According to the equations the PMOS has a number of l capacitively coupled inputs to the floating gate, while this number is m for the NMOS.
If all inputs to the universal FGUVMOS element (figure 2.17) are short circuited the element ends up as a basic inverter. In that case, and assuming that practically 100% of the input signal gets through to the floating-gates, the equations may be rewritten:

$$I_{ds,p} = I_{beq} exp\{\frac{1}{nU_t}(V_{dd}/2 - V_{in})\} \tag{2.27}$$

$$I_{ds,n} = I_{beq} exp\{\frac{1}{nU_t}(V_{in} - V_{dd}/2)\} \tag{2.28}$$

The equilibrium current I_{beq}, for a certain produced circuit adjustable by UV-programming. In the above equations the only parts of the equations differing as a function of applied voltages are the parts of the exponents containing V_{dd} and V_{in}. Here they are expressed:

$$e_p = (V_{dd}/2 - V_{in}) \tag{2.29}$$

$$e_n = (V_{in} - V_{dd}/2) \tag{2.30}$$

If we choose to follow a UV-programming procedure letting all driven nodes stabilize at $V_{dd}/2$, like in [BeWi97], $e_p = e_n = 0$, then the circuit is at the switching point, with maximum currents, I_{beq}, through both transistors:

$$I_{ds,p} = I_{beq} \tag{2.31}$$

$$I_{ds,n} = I_{beq} \tag{2.32}$$

For a given V_{dd} the changes in e_p and e_n change the level of the output voltage. When the size of the capacitances between the actual input(s) and the floating gate(s), relative to the total capacitance seen from the floating gate, is large enough, the voltage perturbation on the floating gate is sufficient for proper operation. The drain currents for the PMOS and NMOS transistors, as well as their output resistances, then change. The drain currents are reduced when the circuit leaves the equilibrium state. When the inverter has reached a steady state with the input constantly at V_{dd} or V_{ss}, the output resistances, of the PMOS relative to the NMOS, determined if the output moved towards V_{dd} / "high" / "1"-level, or V_{ss} / "low" / "0"-level.

A simulation of an FGUVMOS inverter is shown in figure 2.18, and a truth table in figure 2.19. The behavior of inverters can be extended to explain the behavior of more complex gates as NAND, NOR or XOR, which in turn might be used for modules like multipliers and processors [Raba96]. The characteristics in figure 2.18 resemble the voltage-transfer-characteristic, sometimes called the DC transfer characteristic, which plots the $V_{out} = f(V_{in})$. Actually it is a transient simulation where the inverter switches at a relatively low speed so that if the horizontal time axis is changed with that of a corresponding input voltage the characteristics approach a DC characteristic for the purposes here. The ideal DC characteristic in this case (figure 2.18) would have $V_{out}/V_{in} = -\infty$ [Raba96] and $V_{in} = V_{out} \Leftrightarrow V_{in} = V_{dd}/2$.

Increasing the size of the drawn capacitances between the input and the floating gates increases the amount of the input signal getting through to

the floating gates, as illustrated in figures 2.7, 2.13 and 2.16, and thereby the voltage gain. The bigger the "0" and "1" intervals, the better the noise margin, as defined in [Raba96] p. 112. A digital gate also needs to have a transient region where the gain is greater than 1, in absolute value, to be regenerative [Raba96].

The voltage might degree on the current level, to some extent, as can be seen in figure 2.20. The curves were simulated for equilibrium currents of 1 nA, 10 nA and 100 nA, on a netlist extracted from layout. Dynamic behavior, such as maximum operating speed of the gates also depends on the sizing of the capacitances. For an FGUVMOS gate implemented in CMOS, some relevant parasitic capacitances do not change much with the current level, in the 1 to 100 nA range, an attractive I_{beq} range for our purposes, as shown in figure 2.11.

A switching current which might be (re-)adjusted by UV-programming over several orders of magnitude will have a great impact on dynamic properties like switching speed and power dissipation. The output of each FGUVMOS element is, for practical purposes, regarded as driving a purely capacitive load. The capacitances store charge, and the currents are the rate of change of charge. The higher the current levels, as a function of I_{beq}, the faster the operating speeds of the circuits.

A more general equation is:

$$\frac{dV}{dt} = \frac{I}{C} \tag{2.33}$$

If the capacitance C is treated like a constant, the value of the current I determines the maximum change of the voltage per time unit.

Another variant of the inverter is the "analog inverter" [ShKo93], which has also been implemented using the FGUVMOS technique [BeNa99a], [BeNa99b], [NaBe99]. A schematic is shown in figure 2.21. Ideally, the analog inverter converts the input voltage to the voltage of V_{dd} minus the input voltage. An analog inverter has been simulated in figure 2.22. This building block has been used in an analog multiplier [BeNa99a] and different transconductance amplifiers [NaBe99], [BeLa01a].

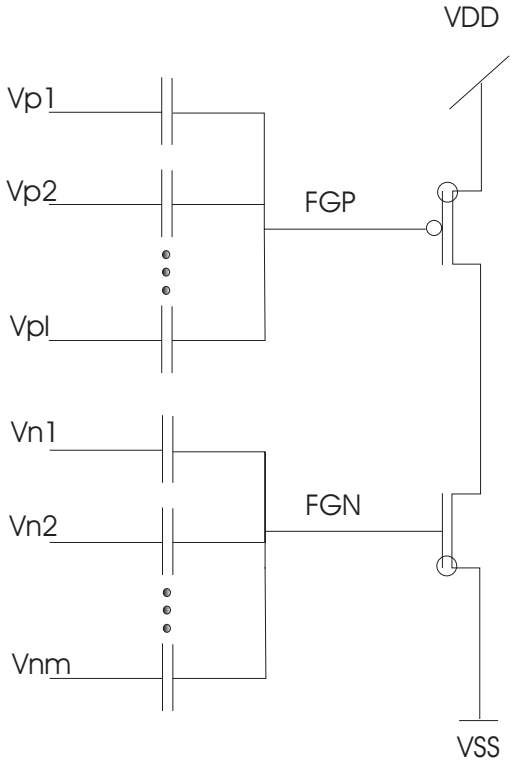

Figure 2.17: Schematic for universal FGUVMOS element.

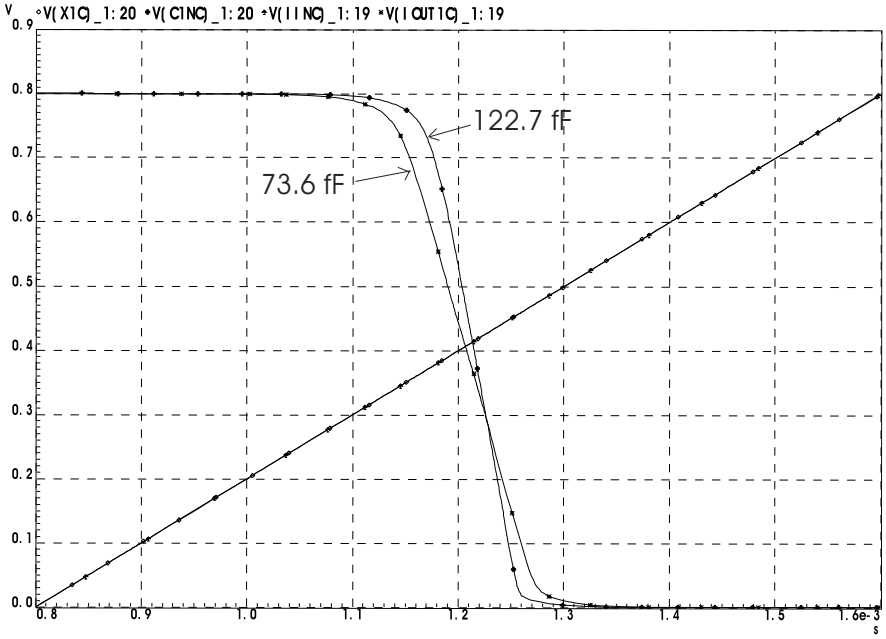

Figure 2.18: Output voltages are simulated for an increasing input voltage, for two different drawn capacitances between input and floating gates.

IN	e_p	e_n	OUT
0	$V_{dd}/2$	-$V_{dd}/2$	1
1	$-V_{dd}/2$	$V_{dd}/2$	0

Figure 2.19: Truth table for FGUVMOS inverter.

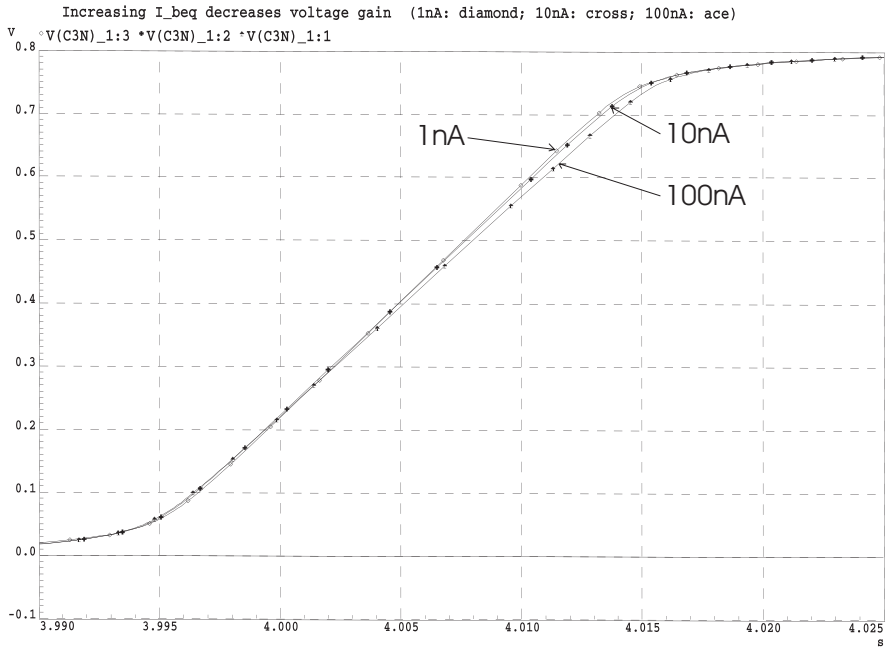

Figure 2.20: Output voltages, for a decreasing input voltage (not shown) are simulated for an inverter at different equilibrium current levels. The lower the current level, the steeper the slope, and the higher the voltage gain. Input voltage shrank linearly from 0.8 to 0 V.

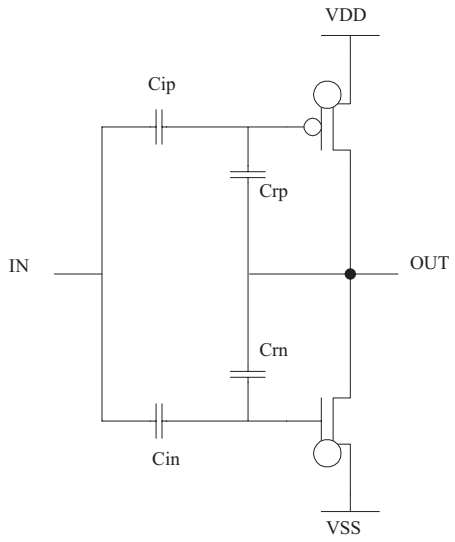

Figure 2.21: Analog inverter. $C_{ip} = C_{rp} = C_{in} = C_{rn}$. Additional inputs may be used.

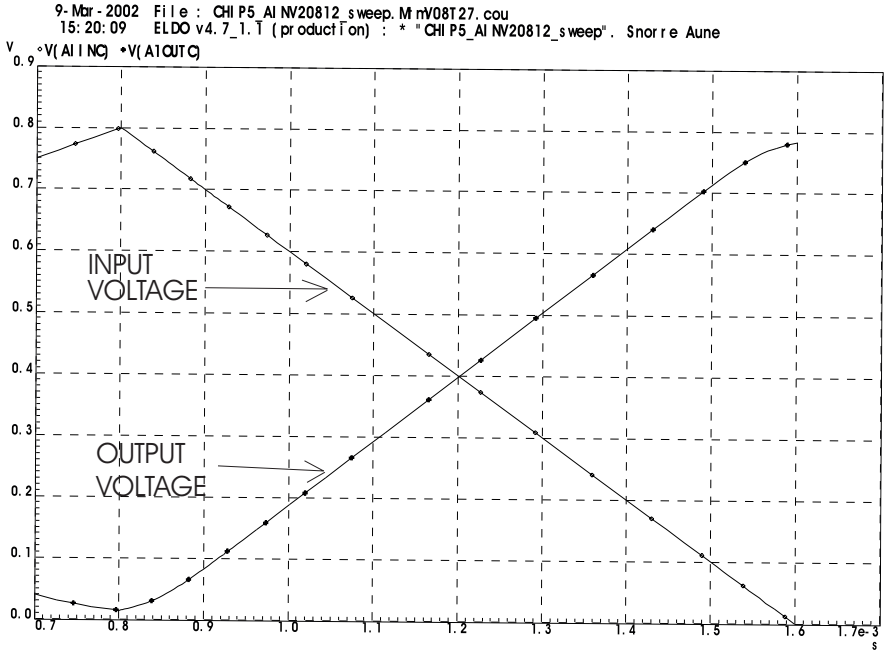

Figure 2.22: An analog inverter is simulated. The output voltage, V(A1OUTC) is ideally V_{dd} minus the input voltage, V(AIINC). (Netlist: *CHIP5_capall_netlist* [Aune02])

2.3 A 2-MOSFET 3-input reconfigurable "single-ended" circuit

To illustrate some traits of the reconfigurable floating-gate circuits, the "P1N3" circuit in figure 2.23, from [AuBe01b] is used here. The capacitances between X,Y and Z are all designed for equal size. Also the sum of capacitances coupled to the NMOS (figure 2.23) equals the capacitances connected to the PMOS. ($C_{x2} + C_{y2} + C_{z2} = C_{y1}$). That is common for all new circuits presented here, unless otherwise mentioned. The numbers of capacitively weighted inputs have been used to name the circuits, counting ordinary inputs and other inputs used for control of behavior to each two-MOSFET element. The circuit in figure 2.23 gets the name P1N3 due to this naming system. The most used inverter may be called P1N1.

The circuit has previously been presented as a stand-alone circuit or building block [AuBe01b], [AuBe01c], [AuBe01a]. Basic assumptions are similar as for the previous inverter analysis.

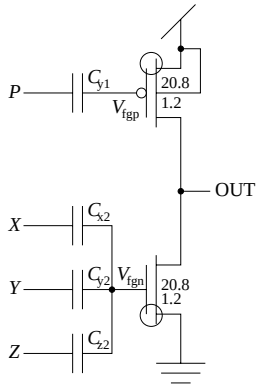

Figure 2.23: A CARRY'-,NAND-,NOR and INVERT-circuit [AuBe01b].

Using a similar approach as was used for the inverter gives:

$$I_{ds,p} = I_{beq} exp\{(\frac{1}{nU_t}(\frac{V_{dd}}{2} - V_p)\} \tag{2.34}$$

$$I_{ds,n} = I_{beq} exp\{\frac{1}{nU_t}(\frac{1}{3}V_x + \frac{1}{3}V_y + \frac{1}{3}V_z - \frac{V_{dd}}{2})\} \tag{2.35}$$

V_p, for example, means the voltage on the capacitively weighted input to the PMOS transistor in figure 2.23. If the voltages V_p, V_x, V_y or V_z

V_{dd}	1	HIGH
V_{ss}	0	LOW

Figure 2.24: Meanings of the 3 columns in each row are treated as synonymous

X	Y	Z	e_p	e_n	OUT
0	0	0	0	-$3V_{dd}/6$	1
0	0	1	0	-$V_{dd}/6$	1
0	1	0	0	-$V_{dd}/6$	1
0	1	1	0	$V_{dd}/6$	0
1	0	0	0	-$V_{dd}/6$	1
1	0	1	0	$V_{dd}/6$	0
1	1	0	0	$V_{dd}/6$	0
1	1	1	0	$3V_{dd}/6$	0

Figure 2.25: The table shows parts of the exponentials, e_p, e_n, and output values, for all possible binary values of inputs X,Y,Z when V_p=$V_{dd}/2$. "OUT" provides the CARRY' function for a FULL-ADDER.

are all equal to $V_{dd}/2$ the exponentials in the above equations equal 0, and we have the equilibrium condition with $I_{ds,p}$=$I_{ds,n}$=I_{beq}. To see how the circuit functions logically, a truth table is used here. As part of the truth table, e_p and e_n are used; these are the parts of the exponentials directly dependent on input signals or the supply voltage, V_{dd}. For this particular circuit that means

$$e_p = (\frac{V_{dd}}{2} - V_p) \tag{2.36}$$

$$e_n = (\frac{1}{3}V_x + \frac{1}{3}V_y + \frac{1}{3}V_z - \frac{V_{dd}}{2}). \tag{2.37}$$

When parasitic capacitances are not accounted for, each of the inputs at nodes X,Y and Z are weighted by 1/3, a more optimistic estimate than would have resulted from including parasitics. When $V_p = V_{dd}/2$ and X, Y and Z are allowed to have the binary values according to the table in figure 2.24, figure 2.25 is the result. When $e_p > e_n$ the output approaches the V_{dd} level. In the opposite case it goes low.

By inspecting the truth table (figure 2.25) it is clear that the value of the output depends on the number of 1's and 0's on the inputs only. Inspecting the truth table (figure 2.25), just counting 1's and 0's in the input vector, makes it possible to get enough information out from a table with a simpler form, as in figure 2.26.

P	number of "1"'s	e_p	e_n	OUT
$V_{dd}/2$	0	0	-$3V_{dd}/6$	1
$V_{dd}/2$	1	0	-$V_{dd}/6$	1
$V_{dd}/2$	2	0	$V_{dd}/6$	0
$V_{dd}/2$	3	0	$3V_{dd}/6$	0

Figure 2.26: The table shows parts of the exponentials, e_p, e_n, and output values, for $V_P = V_{dd}/2$ and different numbers of "1's" on ordinary inputs X,Y,Z. Inputs can be either "0" or "1".

Used this way (figure 2.25, figure 2.26) the circuit computes the inverted carry for a FULL-ADDER:

$$OUT = CARRY' = (XY + XZ + YZ)' \tag{2.38}$$

From the truth table, in figure 2.25, one can see that by letting any one input be "0", the output is "0" if, and only if, both other inputs are "1". Then the circuit implements the 2-input NAND function. If any one input is 1, the output is "1" if and only if both other inputs are "0". Then the circuit works as a 2-input NOR gate. Connecting one input to Vdd or Vss and short-circuiting the other two gives an INVERTER. A 2-input inverting-structure like NAND or NOR is essentially the only function needed to implement any digital function. In figure 2.27 the equilibrium state is when e_p and e_n both are 0, for $V_{dd}/2$ on all inputs. If, and only if, the number of 1's in the input vector [X,Y,Z] is 2 or more, $e_n > e_p$, which forces the output low.

If the e_p values is changed, the "threshold" for when, and if, $e_n > e_p$ changes. In the ideal case depicted in figure 2.27, setting $e_p = -2V_{dd}/6$ makes it necessary to have only 1 binary input at "1" (V_{dd}) to make $e_n = -V_{dd}/6$, which is greater than e_p, and should give a low output. Changing the e_p value to $2V_{dd}/6$ makes it necessary to have all inputs X,Y,Z high, when restricted to Boolean inputs, in order to produce a low output. When perceiving the circuit as digital it is possible to make a table like in

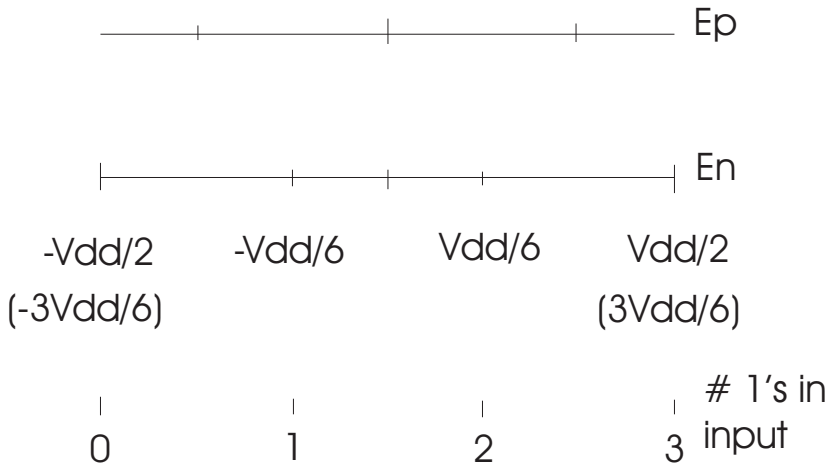

Figure 2.27: An illustration of e_n as a function of the number of 1's in the inputs X,Y,Z for the P1N5 circuit. ($E_p = e_p$, $E_n = e_n$).

e_p	digital functionality
$-2V_{dd}/6$	NOR3, NOR2, INVERT
0	CARRY', NOR2, NAND2, INVERT
$2V_{dd}/6$	NAND3, NAND2, INVERT

Figure 2.28: The table shows part of the exponentials, e_p, e_n, and output values, for $V_P = V_{dd}/2$ and different numbers of "1's" on ordinary inputs X,Y,Z. Inputs can be either "0" or "1".

figure 2.28. Changing the threshold by adjusting the value of the input capacitively coupled to the PMOS can provide a real-time reconfigurable digital circuit.

The inherent functionality for $e_p = 0$, demonstrated by simulation, can be seen in figure 2.29. The voltage on the output goes low if, and only if, 2 or 3 of the inputs X,Y,Z goes high at the same time. Else the output stays high, which in this case is $V_{dd} = 0.3$ V.

The functionality and performance of such circuits will depend on the implementation technology of choice. The number of inputs, their capacitive weights, if an input is coupled both to the PMOS and NMOS are among other factors that can be varied, as seen in, for example [BeWi99], [AuBe01a].

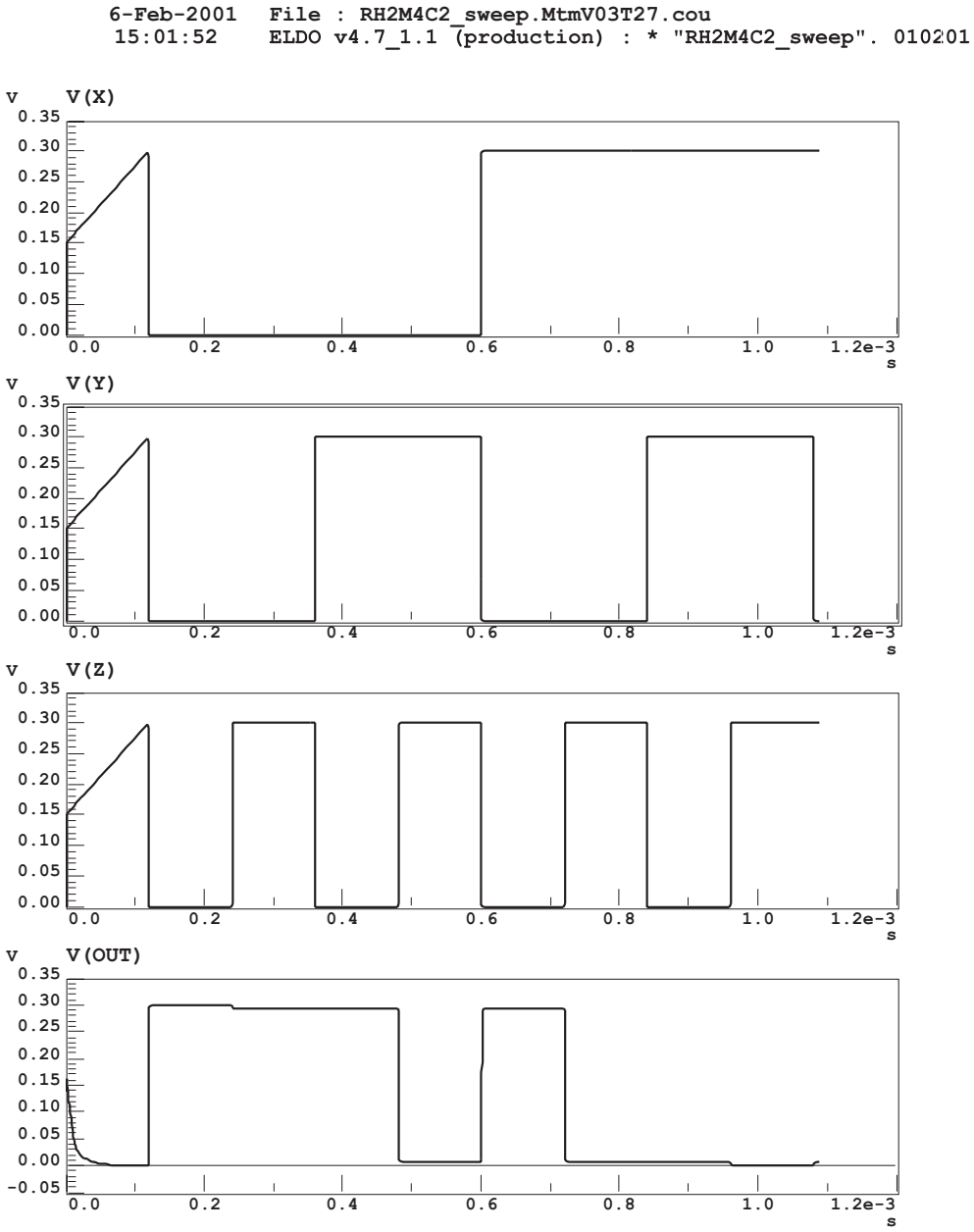

Figure 2.29: The CARRY' function of P1N3 (figure 2.23) is simulated. The curves represent the voltages on X,Y,Z and OUT. $e_p = 0$. If V(X)=0, V(Y) NAND V(Z) is computed. If, and only if, one of the inputs X,Y,Z is held at V_{dd}, the circuit implements the 2-input NOR function. (V(X)=V_x)

Chapter 3

Floating-Gate UV-programmable MOSFETs

3.1 Implementation and layout of FGUVMOS transistors

The first chip contained several different "donut" NMOS and PMOS transistors, laid out in two blocks called "$nmos - rekke$" and "$pmos - rekke$", as in figure 3.1, and dimensioned according to figure 3.4. The chip in AMS 0.6 CMOS [AMS98] was submitted through the EuroPractice MPW service in July 2000. A small plot can be seen in figure 3.3. 10 bonded chips in a ceramic 64 pin package with taped lid were received 3-4 months later. The purposes were to make basic building blocks for use on later chips, as well as to produce experimental data.

The reason for choosing the donut shape was to get a relatively low gate-drain parasitic capacitance (Drain is in the center, and source on the outside of the "donut" gate). A transistor of W=10.4μm/L=0.6μm was the minimum transistor of this kind that could be made without breaking layout rules in [AMS9X]. Capacitors were made in poly1 and poly2 layers, and the dependence between the drawn square poly2 layer size and estimated capacitance values is depicted in figure 3.7.

NMOS transistors, N1, N2,...,N9 shared a common control gate coupled from an I/O-pad to the drawn poly1-poly2 capacitor, where the poly 1 plate forms the floating gate. N1 to N5 had coupling capacitors with an estimated value growing linearly from 10.2 fF to 53 fF. N6, N7, N8 and

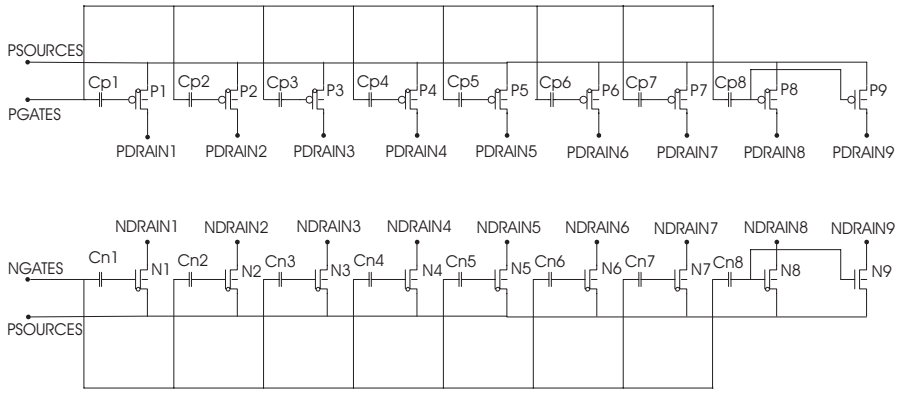

Figure 3.1: PMOS and NMOS transistors. All except P9 and M9 are directly UV-programmable.

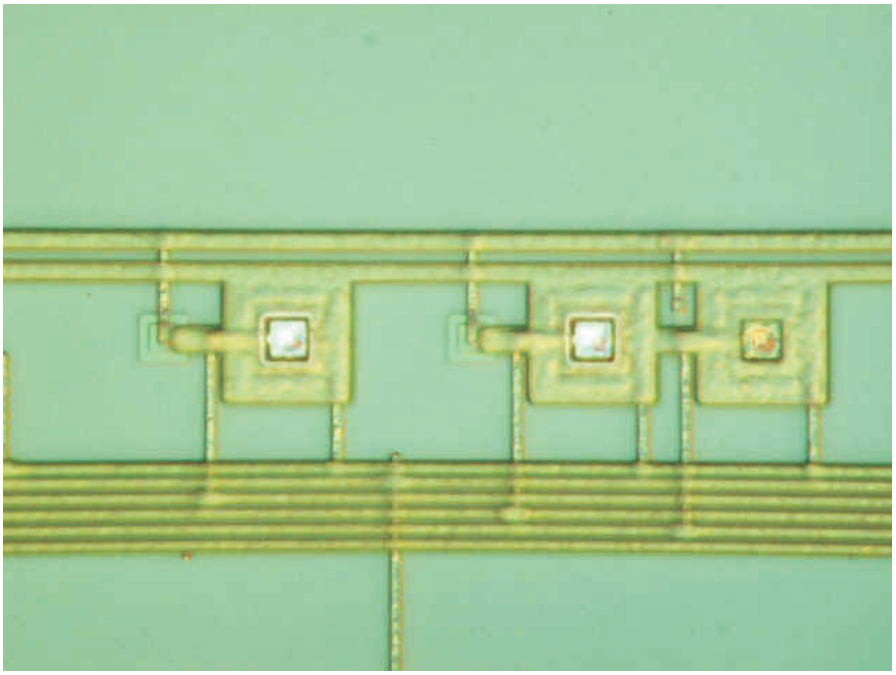

Figure 3.2: 3 transistors are shown. The rightmost one lacks a "UV-hole" in the passivation W/L=20.8μm / 1.2μm.

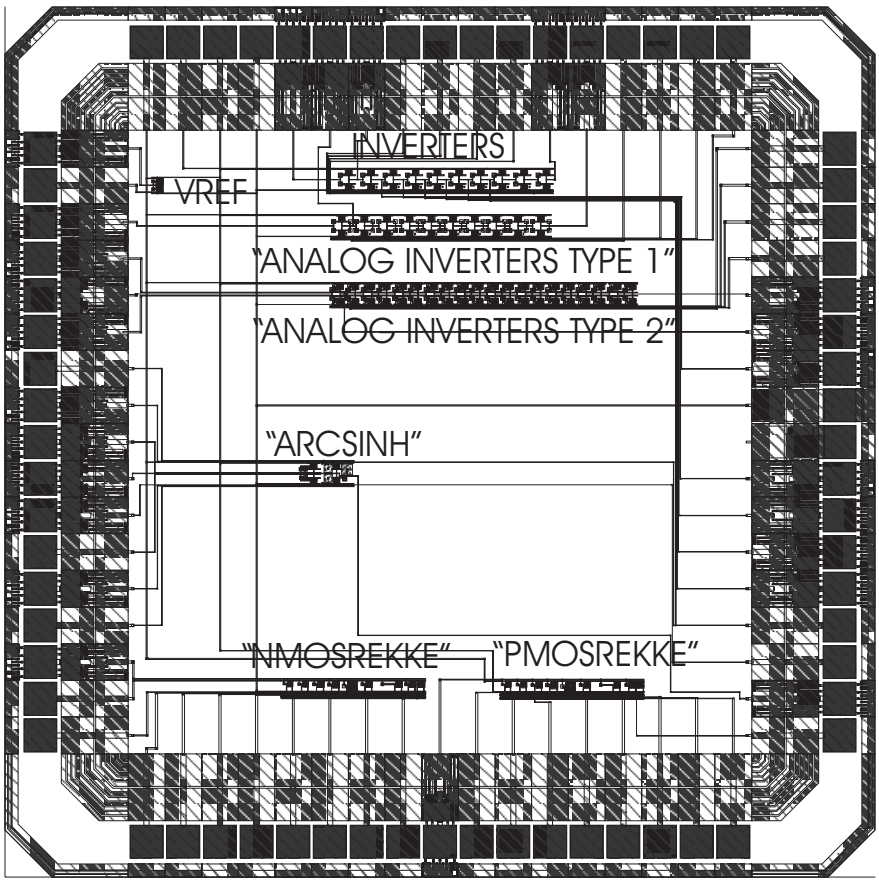

Figure 3.3: Picture of the first 68 pin chip. Pads have been numbered counterclockwise, starting with pad 1 in the middle of the left side.

device	P1	P2	P3	P4	P5	P6	P7	P8	P9
W[μm]	10.4	15.6	20.8	26	52	20.8	20.8	20.8	20.8
L[μm]	0.6	0.9	1.2	1.5	3.0	1.2	1.2	1.2	1.2
C [fF]	10.2	15.5	20.2	25.5	53.0	10.2	10.2	10.2	10.2
Drain to pin no.	19	20	21	22	23	24	25	26	27
device	N1	N2	N3	N4	N5	N6	N7	N8	N9
W[μm]	10.4	15.6	20.8	26	52	20.8	20.8	20.8	20.8
L[μm]	0.6	0.9	1.2	1.5	3.0	1.2	1.2	1.2	1.2
C [fF]	10.2	15.5	20.2	25.5	53.0	10.2	10.2	10.2	10.2
Drain to pin no.	9	10	11	12	13	14	15	16	17

Figure 3.4: Dimensions of PMOS, NMOS and capacitances.C_{p8} and C_{n8} are each shared among two MOSFETs.

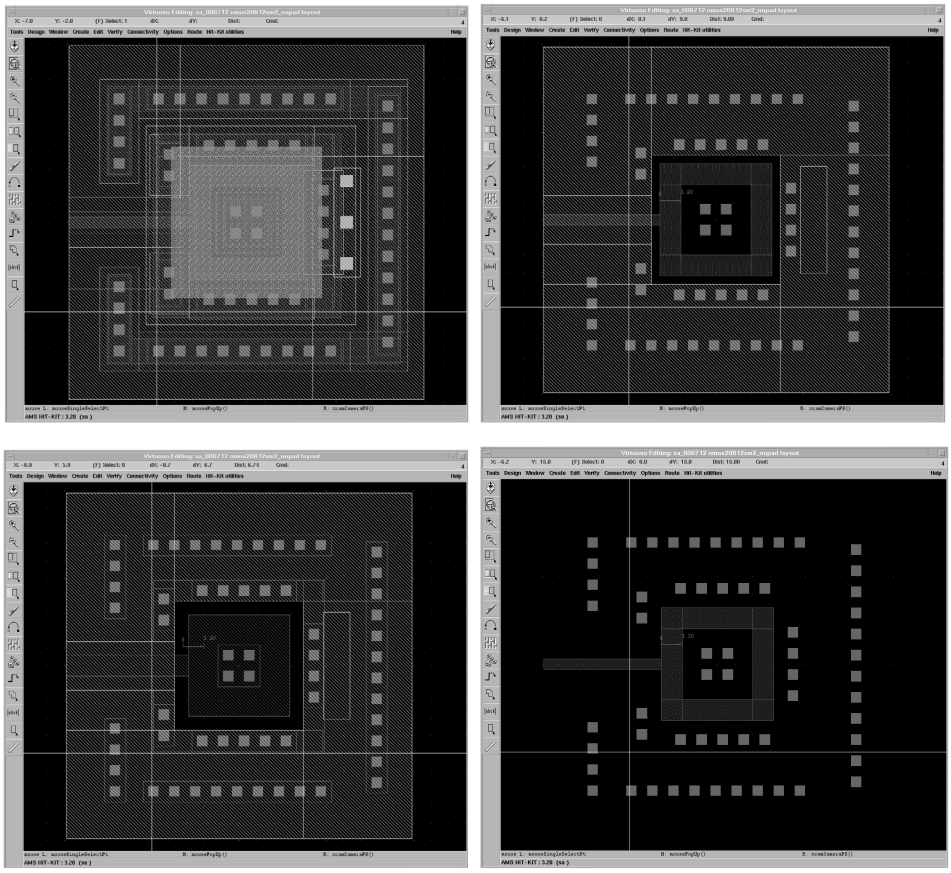

Figure 3.5: NMOS20812 layout, top view. Upper left (UL): All layers switched on. UR: metal 2, contact and poly 1. LL: metal 2, contact and metal 1. LR: contact and poly 1.

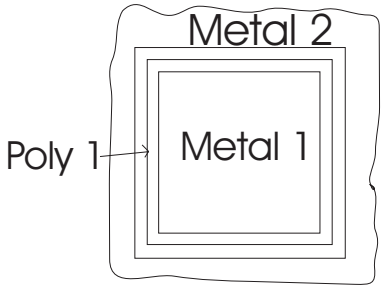

Figure 3.6: Seen from top all PMOS and NMOS transistors had a distance of 0.3 μm from the center, square, metal 1 layer to the outer edge of the poly 1 layer gate, and a distance of minimum 0.8 μm between the four sides of center metal 1 (drain node under regular use) to surrounding metal 2 "shield" (source).

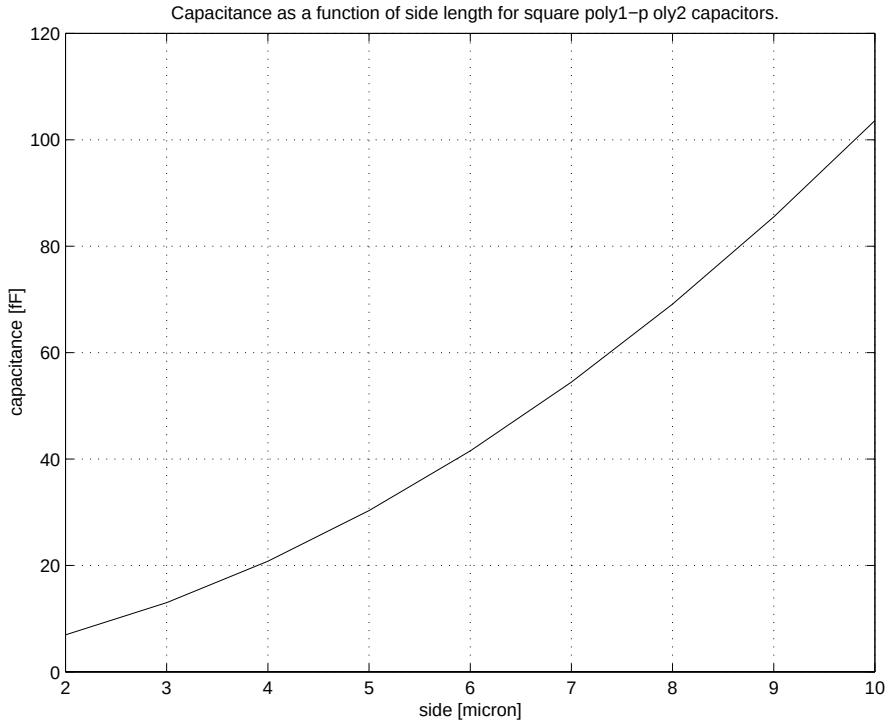

Figure 3.7: The estimated values of square poly1-poly2 capacitors can be found as a function of the side lengths in μm.

N9 had W/L=20.8μ/m1.2μm. N6 was identical to N3. N6 and N7 had identically drawn coupling capacitances of 20.22 fF, but the latter had a slightly bigger active gate area, and hence a slightly bigger total capacitance at its floating gate. N8 and N9 shared a drawn poly1-poly2 capacitor between a common floating gate and the control gate. N9 was identical to all other transistors with the exception that it was covered with passivation, and should therefore be only indirectly UV-programmable. Every transistor had its own drain terminal isolated from the others. The 9 PMOS transistors had a very similar layout, since it was made by taking a copy of the cluster of NMOS transistors and changing NMOS with PMOS but keeping the metal wiring and capacitance dimensions. Transistors with W/L=20.8μm/1.2μm have been used in several of the other circuits.

The layout of such a device showing all drawn layers is shown in the upper left corner of figure 3.5. The hole ("UV-hole") in the passivation layer is here the rather grey shaded area from the center of the picture reaching halfway through 18 of the contacts. In the upper left corner metal 2 is shown, enclosing the L=1.2μm poly 1 gate. The distance from the "donut part" of the gate to metal 2, seen from top, is 0.3 μm. In the lower left picture the metal 2, contact and metal 1 layers are shown, with the distance between the center square part of metal 1 and the surrounding metal 2 layer being 0.8 μm. The lower right part shows only contact- and poly 1- layers. Additional information regarding layout of transistors can be found in figure 3.6.

The dimensioning of the UV-holes was done based on earlier experience at the University of Oslo, as well as assumptions about what could be expected to work, though the design rules were broken. Firstly, we draw UV-holes of lesser size than recommended. Secondly, the Design Rule Check reported between 12000 and 15000 "faults" when the hole in the passivation layer was placed on top of transistors. Placement of UV-holes was therefore among the last changes done to the layout, so that the DRC (Design Rule Check) should not produce too many superfluos warnings.

A die photo shows three different W/L=20.8μm/1.2μm transistors (figure 3.2). The rightmost one differs from the two others because it has no opening in the pad layer on top. The two other transistors have openings, and metal 1 can be seen in the center. The center metal 1 is part of the drain node of the transistor. Outside this metal 1 there is the gap that lets the UV-light between metal layers, and then metal 2 surrounding.

3.2 Laboratory setup and UV-programming

The laboratory measurement setup for UV-programming of the transistors is shown in figures 3.8, 3.9 and 3.10. UV photoinjection of electrons through SiO_2 provides a simple method for programming analog, nonvolatile memories in CMOS circuits [BeKe93]. The first figure shows a simplified setup showing two examples from UV-programming experiments that were done. The applied voltages on the 4-terminal PMOS and NMOS transistors are shown, as well as the corresponding pin numbers of the chip, under UV-exposure. A "reverse-biased mode" was then used [BeLa97], [BeLa99a]. Voltages on what under normal use are the source terminals are different. The voltage on the source node of the NMOS is higher than on the gate, while it is lower than the gate voltage for the PMOS transistor [BeLa97]. The two latter schematics show further details concerning instrumentation for UV-programming and testing of the devices (figures 3.9 and 3.10). Measurement instruments shared ground / zero voltage with the chip and test PCB. Running the programming and test could all be done via the computer terminal after the initial setup.

The photograph in figure 3.11 shows a setup for UV-programming. The test PCB with the chip and the UV-lamp on top can be seen in the lower center part of the picture. The 68 pin chip, in a ceramic package, was mounted in a socket on the test PCB, as shown in 3.12. The package had a removable lid on top, that was taken off for UV-programming. The substrate of the chip was coupled to common ground reference via a wire from pad number 58. During UV-programming the "reverse-mode" was used. A consequence of this was a higher voltage level on what is the source compared to the drain of the NMOS transistors under normal operation of the circuits while under UV-exposure. A similar scheme was used for the PMOS transistors. The Keithley 236 Source-Measure unit was used to apply a voltage of 0.4 Volts for UV-programming, connected to one of the drain terminals. What is normally the source was driven to 0.8, 0.9 or 1.0 V for our experiments, by channel 2 from the Keithley 213 Quad Voltage Source. The Thurlby Thandar Power Supply was connected to pin. no 35, the positive power pad, with a voltage of 4.5 V, and 0 V to pin number 36, the negative / VSS power pad. The HP33120 signal generator was connected to the common control gate of the transistor for measurement purposes while the transistors were in normal "computational mode".

The Matlab script "ProgUVmin.m" run on a Sun Sparc computer was used to control the measurement setup, with chosen parameters in each case. Matlab scripts are shown in Appendix. During the UV-programming

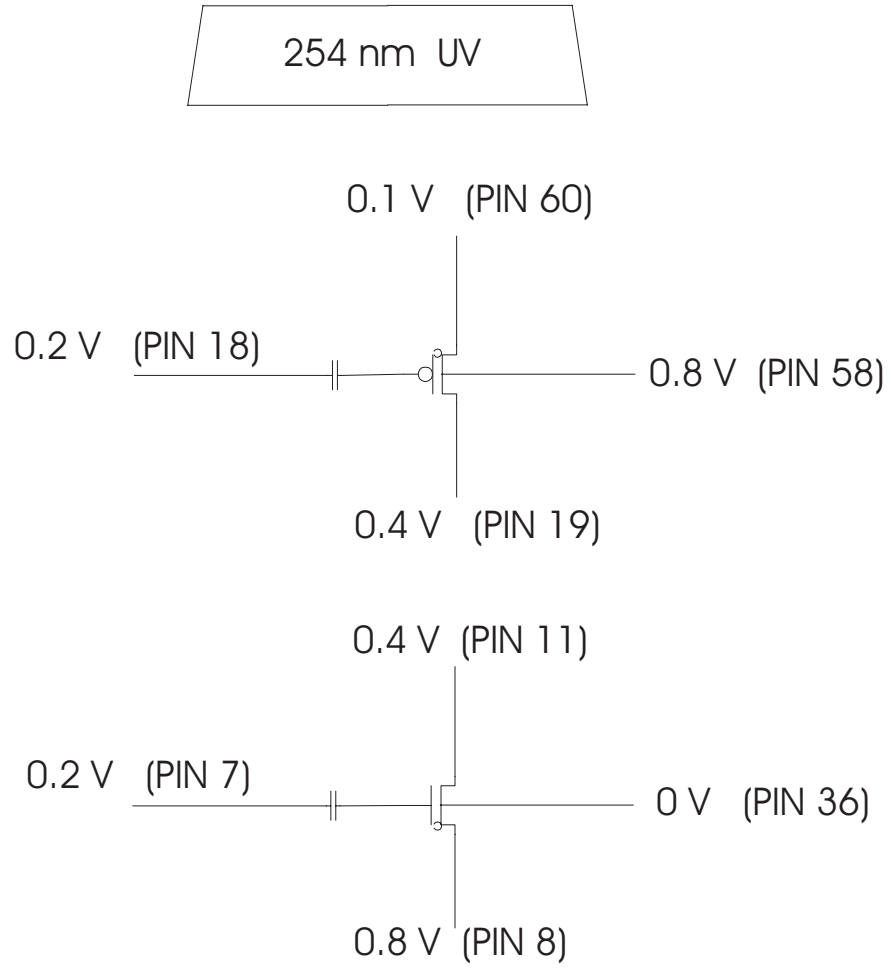

Figure 3.8: Voltages shown are applied under UV-programming of PMOS and NMOS devices. The circle between gate and what in standard computing mode is the source indicates where the wanted UV-activated conductances appear. The voltages on PIN 8 and PIN 60 are the voltages called "programming voltages" in this chapter.

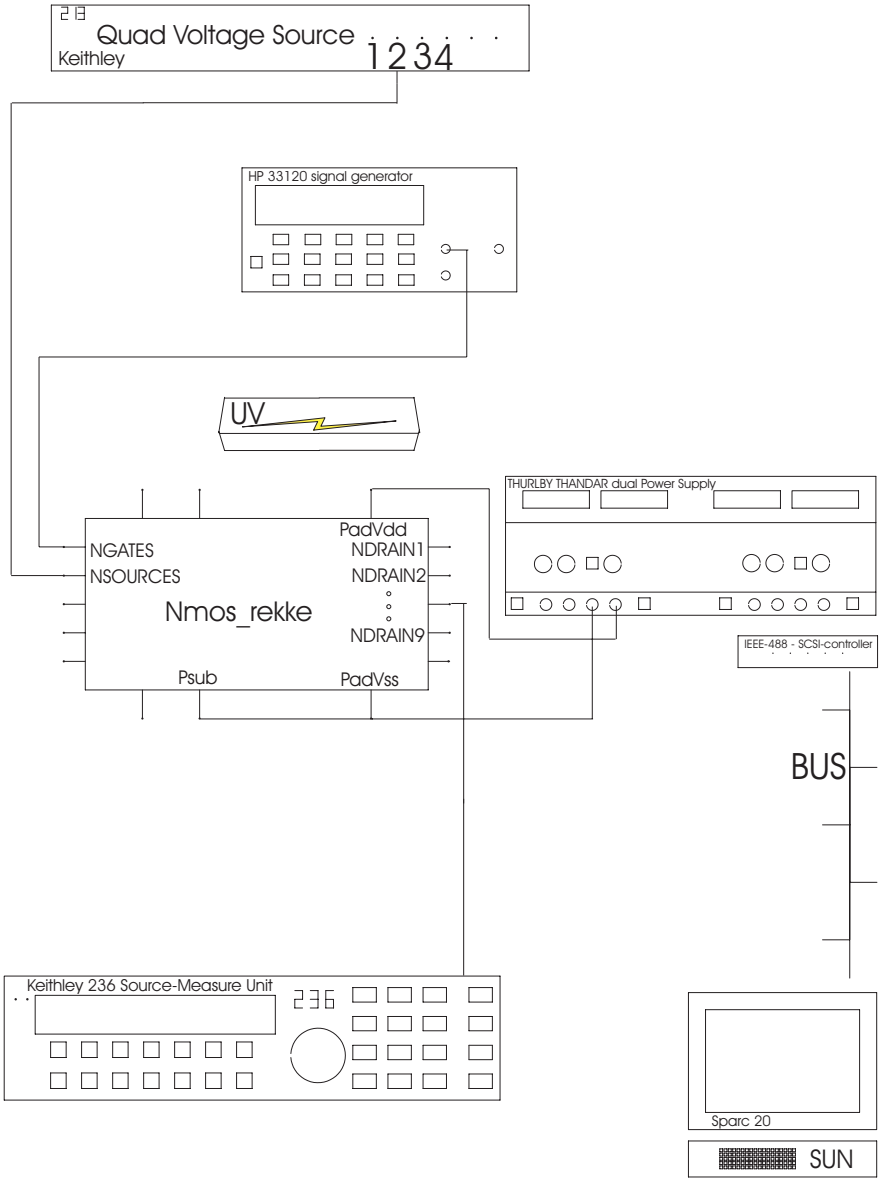

Figure 3.9: Laboratory setup for doing UV-programming of NMOS transistors is drawn.

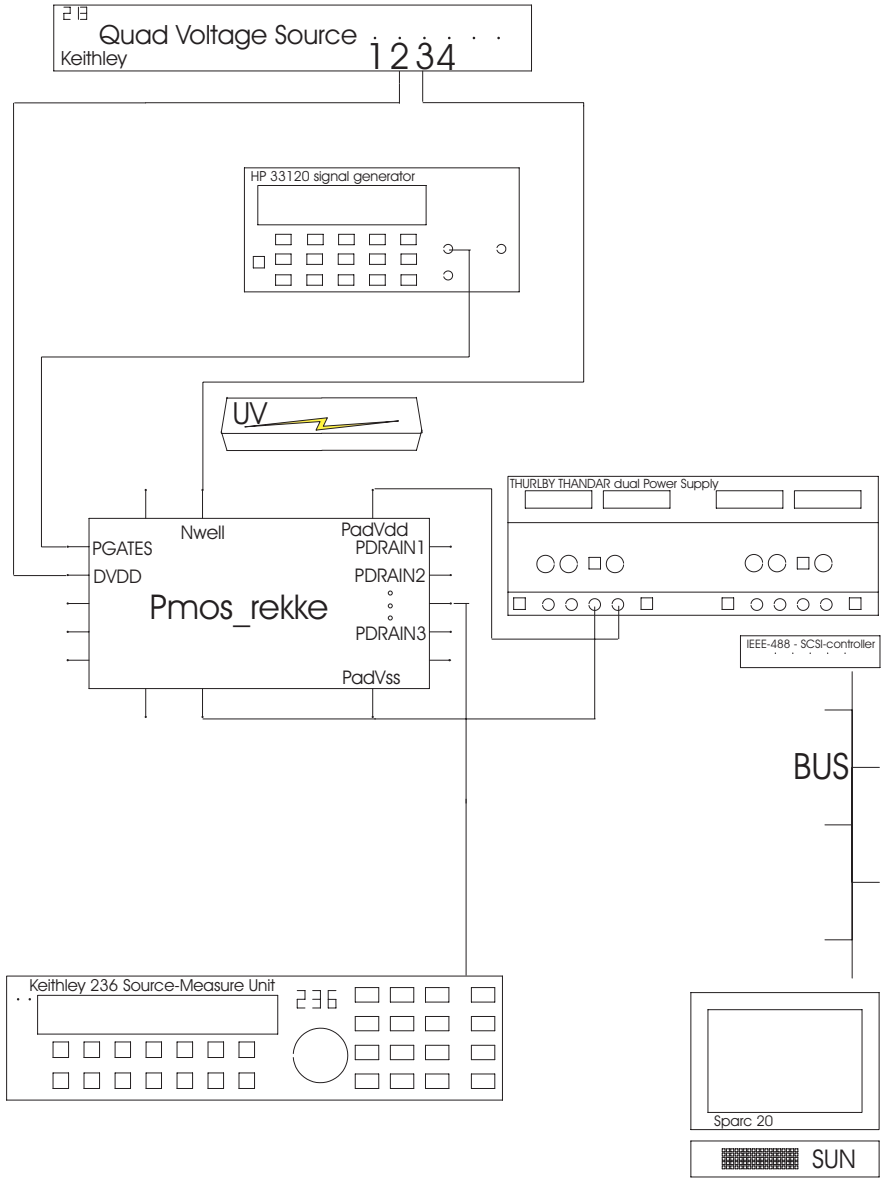

Figure 3.10: Laboratory setup for doing UV-programming of PMOS transistors is sketched.

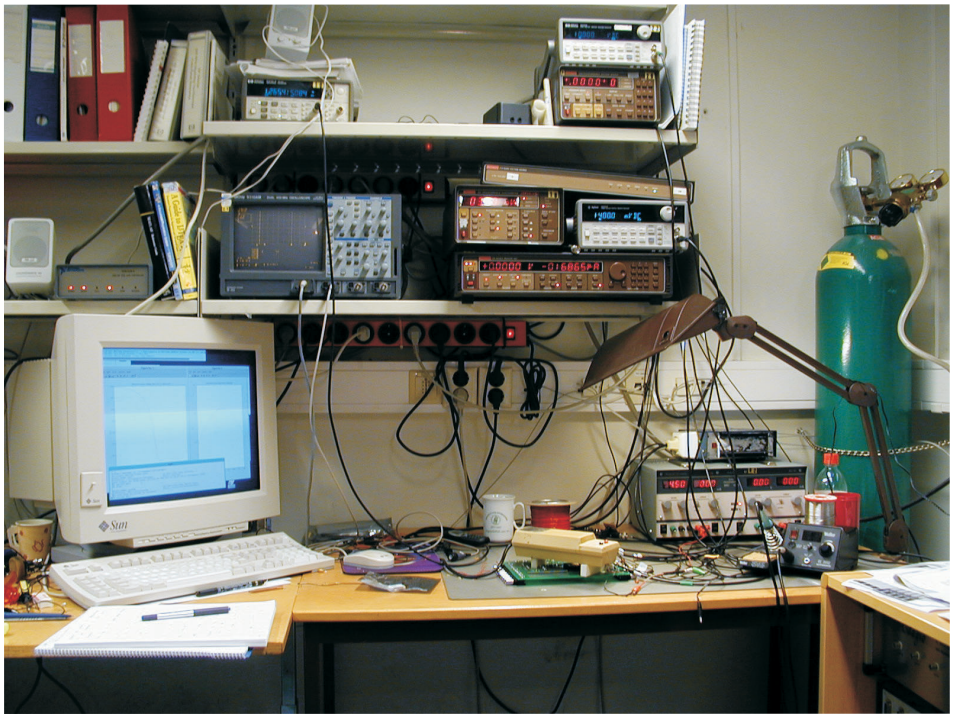

Figure 3.11: VLSI laboratory at the Department of Informatics, University of Oslo.

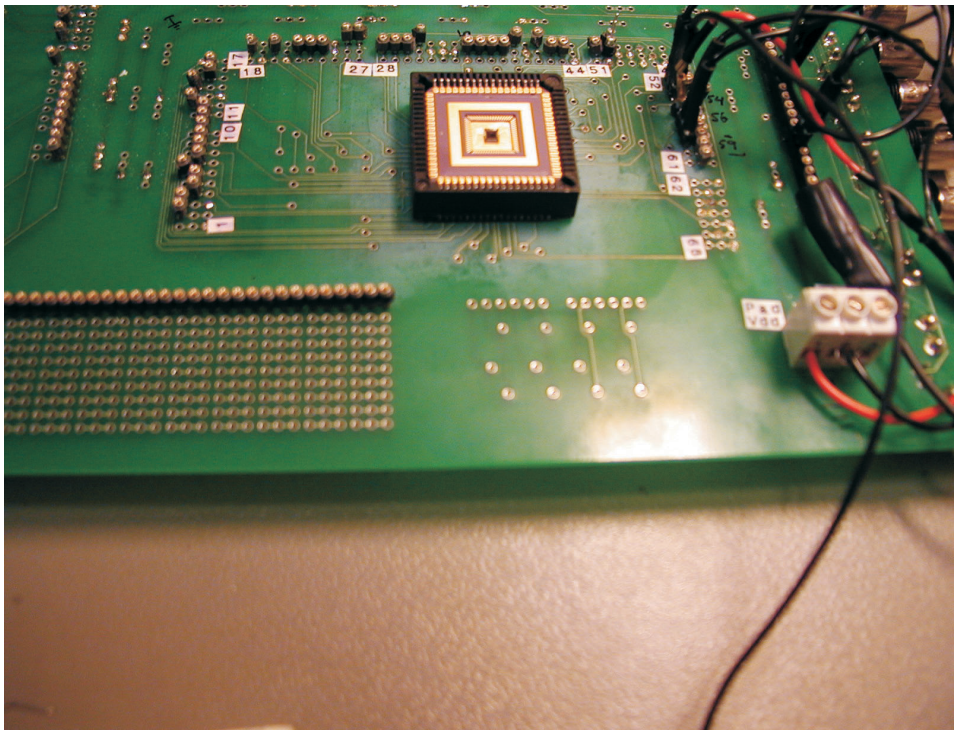

Figure 3.12: First chip at laboratory.

conditions of the measurement instruments were observed via plots on the computer screen. Scripts were also used to save chosen measurement data such as stimulus to devices under test, measured currents and voltages and time periods of UV-exposure of the chip. For testing the different transistors, the UV-programming period was run for less than an hour, just to check that a change in current levels due to UV exposure did occur. Such adaption may take up to the order of 24 hours, according to results published in [BeWi97].

To measure characteristics like drain current dependency on input voltages another piece of Matlab code "InvSweep.m" was used. The connections between the instrument measuring the drain currents and the test PCB were changed for each measurement, as the transistors had their drains connected to individual I/O-cells (pads) on the chip. Source and bulk nodes were set to 0 V, and the drain voltage to $V_{dd}/2$ using the Keithley 236 instrument. The HP33120 signal generator swept the control voltage between "Vinstart" and "Vinstop" values, in a number of steps given by "Vinstep". V_{dd} was the additional input parameter. Relevant data were saved for later use, and figures created and written to the computer screen in real time. PMOS transistors were UV-programmed and measured basically the same way.

3.3 Measurement Results

Drain currents for NMOS transistors in "nmos-rekke" are shown in figure 3.13. N5 lacked proper contact between drain and its dedicated pad, due to a forgotten via contact, therefore measurements for N5 were not included. For N4 there seemed to be no correct measurement result for the drain current as a function of the control voltage, as seen in the above mentioned figure. Since only one of the ten chips was used for this kind of measurements it has not been concluded whether the N4 NMOS was destroyed on this particular chip only. The N7 transistor, which has W/L=20.8μm / 1.2 μm had roughly 2-3 decades higher current levels for the stimuli applied for the measurements in figure 3.13. The slope of a floating-gate transistor's current voltage characteristic plotted on semilog axes is directly proportional to the capacitance of the floating gate [Minc00]. Devices N8 and N9 which shared a common capacitor between the control voltage and their shorted floating gates had a larger total capacitance seen from the floating gate. The needed change in the applied control voltage to change the current one order of magnitude for N8 and N9 is roughly twice the required change in control voltage to produce a similar change in the drain currents of for example N3 and N6, who have the same W/L ratio (figure 3.13).

Measured characteristics for all 9 PMOS transistors in "pmosrekke" are shown in figure 3.14. Smallest measured currents are in the tens of fA range. The range of measured currents spans almost 10 orders of magnitude. Drain currents as a function of the programming voltage for N3 are depicted in figure 3.15. The current level increased under regular use, for an increase in what was the programming voltage under UV-exposure. 0.8, 0.9 and 1.0 V were used as programming voltages (figure 3.8) on what was the source node under regular use. Similar measurements for the P6 transistor are shown in figure 3.16 for logarithmic currents, and for linear currents in figure 3.17.

The current levels for a particular V_{gs} increased for a decrease in the programming voltage for the PMOS, while they grew for a given V_{gs} for the NMOS for an increase in the programming voltage. This is illustrated in figure 3.18. Programming voltages and current levels for control voltages of 0.4 V for NMOS in figure 3.15, and 1.25 V for PMOS in figure 3.16 are shown.

N3 and N6 were identically drawn NMOS transistors. N8 and N9 shared a one piece poly 1 floating gate, but only N8 had a hole in the passivation layer to let UV-light through. N9 was UV-programmed indirectly, through N8's UV-hole. Ideally they should all have been of similar magnitude, which

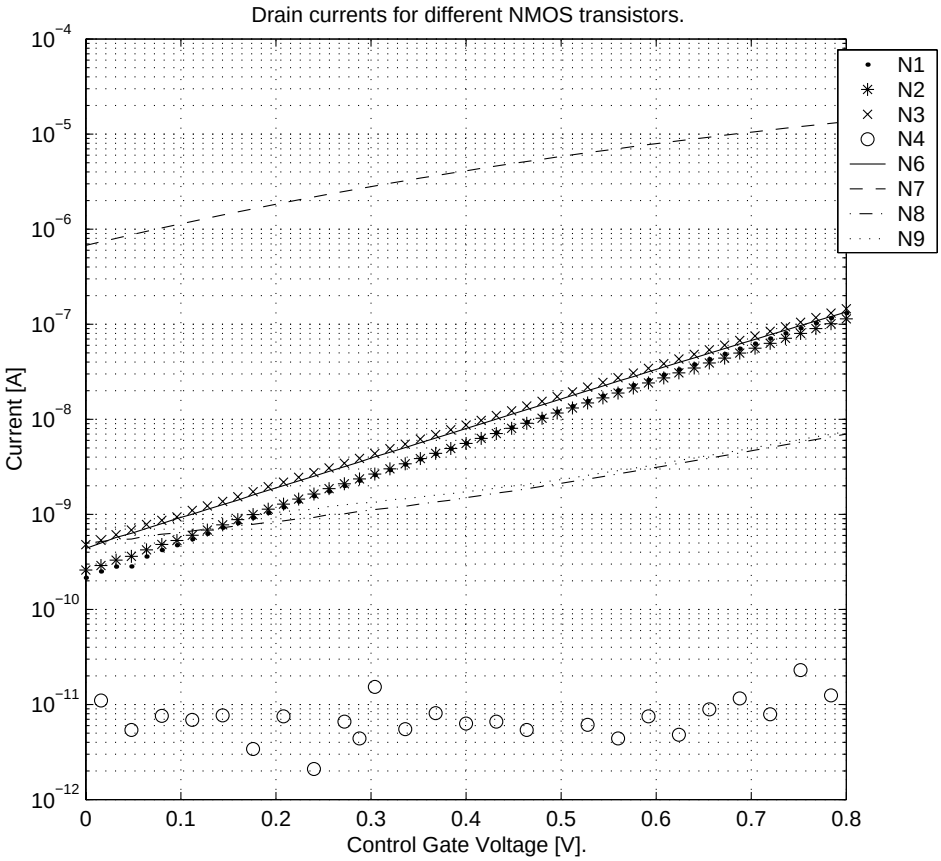

Figure 3.13: Drain currents for transistors N1,N2,N3,N4,N6,N7,N8 and N9 from the first chip. Programming voltage was 0.8 V.

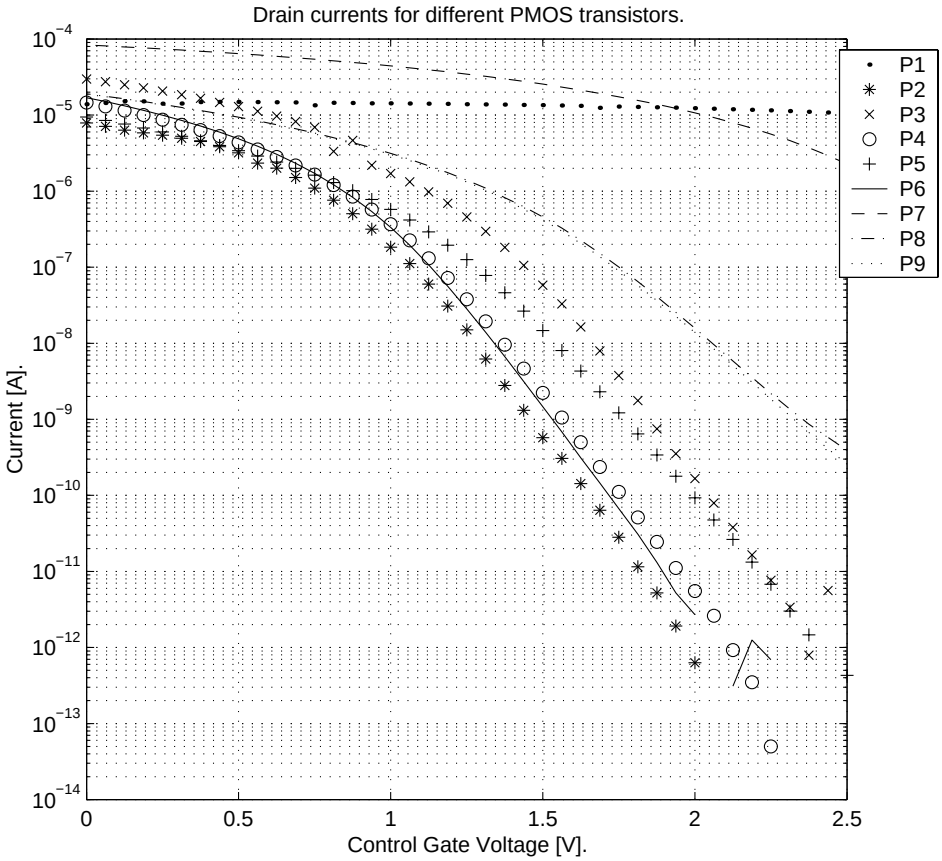

Figure 3.14: Drain currents for transistors P1,P2,P3,P4,P5,P6,P7,P8 and P9 from the first chip. Programming voltage was 0.1 V.

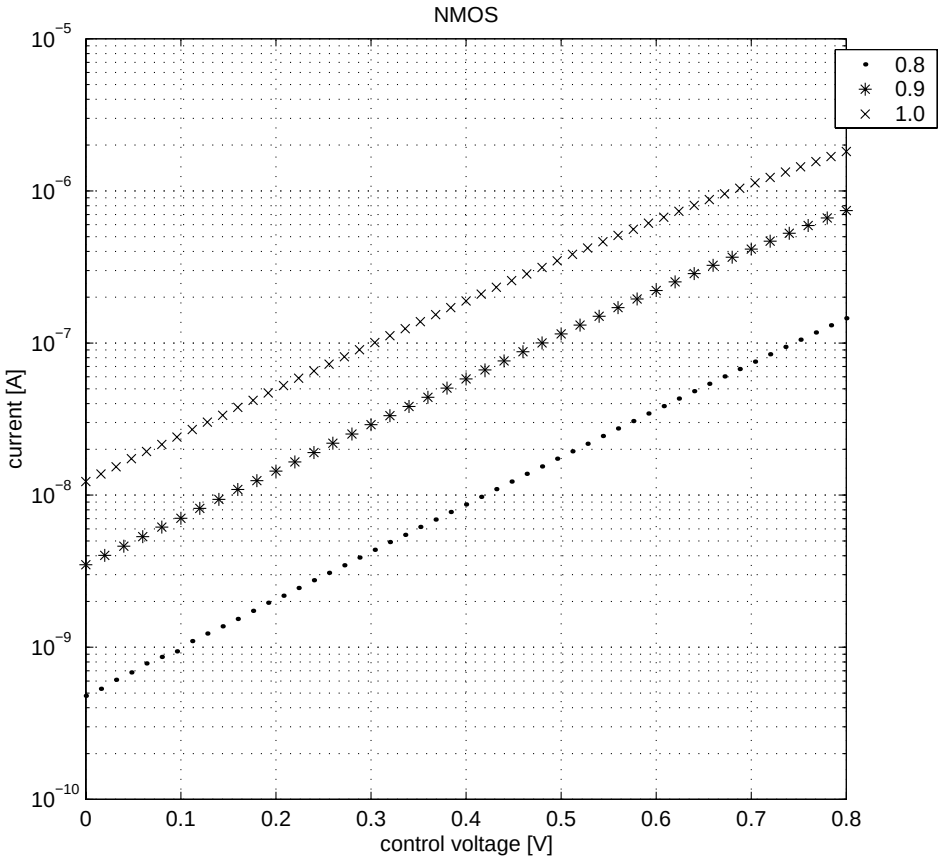

Figure 3.15: Current voltage characteristics for N3 were measured after UV-programming with 0.8, 0.9 and 1.0 V at what was the source node under regular operation.

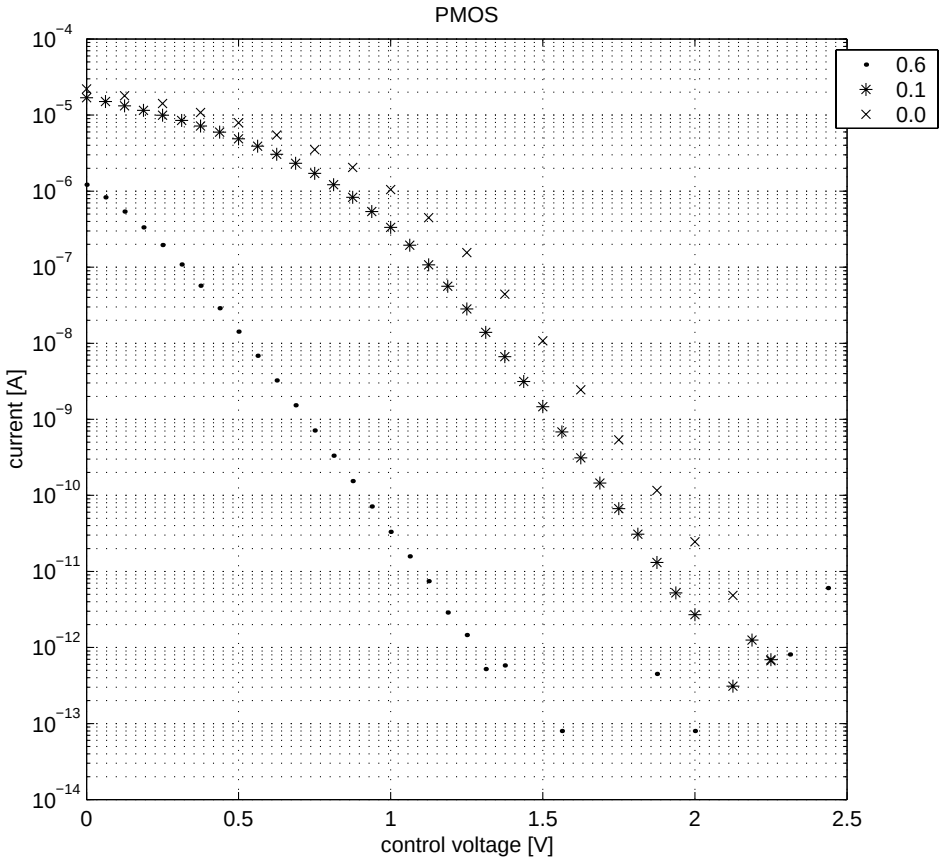

Figure 3.16: Current voltage characteristics for P3 after UV-programming with 0.6, 0.1 and 0.0 V at what was the source node under regular operation.

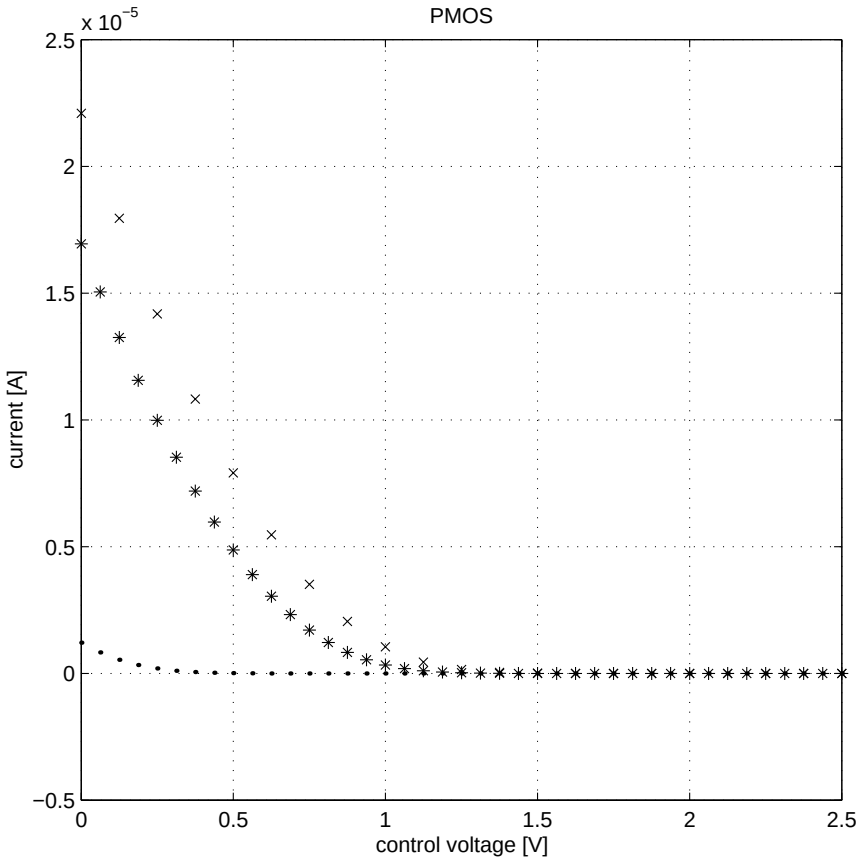

Figure 3.17: Current voltage characteristics for P3 after UV-programming with 0.6, 0.1 and 0.0 V at what is the source node under regular operation. Linear current.

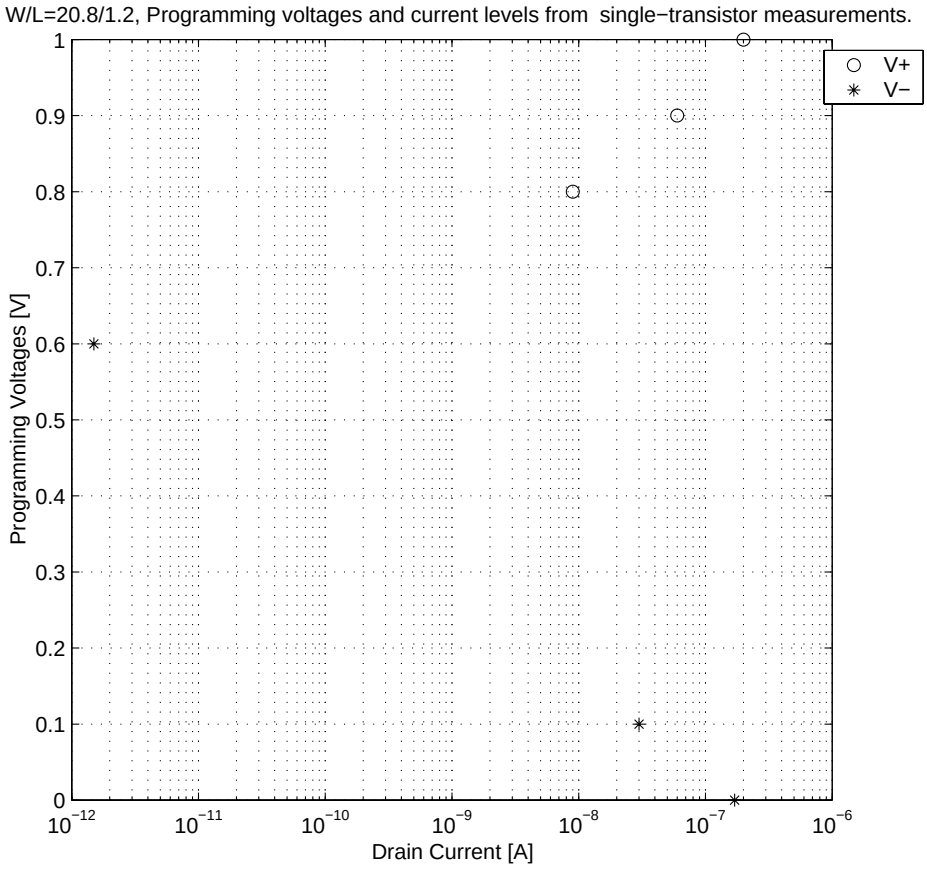

Figure 3.18: Programming voltages and drain currents, for PMOS ("*"), and NMOS ("o"). Measurements for the PMOS current were taken for a voltage of 0.4 V along the horizontal axis in figure 3.15, and for 1.25 V in figure 3.16.

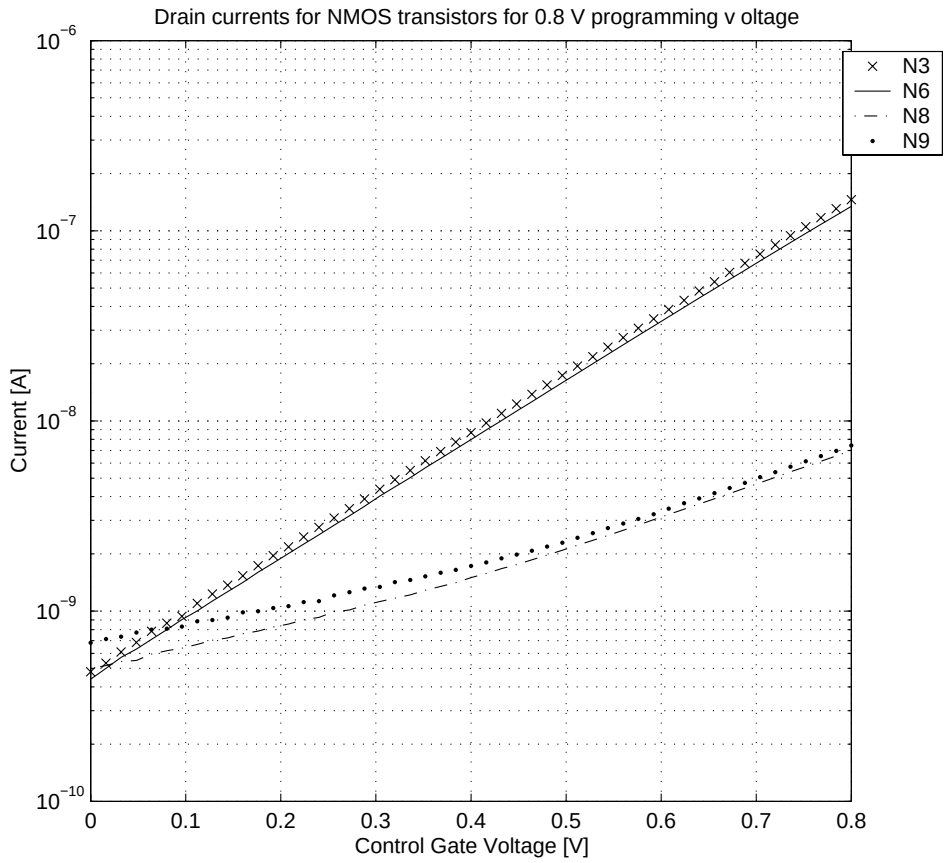

Figure 3.19: NMOS drain currents for a programming voltage of 0.8 V.

is not the case, as can be seen from figures 3.19, 3.20, 3.21.

P3 and P6 were identically drawn PMOS transistors, and P8 and P9 had the same dimensions as well, with P9 only indirectly UV-programmable. The matching of the PMOS transistors, according to these measurements, are generally much worse than with the NMOS (figures 3.19, 3.20, 3.21), as can be seen from figures 3.22, 3.23 and 3.24.

It can also be seen from figure 3.24 that P8 and P9, which shared a common floating gate, did not match well.

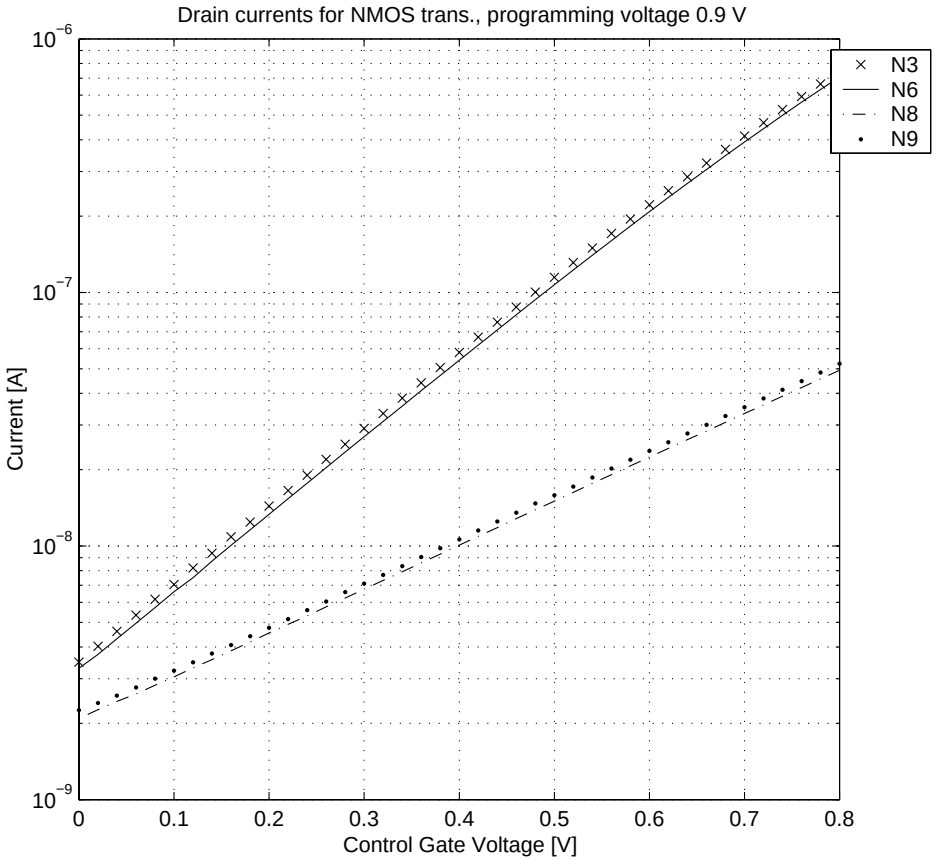

Figure 3.20: NMOS drain currents for a programming voltage of 0.9 V.

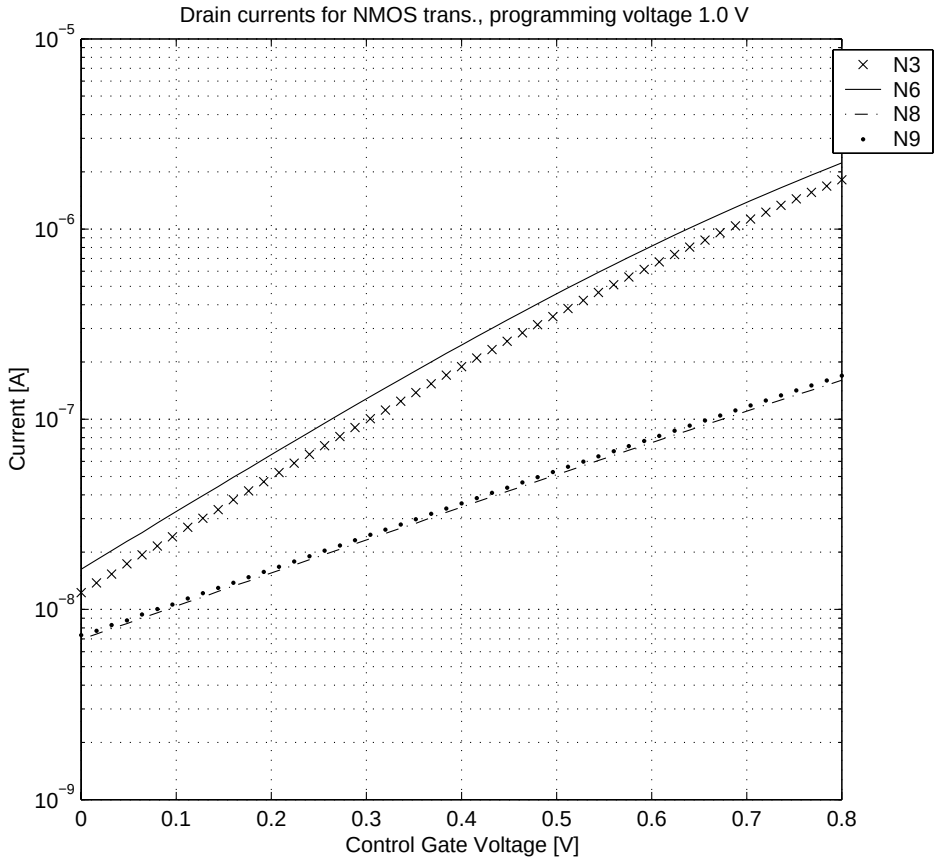

Figure 3.21: NMOS drain currents for a programming voltage of 1.0 V.

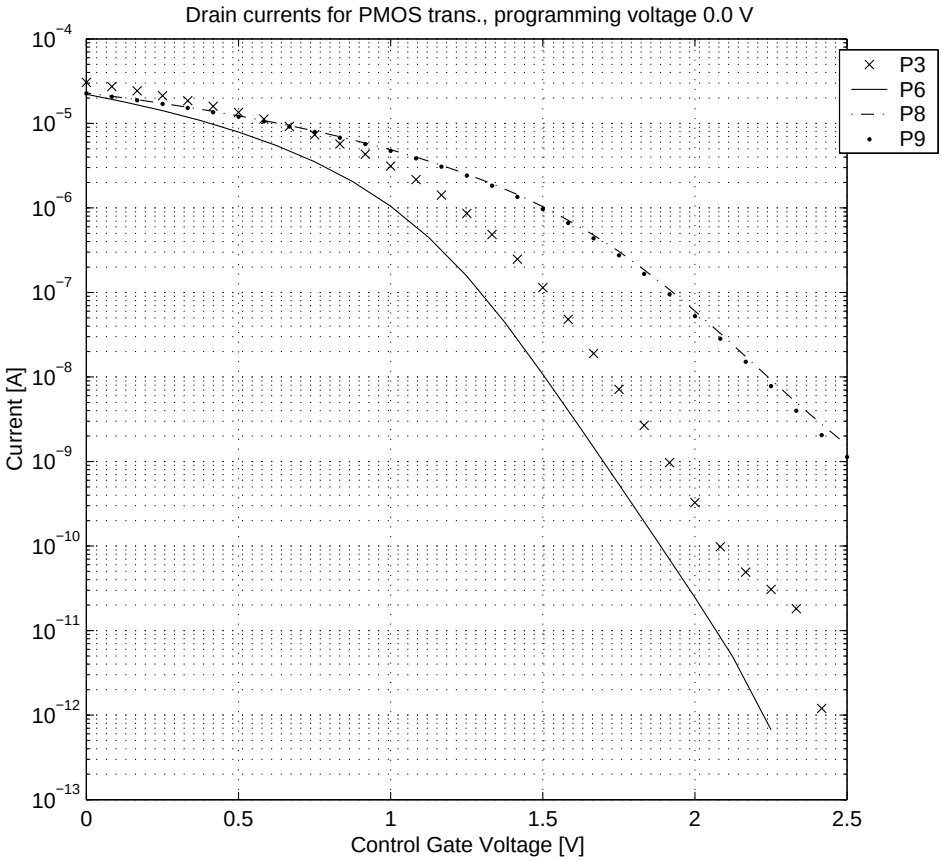

Figure 3.22: PMOS drain currents for a programming voltage of 0.0 V.

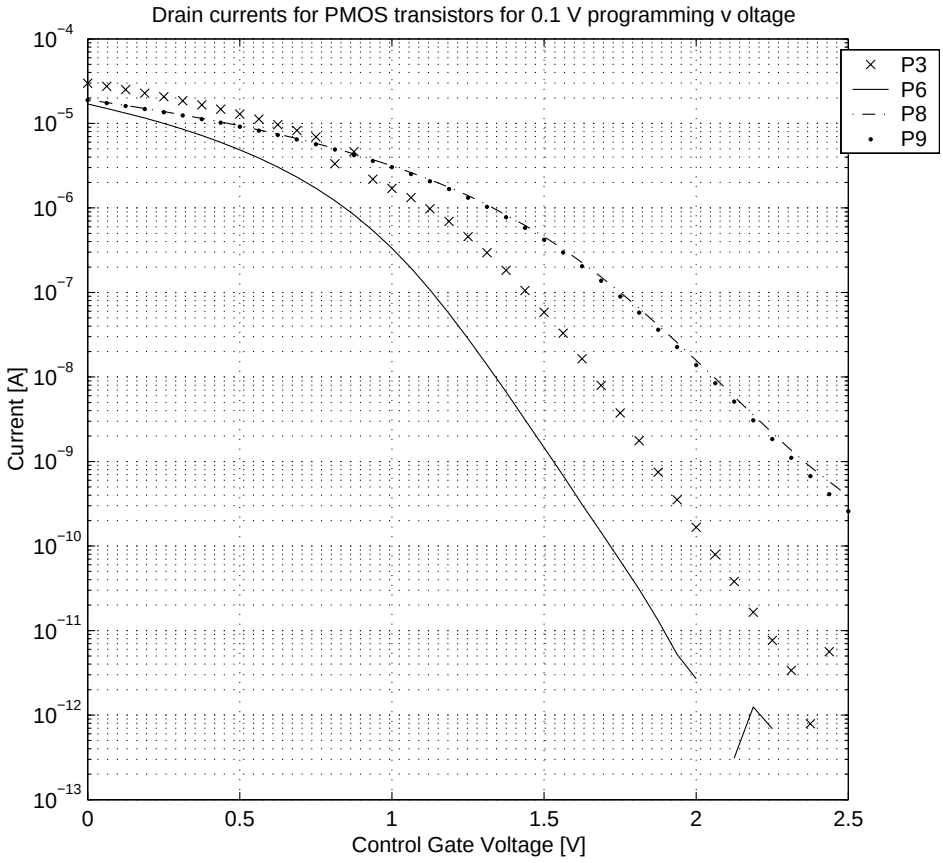

Figure 3.23: PMOS drain currents for a programming voltage of 0.1 V.

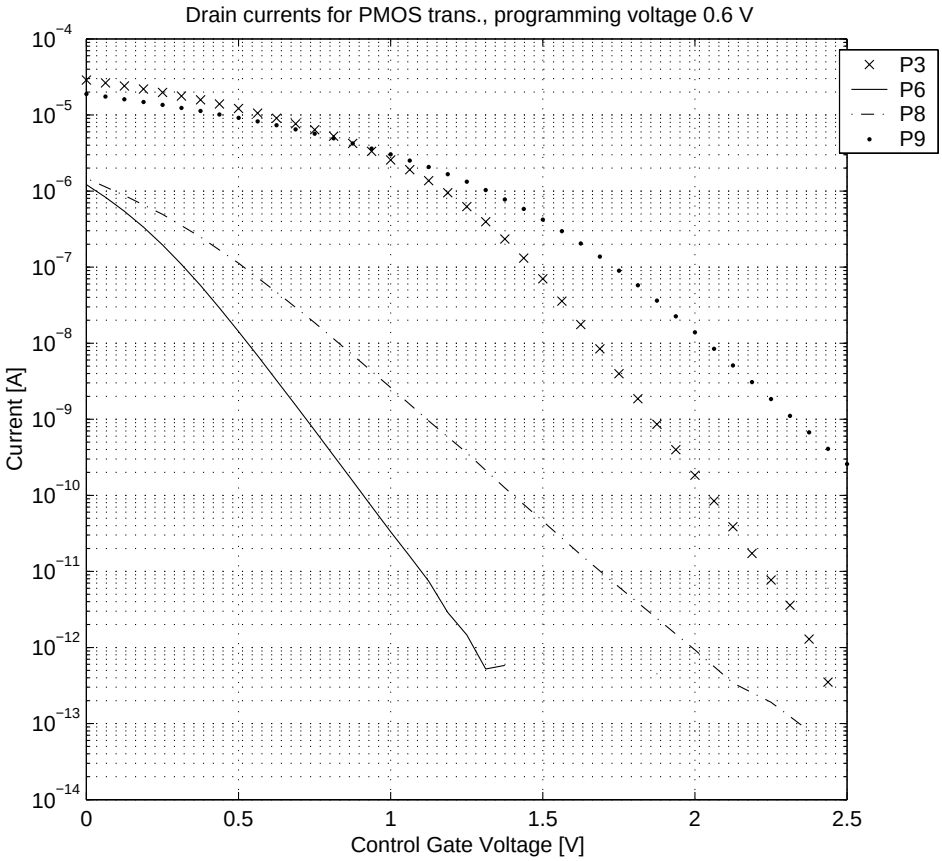

Figure 3.24: PMOS drain currents for a programming voltage of 0.6 V.

3.4 MOSFET discussion

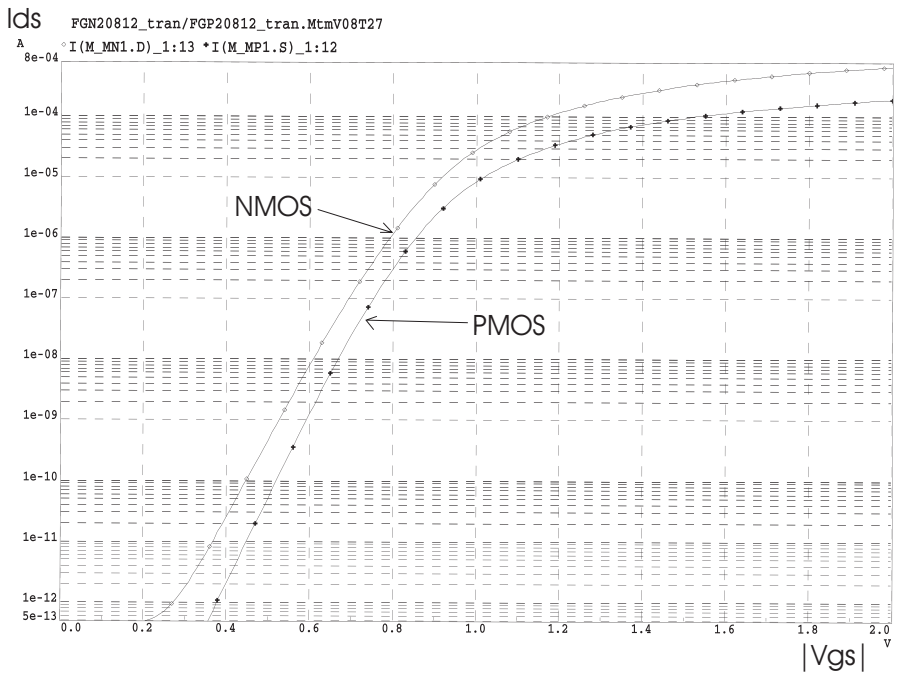

Figure 3.25: Simulated characteristics for PMOS and NMOS transistors with W/L=20.8μm /1.2μm. Uppermost curve is for the NMOS. Floating gate voltage on horizontal axis.

9 new "donut-shaped" floating-gate NMOS and 9 floating-gate PMOS transistors were designed and tested. The majority seemed to be functional.

Among NMOS transistors measurements depicted in figure 3.13, all MOSFETs except N4 showed a clear increase in the drain current for an increase in the control-gate voltage, and thereby the gate voltage. Only one chip was tested. Among reasons for the atypical behavior of N4, a fault in the test-setup, or a damaged device is believed to be among prime candidates. Another possibility might be that the voltage offset on the floating gate after production was too large to allow the limited time for UV-programming to make this test produce an IV-curve similar to the other transistors.

N7 had 2-3 orders of magnitude higher current levels than most other NMOS devices (figure 3.13). N7 had a slightly larger gate area than N3 and

N6, and should normally be expected to behave very similar to N3 and N6, except for steeper maximum slope for the two former. After production, the devices have a rather random charge on their individual floating gates. This might lead to very different characteristics, even for working identically drawn devices, if the UV-exposure continues until the UV-light no longer changes the effective threshold voltages for a given programming setup with fixed voltages to the drain, gate, source and bulk terminals. The UV-adaption might take several times as long as the UV-programming were going on in this case. UV-programming has been shown to last for up till about 24 hours before the process has converged, as demonstrated by measurements in [BeWi97]. Since the UV-programming lasted for some tens of minutes only, there might be a chance that a similar phenomenon was the reason for this somewhat odd result. Or, something else, such as a defect in the circuit may have influenced the measured behavior of the N7 device.

Measurements indicated that all different PMOS transistors seemed to work, though P1 did not demonstrate much of a change in its drain current due to a change in the control voltage, as can be seen in figure 3.14.

The slope of the current- voltage relationship as a straight line when plotted on a semilog axis, can be seen in measurements in figures 3.13 and 3.14, and simulations in figure 3.25. We also saw that N7 and N8, as well as P7 and P8, which had roughly twice the total capacitance at their floating gates showed less dependence on the control voltage than other transistors like, for example, N3 and P3. This was expected. From the equations [BeLa99b]

$$I_{ds,p} = I_{beq} \prod_{i=1}^{m} exp\{\frac{1}{nU_t}(V_{dd}/2 - V_i)k_i\} \tag{3.1}$$

$$I_{ds,n} = I_{beq} \prod_{i=1}^{m} exp\{\frac{1}{nU_t}(V_i - V_{dd}/2)k_i\} \tag{3.2}$$

with, $k_i = C_i/C_{tot}$ as the capacitive division factor of the ith input capacitor, C_i, and C_{tot} as the total capacitance seen from the floating gate, it can be seen that k_i determines how much of the signal that gets through to the gate of a floating-gate transistor. In figure 3.25 PMOS and NMOS transistors were simulated, with the voltage along the horizontal axis connected directly to the gate, with no "damping" of the control voltage via a capacitive network.

Both the NMOS and PMOS FGUVMOS devices showed adaptability to UV-light exposure, which is clearly shown by measurements in figures 3.15 and 3.16. For the NMOS, N3, an increase in the programming voltage from 0.8 via 0.9 to 1.0 V increased the current level for a given control voltage with more than an order of magnitude. The PMOS, P3, a similar increase in the current level comes for a decrease in the programming voltage from 0.6 via 0.1 to 0.0 V (figures 3.16, 3.17). This is in accordance with previous measurements published in [BeLa99a], [Gund00].

PMOS transistors generally did not match as well as NMOS transistors according to these measurements. In [MiKa00] a row of PMOS transistors were exposed to UV, and the conclusion was that the threshold voltage matching was very good, approaching the matching measured for the devices as they came from foundry. Possibly a longer period of UV-period would change these relationships. More empirical data should be made regarding the matching of PMOS vs NMOS if conclusions can be made. If PMOS should be shown to have worse matching capabilities than NMOS counterparts, like as in [AnBo91], this might influence the role of PMOS versus NMOS in FGUVMOS circuitry.

The donut shape of these transistors might lead to reduced parasitics [BeLa01b]. To change important parameters like g_m and r_{out}, the size of capacitances, transistor widths and lengths are among factors the designer can vary. An attempt to treat some aspects of this is summed up in figure 2.16.

In conclusion: Using the transistor building blocks described here we have succesfully made a different type of circuitry. This fact, along with the measurements in this chapter, suggests that these new, unique FGUVMOS transistors are fully functional.

Chapter 4

Floating-gate UV-programmable inverters

4.1 Implementation and layout of inverter

4.1.1 Simulation of the the FGUVMOS inverter

A floating-gate FGUVMOS inverter [BeLa97] is shown in figure 4.1. The floating gates for the PMOS and NMOS are denoted FGP and FGN, respectively.

The inverter function is illustrated by simulation, based on a netlist extracted from the layout ([Aune02]), in figure 4.6. In this case the decision was made to have a V_{dd} of 0.8 Volts and an equilibrium current of 10 nA. Initially in this transient simulation the situation when UV-programming is finished and the circuit is ready for normal operation, is simulated. Initially the circuit must be in the equilibrium state. To get $Vdd/2$ on the input and the output for an I_{beq} of 10 nA particular voltages on each of the two floating gates are needed. This pair of voltages depend on transistor sizing and layout, as well as the desired I_{beq}.

A possible way to find this set of voltages is to find them from the IV-curves, for a V_{ds} of $V_{dd}/2$, in our experience this is rather time-consuming. A more automated and significantly quicker method might be illustrated by the schematic depicted in figure 4.2 [AuBe01c]. Using this scheme, one manually finds the voltage that is needed to produce the desired equilibrium current on only one of the floating gates. After the voltage level for one of the floating gate nodes has been determined, for a given I_{beq}, the simulator may be used. This may be done using a piece of code as input to the simulator as in figure 4.3. For a given V_{dd} and one floating-gate voltage

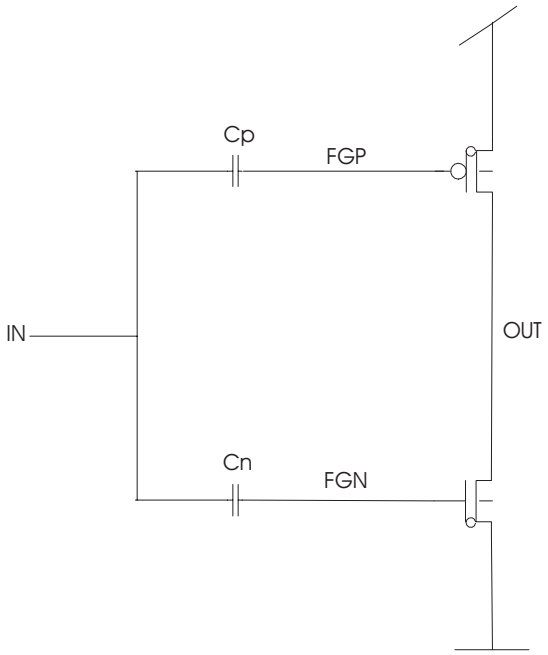

Figure 4.1: Floating-gate inverter. The additional circles in the MOSFET symbols indicate the UV-programmability.

the other floating gate is automatically computed for a given equilibrium current, I_{beq}, similar to the scheme in figure 4.2. Afterwards the pair of floating-gate voltages resulting is used as initial conditions in the simulation, as for example in figure 4.4, which has resulted in the simulation trace depicted in figure 4.5.

The older approach used included manually adjusting the floating-gate voltages most times within a mV, or fractions of a mV so that the driven nodes were within $V_{dd}/2$ + 1 mV. This was used as a "rule of thumb", based on practical experience with SPICE-based simulators, and could take several minutes for each NMOS and PMOS pair.

When simulating an FGUVMOS circuit, the initial conditions put the circuit in the equilibrium condition initially, before inputs start to change. For about the first 1/8 of the time period, the voltage on the floating gate of the PMOS, V(FGP8), as well as for the NMOS, V(FGN8) were kept constant, for an I_{beq} of 10 nA (figure 4.5). After this period the input voltage, V(IINC), started increasing from $V_{dd}/2$ towards V_{dd}=0.8 V. V(FGN8) and V(FGP8) both increased, which meant that the voltage between gate and source of the NMOS grew, while the voltage between

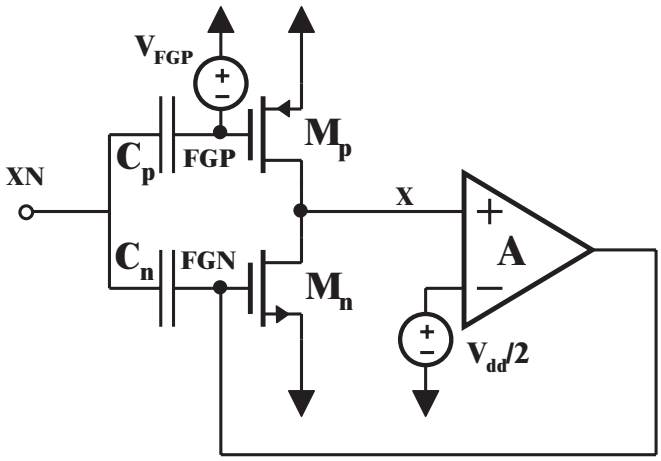

Figure 4.2: Schematic illustrating parts of method used in balancing of the circuit [AuBe01c].

```
* "inv2_20812_bal". Snorre Aunet, 010315
**** MOS parameters = tm / typical mean:
.LIB /dak2/ams/eldo/cux/cmos53tm.mod
****************************************
***** Supply voltage = 0.8V:
.temp=25.8
.param vdd=0.8V
.option ITOL=0.1e-15 RELTOL=1.e-7 RELTRUNC=1.e-6
.option VNTOL=10.e-6 NGTOL=0.1e-3 FLXTOL=10.e-12
.option CHGTOL=1.0e-15
.option GMIN =1e-50

vfgp fgp 0 dc 0.1310
vdd avdd 0 dc vdd
vdd2 vdd2 0 dc {vdd/2}
vss avss 0 dc 0
*cn fgn in 7.36e-14
*cp fgp in 7.36e-14
mp out fgp avdd avdd modp w=21.6u l=1.2u
mn out fgn avss avss modn w=21.6u l=1.2u

e1 fgn 0 out vdd2 10000000
.op
.plot dc v(fgn)
.extract dc v(fgn)
```

Figure 4.3: Input to the Eldo simulator for generating a pair of voltages on PMOS and NMOS floating gates for the initial equilibrium condition.

source and gate of the PMOS decreased.

As V_{gs} for the NMOS increases for a given drain current, the output resistance decreases. When at the same time the V_{sg} voltage decreases for the PMOS its output resistance increases. The voltage at the output of the inverter stems from the voltage division. In the mentioned simulation it went low (figure 4.5). The PMOS limited the current level as the output voltage, V(IOUT1C), moved towards $V_{ss} = 0$ V.

```
* "CHIP5_INV2_20812_sweep.cir". Snorre Aunet, 010502
**** MOS parameters = tm / typical mean:
.LIB /dak2/ams/eldo/cux/cmos53tm.mod
*****************************************
**** Supply voltage = 0.8V:
.param _vdd=0.8v
.param vdd=0.8v
***********************************
**** Temperature =27 degr. Celsius:
.temp=27
***********************************
.option ITOL=0.1e-15 RELTOL=1.e-7 RELTRUNC=1.e-6
.option VNTOL=10.e-6 NGTOL=0.1e-3 FLXTOL=10.e-12
.option CHGTOL=1.0e-15
*.option GMIN =1e-50
.LIB profile_SA.opt
*ut
.IC V(iout3)=0.400
.IC V(iout2c)=0.400
.IC V(iout1c)=0.400
*inn

.IC V(iinc)=0.400
.op
*10 nA
.IC V(FGN8)=0.60882
.IC V(FGP8)=0.1310
.IC V(FGN9)=0.60882
.IC V(FGP9)=0.1310
.IC V(FGN10)=0.60882
.IC V(FGP10)=0.1310
.connect VDD PADVDD2
.connect VSS 2
.connect nwell2 ivdd
.connect VDD ivdd

VVDD vdd 0 _VDD
VVSS AVSS 0 0
.connect vdd AVDD
.connect VSS ivss1
.connect ivss1 ivss2
.connect 0 VSS

* FOR INITIAL BALANCING OF CIRCUIT:
V1 N1 0 PWL(0 {VDD/2}100n 0)
V4 iinc N1 PWL(0n 0 100n {VDD/2}100U {VDD/2}800U {VDD}1600U 0)
*V6 IN2 N1 PWL(100n 0 150n {VDD/2}800U 0 1600U {VDD})
.include CHIP5_capall_netlist
*.include CHIP5_netlist
.tran 800us 1600.2us 0us UIC
.plot tran V(vdd)  V(iout1c) V(iinc) V(FGN8) V(FGP8)
.plot tran I(M2371.D) I(M2309.S)

.end
```

Figure 4.4: Input to the Eldo simulator for transient simulation. Initial conditions (.IC commands) are used for floating gates and driven nodes.

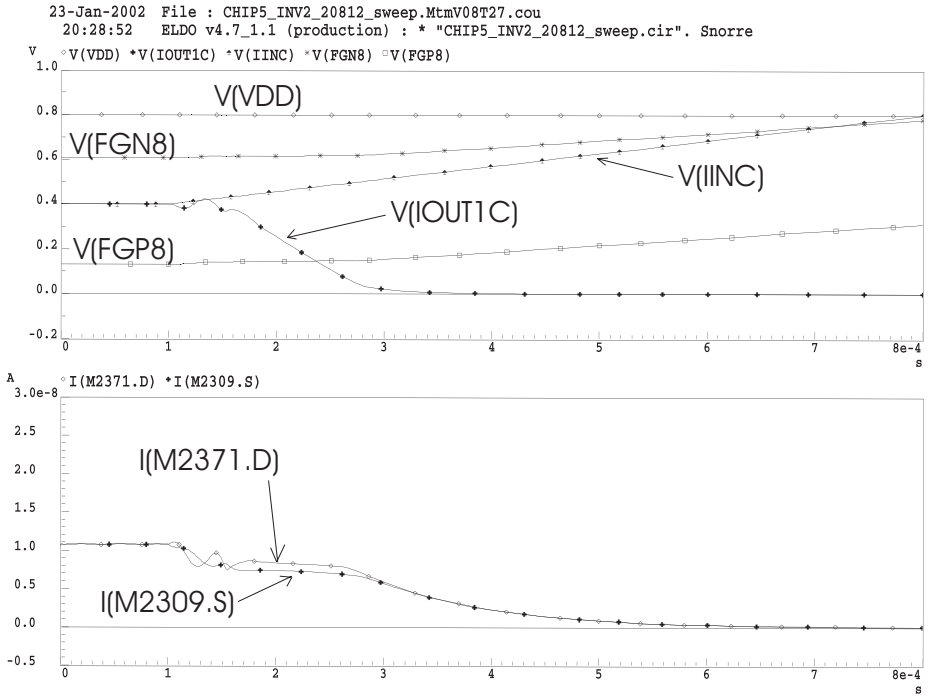

Figure 4.5: Simulated inverter with W/L=20.8μm/1.2μm and $C_p = C_n$=73.6fF. The node names correspond directly to the names in the netlist, "*CHIP5_capall_netlist*" [Aune02]. M2309 is the same as the PMOS.

4.1.2 Layout of the FGUVMOS inverter

The layouts for the inverters were made using the W/L=20.8 μm/1.2μm PMOS and NMOS transistors described in chapter 2 where the MOSFET building blocks are presented.

Both capacitances between the input and the floating gates are of poly1-poly2 type and 73.6 fF by design. The greyish areas in the centers of the two transistors are the "UV-windows", or holes in the passivation layer. The greater part of the horizontal metal lines are made from the 2nd metal layer, "metal 2". Metal 2 is used for protecting selected parts of the transistor from UV-light, as well as for wiring.

There is a limit in the design rules regarding the maximum size of the plate of the drawn capacitances that is shared with the active gate poly [AMS9X]. If it is exceeded it may damage the circuits under production. In some cases it set upper limits for the sizing of poly1-poly2 capacitances.

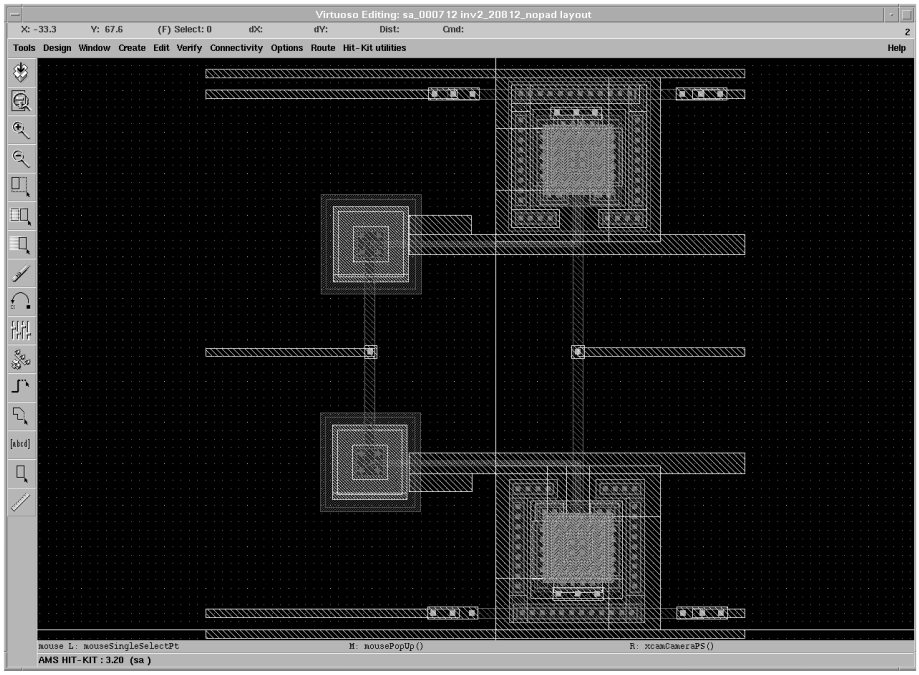

Figure 4.6: Layout for single inverter, including W/L= 20.8 μm / 1.2 μm PMOS and NMOS elements. Drawn capacitances between the input and both floating gates are 73.6 fF.

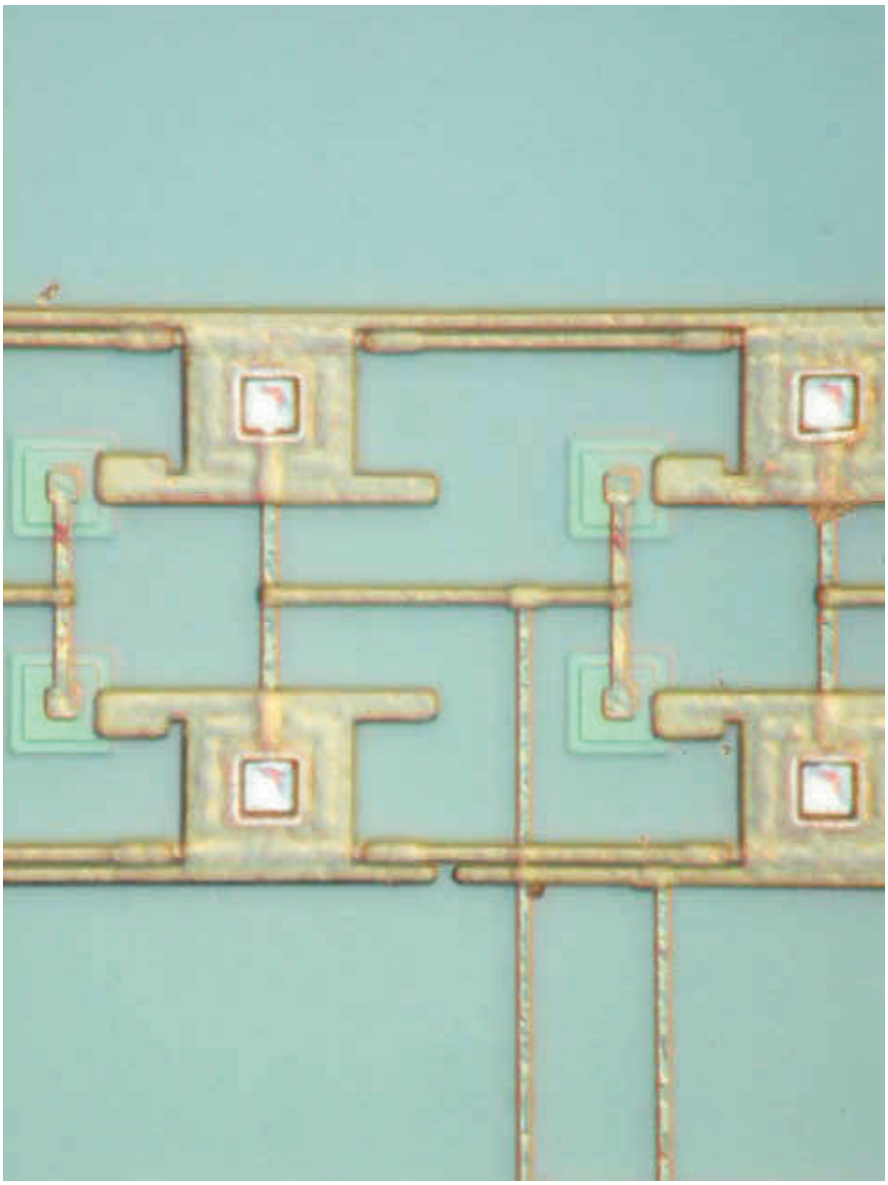

Figure 4.7: Chip photograph for inverters, including W/L= 20.8 μm / 1.2 μm PMOS and NMOS elements.

4.2 UV-programming and test setup

4.2.1 Basic information regarding UV-programming and test setup

Important goals of the UV-programming are to ensure that the subcircuits operate with the desired equilibrium current(s) for a chosen V_{dd}.

The equilibrium current, I_{beq}, of a general FGUVMOS element, when all inputs and driven nodes are at $V_{dd}/2$, can be varied over several decades [BeLa99a]. The circuits might be reprogrammed for different I_{beq}s , which correspond to different pairs of threshold voltages for PMOS and NMOS transistors, "seen" from their driving nodes. The most commonly used programming method [BeLa99a], [BeLa01b], which was also the basic one for this work, will be briefly described here.

In figure 4.8 an FGUVMOS inverter is shown in both the "computational" and "UV-programming" modes. The circuit will always have a potential at the floating gates, FGN and FGP, determining the effective threshold voltages of the PMOS and NMOS seen from the driving input(s).

In the UV-programming mode a "reverse biasing" scheme is used to make use of UV-actived conductances to let charges flow to the floating gates. For an input voltage of $V_{dd}/2$, the voltage levels on the power rails and substrates are adjusted so that the driven node approaches $V_{dd}/2$, for a certain equilibrium current. When the UV-light is removed, the normal biasing can be used.

The programming technique has been described by the following steps [BeLa99a], and used in several later works [Gund00], [Bahr01], [Dani01], [Flat01].

The main steps are [BeLa99a]:

1. Decide the operative (normal biasing) supply voltage, V_{dd}.
2. Apply $V_{dd}/2$ to all external inputs.
3. Apply the programming voltages at the supply rails, V_- at V_{dd} and V_+ at gnd/vss.
4. Terminate the programming by removing the UV-light source when an (any) output (external) converges to $V_{dd}/2$.
5. Set the biasing voltages to normal values.

In reality there might be some differences, such as running point 2 to 4 several times, before the output has converged towards $V_{dd}/2$. By observing one of the output voltages and/or the current flowing through the circuit in real time, the UV-programming can be corrected. If, for example, the output voltage (step 4) seems to converge towards a value

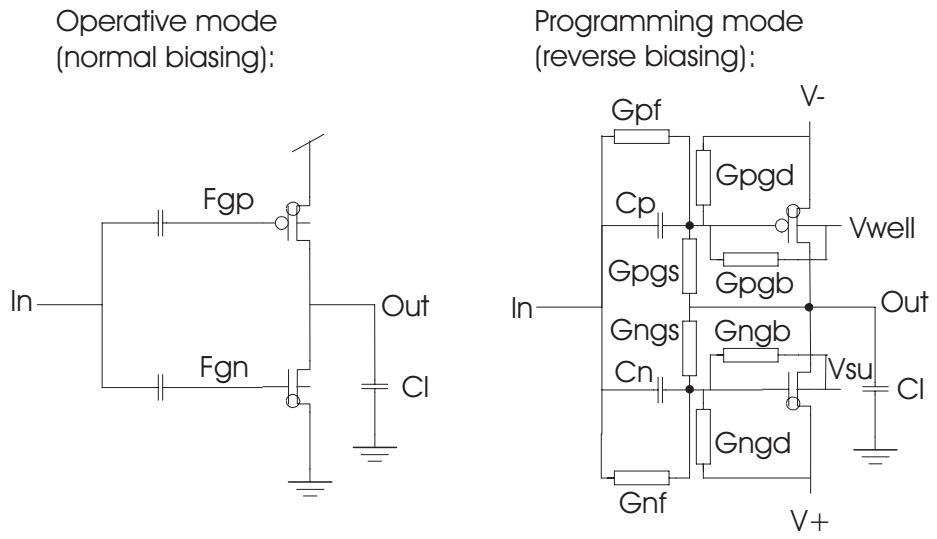

Figure 4.8: FGUVMOS circuit under operative mode and UV-programming mode [BeLa99a].

different from $V_{dd}/2$, the programming voltages can be changed. By making such a correction the UV-programming process can be speeded up.

Different sets of programming voltages applied to power rails and substrates can give equilibrium currents varying over several decades, as demonstrated by measurements in [BeLa99a], [BeLa01b].

Sometimes the nwell voltage level has been used to adjust the DC transfer characteristics after some UV-programming, to force the switching voltage point of the FGUVMOS element(s) towards $V_{dd}/2$.

The UV-programming process and data acquisition are controlled from a computer terminal after the initial instrument setup, as illustrated in figure 4.9. Figure 4.10 shows an inverter tested, with node names from netlists from [Aune02]. More details regarding instruments used can be found in figure A.2, under the "inverter" column. The test chip and test PCB were the same as in figure 3.12.

Important parameters could be observed on the computer screen in real time, as in figure 4.11. The first one displayed the output voltage of an inverter under UV-programming and normal operation, as a function of time. Here the V_{dd} was 0.8 V, so they should both approach $V_{dd}/2$=0.4 V. The second one ("Figure No. 106") showed a measurement of the current through the inverter as a function of the input voltage, with an I_{beq} about 20 μA. "Figure No. 105" showed the output voltage as a function of the

input voltage. "Figure No. 121" displayed the derivatives of the curves in the first picture, while "Figure No. 102" provided a closer look at the output while the devices were UV-radiated. The final one, "Figure No. 162" displayed the voltage gain of the inverter.

4.2.2 UV-programming taking from tens of minutes to many hours for a few transistors

UV-programming for the circuitry demonstrated in this work took about half an hour or more, after suitful programming voltages have been found. That in itself might take much more time. Often the nwell voltage were adjusted instead of letting the UV-programming converge, which was done to speed up the production of data.

4.2.3 Additional information regarding UV-programming and test setup

Additional details regarding measurement setups can be found in pieces of matlab code that were used. Matlab codes very similar to ProgUVmin (figures B.1, B.2, B.3), were used in the UV-programming, while InvSweep (figure B.4) was the main code used for measurements on the programmed chip. Other minor code parts are also used as routines, for example activated by ProgUVmin. This includes code in figures B.5 and B.6.

UV-programming was run several times before the final measurements were taken. Saved data from those runs are listed in figure 4.12. This UV-programming was done prior to the inverter measurements presented later. There are several abbreviations (figure 4.12). PVdd and PVss mean $V-$ and $V+$ in figure 4.8, while Pwell is the applied voltage to the well under UV- programming. TVdd and Twell denote the applied voltages on Vdd and the well while doing a measurement in operative mode (figure 4.8). Vinstop is the final applied voltage to the input during test. The rest of the parameters affected number of measurements or programming time. Programming these circuits seemed to take a normal amount time from experience with the AMS 0.6 CMOS process, expressed as "from 10-15 minutes to hours" [Loms02].

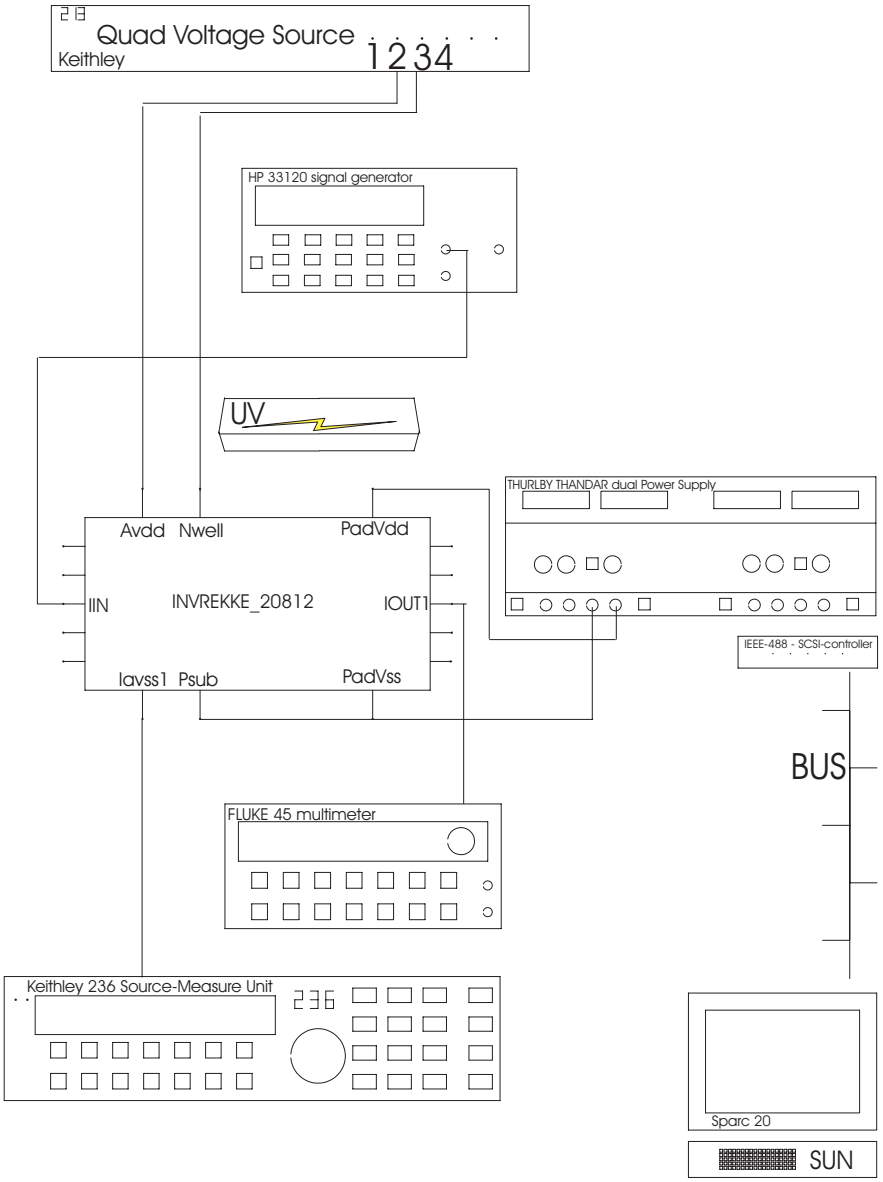

Figure 4.9: Laboratory setup for inverter measurements from chip no. 1

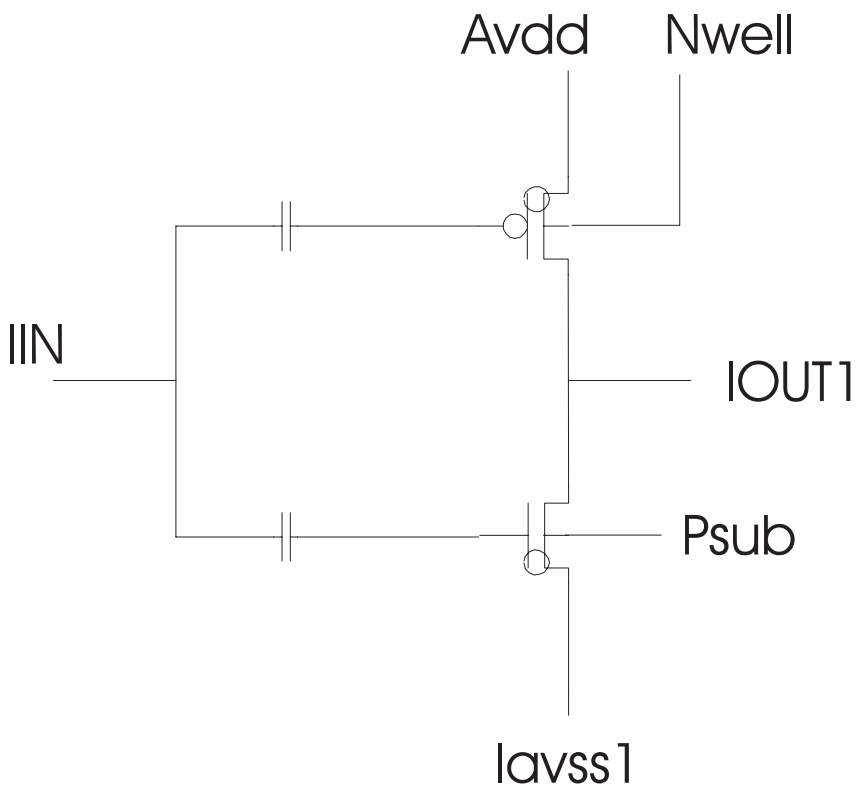

Figure 4.10: The first out of 10 inverters in "$INVREKKE\_20812$" from figure 4.9.

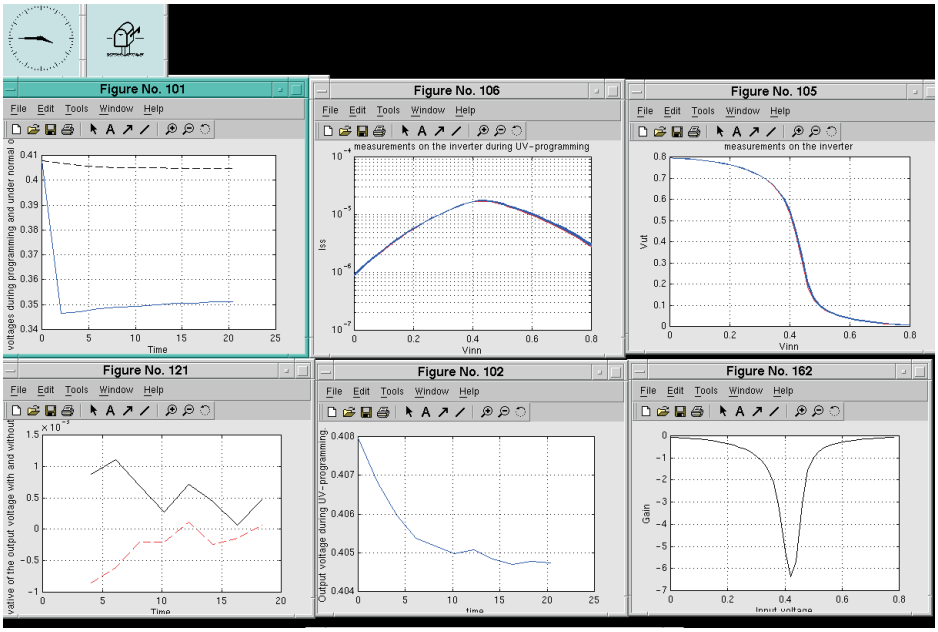

Figure 4.11: Data that were displayed on the computer screen during UV-programming.

matlab	TVdd	Twell	TStep	PVdd	PVss	Vinstop	Pwell	PTid
P1-P5	0.8 V	1.5 V	10	0.38 V	1.0 V	0.8 V	0.8 V	50min.
P6	0.8 V	1.2 V	20	0.49 V	1.2 V	0.8 V	1.6 V	20min.

Figure 4.12: Different applied voltages for UV-programming and test for inverters on the 68 pin chip. Programming was run 5 times with the values in the "P1-P5" row and, finally, once with the values in the "P6" row.

4.3 Inverter measurement results

Figures 4.13, 4.14, 4.15 and 4.16 present measurement results for the inverter after UV-programming for a V_{dd} of 0.8 V, for programming and test setup as in figures 4.9, and stimulus as in first line in figure 4.12. The input voltages can be seen along the horizontal axis. Without reprogramming the circuit, the supply voltage was lowered in steps to 95 mV for the results in figure 4.13.

The voltage gain, from input to output, was less than -1 for a V_{dd} down to 95 mV, and up to 800 mV (figure 4.14). Currents for different supply voltages and as a function of input voltage are plotted on linear scales in figure 4.15, while a logarithmic scale for the current, for $V_{dd} = 0.8V$, is used in figure 4.16.

For figure 4.17 the 2nd, "P6", line was used (figure 4.12). 93 mV is the lowest V_{dd} for which we have measured a minimum voltage gain less than -1. Results were published in [AuBe01a].

In figure 4.18 the same inverter was programmed differently, once with the P6 scheme in figure 4.12, and once with P1 (figure 4.12). The highest current level and lowest voltage gain is correlated with P6 (figure 4.12).

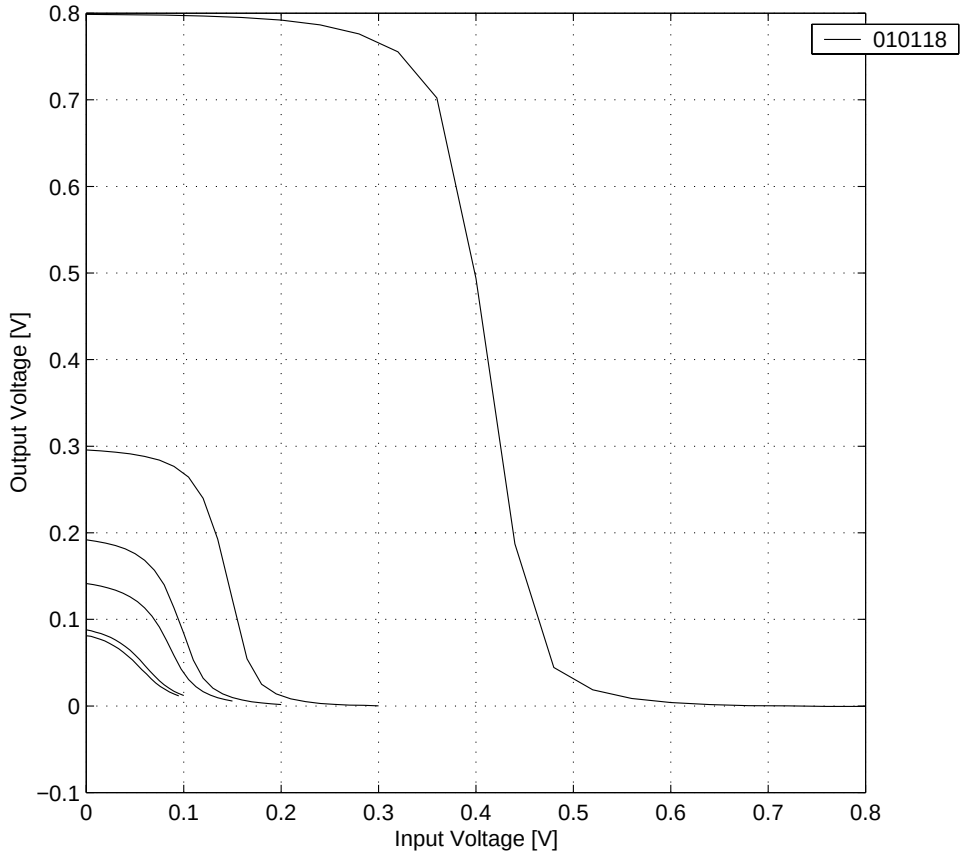

Figure 4.13: Measured voltage transfer curves for power supply voltages of 95 mV, 100 mV, 150 mV, 200 mV, 300 mV and 800 mV.

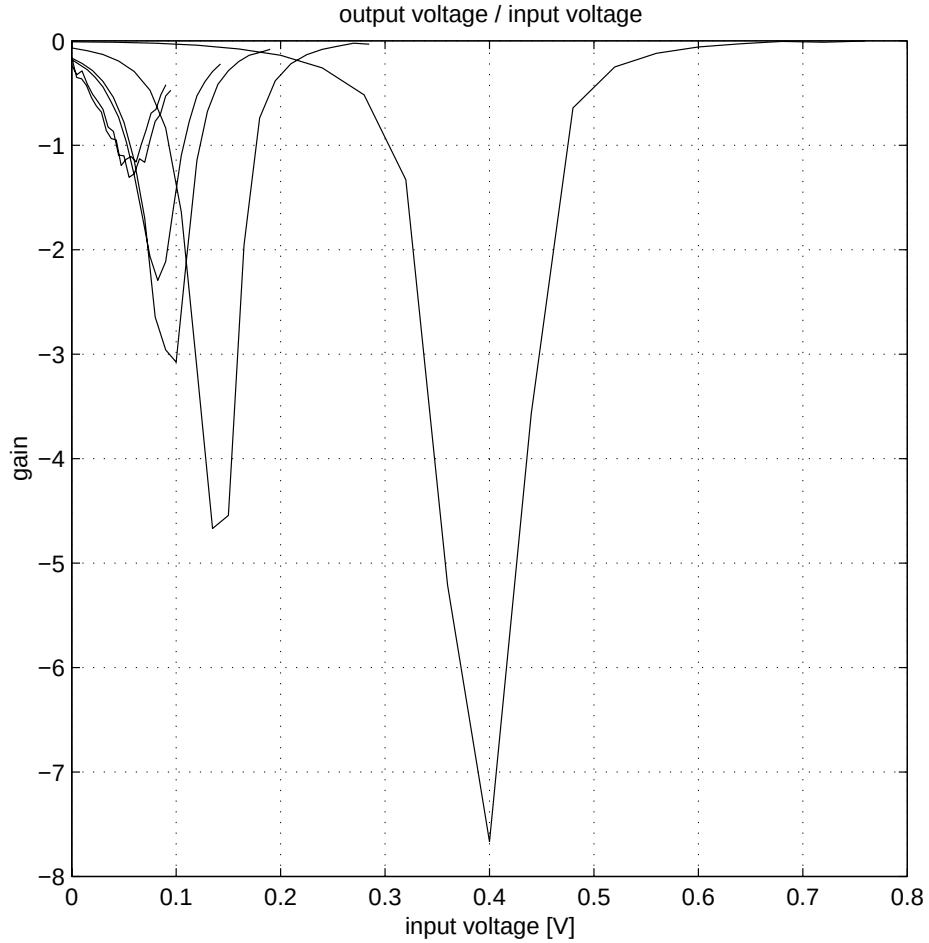

Figure 4.14: Voltage gain based on measurements, for power supply voltages of 95 mV, 100 mV, 150 mV, 200 mV, 300 mV and 800 mV.

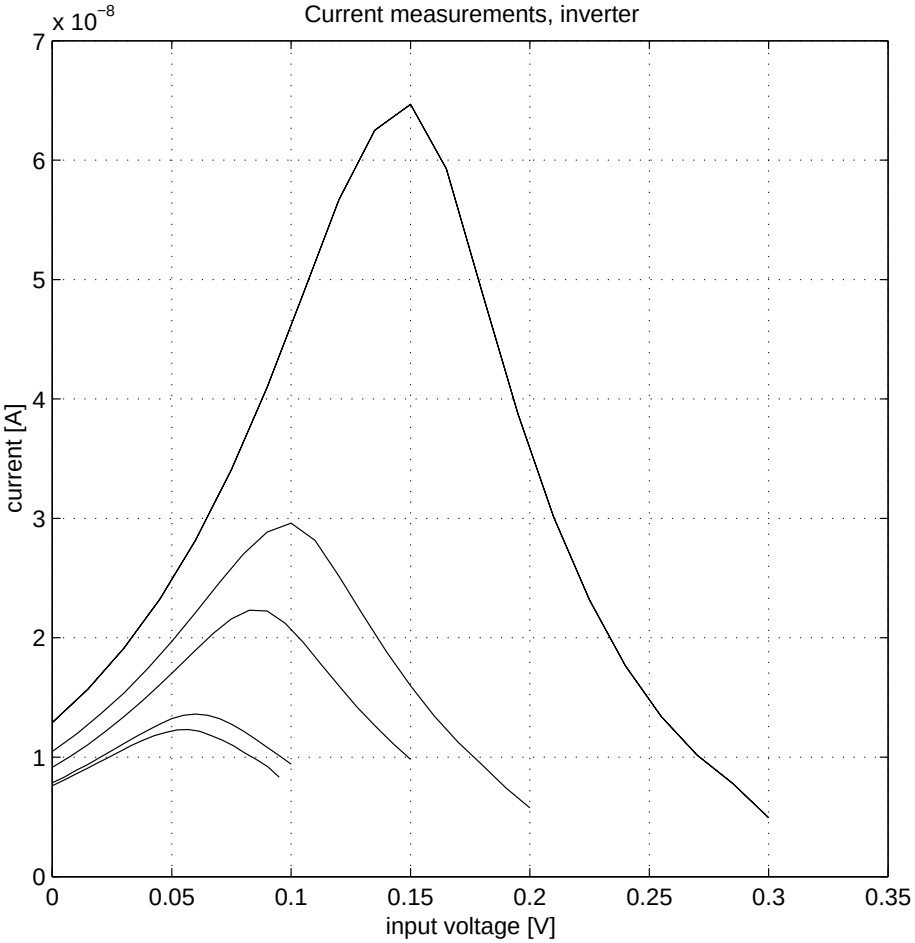

Figure 4.15: Measured currents through Vss, for power supply voltages of 95 mV, 100 mV, 150 mV, 200 mV, and 300 mV.

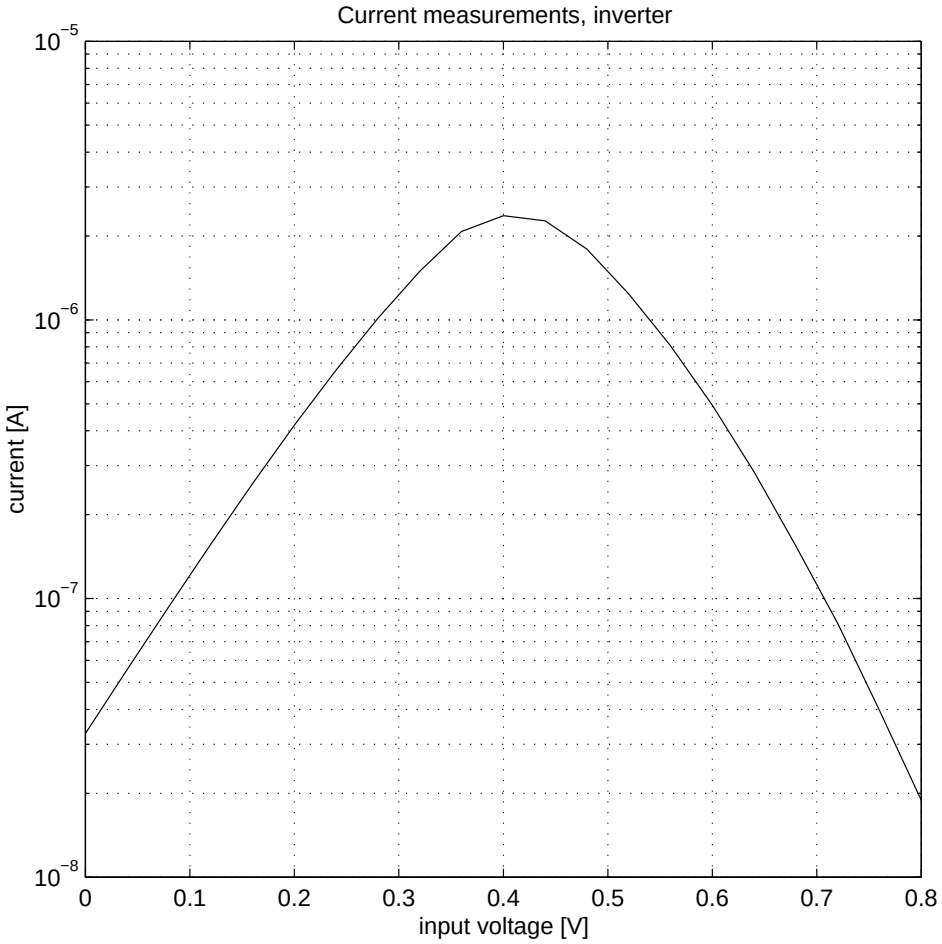

Figure 4.16: Measured current through Vss for a power supply voltage of 800 mV.

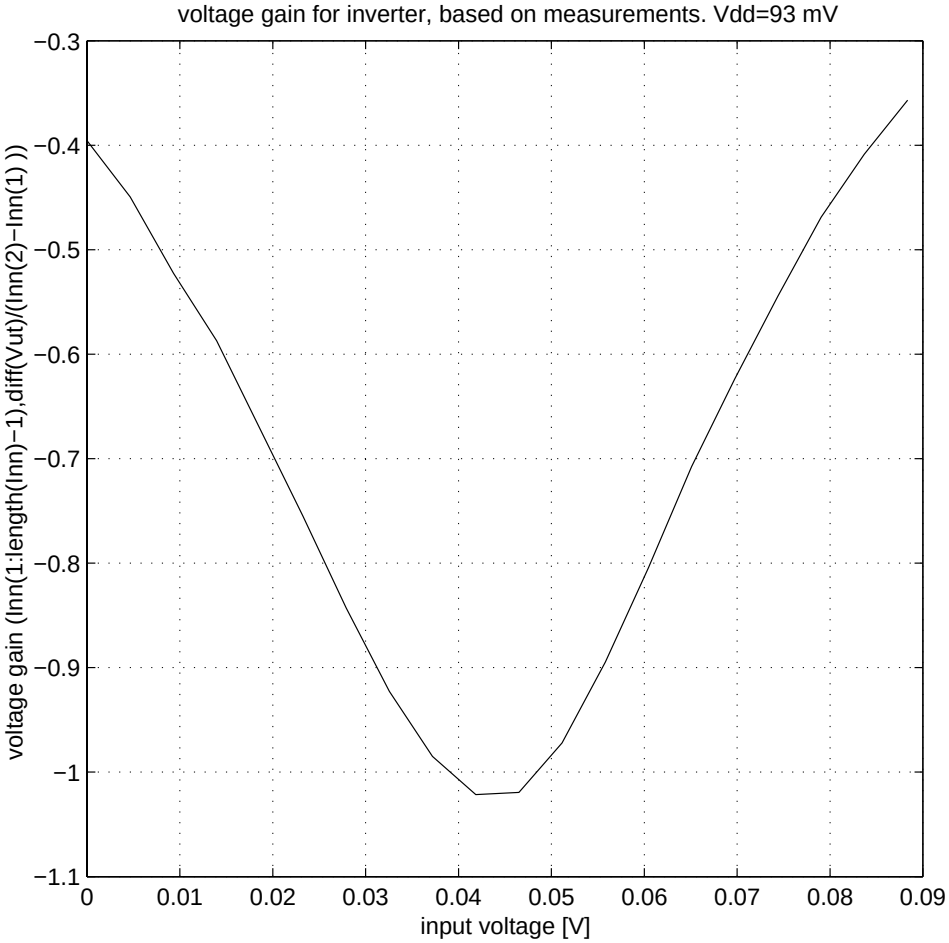

Figure 4.17: Voltage gain based on measurements for a power supply voltage of 93 mV.

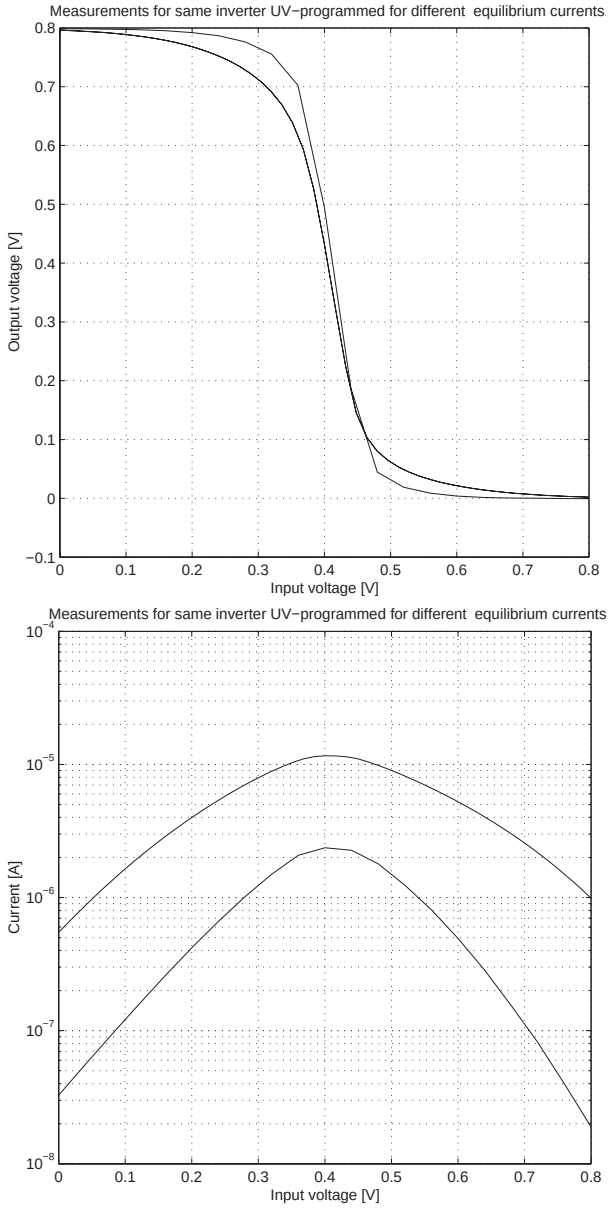

Figure 4.18: Measurements of current levels and voltage transfer characteristics for the same inverter after two UV-programming for different current levels / threshold voltages.

4.4 Simulations of capacitors between inputs and floating gates, and operational speed

In order to find the effect of the sizing of capacitances between inputs and floating gates has on the operational speed of the circuits, some simulations were done for the inverter, using drawn capacitances of 73.6 fF and 122.7 fF and W/L=20.8μm / 1,2μm (figures 4.19 and 4.20). The simulations were based on a netlist extracted from layout (*CHIP5_capall_netlist* [Aune02]), hopefully to give more realistic results than if a netlist extracted from schematics had been used.

t_r and t_f are rise-time and fall-time, respectively, measured on the output of the 2nd inverter for three identically drawn inverters in series. The inverters can be seen in the photo in figure 6.7.

V_{dd} [V]	I_{beq} [nA]	t_r[ns]	t_f[ns]	$(t_r + t_f)$ / 2[ns]
0.2	1	17616	19148	18421
0.2	10	2016	2119	2072
0.2	100	269	270	270
0.4	1	5933	7011	6496
0.4	10	799	871	838
0.4	100	134	131	133
0.8	1	751	907	834
0.8	10	161	161	162
0.8	100	51	41	46

Figure 4.19: Layout-based simulation, for 73.6 fF (C_n) capacitances between input and floating gates.

As can be seen (figures 4.19 and 4.20), the size of the capacitor seems to be important regarding the potential maximum operational speed of the inverters.

An optimum value for the sizing of capacitances regarding speed could be expected, since when the load capacitance for a certain current level increases, at some point the speed will decrease because the current level is too low to be able to keep up the speed of charging and decharging the capacitance ($dV/dt = I/C$).

In search for a kind of optimum capacitance size, with regard to operating speed, the rise-times were simulated like before, but capacitances varied. For simplicity, the capacitances between the input and the floating gates were increased while capacitances extracted from layout, between the

V_{dd} [V]	I_{beq} [nA]	t_r[ns]	t_f[ns]	$(t_r + t_f)$ / 2[ns]
0.2	1	15819	18801	18421
0.2	10	1893	2054	2072
0.2	100	259	248	270
0.4	1	5185	6987	6496
0.4	10	765	853	838
0.4	100	138	122	133
0.8	1	577	745	834
0.8	10	149	132	162
0.8	100	51	31	46

Figure 4.20: Layout-based simulation for 122.7 fF capacitances between input and floating gates.

V_{dd} [V]	I_{beq} [nA]	t_r[ns]	t_f[ns]	$(t_r + t_f)$ / 2[ns]
0.2	1	17646	19195	18421
0.2	10	2019	2124	2072
0.2	100	269	271	270
0.4	1	5949	7042	6496
0.4	10	801	874	838
0.4	100	134	132	133
0.8	1	754	914	834
0.8	10	162	162	162
0.8	100	51	41	46

Figure 4.21: Layout-based simulation for inverter for C_n=74 fF. "Parametrized" capacitances between floating gates and substrate.

floating- gate and substrate, were scaled to 28% of the "input capacitance" for the NMOS and 27% for the PMOS, based on relationships in the netlist [Aune02]. The results of such a simulation for a capacitance between input and floating gates of 74 fF are reported in figure 4.21. They might be compared to those of figure 4.19.

For the simulation results in figures 4.22 and 4.23, the capacitances were varied for different supply voltages, V_{dd} and equilibrium currents, I_{beq}. C_n refers to the capacitance between input and the floating gates. Results can be viewed graphically in figures 4.24, 4.25 and 4.26.

For all supply voltages and the capacitance values of this simulation there is a relatively rapid decrease in the fall time when the capacitance values are increased up to roughly 100 to 200 fF. Afterwards there is little or nothing to earn regarding operational speed of the circuit, for an increase in the values of the capacitances. This is common for the current levels of 1 nA, 10 nA or 100 nA. From figure 4.26 it appears that the optimum capacitance value for a minimal fall-time is slightly lower for a V_{dd} of 0.2 V, than for 0.4 V or 0.8 V.

V_{dd} [V]	I_{beq} [nA]	C_n [fF]	t_f[ns]
0.8	1	41	2949
0.8	1	74	914
0.8	1	204	286
0.8	1	409	221
0.8	1	1000	246
0.8	10	41	439
0.8	10	74	162
0.8	10	204	64
0.8	10	409	54
0.8	10	1000	63
0.8	100	41	87
0.8	100	74	41
0.8	100	204	21
0.8	100	409	19
0.8	100	1000	24
0.4	1	41	13417
0.4	1	74	7042
0.4	1	204	3716
0.4	1	409	3379
0.4	1	1000	4171
0.4	10	41	1565
0.4	10	74	874
0.4	10	204	461
0.4	10	409	439
0.4	10	1000	593
0.4	100	41	215
0.4	100	74	132
0.4	100	204	84
0.4	100	409	82
0.4	100	1000	105

Figure 4.22: Fall-time, t_f, as a function of power supply voltage, equilibrium current and capacitance between input and floating gates.

V_{dd} [V]	I_{beq} [nA]	C_n [fF]	t_f[ns]
0.2	1	41	29123
0.2	1	74	19195
0.2	1	204	13886
0.2	1	409	14182
0.2	1	1000	18938
0.2	10	41	3089
0.2	10	74	2124
0.2	10	204	1580
0.2	10	409	1631
0.2	10	1000	2194
0.2	100	41	386
0.2	100	74	271
0.2	100	204	209
0.2	100	409	220
0.2	100	1000	299

Figure 4.23: Fall-time, t_f, as a function of power supply voltage, equilibrium current and capacitance between input and floating gates.

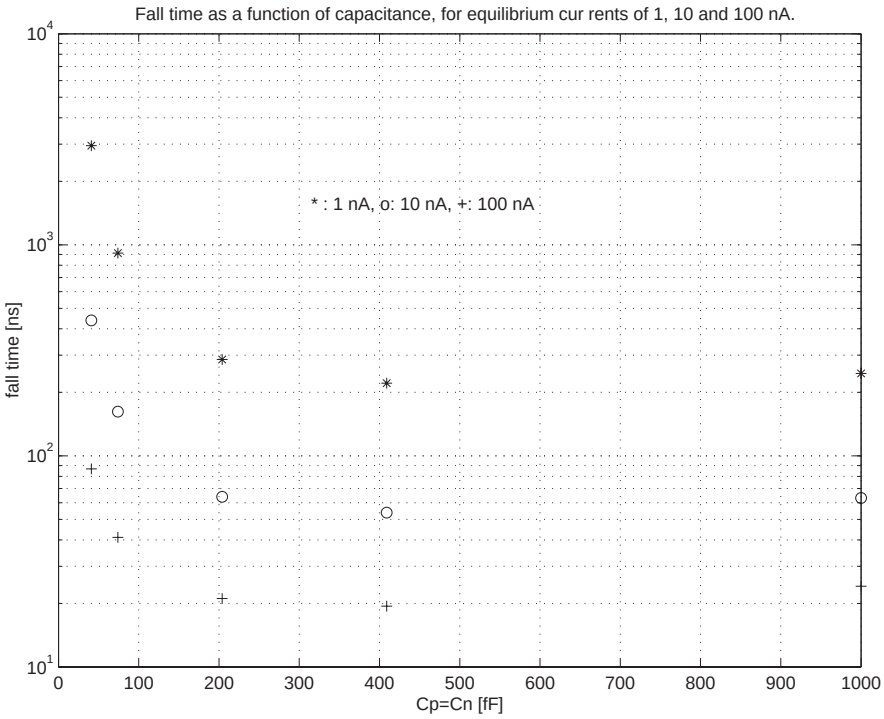

Figure 4.24: Fall time as a function of capacitance, for different equilibrium currents and Vdd=0.8 V.

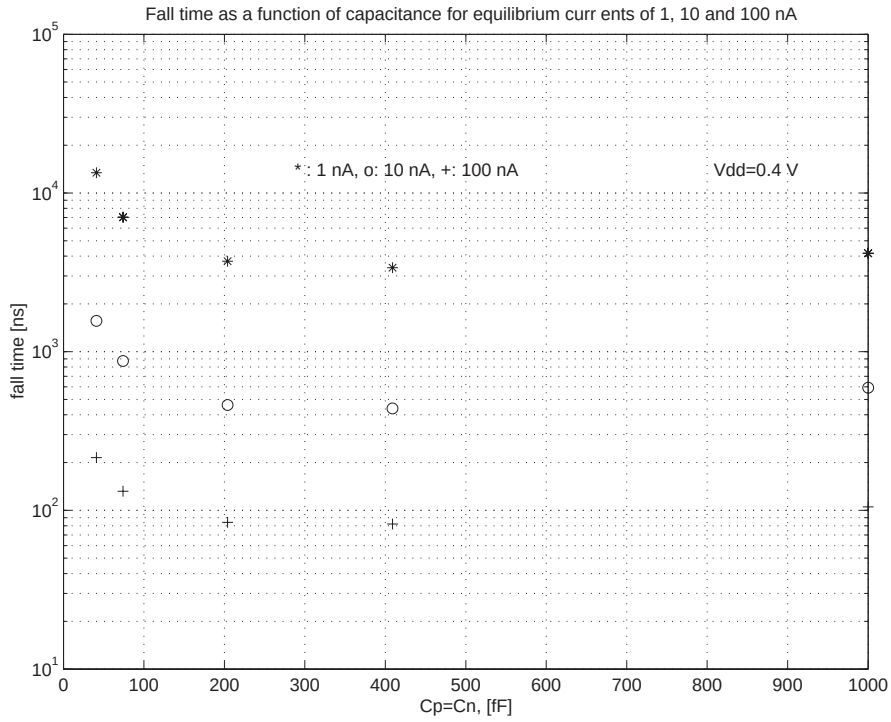

Figure 4.25: Fall time as a function of capacitance, for different equilibrium currents and Vdd=0.4 V.

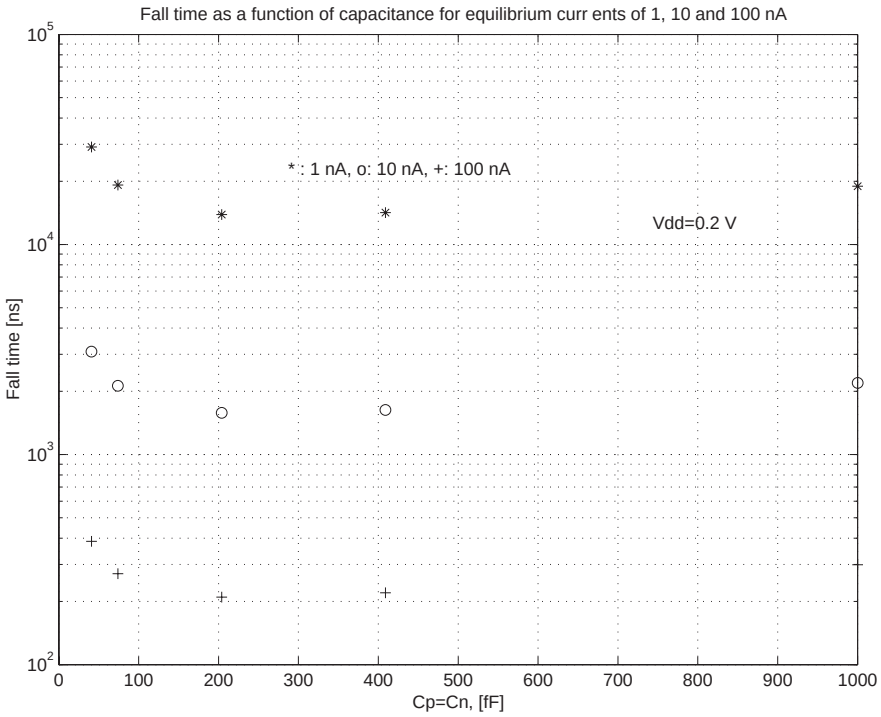

Figure 4.26: Fall time as a function of capacitance, for different equilibrium currents and Vdd=0.2 V.

4.5 Inverter discussion

Measurements have shown that the inverters were functional. It was possible to UV-program the inverters initially for a V_{dd} of 0.8 V and then adjust it in steps down to 95 mV and still maintain voltage gain $\leq$ -1, about the switching point of $V_{dd}/2$. The lowest V_{dd} measured here while maintaining the above mentioned gain was 93 mV [AuBe01a].

The reprogrammability of these inverters, showing that they can be programmed for different current levels, is illustrated in figure 4.18.

The DC-leakage current is minimum in the 10 nA range (figure 4.18). DC-leakage currents should be reduced only if the circuits are doing no work, according to [BuCh96]. If there should ever happen to be very many FGUVMOS transistors on the same die, there might be reasons to reduce these currents. In [BeLa99a] the lowest DC currents are down in the pA to 10 pA range.

Other published inverter measurement results are published in figure 4.27. These data from 2001 and onwards were produced using a 0.6 μm CMOS technology, while the previous data came from a similar technology, from AMS, with 0.8 μm minimum gate length.

V_{dd} [V]	I_{beq} [nA]	$Gain$, V_{out}/V_{in}	I_{beq} variation [nA]	$V_{dd_{min}}$[V]	reference
0.8	600	-	-	0.8	[BeWi97]
0.5	-	-	-	0.5	[BeWi98]
0.3	4	-	4 - 200	0.3	[BeLa99a]
0.8	90	-	90-1700	0.3	[Gund00]
0.8	200	-15	-	0.3	[Bahr01]
0.3	5	-9	3-500	0.3	[Bahr01]
0.8	5500	-10	1.3k-5.5k	0.3	[Dani01]
0.3	17	-6	17-122	0.3	[Dani01]
0.8	1000	-		0.8	[Flat01]
0.8	1500	-	1.5k-10k	0.093	[AuBe01a]
0.2	600	-	-	0.2	[AuBe01c]
0.8	100	-4.2	-	0.8	[Jens02]
0.8	2000	-15	-	0.8	[Loms02]

Figure 4.27: Some published results for FGUVMOS inverters.

The measured data presented here may be regarded as typical compared

to other work (figure 4.27).

As seen from the table (figure 4.27) the I_{beq} range is in the nA to μA range. Seen from the same table, the voltage gain is relatively low compared to most classic amplifiers, with a maximum reported gain of -15.

Variation in the I_{beq} has been shown to be up to 50 times for one and the same element for a given V_{dd} [BeLa99a]. The inherent possibilities for I_{beq} variation might have been larger in most cases, but might not have been tested out due to relatively time-consuming UV-programming procedures from this test environment. To test out some circuit concepts quicker, a method used in [Loms02] may be used. With the method briefly described, a pair of dedicated additional NMOS charging transistors are used to place the voltage of the floating gates at a desired level, however unwanted leakage from the floating gates may occur by using this scheme, according to [Loms02]. A similar programming scheme is published in [HøWi01].

The minimum V_{dd} required to obtain a certain maximum gain for an inverter is given by [Sven97]:

$$V_{dd} = 2\frac{kT}{q} ln(1 + nG_{max}) \tag{4.1}$$

The parameters here are in accordance with those used in earlier treatment of single transistors. G_{max} is the absolute value of the maximum DC gain of the inverter $|dV_{out}/dV_{in}|$, which must be well above 1 to have a good inverter function, according to [Sven97].

The absolute lower bound for the supply voltage, based on idealized transistors is then, at room temperature

$$V_{dd} = 2\frac{kT}{q} ln(1 + 1 \cdot 1) = 2\frac{kT}{q} ln2 = 36mV \tag{4.2}$$

This number of 36 mV is in accordance with the absolute lower bound for V_{dd}, not achievable by any technology, calculated in [ScPi96]. According to [Sven97] one can argue that in terms of the steepest voltage dependence, the subthreshold characteristic is the best that can be obtained in an electronic device at a given temperature, with n=1. The limit is very general and applies to bipolar, HEMT and other technologies as well [Sven97]. With a more realistic n=1.5 [Sven97] the minimum V_{dd} under the same conditions as the above equation would be 47 mV. Minimum V_{dd}s for an inverter, as a function of gain and slope factors, n, of 1, 1.5 and 2 are shown in figure 4.28.

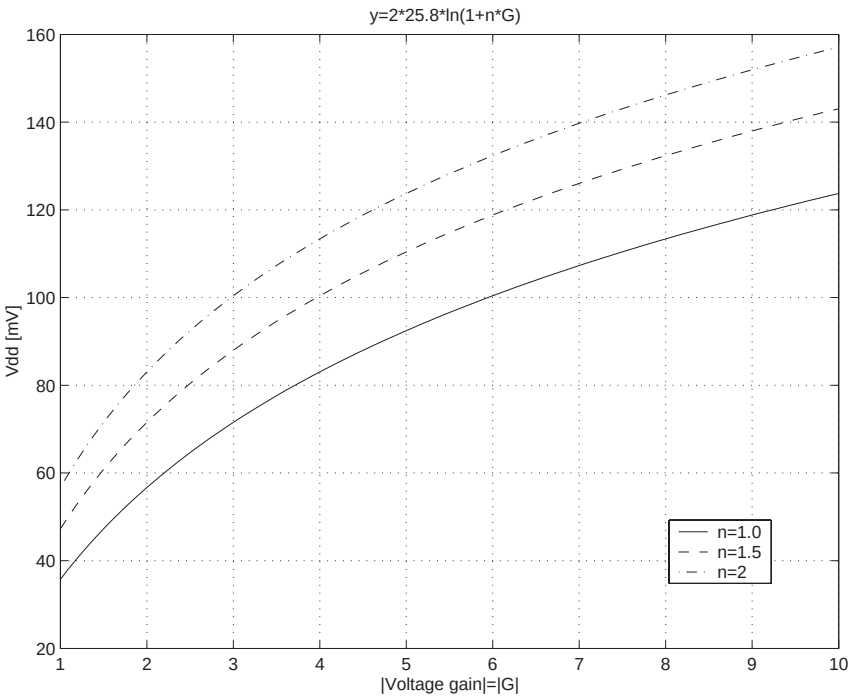

Figure 4.28: Minimum theoretical power supply voltage at room temperature, as a function of the absolute value of the voltage gain, for a standard inverter. Of the three values, n=1.5 is the most typical for CMOS.

To make a quick comparison between measured data and what could be expected from measurements, we can try to develop a simple model directly from the equation for minimum V_{dd}. For the FGUVMOS circuit and a C_n of 73.6 fF, about 53% of the signal would be expected to get through to the floating gate, according to chapter 2, while 100% would be the case for a standard CMOS inverter. Then the value counted from the the simple model should be multiplied by (100/53) in this case, and the n=1/κ=1.5 be used. This gives a minimum V_{dd} for a gain of -1 of:

$$V_{dd} = \frac{100}{53} 2 \frac{kT}{q} ln(1 + 1.5 \cdot 1) = 89mV \tag{4.3}$$

The minimum gain of -1 found by measurements were 93 mV, as compared to the 89 mV calculated. Increasing the size of capacitances between the input and floating gates would allow a larger percentage of the signal

on the inputs to pass through to the floating gates. If the simple model in figure 2.7 holds, an increase of C_n of 10 times would increase the percentage of 53 to more than 90. In that case the minimum V_{dd} based on the assumptions above might be reduced to

$$V_{dd} = \frac{100}{90} 2 \frac{kT}{q} ln(1 + 1.5 \cdot 1) = 2 \frac{kT}{q} ln2 = 53mV \tag{4.4}$$

These models are simple, and only measurements should be regarded as proofs. Anyway, it might not be impossible to run FGUVMOS inverters with a supply voltage down to about 50 mW, while voltage gain from input to output still remains $\leq$ -1. In theory, that could be enough of a V_{dd} to run ring-oscillators, for example. In practice there could be problems regarding mismatch, noise and other phenomena, not considered here. The lower voltage bound for the FGUVMOS inverters might be close to the fundamental limits for the technology, which is around 47 mV at room temperature due to limitations in the gate voltage's ability to control the voltage in the channel, and the "ultimate" 36 mV limit, not likely to be reached by any technology [Sven97].

The inverters measured here all had the same W/L-ratio and drawn capacitances between input and floating gates of 73.6 fF. Gain could be increased by increasing the size of these capacitances, as illustrated in figure 2.18, thereby increasing the transconductance seen from the input as well as the output resistance. Other parameters that can be important are listed in 2.16. Transconductance and output resistance determines the voltage gain, and the lower limit for the power supply voltage, V_{dd}, which affects most static and dynamic parameters. To enhance important static and dynamic parameters of the circuits the capacitances of 73.6 fF could be increased.

A complete UV-programming of an FGUVMOS inverter could take hours, to our experience. It has been shown that settling of the UV-programming process, for an inverter, have taken up to about 24 hours, reported in [BeWi97]. UV-programming of minimum-sized donut transistors in [Dani01] took up to 4 hours [Flat01], while very similar transistors were programmed for 15 to 20 minutes for results reported in [Flat01]. This can partly come from random charge left on the floating gates after production. The UV-programming for the works in [Dani01] and [Bahr01] was done over a period of about 6 months. The measurements serving as a background for this work were done during three or four visits to the University of Oslo, lasting two to three weeks alltogether. Making similar measurements for standard circuitry could in many cases probably have been done using only a few hours. To speed up the process the layouts could

be optimized [BeLa01b]. It is also expected to take significantly less time with more modern CMOS processes, since the programming time is said to be proportional to the floating-gate capacitance [BeLa01b]. The UV-source energy density is also important, as the UV-activeted conductances varies nearly linearly with it [BeKe93]. Distance to the chip and strength of the UV-source are thus among factors that determine the programming time.

The low-voltage potential has been demonstrated, and might be compared to other low-voltage approaches than FGUVMOS, reported. CMOS chips working at 0.2 V supply voltage have been reported in [Sven97], while an inverter functioning at a V_{dd} of 70 mV at room temperature has been made [Burr95]. An inverter optimized for such experiments, especially by having larger drawn capacitances, could probably work with a significantly lower V_{dd} than the 93 mV proved by measurements in this work, as has been discussed.

Chapter 5

2-MOSFET floating-gate elements with identically weighted inputs to both PMOS and NMOS; "PMNM"

5.1 Implementation and layout of "PMNM" elements

5.1.1 P3N3 element

All capacitances shown in figure 5.1 have the same values. Starting again with

$$I_{ds,p} = I_{beq} \prod_{j=1}^{l} exp\{\frac{1}{nU_t}(V_{dd}/2 - V_j)k_j\} \tag{5.1}$$

$$I_{ds,n} = I_{beq} \prod_{i=1}^{m} exp\{\frac{1}{nU_t}(V_i - V_{dd}/2)k_i\} \tag{5.2}$$

, and using the same approach as for figure 2.23 results in the truth table in figure 5.2. Here

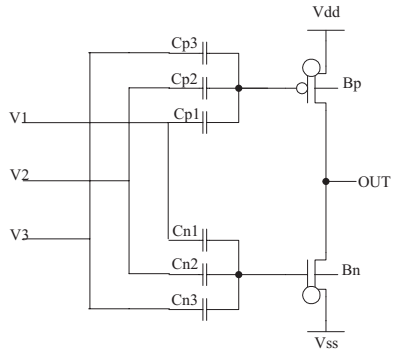

Figure 5.1: Schematic for real time reconfigurable floating-gate circuit with 3 equally weighted inputs to both PMOS and NMOS (P3N3).

$$e_p = (\frac{V_{dd}}{2} - \frac{1}{3}V_x - \frac{1}{3}V_y - \frac{1}{3}V_z) \tag{5.3}$$

$$e_n = (\frac{1}{3}V_x + \frac{1}{3}V_y + \frac{1}{3}V_z - \frac{V_{dd}}{2}). \tag{5.4}$$

Simplifying the table, counting 1s in the binary input vector gives figure 5.3. This circuit does not have the possibility to change its function in real time unless one or more of the ordinary inputs, V_x,V_y,V_z are used as control inputs. The circuit can compute NAND2, NOR2 and INVERT in addition to CARRY' for a FULL-ADDER [AuBe01b]. This is seen from figure 5.2. If one of the inputs is tied to 0, the circuit will compute the 2-input NAND function of the two inputs left. If one input is held at 1, the circuit is able to compute the 2-input NOR function.

X	Y	Z	e_p	e_n	OUT
0	0	0	$3V_{dd}/6$	$-3V_{dd}/6$	1
0	0	1	$V_{dd}/6$	$-V_{dd}/6$	1
0	1	0	$V_{dd}/6$	$-V_{dd}/6$	1
0	1	1	$-V_{dd}/6$	$V_{dd}/6$	0
1	0	0	$V_{dd}/6$	$-V_{dd}/6$	1
1	0	1	$-V_{dd}/6$	$V_{dd}/6$	0
1	1	0	$-V_{dd}/6$	$V_{dd}/6$	0
1	1	1	$-3V_{dd}/6$	$3V_{dd}/6$	0

Figure 5.2: The table shows parts of the exponentials, e_p, e_n, and output values, for all possible binary values of inputs X,Y,Z. (X=1 if, and only if, $V_x = V_{dd}$.)

number of "1"'s	e_p	e_n	OUT
0	0	$-3V_{dd}/6$	1
1	0	$-V_{dd}/6$	1
2	0	$V_{dd}/6$	0
3	0	$3V_{dd}/6$	0

Figure 5.3: Truth table for circuit with same [X,Y,Z] binary inputs coupled to both PMOS and NMOS.

5.1.2 P5N5 element

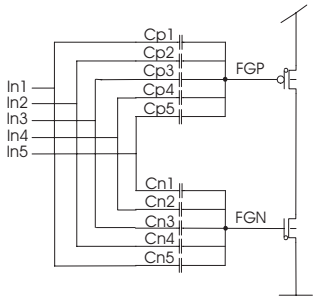

Figure 5.4: 5 P/N-input universal element. All capacitances in the schematic are of equal size.

Using the same approach for the circuit in figure 5.4 as for P3N3, and a binary input vector [U,V,X,Y,Z] leads to:

$$I_{ds,p} = I_{beq} exp\{\frac{1}{nU_t}(\frac{V_{dd}}{2} - \frac{1}{5}V_u - \frac{1}{5}V_v - \frac{1}{5}V_x - \frac{1}{5}V_y - \frac{1}{5}V_z)\} \tag{5.5}$$

$$I_{ds,n} = I_{beq} exp\{\frac{1}{nU_t}(\frac{1}{5}V_u + \frac{1}{5}V_v + \frac{1}{5}V_x + \frac{1}{5}V_y + \frac{1}{5}V_z - \frac{V_{dd}}{2})\} \tag{5.6}$$

and

$$e_p = (\frac{V_{dd}}{2} - \frac{1}{5}V_u - \frac{1}{5}V_v - \frac{1}{5}V_x - \frac{1}{5}V_y - \frac{1}{5}V_z) \tag{5.7}$$

$$e_n = (\frac{1}{5}V_u + \frac{1}{5}V_v + \frac{1}{5}V_x + \frac{1}{5}V_y + \frac{1}{5}V_z - \frac{V_{dd}}{2}). \tag{5.8}$$

This circuit ideally produces a low output if, and only if, there are 3 or more binary inputs at high level. In [AuBe01a], inspired by [KoSh92], two of the inputs were treated as always having the same input voltage. When two of the inputs are connected together and the resulting input denoted "W", as in figure 5.6, e_p and e_p changes to:

number of "1"'s	e_p	e_n	OUT
0	$5V_{dd}/10$	-$5V_{dd}/10$	1
1	$3V_{dd}/10$	-$3V_{dd}/10$	1
2	$V_{dd}/10$	-$V_{dd}/10$	1
3	-$V_{dd}/10$	$V_{dd}/10$	0
4	-$3V_{dd}/10$	$3V_{dd}/10$	0
5	-$5V_{dd}/10$	$5V_{dd}/10$	0

Figure 5.5: The table shows parts of the exponentials, e_p, e_n, and output values as a function of number of "1"'s for the binary inputs U,V,X,Y,Z for P5N5

$$e_p = (\frac{V_{dd}}{2} - \frac{2}{5}V_w - \frac{1}{5}V_x - \frac{1}{5}V_y - \frac{1}{5}V_z) \tag{5.9}$$

$$e_n = (\frac{2}{5}V_w + \frac{1}{5}V_x + \frac{1}{5}V_y + \frac{1}{5}V_z - \frac{V_{dd}}{2}). \tag{5.10}$$

Such a circuit is shown in figure 5.6. If W is used as a control signal, having twice the capacitive weight of each of X, Y, Z, which are treated as regular inputs, the following table in figure 5.7 can be made [AuBe02a].

Here the control input W was not restricted to Boolean values, making this variant of the P5N5 circuit being able to implement CARRY' as well. Using two similar inputs, as in figure 5.4, would provide the same functionality if the first input of the two was tied to V_{dd} and the second to V_{ss}.

The table in figure 5.8 lists digital functionality for the P5N5 circuit. Functionality is demonstrated by simulation in figure 5.9.

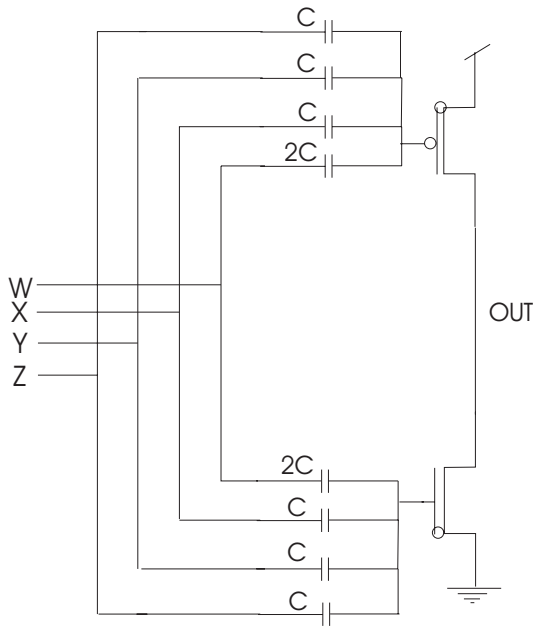

Figure 5.6: P5N5 element used as in [AuBe02a].

W	number of "1"'s	e_p	e_n	OUT
V_{dd}	0	$V_{dd}/10$	-$V_{dd}/10$	1
V_{dd}	1	-$V_{dd}/10$	$V_{dd}/10$	0
V_{dd}	2	-$3V_{dd}/10$	$3V_{dd}/10$	0
V_{dd}	3	-$5V_{dd}/10$	$5V_{dd}/10$	0
$V_{dd}/2$	0	$3V_{dd}/10$	-$3V_{dd}/10$	1
$V_{dd}/2$	1	$V_{dd}/10$	-$V_{dd}/10$	1
$V_{dd}/2$	2	-$V_{dd}/10$	$V_{dd}/10$	0
$V_{dd}/2$	3	-$3V_{dd}/10$	$3V_{dd}/10$	0
0	0	$5V_{dd}/10$	-$5V_{dd}/10$	1
0	1	$3V_{dd}/10$	-$3V_{dd}/10$	1
0	2	$V_{dd}/10$	-$V_{dd}/10$	1
0	3	-$V_{dd}/10$	$V_{dd}/10$	0

Figure 5.7: The table shows part of the exponentials, e_p, e_n, and output values, for different values of W, and different numbers of "1s" on ordinary inputs X,Y,Z. Lines 2-5 show the 3-input NOR function, lines 6-9 the CARRY'-, and lines 10-13 the NAND-function.

W	digital functionality
V_{dd}	NOR3, NOR2, INVERT
$V_{dd}/2$	CARRY', NOR2, NAND2, INVERT
$V_{ss} = 0$	NAND3, NAND2, INVERT

Figure 5.8: The table shows some functionality implemented by the circuit in figure 5.6, as a function of the control input, W.

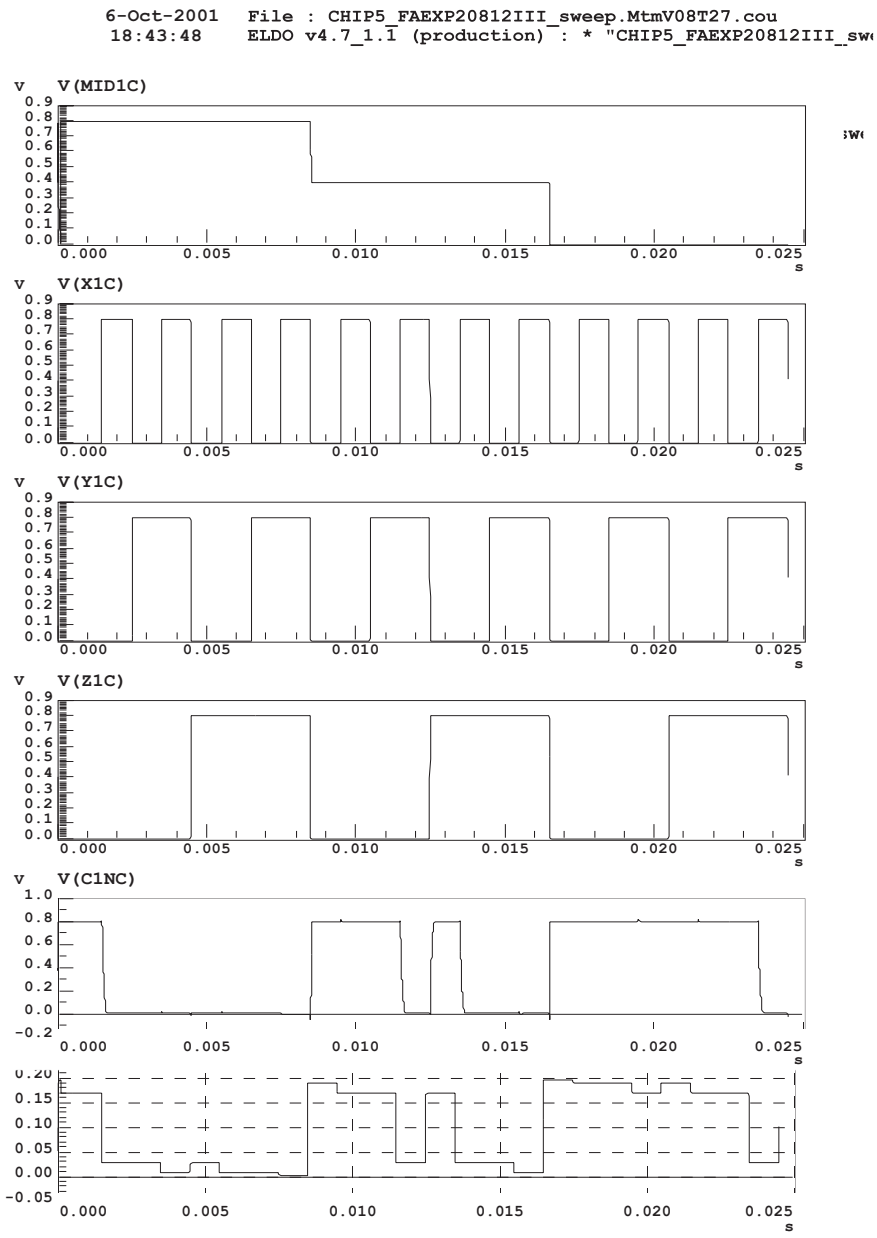

Figure 5.9: The six signals, from top to bottom, correspond to "W","X","Y","Z","OUT" and "OUT" in figure 5.6. The fifth and sixth signals are the outputs for a V_{dd} of 0.8 V and 0.2 V respectively [AuBe02a].

5.1.3 "PMNM" layout

For layout of the P3N3 and P5N5 elements the 20.8 μm/ 1.2 μm MOSFETs described earlier were used. All capacitances were dimensioned for values of 25.54 fF, and the structures can be seen in figure 5.10. While threshold voltage mismatch can be compensated by adjusting the floating-gate charge, mismatch in control-gate capacitances can not [Minc00].

A design should not be too dependent on absolute values of components, and relative matching are far better than absolute accuracies for CMOS technology. Also, the larger the dimensions, the better the relative accuracy that can be achieved [LaSa94]. A special practical limit that must be taken into account for layout of floating-gate circuits exists for the AMS 0.6 process [AMS9X]. For this process the ratio of the "passive" poly to the "active" "gatepoly" could not exceed a certain limit, imposing a maximum size of the capacitances for a structure like P5N5 for example. During this work the size of certain capacitances had to be reduced not to break this limit.

Capacitance values at the end of arrays tend to be systematically different from the rest, as found in [Minc00]. Therefore dummy structures were used for P3N3. This can be observed from figure 5.10 since there are no vertical metal lines connecting the dummy capacitances to the horizontal signal wires lying between the PMOS and NMOS transistors. The reason that dummy capacitances were not used for P5N5 was that the special design rule mentioned above did not allow it if the other capacitances (not considering parasitics) were not to be reduced.

A layout using minimum transistors, W=0.8μm and L=0.6μm, from the 2nd chip sent for processing via EuroPractice in spring 2001 [AuBe01e] is shown in figure 5.12. Dummy capacitances were used for this P5N5 circuit. The area for one such element equals approximately 36um x 13um.

P5N5 elements with W=1.2μm and L=1.2μm were implemented on a 3rd chip sent for processing in summer 2001, depicted in figure 5.11. They have been used for simulations presented later in this chapter, and in [AuBe01d], [AuBe01e].

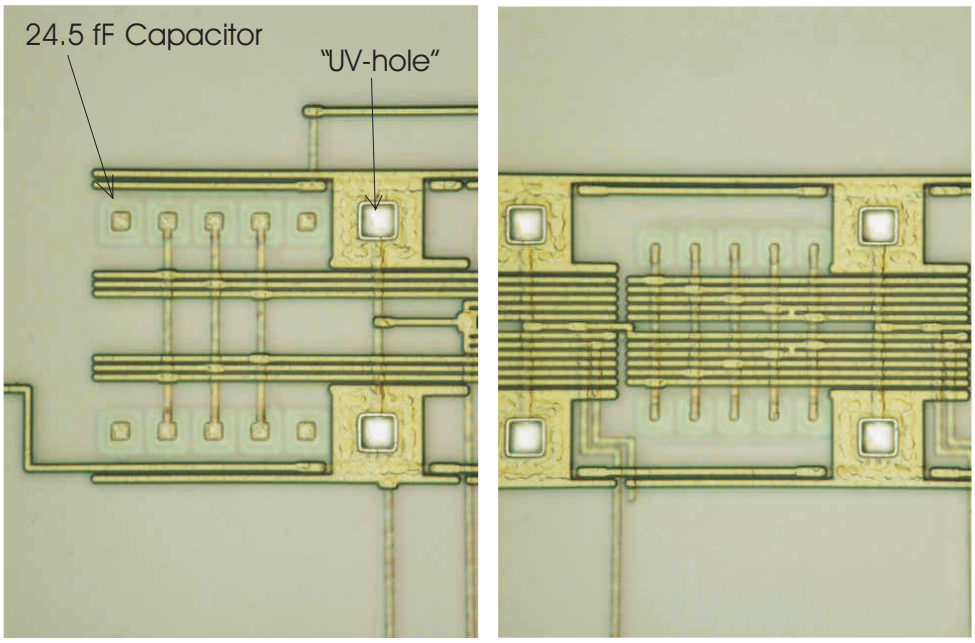

Figure 5.10: P3N3 to the left, and P5N5 to the right.

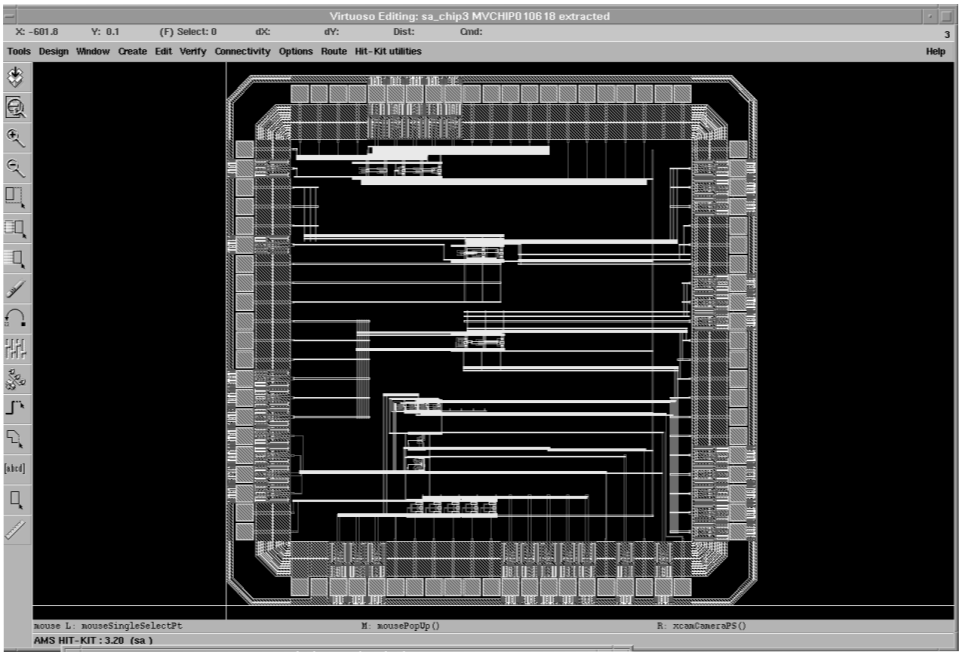

Figure 5.11: 3rd prototype chip sent for processing summer 2001.

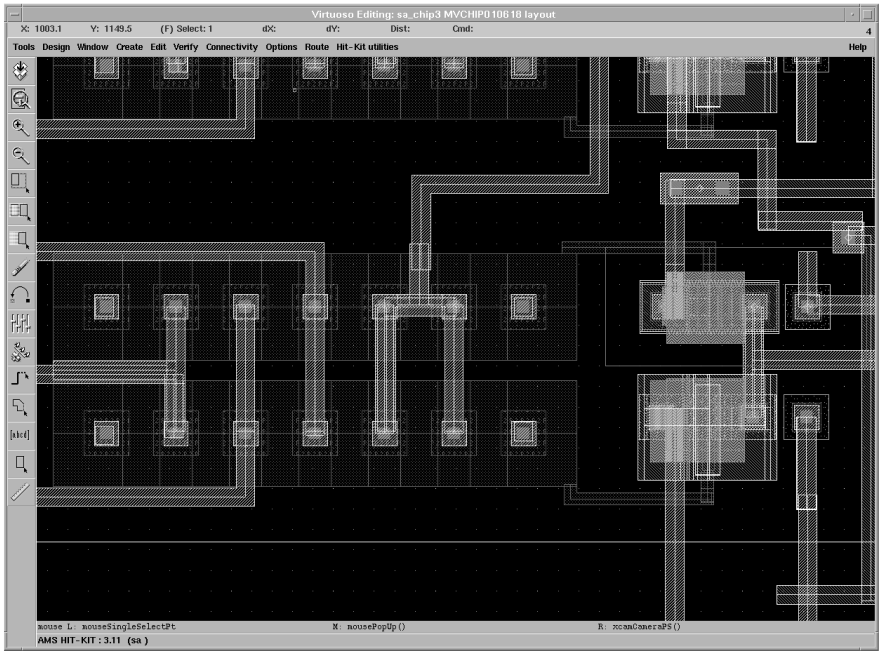

Figure 5.12: Layout for P5N5 universal element. Removing passivation at certain areas and making metal shields make UV-light hit selected parts of the chip surface, while UV-programming after fabrication of chips. UV-programming might be repeated or changed.

5.2 UV-programming and test setup of P5N5 element

The most important goal when doing the measurements was to try to demonstrate the different inherent functionalities of the structures. For the P5N5 element this was the CARRY', NAND3, NAND2, NOR3, NOR2 and INVERT functions.

The UV-programming and testing were done basically in the same manner as previously described for the inverters. The increased number of inputs compared to the inverter made it necessary to bring in some additional measurement instruments, as sketched in figure 5.14. The script in figure B.7 was used to control the computerized instruments setup. The "FA-EXP20812III" circuit (figure 5.14) has node names which can be found in the netlist "$CHIP5\_capall\_netlist$" [Aune02].

The table in figure 5.17 has data related to UV-programming and measurements presented later in figures 5.18, 5.19, 5.20 and 5.21. V_p played the same role as V_+ in figure 4.8, which means that it was the voltage applied to what was V_{dd} under normal operation. Similarly V_m equalled V_-. "ProgWell" (figure 5.17) was the voltage applied to the nwell during UV-exposure, while "Well" denoted the voltage level on the nwell when the circuit was in normal, computing mode.

More details can be found in appendix A.2 and A.3.

"W" was the control input (which was named mid1 in the netlist, $CHIP5\_capall\_netlist$ [Aune02]), and "y" and "z" were two of the inputs to the P5N5 element, which can be seen in figure 5.15.

The inverter function was measured. All inputs were short-circuited in this case, and the "x" input then equalled W=y=z. The voltage x was changed in steps from 0 V=V_{ss} to 0.8 V=V_{dd}. "Vstart" and "Vstop" were the maximum "high" and "low" voltages respectively.

Data regarding UV-programming and measurements for the NAND3, NOR3 and CARRY' functions can be found in a similar way from figure 5.17.

Symmetric Booelan functions are functions for which the output depends on the number of 1s in the input vector irrespective of their position [Boho98]. The transitions between 1 and 0 on the output were what was measured for the CARRY', NAND3, NOR3 and INVERT functions.

During measurements it was discovered that when the circuit was in normal computational mode, the source of the NMOS sometimes had a potential of hundreds of mV, instead of 0 V. This disturbed the measurements. When the current measuring instrument was removed this phe-

nomenon seemed to vanish. For later measurements the instrument was removed, as in figure 6.6. The instrument used for measuring current was a Keithley 617. Recently certain problems regarding this instrument used as an ampere meter was reported, among other useful hints regarding instrumentation for FGUVMOS circuits [Gjer02]. It was pointed out that polarity had to be chosen correctly when using the instrument for current measurements.

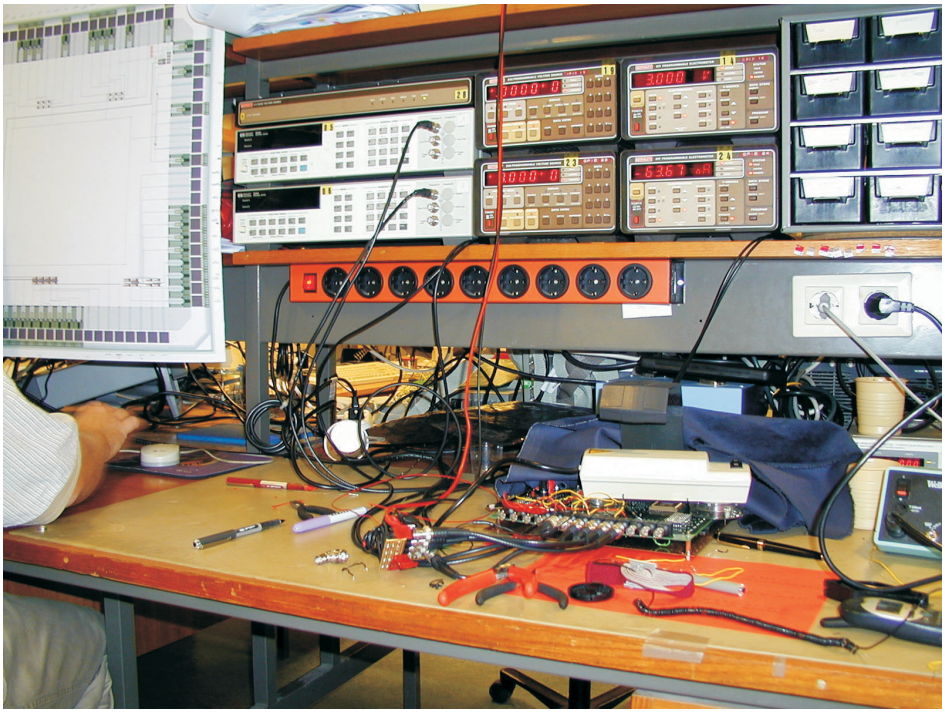

Figure 5.13: Laboratory environment at Ifi, University of Oslo, while testing P5N5 and P1N3 circuits.

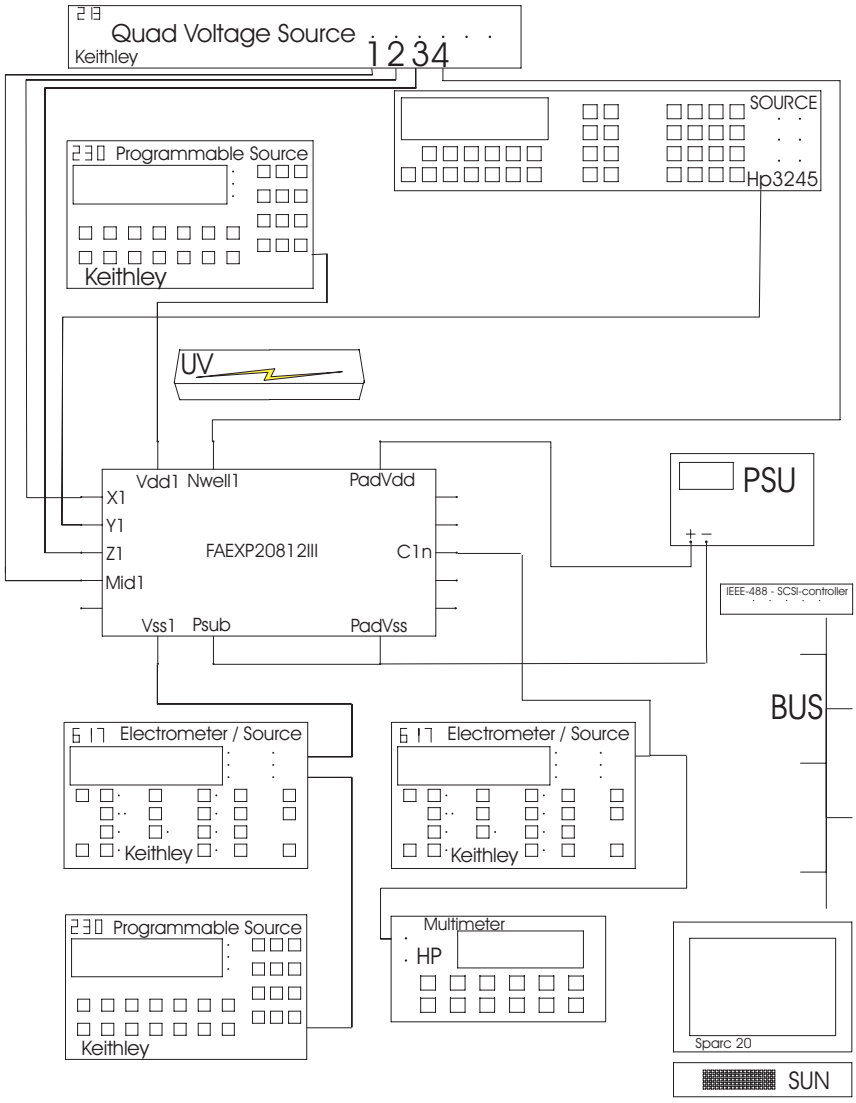

Figure 5.14: Test setup. Ground / zero voltage potential for the sources and measurement equipment were wired together with "Psub" and "PadVss" on the chip.

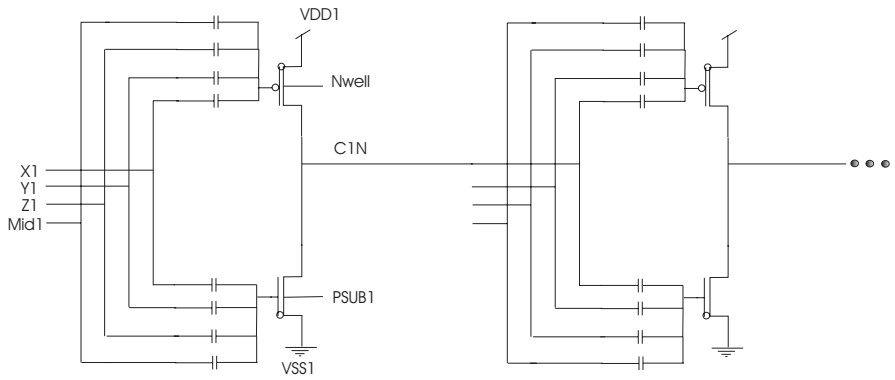

Figure 5.15: FAEXP20812III includes four P5N5 elements in a row. The nodes shown are relevant for 5.14 and in $CHIP5\_capall\_netlist$ [Aune02].

5.3 P5N5 results

5.3.1 Simulation results

A P5N5 element using W=L=1.2μ m, I_{beq}=10 nA, temperature 27 degrees C, and V_{dd}=200 mV has been simulated. Results are shown in figure 5.16.

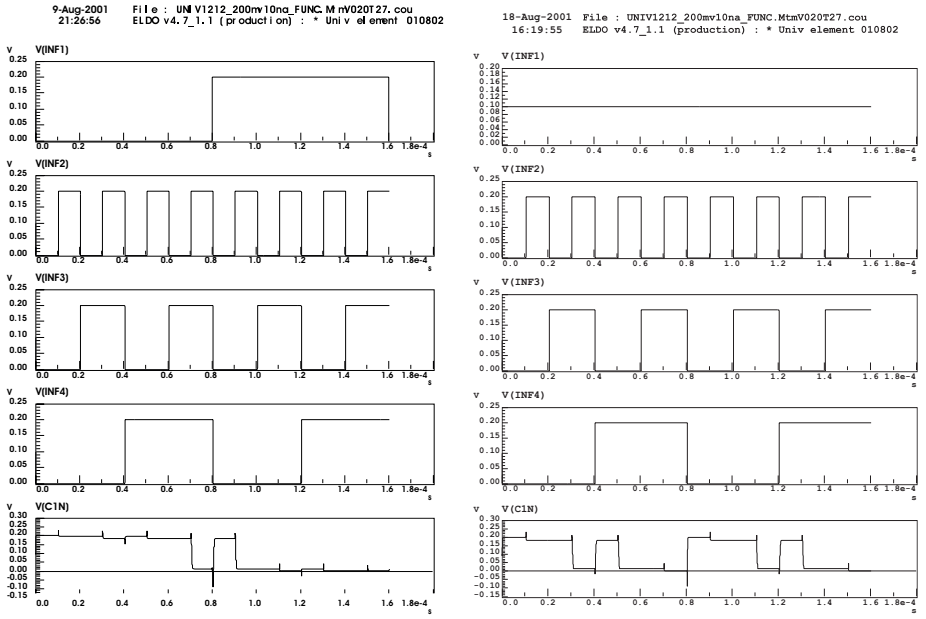

Figure 5.16: Simulating our circuit for NAND, NOR and CARRY' functions [AuBe01e]. From the start of the left simulation trace V(INF1) / W="0", making it realize the NAND function of the 3 inputs INF2, INF3, INF4. Around midway through the V(INF1) goes to "1", making it implement a NOR function. CARRY' is shown to the right. On the bottom: OUT="V(c1n)".

5.3.2 Measurement results

Both measurements and simulations have been done to compare the results for the P5N5 circuit configured as an inverter. As can be seen in figure 5.18, the voltage gain was lower according to the measurements than it

Vp [V]	Vm [V]	ProgWell [V]
1.8	0.4	0.4
Vdd [V]	Well [V]	I_{beq} [nA]
0.8	2.8	49.7
W [V]	y [V]	z[V]
x	x	x
Vstart [V]	Vstop [V]	Function
0.799	0.114	INVERT

Vp [V]	Vm [V]	ProgWell [V]
1.8	0.4	0.4
Vdd [V]	Well [V]	I_{beq} [nA]
0.8	2.8	43.4
W [V]	y [V]	z[V]
0	1	1
Vstart [V]	Vstop [V]	Function
0.796	0.117	NAND3

Vp [V]	Vm [V]	ProgWell [V]
1.8	0.4	0.4
Vdd [V]	Well [V]	I_{beq} [nA]
0.8	3.0	16.7
W [V]	y [V]	z[V]
0.8	0	0
Vstart [V]	Vstop [V]	Function
0.786	0.134	NOR3

Vp [V]	Vm [V]	ProgWell [V]
1.4	1.2	0.9
Vdd [V]	Well [V]	I_{beq} [nA]
0.8	0.8	33.8
W [V]	y [V]	z[V]
0.7	0	0.8
Vstart [V]	Vstop [V]	
0.787	0.032	CARRY'

Figure 5.17: Different applied voltages for UV-programming and testing for inverters on the 84 pin chip.

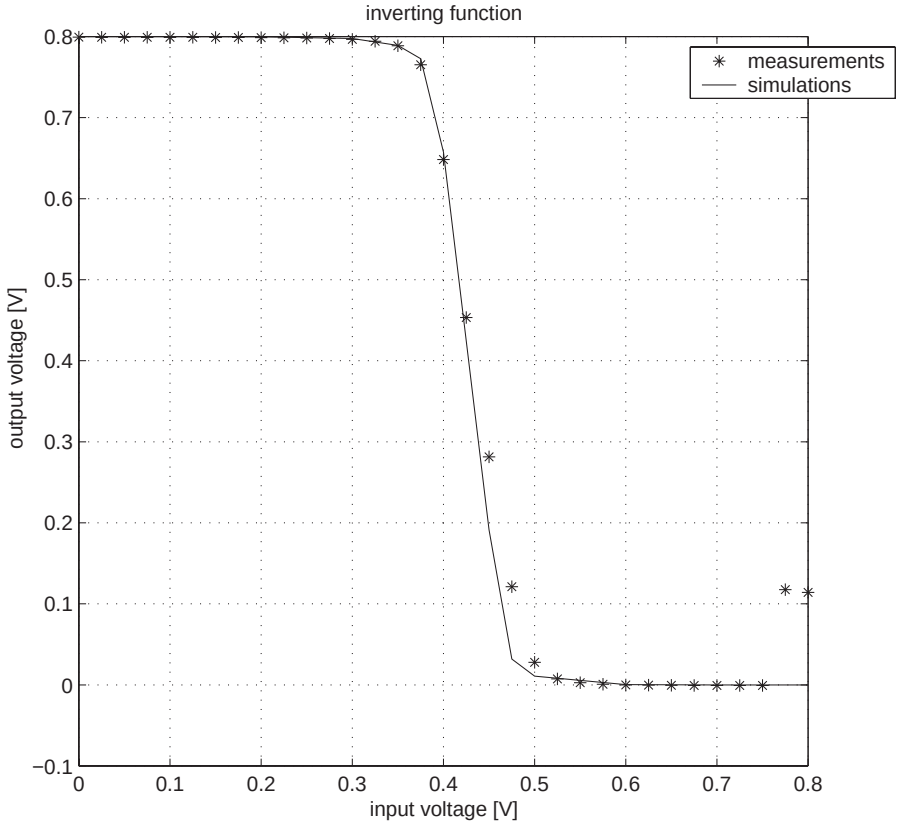

Figure 5.18: Measurements and simulations of output voltage as a function of input voltage, for a P5N5 circuit included in the "FAEXP20812III" circuitry.

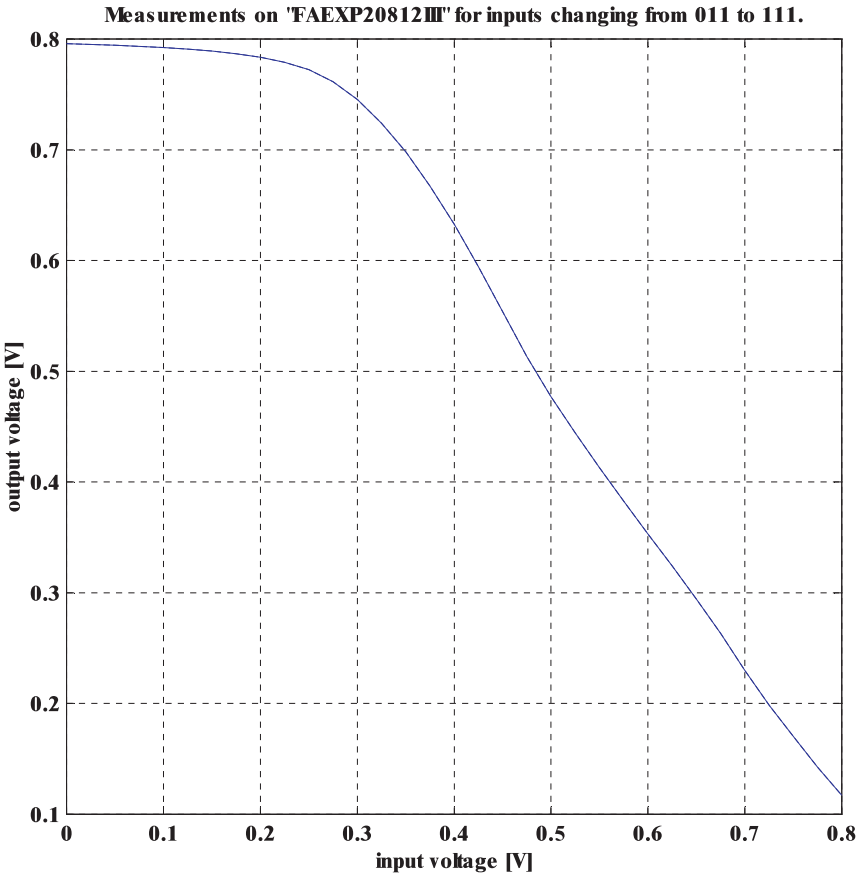

Figure 5.19: The measurements are made on a P5N5 circuit while it was operating as a 3-INPUT NAND [AuBe02a].

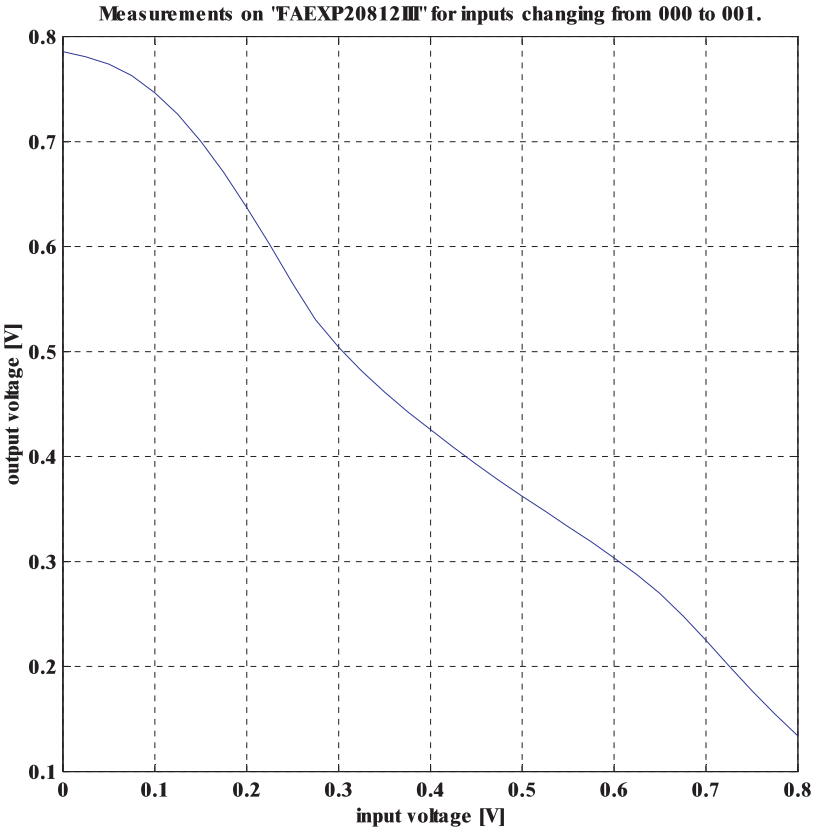

Figure 5.20: The measurements are made on a P5N5 circuit while it was operating as a 3-INPUT NOR [AuBe02a].

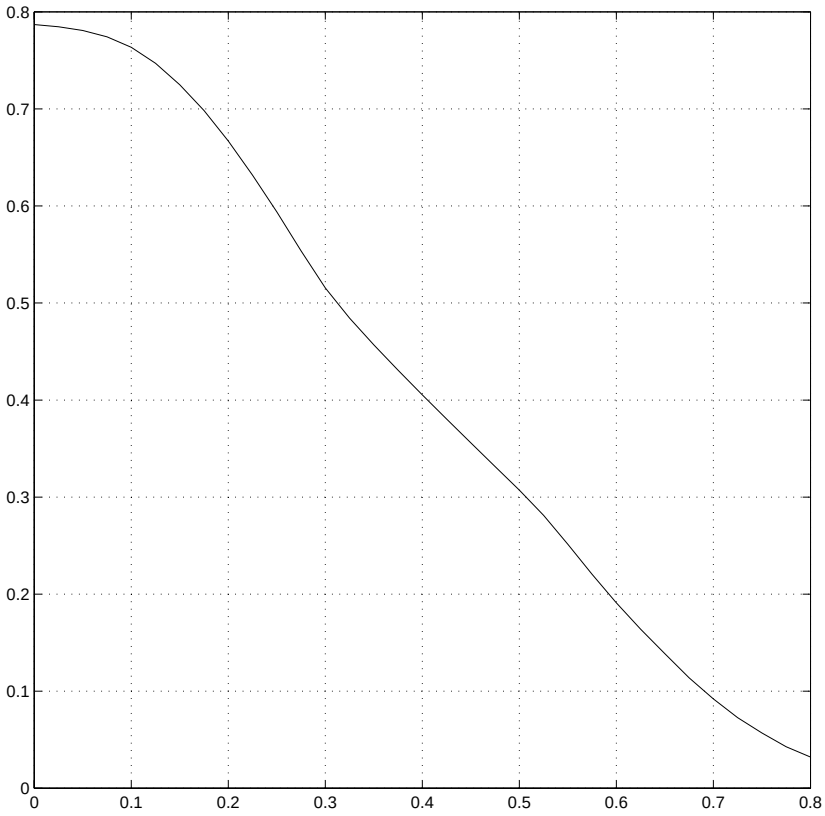

Figure 5.21: The measurements are made on a P5N5 circuit while it was operating as a INVERTED CARRY . "FAEXP20812III" is a structure on chip containing P5N5 elements [AuBe02a].

was according to the simulations. There was also an irregularity for the measurements when the input voltage came close to V_{dd}, as the output jumped up to above 0.1 V instead of further approaching V_{ss}. When the input voltage was 0.8 V, the measured output voltage was at 0.114 V (figure 5.17). Measurements and simulations of the current through the inverter, done at the same time as data for figure 5.18 were produced, can be found in 2.14. The temperature for the Spice simulation was 25.8 degrees C. The measured current through the source of the NMOS when the input voltage was low was 130 pA, and 160 pA when the input voltage was 0.8 V.

The P5N5 circuit configured as 3-input NAND circuit should produce a low level on its output if, and only if, all Boolean inputs were high. To measure this the control input, "W" was 0 V, while two of the other inputs were high. The third input was swept between low and high while the output value was being measured. Figure 5.19 shows an example from such a measurement. For input voltages of 0 V and 0.8 V, the " high" and "low" output voltages were 0.796 and 0.117, respectively (figure 5.17).

To do measurements for the NOR3 function the control signal at "W" was high, as in the simulation in figure 5.9, while two ordinary inputs were low and the third was swept between low and high. These measurements were made according to figure 5.17, with maximum measured high level of 0.786 V and minimum low level of 0.134 V.

Measurements for the CARRY' function were also made using, and producing, data in figure 5.17. When the number of high level Boolean inputs grew from one to two, the output went down from 0.787 V to 0.032 V, for a V_{dd} of 0.8 V.

5.4 PMNM discussion

Layout-based simulations have demonstrated the circuit working at $V_{dd} = 180mV$ [AuBe01d]. V_{dd}=200 mV simulations are shown in figure 5.16.

The P5N5 circuit's ability to function as an inverter has been proven. In figure 5.18 results from both measurements and simulations are shown together. The voltage gain was slightly less from measurements than from simulation. The output voltage also jumped up a bit for an input voltage close to V_{dd}. Currents through the element behaving as an inverter, corresponding to figure 5.18, are shown in figure 2.14.

Production spread lessens the chances for identical simulations and measurements results. In addition the simulator as well as the parameters from the factory might not be 100% correct. A similar comparison was done in [AuBe01c], which showed measurement results for output voltage as a function of input voltage falling between outer limits for production spread, when it was taken into the account in the simulation.

V_{dd} [V]	I_{max} [nA]	I_{O_H}[nA]	I_{O_L}[nA]	$\frac{2I_{max}}{I_{O_L}+I_{O_H}}$	reference
0.8	200	1	0.1	364	[Bahr01]
0.8	10000	600	1000	12.5	[AuBe01a]
0.8	1000	15	7	91	[AuBe01a]
0.8	49.7	0.130	0.160	343	figure 2.14

Figure 5.22: Maximum current, I_{max}, at the switching point and for high and low output values.

The important change in current level as a function of input voltage was between two and three orders of magnitude from both the measurements and the simulations. The minimum DC currents of 130 and 160 pA are denoted I_{O_L} and I_{O_L}. I_{O_H} means the measured DC-current through the V_{ss} node under normal operation and a high output level. The table in figure 5.22 has been made for a few comparisons. Inverting functions from [AuBe01a] (figure 5.22) and figure 2.14 used identically dimensioned PMOS and NMOS transistors, but the ones in figure 2.14 were from a different fabrication run, and had an effective drawn capacitance between the input and the floating gates of 5 unit capacitances of 24.54 fF each, giving a total of 122.7 fF. The others [AuBe01a] had 3 similar unit capacitances, giving a total drawn capacitance between the input and the floating gates of the NMOS and PMOS of 73.6 fF.

The inverters with an effective drawn capacitance of 122.7 fF, between input and each floating gate, had a $\frac{2I_{max}}{I_{O_L}+I_{O_H}}$ -ratio of 343 (figure 5.22), as opposed to the 73.6 fF case with factors of 91 and 12.5 for the same ratio. Both the higher capacitance value and the lower current levels in the former case made the higher ratio for 122.7 fF expected. There are very few empirical data, and some of the measured structures came from different production runs, which should be kept in mind when trying to draw conclusions.

The problems with the V_{ss} node not always staying at 0 V during measurements might have reduced the effective V_{dd} at some times, and thereby disturbed measurements. A theoretically reduced effective V_{dd} would lead to lower measured voltage gains and transconductances for the transistors, thereby having implications on most relevant measurements. If such an increase in the voltage level occured during measurements of the inverter function of the P5N5 circuit, it may have something to to with the irregular output level in figure 5.18. In that case the output level that ideally should stay at 0 V suddenly jumped up to above 100 mV when the input voltage approached 0.8 V.

Measurements have shown that the circuit is functional for these digital functions in the sense that the switching between high and low levels occured when they should. Voltage gains were somewhere around -1 for the important parts of the slope at, and surrounding, the switching point.

Measurements for the NAND3 function were shown in figure 5.19, the NOR3 function in figure 5.20 and the CARRY' function in figure 5.21. When we had the CARRY' functionality, we also had the 2-input NAND and 2-input NOR functions embedded. The circuit was functional for these digital functions in the sense that the switching between high and low levels occured when they should. Voltage gains were somewhere around -1 for the important parts of the slope at, and surrounding, the switching point.

The voltage gain should probably be increased by increasing the relative transconductance and output resistance for the MOSFETs. An approach that has apparent disadvantages such as increasing chip area and complexity, is to regenerate signals at certain instances by using relatively high gain inverting elements, as for example in 5.18. How transconductance and output resistance could be increased has been discussed earlier and was summed up in connection with figure 2.16.

The functionality of linear threshold elements with identical capacitively weighted inputs to PMOS and NMOS transistors has been demonstrated by theory, computer simulations and measurements on a CMOS chip.

Chapter 6

2-MOSFET floating-gate elements with identically weighted inputs to only PMOS or NMOS; "P1NM"/"PMN1"

6.1 Implementation and layout of "P1NM" elements

6.1.1 P1N2

The "P1NM" circuits have one capacitively weighted input to the PMOS, and m capacitively coupled inputs to the NMOS. A P1N2 circuit is shown in figure 6.1.

Using the approach firstly used for the schematic in figure 2.23 results in:

$$I_{ds,p} = I_{beq} exp\{(\frac{1}{nU_t}(\frac{V_{dd}}{2} - V_p) \tag{6.1}$$

$$I_{ds,n} = I_{beq} exp\{\frac{1}{nU_t}(\frac{1}{2}V_x + \frac{1}{2}V_y - \frac{V_{dd}}{2})\} \tag{6.2}$$

and

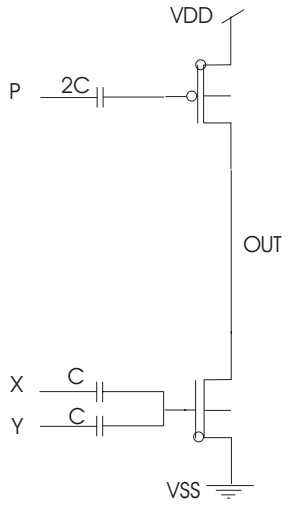

Figure 6.1: P1N2 universal element.

$$e_p = (\frac{V_{dd}}{2} - V_p) \tag{6.3}$$

$$e_n = (\frac{1}{2}V_x + \frac{1}{2}V_y - \frac{V_{dd}}{2})\} \tag{6.4}$$

Since the inputs V_x and V_y have the same capacitive weight, it is the number of 1s in the input vector [V_x,V_y] that makes a difference in the output signal, irrespective of their position. The values in the input vector are restricted to binary, 0 or 1.

This gives the truth table in figure 6.2. In two cases the output cannot be defined. From the table (figure 6.2), it appears that if V_x or V_y was constantly tied to $V_{ss} = 0$ V, the output would always be the inverted of the P signal. The value of the last input, not constantly at V_{ss}, could then be changed in real time, making a variable threshold inverter. Variable threshold inverters are described in [ShOh91] and [GoAv00].

If this circuit is to be used as a 2-input NAND, it should produce 0 on the output if, and only if, both V_x and V_y are "1". From the table in figure 6.2 we see that $e_n = V_{dd}/2$ for 2 1s in the input vector. For only one 1 in the input vector (and the other signal being 0), $e_n = 0$. To make the circuit implement the 2-input NAND function we let $e_p = V_{dd}/4$, between $V_{dd}/2$ and 0.

P	number of "1"'s	e_p	e_n	OUT
0	0	$V_{dd}/2$	-$V_{dd}/2$	1
0	1	$V_{dd}/2$	0	1
0	2	$V_{dd}/2$	$V_{dd}/2$	X /($V_{dd}/2$)
1	0	-$V_{dd}/2$	-$V_{dd}/2$	X /($V_{dd}/2$)
1	1	-$V_{dd}/2$	0	0
1	2	-$V_{dd}/2$	$V_{dd}/2$	0

Figure 6.2: The table shows part of the exponentials, e_p, e_n, and output values, for different values of P, and different numbers of "1's" on ordinary inputs V_x,V_y. If P is used as an input signal, the circuit can work as a variable threshold inverter.

$$e_p = (\frac{V_{dd}}{2} - V_p) = V_{dd}/4 \tag{6.5}$$

giving

$$V_p = V_{dd}/4 \tag{6.6}$$

In a similar way, the circuit can implement 2-input NOR by letting $e_p = -V_{dd}/4$, which means $V_p = 3V_{dd}/4$

The P1N2 circuit can function as NAND2, NOR2 and INVERT, as shown in figure 6.3.

P	number of "1"'s	e_p	e_n	OUT
0	0	$V_{dd}/2$	-$V_{dd}/2$	1
0	1	$V_{dd}/2$	0	1
0	2	$V_{dd}/2$	$V_{dd}/2$	X
$Vdd/4$	0	$V_{dd}/4$	-$V_{dd}/2$	1
$Vdd/4$	1	$V_{dd}/4$	0	1
$Vdd/4$	2	$V_{dd}/4$	$V_{dd}/2$	0
$3Vdd/4$	0	$-V_{dd}/4$	-$V_{dd}/2$	1
$3Vdd/4$	1	$-V_{dd}/4$	0	0
$3Vdd/4$	2	$-V_{dd}/4$	$V_{dd}/2$	0
1	0	-$V_{dd}/2$	-$V_{dd}/2$	X /($V_{dd}/2$)
1	1	-$V_{dd}/2$	0	0
1	2	-$V_{dd}/2$	$V_{dd}/2$	0

Figure 6.3: The table shows some parts of the exponentials, e_p, e_n, and output values, for different values of P, and different numbers of "1's" on ordinary inputs V_x,V_y. If $P = V_{dd}/4$ the circuit implements the 2-input NAND function. If $P = 3V_{dd}/4$ the circuit implements the 2-input NOR function.

6.1.2 P1N3

The P1N3 circuit was described earlier, starting with the schematic in figure 2.23. The layout can be seen in figure 6.4. It is very similar to the layouts described earlier for the P3N3 and P5N5 circuitry. It has dummy capacitors on both edges of the two horizontal rows of capacitors to improve matching of the capacitances.

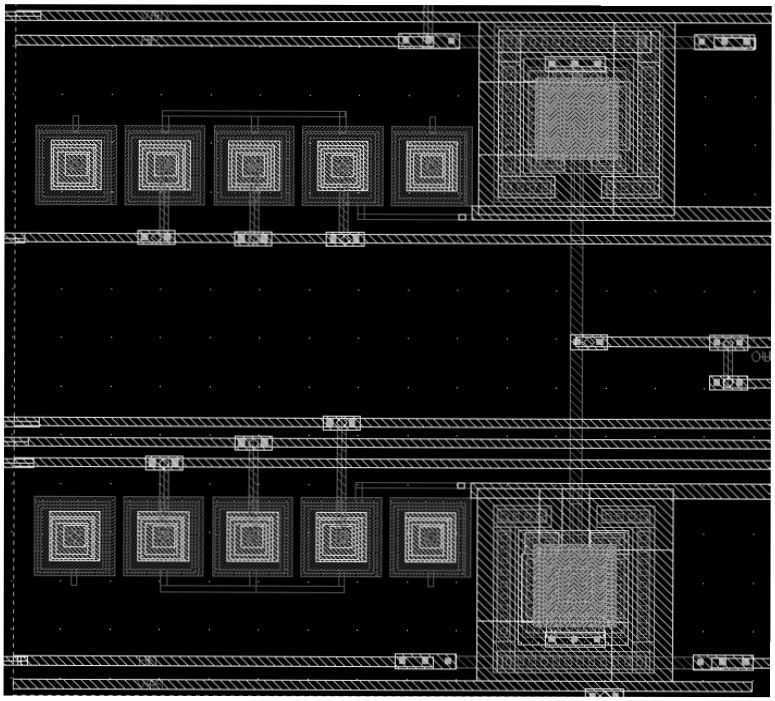

Figure 6.4: P1N3 universal element.

6.2 UV-programming and test setup of the P1N3 element

The UV-programming and testing of the P1N3 circuit were done in a very similar way as for the inverters and the P5N5 element described earlier. Due to the problems with keeping the correct V_{ss} level of 0 V, the instrument for measuring current (Keithley 236) was removed, giving the test setup

sketched in figure 6.6. Not having the current level information anymore, the voltage characteristics and former experience had to be relied upon. Apart from that, the same instruments were used, but in most cases coupled to new I/O-cells of the chip. More details can be found in figures A.2 and A.3 A die photo from the chip can be found in figure 6.7.

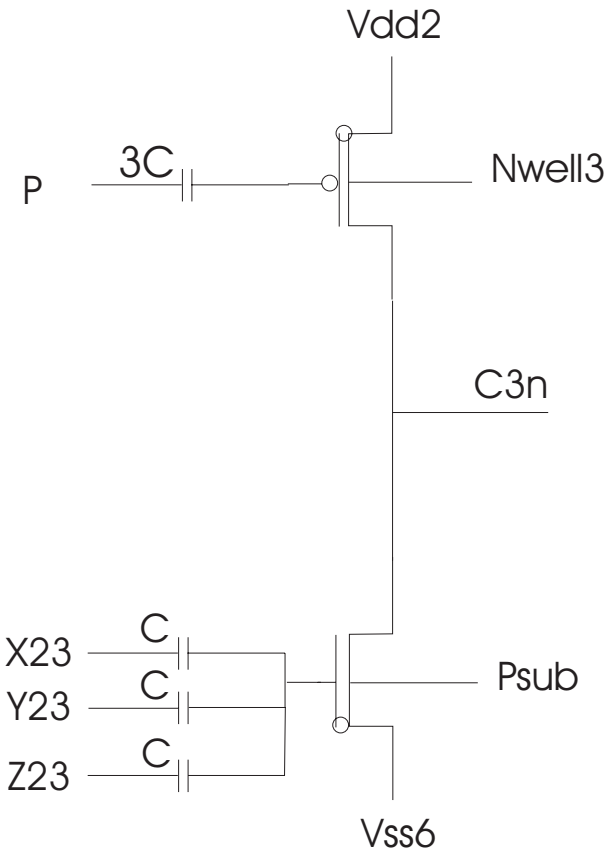

Figure 6.5: FAEXP20812 (figure 6.6) contains the P1N3 element shown. Node names are referred to the $CHIP5\_capall\_netlist$ [Aune02].

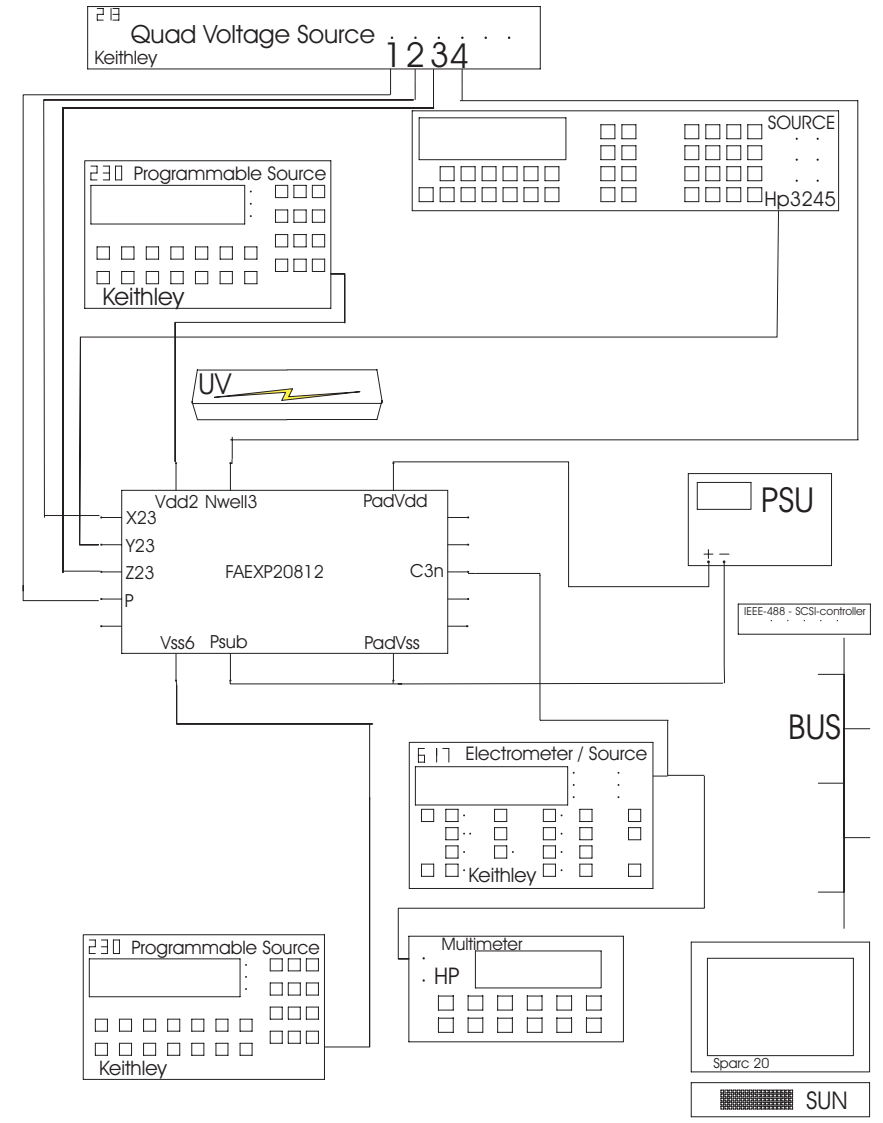

Figure 6.6: Test setup. Ground / zero voltage potential for the sources and measurement equipment were short-circuited to "Psub" and "PadVss" on the chip.

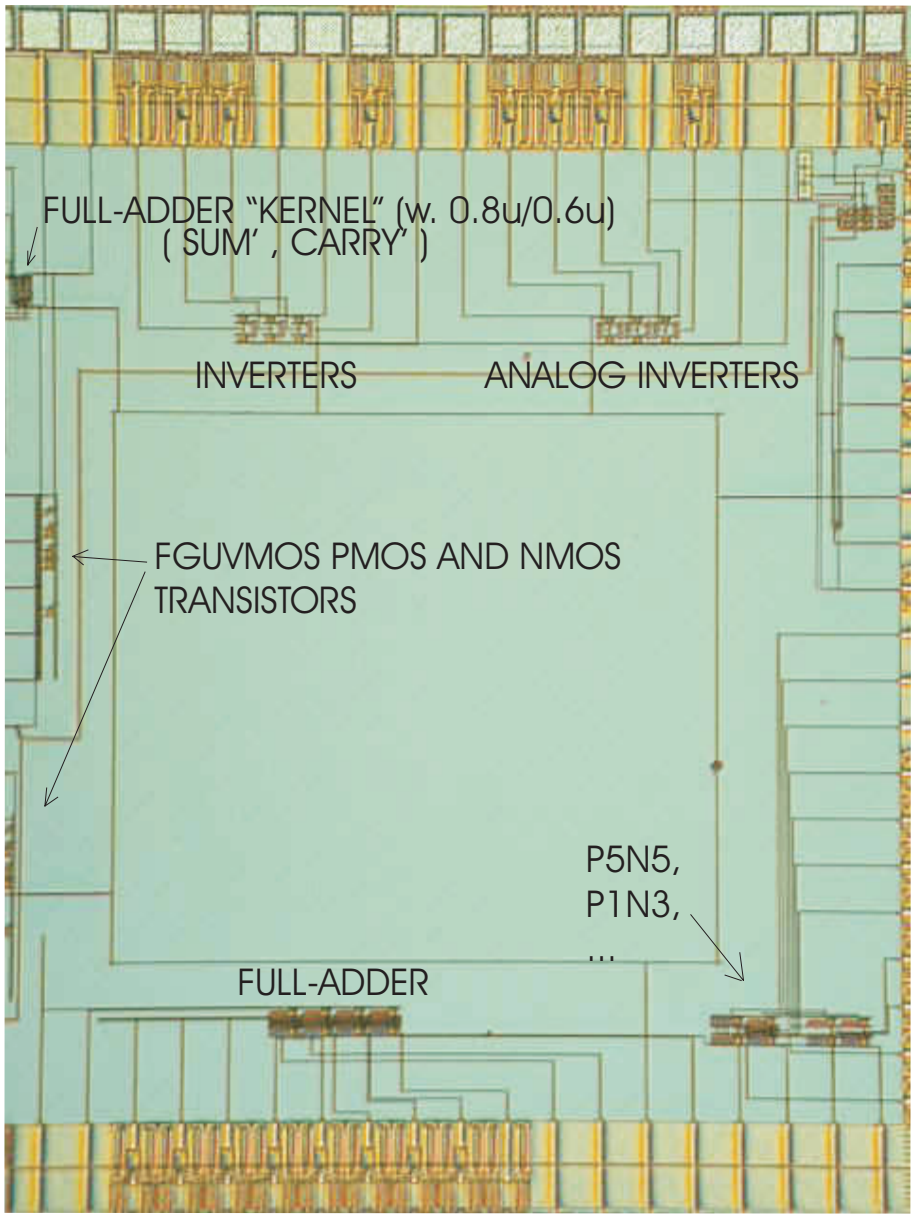

Figure 6.7: Die photo from the 84 pin chip, sent for processing in spring 2001.

6.3 P1N3 results

A perfect P1N3 circuit has the functionality described in figure 2.28.

The INVERT function was UV-programmed and measured in accordance with figure 6.8, for well voltages differing 100 mV. The higher voltage level on the well should give the lowest high level for the inverter. It did, according to the table (figure 2.28), and the change was from a high level of 0.799 V to 0.796 V. The PMOS was "weakened", and this affected the voltage transfer characteristics as demonstrated by measurements in figure 6.9.

CARRY', NAND2, NOR3, INVERT functionality were tested according to figure 6.10, and the results are shown in figure 6.11. The three curves correspond to binary input values for [X,Y,Z] changing between [0,0,0] and [1,0,0] for the uppermost line, between [0,0,1] and [1,0,1] for the middle line and between [0,1,1] and [1,1,1] for the lower line.

CARRY', INVERT, NOR3 and NAND3 functionalities measurements were done according to data in figure 6.12, and results are displayed in figure 6.13.

Vp [V]	Vm [V]	ProgWell [V]
1.5	0.4	0.0
Vdd [V]	Well [V]	I_{beq} [nA]
0.8	1.5	none
W [V]	y [V]	z[V]
x	x	x
Vstart [V]	Vstop [V]	Function
0.796	0.000	INVERT

Vp [V]	Vm [V]	ProgWell [V]
1.5	0.4	0.0
Vdd [V]	Well [V]	I_{beq} [nA]
0.8	1.4	not measured
W [V]	y [V]	z[V]
x	x	x
Vstart [V]	Vstop [V]	Function
0.799	0.001	INVERT

Figure 6.8: Different applied voltages for UV-programming and testing of INVERTER functionality of P1N3.

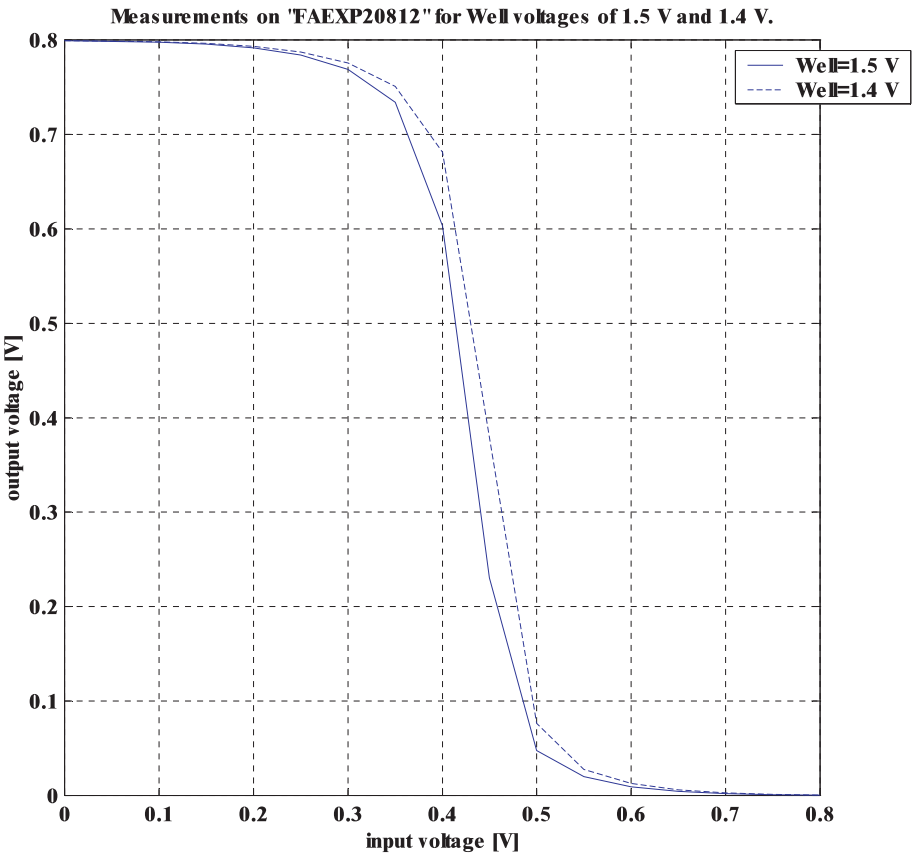

Figure 6.9: The measurements are taken on the "FAEXP20812" circuit when it is operating as an inverter. Increasing the well potential weakens the PMOS, making the output for an increasing input voltage go to "0" earlier.

Vp [V]	Vm [V]	ProgWell [V]
1.5	0.4	0.0
Vdd [V]	Well [V]	I_{beq} [nA]
0.8	1.5	none
W [V]	y [V]	z[V]
0.4	0	0
Vstart [V]	Vstop [V]	Function
0.797	0.744	CARRY'

Vp [V]	Vm [V]	ProgWell [V]
1.5	0.4	0.0
Vdd [V]	Well [V]	I_{beq} [nA]
0.8	1.5	not measured
W [V]	y [V]	z[V]
0.4	0	0.8
Vstart [V]	Vstop [V]	Function
0.742	0.066	CARRY'

Vp [V]	Vm [V]	ProgWell [V]
1.5	0.4	0.0
Vdd [V]	Well [V]	I_{beq} [nA]
0.8	1.5	none
W [V]	y [V]	z[V]
0.4	0.8	0.8
Vstart [V]	Vstop [V]	Function
0.072	0.021	CARRY'

Figure 6.10: Different applied voltages for UV-programming and testing of CARRY' functionality of P1N3.

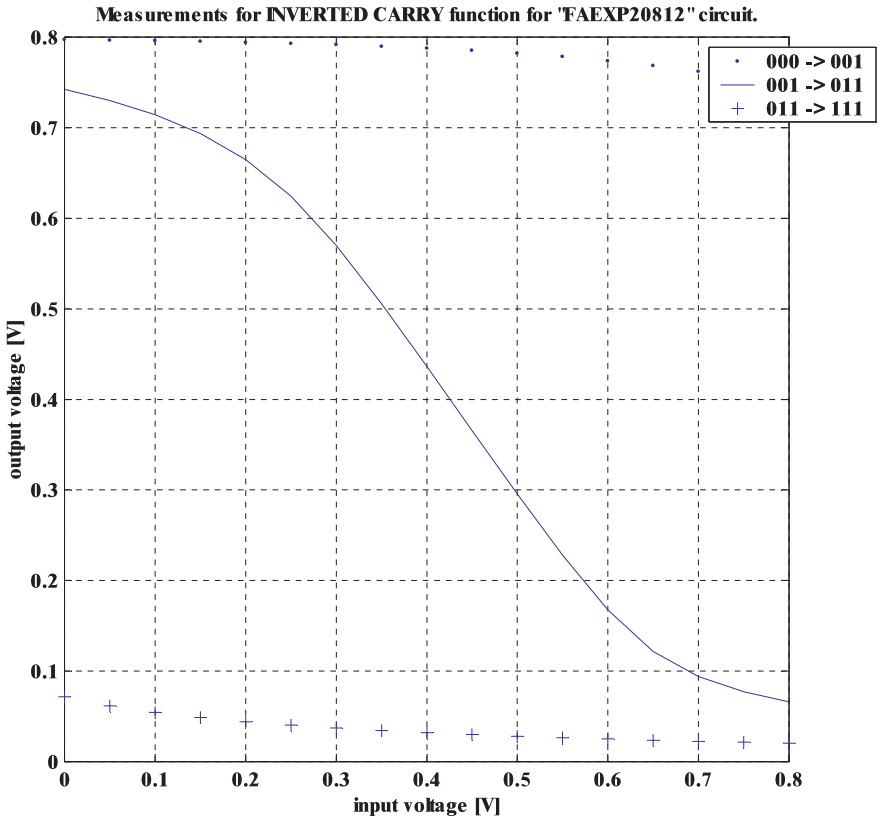

Figure 6.11: The measurements are taken from the "FAEXP20812" circuit when it is producing the CARRY' of a FULL-ADDER.

Vp [V]	Vm [V]	ProgWell [V]
1.5	0.4	0.0
Vdd [V]	Well [V]	I_{beq} [nA]
0.8	1.4	none
W [V]	y [V]	z[V]
x	x	x
Vstart [V]	Vstop [V]	Function
0.799	0.001	INVERT

Vp [V]	Vm [V]	ProgWell [V]
1.5	0.4	0.0
Vdd [V]	Well [V]	I_{beq} [nA]
0.8	1.4	not measured
W [V]	y [V]	z[V]
0.8	0.0	0.0
Vstart [V]	Vstop [V]	Function
0.697	0.015	NOR3

Vp [V]	Vm [V]	ProgWell [V]
1.5	0.4	0.0
Vdd [V]	Well [V]	I_{beq} [nA]
0.8	1.4	none
W [V]	y [V]	z[V]
0.4	0.0	0.8
Vstart [V]	Vstop [V]	Function
0.771	0.207	CARRY'

Vp [V]	Vm [V]	ProgWell [V]
1.5	0.4	0.0
Vdd [V]	Well [V]	I_{beq} [nA]
0.8	1.4	none
W [V]	y [V]	z[V]
0.4	0.8	0.8
Vstart [V]	Vstop [V]	Function
0.718	0.3100108	NAND3

Figure 6.12: Different data regarding UV-programming and testing of P1N3.

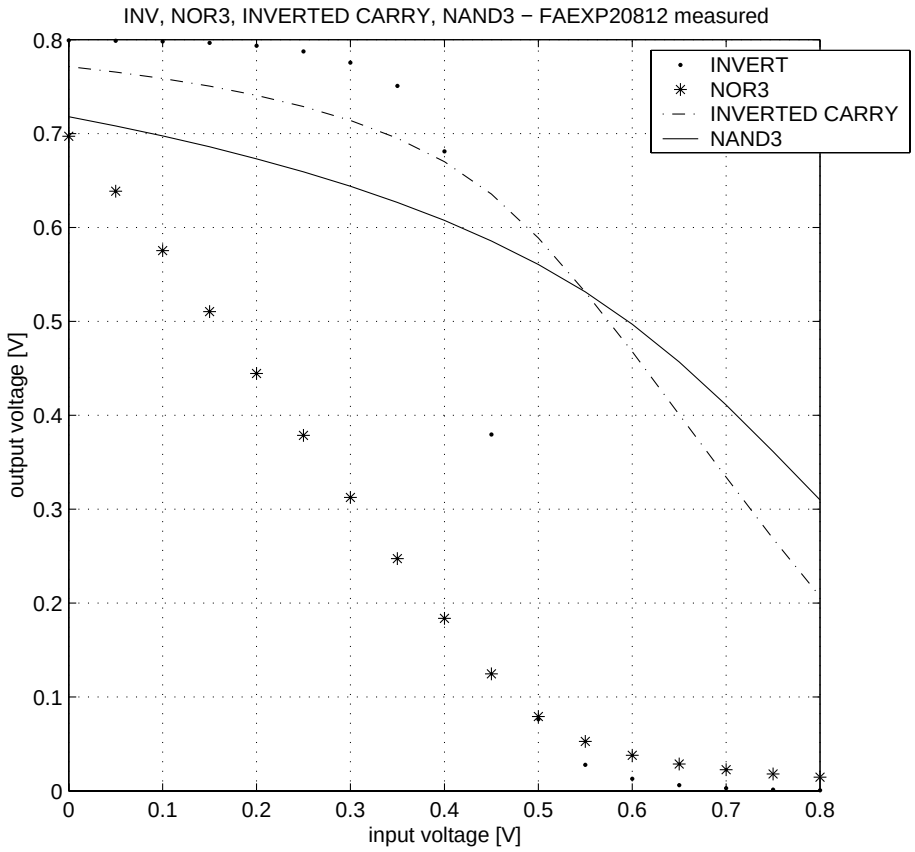

Figure 6.13: CARRY', NAND3, NOR3, INVERT measurements. Relevant data are found in figure 6.12.

6.4 P1NM / PMN1 discussion

Computing elements like the ones in this chapter compute NAND or NOR functions depending on the number of capacitively weighted inputs to the NMOS transistor. For an n-input element, up to n-input NAND or n-input NOR can be computed. The technology of choice will restrict the number of inputs.

The inverter function for the P1N3 element has been demonstrated by measurements, and the input-output voltage characteristics seem to be in correspondence with earlier measurements for the inverter function implemented by using the P5N5 element. The effect of adjusting the well potential was demonstrated.

The CARRY' function was demonstrated, and showed a gain $\leq$ -1 for important parts of the characteristics. By demonstrating the CARRY' function the NAND2, NOR2 and INVERT functions are proven by measurements at the same time, because they are embedded in the functionality of inverted CARRY. Measurements have proven that the function can be implemented using only 2 MOSFETs in s standard double poly CMOS technology.

Measurements in figure 6.13, trying to measure CARRY', NAND3, NOR3 and INVERT after UV-programming once and keeping the same well potential for all functions might at best be said to show desired functionality, but with low performance for digital purposes. Voltage gain and noise margins should be improved basically like for the P5N5 element, INVERTERS and single transistors mentioned in earlier discussions.

Chapter 7

INVERTERS, "P1NN" and "PMNM" as building blocks

7.1 Basic digital functions

7.1.1 Generating Boolean functions using PMNM and P1NM building blocks

A truth table for the CARRY' function is shown in 7.1. This function can be implemented by using the P5N5 [AuBe01e], P3N3 or P1N3 circuits [AuBe01a], [AuBe01b], for example.

X	Y	Z	OUT
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	0

Figure 7.1: The table shows the CARRY'-function of a FULL-ADDER.

If the output from the CARRY' function is fed to the input of a circuit having the truth table as in 7.3, a possible circuit diagram is as in figure 7.2. The P5N5 part of the circuit (figure 7.2) will get the same inputs X,Y,Z as the subcircuit producing CARRY', with this CARRY' signal equaling its W

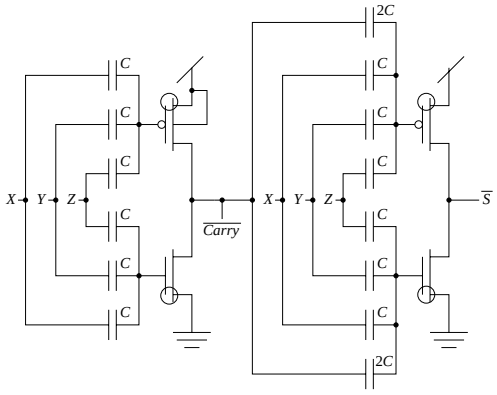

Figure 7.2: Schematic for 14 capacitor version of the circuit [AuBe01a].

input, as in figure 7.5. The truth table in figure 7.3 contains information from figure 5.7 written in a more traditional way.

The P5N5 block (figure 7.2) "uses" the subset restricted to the lines 4,6,7,8,9,10,11 and 13 in figure 7.3. Considering the inputs X,Y,Z only, these lines correspond to the SUM' ("S"") for a FULL-ADDER, as shown in figure 7.4. The 4-transistor circuit can then produce both SUM' and CARRY' (C') for a FULL-ADDER, which can be expressed by the following equations:

$$S' = x'y'z' + x'yz + xy'z + xyz' \tag{7.1}$$

$$C' = x'y'z' + x'y'z + x'yz' + xy'z' = (xy + xz + yz)' \tag{7.2}$$

The SUM' is equal to the 3-input equivalence, or Exclusive-NOR function. It outputs 1 if, and only if, the input variables have an equal number of 1s. The function is sometimes denoted "XNOR". By inspecting this truth table it becomes apparent to us that the first output in the chain, which is the C' output, or "OUT1", can produce CARRY, NAND2, NOR2 and INVERT. If a P5N5 element is used for generating OUT1, as in [AuBe01d], [AuBe01e], the NAND3 and NOR3 functions are embedded in addition.

From the truth table (figure 7.4) it is clear that letting any input, for example X, be constantly at 0 provides the XNOR2 function of the other 2 inputs. Similarly, setting X=1 makes the OUT2 node generate XOR2. Inverting the OUT1 and OUT2 nodes using two extra 2-MOSFET

W	X	Y	Z	OUT
0	0	0	0	1
0	0	0	1	1
0	0	1	0	1
0	0	1	1	1
0	1	0	0	1
0	1	0	1	1
0	1	1	0	1
0	1	1	1	0
1	0	0	0	1
1	0	0	1	0
1	0	1	0	0
1	0	1	1	0
1	1	0	0	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	0

Figure 7.3: The table shows the output of the P5N5 circuit as a function of all possible binary values of inputs W,X,Y,Z.

X	Y	Z	S'=OUT2	C'=OUT1
0	0	0	1	1
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	0	0

Figure 7.4: Truth table for the 4-transistor reconfigurable circuit.

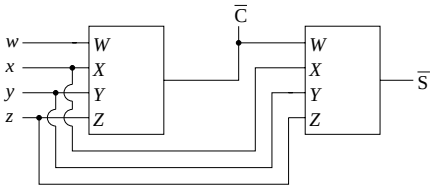

Figure 7.5: Schematic using two identical circuit blocks to implement CARRY' and SUM' for FULL-ADDER [AuBe01d], [AuBe01e].

elements increases functionality, for example giving the complete FULL-ADDER function [AuBe01e].

Two P5N5 elements can be used to implement the C' and S' functions, as in figure 7.5.

The truth table in figure 7.4 can be used to formally show inherent functionality. The output from the first block (figures 7.2 or 7.5) can be denoted OUT1, before finding other potential functionality. If z=0:

$$OUT1 = x'y'z' + x'y'z + x'yz' + xy'z' = (xy + xz + yz)' \wedge z = 0 \quad (7.3)$$

$$\Updownarrow$$

$$OUT1 = (xy)' \quad (7.4)$$

If, and only if, one of the inputs is constantly at 0, the circuit can compute the 2-input NAND function of the other two input variables. The 2-input NAND, or "NAND2", is basically enough to build any Boolean function.

The OUT1 node can also provide the INVERTER function. If, for example, z=0 and x=1, the last equation is reduced to $OUT1 = y'$. Another way of producing the inverter function could be to short circuit all three inputs, getting $OUT1 = (xy + xz + yz)' \wedge x = y = z \iff OUT1 = (xx + xx + xx)' \iff OUT1 = (x + x + x)' \iff OUT1 = ((x + x) + x)' \iff OUT1 = ((x) + x)' \iff OUT1 = x'$.

If z is changed to 1:

$$OUT1 = x'y'z' + x'y'z + x'yz' + xy'z' = (xy + xz + yz)' \wedge z = 1 \quad (7.5)$$

$$\Updownarrow$$

$$OUT1 = x'y' = (x'y')'' = (x'' + y'')' = (x + y)' \quad (7.6)$$

The OUT1 node can compute 2-input NOR.

Letting z=0 gives the Exclusive-NOR (XOR) / equivalence function on the OUT2 node:

$$OUT2 = x'y'z' + x'yz + xy'z + xyz' \wedge z = 0 \quad (7.7)$$

$$\Updownarrow$$

$$OUT2 = x'y' + xy = x \odot y \quad (7.8)$$

In the opposite case, letting one of the inputs be 1, gives the Exclusive-OR (XOR) function:

$$OUT2 = x'y'z' + x'yz + xy'z + xyz' \wedge z = 1 \quad (7.9)$$

$$\Updownarrow$$

$$OUT2 = x'y + xy' = x \oplus y \quad (7.10)$$

One instance of a circuit like P1N3, P3N3, P1N5 or P5N5 can in addition to the CARRY' function provide NAND2, or NOR2 or INVERT functions. Then, if a similar element used as an inverter is added to the output, the AND, or OR or Buffer functions might be realized as well. The OUT2 node could provide the XOR or XNOR functionalities. This means that all basic Digital logic gates, shown in [Mano84], p. 57 can be made using 4 transistors at most.

Functionality can be changed in real time by applying voltages for selecting functionality to the input(s).

7.1.2 Simulation of a 4-transistor circuit able to generate SUM' and CARRY'

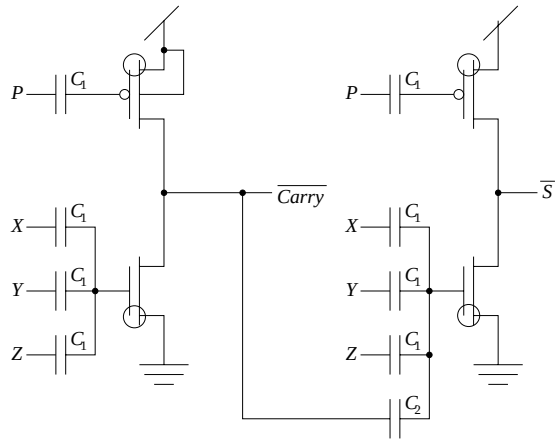

Figure 7.6: Schematic for 9-capacitor circuit able to compute SUM' and CARRY' [AuBe01a].

The ability of the 4-transistor circuit to produce SUM' and CARRY' for a FULL-ADDER is demonstrated by simulation in figure 7.7 [AuBe01a]. The SUM' signal is 0 if, and only if, an odd number of inputs are 1. The CARRY' signal is 0 if, and only if, two or three inputs are 1. The same simulation is done for two different types of the circuits. The 14 capacitor version is the one shown in figure 7.2, while the 9 capacitor version is the one in figure 7.6.
Using two P5N5 elements, or two P1N3 elements, is shown in figures 7.10 and 7.13, respectively, as integral parts of FULL-ADDERs.

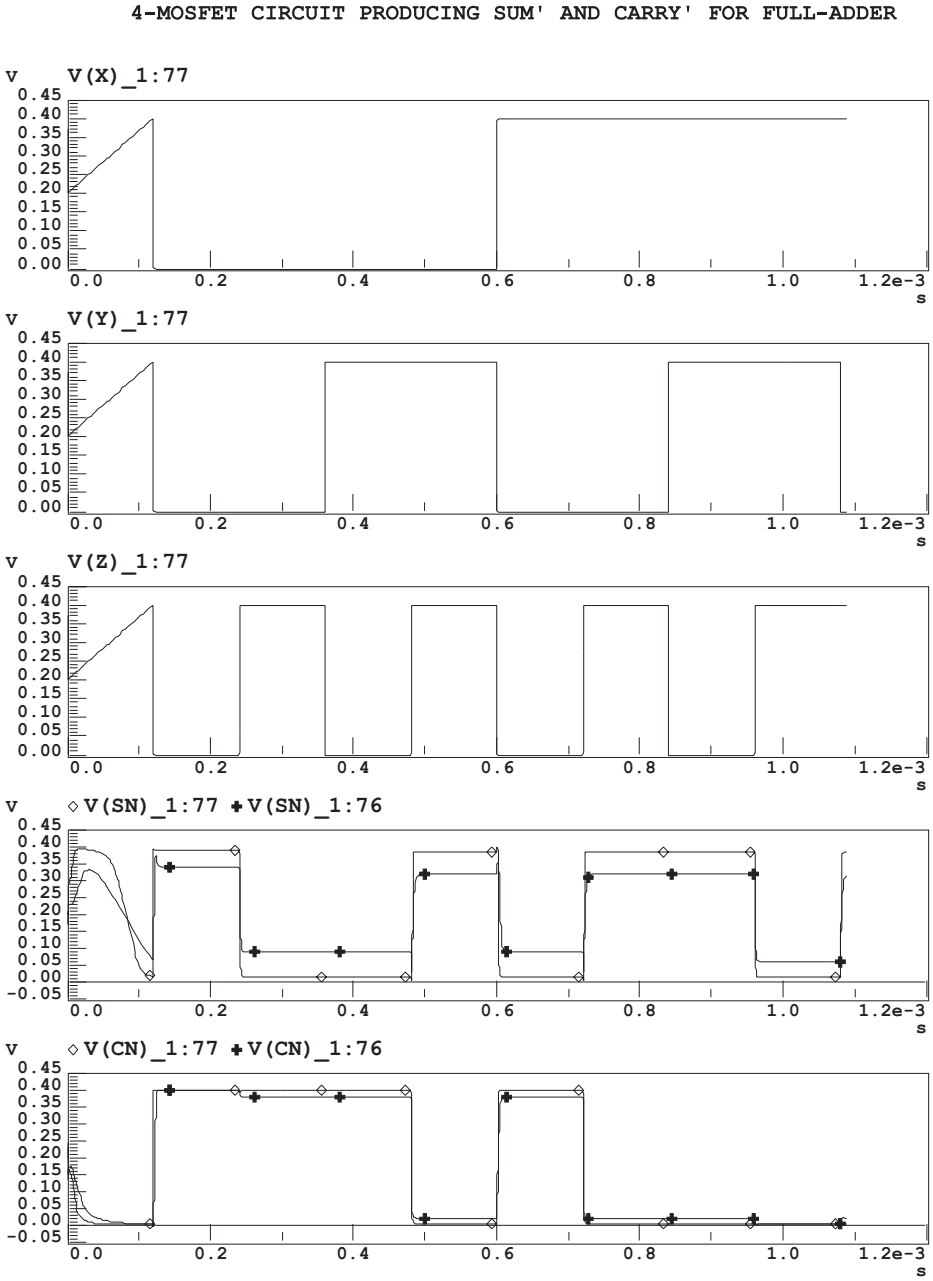

Figure 7.7: Simulation of SUM' and CARRY' function. The seven curves represent from top to bottom, the voltages on X,Y,Z, S'('SN') and C'('CN') for 14 and 9 capacitor [AuBe01a] circuits. The 14 capacitor version has the best noise margins for both the S' and C' nodes.

7.1.3 Area of a Boolean function generator using minimum transistors

The picture in figure 7.8 is taken from the 84 pin chip sent for processing in spring 2001. The area equals approximately $1600(\mu m)^2$, which could be doubled for the complete FULL-ADDER function, for example.

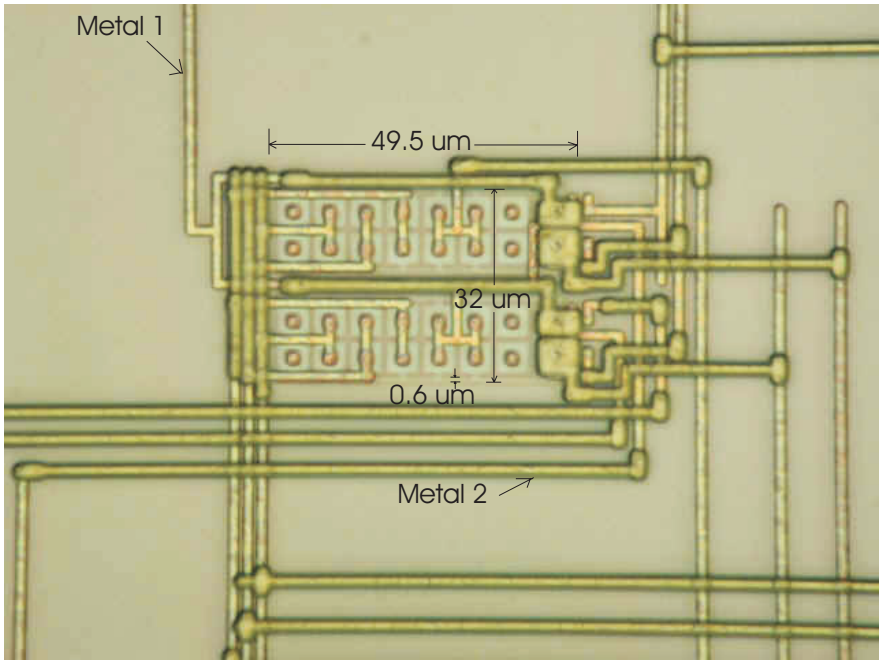

Figure 7.8: Circuit for generating SUM' and CARRY', or other digital functions. W/L=0.8μm/0.6μm.

7.1.4 Discussion regarding FGUVMOS binary function generation

These circuit might be compared to the "Soft-Hardware" circuits in [ShOh91], [AnGo97a], [GoAv00].

no. of. func./no. tran.	V_{dd}[V]	no. of transistors	problems	reference
16/12	5	12	matching	[ShOh91]
4/8	5	8		[GoAv00]
4/5	5	5		[AnGo97a]
6/2	$\leq$0.8	2	$UV-progr.$	[AuBe01b]
1/2	$\leq$0.8	2	$UV-progr.$	[AuBe01c]
10/4	$\leq$0.8	4	$UV-progr.$	[AuBe01a]

Figure 7.9: Some numbers for "multifunction binary circuits".

The Soft Hardware circuit in [ShOh91] is able to represent 16 different logic functions by just changing some control signals.

Functionality mentioned in [GoAv00] are the NAND2 and EXOR2 functions. If buffer and invert are added, the functionality count (figure 7.9) is increased.

In [AnGo97a] an output for CARRY, AND and OR is taken out from a circuit resembling a differential pair.

The NAND2 and NOR2 circuits from [AuBe01c] were conceptually viewed upon as having fixed functionality. CARRY', NAND3, NAND2, NOR3, NOR2 and INVERT functions are the background for the count for the two-transistor P1N3 circuit in [AuBe01c], while [AuBe01a] has the 6+4 functionalities of the 1st and 2nd stage, respectively. An ordinary FULL-ADDER can also implement a wide range of Boolean functions in addition to the SUM and CARRY functions. The reason why it is not desired is that a standard implementation, like in figure 7.18, might use close to 30 transistors, making it unattractive for implementing a function like the inverter, since it uses only two MOSFETs in the standard approach.

A major obstacle for the use of neuron MOS transistors, in strong inversion, in high precision circuits is the mismatch typically found between devices, and 6 nm accuracy is required in the fabrication of the transistor gate lengths, according to [RaFr99]. Mismatch represents a limit for FGUVMOS as well, but maybe to a lesser extent, due to the possibility to exploit very steep subthreshold slopes. Another type of subthreshold circuits has

been found to be more robust than their above threshold counterparts, mainly because of the device subthreshold characteristics and exponential relationships [SoRo99]. The topic should be further researched before conclusions may be drawn.

The circuit able to generate carry' from [AuBe01b] reduces the transistor count for this function more than than 90%, from 22 to 2, compared to the FGUVMOS full-adder in [BeWi99].

Multi-function basic building blocks [ShOh91], [ShOh93], [AuBe01b], [AuBe01a] also seem to provide high functionality per transistor, due to the fact that the transistors are used as something more than "switches", as in traditional digital design. The circuits are real time reconfigurable, which means that the functionality can be changed during operation without reprogramming the floating gates. The region of operation, combined with low current levels, allows for ultra low-voltage/low-power usage.

7.2 8-transistor FULL-ADDERs

7.2.1 Implementation and layout of FULL-ADDERs

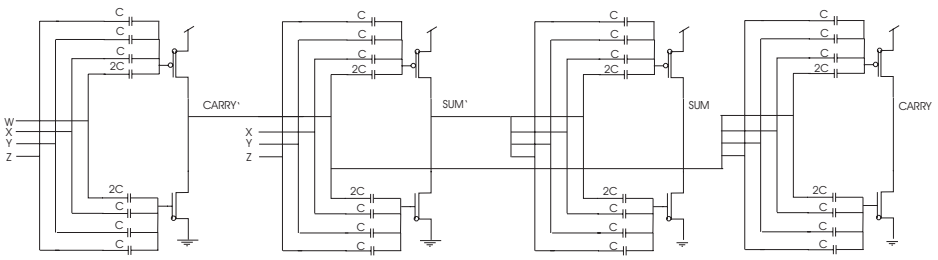

Figure 7.10: Four P5N5 elements used for the FULL-ADDER function.

A FULL-ADDER using P5N5 elements with the $20.8\mu m/1.2\ \mu m$ PMOS and NMOS transistors and 25.54 fF drawn unity capacitances between inputs and floating gates has been implemented [AuBe02a], as seen in figure 7.10. The layout is depicted in 7.11.

Another 2-bit ripple-carry adder, using P5N5 elements was implemented in the same AMS 0.6 CMOS process and sent for fabrication in summer 2001. An important point was to have it as a kind of backup in case there should be something fundamentally wrong with the 84-pin chip sent for processing in spring 2001. The netlist is not accessible, since the chip

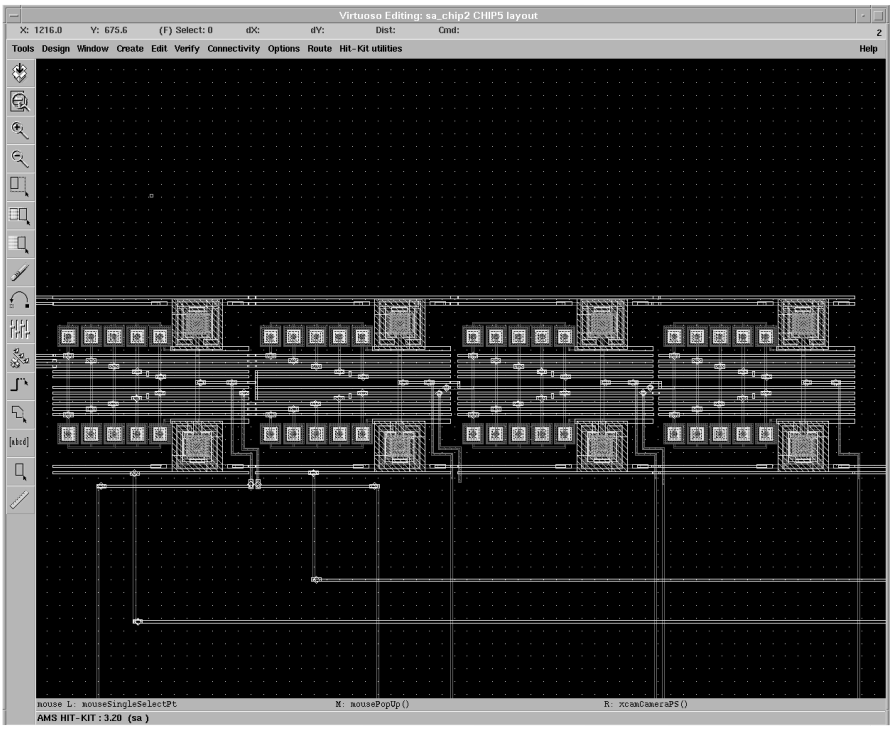

Figure 7.11: Layout with four circuits connected as a FULL-ADDER. All W/L drawn are 1.2μm/1.2μm, and poly1-poly2 capacitors designed for 14.7 fF units.

contains various research circuits owned by the University of Oslo. The layout is shown in figure 7.12.

A FULL-ADDER based on four P1N3 elements is shown in figure 7.13.

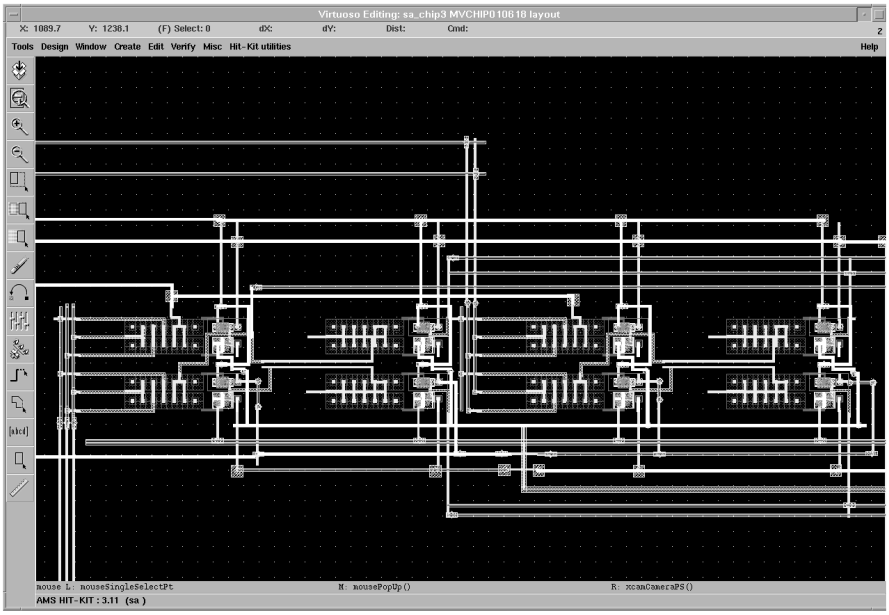

Figure 7.12: Layout for 2-bit ripple-carry adder containing eight universal elements. All W/L=1.2μm/1.2μm.

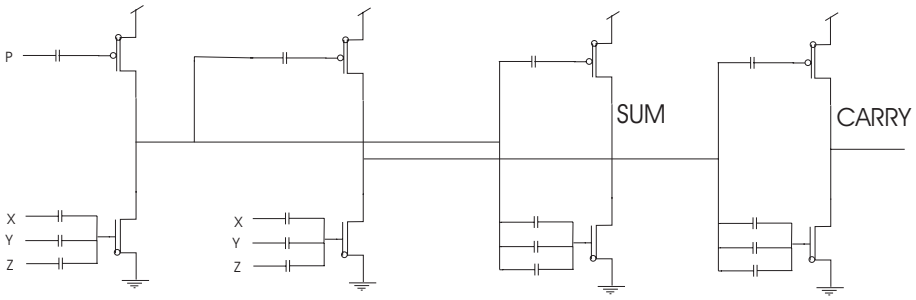

Figure 7.13: Schematic for FULL-ADDER using P1N3 (figure 6.5) elements.

7.2.2 Functionality and Power-Delay-Product simulations of 8-transistor FULL-ADDER

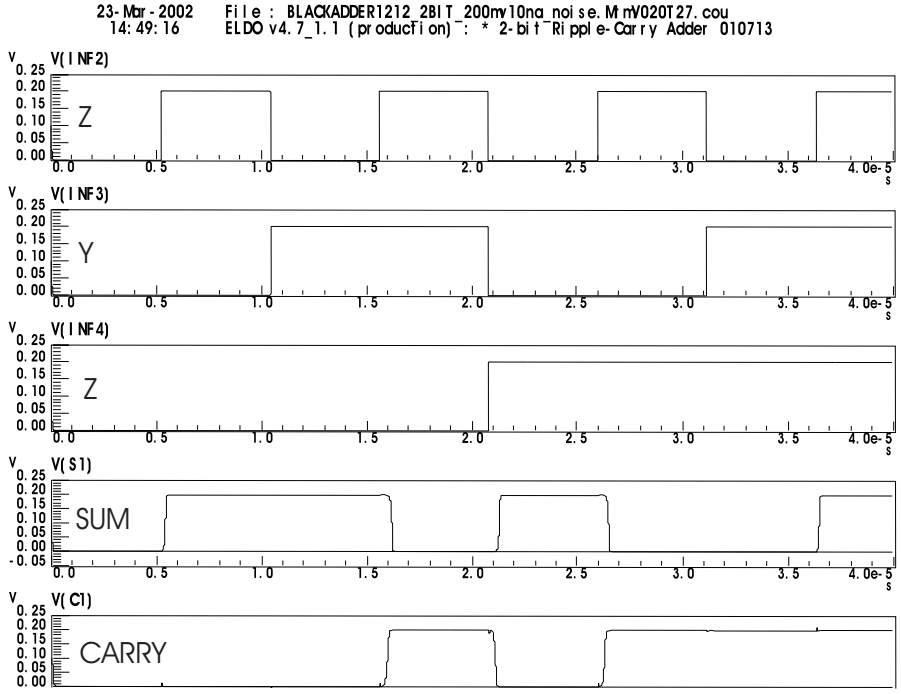

Figure 7.14: Simulation of FULL-ADDER from [AuBe01e].

The functionality of the 8-transistor FULL-ADDER in figure 7.10 was demonstrated by simulation in figure 7.14, for a supply voltage of 200 mV [AuBe01e].

Energy x Time is often considered to be the metric of choice for low-power applications [SvMa00]. Some simulations regarding power consumption and speed of the circuit have been done [AuBe01e], and the results are shown in figure 7.15, for Vdd of 200 and 800 mV, and different equilibrium, I_{beq}, currents. t_r and t_f are worst case delays for S1 ("SUM1") rising and falling, respectively. P was the average power [W] when the circuit is operating at maximum frequency. $((t_r + t_f) * P)/2$ was used for finding "PDP" / "Power-Delay-Product". The results indicate that lowering the threshold voltage betters the PDP numbers, which were roughly constant for the V_{dd}=200 mV simulations.

For "EDP" the PDP values were multiplied with $(t_r + t_f)/2$. EDP numbers were improved for increasing current levels. For 200 mV and 1 uA equilibrium currents the circuit no longer worked properly. Lowest PDP was 2.3 fJ, according to simulations in figure 7.15.

V_{dd} [mV]	I_{beq}[nA]	t_r[s]	t_f[s]	F_{max} [Hz]	P[W]	PDP [pW]	EDP [Ws]
200	1.0	4.1u	4.9u	111k	5.2e-10	0.0023	1.05e-20
200	10	0.5u	0.5u	1M	4.45e-9	0.0022	1.1e-21
200	100	73n	73n	6.8M	4.2e-8	0.0031	2.2e-22
200	1000	-	-	-	7.3e-7	-	-
800	1.0	2.2u	2.4u	217k	5.95e-9	0.0137	3.14e-20
800	10	346n	297n	1.01M	3.9e-8	0.0125	4.0e-21
800	100	74n	53n	7.9M	2e-7	0.0127	8.1e-22
800	1000	18n	13n	32M	9.6e-7	0.0149	2.3e-22

Figure 7.15: Some simulation results for an 8-transistor FULL-ADDER (1-bit adder) [AuBe01e].

The P1N3 based FULL-ADDER has been simulated in figure 7.16. From the simulation it is apparent that the SUM signal went high if, and only if, there were 1 or 3 high input signals. The CARRY signal went high if, and only if, there were 2 or 3 high input signals.

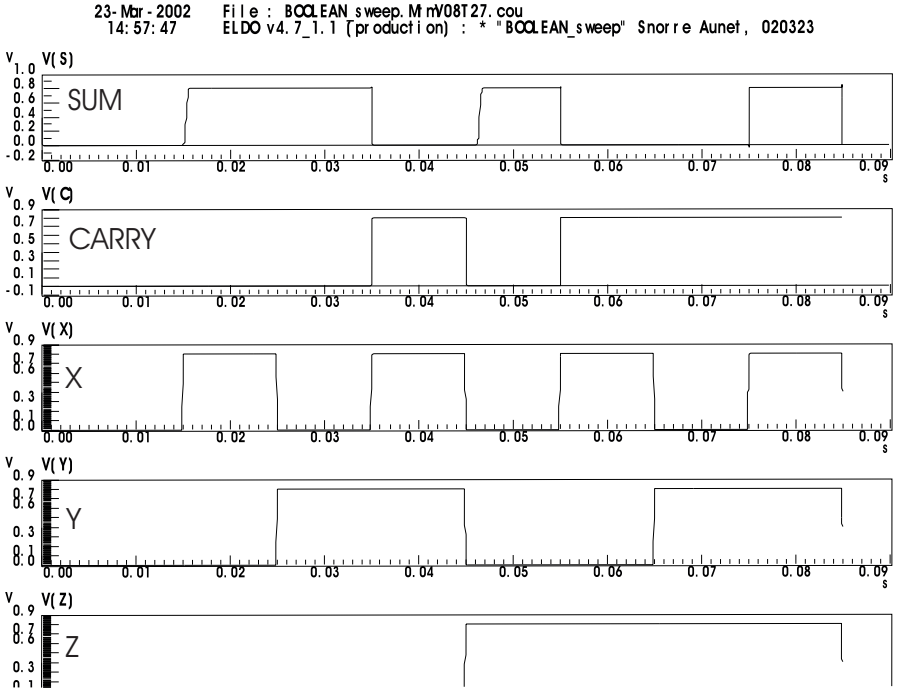

Figure 7.16: Simulation for a P1N3 based FULL-ADDER. W/L=20.8μm/1.2μm. Unit capacitances were 70 fF.

7.2.3 FULL-ADDER discussion

PDP

In [ShBa99] two FULL-ADDER cells simulated from a netlist extracted from layout, in a 0.6 μm CMOS technology, were compared. In [ShBa00] 25 different FULL-ADDER cells were simulated, based on netlists extracted from layout, in a 0.35 μm CMOS technology. Power-Delay-Product (PDP) was among the parameters that were compared for the different cells.

name	no. of. tran.	W·e-4	s·e-10	fWs	reference
"worst"0.6	16	2.728	3.378	92	[ShBa99]
"best"0.6	20	2.220	2.688	60	[ShBa99]
"worst"0.35	22	6.363	4.571	290.5	[ShBa00]
"best"0.35	16	0.638	2.887	18.4	[ShBa00]
FA200	8	0.0000052	45000	2.3	[AuBe01d]
FA800	8	0.00031	4800	15	[AuBe01d]

Figure 7.17: Some simulation results for the FULL-ADDER (1-bit adder).

In figure 7.17 "worst" and "best" refer to PDP numbers. " "FA200" and "FA800" are numbers for a FULL-ADDER implemented with four P5N5 elements and 200 mV and 800 mV supply voltages, respectively [AuBe01d]. Results indicate that lowering the effective threshold voltage betters the PDP numbers, which are roughly constant for a given supply voltage, at least for our 200 mV simulations (figure 7.15). The circuits from [ShBa99] implemented in a 0.6 μm technology had from about 26 to 40 times higher PDP numbers as the best FGUVMOS case with a V_{dd} of 200 mV, as can be seen in the table (figure 7.17). A comparison of the 0.35 μm technology gives factors 8 to 120 in favor of the FGUVMOS circuit.

The simulation results in figure 7.17 also indicate that the FGUVMOS circuits ([AuBe01d]) might be several orders of magnitude slower than the others, but at the same time use several orders of magnitude less energy per switching operation.

A paper [BrBr01] recently reported an inverter power-delay product of less than 0.1 fJ/stage at 25 C, and V_{dd}=0.1 V, in a 180 nm CMOS technology. It was claimed that they believed that it was the lowest reported. The 4-stage FULL-ADDER presented here has, according to simulation, a PDP of 2.3 fJ, which means about 0.6 fJ/stage.

It has also been argued that subthreshold logic consumes less power

than other known low-power circuits, including adiabatic logic [SoRo99], [SoRo01].
Two subthreshold logic styles were compared to an adiabatic circuit, using TSMC 0.35 μm process technology in [SoRo99]. Energy per switching was from about 2 to 25 times better for an inverter and a 4-input NOR compared to the adiabatic counterparts. The inverter in "sub-CMOS" logic style used 1.1 fJ per switching. The four universal elements in the FULL-ADDERs proposed here, put together, use about 2.3 fJ per switching, according to simulations. If the simulations gave realistic resuls there is reason to believe that the FGUVMOS approach might provide comparable, or even better performance, also since a 0.35 CMOS technology here is compared to an older 0.6 CMOS technology. Due to scaling, every new process generation tends to use less energy per switching than the previous. If the scaling factor is 0.7 between each generation [ITRS01], there is roughly one technology generation between the two. With a simple scaling model, energy per operation scales by 1/S [GuAb98]. Using this simple model would give approximately 1.6 fJ per switching for the sub-CMOS inverter if implemented in a 0.6 μm CMOS technology. A FULL-ADDER using sub-CMOS might use several times this amount of energy per switching, and probably more than the 2.3 fJ of the 8-transistor FGUVMOS FULL-ADDER.
A better comparison could be to implement comparable circuitry in equal technologies, extract netlists from layout, and do computer simulations for comparisons, as in [ShBa00].

In [BeWi99] it was concluded that the optimum operating point is "ultra-low supply voltage and large offset", connected to arguments regarding Energy-Delay-Product (EDP). If a large offset is similar to the larger current levels in figure 7.15, findings in [AuBe01d] support the same conclusion.

Though the numbers might not be directly comparable, due to slightly different simulations and technologies, as well as lack of accurate models in subthreshold [LoHa99], the circuits seem promising from a low-power point of view.

Comparisons with standard cell implementations in the AMS 0.6 CMOS technology - power dissipation when running at 1 MHz.

To try and give an indication of the low-power potential simulations on an FGUVMOS FULL-ADDER have been compared to an implementation using AMS standard cells in the same technology, a frequently used approach

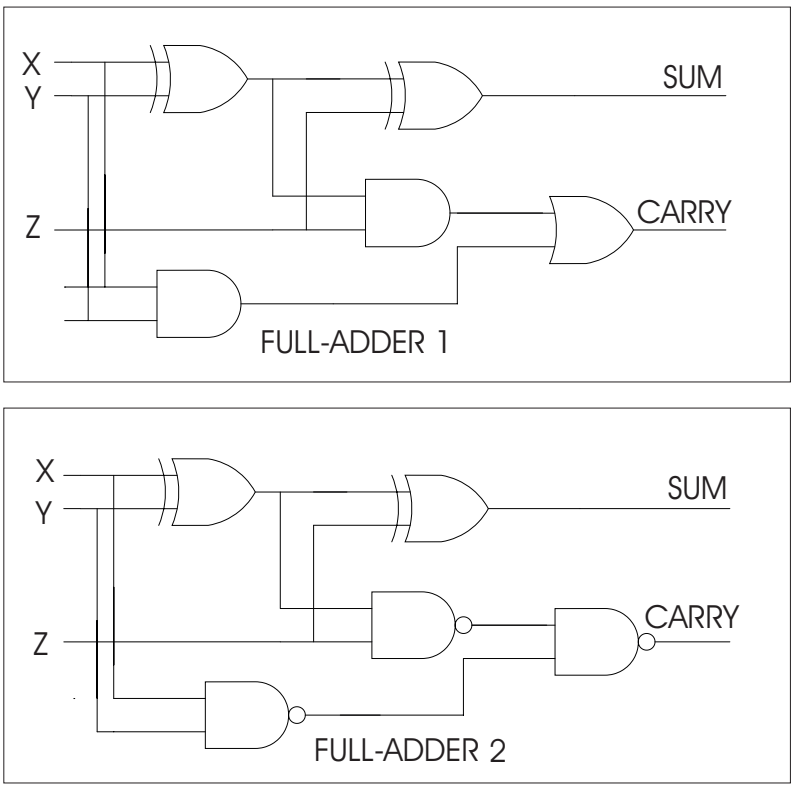

Figure 7.18: Two different FULL-ADDER implementations.

[ShBa99]. Implementing the FULL-ADDER in [Mano84] p. 123 can be used using 2 EXOR, 2 AND and an OR gate as in figure 7.18. The Standard cells for a complete "$FULL-ADDER1$" would use approximately 11 μW/MHz, or 11 μW while running on 1 MHz, while "$FULL-ADDER2$" (figure 7.18) uses about 10 μW. Simulation presented in figure 7.15 indicates 4.45 nW for the FGUVMOS FULL-ADDER with a supply voltage of 200 mV, or roughly 2500 times less. This seems to be in accordance with [SoRo01] who claimed that several orders of magnitude less power could be achieved by operating digital logic in subthreshold instead of the classical above threshold region, for a given operation frequency. One should be aware that the highest switching speeds of digital circuits are generally regarded as not attainable by subthreshold, or weak inversion, operation today.

description	cell	area [sq. mils]	power [μW/MHz]
2-input AND	AND2	0.54	2.19
2-input XOR	XO1	0.81	2.43
2-input OR	OR2	0.54	2.09
2-input NAND	NA2	0.41	1.62

Figure 7.19: Data for standard cells for the 0.6-Micron Standard Cell Databook [AMS97].

Complexity

# tran.	# tran.	logic depth	logic depth	# different blocks	reference
$CARRY'$	SUM'	$CARRY'$	SUM'		
22	16	5	4	6	[BeWi99]
2	4	1	2	2	[AuBe01a]
2	4	1	2	1	[AuBe01d]

Figure 7.20: Some comparisons between FGUVMOS FULL-ADDERs (1-bit adder). ("#tran.": number of transistors)

Many full-adders compute CARRY' and SUM' and then use two inverters to provide CARRY and SUM.

Some figures regarding numbers of active and passive elements to produce CARRY' and SUM' for different FGUVMOS FULL-ADDERS are presented in figure 7.20. for example the number of transistors involved in producing CARRY' is reduced from 22 to 2 from the implementation in [BeWi99], compared to the two others mentioned. Similar numbers for the SUM' function show a reduction in transistor count from 16 to 4.

In figure 7.21 the number of transistors and capacitors are counted, from figure 1.3.

The logic depths might be reduced from 5 to 1 for the CARRY' function, and 4 to 2 for SUM', according to figure 7.20. The signals thus have a "shorter" way to travel, which might lead to a potentially increased operational speed of the circuit in [AuBe01d], compared to the one in [BeWi99].

The FGUVMOS FULL-ADDER implementation using fewest capacitances is the one based on P1N3 elements (figure 7.13), needing only 8

circuit element	block 1	block 2	block 3	block 4	block 5	block 6	added
	4·INV	2·XOR2	1·NOR2	1· NAND2	1·GA_X	1·GA_Y	
PMOS	4· 1	2·2	1·1	1·2	1·1	1·1	14
NMOS	4· 1	2·1	1·2	1·1	1·1	1·1	12
CAP_p	4· 1	2·4	1·2	1·2	1·2	1·1	19
CAP_n	4· 1	2·4	1·2	1·2	1·2	1·1	19

Figure 7.21: FGUVMOS FULL-ADDER from [BeWi99] - elements for producing SUM' and CARRY'. 26 transistors and 38 capacitors were used. CAP_p means capacitors between inputs and floating gates of PMOS transistors.

capacitors for SUM' and CARRY' compared to 38, a 79% reduction.

Fan-in

A characteristic that isolated is worse regarding the circuits proposed in [AuBe01a] and [AuBe01d], compared to the one in [BeWi99] is that the two first mentioned have an increased number of capacitively weighted inputs to a single transistor compared to the FULL-ADDER in [BeWi99], which has a maximum number of two. This is called "fan-in", and will be discussed later. Increased "fan-in", or increased sizing of drawn capacitances, might lead to a need for higher supply voltage, compared to a lower fan-in, for circuits to function.

Area

The area of the first FULL-ADDER in figure 7.18 equals (at least) 3.24 $(mils)^2$=2090 $(\mu m)^2$ as computed from data in figure 7.19. This is about 2/3 of the area found by doubling the area of the layout in figure 7.8. The area of the latter could be reduced by allowing far less area for the inverter functions, though without having to double the area of the cell in the photo. In addition the dummy capacitances could maybe be removed, and area saved in other ways.

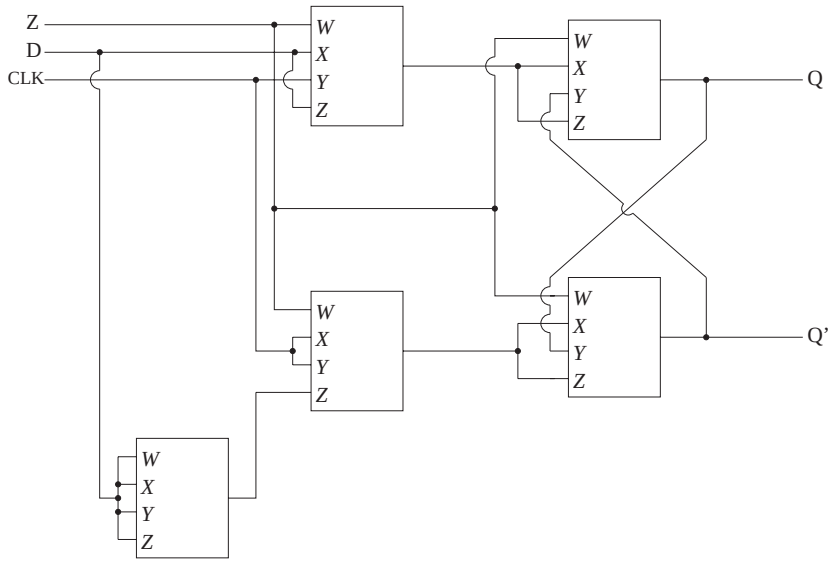

Figure 7.22: D-LATCH.

7.3 D-Latch

The D-latch resembles the one in [Mano84], and is depicted in figure 7.22. In this circuit the P5N5 universal element was used four times as a 2-input NAND-gate, and once as an inverter [AuBe01e]. The input to the capacitors of twice the size of the others, "Z", was grounded. Two of the other inputs are wired together under ordinary operation, for the circuits used as 2-input NAND. The universal element used as an inverter has all it's inputs wired together. A simulation trace is shown in figure 7.23. The characteristic equation is Q(t+1)=D, and input D was sampled during occurence of a clock pulse. A V_{dd} of 400 mV was used, which could have been lower.

The latch was made basically for two reasons: to make a FGUVMOS static memory circuit, and to demonstrate an example of the reconfigurability of the basic building blocks. In general static memory and basic digital functions is sufficient for building finite state machines.

With a simple memory and logic elements, low-power field programmable gate arrays could may be a conceivable future goal, as proposed in [Boho98].

A dynamic FGUVMOS D flip-flop was functional at a maximum clock

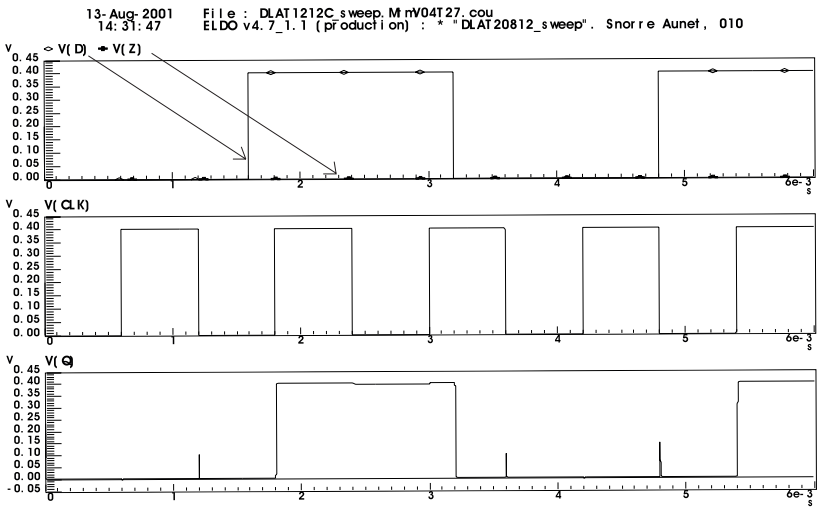

Figure 7.23: D-LATCH simulation [AuBe01d].

frequency of more than 200 MHz in simulations in [BeWi99], indicating some of the operational speed potential of FGUVMOS.

7.4 Implementation of a 6-transistor 3-bit Analog-to-Digital converter / Frequency synthesizer

7.4.1 Implementation and simulation of ADC3 / Frequency Synthetisator

This analog-to-digital converter was made after inspiration from [RaFr01], though it is different in many respects, such as region of operation, UV-programmability and modularity. All three building blocks are exactly the same. Only the external wiring differs for each block, as can be seen in figure 7.25.

An implementation using ordinary CMOS devices and conventional design using 174 transistors were used as a comparison with neuron MOS implementations in [KoSh92] and [RaFr01]. The 3-bit ADC in [KoSh92] used 16 transistors.

The 18 and 6 transistor versions in [RaFr01] need threshold voltage adjustment of transistors or calibration techniques in addition to the signal processing circuitry. The first design here was made to illustrate how the universal element "P7N7" could be used to reduce the total number of devices for implementing a certain function.

From figure 7.24 it can be seen that the circuit might be used for direct frequency synthesis as well. The frequency of the input voltage, V(A), is multiplied once at the first node, V(BIT1), two times at V(BIT2) and four times at V(BIT3).

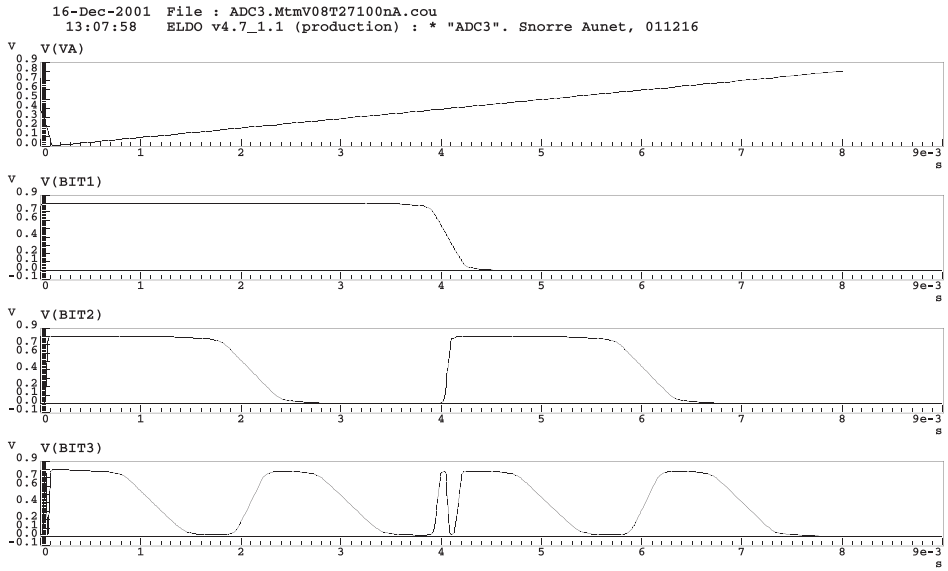

Figure 7.24: The uppermost signal growing approximately from V_{ss} to V_{dd} is analog-to-digital converted with 3 bit resolution.

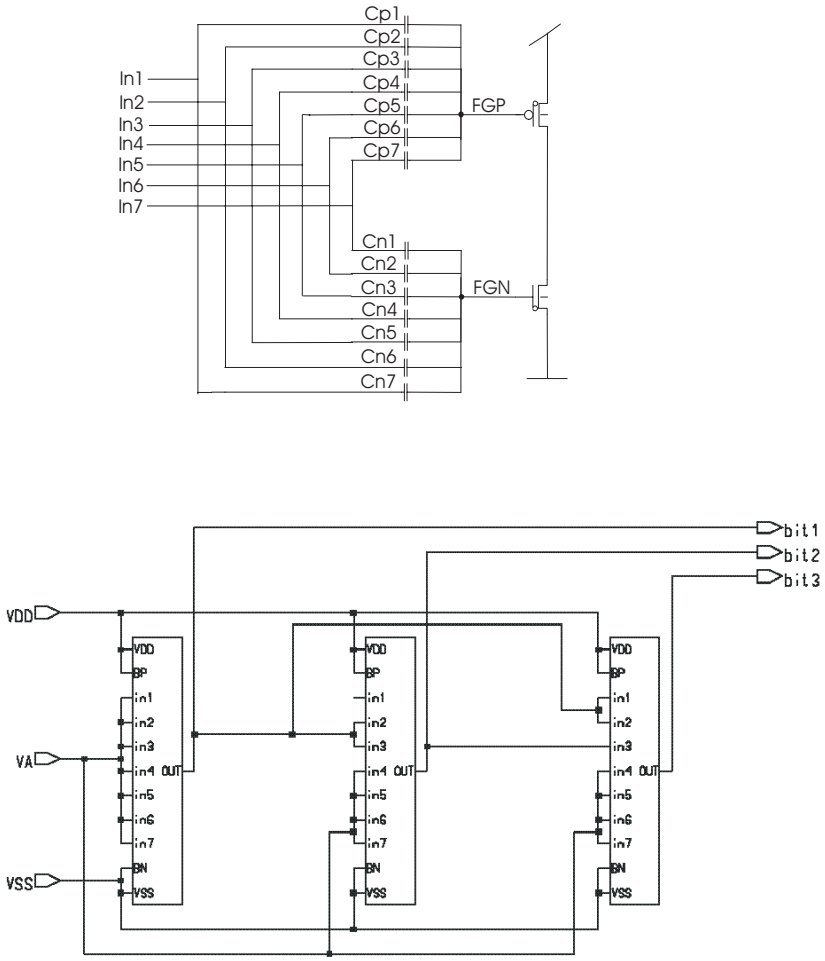

Figure 7.25: "P7N7" element on top. Below is a 3-bit ADC made from connecting three P7N7 elements. The drawn capacitances in the P7N7 element were all 24.5 fF. W/L for PMOS and NMOS were 20.8μm/1.2μm.

7.4.2 ADC / frequency synthesizer discussion

The circuit could probably get better due to improvement of the basic building block, P7N7, as well as the introduction of buffers between the stages. It is primarily an example on how the number of active elements might be reduced using floating-gate techniques, and that the reconfigurable circuits are not restricted to plain digital use only.
Another 3-bit ADC is being published in [BeNa02b]. It is not as modular as this one.
In the AMS 0.6 process this might be a problem if the capacitances at one and the same input get to be too many and too big, since there are upper bounds regarding the construction of large poly 1 layers. This has to do with chemical and thermal aspects during processing, charging up floating-layers which might be destroyed by accumulated charge [Gjer02]. During the layout of one of the chips for this work a design rule was broken in the initial design, and the layout had to be adjusted. If matching among capacitances is not too important, using dummy capacitances might not be necessary, freeing layout area for other drawn capacitances. There may be possibilities to get around the problem, or simply using another process were such a practical limit is not reached too fast.
This design constraint in the AMS process is a general problem that might limit the freedom of the designer for any FGUVMOS circuit needing too much "passive" vs "active" gate poly. Both ADCs mentioned here might get their functionality and/or performance restricted by this.
From figure 7.24 it is apparent that the low voltage gain of the P7N7 element can hinder the ADC from producing a proper digital output. The gain could be increased in similar ways as described for the inverter, or relatively high gain inverters could be placed between the stages, increasing the number of transistors to 16, which is still very few compared to most traditional implementations.

7.5 Inverter-only based logic

The simulation in figure 7.26 is produced by taking the 3 P7N7 blocks from the 3-bit ADC and letting all three outputs drive the same output node while inputs V(X), V(Y), V(Z) are given to inverter 1,2 and 3 respectively. Allowing several outputs connected together have been verified done in [BeWi99] and [Bahr01]. Here the 3-input NOR function was simulated.

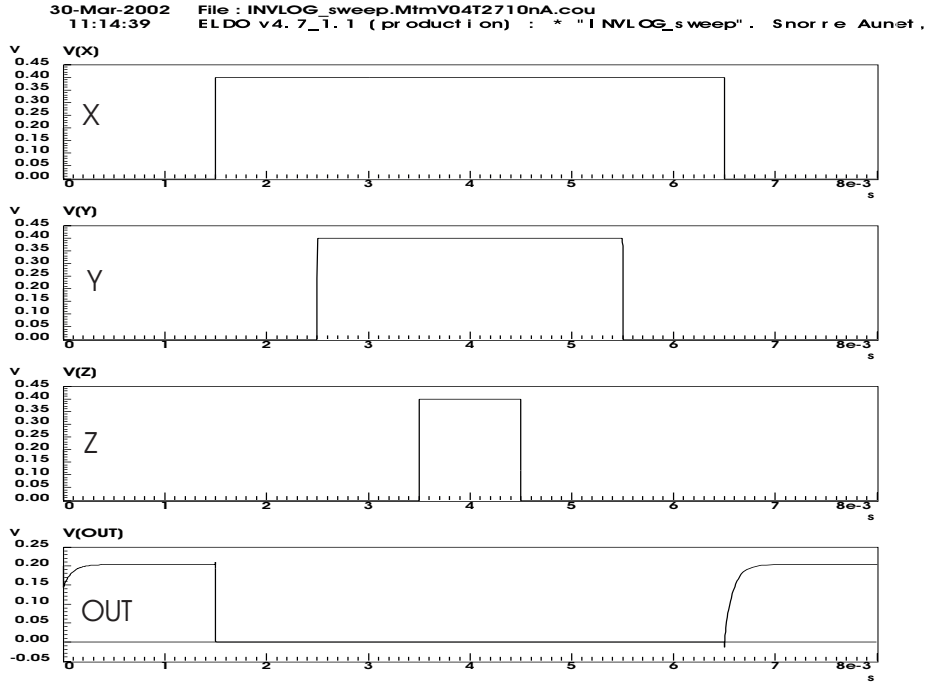

Figure 7.26: Inverter-only logic. When one or more of the inputs go high, the output goes low. (V(X)=V_{dd} means the same as X="1".)

When there are few inputs, like for the inverter, design rules allow for larger capacitances between one input and floating gate(s) than if there are several inputs. This in turn, may allow for an increased voltage gain of the single 2-MOSFET building block, which in many cases is important.
Another threshold gate composed of inverters was found in [BuTh01]. In that case an extra inverter was used to restore full output swing.

7.6 Theoretical lower voltage bound and voltage gain.

A minimum value of V_{dd} needed to obtain a certain maximum voltage gain, of absolute value G_{max}, can be written [Sven97]:

$$V_{dd} = 2 \cdot kT/q \cdot \ln(1 + n \cdot G_{max}) \tag{7.11}$$

This is in agreement with [ScPi96]. If considering room temperature, kT/q = 25.8 mV. When the number of inputs to a floating gate grows, the possible change of voltage on the floating gate, due to a change in the input voltage of an individual input, decreases. m binary inputs to a floating gate give $m + 1$ possible states, or steady-state voltage levels, on that floating gate.

Figure 7.27 shows the V_{dd} needed for a certain voltage gain absolute value, G as a function of the number of capacitively coupled inputs, m, to the floating gate in ideal and unrealistic cases. Here equally capacitively weighted inputs to the floating gates is an assumption, an equal number of capacitively weighted inputs to both the PMOS and the NMOS transistors is also an assumption. The slope factor is the unrealistic n = $1/\kappa$=1.0.

$$y = 2m \cdot 25.8 \cdot \ln(1 + n \cdot |G|) \tag{7.12}$$

Other limitations not taken into account in this equation are, as has been seen, that the capacitors between inputs and floating gates allow less than 100% of the input signals through to the floating gates. The parasitic capacitances between the floating gates and drains limits the voltage gain.

An analog inverter (figure 2.21) has a maximum voltage gain of -1, based on equal capacitances between the input and the floating gates and the drain and the floating gates. From [BeAu01e] the analog inverter has a maximum voltage gain equal to -1/M, where M=C_r/C_n. If the capacitances between the input and the floating gates are C_n=75 fF, while C_r=7.5 fF, the maximum voltage gain would never exceed -75/7.5=-10.

The parasitic capacitance between gate and drain also works against the desired change of voltage level at the floating gate, especially in the region with the highest voltage gain. When, for example, the input voltage increases from V_{ss} towards V_{dd}, the voltages on the floating gates will increase. When the input voltage passes $V_{dd}/2$, the output voltage shifts from $V_{dd}/2$ towards V_{ss}. At the same time the output voltage is coupled to the floating gates via the capacitances between floating gates and drain /

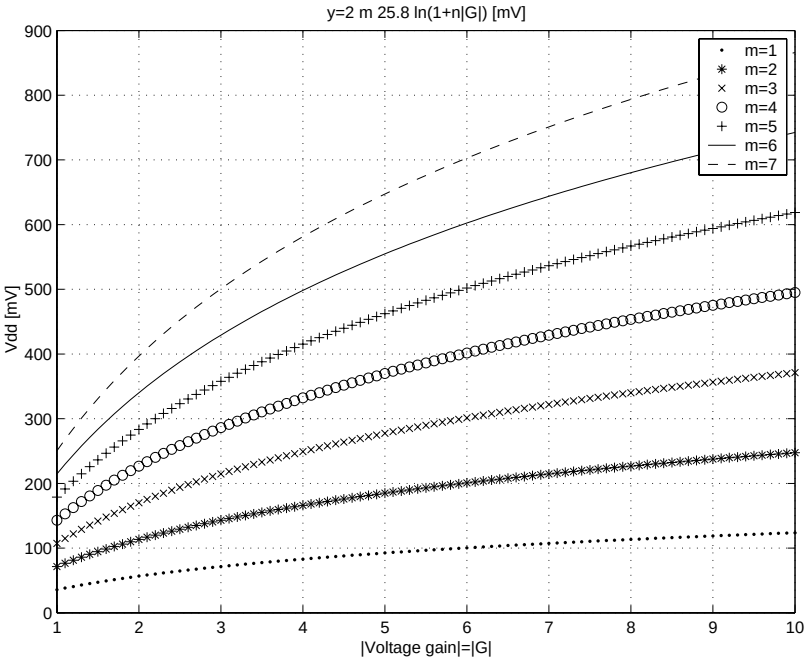

Figure 7.27: Minimum theoretical power supply voltage at room temperature, as a function of the absolute value of the voltage gain, as a function of fan-in.

output. This feedback mechanism reduces the maximum voltage gain. In this way the floating gate gets a capacitively weighted unwanted feedback from the output. Unfortunately the voltage change on the floating gate due to a transition on the output voltage impairs the voltage gain increasingly with an increasing number of capacitively weighted input signals.

Even if figure 7.27 is too optimistic for practical implementations , it can provide some useful information, together with the above equation.

It gives some estimates about the lower bounds for CMOS technology, as can be shown with a few examples, and compared to some examples in the literature:

To maintain a gain of -1 at room temperature, using an inverter, $m = 1$ and an ideal slope factor, the minimum V_{dd} is $2 \cdot 1 \cdot 25.8 \cdot \ln(1 + 1 \cdot 1)$ [mV] = 35.8 mV. In the ideal case this is sufficient for running a ring oscillator [ScPi96]. The lowest operable voltage for $n = 1$ according to [BrBr01] is 52 mV at room temperature. Assuming a voltage gain of 10, and an n of 1.5, [Sven97] decided that V_{dd}=140 mV was a "reasonably good estimate

of the lower bound for CMOS technology".

The more realistic n=1.5, for CMOS, gives a minimum V_{dd} of 47 mV under otherwise identical assumptions.

The P5N5 element would, at room temperature and n=1.5, need at least a V_{dd} of $2 \cdot 5 \cdot 25.8 \cdot \ln(1 + 1.5 \cdot 1)$ [mV] = 236.4 mV under these optimistic assumptions. Even for a relatively high input voltage the best voltage gain assuming C_n=122 fF and C_r=7.5 fF would be about 16. Since there are 5 inputs of 25.5 fF each, the maximum gain might be reduced to -25.5 / 7.5 = -3.4. If only 50% of the input voltage swing gets through to the floating gate, the voltage gain could be expected to be divided by two, reducing it to -1.7. The measurements for the P5N5 circuit showed a voltage gain of about -1, but then there were some problems with the measurement setup, potentially reducing the effective V_{dd}.

If a P15N15 element were to be built, in an attempt to make a 4-bit ADC, using n=1.5, m=15 and for whatever reason each element having a voltage gain of at least 4; could it operate with a V_{dd} of 0.8 V even with zero capacitances between gates and drains ?

Then $2 \cdot 15 \cdot 25.8 \cdot \ln(1 + 1.5 \cdot 4)$ [mV] = 1506 [mV]. The answer is no. If gain were reduced to -1, 709 mV would suffice, according to the same sort of calculation. Then we have the C_{gd} and the restrictions regarding maximum size of the capacitances. To make such a circuit work in practice, under these conditions, one may need a new technology.

These simple models are not accurate, but such calculations may be used to get at least some idea about realism regarding performance for the circuits.

7.7 Implementing linear threshold functions

7.7.1 Linear threshold elements and neural networks

Circuits such as P1N3 and P5N5, proposed in this dissertation, are linear threshold elements [AuBe02a]. Linear threshold elements are basic processing units in neural networks [SiBr90], which are human attempts to imitate the computational power of networks of biological neurons.
Despite the power of digital computers they are not clever enough in the sense of a biological processing like seeing an object in the visual field, recognizing what it is, and taking proper action in real time. For biological systems, including humans, in general those are generally effortless tasks [Shib00]. Such tasks are extremely difficult even for state-of-the-art computers. The performance gap could never be narrowed by just increasing the clock frequencies of MPU's, integration densities of memories and further sophistication of software programs, is an opinion of [Shib00].
The human cerebral cortex is estimated to consist of about 10 billion, relatively slow neurons, higly interconnected and operating in a massively paralell way [Hamm00]. Since a fundamental purpose of neurons is to integrate information from other neurons, the number of inputs received by each neuron is an especially important determinant of neuronal function. In the human nervous system the number of inputs received by each nerve cell ranges from 1 to about 100 000 [PuAu97]. It has been said [Mead89] that regarding implementation of silicon neural systems we are limited by the paucity of our understanding, especially when it comes to the organizing principles. Struggles with digital computers have taught us much about how neural computation is *not* done, partly because a large proportion of neural computation is done in an *analog* rather than a digital manner [Mead89].
The classic model of a neuron is a linear threshold device, which computes a linear combination of the inputs, compares the value with a threshold, and outputs +1 or (-1) if the value is larger than the threshold [SiBr90].
Real neurons are found in the biological nervous systems, including the human brain. Human brains are by far superior to computers in solving hard problems such as combinatorial optimization and image and speech processing, although their basic building blocks are several orders of magnitude slower, which have boosted interest in the field of artificial neural networks [Boho98], [Hamm00].
While neural networks have found wide application in many areas, the limitations and behavior of such networks are far from being understood

[SiBr93].

7.7.2 Mathematical definition of the FGUVMOS linear threshold gates

Using equations from [CeAl00] and interchanging the output values gives the following relations for operation of the proposed threshold gates:

$$Y = 0 if \sum_{i=1}^{n} W_i X_i \geq T \tag{7.13}$$

$$Y = 1 if \sum_{i=1}^{n} W_i X_i \leq T \tag{7.14}$$

$X_i \in 0, 1, i = 1, ..., n$ are the binary input variables and $Y \in 0, 1$ is the Boolean function realized by the threshold gate. W_i is the weight corresponding to the ith input variable X_i. T represents the gate threshold and is generally a real number satisfying

$$0 \leq T \leq if \sum_{i=1}^{n} W_i \tag{7.15}$$

7.7.3 Circuit complexity and costs of linear threshold gates

Assuming that each threshold gate can be built at a cost comparable to that of traditional AND, OR, NOT ("AON") logic, neural networks can be much more powerful than traditional logic circuits [SiBr90], [SiBr93]. According to [BuTh01], threshold logic reduces the amount of interconnect, but uses more complex basic elements. They show a circuit using a two-phase clocking scheme and 25 transistors. Since FGUVMOS gives the opportunity to enhance functionality per transistor, such a view might be revalued.

Boolean functions that can be realized by neurons are called linear threshold functions, and a network of neurons can implement any Boolean function [SiBr90].

In CMOS the production costs of a chip has a strong dependency on the chip area. Since the area of basic linear threshold elements, like P5N5 and P1N3, are in the same order as normal gates, the costs might be comparable.

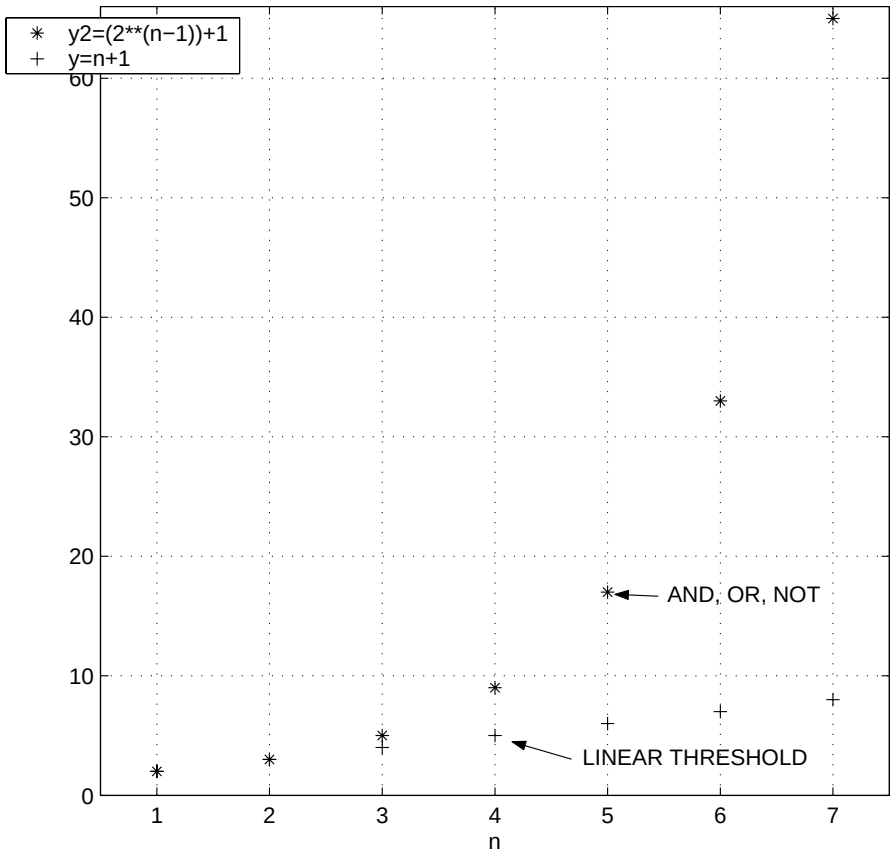

Figure 7.28: Number of gates necessary to implement certain functions, as a function of number of bits, n

For some functions, like XOR, the number of elements in a traditional AON circuit will grow exponentially with the number of bits in the input, while when implemented using linear threshold elements the number of gates are linear in the number of input bits [Boho98]. Generally, a depth-2, AON circuit computing XOR of n bits requires at least $2^{n-1}+1$ gates. A Linear Threshold circuit needs only n+1 gates. Figure 7.28 illustrates these relationships for n=$1, \cdots, 7$.

SUM equals the 3-input XOR function. According to [Boho98], using linear threshold logic one should need 3+1 gates for this function. Decomposing the 8-bit FULL-ADDER from [AuBe01d] gives 3 gates only, one less than predicted in [Boho98].

Another example of potential use of threshold logic [SiBr90] is: Whereas

any logic circuit of polynomial size (in n) that computes the product of two n-bit numbers requires unbounded delay, such computations can be done in a neural network with "constant" delay. The product of two n-bit numbers and sorting of n n-bit numbers can be computed by a polynomial-size neural network using only 4 and 5 unit delays, respectively. Unit delay is equal to a "depth" of one for an artificial neural network [SiBr93].

Symmetric Boolean functions depend only on the sum of input values, and since the parity function is symmetric it can be computed in two layers of a neural network whereas it takes unbounded delay to compute parity in a logic circuit [SiBr90].

For many years the topic of linear threshold logic has been approached in two different ways: theory on computational circuit complexity on one hand, and hardware implementation on the other. There has been very little interaction between the two approaches, as was stated in [Boho98].

7.7.4 A new type of threshold gate

It has been said in [CeAl00] that there are two types of threshold gates, which are neuron-MOS and Capacitive Threshold-Logic Gates (CTL) methods. The FGUVMOS elements should therefore represent something different in this field. Neuron-MOS [ShOh95], [ShOh93] has been operating in the classic above threshold regime. Neuron-MOS is utilized in VLSI systems inspired by a psychological brain model, where the "core of the intelligent data processing is directly carried out in the VLSI hardware", according to [Shib00].

7.7.5 The pFET synapse transistor - for neural networks in hardware, or systems-on-a chip.

Research on "synapse transistors" have been going on for several years [DiMa95], [DiHs02]. Such a structure is shown in figure 7.29. Fowler-Nordheim tunneling is used to remove electrons from the floating gate of the "pFET synapse", and impact-ionized hot-electron injection for adding electrons to the floating gate [DiHs02].

Contemporary implementations of neural networks and machine-learning algorithms are almost entirely software based. If hardware versions could be built, there are prospects for huge performance gains [DiHs02]. Synapses and neurons in brains encode and process information using electrical and chemical signalling very effectively, under tight power and supply voltage constraints. The synapses and neurons are poorly matched accross nerve

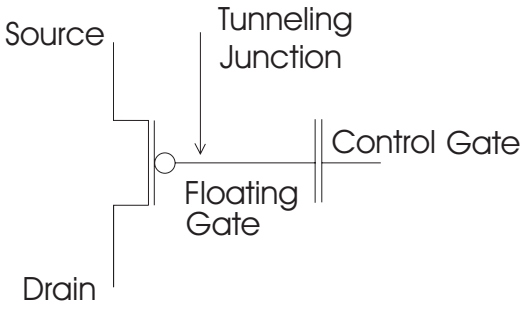

Figure 7.29: Circuit symbol for pFET synapse transistor [HsFi02].

tissue, and degrade over life. There are also no common supply voltages or common ground levels [DiHs02]. Local adaption to tune performance of neural circuits is believed to be the key component of intelligent behavior and efficiency of biological organisms [DiHs02]. The local adaptivity of the pFET synapses are believed to be a step towards building better biologically inspired adaptive systems than available today. It is also said in [DiHs02] that: "The circuits we have built to date are small and simple, primarely because they represent our baby steps in exploring a new technology. As we and others learn how to use local adaption effectively, the circuits we build will mature. Our confidence in this technology is rooted not in what we have built to date, but rather in the existence proof provided by neurobiology. ... We are merely copying nature."
Neural Networks are not the only arena for the pFET synapses. Technology scaling increases density and speed of digital CMOS, but provides poor transistor matching and absence of high-valued resistors, high-Q inductors or linear capacitors especially problematic for the analog parts of mixed-signal systems [DiHs02].
In system-on-chip applications, implemented in digital CMOS, the synapse transistors are used for performance tuning of circuitry during normal operation. Example include a digital-to-analog converter with 6-bit intrinsic accuracy that trims electrically to 14 bits [DiHs02]. They have also been used to store direct currents and voltages, match multiple current sources, set operating points for capacitive-feedback operational amplifiers, balance mixers and store nonvolatile memories [DiHs02]. Analog circuits can be trimmed to 16-bit accuracy by adjusting the floating gate charge [DiHs02]. A company based on the ideas of "self-adaptive-silicon" and floating gate transistors is the Seattle-based Impinj [Impi02].
Synapse transistors have technological and reliability issues similar to other

non-volatile memory technologies, of which the most critical are tunneling- and injection induced damage to the gate oxide, and charge leakage off the floating gate [DiHs02]. Oxide damage has not been an issue according to [DiHs02], though it do limit the number of read/write cycles. Scaling gate oxides to less than 7 nm causes the floating gate to leak, a problem that can be avoided by using thicker oxide, available in most dual gate-oxide processes [DiHs02]. This might be a little bit more conservative estimate than the 5 nm given in [Gjer02], [HaslXY]. It is said in [HaslXY] that "very good floating gates can be made in the near future, and that classic theory holds reasonably well down to L=100 nm", approximately the technology planned for year 2006.

7.7.6 FGUVMOS linear threshold elements

If we can use analog devices to build threshold gates with a cost that is comparable to that of AND, OR logic gates, we can compute many basic functions much faster than using traditional circuits [SiBr90]. The area of FGUVMOS threshold gates can probably be said to be comparable to traditional circuits, since it does not differ by orders of magnitude for basic circuitry. The delay for a basic FGUVMOS linear threshold gate can be several orders of magnitude higher than traditional logic. In comparison, the typical switching time for biological neurons is on the order of a few milliseconds [Hamm00]. There are also added costs for UV-programming, even if the complexity of FGUVMOS circuits can grow from the few transistors proven in silicon of today.
There are more or less powerful threshold elements. How elements in FGUVMOS technology fit in has not been among important considerations in this work, but could be researched further.
Only late in this work it was noticed that there could be a natural link between the hardware implementations of circuitry that were perceived mainly from a digital point of view, like in [AuBe01d], [AuBe01e], and implementations of synthetic neural networks.
Since "neuron MOS" threshold elements are used as integral elements of biologically inspired VLSI circuits [Shib00], [ShYa01], FGUVMOS linear threshold elements might also be used in this context, if larger systems are to be proven in silicon. Exponential voltage-current relationships in FGUVMOS circuitry are common with building blocks of pioneering biologically inspired systems in [Mead89]. The low-power operation of FGUVMOS is probably closer to its biological ideal as well.

On the other hand the maximum size of the circuitry proven in silicon in this thesis consists of 2 transistors. If that number could be doubled every 18 months, one still has about 30·18 months, or between 50 and 60 years of development to catch up with the number of neurons in the human brain, for example.

The synapse transistor in figure 7.29 differs from FGUVMOS transistors since it rely on other mechanisms to add and subtract charge on the floating gate, and that adjustments of floating gate charge can go on under normal operation. The FGUVMOS transistors are not programmed during normal operation, but once prior to a longer time of operation, though they can be reprogrammed. The synapse transistors get their current level, for a given control gate input, adjusted in real time from a change of the floating gate charge. The current level of some of the circuits presented in this work could be adjusted by changing the voltage on one or several of the inputs, under normal use, but after an UV-programming.

A question that pops up is to which extent FGUVMOS transistors could switch places with synapse transistors in CMOS circuitry. In such a context the performance of FGUVMOS viewed with "digital eyes" only might be insufficient or even wrong, mainly because the brain uses principles different from digital logic. When that is said, one might also add that the brain probably also use principles very far from what can be mimicked by linear threshold elements. Support for such a view might be found in for example [Koch97] and [DiHs02].

7.8 Possibilities regarding simultaneous UV- programming of different circuit structures

7.8.1 Problems connected to programming different basic circuit structures on the same die

There are probably several good reasons for keeping structures as regular as possible. It improves matching of components and could probably ease the UV-programming.

One of the good properties one hopes to utilize from UV-programming is being able to share programming lines between circuitry on a chip, or even a wafer, with virtually no additional programming circuitry [LaWi96], [BeWi98], as opposed to most existing floating-gate techniques [LaWi96].

If there is to be a minimum area overhead using the FGUVMOS technique, all circuits should share common programming lines / rails. If that is not possible, the number of programming lines and additional pads or programming circuitry should be kept to a minimum, since they both increase the costs and complicate programming and testing, and make circuits more difficult to use.

Several attempts have been made lately to UV-program different FGUVMOS building blocks on the same die. This is not easy, as will be argued.

Figure 7.31 shows measured equilibrium currents as a function of one of the programming voltages, and is approximate numbers taken from in [Gund00] p. 22. This data showed the impossibility of reaching the same equilibrium current for the same programming voltage for three different, widely used, basic FGUVMOS circuit structures. If these three circuits had the same programming voltages, it would have been impossible to make them work in concert, at least in the nA to μA range.

Support for this wiew might be found in [Bahr01] p. 29 where it is expressed that it has been difficult to achieve low output currents when programming circuits consisting of more than one element. Several types of elements were considered, not several instances of the same circuitry.

In the conclusion of the same thesis [Bahr01] it was stated that: "For testing purposes it could be wise to implement the whole circuits , separated subcircuits and even single transistors. So far our experience shows that there are some problems with programming the circuit as a whole. Particularly if we want to achieve an equilibrium current in order to attend weak inversion behavior. Different subcircuits, in our case demand different programming voltages."

In [Loms02] it is stated this way "Another thing worth noticing, and

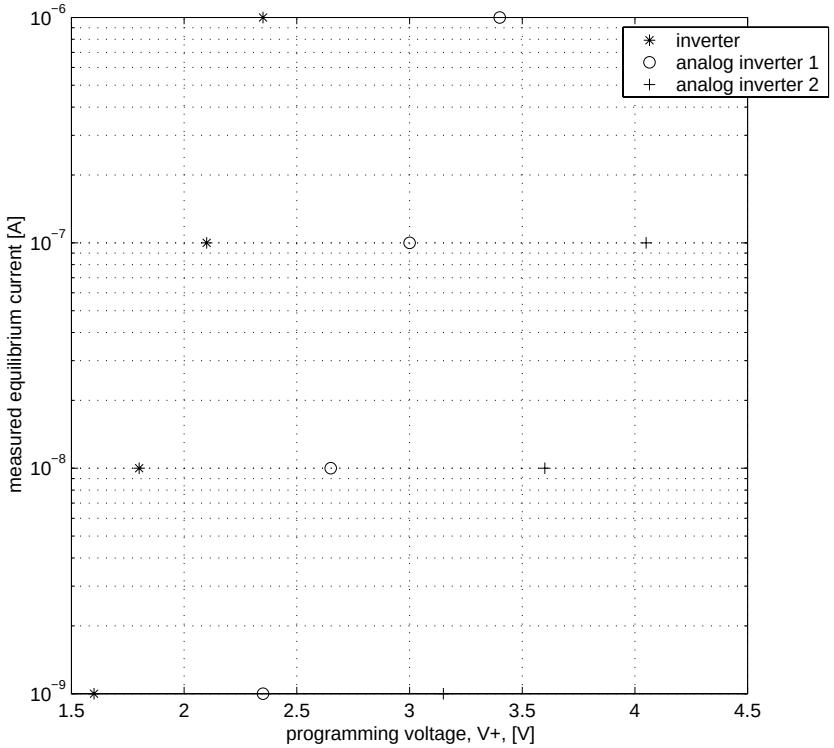

Figure 7.30: Measured equilibrium currents for different programming voltages for 3 different FGUVMOS circuits [Gund00].

so far considered a problem, is that small differences in layout may cause different switching point, and decades of differences in current levels. This makes it hard to program multilevel circuits not only using identical building blocks".

Attempts have also been made to use the sizing and shaping of the UV-holes to program different effective threshold voltages seen from driving nodes [Dani01]. In this way different UV-activated conductances, examplified in figure 7.31 have been made, with different stored charges on the floating gates as a result. It was concluded [Dani01] that correlations between the UV-hole sizing and the current levels of the different transistors were not easy to find. Especially the PMOS transistors were difficult to program, which has been pointed out earlier [BeLa99a].

The exponential relationships in the actual area of operation of the transistors might make such an attempt especially difficult. There are also inherent production spread and matching problems.

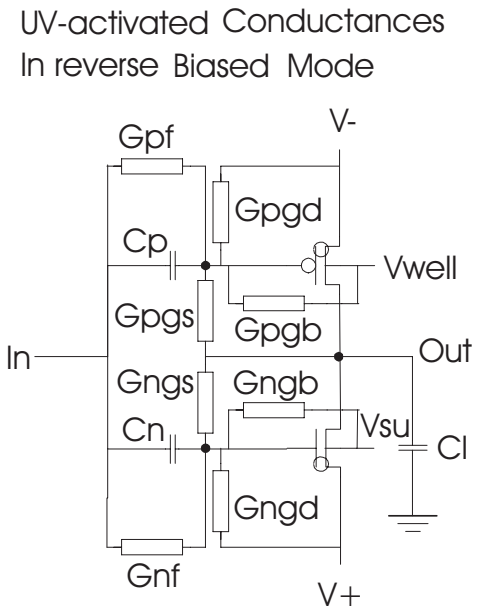

Figure 7.31: UV-activated conductances [BeWi99].

The rules for this process do not allow UV-holes of lesser size than 15 μm x 15 μm, and there might be a danger that the UV-hole is getting partly closed at the repeated deposits of the passivation layer [Flat01]. If a better process were used, and more research were done, it might be possible to control threshold voltages satisfactorily. Some spread in process parameters may also affect the programming time [Flat01]. It is mentioned [Flat01] that in spite the fact that the layout of almost similar ring-transistors for two different chips from [Dani01] and [Flat01], the programming time of circuitry on the former was up to 4 hours, while 15 to 20 minutes was enough for the latter. A hypothesis was that the UV-holes were partially "cladded" for the chips in [Dani01]. If that is true it could be impossible to control effective threshold voltages by the shaping and sizing of UV-holes, unless they can be under better control while being fabricated.

In [LaWi96] a different UV-programming procedure was used, and an Orbit process without nitride in the passivation. (The "FGUVMOS" procedure appeared in [BeLa97b]) For FGUVMOS circuits reported to our knowledge, the AMS 0.8 and 0.6 μm CMOS processes have been used. These are standard double-poly CMOS processes. A viewpoint in [Flat01] was that if the FGUVMOS principle should be used commercially, some

deal with a producer of integrated circuits should be done to improve quality.

7.8.2 Possibilities towards programming different basic circuit structures on the same die

To improve matching and hopefully to increase chances to be able to make more complicated FGUVMOS circuitry than reported until now, an approach using as few different FGUVMOS basic building blocks as possible has been suggested [AuBe01d], [AuBe01e].

Another nice property using this attempt is that the functionality per transistor increases a lot compared to the method used for the FULL-ADDER in [BeWi99], for example. The number of transistors to produce the inverted CARRY gets reduced from 22 to 2 [AuBe01a], and the number of capacitors from 30 to 4 (or 6). This saves complexity and area of circuitry. On the downside there are decreased noise margins following from an increase in the number of capacitive inputs to the floating gates [BeWi97], for the circuits reported in [AuBe01d], [AuBe01e] and [AuBe02a], which is treated elsewhere in this work. To compensate for the increased number of inputs to the floating gates the V_{dd} level or the drawn capacitances between inputs and floating gates can be increased.
If the price of an increased number of capacitively inputs can not be paid, there might be other possibilities. One is to build the entire circuitry from gates with a maximum of two inputs to a floating- gate only, for example using the P1N2 element. This comes to the costs of added chip area, but the matching properties should be relatively good due to the repetition of the same block over larger portions of the chip.

An approach using a maximum of one capacitively coupled input to a floating gate is to use the proposed inverter-only logic.

To improve matching, save chip area, ease programming and hopefully to be able to build more complicated FGUVMOS circuitry, it is probably a good thing to use as few different modules as possible, at least until someone finds a better solution.

Measurements for 12 inverters in series, sharing V_{dd}, V_{ss}, and substrate potentials

The FGUVMOS elements mentioned here can all be configured to become an inverter. Measurements and simulations from using the P5N5 and P1N3 elements as inverters were shown in figures 2.14 and 6.9, respectively. The P5N5 circuits used in the full-adders and the P7N7 circuit, for example,

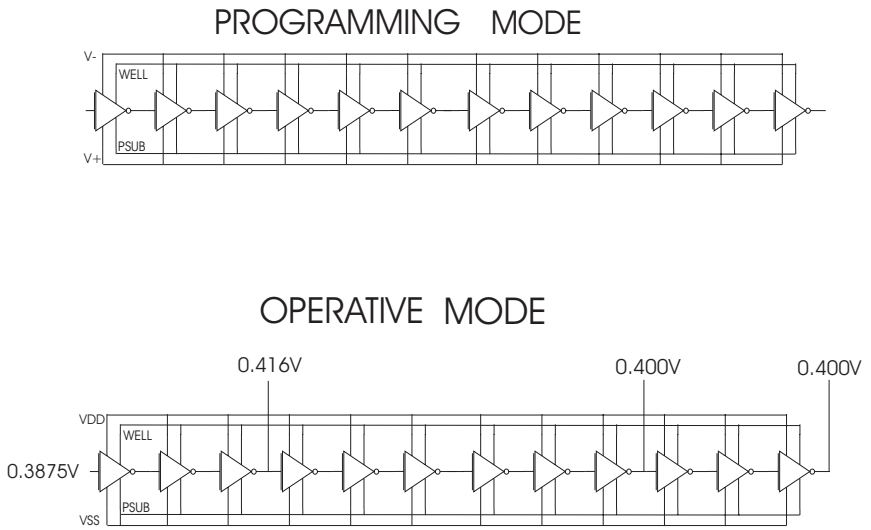

Figure 7.32: Measured switching points for 12 inverters programmed with common rail-, well- and substrate potentials [Gund00].

used for the 3-bit ADC uses exactly the same block repeatedly, with only interconnects differing. A hypothesis is that programming elements like the P5N5 in series should be practically similar to programming a row of standard FGUVMOS inverters in series, which has been done successfully.

In [Gund00] twelve identical inverters were coupled in series, with all drawn capacitors being 18.4 fF and W/L=10μm/0.6μm, and programmed using common programming voltages on all rails and wells. The outputs were afterwards measured at the switching point for inverters number 3, 9 and 12 in the chain. Inverter number 3 was used as a reference. The offsets between this reference point and the voltages on outputs of number 9 and 12 were less than 10 mV. Programming identical inverters saves chip area, and is probably far simpler than programming several different building blocks at a time. Our elements should resemble this type of inverter chain, while being UV-programmed, if coupled in series. We hope they can make programming of digital circuits easier than previous approaches, and help us come able to build more complex systems using UV-programming, and save chip area at the same time. To try and program circuits other than the inverters, coupled in series while using common wires for rails and substrates could be very interesting, in search for proofs for the new concept.

Chapter 8

Conclusions

8.1 Major Contributions

One contribution to UV-programmable floating-gate (FGUVMOS) circuits is a new class of CMOS circuit building blocks that are real time reconfigurable. This means that the Boolean function can be changed during normal operation, without doing a new UV-programming, but by applying certain voltages on one or more inputs chosen to be used as control inputs instead of handling signals. It has been demonstrated theoretically, by computer simulations and implementations in a standard CMOS technology, that a FGUVMOS circuit containing 2 transistors, working in weak and moderate inversion, can compute the CARRY', NAND, NOR and INVERT ("NOT") functions.

Earlier FGUVMOS approaches have used dedicated circuitry for each basic function, such as NAND, NOR and INVERT, and built more complicated functions, like the inverted CARRY for binary addition, by wiring together several such building blocks. The new approach presented is possible by taking advantage of the amplifying characteristics of the MOSFET transistor and using the transistors as something else than the switching function they perform in traditional digital logic circuits. This makes it possible to create some circuit structures using significantly fewer active components than most known. By operating the circuits outside the classical regime of operation of the MOSFET transistor, this can lead to very low power consumption compared to most known CMOS circuitry.

A second contribution is the idea of using as few different basic building blocks as possible for implementing FGUVMOS circuitry. This improves the matching of components. Relative matching of CMOS components is far better than absolute matching. This means that if one tries to make

two identical components like capacitors of 100 fF each, the chances are relatively good they would become up to +/- 20 % away from the desired value. The nice thing about it is that statistically both would get very similar values, for example 91.3 and 91.8 fF, a quality which can be utilized by designers. The matching properties have consequences for every component on a CMOS chip or wafer. In weak inversion important parameters are exponentially dependent on parameters like temperature. Matching should probably be as good as possible. Here is how the new circuits come in. By using the same basic circuit extensively, with only external metal wiring differing, the inherent matching properties should be the best attainable for a given technology if handled with care. This should increase possibilities of making working circuitry.

UV-programming of circuitry consisting of several different building blocks at gate or basic circuit level has proven to be very difficult. It is often not possible to make circuitry consisting of several basic different building blocks work as intended if common programming voltages on rails and substrates are to be used. When this is impossible, separate programming voltages on rails and wells are needed for each basic building block. It complicates things, and increases the amount of circuitry on chip, as well as I/O cells and laboratory instrumentation. Things simply get far more complicated. Measurements for identically drawn inverters with a logical depth of twelve is believed to be promising for the new approach using similar basic building blocks only. This approach hopefully makes it possible to make larger FGUVMOS circuits than previously and prove them in silicon. Such an approach might be used using any of the P1N2, P1N3, P3N3 or P5N5 circuits.

8.2 Secondary results

Various PMOS and NMOS transistor building blocks were demonstrated by implementation in silicon as well as measurements on the laboratory. Based on theory, methods to improve important parameters such as output resistance and transconductance are suggested. The layout is important, including Width and Length for transistors and especially the sizing of the drawn capacitances between inputs and floating-gates. Regarding operational speed there seems to be an optimum value of the drawn capacitances. The voltage gain is also strongly dependent on the sizing of capacitances.

Different threshold logic building blocks like P1N3 and P5N5 were explored theoretically, by circuit simulations and laboratory measurements. Functionality as well as ultra-low voltage / low-power potential are studied.

Measurements demonstrated an inverter working down to a supply voltage of 93 mV. Measurements demonstrated reprogrammability and reconfigurability of proposed circuit elements.

Comparisons with the FULL-ADDER function using standard cells in the same technology running at 1 MHz indicates that the standard cells used 2500 times as much energy per switching as an 8-transistor FULL-ADDER implementation suggested.

On circuit level, an 8-transistor FULL-ADDER and 6-transistor 3-bit analog-to-digital converter were used as examples of how the number of active devices might be cut from 28 and 174 respectively, compared to standard CMOS implementations.

A memory cell was shown, in form of a D-latch, using a universal element. Having memory and combinatorial logic enable Finite State Machines, in theory.

New circuits suggested form a new class of threshold logic circuits, which can be used both for implementing digital logic and as parts of neural networks.

8.3 Further Work

The work has created a lot of new questions. On the fundamental technology and device level there is probably a lot to be done. More verification of circuit concepts presented herein should be done by measurements, and circuitry could be improved along suggested guidelines and in other ways.

Maybe backbiasing, as in [SvMa00], could be combined with multiple-input linear threshold elements using only UV-erase, to avoid the "exotic" UV-programming. Or an element like P1N3 could be used with the input to the PMOS adjusted at certain intervals [YtAu02] to adjust for changes in temperature or supply voltage to be able to keep up regular operation of circuity. Such approaches may reduce risks of the FGUVMOS approach, while enabling utilization of the ultra low-power potential of the technology and the increased functionality per transistor compared to traditional digital approaches.

Bibliography

[AlHo87] P. E. Allen, D. R. Holberg *CMOS Analog Circuit Design* Holt, Rinehart and Winston, The Dryden Press, Sqaunders College Publishing, ISBN 0-03-006587-9, 1987.

[AMS97] Austria Mikro Systeme International AG *0.6-Micron Standard Cell Databook* 3/1997

[AMS98] Austria Mikro Systeme International AG *0.6 um CMOS CUP Process Parameters* Document No. 9933011, Rev. B, October 1998.

[AMS9X] Austria Mikro Systeme International AG *0.6 um CMOS Design Rules* Document No. 9933025, Rev. 2.0.

[AnBo91] A. G. Andreou, K. A. Boahen, P. O. Pouliquen, A. Pavasovic, R. E. Jenkins, K. Strohbehn *Current-Mode Subthreshold MOS Circuits for Analog VLSI Neural Systems* IEEE Transactions on neural Networks, Vol. 2, pp. 205-213, March 1991.

[AnGo97a] J. Ramirez Angulo, G. Gonzalez-Altamarino *A new programmable logic family using multiple-input floating-gate transistors* Proceedings of the 40th Midwest Symposium on Circuits and Systems, 1997, Vol. 1, pp. 354-357.

[AnGo97b] J. Ramirez Angulo, G. Gonzalez-Altamarino, S. C. Choi *Modelling multiple-input floating-gate transistors for analog signal processing* Proceedings of the IEEE International Symposium on Circuits and Systems, 1997, Vol. 3, pp. 2020-2023.

[AuBe01a] S. Aunet, Y. Berg, O. Tjore, Ø. Næss, T. Sæther *Four-MOSFET Floating-Gate UV-Programmable Elements for Multifunction Binary Logic* Proceedings of the 5th World Multiconference on Systemics, Cybernetics and Informatics, Orlando, FL, USA, Volume 3,2001, pp 141-144..

[AuBe01b] S. Aunet, Y. Berg, Ø. Næss, T. Sæther *Novel Reconfigurable Two-MOSFET UV-Programmable Floating-Gate Circuits for CARRY', NAND, NOR and INVERT functions* Proceedings of the 8th IEEE International Conference on Electronics, Circuits and Systems, 2001. Volume 2, pp 581-584.

[AuBe01c] S. Aunet, Y. Berg, T. Ytterdal, Ø. Næss, T. Sæther *A Method for Simulation of Floating-Gate UV-Programmable Circuits with*

Application to Three New 2-MOSFET Digital Circuits Proceedings of the 8th IEEE International Conference on Electronics, Circuits and Systems, 2001, Volume 2, pp 1035-1038.

[AuBe01d] S. Aunet, Y. Berg, Ø. Næss, T. Sæther *A New Universal UV-Programmable Floating-Gate Digital Element with Application to an 8-Transistor FULL-ADDER and a D-Latch* Proceedings of the 19th IEEE Norchip Conference, Kista, Sweden, 12-13 November 2001, pp. 222-227.

[AuBe01e] S. Aunet, Y. Berg, T. Sæther *A New 2-MOSFET Universal Floating-Gate Element for Reconfigurable Digital Logic* Proceedings of the 19th IEEE Norchip Conference, Kista, Sweden, 12-13 November 2001, pp. 240-245.

[AuBe02a] S. Aunet, Y. Berg, T. Sæther *Floating-Gate Low-Voltage/Low-Power Linear Threshold Element for Neural Computation* Proceedings of the 2001 IEEE International Symposium on Circuits and Systems, Arizona, 2001, Volume 2, pp 528-531.

[Aune02] S. Aunet *Netlists Extracted from two FGUVMOS Integrated Circuits* FYS.EL.-rapport 2002:2, available at the Department of physical electronics, Norwegian University of Science and Technology.

[Bahr01] R. Bahr *A Design of Linear Four Quadrant Analog Multipliers Using Floating-Gate Transistors* thesis for the cand. scient. degree, University of Oslo, Faculty of Mathematics and Natural Sciences, Department of informatics, May 2001.

[BeAu01a] Y. Berg, S. Aunet, Ø. Næss, M. Høvin *Extreme Low-Voltage Floating-Gate CMOS Transconductance Amplifier* Proceedings of the 2001 IEEE International Symposium on Circuits and Systems, 2001, Volume 1, pp 37-40.

[BeAu01b] Y. Berg, S. Aunet, Ø. Næss, M. Høvin *Floating-Gate CMOS Differential Analog Inverter for Ultra Low-Voltage Applications* Proceedings of the 2001 IEEE International Symposium on Circuits and Systems, 2001, Volume 1, pp 9-12.

[BeAu01c] Y. Berg, S. Aunet, Ø. Næss, M. Høvin *Exploiting sinh and tanh shaped Ultra Low-Voltage Floating-Gate Transconductance Amplifiers to Reduce Harmonic Distortion* Proceedings of the 2001

IEEE International Symposium on Circuits and Systems, 2001, Volume 4, pp 838-841.

[BeAu01d] Y. Berg, S. Aunet, Ø. Næss, M. Høvin *A Novel Low-Voltage Floating-Gate CMOS Transconductance Amplifier with SINH(TANH) Shaped Output Current* Proceedings of the 8th IEEE International Conference on Electronics, Circuits and Systems, 2001, Volume 2, pp 1035-1038.

[BeAu01e] Y. Berg, S. Aunet, Ø. Næss, H. Gundersen, M. Høvin *A 0.3 V Floating-Gate Differential Amplifier Input Stage with Tunable Gain* Proceedings of the 8th IEEE International Conference on Electronics, Circuits and Systems, 2001, Volume 2, pp 1035-1038.

[BeAu02] Y. Berg, S. Aunet, Ø. Næss, Ø. Hagen, M. Høvin, *A Novel Floating-Gate Multiple-Valued CMOS Full-Adder* Proceedings of the 2001 IEEE International Symposium on Circuits and Systems, Arizona, 2002, Volume 1, pp 877-880.

[BeKe93] R. G. Benson, D. A. Kerns *UV-activated Conductances Allow For Multiple Time Scale Learning* IEEE Transactions on Neural Networks, Vol. 4, No. 3, pp. 434-440, May 1993.

[BeLa97] Y. Berg, T. S. Lande, S. Næss *Low-Voltage Floating-Gate Current Mirrors* Proceedings of the the Tenth Annual IEEE International ASIC Conference and Exhibit (ASIC), Portland, OR, USA, 7-10 Sept. 1997, pp 21-24.

[BeLa97b] Y. Berg, T. S. Lande *Programmable Floating-Gate MOS Logic for Low-Power Operation* Proceedings of the IEEE International Symposium on Circuits and Systems, Hong Kong, June 9-12, 1997, Vol. 3, pp. 1792-1795.

[BeLa99a] Y. Berg, T. S. Lande *Area Efficient Circuit Tuning with Floating-Gate Techniques* Proceedings of the the IEEE International Symposium on Circuits and Systems, Orlando, FL, USA, 1999, Vol. 2, pp. 396-399.

[BeLa99b] Y. Berg, T. S. Lande *Tunable Current Mirrors for Ultra Low Voltage* Proceedings of the the IEEE International Symposium on Circuits and Systems, Orlando, FL, USA, 1999, Vol. 2, pp. 17-20.

[BeLa01b] Y. Berg, T. S. Lande, Ø. Næss *Programming Floating-Gate Circuits with UV-Activated Conductances* IEEE Transactions on Circuits and Systems II: analog and digital signal processing, special issue on floating-gate devices, circuits, and systems, Vol. 48, Issue 1, pp. 12-19, January 2001.

[BeLa01a] Y. Berg, T. S. Lande, Ø. Næss, H. Gundersen *Ultra Low-Voltage Floating-Gate Transconductance Amplifiers* IEEE Transactions on Circuits and Systems II: analog and digital signal processing, special issue on floating-gate devices, circuits, and systems, Vol. 48, issue 1, pp. 37-44, January 2001.

[BeNa99a] Y. Berg, Ø. Næss *A four transistor rail to rail ultra lowvoltage transconductance amplifier* Proceedings of the 1999 IEEE Norchip Conference, Oslo, 1999, pp. 183-188.

[BeNa99b] Y. Berg, Ø. Næss *Symmetric ultralow-voltage floating-gate rail to rail four quadrant analog multiplier* Proceedings of the 1999 IEEE Norchip Conference, Oslo, 1999, pp. 158-164.

[BeNa02b] Y. Berg, Ø. Næss, S. Aunet, R. Jensen, M. Høvin *Novel Floating-Gate Multiple-Valued Signal to Binary Signal Converters for multiple valued CMOS logic* Proceedings of the 2001 IEEE International Symposium on Circuits and Systems, Arizona, May 2001, Volume 5, pp 385-388.

[BeNa02c] Y. Berg, Ø. Næss, S. Aunet, M. Høvin *A Novel Floating-Gate Multiple-Valued Signal to Binary Signal Converter* Accepted for the 9th IEEE International Conference on Electronics, Circuits and Systems, Dubrovnik, 2002.

[BeNa02d] Y. Berg, Ø. Næss, S. Aunet, J. Goplen Lomsdalen, M. Høvin *A Novel Floating-Gate binary signal to multiple-valued signal converter for multiple valued CMOS logic* Accepted for the 9th IEEE International Conference on Electronics, Circuits and Systems, Dubrovnik, 2002.

[BeWi97] Y. Berg, D. T. Wisland, T. S. Lande *Floating-Gate UVMOS inverter* Proceedings of the 1997 IEEE Norchip seminar, Talinn, 1997, pp. 193-200.

[BeWi98] Y. Berg, D. T. Wisland, T. S. Lande, S. Mikkelsen *Ultra Low-Voltage Digital Floating-Gate UVMOS (FGUVMOS) Circuits*

Proceedings of the 1998 IEEE International Symposium on Circuits and Systems, 1998, Vol. 2, pp. 37-40.

[BeWi99] Y. Berg, D. T. Wisland, T. S. Lande *Ultra Low-Voltage/Low-Power Digital Floating-Gate Circuits* IEEE Transactions on Circuits and Systems II, analog and digital signal processing, Vol. 46, Issue 7, pp. 930-936, July 1999.

[Boho98] V. Bohossian *Neural Logic: Theory and Implementation* Dissertation for the Ph.D. degree, California Institute of Technology, 1998.

[BrBr01] A. Bryant, J. Brown, P. Cottrell, M. Ketchen, J. Ellis-Monaghan, E. J. Nowak *Low-Power CMOS at Vdd=4kT/q* Proceedings of the Device Research Conference, Notre Dame, IN, USA, 2001, pp. 22-23

[BuCh96] J. Burr, Z. Chen, B. M. Baas *Stanford's ultra-low-power CMOS technology and applications* in Low-Power HF microelectronics a unified approach, edited by G. A. S. Machado, IEE Circuits and systems series 8, ISBN 0 85296 874 4, 1996.

[Burr95] J. Burr *Cryogenic ultra low power CMOS* Proceedings of the IEEE Symposium on low-power electronics, San Jose, USA, 1995, pp. 82-83.

[BuTh01] C. Burwick, M. Thomas, J. Dienstuhl, K. F. Goser *Threshold-Gates in Arithmetic Circuits* Proceedings of the 8th IEEE International Conference on Electronics, Circuits and Systems, 2001, Volume 2, pp. 909-912.

[CeAl00] P. Celinski, S. Al-Sarawi, D. Abbott *Delay Analysis of neuron-MOS and Capacitive Threshold-Logic* Proceedings of the 7th IEEE International Conference on Electronics, Circuits and Systems, 2000, Volume 2, pp 932-935.

[ChKi94] W. L. Choi, D. M. Kim, I. H. Choi *A New Measurement Technique for Capacitive Coupling Coefficients and 3-D Capacitance Characteristics in Floating-Gate Devices* International Electron Devices and Materials Symposium (EDMS), 1994, pp. 9-7-27 - 9-7-32.

[Dani01] R. Danielsen *Effektoptimalisering av digitale CMOS-kretser (floating-gate)* thesis for the cand. scient. degree, University of Oslo, Faculty of Mathematics and Natural Sciences, Department of informatics, May 2001.

[DiHs02] C. Diorio, D. Hsu, M. Figueroa *Adaptive CMOS: From Biological Inspiration to Systems-on-a-Chip* Proceedings of the IEEE, Vol. 90, No. 3, pp. 345-357, March 2002.

[DiMa95] C. Diorio, S. Mahajan, P. Hasler, B. Minch, C. Mead *A High-Resolution Non-Volatile Analog Memory Cell* Proceedings of the 1995 IEEE International Symposium on Circuits and Systems, 1995, Volume 3, pp 2233-2236.

[Eske02] F. Eskelund *Evolution of robust circuits in a simulated environment* Thesis for the sivilingeniør degree, Faculty of Information technology, Mathematics and Electrical Engineering, Department of Computer and Information Science, Norway, 2002.

[Flat01] Ole E. Flaten *Lavspennings-frekvens-delta-sigma-modulator* thesis for the cand. scient. degree, University of Oslo, Faculty of Mathematics and Natural Sciences, Department of informatics, August 2001.

[FuAr01] H. Fujiwara, M. Arimoto, S. Sudo, K. Kurooka, H. Nagasawa, T. Hiroshima, K. Mameno *A New Method for Measuring the Coupling Coefficient of a Split-Gate Flash EEPROM without an Additional Test Structure* Proceedings of the IEEE International Conference on Microelectronic Test Structures, March 2001, Vol. 14, pp. 43-46.

[FuOm98] P.M. Furth, H. A. Om'mani *A 500-nW Floating-Gate Amplifier with Programmable Gain* Proceedings of the Midwest Symposium on Circuits and Systems, Vol. 1, 1998, pp. 264-267.

[GoAv00] R. Goettsche, A. Avellan, D. Preikszat, W. Krautschneider *Multifunktionelle Logikeinheiten als rekonfigurierbare Redundanzelemente* Proceedings of the German workshop of the Information and Technology Community (ITG), "Mikroelectronics for the Informationtechnology", November 2000, pp. 151-155.

[GuAb98] A. Guyot, S. Abou-Samra *Low Power CMOS Digital Design* Proceedings of the Tenth International Conference on Microelectronics, December 1998, pp. IP6-I13.

[Gjer02] T. Gjerrestad *Konstruksjon av lavspennings pixler* thesis for the cand. scient. degree, University of Oslo, Faculty of Mathematics and Natural Sciences, Department of informatics, February 2002.

[Gund00] H. Gundersen *Design of Low-Voltage Analog Amplifiers Using Floating-Gate Transistors* thesis for the cand. scient. degree, University of Oslo, Faculty of Mathematics and Natural Sciences, Department of informatics, May 2000.

[HaLa01] P. Hasler, T. S. Lande *Overview of Floating-Gate Devices, Circuits, and Systems* IEEE Transactions on Circuits and Systems II: analog and digital signal processing, special issue on floating-gate devices, circuits, and systems, Vol. 48, Issue 1, pp. 1-3, January 2001.

[HaCh01] *A 4 digit CMOS quaternary to analog converter with current switch and neuron MOS down-literal circuit* Proceedings of the 31st IEEE International Symposium on Multiple-Valued Logic, 2001, pp. 67-71.

[Hamm00] *Computational Neurobiology Meets Semiconductor Engineering* Proceedings of the 30th IEEE International Symposium on Multiple-Valued Logic, 2000, pp. 3-12.

[HsFi02] D. Hsu, M. Figueroa, C. Diorio *Competitive Learning With Floating-Gate Circuits* IEEE Transactions on Neural Networks, Vol. 13, No. 3, pp. 732-744, May 2002.

[HaslXY] P. Hasler *Floating-Gate Devices / Circuits* Available: *http* : $//users.ece.gatech.edu/phasler/FG/Talk_slides/devices.ppt$.

[HøWi01] M. Høvin, D. Wisland, Y. Berg, T. S. Lande *A Delta-Sigma Analog-to-Digital Converter Operating at a Power-Supply Voltage of 0.6 V* Proceedings of the 19th IEEE Norchip Conference, Kista, Sweden, 12-13 November 2001, pp. 246-251.

[Impi02] *http://www.impinj.com* WWW-site, 2002.

[IsFi94] M. Ismail, Terri Fiez (editors) *Analog VLSI - signal and information processing* McGraw-Hill International series, ISBN 0-07-032386-0, 1994.

[ITRS01] International Technology Roadmap for Semiconductors *International Technology Roadmap for Semiconductors 2001 Edition - Executive Summary* 2001

[Jens02] R. Jensen *Linear Ultra Low-Voltage Floating-Gate Transconductance Amplifier Exploiting Hyperbolic Functions* Thesis for the cand. scient. degree, University of Oslo, Faculty of Mathematics and Natural Sciences, Department of informatics, February 2002.

[Koch97] C. Koch *Computation and the single neuron* Nature, pp. 207-210, 385, 1997 .

[KoGo01] V. F. Koosh, R. Goodman *Dynamic Charge Restoration of Floating-Gate Subthreshold MOS Translinear Circuits* Proceedings of the IEEE International Symposium on Circuits and Systems, Sydney, 2001, Vol. 1, pp. 33-36.

[KoSh92] K. Kotani, T. Shibata *Neuron-MOS binary-logic circuits featuring dramatic reduction in transistor count and interconnections* Technical Digest, International Electron Devices Meeting, 13-16 Dec. 1992, pp 431-434.

[LaSa94] K. R. Laker, W. M. C. Sansen *Design of analog integrated circuits and systems* McGraw-Hill International Editions, ISBN 0-07-113458-1, 1994.

[LaWi96] T. S. Lande, D. T. Wisland, T. Sæther, Y. Berg *FLOGIC - Floating-Gate Logic for Low-Power Operation* Proceedings of the 3rd IEEE International Conference on Electronics, Circuits and Systems, 1996, pp 1041-1044.

[LoHa99] A. Low, P. Hasler *Cadence-based simulation of floating-gate circuits using the EKV model* Proceedings of the 42nd Midwest Symposium on Circuits and Systems, 1999, Vol. 1, pp. 141-144.

[Loms02] Johannes Goplen Lomsdalen *Subtraction of two "sinh" shaped output currents implemented in FGUVMOS* thesis for the cand. scient. degree, University of Oslo, Faculty of Mathematics and Natural Sciences, Department of informatics, February 2002.

[Mano84] M. Morris Mano *Digital Design* Prentice/Hall International, Inc., ISBN0-13-212325-8, pp 207, 1984.

[Mead89] C. Mead *Analog VLSI and Neural Systems* Addison Wesley Publishing Company, 1989.

[Minc00] B. A. Minch *Floating-Gate Techniques for Assessing Mismatch* Proceedings of the 2000 IEEE International Symposium on Circuits and Systems, 2000 Vol. 4, pp. 385-388.

[MiKa00] W. P. Millard, Z. K. Kalayjian, A. Andreou *Calibration and matching of floating gate devices* Proceedings of the 2000 IEEE International Symposium on Circuits and Systems, 2000, Vol. 4, pp. 389-392.

[NaBe99] Y. Berg, Ø. Næss *Ultra low-Voltage Transconductance Amplifier with dynamic load* Proceedings of the 1999 IEEE Norchip Conference, Oslo, 1999, pp. 171-176.

[PuAu97] D. Purves, G. J. Augustine, D. Fitzpatrick, L. O. Katz, A.-S. LaMantia, J. O. McNamara (Editors) *Neuroscience* Sinauer, ISBN 0-97893-747-1, 1997

[Raba96] J. M. Rabaey *Digital Integrated Circuits - a design perspective* Prentice-Hall, Inc., ISBN 0-13-394271-6, 1996

[RaPe96] J. Rabaey, M. Pedram, P. Landman *Low Power Design Methodologies* in J. Rabaey, M. Pedram (editors), Low Power Design Methodologies, Kluwer Academic Publishers, 1997.

[RaFr99] A. Rantala, S. Franssila, K. Kaski, J. Lampinen, M. Åberg, P. Kuivalainen *High-precision neuron MOSFET structures* IEE Electronics Letters, Vol. 35, No. 2, pp. 155-157, Jan. 1999.

[RaFr01] A. Rantala, S. Franssila, J. Lampinen, M. Åberg, P. Kuivalainen *Improved neuron MOS-transistor structures for integrated neural network circuits* IEE Proceedings - Circuits and Systems, Vol 148, No. 1, pp. 25-34, February 2001.

[Sæth91] T. Sæther *Analog CMOS IC Design - Switched-Capacitor Circuit Cell Library With Integrated Substrate Shielding* Dissertation for the dr. ing. degree, Norwegian Institute of Technology, ISBN 82-7119-325-2, 1991.

[ScPi96] G. Schrom, C. Pichler, T. Simlinger, S. Selberherr *On the lower bounds of CMOS Supply Voltage* Solid-State Electronics, Vol. 39, No. 4, pp. 425-430, 1996.

[ShBa00] A. Shams, M. Bayoumi *A Framework for Fair Performance Evaluation of 1-bit CMOS Adder Cells* Proceedings of the 42nd Midwest Symposium on Circuits and Systems, 2000, Vol. 1, pp. 6-9.

[ShBa99] A. M. Shams, M. Bayoumi *Performance Evaluation of 1-bit CMOS Adder Cells* Proceedings of the 1999 IEEE International Symposium on Circuits and Systems, 1999, Volume 1, pp 27-30.

[Shib00] T. Shibata *Intelligent VLSI Systems Based on a Psychological Brain Model* Proceedings of the 2000 IEEE International Symposium on Intelligent Signal Processing and Systems, pp 323-332, Hawaii, U.S.A., November 5-8, 2000.

[ShYa01] T. Shibata, M. Yagi, T. Yamasaki *Bio-Inspired Hardware Computation for Robust Image Recognition* 2001 International Symposium on Nonlinear Theory and its Applications, October 28-November 1, 2000, Miyagi, Japan.

[ShOh91] T. Shibata, T. Ohmi *An Intelligent MOS Transistor Featuring Gate-Level Weighted Sum and Threshold Operations* Technical Digest, International Electron Devices Meeting, , 8-11 Dec. 1991, pp. 919-922.

[ShOh92] T. Shibata, T. Ohmi *A Functional MOS Transistor featuring Gate-Level Weighted Sum and Threshold Operations* IEEE Transactions on Electron Devices, Vol. 39, No. 6, pp. 1444-1453, June 1992.

[ShKo93] T. Shibata, K. Kotani, T. Ohmi *Real-Time Reconfigurable Logic Circuits Using Neuron MOS Transistors* Technical Digest, IEEE International Solid-State Circuits Conference, 1993, pp. 238-239, 295.

[ShOh93] T. Shibata, T. Ohmi *Neuron MOS Binary-Logic Integrated Circuits - Part II: Simplifying Techniques of Circuit Configuration and their Practical Applications* IEEE Transactions on Electron Devices, Vol. 40, No.5, 974-979, May 1993.

[ShOh95] T. Shibata, T. Ohmi *Neurotransistor: A Neuron-Like High-Functionality Transistor Implementing Intelligence on Silicon* IEEE Signal Processing Society Workshop on VLSI Signal Processing, 1995, VIII, pp. 28-37.

[SiBr90] K. Y. Siu, J. Bruck *Neural Computation of Arithmetic Functions* Proceedings of the IEEE, No. 10, pp. 1669-1675, October 1990.

[SiBr93] K. Y. Siu, J. Bruck, T. Kailath, T. Hofmeister *Depth Efficient Neural Networks for Division and Related Problems* IEEE Transactions on information theory, Vol. 39, No. 3, pp. 946-956 , May 1993.

[SoRo01] H. Soeleman, K. Roy, B. C. Paul *Robust Subthreshold Logic for Ultra-Low Power Operation* IEEE Transactions on Very Large Scale Integration Systems, Vol. 9, No. 1, pp. 90-99, February 2001.

[SoRo99] H. Soeleman, K. Roy *Ultra-Low-Power Digital Subthreshold Logic Circuits* Proceedings of IEEE International Symposium on Low-Power Electronics and Design, 1999, pp 94-96.

[Sven97] C. Svensson *Low Voltage Technologies* in W. Nebel and J. Mermet (editors), Low Power Design in Submicron Electronics, Kluwer Academic Publishers, 1997.

[SvMa00] V. Svilan, M. Matsui, J. B. Burr *Energy-Efficient 32 x 32-bit Multiplier in Tunable Near-Zero Threshold CMOS* in Proceedings of the 2000 International Symposium on Low Power Electronics and Design, 2000, pp 268-272.

[Tsiv99] Y. Tsividis *operation and Modeling of the MOS Transistor* WCB/McGraw-Hill, ISBN 0-07-065523-5, 1999.

[YtAu02] T. Ytterdal, S. Aunet *Compact Low-Voltage Self-Calibrating Digital Floating-Gate CMOS Logic Circuits* Proceedings of the 2001 IEEE International Symposium on Circuits and Systems, Arizona, 2001, Volume 5, pp 393-396.

[WoLi92] M. Wong, D. K.-Y. Liu, S. S.-W. Huang *Analysis of the Subthreshold Slope and the Linear Transconductance Techniques for the Extraction of the Capacitance Coupling Coefficients of Floating-Gate Devices* IEEE Electron Devices Letters, November 1992, Vol. 13, No. 11, pp. 566-568.

[YaAn93] K. Yang, A. G. Andreou *Subthreshold Analysis of Floating-Gate MOSFET's* Proceedings of the 10th Biennual University / Government / Industry Microelectronics Symposium, 1993, pp. 141-144.

Appendix A

Measurement setup details

Instruments from the laboratory are listed in figure A.1. They have are numbered from 1 to more than 20, and has a laboratory number in addition. For an example the instruments number 2 and 3 thus has the numbers 28w and 28y associated as well. Some instruments had the same number, but different colors on the labels denoting the actual number. "28w" means laboratory number 28 and a white label, while "28y" has got a yellow label.

The numbering 1,2,3,...,20,... is used in figure A.2, which shows which instruments were used for measuring different types of circuitry. A "1" in the corresponding column indicates that an instrument were used in the actual test setup, while a "0" indicates that it were not used. An "X" indicates that an instrument of the actual type were used, and that it might have been that one. "P1NM" indicates measurements on circuits with one capacitively coupled input to the PMOS, while there could be several to the NMOS. "PMNM" indicates measurements on circuits with an equal number of capacitively coupled input signals to PMOS and NMOS transistors.

The main difference between "PMNM" and "P1NM" measurements were that the Keithley 617 for measuring the current through the circuit was not included in the latter case. Temperature less than 5 cm from the chip while doing "PMNM" and "P1NM" measurements was usually around 25-26 degrees Celsius.

For the padring 3 different types of I/O-cells were used. "Power-pads", PPA1P and PPA2P were used once each. Standard I/O-pads with 2-300 Ohms series resistance and ESD protection were used elsewhere, except for connections to drawn poly1-poly2 capacitances which had one side directly connected to the floating-gate. As examples, x1, y1, z1 in figure A.3 can be mentioned.

name	type	number/ifi	serial number
Keithley 213	voltage source	1/27	0670480
Keithley 213	voltage source	2/28w	0695575
Keithley 213	voltage source	3/28y	0670497
Keithley 213	voltage source	4/27	0802405
Keithley 617	electrometer /source	5/14	526963
Keithley 617	electrometer /source	6/24	438101
Keithley 617	electrometer /source	7/18y	466242
Keithley 617	electrometer /source	8/21w	0550723
Keithley 236	electrometer /source	9/12y	0646130
Keithley 236	electrometer /source	10/30w	0612571
Keithley 230	progr. source	11/20y	0551294
Keithley 230	progr. source	12/19	468084
Keithley 230	progr. source	13/23	441411
HP 3245A	source	14/05"A"	2831A03131
HP 3245A	source	15/06"B"	2831A03133
E3610A	power supply	16/06"B"	KR30707474
HP 3611A	power supply	17	KR40606980
FLUKE 45	digital multimeter	18	5015102
Keithley 6514	progr. electrometer	19	0763947
Thurlby Thandar	power supply	20	-
HP33120	signal generator	21	-
UVP Mod. UVG11	4W, 254 nm lamp	22	-
test PCB	68 pin socket	23	-
test PCB	84 pin socket	24	-

Figure A.1: Instruments from the VLSI laboratory at the Department of informatics, University of Oslo.

number	transistors	inverter	"PMNM"	"P1NM"
1	X	X	0	0
2	X	X	0	0
3	0	0	1	1
4	0	0	0	0
5	0	0	0	0
6	0	0	1	0
7	0	0	1	1
8	0	0	0	0
9	X	X	0	0
10	X	X	0	0
11	0	0	0	0
12	0	0	1	1
13	0	0	1	1
14	0	0	1	1
15	0	0	0	0
16	0	0	0	0
17	0	0	0	0
18	0	1	1	1
19	0	0	0	0
20	1	1	0	0
21	1	1	0	0
22	X	X	1	1
23	1	1	0	0
24	0	0	1	1

Figure A.2: Instruments from the VLSI laboratory at the Department of informatics, University of Oslo. "1" means "used". "0" means "not used". "X" means "An instrument of this type were used and it may have been this one".

CHIP5_capall_netlist	"PMNM"	"P1NM"
padvdd	12	12
padvss	13	13
psub	27	27
vdd1	14	-
vss1	25	-
nwell1	15	-
x1	16	-
y1	17	-
z1	18	-
mid1	20	-
vdd2	-	41
nwell3	- 42	
p	-	40
vss6	-	31
x23	-	37
y23	-	38
z23	-	39

Figure A.3: Names in the netlist and corresponding pin numbers on the 84 pin chip.

Appendix B

Matlab code for UV-programming and test

```
% Department of informatics, University of Oslo
% Snorre Aunet, 010118
% ProgUVmin(0.8,1.2,40,0.38,1.20,0.8,1.6,10)

function [A,B,C,AA,BB] = ...

ProgUVmin(TVdd,Twell,TStep,PVdd,PVss,Vinstop,Pwell,PTid,k)

%Initialisering
initOK = Init;

%Tid-Vut med UV-lys
figure(101);clf;hold off;

%Tid-Vut uten UV-lys
figure(102);clf;hold off;

figure(112);clf;hold off;
figure(121);clf;hold off;

%Vut põ Vinn
%figure(103);clf;hold off;

%Iut põ Vinn
%figure(104);clf;hold off;

% Sweep under prog.
figure(105);clf;hold off;
figure(106);clf;hold off;


%Skrur põ PadVdd
PL330DP_SetVolt(1,4.5);


%Setter variable
PVin=TVdd/4;
%PVin=0.17;
TVin=TVdd/4;
TVss=0;
%Twell=TVdd;

%Lager matriser
A = [];  %Spenning ferdig
B = [];  %Str°m ferdig
C = [];  %Belysning
AA = []; % Spenning under prog.
BB = []; % Str°m under prog.


% Setter põ normale verdier og kj°rer et sweep
K213_SetVoltage(TVdd,2);
K236_SetVolt(TVss);
K236_Operate;
K213_SetVoltage(Twell,3);
HP33120_SetVolt(0);

%Lager innmatrise
inn =  0:TVdd/(2*TStep):Vinstop/2;
inn
Inn = 0:TVdd/(TStep):Vinstop;

i = length(inn);

%Mõle°kke som setter inngang og lagrer ut spenning og
str°m
for j = 1:i
  HP33120_SetVolt(inn(j));
  %  K213_SetVoltage(inn(j),3)
  pause(1);
  Vut(j) = FL45_ReadQuick(1);
  Iut(j) = K236_ReadQuick*-1;
end
AA = [AA,Vut']; % F°r programmeringen
BB = [BB,Iut']; % F°r programmeringen

  figure(105);
  plot(Inn,Vut,'r');
  grid on;
  xlabel('Vinn');
  ylabel('Vut');
  hold on;
  title 'measurements on the inverter'

  figure(162);
  plot(Inn(1:length(Inn)-1),diff(Vut)/(Inn(2)-Inn(1)));
  grid;
  xlabel('Input voltage');
  ylabel('Gain');

  %Skriver ut innspenning/utstr°m
  figure(106);
  semilogy(Inn,Iut,'r');
  grid on;
  xlabel('Vinn');
  ylabel('Iss');
  hold on;
  title 'measurements on the inverter during UV-programming'

%Setter põ programmeringsspenninger
K213_SetVoltage(PVdd,2);
K236_SetVolt(PVss);
K236_Operate;
K213_SetVoltage(Pwell,3);
HP33120_SetVolt(PVin);


%Setter hvor mange utskrifter pr. programmeringsminutt
%k = PTid/1


%Matriser til plotting under programmering
PlotTid(1) = 0;
VutPUV(1) = FL45_ReadQuick(1);
VutP(1) = FL45_ReadQuick(1);
```

Figure B.1: "ProgUVmin" matlab code, part one.

```
tiiden = PTid/k;


for l = 1:k
%Setter pÕ programmeringsspenninger
K213_SetVoltage(PVdd,2);
K236_SetVolt(PVss);
K236_Operate;
K213_SetVoltage(Pwell,3);
HP33120_SetVolt(PVin);

  %Programmerer
  [PSannTid,IVssUV,IVssEtter,VoutUV,VoutEtter] = NUV_prog_inv(tiiden);

  PlotTid(l+1) = PSannTid/60 + PlotTid(l);
  VutPUV(l+1) = VoutUV;
  VutP(l+1) = VoutEtter;

  %Plotter spenninger
  figure(101);
  plot(PlotTid,VutPUV,'b',PlotTid,VutP,'--');%,PlotTid,VutPUV,'+'
  grid on;
  xlabel('Time');
  ylabel('Output voltages during programming and without UV-
exposure');
  hold on;

  figure(102);
  plot(PlotTid,VutP,'b');%,PlotTid,VutP,'+'
  grid on;
  xlabel('time');
  ylabel('Output voltage during UV-programming.');
  hold on

figure(112);
plot(PlotTid,VutP-VutPUV);
grid on;
xlabel('Time');
ylabel('Output under regular use minus output during
programming');
hold on;

  if l > 2,
    figure(121);
    plot(PlotTid(3:l),diff(VutPUV(2:l)),PlotTid(3:l),diff(VutP(2:l)),'--');
    grid on;
    xlabel('Time');
    ylabel('The derivative of the output voltage with and without UV-
light');
    hold on;
  end



  % Setter pÕ normale verdier og kj°rer et sweep under prog
  K213_SetVoltage(TVdd,2);
  K236_SetVolt(TVss);
  K236_Operate;
  K213_SetVoltage(Twell,3);
  HP33120_SetVolt(0);

  % Lager innmatrise
  inn =  0:TVdd/(2*TStep):Vinstop/2;
  inn

  i = length(inn);

  % MÕlel°kke som setter inngang og lagrer ut spenning og str°m
  for j = 1:i
    HP33120_SetVolt(inn(j));
    pause(1);
    Vut(j) = FL45_ReadQuick(1);
    Iut(j) = K236_ReadQuick*-1;
  end
  AA = [AA,Vut']; % under  programmeringen
  BB = [BB,Iut']; % under programmeringen

  %Skriver ut innspenning/utspenning under prog.
  figure(105);
  plot(Inn,Vut,'b');
  %Skriver ut innspenning/utstr°munder prog.
  figure(106);
  semilogy(Inn,Iut,'b');
save (['invrekke_208120_10118x' int2str(l)])

end

%-----------------------------------------
 C = [C,VutPUV']; %Lagrer verdiene en matrise
%-----------------------------------------

%Tester den programmerte kretsen

%Setter pÕ vanlige spenninger
K213_SetVoltage(TVdd,2);
K236_SetVolt(TVss);
K213_SetVoltage(Twell,3);
HP33120_SetVolt(PVin);

pause(1)


% Sjekker midtpunkt
Oppwell=Twell;
Nedwell=Twell;

HP33120_SetVolt(TVdd/2);
mid=FL45_ReadQuick(1);
% 001110: FJERNES FOR ◂ SPARE TID N◂ ..
%while abs(mid-(TVdd/2)) > (0.01) & Oppwell < 5 & Nedwell > 0
%pause(1);
%  if mid > PVin/2
%    Oppwell=Oppwell+0.01;
%    K213_SetVoltage(Oppwell,3); %opp med br°nn gir mindre pÕ
utg
%    mid=FL45_ReadQuick(1);
%    well=Oppwell;
%  else
```

Figure B.2: "ProgUVmin" matlab code, part two.

```
%    Nedwell=Nedwell-0.01;
%    K213_SetVoltage(Nedwell,3); %ned med br°nn gir °kning põ
utg
%    mid=FL45_ReadQuick(1);
%    well=Nedwell;
%  end
%end
K213_GetVoltage(3)

%Sweeper inngangen

%Lager innmatrise
inn =  0:TVdd/TStep:TVdd;
inn

i = length(inn);

%Mõle¡°kke som setter inngang og lagrer ut spenning og str°m
for j = 1:i
  HP33120_SetVolt(inn(j));
  %  K213_SetVoltage(inn(j),3)
  pause(1);
  Vut(j) = FL45_ReadQuick(1);
  Iut(j) = K236_ReadQuick*-1;
end

%-----------------------------------------
 A = [A,Vut']; %Lagrer verdiene en matrise
 B = [B,Iut']; %Lagrer verdiene en matrise
%-----------------------------------------

%Skriver ut inn/ut-spenning
%figure(103);
%  plot(Inn,Vut,'b');
%  grid on;
%  xlabel('Vinn');
%  ylabel('Vut');
%  hold on;
%  title 'Mõleresultater fra inverteren'
%  %Skriver ut innspenning/utstr°m
%  figure(104);
%  semilogy(Inn,Iut,'b');
%  grid on;
%  xlabel('Vinn');
%  ylabel('Iss');
%  hold on;
%  title 'Mõleresultater fra inverteren'

%end
%  save (['inverterprog001010_vdd' int2str(p)]), A, B, C, inn, Ibec,
Vss, Vdd;
% save maaleres/inverterprog220900x1 A B C inn Ibec Vss PVdd;
%save inverterprog010118 A B C inn AA BB well;
save inverterprog010119 A B C inn AA BB Tvdd PVdd PVss Pwell PTid
k;
%figure(1);print -depsc2 figurer/vutmedlys010118invmaxuv.eps

Ok=1;
```

Figure B.3: "ProgUVmin" matlab code, part three.

```
% Department of informatics, University of Oslo.
% 010118, Snorre Aunet
% This code is used for doing measurements after UV-programming.
function [AA,BB,Inn] = InvSweep(Vdd,Vwell,Vinstart,Vinstop,Vinstep)
%OBS: Agilent 33120A gives twice the DC shown on display.
%InvSweep(0.8,0.8,0.0,2.0,20)
%[VOUT,ISS,INN]=InvSweep(0.11,1.1,0.0,0.11,20);

figure(3);clf;hold off;
figure(4);clf;hold off;

K213_SetVoltage(Vdd,2);
K236_SetVolt(0.0);
K236_Operate;
K213_SetVoltage(Vwell,3);
HP33120_SetVolt(0);

AA = []; % Spenning under prog.
BB = []; % Str°m under prog.

Vinstart/2
(Vinstop - Vinstart)/2*Vinstep
Vinstop/2

%Lager innmatrise
inn =  Vinstart/2:(Vinstop - Vinstart)/(2*Vinstep):Vinstop/2
inn

Inn = Vinstart:(Vinstop - Vinstart)/(Vinstep):Vinstop;

i = length(inn)

%MÕlel°kke som setter inngang og lagrer ut spenning og str°m
for j = 1:i
  HP33120_SetVolt(inn(j));
  %  K213_SetVoltage(inn(j),3)
  pause(1);
  Vut(j) = FL45_ReadQuick(1);
  Iut(j) = K236_ReadQuick*-1;
end
AA = [AA,Vut']; % F°r programmeringen
BB = [BB,Iut']; % F°r programmeringen

  figure(1);
  plot(Inn,Vut,'r');
  grid on;
  xlabel('Vinn');
  ylabel('Vut');
  hold on;
  title 'Output Voltage'
  %Skriver ut innspenning/utstr°m
  figure(2);
  semilogy(Inn,Iut,'r');
  grid on;
  xlabel('Vinn');
  ylabel('Iss');
  hold on;
  title 'Current '


save inv010119_0_095
```

Figure B.4: ”InvSweep” matlab code

```
% Department of informatics, University of Oslo,
% Snorre Aunet, 010118.

function[ok] = Init()

%FLUKE 45 is initialized and set to measure voltage:
FL45_Init;
FL45_SetMode(1,'VD');


% "617" is initialized and thereafter set to measure voltage:
%K617_Init;
%K617_SetMode('V');

% After initialization, the Keithley 236 is used as a combined
% voltage source and amperemeter.
K236_Init;
K236_SetSource('V');
K236_SetILimit(0.1);

% Thurlby Thandar PL330DP  is used for the padframe and the
% UV-lamp. (UVP Model DE-4 Eprom eraser, 254 nm UV.)
% Padframe (1), UV-Light (2)
PL330DP_Init;
PL330DP_SetILimit(1,0.5);
PL330DP_SetILimit(2,2.0);
PL330DP_SetVolt(1,4.5);
%PL330DP_SetVolt(2,5);
PL330DP_SetVolt(2,0);

% Vdd (channel 2), WELL (channel 3)
% NB!: "(voltage, channel)"
K213_Init;
K213_SetVoltage(0,2);
K213_SetVoltage(0,3);

% INPUT
HP33120_Init;
HP33120_Init('/dev/HP33120A');
HP33120_SetVolt(0);
HP33120_SetVolt(0,'/dev/HP33120B')


K213_SetVoltage(0,2);
K213_SetVoltage(0,3);
K236_SetVolt(0);
K236_Disable;
K236_Operate;
K213_SetVoltage(0,1);
HP33120_SetVolt(0);
HP33120_SetVolt(0,'/dev/HP33120A');

ok = 1;
```

```
physcmprc

unix('setenv LD_LIBRARY_PATH /NICgpib/lib:$LD_LIBRARY_PATH');

path(path,['/local/vlsi/lib/matlab/physcmp/m-util/misc:', ...
'/local/vlsi/lib/matlab/physcmp/mex-util:', ...
'/local/vlsi/lib/matlab/physcmp/m-util/MPlot182:', ...
'/local/vlsi/lib/matlab/physcmp/gpib/GPIBM-files/EGG5302:', ...
'/local/vlsi/lib/matlab/physcmp/gpib/GPIBM-files/MFit182:', ...
'/local/vlsi/lib/matlab/physcmp/gpib/GPIBM-
files/MFreqResp182:', ...
'/local/vlsi/lib/matlab/physcmp/gpib/GPIBM-files/MHP3582A:', ...
'/local/vlsi/lib/matlab/physcmp/gpib/GPIBM-files/MHP601A:', ...
'/local/vlsi/lib/matlab/physcmp/gpib/GPIBM-files/MK230:', ...
'/local/vlsi/lib/matlab/physcmp/gpib/GPIBM-files/MK617:', ...
'/local/vlsi/lib/matlab/physcmp/gpib/GPIBM-files/MK236:', ...
'/local/vlsi/lib/matlab/physcmp/gpib/GPIBM-files/MScope:', ...
'/local/vlsi/lib/matlab/physcmp/gpib/GPIBM-files/MScope182:', ...
'/local/vlsi/lib/matlab/physcmp/gpib/GPIBM-files/MTake182:', ...
'/local/vlsi/lib/matlab/physcmp/gpib/GPIBM-files/MTEK2440:', ...
'/local/vlsi/lib/matlab/physcmp/gpib/GPIBM-files/MPL330DP:', ...
'/local/vlsi/lib/matlab/physcmp/gpib/GPIBM-files/MFL45:', ...
'/local/vlsi/lib/matlab/physcmp/gpib/GPIBM-files/MHP53131A:', ...
'/local/vlsi/lib/matlab/physcmp/gpib/GPIBM-files/MHP34401A:', ...
'/local/vlsi/lib/matlab/physcmp/gpib/GPIBM-files/MHP3245A:', ...
'/local/vlsi/lib/matlab/physcmp/gpib/GPIBM-files/MHP8116A:', ...
'/local/vlsi/lib/matlab/physcmp/gpib/GPIBM-files/MK6512:', ...
'/local/vlsi/lib/matlab/physcmp/gpib/GPIBM-files/MIN347:', ...
'/local/vlsi/lib/matlab/physcmp/gpib/GPIBM-files/MKLIMA:', ...
'/local/vlsi/lib/matlab/physcmp/gpib/GPIBM-files/MHPL1500:' ...
'/local/vlsi/lib/matlab/physcmp/gpib/GPIBM-files/MLC9310:' ...
'/local/vlsi/lib/matlab/physcmp/gpib/GPIBM-files/MHP33120A:' ...
'/local/vlsi/lib/matlab/physcmp/gpib/GPIBM-files/MHP34970:' ...
'/local/vlsi/lib/matlab/physcmp/gpib/GPIBM-files/MK213:']);

v=version;
if str2num(v(1))==5
  path(path,'/local/vlsi/lib/matlab/physcmp/gpib/GPIBM-
files/MGPIB5:')
else
  path(path,'/local/vlsi/lib/matlab/physcmp/gpib/GPIBM-
files/MGPIB:')
end

path(path,'/local/vlsi/lib/matlab/physcmp/gpib/GPIBMEX/sol');

clear v w s
```

Figure B.5: "Init" and "startup" matlab code, to the left and right, respectively.

```
% Department of informatics, University of Oslo.
% Snorre Aunet, 010118

function [time,Id1,Idoff,Vd,Voff] = NUV_prog_inv(seconds)

% Skru på lys
PL330DP_SetVolt(2,5);

tic;

pause(seconds*60);

Id1=K236_ReadQuick*-1;
Vd=FL45_ReadQuick(1);

time = toc;

%Skru av lys
PL330DP_SetVolt(2,0);

pause(2);

Idoff=K236_ReadQuick*-1;
Voff=FL45_ReadQuick(1);
```

```
function [Ok]= skruav();

PL330DP_SetVolt(2,0);
tic
toc
Ok=1;
```

Figure B.6: "NUV_prog_inv" and "$skruav$" matlab code.

```
function[Vin,Vss,Vout,Inout,tid,mpnkt,Vdd,Well,Vp,Vm,Vb,Vutlys,Vutul
ys,Temperatur] = Prog(tid,mpnkt,Vdd,Well,Vp,Vm,ProgWell,Y,Z,W)

avlys; Setnull;

p    = 32;
%p    = 32;
Vss  = 0;
%Vdd = 0;
figure(1); hold off; if ~(nargin==8), clf; end; hold on; grid on;
figure(2); hold off; if ~(nargin==8), clf; end; hold on; grid on;
figure(3); hold off; grid off; if ~(nargin==8), clf; end;


Vutlys  = [];
Vutulys = [];
tid2    = [];
Vin     = []; Vin = [0:Vdd/p:Vdd];
Vout    = [];
Inout   = [];

K230_SetVolt(Vdd,'/dev/K230A');
K213_SetVoltage(Well,4,'/dev/K213A');
K230_SetVolt(Vss,'/dev/K230B');

pause(3);

Vt  = [];
%It  = [];

y = Vdd/2;
z = Vdd/2;
w = W;
%w = Vdd/2;
if Y == 1, y = Vdd; end
if Y == 0, y = 0; end
if Z == 1, z = Vdd; end
if Z == 0, z = 0; end
if W == 1, w = Vdd; end
if W == 0, w = 0; end

%Prog(10,5,0.8,0.8,2.0,0.0,0.4,-1,-1,-1)
% sweepe inngangen, mõle str°m i nmos og spenning ut
for j=1:p+1
  K213_SetVoltage(Vin(j),2,'/dev/K213A');
  if Y == -1, HP3245_SetVolt(Vin(j),'/dev/HP3245'); else
HP3245_SetVolt(y,'/dev/HP3245'); end
  if Z == -1, K213_SetVoltage(Vin(j),3,'/dev/K213A'); else
K213_SetVoltage(z,3,'/dev/K213A'); end
  if W == -1, K213_SetVoltage(Vin(j),1,'/dev/K213A'); else
K213_SetVoltage(w,1,'/dev/K213A'); end
  pause(2);
  Vt  = [Vt,K617_ReadQuick('/dev/K617A')];
%  It  = [It,K617_ReadQuick('/dev/K617B')];
end;

  figure(2); plot(Vin,Vt,'r');
%  figure(3); semilogy(Vin,It,'r'); grid on; hold on;
  figure(1);

  Vout  = [Vout,Vt'];
%  Inout  = [Inout,It'];

% loop , se hvordan karakteristikkene endrer seg ved belysning
for i=1:mpnkt
  % Setter programmeringsspenninger

  K213_SetVoltage(Vdd/2,1,'/dev/K213A');
  K213_SetVoltage(Vdd/2,2,'/dev/K213A')
  K213_SetVoltage(Vdd/2,3,'/dev/K213A')
  HP3245_SetVolt(Vdd/2,'/dev/HP3245');
  K230_SetVolt(Vp,'/dev/K230B');
  K213_SetVoltage(ProgWell,4,'/dev/K213A');
  K230_SetVolt(Vm,'/dev/K230A');

  pause(1);

%  [a,b]  = Belysning(tid*60/mpnkt);  % belyser i 'tid' antall minutter
  [a,b]  = Belysning(60);  % belyser i 'tid' antall minutter
  Vutlys  = [Vutlys,a];
  Vutulys = [Vutulys,b];

  tid2 = [tid2, (tid/mpnkt)*i]
  length(tid2)
  length(Vutlys)
  length(Vutulys)
  figure(1); ylabel('Vut spenning');plot(tid2,Vutlys,tid2,Vutulys);
  legend('med UVlys','uten UVlys');


  K230_SetVolt(Vdd,'/dev/K230A');
  K213_SetVoltage(Well,4,'/dev/K213A');
%  K213_SetVoltage(InnA,2,'/dev/K213A');
%  K213_SetVoltage(InnB,3,'/dev/K213A');
  K230_SetVolt(Vss,'/dev/K230B');

  Vt = [];
%  It = [];

  % sweepe inngangen, mõle str°m i nmos og spenning ut.
  for j=1:p+1
  K213_SetVoltage(Vin(j),2,'/dev/K213A');
    y
    z
    w
    Y
    Z
    W
    if Y == -1, HP3245_SetVolt(Vin(j),'/dev/HP3245'); else
HP3245_SetVolt(y,'/dev/HP3245'); end
    if Z == -1, K213_SetVoltage(Vin(j),3,'/dev/K213A'); else
K213_SetVoltage(z,3,'/dev/K213A'); end
    if W == -1, K213_SetVoltage(Vin(j),1,'/dev/K213A'); else
K213_SetVoltage(w,1,'/dev/K213A'); end

    pause(2);
    Vt  = [Vt,K617_ReadQuick('/dev/K617A')];
%    It  = [It,K617_ReadQuick('/dev/K617B')];
  end;

  figure(2); ylabel('Spenning');plot(Vin,Vt);
%  figure(3); semilogy(Vin,It);ylabel('Str°m');
  figure(1);

  Vout  = [Vout,Vt'];
%  Inout = [Inout,It'];
end;


save Progtemp_finmaaling.mat;
! mv Progtemp_finmaaling.mat Prog_finmaaling_`date '+%y%m%d%H%M%S'`.mat ;
! cp Prog.m Prog_finmaaling_`date '+%y%m%d%H%M%S'`.m ;
ls
```

Figure B.7: "Prog.m" for UV-programming and measurements on the 84 pin chip.

Appendix C

Pin lists

pin / signal / pad	connected exclusively to module	comment
1 / ASVBP1 / 2	*asinh*_20812	
2 / ASIN / 5	*asinh*_20812	
3 / ASAVSS1 / 2	*asinh*_20812	
4 / ASVBN1 / strip	*asinh*_20812	
5 / ASVBP2 / 2	*asinh*_20812	
6 / ASNWELL2 / strip	*asinh*_20812	
7 / NGATES / 2	nmosrekke	biasing voltage for f.-gates
8 / NSOURCES / strip	nmosreke	common source
9 / NDRAIN1 / strip	nmosrekke	drain node
10 / NDRAIN2 / strip	nmosrekke	drain node
11 / NDRAIN3 / strip	nmosrekke	drain node
12 / NDRAIN4 / strip	nmosrekke	drain node
13 / NDRAIN5 / strip	nmosrekke	drain node
14 / NDRAIN6 / strip	nmosrekke	drain node
15 / NDRAIN7 / strip	nmosrekke	drain node
16 / NDRAIN8 / strip	nmosrekke	drain node
17 / NDRAIN9 / strip	nmosrekke	drain node

Figure C.1: Pin numbering, signal names, pad types and circuitry on the 68 pin chip, pins 1-17.

pin / signal / pad	connected exclusively to module	comment
18 / PGATES / 2	pmosrekke	biasing voltage for f.-gates
19 / PDRAIN1 / strip	pmosrekke	drain node
20 / PDRAIN2 / strip	pmosrekke	drain node
21 / PDRAIN3 / strip	pmosrekke	drain node
22 / PDRAIN4 / strip	pmosrekke	drain node
23 / PDRAIN5 / strip	pmosrekke	drain node
24 / PDRAIN6 / strip	pmosrekke	drain node
25 / PDRAIN7 / strip	pmosrekke	drain node
26 / PDRAIN8 / strip	pmosrekke	drain node
27 / PDRAIN9 / strip	pmosrekke	drain node
28 / ASOUT / 5	*asinh*_20812	
29 / ASVSS2 / strip	*asinh*_20812	
30 / ASVDD2 / strip	*asinh*_20812	
31 / IOUT6 / 2	*invrekke*_20812	output, inverter 6
32 / IOUT5 / 2	*invrekke*_20812	output, inverter 5
33 / IOUT4 / 2	*invrekke*_20812	output, inverter 4
34 / IOUT3 / 2	*invrekke*_20812	output, inverter 3

Figure C.2: Pin numbering, signal names, pad types and circuitry on the 68 pin chip, pins 18-34.

pin signal	connected exclusively to module	comment
35 / PPA2P / POWER		AVDD power pad
36 / PPA1P / POWER		AVSS power pad
37 / IOUT2 / 2	*invrekke*_20812	output, inverter 2
38 / A2OUT1 / 2	*ainv*2*brekke*_20812	
39 / A2OUT / 5	*ainv*2*brekke*_20812	
40 / A2AVSS4 / strip	*ainv*2*brekke*_20812	
41 / A2AVSS3 / strip	*ainv*2*brekke*_20812	
42 / A2AVSS2 / strip	*ainv*2*brekke*_20812	
43 / A2AVSS1 / strip	*ainv*2*brekke*_20812	
44 / A1AVSS4 / strip	*ainv*1*rekke*_20812	
45 / A1AVSS3 / strip	*ainv*1*rekke*_20812	
46 / A1AVSS2 / strip	*ainv*1*rekke*_20812	
47 / A1AVSS1 / strip	*ainv*1*rekke*_20812	
48 / A1OUT / 5	*ainv*1*rekke*_20812	
49 / IOUT / 5	*inv*1*rekke*_20812	output, inverter 10
50 / I1AVSS4 / strip	*inv*1*rekke*_20812	VSS, inverter 1
51 / I1AVSS3 / strip	*inv*1*rekke*_20812	VSS, inverters 2,4,5,6,7,8,9

Figure C.3: Pin numbering, signal names, pad types and circuitry on the 68 pin chip, pins 35-51.

pin signal	connected exclusively to module	comment
52 / I1AVSS2 / strip	*inv*1*rekke*_20812	VSS, inverter 3
53 / I1AVSS1 / strip	*inv*1*rekke*_20812	VSS, inverter 10
54 / IOUT1 / 2	*invrekke*_20812	output, inverter 1
55 / A1OUT1 / 2	*ainv*1*rekke*_20812	
56 / IIN / 5	*invrekke*_20812	input, inverter 1
57 / PPA1P / strip		connected to substrate
58 / NWELL / strip		conn. to shared nwell
59 / DVDD / strip	*invrekke*_20812	VDD for inverters
60 / AVDD / strip		VDD for pmosrekke and other blocks
61 / ROUT / 5	refer20812	
62 / RVSS / strip	refer20812	
63 / A1IN / 5	*ainv*1*rekke*_20812	
64 / A2VBN / 2	*ainv*2*brekke*_20812	
65 / A2IN / 5	*ainv*2*brekke*_20812	
66 / A2VB / 2	*ainv*2*brekke*_20812	
67 / ASPSUB2 / strip	*asinh*_20812	
68 / ASVBN2 / 2	*asinh*_20812	

Figure C.4: Pin numbering, signal names, pad types and circuitry on the 68 pin chip, pins 52-68.

pin / signal / pad	connected exclusively to module(s)	comment
1 / pd3 / strip	*PMOS_ROW*	
2 / pd4 / strip	*PMOS_ROW*	
3 / pd5 / strip	*PMOS_ROW*	
4 / ngs / 2	*NMOS_ROW*	
5 / ns / strip	*NMOS_ROW*	
6 / nd1 / strip	*NMOS_ROW*	
7 / nd2 / strip	*NMOS_ROW*	
8 / nd3 / strip	*NMOS_ROW*	
9 / nd4 / strip	*NMOS_ROW*	
10 / nd5 / strip	*NMOS_ROW*	
11 / vdd5 / strip	TY	
12 / padvdd / PPA2P		
13 / padvss / PPA1P		
14 / vdd1 / strip	FAEXP20812III	vss, 1st P5N5 element
15 / nwell1 / strip	FAEXP20812III	well 1
16 / x1c / 2	FAEXP20812III	input X
17 / y1c / 2	FAEXP20812III	input Y

Figure C.5: Pin numbering, signal names, pad types and circuitry on the 84 pin chip, pins 1-17.

pin / signal / pad	connected exclusively to module(s)	comment
18 / z1c / 2	FAEXP20812III	input Z
19 / c1nc / 2	FAEXP20812III	output, 1st P5N5 element
20 / mid1c / 2	FAEXP20812III	input W, 1st P5N5 element
21 / s1nc / 2	FAEXP20812III	output, 2nd P5N5 element
22 / c1c / 2	FAEXP20812III	output, 3rdd P5N5 element
23 / mid2c / 2	FAEXP20812III	input W, 3rd and 4th P5N5 element
24 / s1 / 5	FAEXP20812III	output, 4th P5N5 element
25 / vss1 / strip	FAEXP20812III	vss, 1st P5N5 element
26 / vss2 / strip	FAEXP20812III	vss, 2nd P5N5 element
27 / vss3 / strip	FAEXP20812III	to substrate
28 / psub1 / strip	FAEXP20812III	vss, 3rd and 4th P5N5 elem.
29 / vss4 / strip	FAEXP20812II	
30 / vss5 / strip	FAEXP20812II	
31 / vss6 / strip	FAEXP20812	vss, 1st P1N3 element
32 / vss7 / strip	FAEXP20812	vss, 2nd P1N3 element
33 / c2n / 5	FAEXP20812II	
34 / s2n / 5	FAEXP20812II	
35 / c3n / 5	FAEXP20812	output, 1st P1N3 element

Figure C.6: Pin numbering, signal names, pad types and circuitry on the 84 pin chip, pins 18-35.

pin / signal / pad	connected exclusively to module(s)	comment
36 / s3n / 5	FAEXP20812	output, P1N5 element
37 / x23c / 2	FAEXP20812II/ FAEXP20812	input X,
38 / y23c / 2	FAEXP20812II / FAEXP20812	input Y,
39 / z23c / 2	FAEXP20812II / FAEXP20812	input Z,
40 / pc /	FAEXP20812	input PC ("control")
41 / vdd2 / strip	FAEXP20812II / FAEXP20812	vdd
42 / nwell3 / strip	FAEXP20812II / FAEXP20812	well 3 P3N3, P1N3
43 / nwell4 / strip	FAEXP20812II / FAEXP20812	P5N5, P1N5
44 / yai / 2	YBA	
45 / vdd4 / strip	YBA,YBD,TY	
46 / vss10 / strip	YBA	
47 / nwell7 / strip	YBA	
48 / yao / 2	YBA,YBD,TY	
49 / vss11 / strip	YBD	
50 / nwell8 / strip	YBD	
51 / ydoc / 2	YBD	
52 / vss12 / strip	TY	
53 / tync / 2	TY	

Figure C.7: Pin numbering, signal names, pad types and circuitry on the 84 pin chip, pins 36-53.

pin / signal / pad	connected to module	comment
54 / typc / 2	TY	
55 / nwell9 / strip	TY	
56 / aivdd / strip	ainv20812	
57 / nwell2 / strip	ainv20812	
58 / aiinc / 2	ainv20812	
59 / aivss1 / strip	ainv20812	
60 / a1outc / 2	ainv20812	
61 / a2outc / 2	ainv20812	
62 / a3outc / 2	ainv20812	
63 / aivss2 / strip	ainv20812	
64 / ivdd / strip	*inv*2_20812	
65 / iinc / 2	*inv*2_20812	
66 / ivss1 / strip	*inv*2_20812	
67 / ivss2 / strip	*inv*2_20812	
68 / iout1c / 2	*inv*2_20812	
69 / iout2c / 2	*inv*2_20812	
70 / iout3 / 5	*inv*2_20812	

Figure C.8: Pin numbering, signal names, pad types and circuitry on the 84 pin chip, pins 54-70.

pin / signal / pad	connected to module	comment
71 / vdd3 / strip	FAEXP00806IV	
72 / nwell5 / strip	FAEXPOO806IV	
73 / nwell6 / strip	FAEXP00806IV	
74 / vss8 / strip	FAEXP00806IV	
75 / vss9 / strip	FAEXP00806IV	
76 / s4nc / 2	FAEXP00806IV	
77 / c4nc / 2	FAEXP00806IV	
78 / mid0c / 2	FAEXP00806IV	
79 / x4c / 2	FAEXP00806IV	
80 / y4c / 2	FAEXP00806IV	
81 / z4c / 2	FAEXP00806IV	
82 / pgsc / 2	*PMOS_ROW*	
83 / pd1 / strip	*PMOS_ROW*	
84 / pd2 / strip	*PMOS_ROW*	

Figure C.9: Pin numbering, signal names, pad types and circuitry on the 84 pin chip, pins 71-84.

Appendix D

Publications

1) "Floating-gate low-voltage/low-power linear threshold element for neural computation ":

FLOATING-GATE LOW-VOLTAGE/LOW-POWER LINEAR THRESHOLD ELEMENT FOR NEURAL COMPUTATION

Snorre Aunet, Yngvar Berg *, Trond Sæther

Department of physical electronics, NTNU, O. S. Bragstads plass 2A, N-7034 Trondheim, Norway.
* Department of informatics, University of Oslo, Gaustadaleen 23, N-0316, Norway.

ABSTRACT

We present analysis, simulation and measurements for a new reconfigurable 2-MOSFET linear threshold element and show how it can be used to implement INVERT, NAND3, NAND2, NOR3, NOR2 and CARRY'. The threshold, and thereby the Boolean function, might be changed in real time by changing the voltage on one or more of the inputs. Simulation suggests that it can function as a generator of CARRY' for binary addition with a power supply voltage down to about 200 mV.

1. INTRODUCTION

The human brain can be a factor of 10 million more effective than the best digital technology that we can imagine [1]. Human brains are by far superior to computers in solving problems like for an example combinatorial optimization and image and speech recognition, although their basic building blocks are several orders of magnitude slower [2]. Such observations have boosted interest in the field of artificial neural networks, which are built by interconnecting artificial neurons inspired by biology [2]. A linear threshold device computes a linear combination of the inputs, compares the value with a threshold, and outputs -1 or +1 if the value is larger or smaller than the treshold. It is the basic building block in neural networks [3]. Neural networks can be much more powerful than traditional logic circuits, assuming that each threshold element can be built at a cost comparable to that of AND, OR, NOT elements [3]. A network of neurons can implement any Boolean function [2]. We demonstrate elements computing symmetric Boolean functions, for which the output depends on the number of 1s in the input vector, irrespective of their position [2]. We have designed and implemented some new threshold elements in silicon, using floating-gate circuits and the "FGUVMOS" programming approach, described in [4]. First we present a simple analysis, then Spice-based simulations and thereafter measurement results from circuits implemented in a standard CMOS technology [5].

2. FGUVMOS CIRCUITS

Manipulating the charge on floating-gates can be done using Fowler-Nordheim tunneling and hot-electron injection, or by using UV-light [6]. We use UV-light both to add and remove charge on the floating-gates (FGUVMOS circuits). The technique can be used for making different effective threshold voltages for transistors seen from a driving node. FGUVMOS circuits may be used to implement analog and digital low-power/low-voltage signal processing circuits typically operating with a V_{dd} in the 300 - 800 mV area, with both NMOS and PMOS transistors constantly working in subthreshold. The inverter function has been demonstrated by measurements for a V_{dd} range of 93 to 800 mV in [12].

3. SIMPLE ANALYSIS OF OUR CIRCUIT

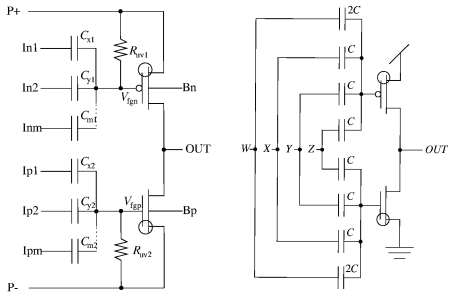

Fig. 1. The effective threshold voltages is determined in a programming procedure involving UV-light activated conductances, and a special biasing scheme. A simple illustration of this scheme, for a multi-input FGUVMOS circuit is shown to the left. The linear threshold element is shown rightmost, with additional circles in the MOSFET symbols indicating FGUVMOS transistors.

We can use the following equations to describe the subthreshold behaviour of FGUVMOS transistors [4][7]:

$$I_{ds,p} = I_{beq} \prod_{i=1}^{m} exp\{\frac{1}{nU_t}(V_{dd}/2 - V_i)k_i\} \quad (1)$$

$$I_{ds,n} = I_{beq} \prod_{i=1}^{m} exp\{\frac{1}{nU_t}(V_i - V_{dd}/2)k_i\} \quad (2)$$

W	number of "1"'s	e_p	e_n	OUT
V_{dd}	0	$V_{dd}/10$	-$V_{dd}/10$	1
V_{dd}	1	-$V_{dd}/10$	$V_{dd}/10$	0
V_{dd}	2	-$3V_{dd}/10$	$3V_{dd}/10$	0
V_{dd}	3	-$5V_{dd}/10$	$5V_{dd}/10$	0
$V_{dd}/2$	0	$3V_{dd}/10$	-$3V_{dd}/10$	1
$V_{dd}/2$	1	$V_{dd}/10$	-$V_{dd}/10$	1
$V_{dd}/2$	2	-$V_{dd}/10$	$V_{dd}/10$	0
$V_{dd}/2$	3	-$3V_{dd}/10$	$3V_{dd}/10$	0
0	0	$5V_{dd}/10$	-$5V_{dd}/10$	1
0	1	$3V_{dd}/10$	-$3V_{dd}/10$	1
0	2	$V_{dd}/10$	-$V_{dd}/10$	1
0	3	-$V_{dd}/10$	$V_{dd}/10$	0

Fig. 2. The table shows part of the exponentials, e_p, e_n, and output values, for different values of W, and different numbers of "1's" on ordinary inputs X,Y,Z. Lines 2-5 show the 3-input NOR function, lines 6-9 the CARRY'-, and lines 10-13 the NAND-function.

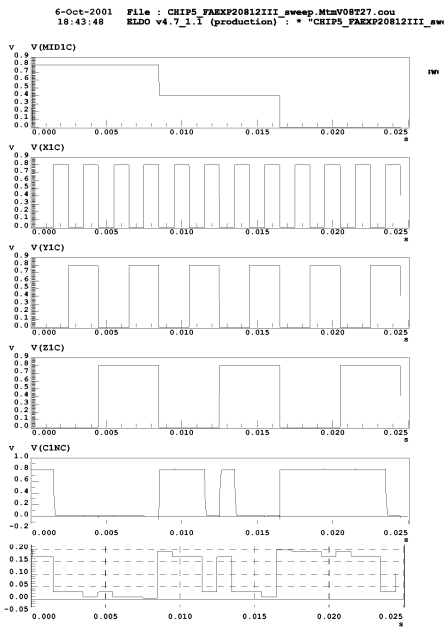

Fig. 3. The six signals, from top to bottom, correspond to "W","X","Y","Z","OUT" and "OUT" in figure 1. The fifth and sixth signals are the output for a V_{dd} of 0.8 V and 0.2 V respectively.

Here, $k_i = C_i/C_{tot}$ is the capacitive division factor of the ith input capacitor, C_i, and C_{tot} is the total capacitance seen from the floating-gate. I_{beq} is the balanced equilibrium current, which is the drain current of the transistors when all input signals and driven nodes have a voltage level equalling $V_{dd}/2$. I_{beq} is a function of the effective threshold voltages seen from the driving nodes, and measurements indicate typically current levels in the nA to uA range. For a certain I_{ds} level for PMOS and NMOS there is one set of effective threshold voltages. An FGUVMOS circuit can have different I_{ds} levels for different transistors, due to different layouts, transistor sizes and topologies. Our circuit is shown to the right in figure 1. We use the W-input as an input for choosing which Boolean function for the circuit to compute, and X, Y, Z as regular input signals. If we use the equations above, and define $e_p = (V_{dd}/2 - V_i)$ and $e_n = (V_i - V_{dd}/2)$, and let regular inputs be at V_{ss} or V_{dd} ("0" or "1", respectively), we get the table in figure 2. Since the circuit is initially in the equilibrium condition, due to the UV-programming initialization procedure, the output will be "1" if $e_p > e_n$, and "0" in the opposite case. When W=V_{dd}, the circuit implements the NOR3 function. When W=0, the circuit implements the NAND3 function. For W=Vdd/2 the CARRY' for binary addition is computed. These functions are symmetric Boolean functions, for which the output depends on the number of 1s in the input vector [2].

4. SIMULATIONS

We operate our circuits outside the classical above-treshold regime. This is not straightforward due to the lack of accurate models [8]. Simulation of floating-gate circuits generally is not straightforward either. Convergence problems may often result [9],[10], and device models may have poor predictability of output resistance of MOS devices [8], as well as other parameters [10]. Our way of simulating floating-gate circuits operating in subthreshold is described in [11], and is also used here, based on a netlist extracted from the layout shown in figure 4. For about the first third of the transient analysis in figure 3, the circuit computes the 3-input NOR function. For the 2nd and the last third of the simulation period the CARRY' and 3-input NAND functions are verified, respectively. The sixth, and lowermost, signal from the top is for a V_{dd} of 0.2 V and $I_{beq} = 1nA$, for which the noise margins are severely detoriated.

5. MEASUREMENTS

We have made measurements on a prototype chip, and here present results for the first stage (out of four) in the layout in figure 4. Figure 5 demonstrate the inverting function for one and the same circuit for two different current levels. The input is changed stepwise from 0 V to 0.8 V /V_{dd}. The lower current level of the two corresponds to the steepest voltage slope, or highest voltage gain. Leftmost in figure 6 the output for the CARRY' function, changing from 0.787

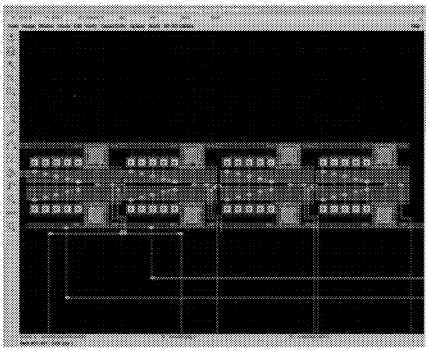

Fig. 4. Layout with four circuits connected as a FULL-ADDER. All W/L drawn are 20.8u/1.2u, and poly1-poly2 capacitors designed for 14.7 fF units.

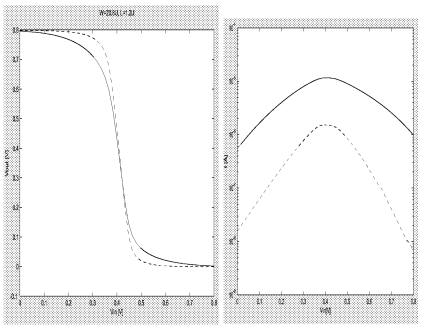

Fig. 5. Measurements for the inverting function of our circuit. Output voltage as function of input voltage is shown(left), and corresponding current (right) for I_{beq}-levels of roughly 10 and 1 uA, and a V_{dd} of 0.8 V.

to 0.032 V is depicted, corresponding to the output transition between lines 7 and 8 in figure 2. V_{dd}=0.8 V. The 1 to 0 transitions for the NOR and NAND functions are shown in the middle and rightmost in figure 6, respectively. The output for the 3-input NOR function changes from 0.786 to 0.134 V. For the NAND function the high/1 output is 0.796 V and the low/0 output 0.117 V, when the number of 1s on the inputs changes from two to three.

6. DISCUSSION

We have verified the inverter function in figure 5, and the CARRY' function by measurements shown leftmost in figure 6. When we have the CARRY' functionality, we also have the 2-input NAND and 2-input NOR functions embedded [12]. Measurements related to the 3-input NOR and 3-input NAND functionalities are shown in the middle and

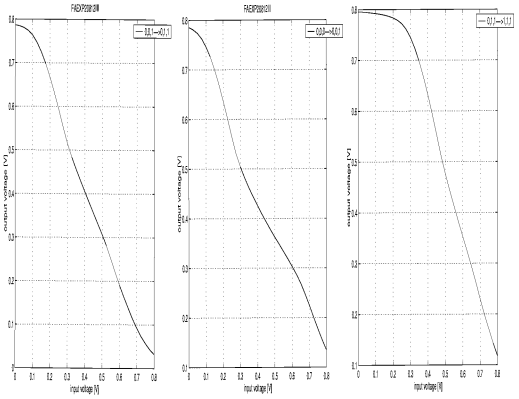

Fig. 6. Measurements of output voltages as functions of input voltages increasing from 0.0 to 0.8 Volts, for CARRY', NOR3 and NAND3 functions are shown. $I_{beq} = 35nA$.

rightmost in figure 6, respectively. The interesting 1 to 0 transistions, depending on number of 1s in the input vector are are depicted. The voltage gains in figure 6 are about -1. Increasing the gain could be done by lowering current levels or increasing output resistances for transistors for a given topology. Increasing the size of all capacitors drawn in figure 1 could also probably increase voltage gain from the input to the output. Signals might also be regenerated by using inverters with relatively high gain, like in figure 5. Anyway, gains and noise margins are not as good as expected from other experience and measurements on comparable circuits. Suboptimal test stimulus to the circuit are among suspected reasons.

The circuit bears resemblance to neuron MOS building blocks proposed in [16]. There, variable threshold inverters are used to build a Soft Hardware Logic Circuit. By wiring X,Y,Z together for our circuit and regulate the level W, we can make variable threshold inverters. A 3-bit A/D converter using only 6 MOSFETs can also be built using a similar building block, by adding the number of inputs, very similar to the one shown in [17].

Two different programming voltages are applied to VDD and VSS of FGUVMOS circuits under UV-programming. The more circuitry on chip that can share VDD and VSS the better. Different basic circuit building blocks might still need individual access to power rails and wells under UV-programming and test [14]. This can be necessary to make all circuit blocks work in subthreshold [13],[14], to control current levels, or to match switching voltages for diverse inverting structures when one have a certain logical depth of the circuitry, for an example [15]. This unfortunately makes the design process more cumbersome, and increases both the area of the chip and the number of pads. All compound digital FGUVMOS circuits reported earlier, like full-adder structures, latches and others for binary addition in [7],[12], have relied on different building blocks for different basic functions like NAND, NOR etc. If we wire together the inputs of our circuit, it becomes an inverter, with two

identical capacitors between the input and floating-gates of the PMOS and NMOS respectively. In [13] twelve identical inverters were coupled in series, with "coupling capacitors" both 18.4 fF and W/L=10u/0.6u, and programmed using common programming voltages on all rails and wells. The outputs were afterwards measured at the switching point for inverters number 3, 9 and 12 in the chain. Inverter number 3 were used as a reference. The offsets between this reference point and the voltages on outputs of number 9 and 12 were less than 10 mV. Programming identical inverters saves chip area, and is probably far simpler than programming several different building blocks at a time. Our elements should resemble this type of inverter chain, while being UV-programmed, if coupled in series. We hope they can ease programming of digital circuits compared to previous approaches, and help us being able to build more complex systems using UV-programming, and save chip area at the same time.

Simulations have indicated Power-Delay-Product down to 5 fJ. Additional chip measurements should be done to verify the actual low-power performance.

7. CONCLUSION

The circuits are behaving functionally as expected, though the performance could probably be better. This might be verified through new measurements. It would also be interesting to try to program a row of our elements, to try and verify by measurements their usefullness as universal building blocks compared to earlier digital approaches. The functionalities CARRY', NAND, NOR and INVERT are shown theoretically, by simulation and measurements. Inverters based on the element can have different thresholds. The Boolean function can be changed in real time, without reprogramming the floating-gates.

8. REFERENCES

[1] Carver Mead Neuromorphic Electronic Systems, Proceedings of the IEEE, No. 10, Oct. 1990, pp. 1629-1636.

[2] Vasken Bohossian Neural Logic: Theory and Implementation, Ph.D. thesis, California Institute of Technology, July 1998.

[3] K. Y. Siu, J. Bruck Neural Computation of Arithmetic Functions Proc. of the IEEE, Oct. 1990, pp. 1669-1675.

[4] Y. Berg, T. S. Lande Area Effecient Circuit Tuning with Floating-Gate Techniques Proc. of the IEEE International Symposium on Circuits and Systems, 1999, Vol. 2, pp. 396-399.

[5] Austria Mikro Systeme International AG 0.6 um CMOS CUP Process Parameters Document no. 9933011, Rev. B, Oct. 1998.

[6] P. Hasler, T. S. Lande Overview of Floating-Gate Devices, Circuits and Systems IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing, January 2001, Vol. 48, Issue 1, pp. 1-3.

[7] Y. Berg, D. T. Wisland,T. S. Lande Ultra Low-Voltage/Low-Power Digital Floating-Gate Circuits, IEEE Transactions on Circuits and Systems II, July 1999, Vol. 46, Issue 7, pp. 930-936.

[8] A. Low, P. Hasler Cadence-based simulation of floating-gate circuits using the EKV model Proceedings of the 42nd Midwest Symposium on Circuits and Systems, 2000, Vol. 1, pp. 141-144.

[9] J. Ramirez-Angulo, G. Gonzalez Altamarino, S. C. Choi "Modeling multiple-input floating-gate transistors for analog signal processing. Proc. of the IEEE International Symposium on Circuits and Systems, 1997, Vol. 3, pp. 2220-2223.

[10] E. Rodriguez-Villegas, G. Huertas, M. J. Avedillo, J. M. Quintana, A. Rueda A Practical Floating-Gate Muller-C Element Using vMOS Threshold Gates IEEE Transactions on Circuits and Systems II, Jan. 2001, Vol. 48, pp. 102-106.

[11] S. Aunet, Y. Berg, T. Ytterdal, Ø. Næss, T. Sæther A Method for Simulation of Floating-Gate UV-Programmable Circuits with Application to Three New 2-MOSFET Digital Circuits. Proc. of the IEEE International Conference on Electronics, Circuits and Systems, 2001, Vol. 2, pp. 1035-1038.

[12] S. Aunet, Y. Berg, O. Tjore, Ø. Næss, T. Sæther Four-MOSFET Floating-Gate UV-Programmable Elements for Multifunction Binary Logic. Proceedings of the 5th World Multiconference on Systemics, Cybernetics and Informatics, SCI 2001, Orlando, FL, USA, July 22-25, 2001, Vol. III, pp. 141-144.

[13] H. Gundersen Design of Low-Voltage Analog Amplifiers using Floating-Gate Transistors, thesis for the cand. scient. degree, Department of informatics, University of Oslo, May 2000.

[14] R. Bahr Design of Linear Four Quadrant Analog Multipliers using Floating-Gate Transistors, thesis for the cand. scient. degree, Department of informatics, University of Oslo, May 2001.

[15] R. Danielsen Effektoptimalisering av digitale CMOS-kretser (floating-gate), thesis for the cand. scient. degree, Department of informatics, University of Oslo, May 2001.

[16] T. Shibata, T. Ohmi An Intelligent MOS Transistor Featuring Gate-Level Weighted Sum and Threshold Operations, Technical Digest, International Electron Devices Meeting, 1991, pp. 919-922.

[17] A. Rantala, S. Franssila, K. Kaski, M. Åberg, P. Kuivalainen Improved neuron MOS-transistor structures for integrated neural network circuits, IEE Proc.-Circuits Devices Syst., Vol. 148, No. 1, February 2001, pp. 25-34.

2) "Novel reconfigurable two-MOSFET UV-programmable floating-gate circuits for CARRY, NAND, NOR or INVERT functions

from Proceedings of the 2001 IEEE International Conference on Electronics, Circuits and Systems, Vol. 2, pages 528-531

Novel Reconfigurable Two-MOSFET UV-programmable Floating-Gate Circuits for CARRY', NAND, NOR or INVERT functions.

Snorre Aunet, Yngvar Berg*, Øivind Næss*, Trond Sæther

Department of physical electronics, Norwegian University of Science and Technology, O.S. Bragstads plass 2A, N-7034 Trondheim, Norway. E-mail: Snorre.Aunet@fysel.ntnu.no
*Department of informatics, University of Oslo, Gaustadaleen 23, N-0316 Oslo, Norway.

ABSTRACT

This article presents two new, reconfigurable multi-function floating-gate circuit to produce either the INVERTED CARRY function for a FULL-ADDER, an INVERTER, or two-input NAND or NOR gates. The circuits contains two MOSFETs and three or four capacitively coupled input signals. SPICE simulations are shown, demonstrating the principal operation, together with preliminary measurements indicating that the circuit might operate with a supply voltage below 100 mV.

1. INTRODUCTION

THERE are continously going on large efforts to make electronic systems more and more economical in terms of power per operation. One out of many approaches are different techniques involving floating-gate systems, originally invented in 1967 as a digital storage element [11]. Different floating-gate elements have during the last decade become used as circuit elements in both analog and digital circuits [1]. Clever circuit design combined with silicon technology device scaling are helping to reduce power requirements. Untraditional design methods might be another way of making more effecient electronic systems in terms of chip area and power dissipation. Floating-gate systems resembling the biological neuron, making possible circuits featuring a factor 5-10 reduction in gate count can be made using standard CMOS, or GaAs technologies [2],[3]. One such approach has been demonstrated by the neuron MOSFET, or neuMOS principle, in 1991 [3],[4]. Many systems include multiple-input floating-gate transistors which calculate the weighted sum of all input signals to the floating-gate to control the transistor. Charge on the floating-gate can be modified by the use of UV-light, by tunneling electrons through the oxide or hot-electron injection [1]. UV-light can be used to change the charge on the floating-gate in any direction, to make both analog and digital "FGUVMOS" circuits [5],[6]. By adjusting the effective threshold voltages and current levels by UV-programming, and operating the circuits in subthreshold, one can make digital circuits operating with a power supply voltage typically in the 0.3 - 0.8 V area, while still operating circuits up to the several-hundred MHz range [6]. The digital circuits earlier reported use different circuits for 2-input NAND and NOR, both needing 3 transistors. They also propose a FULL-ADDER circuit using 30 transistors with 22 transistors contributing to the INVERTED CARRY. This new 2-transistor approach for the INVERTED CARRY function resembles the neuron MOS principle used in [2],[3],[4]. The circuit element proposed can also be used as a 2-input NAND, NOR or INVERTER circuit, depending on the logic level of one of the input signals. To be able to use the same module for different logic functions will probably easily enable us to better the control of current levels compared to earlier design methods. Spice simulations and preliminary measurement results demonstrate the principle, and suggest that supply voltages down to a few hundred mV can be used, for circuits in the AMS 0.6 double-poly CMOS technology [9].

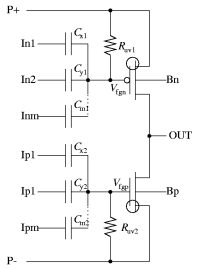

Fig. 1. FGUVMOS circuit in programming mode. Conductances allow flow of charges during period of UV-light exposure, until charge on the floating gates are practically permanently altered when UV-light is removed. Circles indicate FGUVMOS transistors.

2. FGUVMOS CIRCUITS

Typical current levels (I_{beq}) for FGUVMOS circuits based on measurements on chips produced in the AMS 0.8 and 0.6 CMOS processes lie roughly in the uA to nA area, depending on the effective threshold voltages and power supply voltages chosen. In

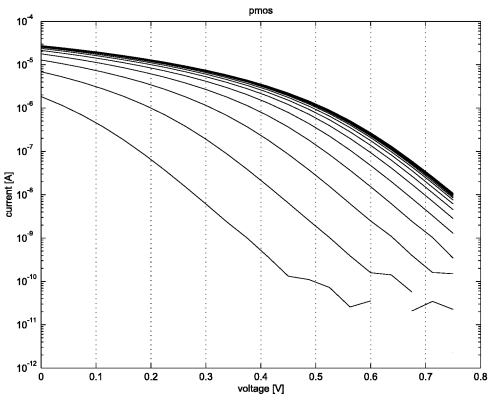

Fig. 2. Measured I-V curves for PMOS (W/L=20.8um/1.2um), and increasing current levels, after 3-minute intervals of UV-programming of effective threshold voltage.

figure 2 the current level increase for a PMOS transistor for each 3-minute period of UV-light exposure. The FGUVMOS transistors use metal and passivation as "shields" to let UV-light easily hit selected parts of the chip surface. More about layout techniques can be found in [5] and [7]. FGUVMOS circuits can be operated both in weak and strong inversion. Here we will view them from a subthreshold point of view. We have one PMOS and one NMOS stacked, with one or several capacitively coupled inputs to the floating-gate of each of the two. One such PMOS/NMOS element, or several in series, could here be viewed as the generic FGUVMOS circuit, like in [5]. For such a circuit the currents through the PMOS and NMOS transistors could be expressed as a modulation of the equilibrium condition.

$$I_{ds,p} = I_{beq} \prod_{i=1}^{m} exp\{\frac{1}{nU_t}(V_{dd}/2 - V_i)k_i\} \qquad (1)$$

$$I_{ds,n} = I_{beq} \prod_{i=1}^{m} exp\{\frac{1}{nU_t}(V_i - V_{dd}/2)k_i\} \qquad (2)$$

$k_i = C_i/C_{tot}$ is the capacitive division factor of the ith input capacitor, C_i, and C_{tot} is the total capacitance seen from the floating-gate. I_{beq} is the balanced equilibrium current, which is the drain current of the transistors when all ordinary input signals are equal to $V_{dd}/2$ If we connect the inputs X,Y,Z in figure 3 together, we have an inverting structure whose netlist is equal to that of a digital inverter. Measured output voltage as a function of input voltage for an implementation of an identical circuit, with W/L=20.8um/ 1.2um and two 73.5fF capacitors, is shown in figure 4. For parts of the slope the gain is < -1 for Vdd down to 93 mV. There are two curves for Vdd equal to 0.8 V. The one with the steepest slope is for an I_{beq} of smaller value than the other. In general, to lower current level leads to higher gain for an FGUVMOS circuit. Current levels corresponding to the voltage curves is shown in figure 5.

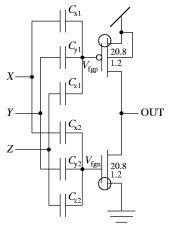

Fig. 3. Schematic for a carry'-,nand-,nor-, invert-circuit.

3. RECONFIGURABLE CIRCUITS

The circuit depicted in figure 3 was simulated using Eldo and the BSIM3v3 model and parameters for the AMS 0.6 CUP double-poly standard CMOS technology. The simulation shown in figure 8, for a Vdd of 300 mV, corresponds to the truth table in figure 9. P- and N- transistors with W/L = 20.8/1.2 um were used, together with 24.5 fF capacitors between each input and the floating-gates, to make it resemble the implemented chip used for preliminary measurements shown in figures 2, 4 and 5. Photographies from the chip and a picture of a "donut" transistor layout are shown in figures 6 and 7, respectively.

$$F = (xy + xz + yz)' \qquad (3)$$

If at least two out of three inputs are '1', the output goes to '0'. This function is identical to the CARRY' for a FULL-ADDER circuit.

From the truth table, in figure 9, one can see that by letting any one input be $'0'$, the output is 0 if, and

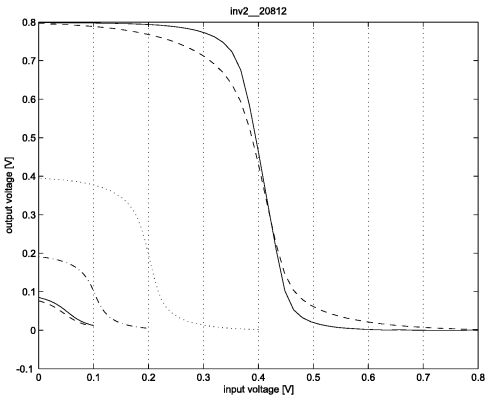

Fig. 4. Output voltage as a function of input voltage for different Vdd levels; 93 mV, 100 mV, 200 mV, 800 mV.

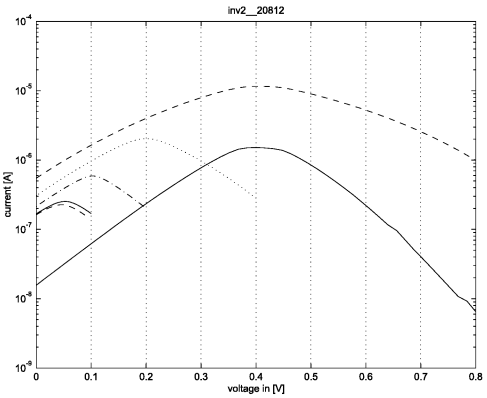

Fig. 5. Measured I_{beq}'s as function of Vdd and input voltages.

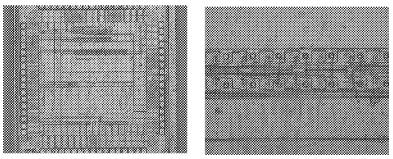

Fig. 6. Photographies from the prototype chip.

only if, both other inputs are $'1'$. Then the circuit implements the NAND function. If any one input is 1, the output is $'1'$ if and only if both other inputs are $'0'$. Then the circuit operates as a 2-input NOR. Connecting one input to Vdd or Vss and short-circuiting the other two give an INVERTER. A 2-input inverting-structure like NAND or NOR is essentially the only function needed to express any given Boolean expression or digital function. Earlier reported digital FGUVMOS circuits used 5 different modules and 22 transistors to generate the CARRY' function [6]. A binary comparator that uses 5 MOSFETs to implement AND / OR / CARRY can be found in [10].

As illustrated in figure 2, the power rails are used to determine the wanted current level or effective threshold voltages. As many modules as possible should share Vdd/Vss. From this viewpoint it could be attractive to use as few different modules as possible. In a digital module one would most often have current levels close to each other in magnitude, though it could sometimes be attractive to "balance" delay

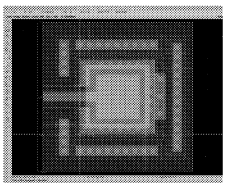

Fig. 7. Layout of "Donut" FGUVMOS transistor of N-type, with W/L=20.8um/1.2um

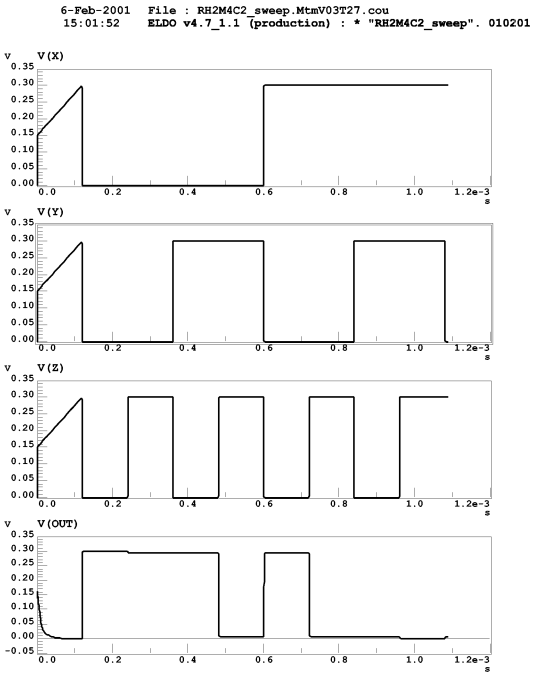

Fig. 8. Simulation of carry' function. The curves represent the voltages on X,Y,Z and F / V(OUT).

X	Y	Z	F
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	0

Fig. 9. Truth table for the reconfigurable circuit.

through different paths in a digital circuit by adjusting the individual effective threshold voltages / current levels [8]. Using the same module extensively could ease the work with matching current levels/ threshold levels, though current levels can be matched far better than if different circuitry for every type of Boolean function were used [7]. Using the proposed circuit could reduce the degree of design care to match current levels substantially, between for example a 2-input NAND and a 2-input NOR, simply by using the same circuit element for both logic functions. One of the three inputs then determine what function to implement in each case.

An important issue determining the lower bounds for Vdd is the noise margin. For this circuit the critical difference is when inputs change between (x,y,z)

Vdd(mV)	V_h(mV)	V_l(mV)	V_{h_p}	V_{l_p}
93	68	22.8	73	75.5
300	293	6.6	97.7	97.8
400	395	4.0	98.8	99.0
500	497	2.5	99.4	99.5
600	598.1	1.6	99.7	99.7
700	698.8	1.0	99.8	99.9
800	799.2	0.6	99.9	99.9

Fig. 10. Noise margins as a function of Vdd. 2nd and 3rd column are output in mV. 4th and 5th are percentage of perfect output level.

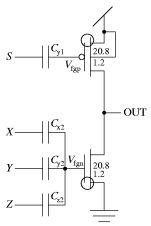

Fig. 11. A 2nd CARRY'-,NAND-,NOR and INVERT-circuit.

= (0,0,1) and (0,1,1). The output voltage needs to define a '0' or '1'. We have simulated the circuit like in figure 8, for an equilibrium current, $I_{beq} = 10nA$. Results can be found in figure 10. Another solution reducing the number of capacitors from 6 to 4 could be made by changing the input capacitors to either the NMOS or PMOS to one, and let this input have a fixed potential, like in figure 11. Some simulation results for circuits in figures 3 and 11, with (W/L)= 1.2/0.6 and capacitors of 10 fF, for two different current levels and load capacitors of 10 fF to both Vdd and Vss, can be found in figures 12 and 13.

I_{beq}(uA)	V_h(mV)	V_l(mV)	$t_r(ns)$	t_f(ns)
0.01	799.9	0.08	103	89
1	794.4	3.9	6.9	3.6

Fig. 12. Simulated noise margins and rise- and fall-times for the 6-capacitor circuit.

I_{beq}(uA)	V_h(mV)	V_l(mV)	$t_r(ns)$	t_f(ns)
0.01	798.2	3.1	178	90
1	789.1	24	19.3	4.5

Fig. 13. Simulated noise margins and rise- and fall-times for the 4-capacitor circuit. Voltage on node S is Vdd/2.

4. DISCUSSION / CONCLUSION

We have suggested two new reconfigurable FGUVMOS circuits for ultra low-voltage applications, which might save area for a given technology, as well as ease matching of current levels. For an example was the number of transistors for the INVERTED CARRY function reduced to two, which is several times lower than conventional methods. Measurements have shown that the proposed circuit worked as an inverter for a Vdd of 93 mV, though this might not be a practical Vdd for more complex circuits, due to low gain and noise margins. A 6-transistor and a 4-transistor implementation were made. The first has better noise margins and more symmetric rise- and fall-times than the latter. A new chip with the proposed circuits hopefully give more insight in performance of circuits based on these ideas. Simulations have demonstrated the main principles, together with relevant measurements from a prototype chip, looking promising for future implementations.

References

[1] P. Hasler, B. A. Minch, C. Diorio *Floating-Gate Devices: They are not just for digital memories anymore.* Proceedings of the IEEE International Symposium on Circuits and Systems, 1999.

[2] D. Abbott, S. F. Al-Sarawi, B. Gonzalez, J. Lopez, J. Austin-Crowe, K. Eshraghian *Neu-MOS for Smart Sensors and Extension to a Novel Neu-GaAs Paradigm* IEEE International Conference on Electronics, Circuits and Systems,1998.

[3] T. Shibata, T. Ohmi *An Intelligent MOS Transistor Featuring Gate-Level Weighted Sum and Threshold Operations*, Technical Digest, International Electron Devices Meeting, 1991.

[4] K. Kotani, T. Shibata, T. Ohmi *Neuron-MOS Binary-Logic Circuits Featuring Dramatic Reduction in Transistor Count and Interconnections* Technical Digest, International Electron Devices Meeting 13-16. December, 1992.

[5] Y. Berg, T. S. Lande *Area Efficient Circuit Tuning with Floating-Gate Techniques*, Proceedings of the IEEE International Symposium on Circuits and Systems,1999.

[6] Y. Berg, D. T. Wisland, T. S. Lande *Ultra Low-Voltage/Low-Power Digital Floating-Gate Circuits*, IEEE Transactions on Circuits and Systems II, July 1999.

[7] H. Gundersen *Design of Low-Voltage Analog Amplifiers Using Floating-Gate Transistors*, Thesis for the cand. scient. degree, Dept. of Informatics, University of Oslo, Norway, May 2000.

[8] A. P. Chandrakasan, S. Sheng, R. W. Brodersen *Low-Power CMOS Digital Design* IEEE Journal of Solid-State Circuits, April, 1992.

[9] Austria Mikro Systeme International AG *0.6 um CMOS CUP Process Parameters* Document no. 9933011, Rev. B, Oct. 1998.

[10] J. Ramirez Angulo, G. Gonzalez-Altamirano *A New Programmable Logic Family Using Multiple-input Floating-Gate Transistors* Proc. of the 40th Midwest Symp. on Circuits and Syst., 1997.

[11] D. Kahng, S. M. Zhe *A floating-gate and its application to memory devices* The Bell System Technical Journal, no. 4, 1967.

3) Four-MOSFET Floating-Gate UV-Programmable Elements for Multifunction Binary Logic :

from Proceedings of the
2001 World Multiconference on Systemics,
Cybernetics and Informatics (SCI 2001),
pages 141-144.

Four-MOSFET Floating-Gate UV-Programmable Elements for Multifunction Binary Logic

Snorre Aunet 1), Yngvar Berg 2), Odd Tjore 1) Øivind Næss 2), Trond Sæther 1)
1) Department of Physical Electronics, Norwegian University of Science and Technology, N-7491 Trondheim, Norway.
2) Department of Informatics, University of Oslo, N-0316 Oslo, Norway.

Abstract— **This article presents two new, reconfigurable multi-input floating-gate circuits able to produce either the INVERTED SUM / INVERTED CARRY functions for a FULL-ADDER, XNOR / NAND, or XOR / NOR. The circuits contain four MOSFETs and fourteen or nine capacitors, respectively. SPICE simulations are shown, demonstrating the principal operation, together with preliminary measurements indicating that the circuits might operate in subthreshold, with supply voltages down to a few hundred mV. Measurements demonstrate the inverter function for V_{dd} from 93 mV up to 800 mV.**

Keywords— **FGUVMOS, universal element, XNOR, NAND, XOR, NOR, full-adder.**

I. Introduction

THERE are continously going on large efforts to make electronic systems more and more economical in terms of power per operation. One out of many approaches are different techniques involving floating-gate systems, originally invented in 1967 as a digital storage element [1,11]. Different floating-gate elements have during the last decade become used as circuit elements in both analog and digital circuits [1]. Clever circuit design combined with silicon technology device scaling are helping to reduce power requirements. Untraditional design methods might be another way of making more effecient electronic systems in terms of chip area and power dissipation. Floating-gate systems resembling the biological neuron, making possible circuits featuring a factor 5-10 reduction in gate count can be made using standard CMOS technologies, or GaAs technologies [2],[3]. One such approach has been demonstrated by the neuron MOSFET, or neuMOS principle, in 1991 [3],[4]. Many systems include multiple-input floating-gate transistors which calculate the weighted sum of all input signals to the floating-gate to control the transistor. Charge on the floating-gate can be modified by the use of UV-light, by tunneling electrons through the oxide or hot-electron injection [1]. UV-light can be used to change the charge on the floating-gate in any direction, to make both analog and digital "FGUVMOS" circuits [5],[6]. By adjusting the effective threshold voltages and current levels by UV-programming, and operating the circuits in subthreshold, one can make digital circuits operating with a power supply voltage typically in the 0.3 - 0.8 V area, while still operating circuits up to the several-hundred MHz range [6]. The digital circuits earlier reported use different circuits for every basic logic function like for example NOR, NAND and XOR [6]. They also propose a FULL-ADDER circuit using 30 transistors with 22 transistors contributing to the INVERTED CARRY (CARRY'), and 16 transistors for the INVERTED SUM (SUM'). We use 2 transistors for the CARRY', and two more for SUM'. This new 4-transistor approach for binary logic functions resembles the neuron MOS principle used in [2],[3],[4]. Since the proposed circuits contain four transistors there are fewer paths between V_{dd} and V_{ss}, which in many cases allow reduced power dissipation. We show one version with 14 and one with 9 capacitors. If circuit elements could be small, in spite of the capacitors, they might not be too costly if used in a parallellization scheme for increased performance. The two circuit structures proposed can also be used as, for example, 2-input XNOR/NAND, XOR/NOR or INVERTER circuits, depending on the logic level of one of the input signals. The realtime reconfigurable options might be used for correcting errors by changing the boolean functions. To be able to use the same module for different logic functions might enable us to better the control of current levels compared to previously reported design methods as in [6]. One might also reduce the number of different modules necessary to fit into a synthesis tool. Spice [12] simulations and preliminary measurement results demonstrate the principle, and suggest that supply voltages down to a few hundred mV can be used, for circuits in the AMS 0.6 um double-poly CMOS technology [9].

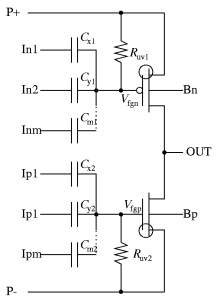

Fig. 1. FGUVMOS circuit in programming mode. Conductances allow flow of charges during period of UV-light exposure, until charge on the floating gates are practically permanently altered when UV-light is removed. Circles indicate FGUVMOS transistors.

II. Floating-Gate UV-programmed CMOS (FGUVMOS) circuits

Typical equilibrium current levels for FGUVMOS circuits based on measurements on chips produced in the AMS 0.8 and 0.6 CMOS processes lie roughly in the uA to nA area, depending on the effective threshold voltages and power supply voltages chosen. In figure 2 the current level increases after periods of UV-light exposure. The FGUVMOS transistors use metal and passivation as "shields" to let UV-light easily hit selected parts of the chip surface. More about layout techniques can be found in [5] and [7]. FGUVMOS

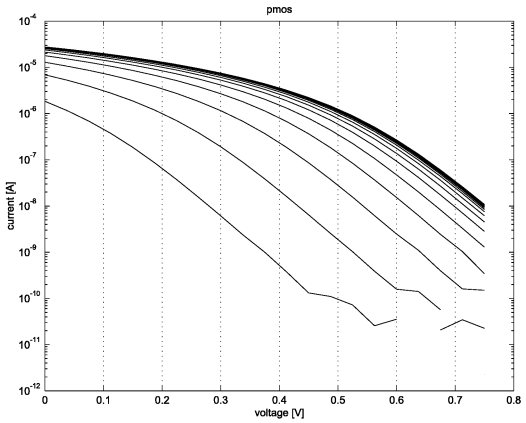

Fig. 2. Measured I-V curves for PMOS (W/L=20.8um/1.2um), and increasing current levels, after 3-minute intervals of UV-programming of effective threshold voltage.

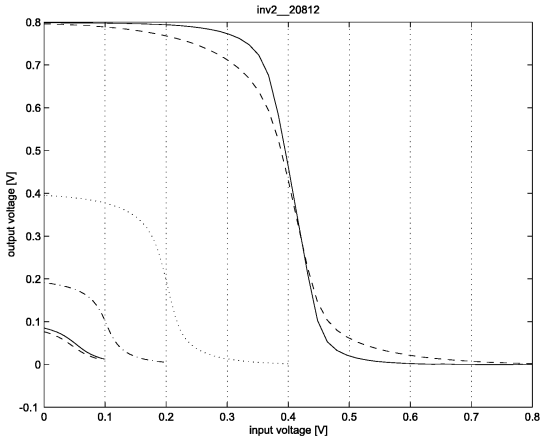

Fig. 4. Output voltage as a function of input voltage for different V_{dd} levels; 93 mV, 100 mV, 200 mV, 400 mV, 800 mV.

circuits can be operated both in subthreshold and above threshold. Here we will view them from a subthreshold point of view. We have one PMOS and one NMOS stacked, with one or several capacitively coupled inputs to the floating-gate of each of the two. One such PMOS/NMOS element, or several in series, could here be viewed as the generic FGUVMOS circuit, like in [5]. For such a circuit the currents through the PMOS and NMOS transistors could be expressed as a modulation of the equilibrium condition, shown in equations 1 and 2.

$$I_{ds,p} = I_{beq} \prod_{i=1}^{m} exp\{\frac{1}{nU_t}(V_{dd}/2 - V_i)k_i\} \quad (1)$$

$$I_{ds,n} = I_{beq} \prod_{i=1}^{m} exp\{\frac{1}{nU_t}(V_i - V_{dd}/2)k_i\} \quad (2)$$

$$k_i = C_i/C_{tot} \quad (3)$$

k_i is the capacitive division factor of the i^{th} input capacitor, C_i, and C_{tot} is the total capacitance seen from the floating-gate. I_{beq} is the balanced equilibrium current, which is the drain current of the transistors when all ordinary input signals are equal to $V_{dd}/2$ If we connect the inputs X,Y,Z in figure 3 together, we have an inverting structure which netlist is equal to that of a floating gate inverter. Measured output voltage as a function of input voltage for an implementation of a similar circuit using single input capacitors, with W/L=20.8um/ 1.2um and two 73.5fF capacitors, is shown in figure 4. For parts of the slope the gain is < -1 for V_{dd} down to 93 mV. There are two curves for V_{dd} equal to 0.8 V. The one with the steepest slope is for an I_{beq} of smaller value than the other. In general, to lower the current level leads to higher gain for an FGUVMOS circuit. Current levels corresponding to the voltage curves is shown in figure 5.

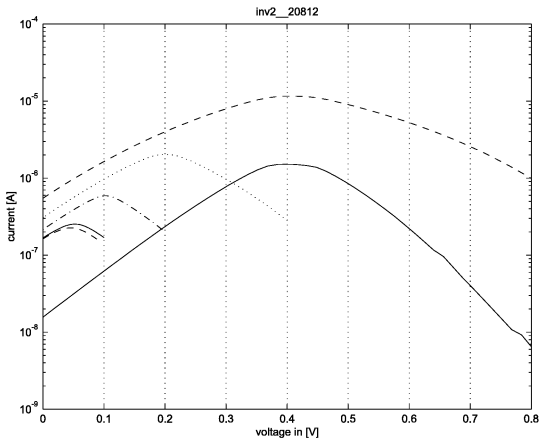

Fig. 5. Measured currents as function of V_{dd} and input voltages corresponding to figure 4.

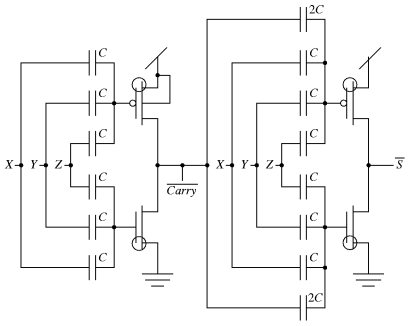

Fig. 3. Schematic for 14 capacitor version of the circuit.

III. Reconfigurable circuits

A. Reducing the number of MOSFETs for implementing CARRY' to 2, and SUM' to 4

The circuits depicted in figure 3 and figure 7 were simulated using Eldo and the BSIM3V3 model and parameters for the AMS 0.6 um CUP double-poly standard CMOS technology [9]. The simulation shown in figure 8, for a V_{dd} of 400 mV, corresponds to the truth table in figure 9. P- and N- transistors with W/L = 20.8/1.2 um were used, together with 73.5 fF total capacitance between each driving node and the floating-gates, to make it resemble the implemented chip used for preliminary measurements shown in figures 2, 4 and 5. Die photos from the chip are shown in figure 6. The Boolean expressions for SUM'(S') and CARRY'(C') can be found in equations 4 and 5.

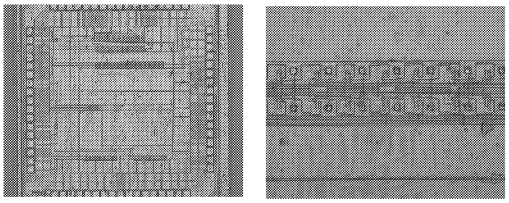

Fig. 6. Die photos from a prototype chip. The photo to the right contains 7 NMOS and 7 PMOS designed for W/L=20.8u/1.2u as well as 14 capacitors designed for 73.5 fF capacitance each.

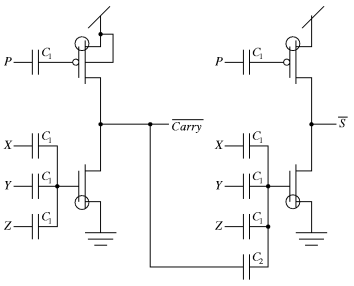

Fig. 7. Schematic for 9-capacitor version of the circuit.

$$S' = x'y'z' + x'yz + xy'z + xyz' \quad (4)$$

$$C' = x'y'z' + x'y'z + x'yz' + xy'z' = (xy + xz + yz)' \quad (5)$$

If at least two out of three inputs are '1', the output goes to '0'. This function is identical to the CARRY' for a FULL-ADDER circuit.

B. Eliminating the need for different circuitry for basic Boolean functions

By inspection of the truth table in figure 9 one can see that if any one input be $'0'$, the output on node S' is $'0'$ if, and only if, both other inputs are equal. Then the circuit S'-node gives the XNOR function. The C' node then provides the NAND function. If any one input is '1' the S' and C' nodes give XOR and NOR functions, respectively. Short-circuiting the other inputs gives the INVERTED input on

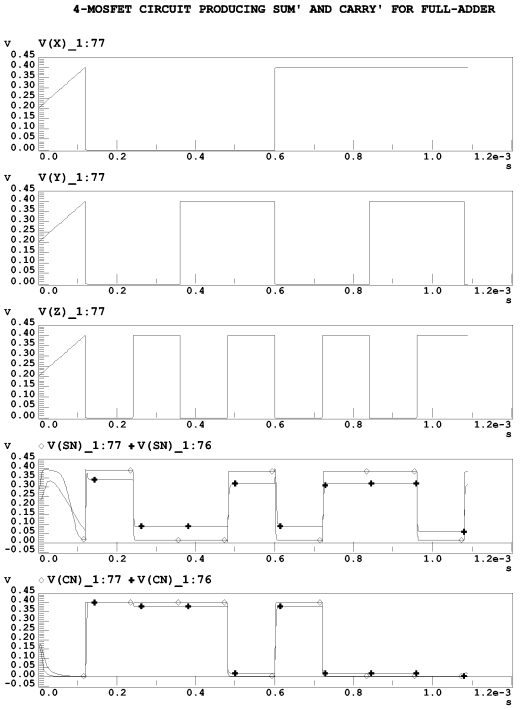

Fig. 8. Simulation of SUM' and CARRY' function. The seven curves represent from top to bottom, the voltages on X,Y,Z, S'('SN') and C'('CN') for 14 and 9 capacitor circuits. The 14 capacitor version has the best noise margins for both the S' and C' nodes.

X	Y	Z	S'	C'
0	0	0	1	1
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	0	0

Fig. 9. Truth table for the reconfigurable circuit.

bot S' and C'. A 2-input inverting-structure like NAND or NOR is essentially the only function needed to express any given Boolean expression or digital function. Another programable logic approach, making a binary comparator that uses 5 MOSFETs to implement AND / OR / CARRY can be found in [10].

C. Matching current levels by using same module for different logic functions.

As illustrated in figure 2, the power rails are used to determine the wanted current level or effective threshold voltages, during UV-programming. As many modules as possible should then share V_{dd} and V_{ss}. From this viewpoint

it could be attractive to use as few different modules as possible. In a digital module one would most often have current levels close to each other in magnitude, though it could sometimes be attractive to "balance" delay through different paths in a digital circuit by adjusting the individual effective threshold voltages / current levels [8]. Using the same module extensively could ease the work with matching current levels/ threshold levels, though current levels can be matched far better than if different circuitry for every type of Boolean function were used [7]. Using the proposed circuit could reduce the degree of design care to match current levels substantially, between for example a 2-input NAND and a 2-input NOR, simply by using the same circuit element for both logic functions. One of the three inputs then determine what function to implement in each case.

D. Noise margins and risetimes/falltimes

An important issue determining the lower bounds for V_{dd} is the noise margin. The output voltage needs to define a '0' or '1'. We have simulated the two circuits, for an equilibrium current, $I_{beq} = 10nA$. Results can be found in figure 10. Some simulation results for risetime and falltime, for circuits in figures 3 and 7, with (W/L)= 1.2/0.6 and capacitors of 10 fF, for 10 nA I_{beq}-level and load capacitors of 10 fF to both V_{dd} and V_{ss}, can be found in figure 11.

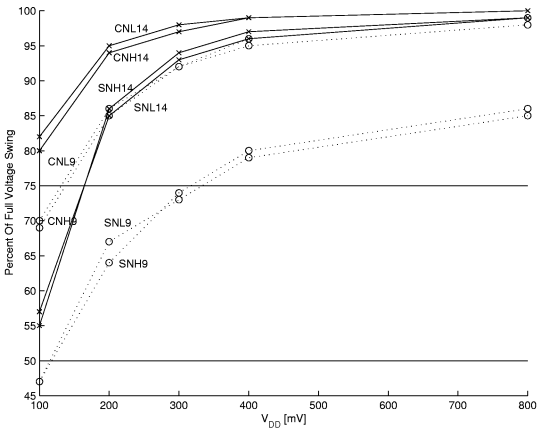

Fig. 10. Simulation of noise margins. "SNL14" means output voltage level in percent of full swing, when a 'low'/'0' level is wanted, and the circuit is the 14 capacitor version.

circuit	$t_{rS'}$(ns)	$t_{fS'}$(ns)	$t_{rC'}(ns)$	$t_{fC'}$(ns)
14 C	80	79	120	98
9 C	530	420	130	66

Fig. 11. Simulated rise- and fall-times for the two 14- and 9- capacitor circuits, for SUM' and CARRY' nodes.

IV. DISCUSSION / CONCLUSION

We have suggested a new reconfigurable FGUVMOS circuit suitable for ultra low-voltage and low-power applications, which might save area for a given technology, as well as ease matching of current levels. For an example was the number of transistors for the CARRY' function reduced to two, and the SUM' function to four, which is several times lower than conventional methods. 14- and 9- capacitor implementations were made. The first has better noise margins and more symmetric rise- and fall-times than the latter. Monte-Carlo simulations and a new chip with the proposed circuits hopefully will give more insight in performance of circuits based on these ideas. Simulations have demonstrated the main principles, together with relevant measurements from a prototype chip, looking promising for future implementations.

REFERENCES

[1] Paul Hasler, Bradley A. Minch, Chris Diorio *Floating-Gate Devices: They are not just for digital memories anymore.* Proceedings of the IEEE International Symposium on Circuits and Systems, 1999.

[2] D. Abbott, S. F. Al-Sarawi, B. Gonzalez, J. Lopez, J. Austin-Crowe, K. Eshraghian *Neu-MOS for Smart Sensors and Extension to a Novel Neu-GaAs Paradigm* IEEE International Conference on Electronics, Circuits and Systems,7-10 September, 1998.

[3] Tadashi Shibata, Tadahiro Ohmi *An Intelligent MOS Transistor Featuring Gate-Level Weighted Sum and Threshold Operations,* Technical Digest, International Electron Devices Meeting, 1991.

[4] Koji Kotani, Tadashi Shibata, Tadahiro Ohmi *Neuron-MOS Binary-Logic Circuits Featuring Dramatic Reduction in Transistor Count and Interconnections* Technical Digest, International Electron Devices Meeting 13-16. December, 1992.

[5] Yngvar Berg, Tor Sverre Lande *Area Efficient Circuit Tuning with Floating-Gate Techniques,* Proceedings of the IEEE International Symposium on Circuits and Systems,1999.

[6] Yngvar Berg, Dag T. Wisland,Tor Sverre Lande *Ultra Low-Voltage/Low-Power Digital Floating-Gate Circuits,* IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing, July 1999.

[7] Henning Gundersen *Design of Low-Voltage Analog Amplifiers Using Floating-Gate Transistors,* Thesis for the cand. scient. degree, Dept. of Informatics, University of Oslo, Norway, May 2000.

[8] Anantha P. Chandrakasan, Samuel Sheng, Robert W. Brodersen *Low-Power CMOS Digital Design* IEEE Journal of Solid-State Circuits, April, 1992.

[9] Austria Mikro Systeme International AG *0.6 um CMOS CUP Process Parameters* Document no. 9933011, Rev. B, Oct. 1998.

[10] Jaime Ramirez Angulo, Gerardo Gonzalez-Altamirano *A New Programmable Logic Family Using Multiple-input Floating-Gate Transistors* Proceedings of the 40th Midwest Symposium on Circuits and Systems, 1997.

[11] D. Kahng, S. M. Zhe *A floating-gate and its application to memory devices* The Bell System Technical Journal, no. 4, 1967.

[12] Mentor Graphics' Eldo transistor level simulator *http://www.mentor.com/eldo/*

4) A New 2-MOSFET Universal Floating-Gate Element for Reconfigurable Digital Logic

A NEW 2-MOSFET UNIVERSAL FLOATING-GATE ELEMENT FOR RECONFIGURABLE DIGITAL LOGIC

Snorre Aunet, Yngvar Berg*, Trond Sæther

Department of physical electronics, Norwegian University of Science and Technology, O.S. Bragstads plass 2A, N-7034 Trondheim, Norway. Telephone: +47 73 59 44 00 Fax: +47 73 59 14 41 E-mail: Snorre.Aunet@fysel.ntnu.no, Trond.Saether@fysel.ntnu.no

*Department of informatics, University of Oslo, Gaustadaleen 23, N-0316 Oslo, Norway. Telephone: +47 22 85 24 10 Fax: +47 22 85 24 01 E-mail: yngvarb@ifi.uio.no

ABSTRACT

We present a new universal digital UV-programmable floating-gate circuit able to work with a supply voltage down to about 180 mV. The two-transistor gate as a stand-alone circuit can implement functions like CARRY' for full-adder, NOR3, NOR2, NAND3, NAND2 and INVERT using only 2 MOSFETs. Using two identical circuits wired together enables us to generate, for example, CARRY' and SUM' for the FULL-ADDER function using 4 MOSFETs, or NAND / XNOR or NOR / XOR, By choosing to use one of the inputs as a control input, one can change the Boolean function of the circuit in real time. Using the same basic circuit structure extensively may save area and complexity for a given function.

1. INTRODUCTION

FLOATING-GATE circuits have become analog and digital signal-processing elements during about the last decade, though they were originally reported for digital memory purposes [1]. Neuron MOSFET circuits, inspired by the biological neuron, have been introduced for analog and digital processing, and might reduce both wiring and transistor count dramatically [3],[4],[5]. Analog memory and adaptive circuits are other areas, and neuromorphic systems yet another area of research including floating-gate circuits [2]. Manipulating the charge on floating-gates can be done using Fowler-Nordheim tunneling and hot-electron injection, or by using UV-light [2],[6]. We use UV-light both to add and remove charge on the floating-gates (FGUVMOS circuits). The technique can be used for making different effective threshold voltages for transistors seen from a driving node. FGUVMOS circuits may be used to implement analog and digital low-power/low-voltage signal processing circuits typically operating with a V_{dd} in the 300 - 800 mV area, with both NMOS and PMOS transistors constantly working in subthreshold, while still being able to operate digital circuits in the MHz range[6],[7],[8]. We will present a simple analysis of a new universal, reconfigurable digital element with significantly lower circuit complexity than earlier approaches. Another difference compared to earlier digital FGUVMOS circuits is that it does not use different circuitry for implementing different functions like for example NAND, NOR and XOR, as in [6],[8],[10].

2. FGUVMOS CIRCUITS

FGUVMOS circuits are here briefly treated from the subthreshold point of view, though many FGUVMOS circuits can be used above threshold as well. For a more thorough introduction to FGUVMOS circuits you might confer [2],[6],[7],[10]. Every FGUVMOS circuit element has one PMOS- stacked on top of one NMOS-transistor, and one or more capacitively coupled connections to the floating-gates, as in figure 1. We can use the following equations to describe the subthreshold behaviour of FGUVMOS circuits [6]:

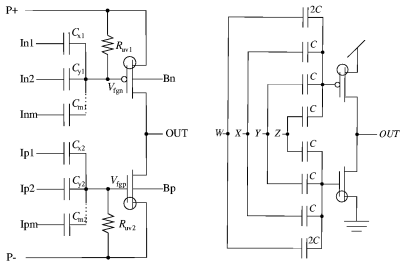

Fig. 1. The effective threshold voltages is determined in a programming procedure involving UV-light activated conductances, and a special biasing scheme. A simple illustration of this scheme, for a multi-input FGUVMOS circuit is shown to the left. Our new universal circuit is shown rightmost, with circles indicating FGUVMOS transistors.

$$I_{ds,p} = I_{beq} \prod_{i=1}^{m} exp\{\frac{1}{nU_t}(V_{dd}/2 - V_i)k_i\} \tag{1}$$

$$I_{ds,n} = I_{beq} \prod_{i=1}^{m} exp\{\frac{1}{nU_t}(V_i - V_{dd}/2)k_i\} \tag{2}$$

Here, $k_i = C_i/C_{tot}$ is the capacitive division factor of the ith input capacitor, C_i, while C_{tot} is the total capacitance seen from the floating-gate. I_{beq} is the balanced equilibrium current, which is the drain current of the transistors when input signals and driven nodes are equal to $V_{dd}/2$. I_{beq} is a function of the effective threshold voltages seen from the driving nodes, and measurements indicate typically current levels in the nA to uA area, for implementations in the AMS 0.6 CUP process [11]. For a certain I_{ds} level for PMOS and NMOS there is one set of effective threshold voltages. An FGUVMOS circuit can have different I_{ds} levels for different transistors, due to different layouts, transistor sizes, topologies and UV-programming. In our example we use the same I_{beq} levels / effective threshold voltages for all transistors.

3. RECONFIGURABLE UNIVERSAL DIGITAL ELEMENT

3.1. Using one two-MOSFET universal element to generate NAND, NOR, CARRY' or INVERT

Using the above equations we get:

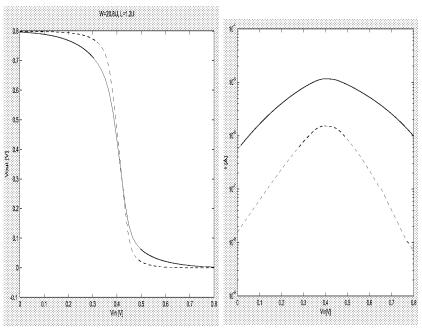

Fig. 2. Measurements for inverting function. Output voltage as function of input voltage (left), and corresponding current (right) for I_{beq}-levels of roughly 10 and 1 uA. The lower current level gives the greater voltage gain. W/L=20.8um/1.2um for both PMOS and NMOS.

$$I_{ds,p} = I_{beq}exp\{(\frac{2}{5nU_t}(\frac{V_{dd}}{2} - V_w)) + (\frac{1}{5nU_t}(\frac{V_{dd}}{2} - V_x)) + (\frac{1}{5nU_t}(\frac{V_{dd}}{2} - V_y)) + (\frac{1}{5nU_t}(\frac{V_{dd}}{2} - V_z))\} \tag{3}$$

$$I_{ds,n} = I_{beq}exp\{(\frac{2}{5nU_t}(V_w - \frac{V_{dd}}{2})) + (\frac{1}{5nU_t}(V_x - \frac{V_{dd}}{2})) + (\frac{1}{5nU_t}(V_y - \frac{V_{dd}}{2})) + (\frac{1}{5nU_t}(V_z - \frac{V_{dd}}{2}))\} \tag{4}$$

$$I_{ds,p} = I_{beq}exp\{\frac{1}{nU_t}(\frac{V_{dd}}{2} - \frac{2}{5}V_w - \frac{1}{5}V_x - \frac{1}{5}V_y - \frac{1}{5}V_z)\} \tag{5}$$

$$I_{ds,n} = I_{beq}exp\{\frac{1}{nU_t}(\frac{2}{5}V_w + \frac{1}{5}V_x + \frac{1}{5}V_y + \frac{1}{5}V_z - \frac{V_{dd}}{2})\} \tag{6}$$

In the case that all inputs are $V_{dd}/2$:

$$I_{ds,p} = I_{beq}exp\{\frac{1}{nU_t}(\frac{V_{dd}}{2} - \frac{5}{5}\frac{V_{dd}}{2})\} \tag{7}$$

$$I_{ds,n} = I_{beq}exp\{\frac{1}{nU_t}(\frac{5}{5}\frac{V_{dd}}{2} - \frac{V_{dd}}{2})\} \tag{8}$$

We then get $I_{ds,p} = I_{ds,n} = I_{beq}$ When we let inputs W,X,Y,Z be either at 0 or V_{dd} we get the exponents, e_p and e_n, in figure 3 for the PMOS and NMOS, respectively. If $e_p > e_n$ the output should normally move towards "1" / V_{dd}. Likewise, if $e_p < e_n$ the output goes to "0", or 0 volts. From the table in figure 3 we see that if input w is held at "0" the circuit produces the NAND function of inputs X,Y,Z. In the case where w is "1" the circuit implements the NOR function of inputs X,Y,Z. Keeping all inputs at the same potential makes the structure implement an INVERTER function, as demonstrated by measurements, on a chip in AMS 0.6 um CMOS technology, in figure 2. For the table in figure 4, $w = V_{dd}/2$, and the circuit implements the CARRY' function of a FULL-ADDER, or $C' = [xy + xz + yz]'$ This circuit containing 2 MOSFETs can realize CARRY', 3-NOR, 2-NOR, 3-NAND, 2-NAND and INVERT. Adding one of our circuits used as an inverter would give CARRY, AND, OR and BUFFER, in addition to the functions mentioned. A floating-gate circuit able to implement CARRY, AND and OR, using 5 MOSFETs and operating from +5 V supply voltage, is reported in [12]. Our circuit has been simulated, based on a netlist extracted

from the layout in figure 6, and results can be seen in figure 7. It works for a V_{dd} of 200 mV. Measurements on an inverter implemented in AMS 0.6 CMOS has demonstrated it working between 93 and 800 mV [8].

W	X	Y	Z	e_p	e_n	OUT
0	0	0	0	$5V_{dd}/10$	$-5V_{dd}/10$	1
0	0	0	1	$3V_{dd}/10$	$-3V_{dd}/10$	1
0	0	1	0	$3V_{dd}/10$	$-3V_{dd}/10$	1
0	0	1	1	$V_{dd}/10$	$-V_{dd}/10$	1
0	1	0	0	$3V_{dd}/10$	$-3V_{dd}/10$	1
0	1	0	1	$V_{dd}/10$	$-1V_{dd}/10$	1
0	1	1	0	$V_{dd}/10$	$-1V_{dd}/10$	1
0	1	1	1	$-V_{dd}/10$	$V_{dd}/10$	0
1	0	0	0	$V_{dd}/10$	$-V_{dd}/10$	1
1	0	0	1	$-V_{dd}/10$	$V_{dd}/10$	0
1	0	1	0	$-V_{dd}/10$	$V_{dd}/10$	0
1	0	1	1	$-3V_{dd}/10$	$3V_{dd}/10$	0
1	1	0	0	$-V_{dd}/10$	$V_{dd}/10$	0
1	1	0	1	$-3V_{dd}/10$	$3V_{dd}/10$	0
1	1	1	0	$-3V_{dd}/10$	$3V_{dd}/10$	0
1	1	1	1	$-5V_{dd}/10$	$5V_{dd}/10$	0

Fig. 3. The table shows the exponentials, e_p, e_n, and output values, for all possible binary values of inputs W,X,Y,Z.

X	Y	Z	e_p	e_n	OUT
0	0	0	$3V_{dd}/10$	$-3V_{dd}/10$	1
0	0	1	$V_{dd}/10$	$-V_{dd}/10$	1
0	1	0	$V_{dd}/10$	$-V_{dd}/10$	1
0	1	1	$-V_{dd}/10$	$V_{dd}/10$	0
1	0	0	$V_{dd}/10$	$-V_{dd}/10$	1
1	0	1	$-V_{dd}/10$	$V_{dd}/10$	0
1	1	0	$-V_{dd}/10$	$V_{dd}/10$	0
1	1	1	$-3V_{dd}/10$	$3V_{dd}/10$	0

Fig. 4. The table shows the exponentials, e_p, e_n, and output values, for all possible binary values of inputs X,Y,Z when W=$V_{dd}/2$. "OUT" provides the CARRY' function for a FULL-ADDER.

3.2. Two universal elements give both CARRY' / SUM' for FULL-ADDER or NAND / XNOR or NOR / XOR

If we use two instances of the proposed circuit connected like in figure 5, it can implement a wider range of Boolean functions due to its inherent reconfigurability. If we use the 1st of these two blocks to implement the thruth table in figure 4, then "OUT" from this table equals the "W" input to the 2nd block, according to figure 5, while inputs X, Y and Z are common for the two blocks in the schematic. If we consider the 2nd block in the schematic it "uses" a subset including lines 4,6,7,8,9,10,11 and 13 out of the 16 lines containing binary information from figure 3. These lines from the truth table corresponds to the SUM' for a FULL-ADDER, or $S' = [z \oplus (x \oplus y)]' = [xy'z' + x'yz' + xyz + x'y'z]'$,which equals 3-XNOR. Keeping one of the inputs at "0" or "1" makes the same node implement 2-XNOR or 2-XOR, respectively. A circuit able to realize SUM' and CARRY' for 1-bit addition in

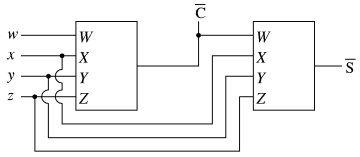

Fig. 5. Schematic using two of our universal circuits to implement CARRY' and SUM' for FULL-ADDER.

this way is also able to generate NAND and XNOR at the same time, or NOR and XOR simultaneously.

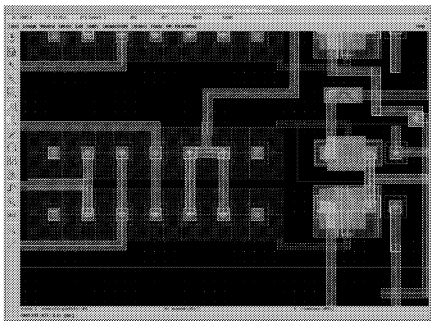

Fig. 6. Layout for the universal element. The area equals approximately 36um x 13um. Removing passivation at certain areas and making metal shields make UV-light hit selected parts of the chip surface, while UV-programming after fabrication of chips. UV-programming might be repeated or changed.

3.3. Other aspects of the FGUVMOS approach

Different basic circuit building blocks might need individual access to power rails and wells under UV-programming and test [7]. This makes circuits more expensive in terms of chip area, pads, programming and test equipment. We hope that being able to use the same basic universal element for implementimg a variety of functions can reduce these costs and make it easier to build more complex systems in silicon than reported until now. Compared to the FGUVMOS FULL-ADDER approach in [6] our new element reduce the number of MOSFETs for implementing the CARRY' and SUM' functions from 22 to 2, and 16 to 4, respectively. The number of capacitors are reduced from 38 to 16 at the same time, saving area and complexity.

4. CONCLUSION

We have shown a simple analysis of a 2-MOSFET and 8-capacitor reconfigurable digital FGUVMOS building block which in principle can be used to implement any Boolean function. Examples demonstrate significant savings in circuit complexity compared to earlier reported FGUVMOS circuits as well as more conventional CMOS implementations. Basic Boolean functions are demonstrated by layout simulations as well as measurements on an inverter programmed for different current levels. The proposed circuit might help reduce chip area for some functions, as well as complexity regarding UV-programming and laboratory test. We have made working FGUVMOS circuitry earlier, and hopefully a new chip submitted through EuroPractice in May 2001 will provide further insight in how these principles can work out.

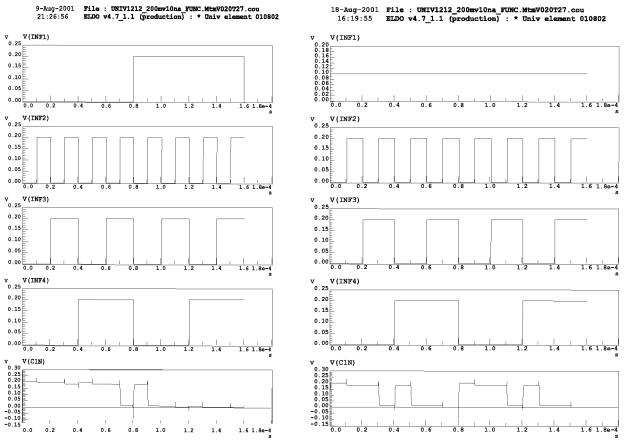

Fig. 7. Simulating our circuit for NAND, NOR and CARRY' functions. From the start of the left simulation trace V(INF1) / W="0", making it realize the NAND function of the 3 inputs INF2, INF3, INF4. Around midway through the V(INF1) goes to "1", making it implement a NOR function. CARRY' is shown to the right. On the bottom: OUT="V(c1n)". I_{beq}=10 nA, temperature 27 degrees C, and V_{dd}=200 mV. Layout-based simulations have demonstrated the circuit working at $V_{dd} = 180mV$.

References

[1] P. Hasler, T. S. Lande *Overview of Floating-Gate Devices, Circuits and Systems* IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing, january 2001.

[2] Y. Berg, T. S. Lande *Area Efficient Circuit Tuning with Floating-Gate Techniques*, Proc. of the IEEE Int. Symp. on Circuits and Systems,1999.

[3] T. Shibata, T. Ohmi *An Intelligent MOS Transistor Featuring Gate-Level Weighted Sum and Threshold Operations*, Technical Digest, International Electron Devices Meeting, 1991.

[4] K. Kotani, T. Shibata, T. Ohmi *Neuron-MOS Binary-Logic Circuits Featuring Dramatic Reduction in Transistor Count and Interconnections* Technical Digest, Int. Electron Devices Meeting 13-16. Dec., 1992.

[5] D. Abbott, S. F. Al-Sarawi, B. Gonzalez, J. Lopez, J. Austin-Crowe, K. Eshraghian *Neu-MOS for Smart Sensors and Extension to a Novel Neu-GaAs Paradigm* IEEE Int. Conf. on Electronics, Circuits and Systems,7-10 September, 1998.

[6] Y. Berg, D. T. Wisland,T. S. Lande *Ultra Low-Voltage/Low-Power Digital Floating-Gate Circuits*, IEEE Transactions on Circuits and Systems II, July 1999.

[7] R. Bahr *Design of Linear Four Quadrant Analog Multipliers using Floating-Gate Transistors*, Thesis for the Cand. Scient. degree, Department of informatics, University of Oslo, May 2001.

[8] S. Aunet, Y. Berg, O. Tjore, Ø. Næss, T. Sæther *Four-MOSFET Floating-Gate UV-Programmable Elements for Multifunction Binary Logic*, Proceedings of the 5th World Multiconference on Systemics, Cybernetics and Informatics (SCI 2001), Orlando, FL, USA, July 22-25, 2001.

[9] S. Aunet, Y. Berg, Ø. Næss, T. Sæther *Novel Reconfigurable Two-MOSFET UV-programmable Floating-Gate Circuits for CARRY' NAND, NOR or INVERT Functions*, Accepted for publication in Proceedings of the 8th International IEEE Conference on Electronics, Circuits, and Systems, Malta, 2-5 Sept. 2001.

[10] S. Aunet, Y. Berg, T. Ytterdal, Ø. Næss, T. Sæther *A Method for Simulation of Floating-Gate UV-Programmable Circuits with Application to Three New Digital Circuits*, Accepted for publication in Proceedings of the 8th International IEEE Conference on Electronics, Circuits, and Systems, Malta, 2-5 Sept. 2001.

[11] Austria Mikro Systeme International AG *0.6 um CMOS CUP Process Parameters* Document no. 9933011, Rev. B, Oct. 1998.

[12] J. Ramirez Angulo, G. Gonzalez-Altamirano *A New Programmable Logic Family Using Multiple-input Floating-Gate Transistors* Proc. of the 40th Midwest Symp. on Circuits and Systems, 1997.

5) A New Universal UV-Programmable Floating-Gate Digital Element With Application to an 8-Transistor Full-Adder and a D-Latch

A NEW UNIVERSAL UV-PROGRAMMABLE FLOATING-GATE DIGITAL ELEMENT WITH APPLICATION TO AN 8-TRANSISTOR FULL-ADDER AND A D-LATCH

Snorre Aunet, Yngvar Berg*, Øivind Næss*, Trond Sæther

Department of physical electronics, Norwegian University of Science and Technology, O.S. Bragstads plass 2A, N-7034 Trondheim, Norway. Telephone: +47 73 59 44 00 Fax: +47 73 59 14 41 E-mail: Snorre.Aunet@fysel.ntnu.no, Trond.Saether@fysel.ntnu.no

*Department of informatics, University of Oslo, Gaustadaleen 23, N-0316 Oslo, Norway. Telephone: +47 22 85 24 10 Fax: +47 22 85 24 01 E-mail: yngvarb@ifi.uio.no, onass@ifi.uio.no

ABSTRACT

We present a universal digital UV-programmable floating-gate circuit able to work with a supply voltage down to less than 200 mV. The two-transistor gate as a stand-alone circuit can implement functions like CARRY' for full-adder, NOR3, NOR2, NAND3, NAND2 and INVERT. The element can be programmed for a wide range of current levels. By choosing to use one of the inputs as some sort of control input, the circuit can change it's Boolean function in real time. Preliminary measurements and simulations are shown, suggesting that it might be interesting to carry on furter research regarding ultra low-power potential for such circuits, due to promising PDP numbers.

1. INTRODUCTION

FLOATING-GATE devices were firstly reported in 1967, originally for use as digital storage elements [1]. During about the last decade floating-gate devices have been used as analog memory elements, and as part of capacitive-based circuits and adaptive circuit elements [1]. To adjust the level of charge on floating-gates ,in standard CMOS technologies, mostly Fowler-Nordheim tunneling in combination with hot-electron injection is used, or short-wave (UV-C) UV-light ("FGUVMOS")[1],[2]. Using UV-light and the FGUVMOS approach [3] enables an adjustment of current levels / effective threshold voltages seen from driving nodes, over orders of magnitude [2],[3]. Operating circuitry in subthreshold, with optimized effective threshold voltages in combination with power supply voltages typically in the 0.3 - 0.8 V area might allow circuits using relatively little energy per operation. Another approach to low-power implementationsations are Neuron MOSFET circuits, said to be inspired by the biological neuron, originally developed in 1991 [4]. Neuron MOS circuits might reduce both wiring and transistor count compared to conventional methods drastically [4],[5],[6],[7]. In this paper we try to combine the FGUVMOS and neuron MOS approaches to make simple ultra low-voltage/low-power digital circuits. Earlier FGUVMOS approaches have always relied on different basic building blocks for different digital functions [2],[8],[9]. Here we present a ripple-carry adder and a D-latch using only one reconfigurable basic cell, used for implementing different Boolean functions like for example NAND, NOR,

CARRY' for FULL-ADDER, or INVERT. This probably makes it far easier to match current levels and reduce circuit complexity.

2. FGUVMOS CIRCUITS

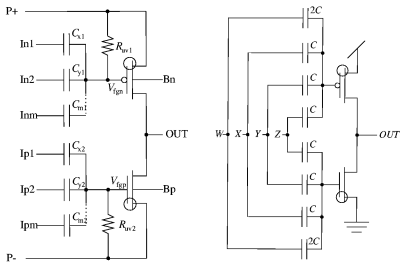

Fig. 1. FGUVMOS in reverse biased programming mode is depicted to the left. Then the circuit is exposed to UV-light, and the effective threshold voltages permanently altered when circuits afterwards are operated normally. Our universal, reconfigurable element is shown to the right.

FGUVMOS circuits are here briefly treated from the subthreshold point of view, though many FGUVMOS circuits can be used above threshold as well. FGUVMOS circuit elements reported until now have one PMOS- stacked on top of one NMOS-transistor, and one or more capacitively coupled connections to the floating-gates, as in figure 4. We can use the following equations to describe the subthreshold currents of the FGUVMOS transistors:

$$I_{ds,p} = I_{beq} \prod_{i=1}^{m} exp\{\frac{1}{nU_t}(V_{dd}/2 - V_i)k_i\} \tag{1}$$

$$I_{ds,n} = I_{beq} \prod_{i=1}^{m} exp\{\frac{1}{nU_t}(V_i - V_{dd}/2)k_i\} \tag{2}$$

Here, $k_i = C_i/C_{tot}$ is the capacitive division factor of the ith input capacitor, C_i, and C_{tot} is the total capacitance seen from the floating-gate. I_{beq} is the balanced equilibrium current, which is the drain current of the transistors when all ordinary input signals and driven nodes are equal to $V_{dd}/2$. I_{beq} is a function of the effective threshold voltages seen from the driving nodes, and measurements indicate typically current levels in the nA to uA area, for implementations in the AMS 0.6 CUP process [11]. In figure 2 input and output voltages as well as currents through the circuit are shown for two different current levels / pair of threshold voltages.

3. REDUCING NUMBER OF DIVERSE ELEMENTS TO SIMPLIFY PROGRAMMING AND TEST

The supply rails and wells are used to provide the desired current levels under programming mode, when the chip is exposed to UV-light. All transistors on chip, or even a wafer, may be programmed simultaneously without additional programming circuitry [12]. Different FGUVMOS circuits with different capacitances seen from floating-gates might use different programming voltages on their power-rails, if their current levels are to be matched,

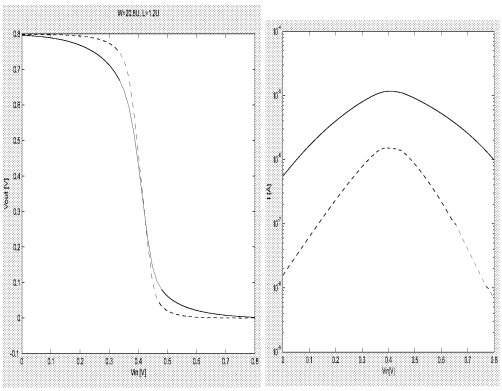

Fig. 2. Measured characteristics for the same FGUVMOS inverter. W/L=1.2um/0.6 um for both transistors. The highest current level (right) corresponds to the lowest voltage gain (left). The two I_{beq} currents lie roughly at the 0.1uA and uA levels.

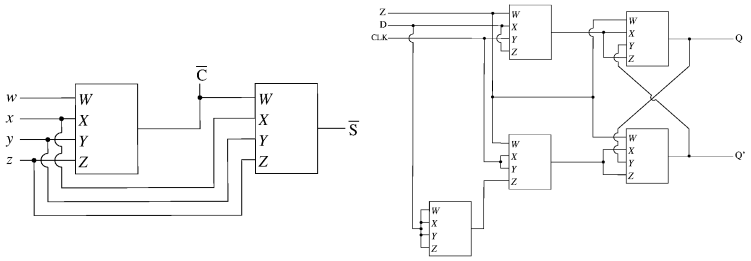

Fig. 3. Schematics for the part of the FULL-ADDER producing CARRY' and SUM', and the D-latch.

especially in subthreshold with several exponential characteristics [13]. In this paper we use the same circuit element to implement every digital function, reducing wiring and number of I/O-pads needed to access power rails and wells for diverse basic circuit elements. This hopefully can save library and production costs as well as simplify programming and test.

4. FULL-ADDER

A 4-MOSFET circuit, shown to the left in figure 3, made from two of our universal elements, can produce CARRY' and SUM' for a FULL-ADDER. Adding two inverters, using the universal element, gives a complete 8-MOSFET FULL-ADDER. 25 different FULL-ADDER structures are evaluated based on HSPICE simulations on layout in a 0.6 CMOS technology in [14]. They all contain from 14 to 20 MOSFETs, while many CMOS implementations use 28 MOSFETs. A netlist extracted from layout, shown in figure 4, of a 2-bit ripple-carry adder were used for simulation of the FULL-ADDER. To the left in figure 5 the FULL-ADDER SUM and CARRY are shown as functions of logic inputs INF2, INF3, INF4. Some simulations regarding power consumption and speed of the circuit were done, and the results are shown in figure 6, for Vdd of 200 and 800 mV, and different equilibrium, I_{beq}, currents. t_r and t_f are worst case delays for S1 ("SUM1") rising and falling, respectively. P is the average power [W] when the circuit are operating at maximum frequency, for the input vectors in figure 5. $(t_r + t_f) * P$ were used for finding "PDP" / "Power-Delay-Product". The best PDP in [14] was 0.5 pJ, roughly 100 times worse than our circuit with a Vdd of 200 mV. Though the numbers might not be comparable due to slightly different simulations and technologies, as well as lack of accurate models in subthreshold [15], the circuits might seem

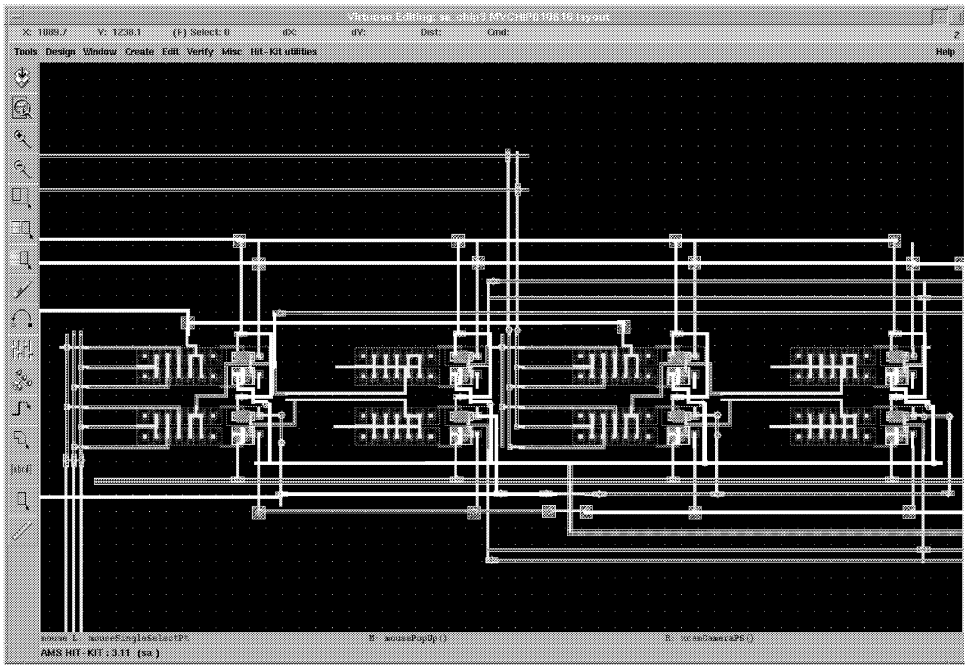

Fig. 4. Layout for 2-bit ripple-carry adder containing eight universal elements. All W/L=1.2u/1.2u.

promising from a low-power point of view. The results indicate that lowering the threshold voltage betters the PDP numbers, which are roughly constant for a given supply voltage, at least for our 200 mV simulations. For "EDP" we multiplied our PDP values with $(t_r + t_f)$. For 200 mV and 1 uA equilibrium current the circuit did no longer work properly, probably because transistors left subthreshold.

5. D-LATCH

The D-latch resembles the one in [16], and is depicted to the right in figure 3. In this circuit our universal element is used 4 times as a 2-input NAND-gate, and once as an inverter. The input to the capacitors of twice the size of the others ("Z") is grounded, while two of the other inputs are wired together under ordinary operation, for the circuits used as 2-input NAND. The universal element used as an inverter has all its inputs wired together. A simulation trace is shown to the right in figure 5. The characteristic equation is Q(t+1)=D, and input D is sampled during occurence of a clock pulse. Here we used a V_{dd} of 400 mV, which may have been lower. In [8] the universal circuit were used as part of a 4-MOSFET circuit, working with a V_{dd} down to around 180 mV, according to simulation. Measurements demonstrated a single inverter working with V_{dd} from 93 mV to 800 mV [10].

6. CONCLUSION

We have introduced a new, universal, reconfigurable, digital circuit element able to produce 3-input NAND, 2-input NAND, 3-input NOR, 2-input NOR, INVERT and CARRY' for FULL-ADDER, used as a stand-alone circuit. We have also shown examples on how it can be used to implement an 8-MOSFET FULL-ADDER and a D-LATCH, which are

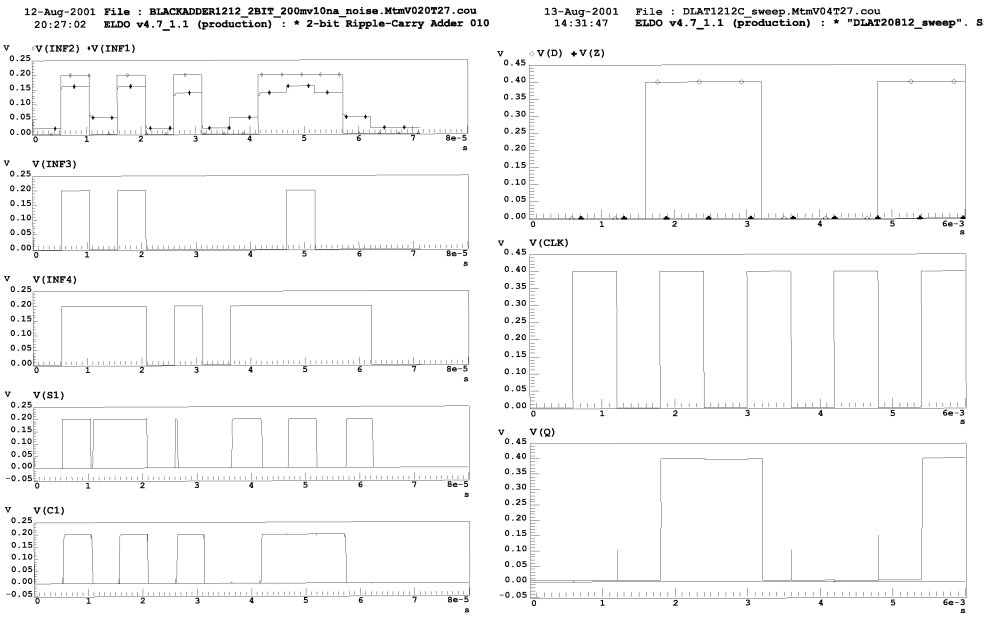

Fig. 5. FULL-ADDER simulation results are shown to the left, for V_{dd}=200 mV and I_{beq}=10nA. The D-latch is simulated in the right half of the figure. Inputs "Z", is held at "0" level to make the universal reconfigurable elements behave as NAND-gates, except for the circuit acting as an inverter.

V_{dd} [mV]	I_{beq}[nA]	t_r[s]	t_f[s]	F_{max}[Hz]	P[W]	PDP[Ws]	EDP[Wss]
200	1.0	4.1u	4.9u	111k	5.2e-10	0.005	4.5e-20
200	10	0.5u	0.5u	1M	4.45e-9	0.005	5e-21
200	100	73n	73n	6.8M	4.2e-8	0.006	8.8e-22
200	1000	-	-	-	7.3e-7	-	-
800	1.0	2.2u	2.4u	217k	5.95e-9	0.027	1.8e-19
800	10	346n	297n	1.01M	3.9e-8	0.039	1.6e-20
800	100	74n	53n	7.9M	2e-7	0.025	3.2e-21
800	1000	18n	13n	32M	9.6e-7	0.030	9.0e-22

Fig. 6. Some simulation results for the FULL-ADDER (1-bit adder).

basic building blocks in many traditional digital systems. The circuit element might operate with supply voltages below 200 mV, and probably with relatively low energy per operation compared to many other methods of implementation. Our universal element might also ease matching of current levels and reduce design and test costs. Preliminary measurements have demonstrated that an implemented FGUVMOS inverter works, giving us important PMOS and NMOS building blocks for future implementations.

References

[1] P. Hasler, T. S. Lande *Overview of Floating-Gate Devices, Circuits and Systems* IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing, Special issue on floating-gate devices, circuits and systems, january 2001.

[2] Y. Berg, D. T. Wisland,T. S. Lande *Ultra Low-Voltage/Low-Power Digital Floating-Gate Circuits*, IEEE Transactions on Circuits and Systems II, July 1999.

[3] Y. Berg, T. S. Lande *Area Efficient Circuit Tuning with Floating-Gate Techniques*, Proc. of the IEEE

Int. Symp. on Circuits and Systems,1999.

[4] T. Shibata, T. Ohmi *An Intelligent MOS Transistor Featuring Gate-Level Weighted Sum and Threshold Operations*, Technical Digest, International Electron Devices Meeting, 1991.

[5] K. Kotani, T. Shibata, T. Ohmi *Neuron-MOS Binary-Logic Circuits Featuring Dramatic Reduction in Transistor Count and Interconnections* Technical Digest, International Electron Devices Meeting 13-16. December, 1992.

[6] A. Rantala, S. Franssila, K. Kaski, M. Åberg, P.Kuivalainen *Improved neuron MOS-transistor structures for integrated neural network circuits*, IEE Proceedings- Circuits, Devices and Systems, February 2001.

[7] D. Abbott, S. F. Al-Sarawi, B. Gonzalez, J. Lopez, J. Austin-Crowe, K. Eshraghian *Neu-MOS for Smart Sensors and Extension to a Novel Neu-GaAs Paradigm* IEEE Int. Conf. on Electronics, Circuits and Systems,7-10 September, 1998.

[8] S. Aunet, Y. Berg, O. Tjore, Ø. Næss, T. Sæther *Four-MOSFET Floating-Gate UV-Programmable Elements for Multifunction Binary Logic*, Proceedings of the 5th World Multiconference on Systemics, Cybernetics and Informatics (SCI 2001), Orlando, FL, USA, July 22-25, 2001.

[9] S. Aunet, Y. Berg, T. Ytterdal, Ø. Næss, T. Sæther *A Method for Simulation of Floating-Gate UV-Programmable Circuits with Application to Three New Digital Circuits*, Accepted for publication in Proceedings of the 8th International IEEE Conference on Electronics, Circuits, and Systems, (ICECS 2001), Malta, 2- 5 September 2001.

[10] S. Aunet, Y. Berg, Ø. Næss, T. Sæther *Novel Reconfigurable Two-MOSFET UV-programmable Floating-Gate Circuits for CARRY', NAND, NOR or INVERT functions*, Accepted for publication in Proceedings of the 8th International IEEE Conference on Electronics, Circuits, and Systems, (ICECS 2001), Malta, 2- 5 September 2001.

[11] Austria Mikro Systeme International AG *0.6 um CMOS CUP Process Parameters* Document no. 9933011, Rev. B, Oct. 1998.

[12] Y. Berg, D. T. Wisland,T. S. Lande, S. mikkelsen *Ultra Low-Voltage Digital Floating-Gate UVMOS (FGUVMOS) Circuits*, Proceedings of the 1998 IEEE International Symposium on Circuits and Systems, 1998.

[13] R. Bahr *Design of Linear Four Quadrant Analog Multipliers using Floating-Gate Transistors*, Thesis for the Cand. Scient. degree, Department of informatics, Faculty of Mathematics and Natural Sciences, University of Oslo, May 2001, pp 59.

[14] A. M. Shams, M. A. Bayoumi *A framework for fair performance evaluation of 1-bit full-adder cells*, Proceedings of the 42nd Midwest Symposium on Circuits and Systems, 2000.

[15] A. Low, P. Hasler *Cadence-based simulation of floating-gate circuits using the EKV-model*, Proceedings of the 42nd Midwest Symposium on Circuits and Systems, 2000.

[16] M. Morris Mano *Digital Design* Prentice/Hall International, Inc., ISBN0-13-212325-8, 1984, pp 207.