,

UNIVERSITY OF OSLO
Department of Informatics

Multiple-valued semi-floating-gate parallel and serial multiplier

Cand. Scient. Thesis

Olav Stanly Kyrvestad

May 2007

Acknowledgments

I would like to thank Professor Yngvar Berg for being my supervisor through the work with this project. Furthermore I would like to thank the indians Claus Limbodalen and Kjetil Meisal for cooperation on chip and PCB design. I would also like to thank the rest of the staff at the Nanoelectronics group at the Department of Informatics at the University of Oslo.

This thesis is a result of a few late nights with fellow students in the laboratories at the Nanoelectronics group. I would like to thank all who have made these late nights fun nights.

Abstract

As shown in this project the thought of using digital circuits for more than representing two values is almost as old as the semiconductor. In this thesis some of the history of multiple-valued circuits is described. One of the main aims of representing more than two values in a system is to save energy. Several methods of power saving are mentioned, such as reduction of Vdd, reduction in swing of signals and power down to mention a few. The main scope as the title implies is on semi-floating-gate circuits. Also a few circuits are presented to explain the technology utilized in the practical part of the project. Finally the use of multiple-valued semi-floating-gate circuits as parallel and serial multipliers is presented.

Contents

Chapter 1

Introduction

My assignment has it's basis from a number of multiple-valued semi-floating-gate circuits[1, 2]. These are all based on the semi-floating-gate multiple-valued semi-floating-gate inverter. I have utilized these circuits in an number of ways in my design. It has been used as an inverter, a NAND-gate and as an adder. Trough this thesis I will introduce the multiple-valued semi-floating-gate multiplier in two main forms, as a parallel multiplier and as a serial multiplier. Today the VLSI technology has reached a line where interconnections rather than devices dominate the use of area, propagation delay and dynamic power dissipation of a chip. A promising approach to solving this problem is the use of multiple-valued logic[3].

Chapter two deals with the preliminary theories of multiple-valued logic and circuits already introduced in earlier articles.

Chapter three contains a description of the serial and parallel multiple-valued multiplier I have implemented.

Chapter four contains the experiments conducted for this thesis.

Chapter five contains the conclusion and suggestions for further work.

Chapter 2

Multiple-valued semi-floating-gate theory and circuits

Multiple-valued CMOS logic is the study of CMOS logic not restricted to the normal two values of binary logic, true and false but replaced by finite or infinitely number of values. It was first invented in 1967 as a digital storage element[4]. Multiple-valued logic circuits has some critical features related to reduction of the number of the interconnections and the increased information content per unit area[5], multiple-valued logic and signals also offers an opportunity to establish an economic balance between the quantized integrity of binary systems and the information density of analog systems[6]. Voltage mode multiple-valued logic circuits is a promising way to solve problems related to overheating and power consumption and transmission lines in digital logic. Development of these techniques is important for battery driven applications as these circuits has good frequency versus power efficiency[7, 8, 9, 10]. On the other side multiple-valued systems require higher noise margins since logic levels is closer than in binary logic circuits with the same supply voltage becomes close to one another[11, 12].

15 years ago the question regarding multiple-valued circuits was fundamental; Is it possible to design multiple-valued integrated circuits? Today the obvious answer is yes. During this time several proposed circuits has been presented. At the same time the digital, or two valued technology has also improved[13]. Micro integrated circuits can have benefits from multiple-valued signals, especially at data communication level because of the reduced interconnections, reduced area consumption and reduced crosstalk. A multiple-level approach is a promising way to solve these problems[14]. Floating-gate systems resembling the biological neuron has

been proposed, making possible circuits featuring a factor five to ten reduction in gate count using standard CMOS technologies[15].

In this project a variant of the floating-gate is used, the semi-floating-gate. By recharging the semi-floating-gates periodically we not only avoid the problems linked to programming or initializing of the floating-gates, but we convert the non-volatile floating-gate to a semi-floating-gate. Thus the control of the actual gate charge in term of predictable long term charge restoration becomes easier. The semi-floating-gate is not influenced by a random floating-gate charge distortion due to a periodic or frequent charge restoration or reset. The recharging of the semi-floating-gate is accomplished by local recharge transistors temporarily connection the output to the floating-gate of a gate. All floating-gate circuits need to be initialized either once initially or frequently. The once and for all initialization is synonymous with programming. By recharging the semi-floating-gate frequently we avoid problems with any leakage currents and random or undesired disturbance of the floating-gate charges[16].

2.1 Multiple-valued circuits on the raise, or still in the dark

Through the ages several number systems have existed. But with the growth of Arabic mathematics around year 1000AD we have used the Arabic number system with ten digits. As often the military needs led to a change in this, the need to fast calculate new trajectories for new artillery cannons. This led to the development of artificial methods of calculation during the second world war. We got our first real computers. Still today the largest number crunching machines remains in the service of military powers around the globe, and they all use two numbers, true and false. Still, in these 60 years of largely boolean dominance it is possible to find some small drips of other number systems introduced in literature. But still there have been no decisive breakthrough for multiple-valued systems. Although they have always existed, artificial multiple-valued systems, in the dark corners of mechanical and later electronic computation, ever since the dawn of artificial computational systems. The use of vacuum tubes and later the development of the transistor made Boolean mathematics the natural choice for expressing how computers should work. But the existence of Post algebra [17, 18] means there should be no theoretical limitations to develop multiple-valued systems. 10-valued systems has been proposed for ease of adoption to our calculation system, but with 4-valued memory as one of only a few commercially available products[19, 20, 21], and then only because of high performance with respect to area consumption, multiple-valued systems have little to no place in todays commercial

electronic industry. There are other uses of multiple-valued systems including controlling large electrical motors. Here multiple-valued logic lets one operate the motor more efficient by modulating the sinus wave using several values instead of two[22]. Multiple-valued techniques have not reached a point where it can be described as a common technique for VLSI-systems.

Only when some technology is reaching some limit has multiple-valued circuits really come to life. And such a period may be around the corner. With the transistor reaching the nano scales several predicted unsolved problems arise. Undesired leakages and gate currents will become larger due to the short tunnel and the very thin isolation under gates. It is predicted that leakage will result in equal power consumption for a transistor in a sub threshold state as in dynamic state. Already one has to use low leakage versions of transistors in digital design to avoid leakage in the transistors, that otherwise would effect the reliability of the circuits designed. Production techniques, the optical lithography especially is reaching its limit and will be hard to improve since the size of structures will be smaller than the wavelength of the light utilized to cut structures into the production masks used. These wavelengths has been used for several years, introducing new wavelengths is a much more cumbersome affair than just making the existing equipment more accurate. It will no longer be practical to use silicon implementations for debugging a system, mainly the time delay but also the cost will be much greater due to production difficulties. Midlife resizing of designs will become harder as the process is not scaled equally throughout the circuit. Problems will also arise with the introduction of new materials introduced in production processes. For instance copper is a very soft metal and is in danger of erosion and out gassing. A circuit is therefore at risk of changing properties during its lifespan. There is a need to reduce the number of transmission lines. These lines are taking up more and more of the chip area. This problem has been sought solved by increasing the number of metal layers. But this solution gives the rise to increased possibilities for unwanted crosstalk. Another problem with transmitter lines is that they will be the main contributor to delay in a circuit. Another source of concern is the contacts used for connecting different layers of a chip. These will be increasingly more difficult to reduce in size, and as they get smaller they will get an increased resistance[23].

The dynamic power consumption of binary circuits is a continuous problem. Power lines and cooling has to be dimensioned to account for a very large power consumption. By utilizing voltage mode multiple-valued logic one can reduce this need for over dimensioning significantly. Binary systems are becoming more and more area inefficient. With the development of sub micron transistors this problem has really come to the surface. With wires so thin that they will have to be made wider to be good trans-

mission lines the area reduction effect of new technology is significantly reduced. The introduction of copper as material in wires has postponed the problem[13, 21]. Another problem arising from the sub micron processes is the limited edge space available for chip communication. The space of the edge connection grows only linear, while the available space for circuit logic grows by n^2. One way of solving this could be to transport more information on each connection, by a multiple-valued signal[6]. But high speed serial links are easier to implement than multiple-valued circuits, to reducing the number of off chip connections.

One could argue that instead of using simple power-delay product as a way of describing how well a chip performs, the performance of a logic integrated circuit should be evaluated by how much functionality that can be created on a given chip-area related to the power consumption.

Neuron-MOS circuits has also been reported having a great potential in reconfigurable circuits[24]. One interesting aspect of multiple-valued systems, especially when looking at neuron-MOS systems is the possibility to replicate natural neuron systems. With its parallel computing the human brain is far more efficient on performing many tasks which todays computers need huge amount of computational power to achieve. This is particular apparent in tasks involving real time calculations. If we were able to make systems do calculations and memory lookups as the human brain does, we could perhaps find ways to make electronic systems interpret images, sounds, smells and such as good as we do ourselves. By implementing parallel algorithms there could be a real breakthrough in such systems[25]. With multiple-valued systems one can also get a significant increase of some variable functions. Particularly Max and Min Functions[19].

All the problems that need to be solved means that there is a need for design improvements to continuously increase the efficiency of circuits. This is where multiple-valued circuits in general, and voltage mode in particular may have their rightful place. Multiple-valued circuits offers some of the signal integrity of binary circuits, as well as some of the information density of analog circuits. Multiple-valued logic systems also has the advantage of not being noisy. They can therefore operate well in close proximity of pure analog circuitry.

With the techniques used in this thesis several of the above mentioned problems may find its solution. By sending a multiple-valued signal through one transmission line, more data can be transferred and thus reducing the number of lines. The communication bottleneck is particularly serious between memory and logic. Also there is a potential of reduced area consumption both as transmission lines are reduced and more direct as a result of using simpler circuits for certain operations. Also reduced need for carry propagation is possible. Multiple-valued circuits generally implement post

algebra in contrast to two value circuits which implement boolean algebra. A correspondence is established between many different voltage or current levels and the multiple different values of the algebra. We can deduce the following consequence from this fact. As the set of multiple different voltages is totally ordered, the set of multiple different values 0 to m-1 is totally ordered, as follows: $0 < 1 < 2 < ... < m - 1$. Thus, the algebra corresponding to multiple-valued circuits is the Post algebra[13]. Interconnection capacitances will especially play a very important role in deep sub micro meter implementations since the fringing capacitances of the interconnect capacitance can become a dominant part of the total capacitance and cease to scale.[26].

2.2 The need for power saving

Much research has been put into increasing the speed of digital systems. This has lead to increased speed, but also to new challenges. One of these problems is the power consumption. A CMOS digital circuits has three major sources of power dissipation. The switching component, direct-path short circuit current and leakage current. The major component here is the switching component. This makes this contribution the one which is the most efficient to reduce and thereby the one needed to be reduced. One good way of reducing the need for current for switching is to use a floating-gate implementation of the inverter. The need for power saving is driven mainly by two different needs. One, the desire to have large computational power available in battery driven applications. But even more important is the second need, at the high end of computation, the need to keep increasingly dense circuits at a temperature where they can operate without needing cooling systems of unmanageable proportions. Also the introduction of new materials with better conducting abilities in regard to signals but worse abilities in regard to transporting heat plays a role here.[23]. In the world of computing today we surround ourselves with high end computing power used for graphic presentation and other applications with large need of calculations. Another aspect of power consumption and computation in the future is the demand for applications with need for real time computation. This is systems such as sound and video compression. These systems will normally work at or near the peak of the computational power of the systems, so there is a real need for using existing techniques of power saving and introducing new ones[26]. As the transistors shrink and the supply voltage continue to be reduced, the available headroom is relatively reduced as the threshold voltage does not reduce itself at the same rate as the supply voltage[27]. Low power design techniques aim to reduce power dissipation in high performance systems and power consumption

in portable equipment. Power dissipation affects packaging, reliability and heat removal costs. Power consumption relate directly to size and battery life. For scaled down voltages in deep sub micron technologies phenomena such as gate oxide tunneling and gate induced drain leakage are likely to become important. The Vdd scaling has mainly been driven by concerns for the reliability of the process, for example the fear of gate oxide breakdown.[28]. The increased ambient temperature worsens the electro migration reliability problems when the power dissipation of a CMOS VLSI chip increases[29].

Reduction of Vdd

Scaling down Vdd is one of the most popular power consumption reduction techniques. Given the expression

$$P_{switching-total} = V_{dd}^2 \cdot f \cdot \sum_{allnodes} a \cdot C_{load} \, [W]$$

we see that the voltage is the dominating variable, and therefore also the most efficient to reduce[26, 28]. But reducing the supply voltage is not something one can do without taking several aspects into consideration. When the supply voltage creeps under one volt several "new" aspects has to be taken into consideration. They are new in that respect that the effects we now have to take into consideration has for a large part been neglectable before. These effects are among others reduction in areas used particularly in analog systems. It will be more difficult to get a transistor to be anything other than true or false. The reduced level for the threshold voltage will lead to a need for an artificial high level on this value to be able to turn a transistor off. High leakage currents will lead to the construction of low leakages gates which in turn will lead to reduced possibilities to take advantage of the new possibilities presented by these new techniques.

Reducing the swing of signals

One such new technique is to reduce the swing of a signal, particularly the clock signal. By reducing the swing on the signal the dynamic power consumption can be significantly reduced, since the consumption is generated by changing the clock signal from GND to Vdd typically. This is achieved by reducing the signal to a minimum value above GND and a maximum value belove Vdd. Up to 32% power saving has been reported for a 250nm technology. This result is archived by reducing the swing of the clock signal on the distribution net with a level shifter. And then as close as possible to where the clock signal is being used increase the signal again by another level shifter[30].

Dynamic frequency clocking and multiple voltages

The voltage a particular circuit is given as its Vdd is normally decided by what processing power you need. Normally this can be traced down to one or a few critical paths. By finding these critical paths in a design one can decide where there is a risk of getting bottlenecks in the signal path. These paths are given the power they need. But less critical paths can maintain their required processing capabilities with less power. Thus by providing a circuit with multiple voltages, one can save power. Energy consumption decreases quadratically with voltage. Of course this give raise to other problems in circuit design such as synchronizing signals which will travel at different speeds in different parts of a circuit, and different speeds depending on which routes a signal propagates through a circuit. Another technique is dynamic frequency clocking. This is a concept where the frequency of the circuit is changed on the fly depending on the ongoing operation[31]. Multiple supply voltages has the advantage of allowing modules on the critical paths to use the highest voltage level, thus meeting the required timing constraints while allowing modules on non critical paths to use lower voltages, thus reducing the energy consumption. The main idea is to synthesize the design to the regions with different supply voltages[29]. Dynamic voltage scaling techniques varies the clock frequency and supply voltage according to workload at run time to reduce dynamic power and save energy. AMD's Mobile Athelon and Intel's XScale are examples of processors using dynamic voltage scaling[32].

Power down

By powering down the circuit when it is not in use obviously saves a lot of power. The problem with this scheme arise in regards to the powering up and down of the circuit. For an advanced circuit it is not just to put the power on and everything works from the first clock cycle. It has to have a power up sequence to start the system itself and to make sure a possible existing pipeline is started correct.

Parallelism

This method implies that instead of making a circuit run at high enough frequency to do a job one designs the circuit to do the job in several parallel pipes. The method of using parallelism to reduce power consumption has two major disadvantages. It needs a lot of area, and synchronization problems will have to be solved. [26]

Semi-floating-gate circuits

Utilizing semi-floating-gate circuits will reduce the maximum power consumption of a circuit. This will lead to two main things. Less heat generated due to the short circuiting of a circuit. And since the circuit will have a lower peak consumption of power wiring for power lines can be reduced, freeing up space for logic and other routing[1].

2.3 Voltage mode? Current mode?

Two kinds of multiple-valued logic circuits based on MOS technology have been developed, namely the current-mode and the voltage-mode multiple-valued logic circuits[33]. The actual difference between voltage mode and current mode circuits comes from the preferences of the different research groups. Since every circuit node has an associated voltage and every branch an associated current, and it is a matter of definition which ones represent signals and which ones do not[34]. Of course there are definitions dividing current mode and voltage mode systems. One such definition is that when an application needs multiple different voltage levels for off chip transmission they are considered voltage mode. When the application corresponds to implementations of threshold functions the circuits are considered current mode[13]. For the work done in this thesis it is worth noticing that current mode multiple-value system has larger theoretical head room, but this can only be achieved by time. This means if you need a high headroom, you have to wait for it, and you can therefore not make fast systems. This is making current mode system unusable. Also current mode circuits is getting worse as the transistors becomes smaller and smaller and the available headroom shrinks[35]. Current mode systems will also have a static power consumption which in many cases will be unacceptable large. A solution to this problem is the use of neuron MOS also called vMOS, or multiple-valued semi-floating-gate techniques. These circuits give less problem in regard to power consumption. These circuits can be fabricated by standard CMOS processes, and they have some useful properties, such as gate-level weighted sum and threshold operations[36, 37, 38]. But it is worth noting that the work done on current mode may give valuable insight to voltage mode designers who need to build low power, low voltage or high speed circuits and are ready to trade in gain variations, distortion, or noise performance to reach their goals[34]. Furthermore only voltage mode multiple-valued circuits could be used for reducing the number of wires between different logic blocks[21].

2.4 New technology; Possibilities or just problems

The continuous development of new technology, here meaning the continuous effort to reduce the size of the building blocks utilized in VLSI constructions leads to new possibilities, new challenges and new problems. Particular for the problem at hand, multiple-valued systems the reduced "workspace" is a major disadvantage. The noise margins are getting smaller and smaller making it more difficult to build robust systems.

2.5 Circuit elements

The semi-floating-gate recharge logic has its basis from one article by T. Shibata and R. Ohmi[38], and a series of articles originating from the Department of Informatics at the University of Oslo[27, 39, 40]. In the article by Shibata and Ohmi the motivations for introducing multiple-valued logic is to reduce the number of transition lines, since these are the ones producing more and more of the delay, and the increased problems related to errors in transmitter lines. These errors are related to the limitations in transmitter lines concerning how close lines can be placed to each other without getting crosstalk. They also outline the possibilities of making certain logical functions using fewer gates. In the articles by Berg et.al. such circuits have been presented. First the circuits presented by Berg et.al. had a UV-programmable gate. The floating-gate by this method received its charge from a programming face done before the chip is used[41]. All the circuits presented are based on a transistor with floating-gate and multiple-input-gates that interact capacitively with the floating-gate[36]. These circuits are suitable for low power circuits. The multiple-valued semi-floating-gate inverter and its many properties when slightly modified makes it an ideal start for an adventure into the domain of the world of multiple-valued electronics. This inverter has a number of properties which makes it a good choice for a number of boolean and arithmetic operation. As the name suggests its primary function is as an inverter. But slightly modified it can be made to do other boolean functions such as AND, OR, NAND and NOR. The same inverter can be utilized as an adder, subtracter and as a memory element in a latch. Trough the next pages there will follow a description of the circuits utilized in the design of this assignment in detail and description of others more superficial. Several circuit elements has been proposed over the years.[42, 43]. By recharging the semi-floating-gate we do not only avoid the problems linking to programming or initialization of the floating-gates, but we convert the non-volatile floating-gates to semi-floating-gates. The control of the actual floating-gate charges in terms of predictable long term charge restoration becomes easier. The semi-floating-gate is not in-

fluenced by a random floating-gate charge distortion due to a periodic or frequent charge restoration or reset[44]. These circuits has good frequency versus power efficiency, they will benefit from reduced area consumption in regard to wiring leading to more available area for logic. With the reduction in wiring one also get the added benefit of reduced crosstalk[45]. One purpose of multiple-valued semi-floating-gate is to level out power dissipated by a digital system to obtain more suitable logic for mixed mode design. There are may problems associated with multiple-valued logic such as noise margins, speed accuracy or precision and memory design. When using semi-floating-gate we avoid problems linked to programming or initializing of the floating-gate. By recharging the semi-floating-gate frequently we avoid the problems with leakage currents and random or undesired disturbance of the floating-gate charges. When reseting or recharging a gate the inputs are recharged simultaneously and not set to a reference voltage normally Vss or Vdd. A random mismatch will effectively limit the resolution in a multiple-valued system, while a systematic mismatch (design error) will result in an increased error depending on logical depth. If a significant systematic error is evident the system will require more or less frequent conversion to and back from binary representation in order to refresh the signals[43].

2.5.1 Capacitors

The capacitor is an important part of these circuits. The most important thing when choosing a capacitor is that it is accurate at the same time as it should be as low in terms of load as possible. The added load of the capacitors is the main reason for the semi-floating-gates somewhat reduced efficiency in terms of frequency. Therefore it is important to have these added loads as small as possible. The easy way to improve the accuracy of a capacitor is to increase the area of which it is made. Therefore the balancing of load versus accuracy is a big issue in the design of semi floating circuits. Two capacitors have been implemented, one a poly poly capacitor and another a fingered several layers metal capacitor. The poly poly capacitor offers ease of implementation where as the metal capacitor offers better accuracy and when larger capacitors is needed it does not increase much in physical size, making it relatively less area hungry than the two plated poly poly capacitor. See figure 2.1.

2.5.2 Auto-zero

An auto-zero-circuit is necessary for the following circuits to work due to the semi-floating-gate. An example of such a circuit is illustrated by figure

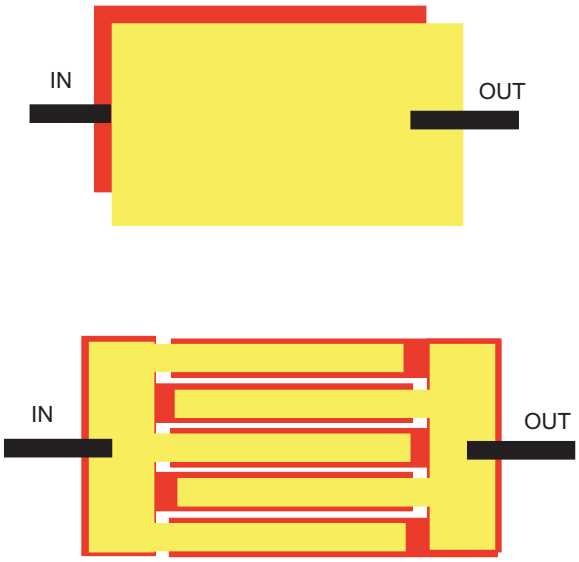

Figure 2.1: A concept figure of a plate and a finger capacitor. The plate capacitor offers ease of implementation whereas the finger capacitor offers more capacitance per unit area.

2.3. This floating-gate need to get a known voltage, and it needs to be equal in all circuits in a system to get a predictable system. The way we set the voltage on the gate is to reset it to a known value each clock cycle. This is done by short circuiting the input of the floating-gate inverter with the output. To make sure we get the value Vdd/2 on the gate we send a Vdd/2 signal in on the circuit as the circuit is short circuited. Two auto-zero circuits has been my main focus through this assignment, though many more exist. The first one[46], see figure 2.2. This circuit is composed of two inverters with two extra transistors controlled by the clock. The other way I have achieved the auto-zero function is by a circuit which is just a simplified version of the one mentioned. It consists the same two inverters, but fewer transistors controlling it. Because of this added simplicity this auto-zero circuit performs slightly better in simulations. I have therefore used this circuit for my experiments see figure 2.3. Schematic simulation of the "three transistor version" can be seen in figure 2.4.

2.5.3 Inverter, from a digital tool to a versatile analog find.

We start with a normal inverter as we see in figure 2.5. By adding a reversed coupling to it with a capacitance, figure 2.6 we get an inverter which we can

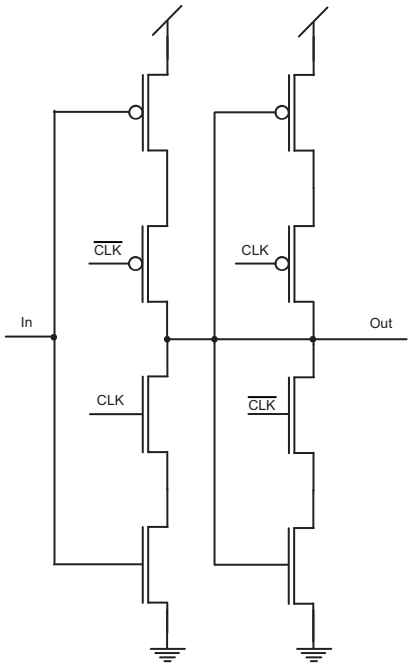

Figure 2.2: Auto-zero-circuit, the “four transistor version”

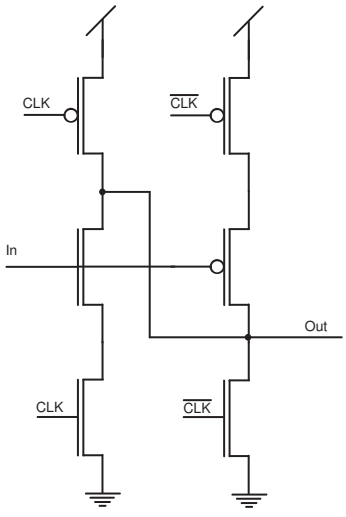

Figure 2.3: Auto-zero-circuit, the “three transistor version”

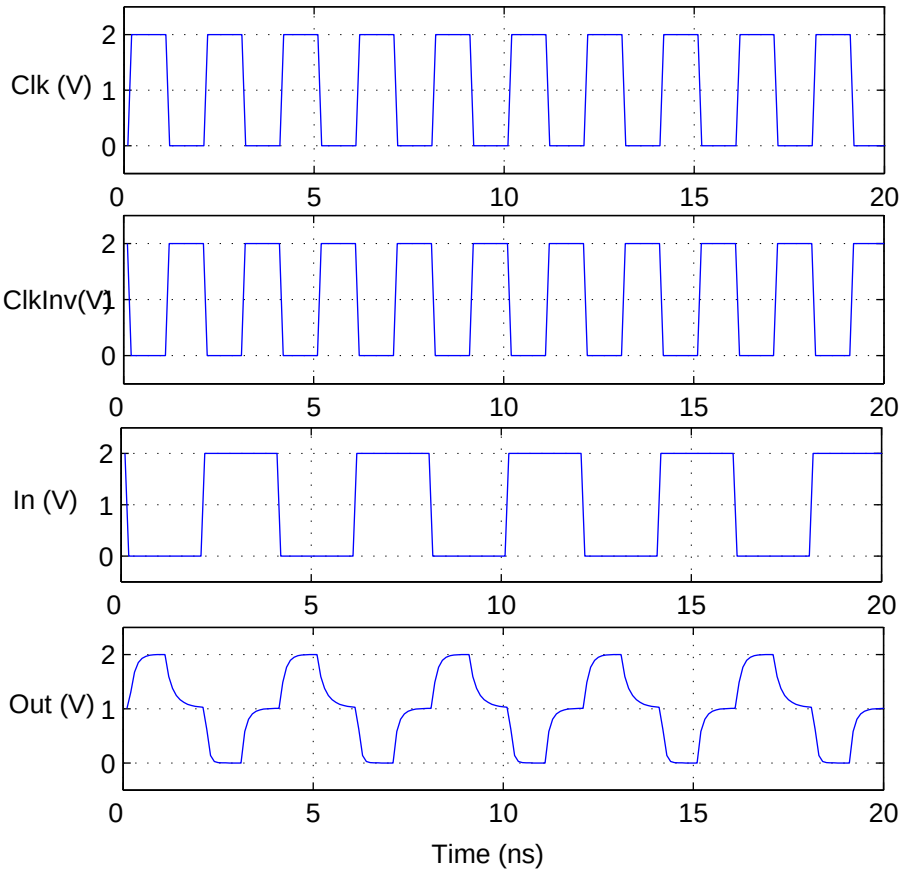

Figure 2.4: Auto-zero simulation in schematic at 500MHz

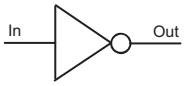

Figure 2.5: Inverter

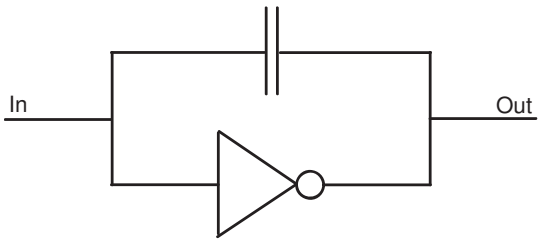

Figure 2.6: Analog-gate inverter

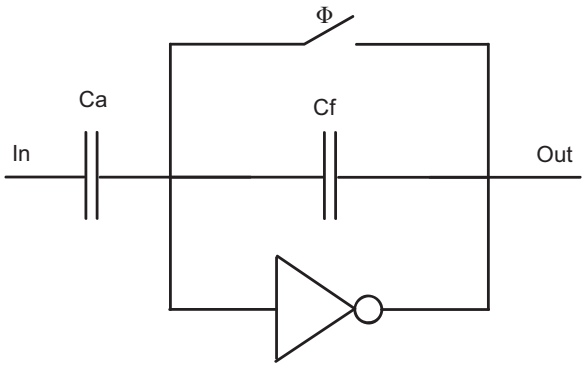

Figure 2.7: Semi-floating-gate analog inverter

adjust the amplification. For this to function properly we need to operate the transistor in saturation. To make it a floating-gate we add a capacitance on the input side. To be able to control the charge on the floating-gate we make it a semi-floating-gate. This we achieve by short circuiting the system regularly, as seen in figure 2.7. As mentioned before, used as an inverter in a multiple-valued system this inverters amplification can be adjusted. This is achieved by adding a feed back capacitance to the system. The semi-floating-gate multiple-valued inverter has a weighted negative feedback mechanism, and ideally the gain is -1. Ideally the capacitor in front of the inverter is of the same size as the one coupled in negative feedback, but due to output conductance and the parasitic capacitance, the negative feedback capacitance has to be a bit smaller[44]. When the capacitance of the capacitor in front of the inverter is equal to the feed back capacitance included the parasitic capacitances of the transistors we have obtained an inverter with a gain of one. This property is the main thing here as any gain other than one will skew a multiple-valued signal. To much gain will make the signal binary, thereby making the signal impossible to decipher. To little gain will likewise make the signal shrink so that any later circuits will have a hard time deciphering the signal and the signal will be lost in this instant as well. Any transistor or capacitor mismatch will limit the multiple-value resolution [42, 46, 47, 48]. Assuming that the feedback capacitor Cf including the parasitic capacitance of the transistors and input capacitor Ca are equal see figure 2.7 for reference, the circuit will produce the analog inverted output of the input. If the input signal is multiple-valued the output will be multiple-valued with the same radix and the semi-floating-gate will remain at Vdd/2 for all input values[43].

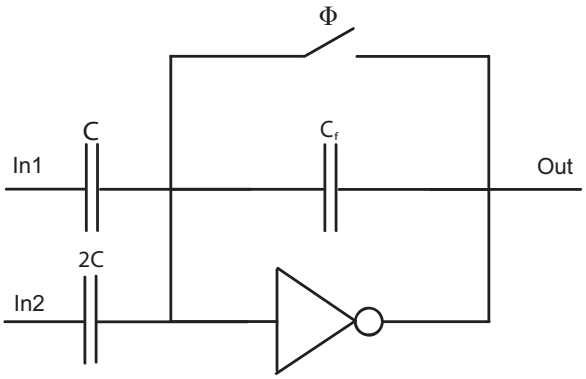

Figure 2.8: Two input digital to multiple-valued converter. where the inputs have different weights.

2.5.4 Encoder or binary to multiple-value converter

The main idea behind introducing multiple-valued recharge logic is to reduce the dynamic power consumption. But we need to realize that the rest of the world is digital, and therefor converters are needed. In figure 2.8 a digital to multiple-valued converter is shown with simulation plots in figure 2.9.

2.5.5 Multiple-input semi-floating-gate inverter

The AND, NAND, OR and NOR functions is achieved by modifying the input of the same inverter described in the previous section and removing the feedback capacitor. The clocking and the third input of the circuit seen in figure 2.10, decides if we get a (N)OR or a (N)AND function. We achieve the NAND and NOR function when the clock is operated normally, if we reverse the clock we get the AND and OR function with a delay of one half period as this cause the signal to be latched. The resulting signal is a normal binary signal with an auto-zero inlaid as shown in figure 2.11. For the multiplication circuit described later in this thesis the NAND operation of this circuit was used.

Another feature this circuit can be used for is additive mixing, by applying two independent signals valid in the same clock phase the signal will be added. By applying two independent signals valid in different clock phase the signal will be subtracted[43].

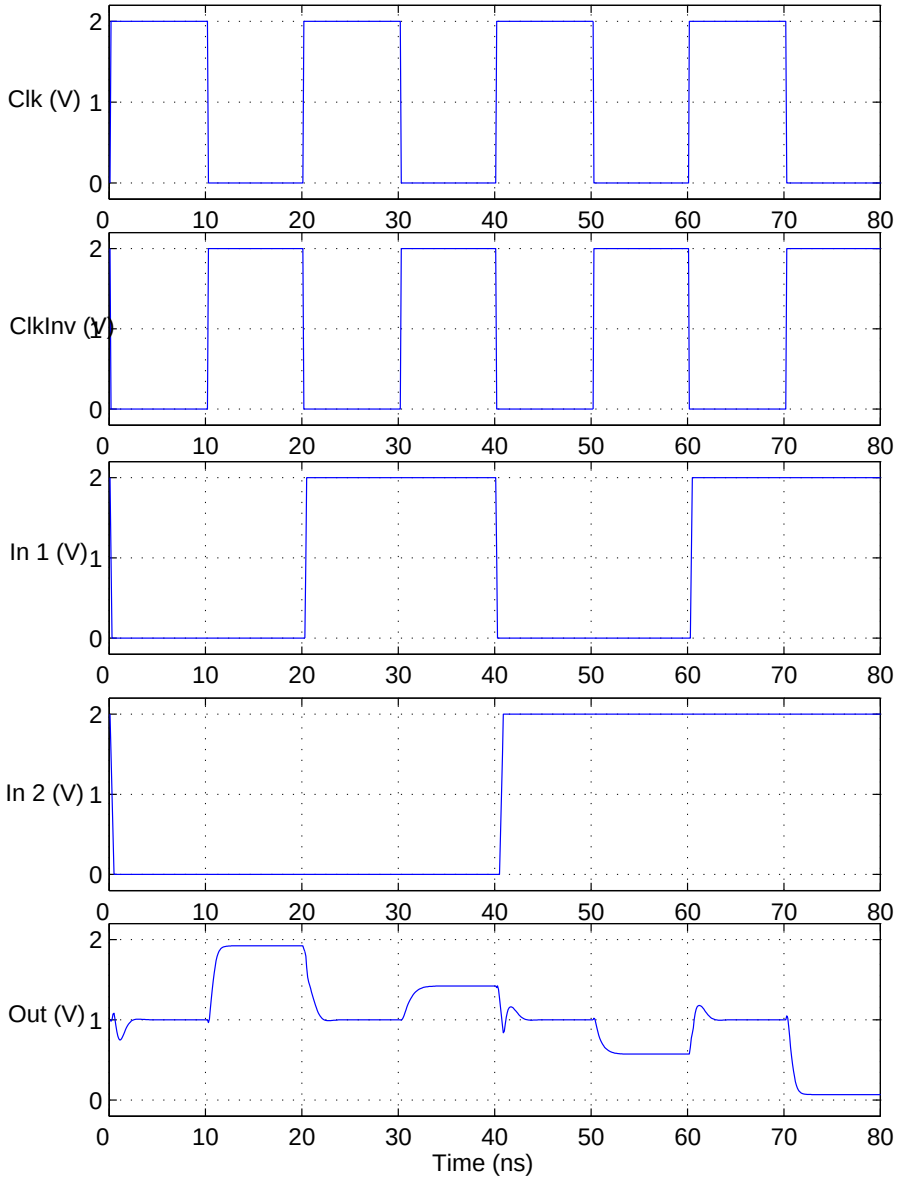

Figure 2.9: Digital to multiple-valued converter simulated in schematic at 50MHz

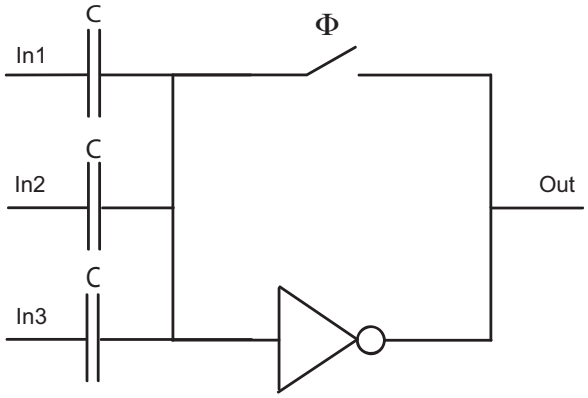

Figure 2.10: Multiple-input semi-floating-gate inverter

2.5.6 Adder

Another circuit it is worth mentioning is the multiple-valued semi-floating-gate full adder seen in figure 2.12. When a signal is used as an input a weight is associated with the signal. The weights can be used in multiple-input gates to combine signals of different significance[43]. This is what is exploited in this circuit. Used this way the adder adds a chosen number of digital or multiple-valued signals to one multiple-valued signal. It can be made to add signals of different magnitude or of the same magnitude. This adder has been reported to give less need for carry propagation. It also, as all multiple-valued semi-floating-gate circuits give less power consumption in terms of peak power consumption and reduced dynamic power consumption. The area consumption for this adder has been reported to be 68% less than for its binary counterpart. We can also reduce the number of transistors required for adding two signals.

2.5.7 Decoder or multiple-value to digital converter.

The analog to digital converting is done by the last part of a full adder shown in figure 2.12. The simple converter provides binary output and is used together with simple Digital to Analog Converters to restore or refresh a multiple-valued signal[49]. It is very difficult to realize a multiple-valued to digital converter with a small number of conventional transistors because each of the conventional transistors can detect only two signal levels. It is expected that the complexity of a multiple-valued digital system can be greatly reduced if a new device capable of detecting multiple signal levels is developed.[3].

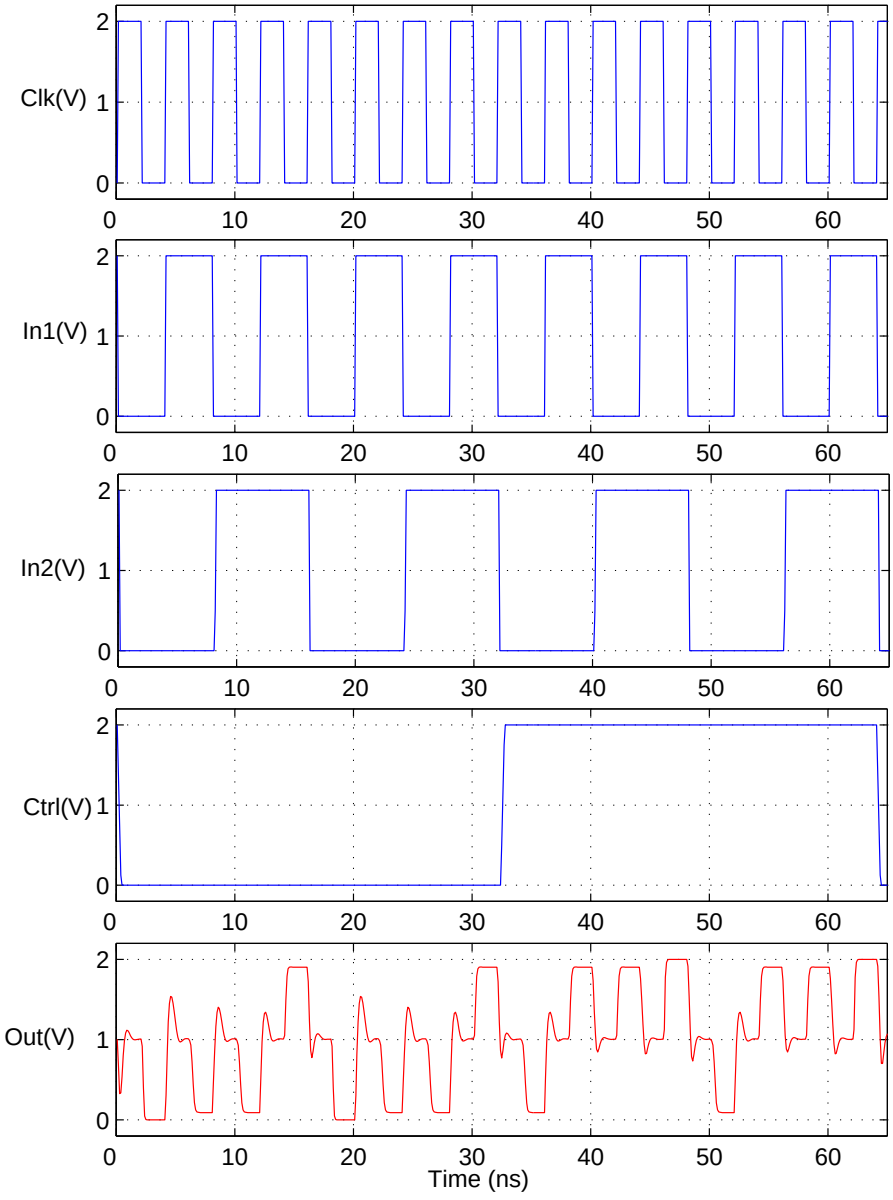

Figure 2.11: Multiple-input semi-floating-gate inverter 250MHz schematics simulation

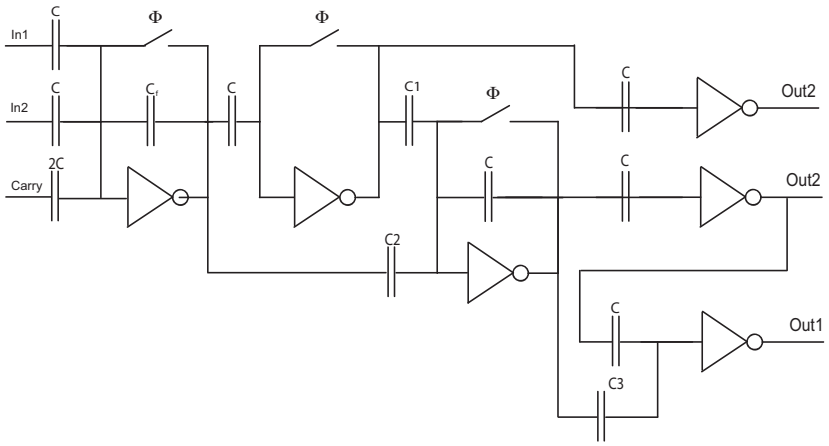

Figure 2.12: A Multiple-valued semi-floating-gate fulladder with decoder. C1 = $\frac{4}{3}$C, C2 = $\frac{7}{3}$C and C3 = $\frac{3}{2}$C

2.5.8 Memory

In general memory is a necessary and space demanding part of any major logic design. As with all micro-electronic the memory parts of the circuit has had its size reduced with the rest of the design. But now there is a need to accelerate this development faster than what increase in lithographic technology can do. This has led to the development of a four level storage DRAM[50]. When four levels are stored in a single memory cell the effective cell size is halved. As circuit elements become smaller the dominating area consumer on a chip is wiring. One promising way to reduce the need for extensive internal wiring is to make the on chip memory multiple-valued. It is demonstrated that the numbers of interconnections and transistors in a 5 value associative memory can be reduced to 25% and 53% in comparison with the corresponding binary implementation.[51]. Memory is as mentioned in this chapter the most promising field for multiple-valued circuits. One way of utilizing multiple-valued semi-floating-gate structures in memory is to use them for the logic in memory, possibly removing communication bottlenecks between memory and logic[52, 53]. One proposal for analog floating-gate memory has been presented[54]. Another way memory easily can be made by two analog inverters in a loop, and inverting the clock on the two circuits, in other words a latch. This is shown in figure 2.13. This memory module has some severe problems in regard to holding a multiple-valued value for even a few clock-cycles. Only the tiniest difference in gain will pull the signal either to one of the rails Vdd or GND, or to Vdd/2. The easiest way to circumvent this problem is to replace one of the inverters, which in practical terms is a digital to ana-

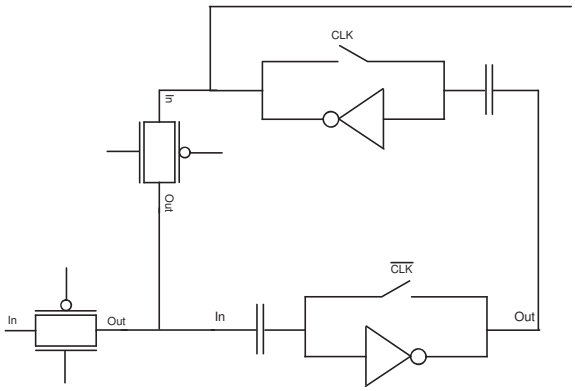

Figure 2.13: Memory module

log converter, with an analog to digital converter. This means that any errors made in a clock cycle, will also be corrected in the same cycle. In other words we need to refresh multiple-valued signals, converting to and from binary signals to avoid problems with accumulation of minor voltage level errors caused by multiple-valued gates[48, 55]. This is a somewhat more area consuming circuit, but unless the memory is only going to keep the values for a known very short period it is essential for the guaranteed functionality of the circuit[2]. Increased memory capacity with the utilization of multiple-valued logic, for example when magnetic storage needs to be replaced with something else. Here the access time is far less critical than in other types of memory. Here there is a possible niche for multiple-valued memory[21]. Phase shift is not considered to be a malfunction or a problem, but merely a question of synchronization[16]. Read only memory design has been presented by Intel, Motorola and General Instrument and RAM has been presented by Motorola[13].

2.5.9 Removing the auto-zero signal.

To go from the multiple-valued world back to the digital world when we have an otherwise binary signal we need to remove the auto-zero element of the signal. This is done by a circuit working in much the opposite way of the auto-zero circuit described above. A schematic view of this circuit is shown in figure 2.14.

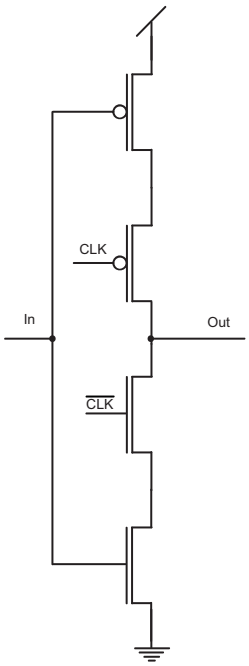

Figure 2.14: Removing the auto-zero component in a signal

2.6 Clocking

Throughout these circuits there has been made an assumption of a perfect world. There is one problem arising in these circuits if we utilize latching, the clocking scheme. For the latch used in the memory element using a two-face clock, one needs the clocks flanks to match perfectly. This is not easily possible in a large system. The problem is this. With a minimal skew one will effectively get a degradation of the signal because during some of the evaluation face the signal will be Vdd/2. This is in general not a good thing, in a system consisting of multiple-valued circuits it could very fast become catastrophic. Establishing that this is a major problem the next step is to try a two face clocking scheme. Analyzing this scheme shows that it will have the same problems as the first clock. The next step then is to try three face clocking. This scheme shows promise. Here we are able to isolate the various in and outputs in a chain of latches, it is a solution. Another proposed approach to this problem would be a single phased clock, but making it non overlapping by having a period of each flank at the Vdd/2 level. This could give the necessary safe margin we are after for a robust system. A fourth and more drastic suggestion for clocking scheme is to use the clock signal to turn the circuit completely off by reversing source and drain by using the clock signal for Vdd and GND.

2.7 Parallel and serial multiplier

By combining the above mentioned circuits I have made a parallel and a serial multiplier. Using the multiple-input semi-floating-gate inverter for the AND operation, and using the multiple-valued semi-floating adder for adding the products. Using the multiple-input floating-gate inverter for this operation gives great flexibility in the design of the multiplier. We actually do the first evaluation using only four transistors and three capacitors. The multiple-valued multipliers gets a real advantage when we start adding the multiplied signals. Depending on the resolution of the adder used we can reduce or eliminate the carry-propagation normally associated with multipliers. Of course combining the multiplier with a look ahead adder will further reduce the delay caused by carry propagation[56]. In my design I have tried to use the same building blocks in the entire design. This should improve the chances of implementing a working circuit[25]. The multipliers are further described in chapter 3.

2.8 Multiple-valued circuits

Multiple-valued logic has the potential of enriching the digital world[19]. Even so multiple-valued circuits has one major problem which binary circuits has not. As long as the binary signal is above or belove Vdd/2 the signal will regenerate it's proper values. Multiple-valued signals have much less headroom. With a large number of values only a small disturbance could change the value of the signal. This leads to the a need for refreshing the signal by converting it to and from binary signals to avoid accumulating small errors in voltage levels. Although this weakness needs to be addressed we can conclude on the basis of past literature that multiple-valued circuits has the potential of reduced area consumption, reduced power consumption and has the potential of solving some of the problems facing future VLSI designs and designers. It also has the potential of providing a mean of increasing the data processing capabilities per unit area[55]. Reduction in the needed die area will also give a corresponding increase in die per wafer yield[6]. Since one now has the additional design parameter of power consumption in addition to area and throughput, multiple-valued circuits may be on the rising[31]. Generally using larger capacitors gives less problems with mismatch, also noise margins are reduced using larger capacitors and larger transistors. Transistor and capacitor mismatch will limit the multiple-valued resolution. A multiple-valued signal need to be refreshed to ensure its liability by eliminating errors. The number of radixes should be chosen as a tradeoff between what you can calculate most efficiently and what you have to use as input and what you have to repres-

ent as the output. But systems with radix of two shows most promise. One can then convert to and from digital systems without using excess logic since the multiple-valued signal will add up exactly[19, 46, 47]. To minimize the effect of noise, the separation between signal levels should be made relatively large. However, when a requirement for speed is added to a logic system, a conflicting requirement on logic value separation is added. This requirement implies that signal value separation should be made as small as possible, particularly on generally distributed signal lines[6]. For multiple-valued circuits to compete successfully against their binary equivalent circuits they should show some performance advantages and exhibit the same exponential performance trends as digital logic[21]. It is possible and demonstrated that these circuits can be produced in standard fabrication processes, but a process with a small parasitic capacitance will be better suited[11]. Multiple-valued circuits and two valued circuits must not be seen as competitors. If they are seen as such, then two valued circuits have already won[13]. Multiple-valued circuits gives the possibility of designing custom logic systems which should give more efficient systems. What you do is you remove don't cares from the system and thereby remove a lot of unnecessary circuitry[57]. The kind of new ways of designing micro electronics as described here may be the way to continue to make circuits with increased efficiency in the future. The multiple-valued semi-floating-gates proposed can, and has been produced in standard CMOS-technology, and has been designed in commercially available software.

Chapter 3

Multiplication with multiple-valued circuits

The philosophy behind my multipliers is this. For the parallel multiplier I have made an array of multiplier elements, the multiple-input semi-floating-gate inverter. These elements as in a digital multiplier first produces a partial answer. This answer is then added in an adder. The serial multiplier consists of one multiplier element and several memory elements. The partial results are fed into the memory elements. These are then sent to a multiple-valued adder and added there. One thing to note is that due to how the multiple-valued adder functions it inverts the result. So a NAND-gate has been used instead of a AND-gate as the element to evaluate the numbers to be multiplied.

3.1 Parallel multiple-valued multiplier

An alternative way to implement a multiplier is to utilize multiple-valued semi-floating-gate adders and the multiple-input semi-floating-gate inverter earlier described. The multiplier resembles a digital array-multiplier. In the digital multiplier the two numbers which is going to be multiplied is fed two and two of the same weight in to the array. Then an AND operation is done on the numbers. This operation is a straight forward operation when done with only two bits and is efficient. The problem with the digital array-multiplier starts when all the results of the AND operation is to be added together. Here we face the risk of massive congestion of the carry signal or bit. Ways to avoid these problems have been investigated and better solutions than driving the carry through with brute force has been proposed. In this thesis I propose to use the multiple-input semi-floating-gate inverter to

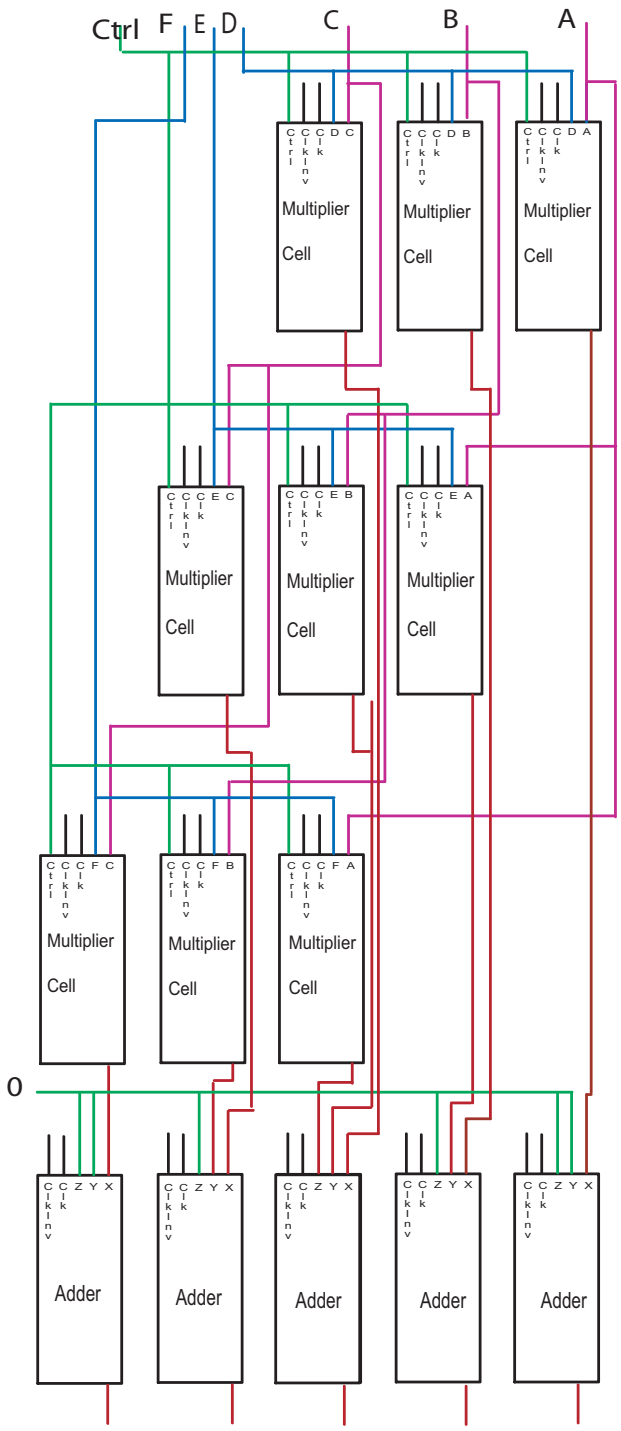

Figure 3.1: The parallel multiplier with the corresponding inputs from table 3.1, the outputs at the bottom from right to left output one through five.

		C	B	A
		F	E	D
		CD	BD	AD
	CE	BE	AE	
CF	BF	AF		
5	4	3	2	1

Table 3.1: This table illustrates the parallel multiplier with the inputs A through F being evaluating giving the partial results from AD through CF and showing the respective partial results weight. The numbers one through five illustrates the multiplied result.

do the AND operation and the multiple-valued semi-floating-gate adder to do the addition.

The multiplication method used in my parallel multiplication is this: Facing two binary numbers I use the boolean NAND operation on the bits of the same weight. I use a NAND operation since this operation reverses the inversion done by the latter adding stage and it also does not give the half clock cycle delay which the AND operation would have given the signal. The results of these NAND operations of the same weight is then added together by a string of adders. For the horizontal line there is no limit for the multiplication possible. But for the vertical line the number of multiplications possible before one needs a carry bit is determined by the number of levels one will allow in the signal. It is in this adding procedure that the multiple-value scheme comes to its right. as the problem with propagating the carrier bit trough the network is at best removed completely, or at the very least reduced significantly. In table 3.1 and in figure 3.1 the concept is illustrated. The partial results with equal weight is added together. These partial results again could then be added with a multiple-valued semi-floating-gate adder. By designing the system in this manner one would get a system with an output in one clock cycle. No carry propagation, no delay. In figure 3.2 a simulation is shown of the corresponding signals from table 3.1 and figure 3.1.

3.2 Serial multiple-valued multiplier

After the parallel multiplier was implemented I came up with the thought to utilizing the multiple-input semi-floating-gate inverter for serial multiplication in combination with a multiple-valued memory element. The memory element used is the one described in section 2.5.8. The serial multiplication is made up of one multiplication element, the multiple-input

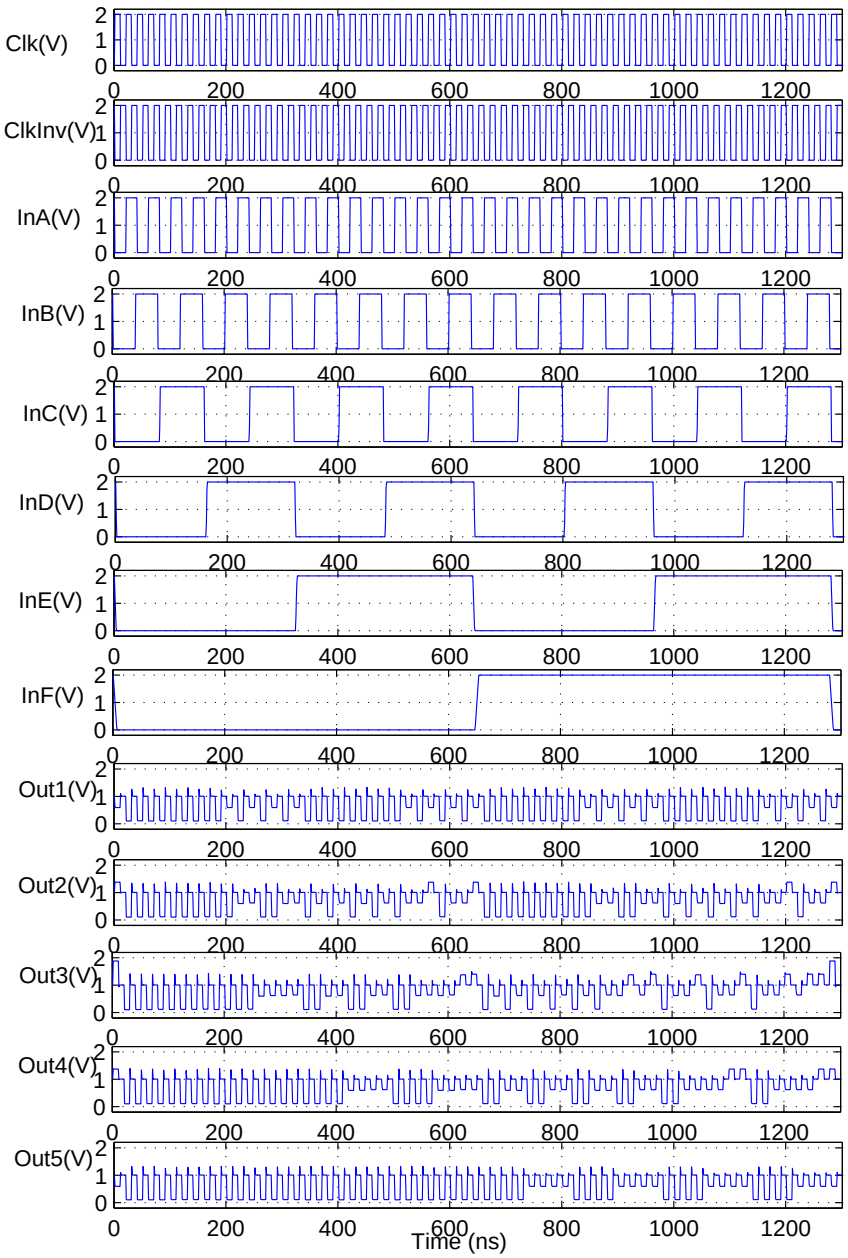

Figure 3.2: Parallel multiplier 50MHz schematics simulation

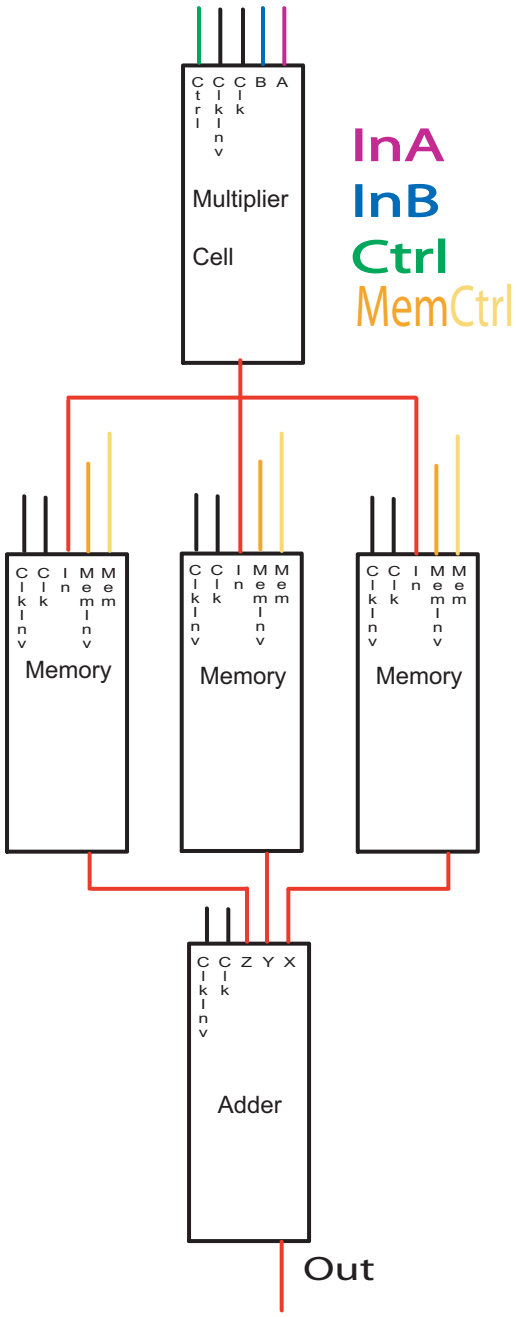

Figure 3.3: The serial multiplier

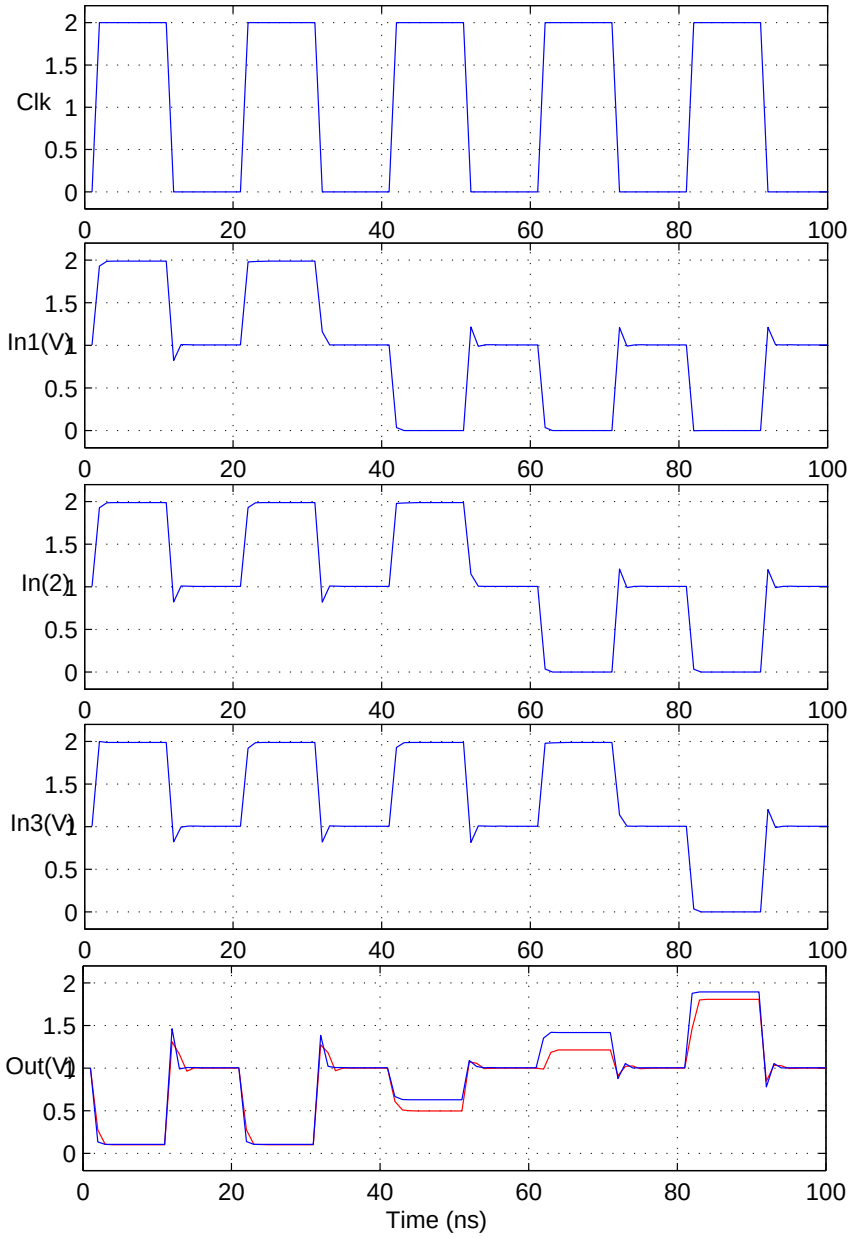

Figure 3.4: Serial multiplier 50MHz schematics and layout simulation. The blue output line illustrating the schematic simulation, and the red line illustrating the layout simulation.

semi-floating-gate inverter which performs a NAND operation on two digital signals. This circuit is illustrated in figure 3.3. The partial results are then stored in different memory cells. A control signal is determining which cell is getting the signal. When the first steps in the process is done an adder adds the partial results and the result is given. A plot of a simulation of this circuit is shown in figure 3.4. The serial multiplier utilizes another feature of the semi-floating-gate inverter. Namely used as a memory element. Here it is not possible to generate the answer in one clock cycle, because of inherent time delay in the memory cell itself. The memory design is the main challenge here. The memory need to hold the value long enough as mentioned before. For this circuit I have used the simplest form of memory, and it is working, but would not be good enough to hold the values for a larger system where the value would have to be held for a longer period of time.

3.3 Multiple-valued multipliers, so what?

The foremost advantage of the multiple-valued multiplier is the option to reduce the use of carrier-bits needing to propagate through the circuit. This option will make multiple-valued systems more flexible in terms of use as the real speed of a system consisting of multiple-valued elements will increase. This increase is necessary to achieve the demands set forth for new technologies to gain a foothold in a set paradigm. To overcome a paradigm one not only has to demonstrate that a system is equally good, it has to be better and it has to have a potential of improvement in the near and distant future. Of course the other advantage and disadvantages of multiple-valued semi-floating-gates mentioned earlier will exist for the multipliers as well. Particularly the reduced power consumption will be a good thing now and in the future. The multiple-input semi-floating-gate inverter offers advantages in the ways of power dissipation and simplicity of design in comparison to the digital AND-gate. Using only one inverter, three capacitor and two transistors for the necessary clock signal to set a value on the floating-gate we have a very neat design. Further, and more beneficial even than the multiple-input semi-floating-gate inverter is the multiple-valued semi-floating-gate adders properties for this design. It offers a way to add a number of the results from the gate making the NAND operation. Meaning that using this adder one is not restricted to using one adder for each AND-gate as in a conventional design. By using multiple-valued adders it is possible to reduce the number of adders needed significantly. In my design I have chosen to add results of the same power. By doing this I have achieved a reduction of adders in a three by three array from three digital half adders and six digital full adders to five multiple-valued adders. Also

serial multiplication, involving a memory element gives great flexibility in design options in regards to how the numbers being evaluated could be feed into the system, and how the results could be presented.

Chapter 4

Experiments

To test my designs I started with simulating and making the layout for a 350nm chip. This process was a fairly straight forward process as several chips in this process has been design at the group. This chip resulted in a working parallel and serial multiplier. Also test elements of structures used in the respective multipliers was made and available for taking measurements on this chip. An auto-zero element, the multiple-input semi-floating-gate inverter and the memory module are among these test elements. During the project a possibility for producing a chip in a for us new technology, a 112nm technology became available. Two fellow students working on an ultra-wideband project had free pads on their chip and was willing to let me use some of the available pins. Originally I intended to make a 8x8 array-multiplier on this chip. But as we got more knowledge on the new process we found several challenges. What affected my test element on this chip the most at first was that the number of pins was reduced. In our original assessment of the availability of pins we thought it would be about as on the other processes we knew where for instance one pin is enough for Vdd for padframe and circuit elements and another pin for GND for padframe and circuit elements. But that was not the case here, not only was it required to separate the pad-ring and the elements powersupply, it was also required to have powersupply pads for the padframe evenly spaced around the frame. Due to this the test element made for this chip was reduced to a 2x2 parallel multiplier. Also the ultra-wideband test elements required possibilities for high frequency input and outputs. So a PCB was produced to accommodate these requirements. Therefore I also constructed my 120nm test element to be measured at a high frequency.

The two processes used in this thesis is of two different generations when it comes to CMOS processing. Whereas the 350 nm technology is as one would expect, a process where the gate and line delay is reduced from the previous processes (600 and 800nm). The 120nm process may be the first

process utilized here at this group where the gate delay continues to reduce in a new process, but the transmission delay between gates is significant increased. Resulting in more area available for design of logic circuits, but not necessary giving anything in the way of increase in speed[23]. Other challenges one faces utilizing minimum transistors in a process like the 120 nm process is the very short tunnel one gets under the gate of the transistor. Also due to the small sizes of the structures in the 120nm process the poly can not only be n-doped. The gate over the p-MOS is p-doped. This is to make the p-MOS better.

4.1 The 350nm chip

The implementation in the 350 nm technology was done as a normal design process. First the circuit was made in schematic to test the logic functions of the circuit. Then the system was transferred to layout and further simulations was conducted to verify the design. Finally the design was tested physically by measuring on an actual chip.

The circuit elements implemented on the 350nm chip is made using minimum sized transistors and as small capacitors as was practical to balance (5fF) as the minimum or unit capacitor. The circuits was designed to function at a supply voltage of 2V. The circuit elements on this chip was found to be working as illustrated by the measurements of the parallel multiplier in figure 4.1. The parallel multiplier multiplies two three bit numbers. The serial multiplier implemented has only three memory elements limiting the size of the numbers it can multiply.

On this chip I also implemented some of the circuits used in the system by themselves so it would be possible to measure them on their own. In figure 4.2 such a measurement is made of the auto-zero circuit. That this circuit functions properly is very important as it is used for all the input-signals in all the systems implemented.

4.2 The 120nm chip

Designing systems in the 120nm process was more challenging than designing the system for the 350nm chip. The circuits on this chip was designed to function with a supply voltage of 900mV. As mentioned before the pad-ring was one obstacle. In the 350nm design I used simple poly poly capacitors to make the needed capacitors for the design. In the 120nm process used here a- second poly layer is not available. Also due to mismatch between the hit-kit and the cad-software used it was not possible to extract parasitic capacitors as one would normally do. So it was not possible for me to

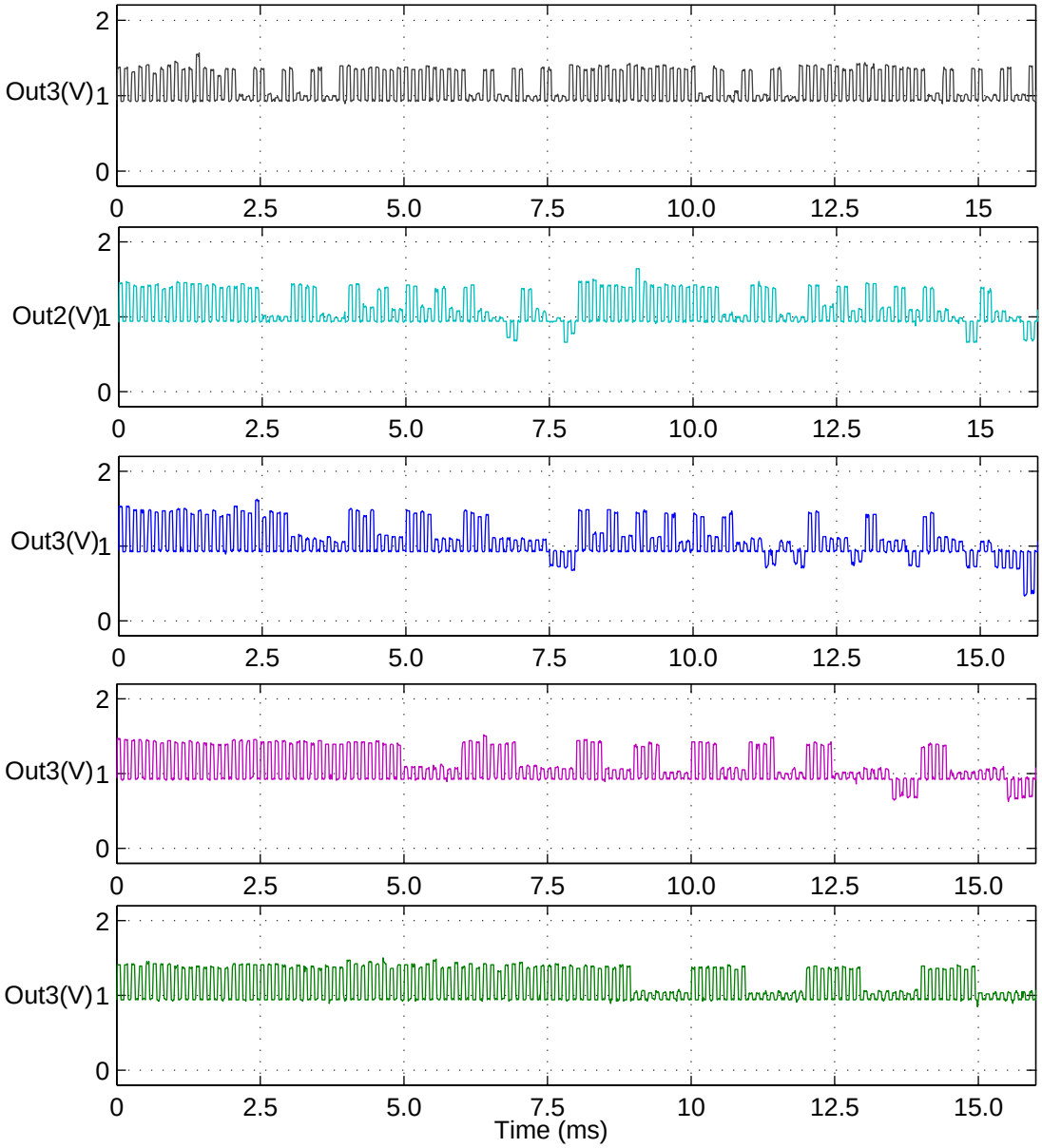

Figure 4.1: Measurement of the 350nm parallel multiplier at 4kHz.

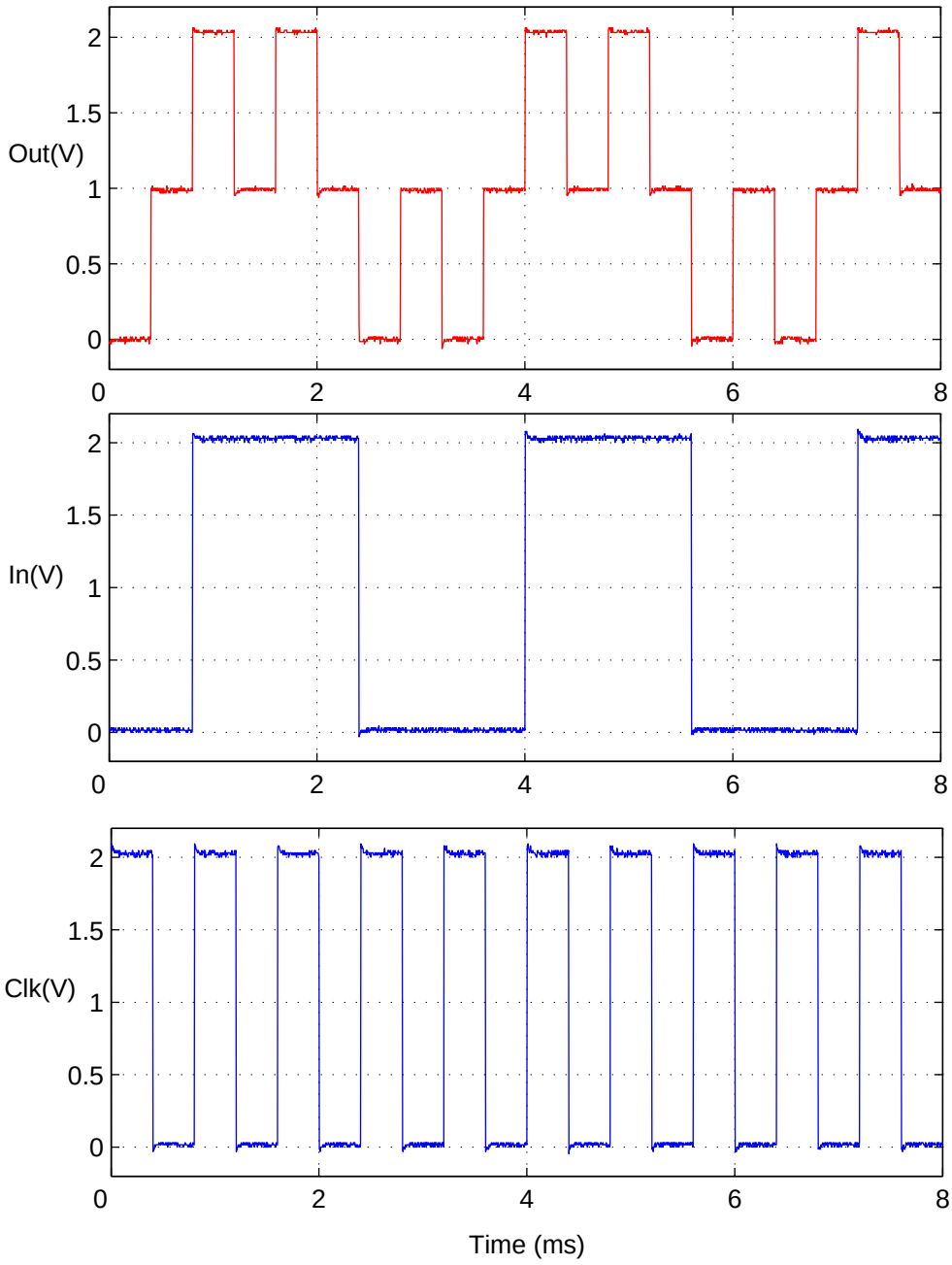

Figure 4.2: Measurement of the auto-zero circuit at 500Hz.

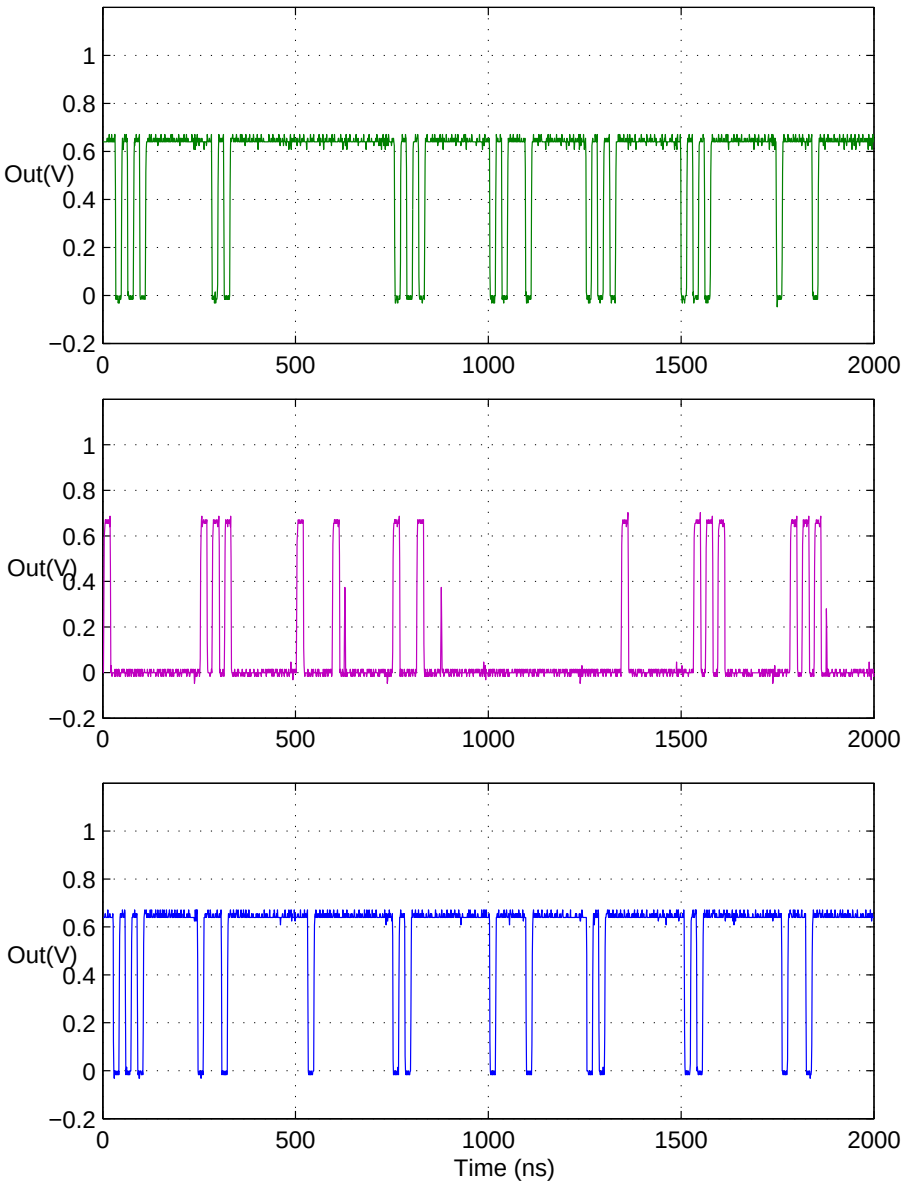

Figure 4.3: Measurement of the 120nm parallel multiplier at 2.5 MHz.

make my own capacitors. This left me with only one option for capacitor, one included in the designkit. This was a capacitor with minimum capacitance of 18fF. This is a much larger capacitor than I would have liked to use. But since there was no alternative the transistors were scaled up to match the large capacitors. Since the size of the design shrinked dramaticly from the one initially planed, the 8x8 array, I instead focused on making a circuit which it would be possible to measure at high frequencies. Therefore the multiplied signals was transferred from multiple-level signals to digital signals, and the auto-zero component was removed. This was done so that the signal could be buffered. The buffers was used to boost the signal of the chip. In figure 4.3 a few outputs from the chip is shown. These outputs are clearly digital, but they are not logically correct. There could be several reasons for this. My two main theories for this error is that multiple-valued signal has been skewed internally on the chip or that the problem originates on the PCB. Even though there was no option for extracting parasites on this design during the design process the design for the 2x2 array is mainly digital, and with the relatively large capacitors and transistors the design should be robust.

4.3 PCB

Due to high frequency demands in testing the 120nm multiplier, possibly up to several hundred MHz, a PCB had to be developed for the test setup. This was done in cooperation with two other students having test elements on the same chip. The PCB was developed using a CAD tool and it was produced externally by Elprint. One major challenge in the design of this board was the frequency demands. It proved hard to find components matching our speed requirements and at the same time being willing to chip small quantities of those elements. The entire process from learning the CAD tool through finding the right components to ordering the board from the manufacturer was done by us. To get the frequency desired for the clock and input signals the PCB was design to multiply the input-signal, and divide it to the different chip inputs. During measurements it became apparent that the board picked up very much noise from the surrounding area, making measuring hard. The amount of noise observed is among the reasons why I think the measuring problems for this chip may lay with the PCB.

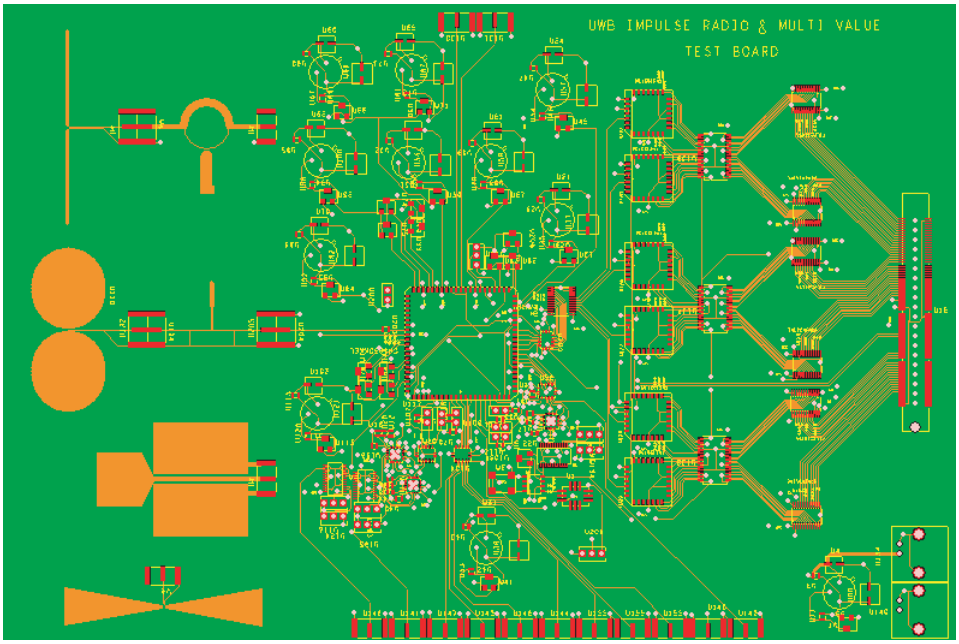

Figure 4.4: Layout of the PCB.

Chapter 5

Conclusion and proposals for further work

Through this thesis several powersaving methods has been descibed. One of these methods, the mutiple-valued semi-floating-gate has been used for practical experiments. I have simulated and implemented a serial multiplier and a two different parallel multipliers, using the semi-floating-gate circuits. These circuits has been produced in a 350nm technology and in a 120nm technology. I have also implemented some other test circuits, the multiple-input semi-floating-gate inverter being the most important of these. The parallel and serial multiplier has been implemented using commercially available software and standard processes. They have been shown to function in simulations and in measurements.

5.1 Further Work

A simpler PCB could be designed to verify the logical function of the 120nm chip.

To further improve on the multiplier design several approaches could be made. Several other multiplying schemes exists. Implementing the multiplier using a better design than the array-multiplication would further improve on the multipliers performance. Also implementing the multiplier using other adders, like carry lookahead adders would benefit the multiplier.

The above mentioned improvements are fairly straight forward to implement. Implementing some of the other powersaving techniques mentioned such as reducing the swing of internal signals and parallelism would be very interesting to try to implement.

Also new multiple-valued circuit designs is published from time to time, implementing the designs mentioned here with other multiple-valued techniques than the semi-floating-gate would be interesting.

Appendix A

Software and instrument overview

The software used to produce the various hardware described through this thesis, and the production of this document:

- **Cadence 5.0** for design and simulation of the chip produced in the AMS 350nm process.
- **Cadence 4.4.6** for design and simulation of the chip produced in the ST Microelectronics 112 nm process.
- **Cadence Concept** for schematic design of the PCB.
- **Cadence Allegro** for layout design of the PCB.
- **LyX** for documentation.
- **JabRef** for generating bib-TEX database.
- **Illustrator CS** for drawing illustrations.
- **Matlab 7.0** for measurements and plotting.

To produce the hardware the following manufacturers were used:

- **AMS** for 350nm chip.
- **CMP** for 120nm chip.
- **Elprint** for PCB production of FR-4 four layer PCB.
- **Nor-Team** for modifications on the PCB.

To test the two chips the following instruments were used:

For the 350nm chip:

- **Hewlett Packard E3610A** DC power-supply.
- **Hewlett Packard E3614A** DC power-supply.
- **Hewlett Packard 54503** oscilloscope.
- **TTi TGA1244** waveform generator.

For the 120nm Chip:

- **Agilent infiniium 54855A DSO** 6GHz/GSa/a Oscilloscope
- **Agilent E3631A** triple output DC power supply
- **Hewlett Packard 85024A** 300kHz-3GHz High Frequency Probe
- **Agilent 33220A** 20MHz Function/Arbitrary Waveform Generator
- **Fluke 189** multimeter

Appendix B

Pinout tables

Pinout table for the 120nm chip:

Description	Function	Pin number
Vdd for system	Vdd	77
GND for system	GND	78
Clock for system	Clk	79
Input circuit 1	In	81
Input circuit 2	In	82
Input circuit 3	In	83
Output Circuit 1	Out	84
Output Circuit 2	Out	1
Output Circuit 3	Out	2
Output Circuit 4	Out	3
Multiple-valued signal out	Out	4
Output Circuit 4	In	5
Output Circuit 5	In	6
Output Circuit 6	In	7

Pinout table for the 350nm chip:

Description	Function	Pin number on socket
Ground for systems and pad ring	GND	B4
Vdd for systems and pad ring	Vdd	A4
Clock for system	Clk	D1
Clock inverted for system	Clk_inv	E3
In auto-zero circuit	In	D2
Out auto-zero circuit	Out	C1
Out serial multiplier	Out	E1
Input serial multiplier, memory Ctrl	In	E2
Input serial multiplier, memory Ctrl	In	F2
Input serial and parallel multiplier	In	F3
Input serial and parallel multiplier	In	G3
Input serial and parallel multiplier	In	G1
Input serial and parallel multiplier	In	G2
Input serial and parallel multiplier	In	F1
Input serial and parallel multiplier	In	H1
Input serial and parallel multiplier	In	H2
Input serial and parallel multiplier	In	J1
Output parallel multiplier	Out	L3
Output parallel multiplier	Out	K4
Output parallel multiplier	Out	L4
Output parallel multiplier	Out	J5
Output parallel multiplier	Out	K5
Output parallel multiplier	Out	L5
Output parallel multiplier	Out	K6
Output parallel multiplier	Out	J6
Output parallel multiplier	Out	J7
Output parallel multiplier	Out	L7
Output parallel multiplier	Out	K7
Output parallel multiplier	Out	L6
Input parallel multiplier	In	L8
Input parallel multiplier	In	K8
Input parallel multiplier	In	L9
Input parallel multiplier	In	K11
Input parallel multiplier	In	J11
Input parallel multiplier	In	H10
Input parallel multiplier	In	H11
Input parallel multiplier	In	F10
Input Multi-input-inverter with auto-zero	In	G10
Input Multi-input-inverter with auto-zero	In	G11

Input Multi-input-inverter with auto-zero	In	G9
Output Multi-input-inverter with auto-zero	Out	F9
Input Multi-input-inverter	In	F11
Input Multi-input-inverter	In	E11
Input Multi-input-inverter	In	E10
Output Multi-input-inverter	Out	E9
Memory cell In	In	A7
Memory cell Ctrl	In	C7
Memory cell Ctrl	In	C6
Memory cell Out	Out	A6

Appendix C

Layout of various circuits

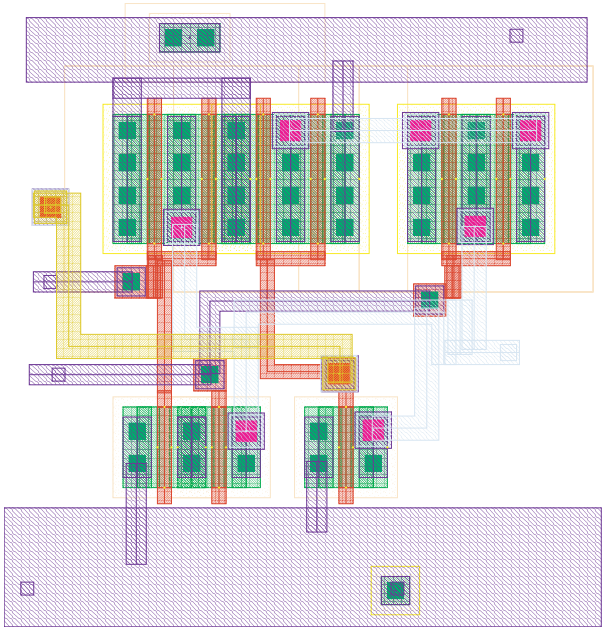

Figure C.1: Layout of the auto-zero circuit used in the 350nm chip.

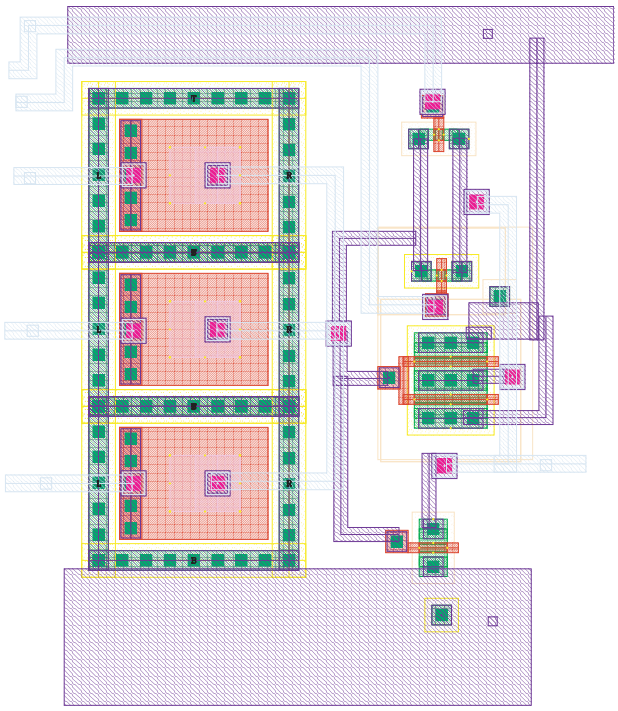

Figure C.2: Layout of the multiple-input inverter used on the 350nm chip

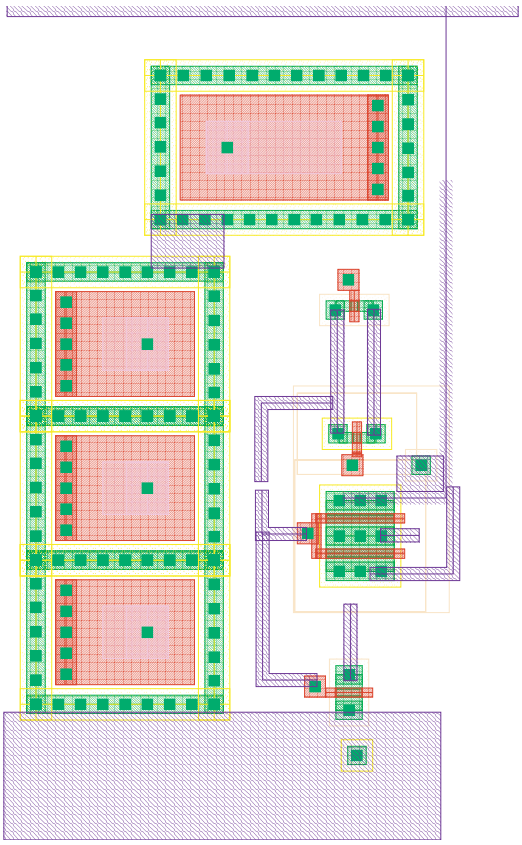

Figure C.3: The adder used on the 350nm chip

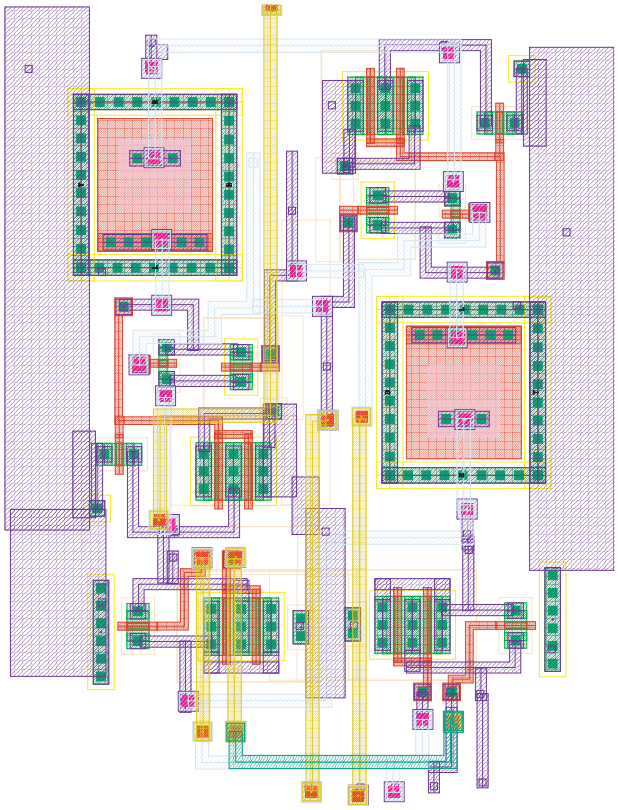

Figure C.4: The memory element for the serial multiplier on the 350nm chip

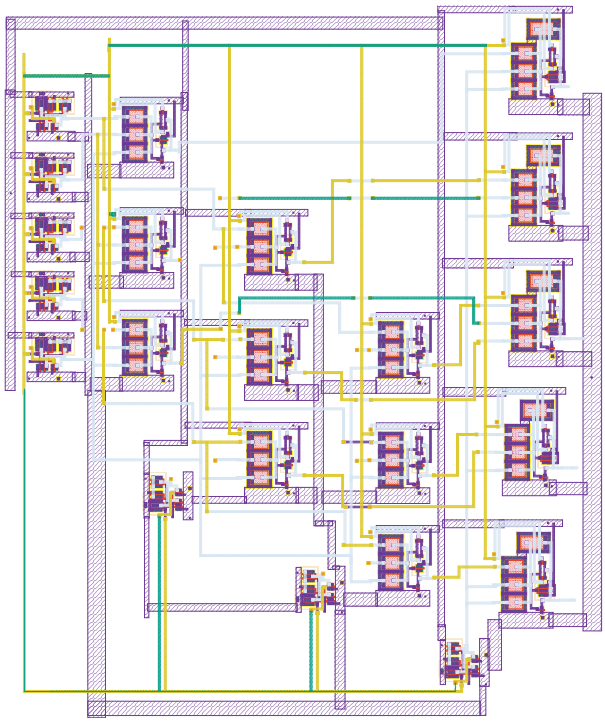

Figure C.5: The parallel multiplier on the 350 chip

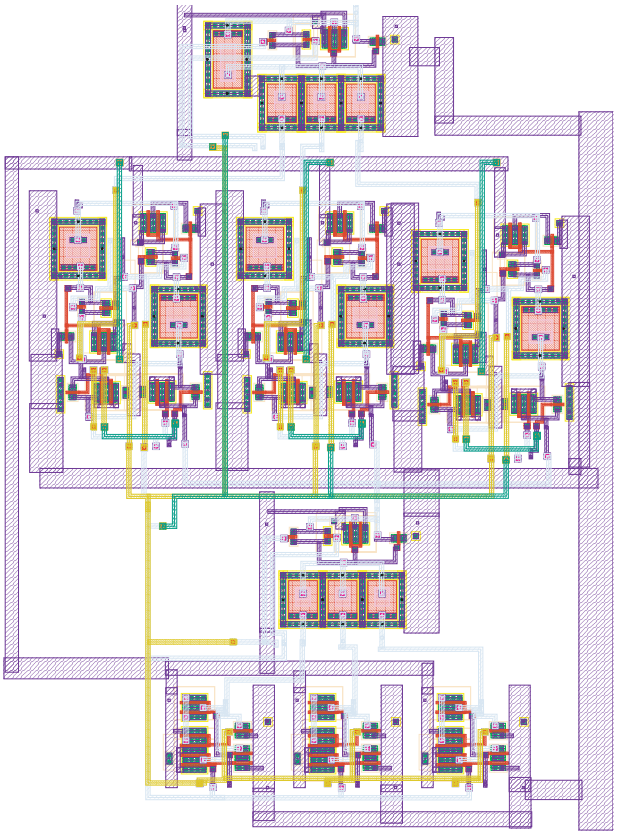

Figure C.6: The serial multiplier on the 350nm chip

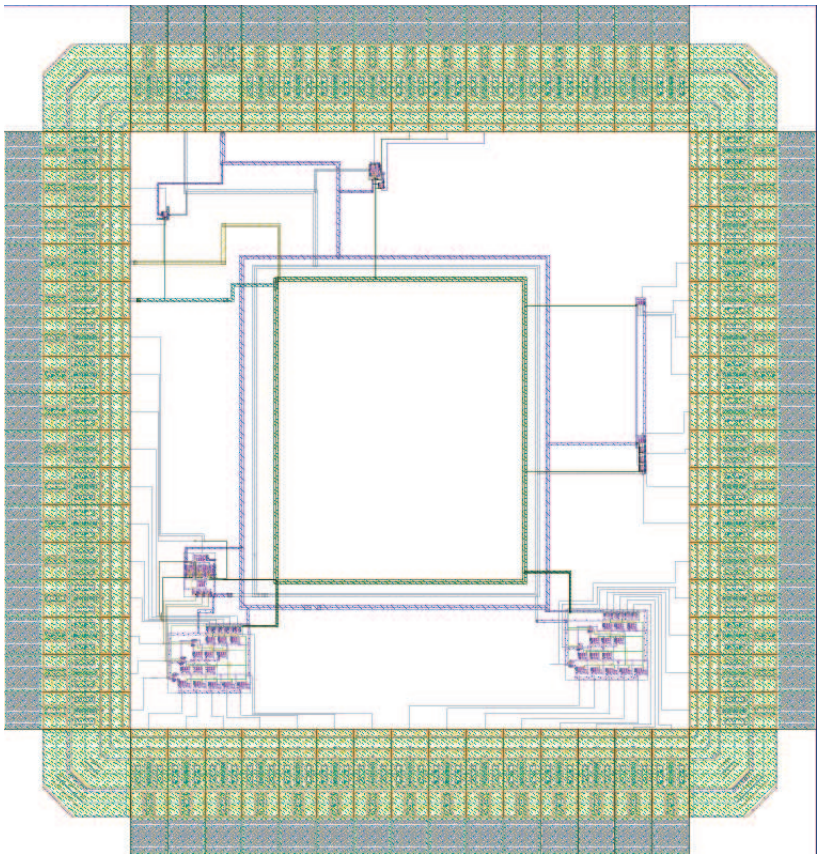

Figure C.7: The entire 350nm chip with pad-frame

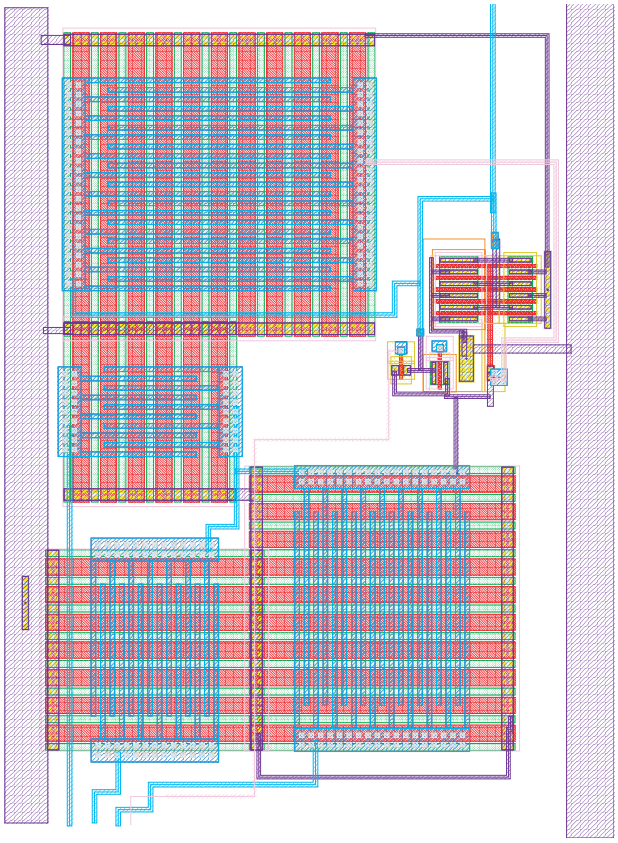

Figure C.8: One of the adders used in designs for the 112nm chip. Note the large capacitors to the left and in the bottom.

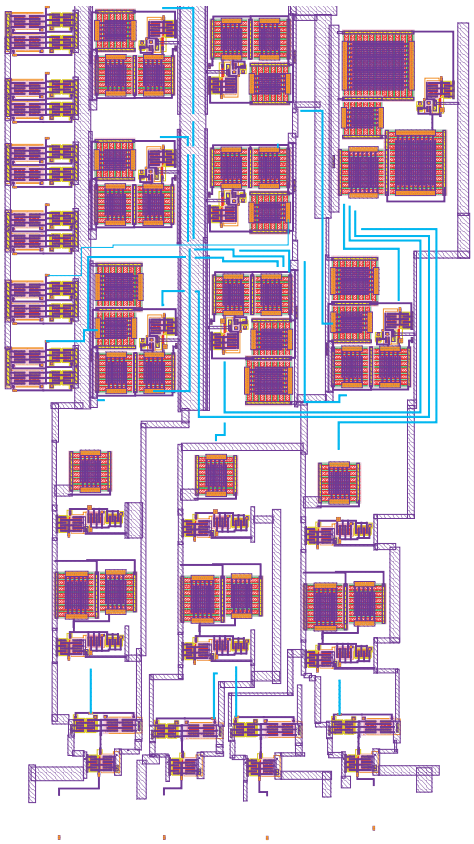

Figure C.9: Entire 112nm chip.

Appendix D

Scripts

An example of matlabscript used to generate input-signals for measurements.

```
%% Reset instruments
TTi1244A = TTi1244_DefaultName;
GPIB_Write('*CLS;*RST;',TTi1244A)
TTi1244B = '/dev/TTi1244B';
GPIB_Write('*CLS;*RST;',TTi1244B)

HP54503=HP54503_Defaultname;



% Innganger på kretsen

bunn = [-2048];
topp = [2047];
antpkt = 64*input('Antall 64x punkter\n');
f=4e3;

CLKinv = [];
klokke = [];
for j=1:antpkt/4,
  for i=1:antpkt/32,
    klokke =[klokke bunn];
    CLKinv = [CLKinv topp];
  end
  for i=1:antpkt/32,
```

```
    klokke = [klokke topp];
    CLKinv = [CLKinv bunn];
  end
end


Inn1 = [];
for j=1:antpkt/8,
  for i=1:antpkt/16,
    Inn1 =[Inn1 bunn];
  end
  for i=1:antpkt/16,
    Inn1 = [Inn1 topp];
  end
end


Inn2 = [];
for j=1:antpkt/16,
  for i=1:antpkt/8,
    Inn2 =[Inn2 bunn];
  end
    for i=1:antpkt/8,
    Inn2 = [Inn2 topp];
  end
end


Inn3 = [];
for j=1:antpkt/32,
  for i=1:antpkt/4,
    Inn3 =[Inn3 bunn];
  end
  for i=1:antpkt/4,
    Inn3 = [Inn3 topp];
  end
end


Inn4=[];
for j=1:antpkt/64,
  for i=1:antpkt/2,
    Inn4 =[Inn4 bunn];
  end
```

```
  for i=1:antpkt/2,
    Inn4 = [Inn4 topp];
  end
end


%figure(1);
%hold off;
%f=1e3;
%t=[0:1:length(Inn1)-1]/f;
%subplot(4,1,4), plot(t,Inn1); ylabel('Ch1=In1 (V)');
%subplot(4,1,3), plot(t,Inn2); ylabel('Ch2=In2 (V)');
%subplot(4,1,2), plot(t,CLKinv); ylabel('Ch3=CLKinv (V)');
%subplot(4,1,1), plot(t,klokke); ylabel('Ch4=klokke (V)');
%xlabel('time (s)');


% setup
GPIB_Write('EOI ON', HP54503);
GPIB_Write('DISPLAY:CONNECT ON', HP54503);
GPIB_Write('BNC PROBE', HP54503);

GPIB_Write('SETUPCH 1',TTi1244A);
GPIB_Write('SYNCOUT ON', TTi1244A);



TTi1244_LockStatus('OFF');

TTi1244_SetChannel(1);
TTi1244_ArbitraryChannelDelete('vin4');
TTi1244_ArbitraryBackDelete('vin4');
TTi1244_ArbitraryDef('vin4', length(Inn4), Inn4);
TTi1244_ArbitrarySetOutput('vin4');
TTi1244_SetTerm('OPEN');
TTi1244_LockMode('MASTER');
TTi1244_ArbitraryFrequency(f);
TTi1244_SetAmplitude(2);
TTi1244_DCoffset(1.0281);
TTi1244_ChannelEnable('ON', 1);

TTi1244_SetChannel(2);
TTi1244_ArbitraryChannelDelete('vin3');
TTi1244_ArbitraryBackDelete('vin3');
```

```
TTi1244_ArbitraryDef('vin3', length(Inn3), Inn3);
TTi1244_ArbitrarySetOutput('vin3');
TTi1244_SetTerm('OPEN');
TTi1244_LockMode('SLAVE');
TTi1244_ArbitraryFrequency(f);
TTi1244_SetAmplitude(2);
TTi1244_DCoffset(1.0281);
TTi1244_ChannelEnable('ON', 2);


TTi1244_LockStatus('ON');

TTi1244_LockStatus('OFF',TTi1244B);

TTi1244_SetChannel(1, TTi1244B);
TTi1244_ArbitraryChannelDelete('vin2', TTi1244B);
TTi1244_ArbitraryBackDelete('vin2', TTi1244B);
TTi1244_ArbitraryDef('vin2', length(Inn2), Inn2, TTi1244B);
TTi1244_ArbitrarySetOutput('vin2', TTi1244B);
TTi1244_SetTerm('OPEN', TTi1244B);
TTi1244_LockMode('MASTER', TTi1244B);
TTi1244_ArbitraryFrequency(f, TTi1244B);
TTi1244_SetAmplitude(2, TTi1244B);
TTi1244_DCoffset(1.0281, TTi1244B);
TTi1244_ChannelEnable('ON', 1, TTi1244B);


TTi1244_SetChannel(2, TTi1244B);
TTi1244_ArbitraryChannelDelete('vin1', TTi1244B);
TTi1244_ArbitraryBackDelete('vin1', TTi1244B);
TTi1244_ArbitraryDef('vin1', length(Inn1), Inn1, TTi1244B);
TTi1244_ArbitrarySetOutput('vin1', TTi1244B);
TTi1244_SetTerm('OPEN', TTi1244B);
TTi1244_LockMode('SLAVE', TTi1244B);
TTi1244_ArbitraryFrequency(f, TTi1244B);
TTi1244_SetAmplitude(2, TTi1244B);
TTi1244_DCoffset(1.0281, TTi1244B);
TTi1244_ChannelEnable('ON', 2, TTi1244B);

TTi1244_SetChannel(3, TTi1244B);
TTi1244_ArbitraryChannelDelete('klokke', TTi1244B);
TTi1244_ArbitraryBackDelete('klokke', TTi1244B);
TTi1244_ArbitraryDef('klokke', length(klokke), klokke, TTi1244B);
TTi1244_ArbitrarySetOutput('klokke', TTi1244B);
```

```
TTi1244_SetTerm('OPEN', TTi1244B);
TTi1244_LockMode('SLAVE', TTi1244B);
TTi1244_ArbitraryFrequency(f, TTi1244B);
TTi1244_SetAmplitude(2, TTi1244B);
TTi1244_DCoffset(1.0281, TTi1244B);
TTi1244_ChannelEnable('ON', 3, TTi1244B);


TTi1244_SetChannel(4, TTi1244B);
TTi1244_ArbitraryChannelDelete('CLKinv', TTi1244B);
TTi1244_ArbitraryBackDelete('CLKinv', TTi1244B);
TTi1244_ArbitraryDef('CLKinv', length(CLKinv), CLKinv, TTi1244B);
TTi1244_ArbitrarySetOutput('CLKinv', TTi1244B);
TTi1244_SetTerm('OPEN', TTi1244B);
TTi1244_LockMode('SLAVE', TTi1244B);
TTi1244_ArbitraryFrequency(f, TTi1244B);
TTi1244_SetAmplitude(2, TTi1244B);
TTi1244_DCoffset(1.0281, TTi1244B);
TTi1244_ChannelEnable('ON', 4, TTi1244B);


TTi1244_LockStatus('OFF',TTi1244B);



GPIB_Write('REFCLK MASTER',TTi1244A);
GPIB_Write('REFCLK SLAVE',TTi1244B);

TTi1244_LockStatus('ON',TTi1244B);



% Dump oscilloscope values
%  DumpSpes1;
%ScreenDumpSimple
input('Prepare scope for readout');
DumpScope;

figure(2);
hold off;
plot(channelX(1,:), channelY(1,:)); grid on;
% ! save chip1_anainv1inp_m1.mat
```

An example of matlabscript used to generate plots.

```
function createfigure(data1, data2, data3)
%CREATEFIGURE(DATA1,DATA2,DATA3)
%  DATA1:  vector of y data
%  DATA2:  vector of y data
%  DATA3:  vector of y data

%  Auto-generated by MATLAB on 02-May-2007 17:29:26

% Create figure
figure1 = figure('PaperSize',[20.98 29.68]);

% Create axes
axes1 = axes('Parent',figure1,...
    'XTickLabel',{'0   ','2','4','6','8','10','12','14','16','18','20'},...
    'OuterPosition',[0 0.6667 1 0.3333]);
box('on');
hold('all');

% Create plot
plot(data1,'DisplayName','data2(:,2)','Parent',axes1);

% Create axes
axes2 = axes('Parent',figure1,...
    'XTickLabel',{'0   ','2','4','6','8','10','12','14','16','18','20'},...
    'Position',[0.13 0.4093 0.775 0.2608]);
box('on');
hold('all');

% Create plot
plot(data2,'DisplayName','data2(:,4)','Parent',axes2);

% Create axes
axes3 = axes('Parent',figure1,...
    'XTickLabel',{'0   ','2','4','6','8','10','12','14','16','18','20'},...
    'Position',[0.1318 0.09818 0.775 0.2608]);
box('on');
hold('all');

% Create plot
plot(data3,'DisplayName','data2(:,6)','Parent',axes3);

% Create textbox
```

```
annotation(figure1,'textbox','String',{'Time (ms)'},'LineStyle','none',...
    'Position',[NaN NaN Inf 0.03466],...
    'FitHeightToText',...
    'on');

% Create textbox
annotation(figure1,'textbox','String',{'Clk(V)'},'LineStyle','none',...
    'Position',[NaN NaN Inf 0.03466],...
    'FitHeightToText',...
    'on');

% Create textbox
annotation(figure1,'textbox','String',{'In(V)'},'LineStyle','none',...
    'Position',[NaN NaN Inf 0.03466],...
    'FitHeightToText',...
    'on');

% Create textbox
annotation(figure1,'textbox','String',{'Out(V)'},'LineStyle','none',...
    'Position',[0.02361 0.848 0.1625 0.03466],...
    'FitHeightToText',...
    'on');
```

Bibliography

[1] Y. Berg, T. S. Lande, Øyvind Næss, and H. Gundersen, "Ultra-low-voltage floating-gate transconductance amplifiers," *IEEE Transactions on Circuits and Systems-II: Analog and Digital Signal Processing*, vol. 48, no. 1, pp. 37–44, 2001.

[2] O. Mirmotahari and Y. Berg, "A novel d-latch in multiple-valued semi-floating-gate recharge logic," *34th International Symposium onMultiple-Valued Logic*, 2004.

[3] D. Xiaowei, T. Hanyu, and M. Kameyama, "Quantum device model based super pass gate for multiple-valued digital systems," *25th International Symposium on Multiple-Valued Logic*, pp. 92 – 97, 1995.

[4] D. Khang and S. M. Sze, "A floating gate and its applikation to memory devices," *The Bell System technical journal*, vol. 46, no. 6, pp. 1288–1295, 1967.

[5] D. S. I. Thoidis, I. Karafyllidis, A. Thanailakis, and T. Stouraitis, "Design methodology of multiple-valued logic voltage-mode storage circuits," *Proceedings of the 1998 IEEE International Symposium on Circuits and Systems, 1998. ISCAS '98.*, vol. 2, pp. 125 – 128, 1998.

[6] K. C. Smith, "The prospects for multipvalued logic: A technology and applications view," *IEEE Transactions On Computers*, vol. C-30, no. 9, pp. 619–634, 1981.

[7] Y. Berg, Ø. Næss, S. Aunet, R. Jensen, and M. Høvin, "Novel floating-gate multiple-valued signal to binary signal converters for multiple-valued cmos logic," pp. 385–388, 2002.

[8] Y. Berg, S. Aunet, O. Mirmotahari, and M. Høvin, "Novel recharge semi-floating-gate cmos logic for multiple-valued systems," *Proceedings of the 2003 International Symposium on Circuits and Systems (ISCAS '03)*, vol. 5, pp. 193–196, 2003.

[9] T. S. Lande, D. T. Wisland, T. Soether, and Y. Berg, "Flogic-floating-gate logic for low-power operation," *Proceedings of the Third IEEE International Conference on Electronics, Circuits, and Systems, ICECS '96.*, vol. 2, pp. 1041 – 1044, 1996.

[10] T. Hozumi, N. Kamiura, Y. Hata, and K. Yamato, "Multiple-valued logic design using multiple-valued exor," *25th IEEE International Symposium on Multiple-Valued Logic*, vol. -, pp. 290–295, 1995.

[11] M. Inaba, K. Tanno, and O. Ishizuka, "Multi-valued flip-flop with neuron-cmos nmin circuits," *Proceedings of the 32nd IEEE International Symposium on Multiple-Valued Logic (ISMV'02)*, vol. -, pp. 282–, 2002.

[12] M. Inaba, K. Tanno, and O. Ishizuka, "Realization of nmax and nmin functions with multi-valued voltage comparators," *31st IEEE International Symposium on Multiple-Valued Logic, Proceedings.*, vol. -, pp. pp. 27 – 32, 22-24 May 2001.

[13] D. Etiemble and M. Israël, "Comparison of binary and multivalued ics according to vlsi criteria," *Computer*, vol. 21, no. 4, pp. 28–44, 1998.

[14] M. Kameyama, "Toward the age of beyond-binary electronics and systems.," *Proceedings of the Twentieth International Symposium on Multiple-Valued Logic*, pp. 162–166, 1990.

[15] S. Aunet, Y. Berg, Øyvind Næss, and T. Sæter, "Novel reconfigurable two-mosfet uv-programmable floating-gate circuits for carry, nand, nor or invert functions," 2001.

[16] O. Mirmotahari and Y. Berg, "A novel multiple-input multiple-valued semi-floating-gate latch," *Proceedings of the 33rd International Symposium on Multiple-Valued Logic (ISMVL 2003)*, pp. 227–230, 2003.

[17] G. Epstein, *Multiple-Valued Logic Design: An Introduction*. IOP Publishing Ltd, 1993.

[18] J. C. Muzio and T. C. Wesselkamper, *Multiple-Valued Switching Theory*. Adam Hilger Ltd, 1986.

[19] K. C. Smith, "Multiple valued logic: A tutorial and appreciation," *Computer*, vol. Volume 21, no. 4, pp. pp. 17 – 27, April 1988.

[20] D. Etiemble, "On the performance of multivalued integrated circuits: Past, present and future," *22nd IEEE International Symposium on Multiple-Valued Logic*, no. 0195-623X/92, pp. 156–164, 1992.

[21] D. Etiemble, "Why m-valued circuits are restricted to a small niche," *J. of Mult.-Valued Logic & Soft Computing*, vol. 10, pp. 109–123, 2003.

[22] L. M. Tolbert and T. G. Habetler, "Novel multilevel inverter carrier-based pwm method," *IEEE Transactions on Industry Applications*, vol. 35, no. 5, pp. pp. 1098 – 1107, Sept.-Oct. 1999.

[23] B. P. Wong, A. Mittal, Y. Cao, and G. Starr, *NANO-CMOS Circuit and Physical Design*. Jhon Wiley & Sons, Inc, 2005.

[24] K. Kotani, T. Shibata, M. Imai, and T. Ohmi, "Clocked-neuron-mos logic circuits employing auto-threshold-adjustment," *IEEE International Solid-State Circuits Conference, 1995. Digest of Technical Papers. 42nd ISSCC, 1995*, pp. 320 – 321, 388, 1995.

[25] S. Aunet, Y. Berg, and T. Sæter, "Real-time reconfigurable linear threshold elements implemented in floating-gate cmos," *IEEE Transactions on Neural Networks*, 2003.

[26] A. P. Chandrakasan, S. Sheng, and R. W. Brodersen, "Low-power cmos digital design," *IEEE Journal of Solid-State Circuits*, vol. 27, pp. 473 – 484, April 1992.

[27] Y. Berg, N. O., and H. M., "Ultra low-voltage floating-gate transconductance amplifier," *The 2000 IEEE International Symposium on Circuits and Systems. Proceedings. ISCAS 2000 Geneva.*, vol. 3, pp. 347 – 350, 2000.

[28] M. R. Stan, "Cmos circuits with subvolt supply voltages," *IEEE Design & Test of Computers*, pp. 34–43, 2002.

[29] L. Wang and H. Selvaraj, "A scheduling and partitioning scheme for low power circuit operating at multiple voltages," *Proceedings of the Euromicro Symposium on Digital System Design*, 2003.

[30] J. Pangjun and S. S. Sapatnekar, "Low-power clock distribution using multiple voltages and reduced swings," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems,*, vol. 10, no. 3, pp. pp. 309 – 318, June 2002.

[31] V. Krishna, N. Ranganathan, and N. Vijaykrishnan, "Energy efficient datapath synthesis using dynamic frequency clocking and multiple voltages," *Twelfth International Conference On VLSI Design*, vol. -, pp. 440 – 445, 1992.

[32] S. Hua and G. Qu, "Approaching the maximum energy saving on embedded systems with multiple voltages," *ICCAD'03*, vol. 1, pp. 26–29, 2003.

[33] I. M. Thoidisa, D. Soudris, I. Karafyllidis, and A. Thanailakis, "The design of low power multiple-valued logic encoder and decoder circuits," *The 6th IEEE International Conference on Electronics, Circuits and Systems. Proceedings of ICECS '99.*, vol. 3, pp. 1623 – 1626, 1999.

[34] H. Schmid, “Why the terms "current mode" and "voltage mode" neither divide nor qualify circuits,” *IEEE International Symposium on Circuits and Systems*, 1992.

[35] A. Abo and S. Mehta, “Cmos current mode adders.” http://kabuki.eecs.berkeley.edu/ abo/, 1993.

[36] J. Shen, K. Tanno, and O. Ishizuka, “Down literal circuit with neuron-mos transitors and its applications,” *Proceedings of the Twenty Ninth IEEE International Symposium on Multiple-Valued Logic*, 1999.

[37] M. Syuto, J. Shen, K. Tanno, and O. Ishizuka, “Multi-input variable-threshold circuits for multi-valued logic functions,” *IEEE International Symposium on Multiple-Valued Logic (ISMVL 2000)*, pp. 27–32, 2000.

[38] T. Shibata and T. Ohmi, “A functional mos transistor featuring gate-level weighted sum and threshold operations,” *IEEE Transactions on Electron Devices*, vol. 39, no. 6, pp. pp. 1444 – 1455, June 1992.

[39] Y. Berg and S. T. Lande, “Programmable floating-gate mos logic for low-power operation,” *IEEE International Symposium on Circuits and Systems, ISCAS ’97, Proceedings of 1997*, vol. 3, pp. 1792 – 1795, 1997.

[40] Y. Berg, D. T. Wisland, and 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, no. 7, pp. 930 – 936, 1999.

[41] Y. Berg, T. Lande, and O. Naess, “Programming floating-gate circuits with uv-activated conductances,” *Circuits and Systems II: Analog and Digital Signal Processing, IEEE Transactions on [see also Circuits and Systems II: Express Briefs, IEEE Transactions on]*, vol. 48, no. 1, pp. pp. 12 – 19, Jan 2001.

[42] Y. Berg, S. Aunet, O. Minnotahari, and M. Hovin, “Novel recharge semi-floating-gate cmos logic for multiple-valued systems,” *Proceedings of the 2003 International Symposium on Circuits and Systems, ISCAS ’03.*, vol. 5, pp. 193–196, 2003.

[43] Y. Berg, S. Aunet, Ø. Næss, and O. Mirmotahari, “Basic multiple-valued functions using recharge cmos logic,” *Proceedings of the 34th International Symposium on Multiple-Valued Logic (ISMVL’04)*, 2004.

[44] H. Gundersen and Y. Berg, “Max and min functions using multiple-valued recharged semi-floating gate circuits,” *Circuits and Systems, 2004. ISCAS ’04. Proceedings of the 2004 International Symposium on ,,* vol. 2, pp. pp. 857–860, 23-26 May 2004.

[45] G.-N. Byun, C.-U. Lee, S.-Y. Park, and H.-S. Kim, "A study on the ternary parallel circuit design with dcg properties based on the matrix equation," *30th IEEE International Symposium on Multiple-Valued Logic, (ISMVL 2000)*, vol. 1, pp. pp. 311 – 316, 23-25 May 2000.

[46] Y. Berg, S. Aunet, Ø. Næss, O. Mirmotahari, and M. Høvin, "Binary to multiple-valued recharge converter for multiple-valued cmos logic," *Proceedings of the 2003 International Symposium on Circuits and Systems*, 2003.

[47] Y. Berg, O. Naess, S. Aunet, J. Lomsdalen, and M. Hovin, "A novel floating-gate binary signal to multiple-valued signal converter for multiple-valued cmos logic," *9th International Conference on Electronics, Circuits and Systems*, vol. 2, pp. pp. 579 – 582, 15-18 Sept. 2002.

[48] Y. Berg, S. Aunet, Ø. Ness, J. Lomsdalen, J. Bush, and M. Høvin, "Binary input multiple-valued recharge cmos full-adders," *Proceedings of the European Conference on Circuit Theory and Design (ECCTD)*, September 2003.

[49] Y. Berg, Ø. Næss, S. Aunet, and M.Høvin, "A novel floating-gate multiple-valued signal to binary signal converter," pp. 575–578, 2002.

[50] I. M. T. O. T. K. N. K. H. K. K. N. K. Y. H. W. H. O. T. Murotani, T. Naritake, "A 4-level storage 4 gb dram," *Digest of Technical Papers. 44th ISSCC., 1997 IEEE International Solid-State Circuits Conference*, pp. 74–75, 433, 1997.

[51] T. Hanyu and T. Higuchi, "A floating-gate-mos-based multiple-valued associative memory," *Proceedings of the Twenty-First International Symposium on Multiple-Valued Logic*, vol. 1, pp. pp. 24 – 31, 26-29 May 1991.

[52] H. T., T. K., and K. M., "Multiple-valued floating-gate-mos pass logic and its application to logic-in-memory vlsi," *28th IEEE International Symposium on Multiple-Valued Logic, 1998. Proceedings*, pp. 270 – 275, 1998.

[53] T. Hanyu, "Challenge of a multiple-valued technology in recent deep-submicron vlsi," *31st IEEE International Symposium on Multiple-Valued Logic*, pp. 241 – 244, 2001.

[54] L. T.S., R. H., I. M., and B. Y., "An analog floating-gate memory in a standard digital technology," *Proceedings of Fifth International Conference on Microelectronics for Neural Networks, 1996.*, vol. 1, pp. 271 – 276, 1996.

[55] Y. Berg, S. Aunet, N. Ø, Ø. Hagen, and M. Høvin, "A novel floating-gate multiple-valued cmos full-adder," pp. 877–880, 2002.

[56] A. Satoh, N. Ooba, K. Takano, and E. D'Avignon, "High-speed mars hardware," *The Third Advanced Encryption Standard (AES) Candidate Conference*, 2000.

[57] D. V. Popel and R. Drechsler, "Efficient minimization of multiple-valued decision diagrams for incompletely specified functions.," *Proceedings of the 33nd IEEE International Symposium on Multiple-Valued Logic (ISMV'03)*, pp. 241–246, 2003.

List of Figures

List of Tables