Cell-Based Array for Deep Sub-Micron Technologies

by

James Boe-Kian Oey

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of
Master of Engineering in Electrical Engineering and Computer Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

August 2003

© James Boe-Kian Oey, MMIII. All rights reserved.

The author hereby grants to MIT permission to reproduce and distribute publicly paper and electronic copies of this thesis document in whole or in part.
Cell-Based Array for Deep Sub-Micron Technologies

by

James Boe-Kian Oey

Submitted to the Department of Electrical Engineering and Computer Science
on August 13, 2003, in partial fulfillment of the
requirements for the degree of
Master of Engineering in Electrical Engineering and Computer Science

Abstract

In this thesis I explore transistor topologies for high density cell-based arrays that allows for dense computation blocks, small memory cells, and strong signal drivers. This involves simulating different circuit types with HSPICE to determine ideal transistor sizes. Using Magic and the results of the HSPICE simulations, I explore transistor topologies with different ratios of nFets to pFets. An analysis on the technology shows important characteristics for digital systems and how they relate to the explored transistor topologies.

Thesis Supervisor: Christopher J. Terman
Title: Senior Lecturer
Acknowledgments

I would like to thank Christopher Terman for his guidance throughout this thesis project. He has helped me by teaching the material in both Computation Structures and Introduction to VLSI Systems, as well as providing support and insight throughout this project. I am very grateful for the chance to work with him. I have learned many lessons from our interactions.

It has been a pleasure meeting and spending time with friends at MIT. It has given me the chance to learn different things that MIT simple cannot teach. May we all be together in the future.

Lastly, I would like to thank my parents, Peter and Theresia, and my two brothers, Michael and Christopher. They are always there for me when I need help, and have always shown love and support throughout my life. I certainly could not have made it through the challenging parts of my life without my family. Thanks a lot for your support!
Contents

1 Introduction ......................................................... 13
  1.1 Problem Description ............................................ 13
  1.2 Prior Work ...................................................... 14
    1.2.1 Sea-of-Gates .............................................. 14
    1.2.2 Sea-of-Gates Structure for Digital and Analog Applications .. 15
    1.2.3 Cell-Based Array Patent ................................... 16
  1.3 General Overview ............................................... 18

2 Circuit Types and Spice Simulations .................. 21
  2.1 Spice Simulations .............................................. 22
  2.2 Ideal Circuits .................................................. 22
    2.2.1 Cell-Based Logic ......................................... 22
    2.2.2 Domino Logic .............................................. 23
    2.2.3 SRAM Cell ................................................. 28
    2.2.4 DRAM cell .................................................. 32
    2.2.5 Spice Decks ............................................... 39
  2.3 Conclusion ..................................................... 40
    2.3.1 Transistor Ratios ......................................... 40
    2.3.2 Transistor Sizes ......................................... 41

3 Diffusion Topology and Routing ......................... 43
  3.1 Magic .......................................................... 43
    3.1.1 Design Rules and Technology ............................. 43
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1.2</td>
<td>General Layout Issues</td>
<td>44</td>
</tr>
<tr>
<td>3.1.3</td>
<td>Sharing Polysilicon Among Multiple Diffusion</td>
<td>44</td>
</tr>
<tr>
<td>3.2</td>
<td>Layouts using equal number of p-type and n-type transistors</td>
<td>46</td>
</tr>
<tr>
<td>3.2.1</td>
<td>SRAM 6-T cell</td>
<td>46</td>
</tr>
<tr>
<td>3.2.2</td>
<td>Logic Cells</td>
<td>49</td>
</tr>
<tr>
<td>3.2.3</td>
<td>DRAM 3-T cell</td>
<td>50</td>
</tr>
<tr>
<td>3.2.4</td>
<td>DRAM 1-T cell</td>
<td>54</td>
</tr>
<tr>
<td>3.3</td>
<td>Layouts using twice the number of n-type devices than p-type devices</td>
<td>58</td>
</tr>
<tr>
<td>3.3.1</td>
<td>SRAM 6-T cell</td>
<td>58</td>
</tr>
<tr>
<td>3.3.2</td>
<td>DRAM 3-T cell</td>
<td>58</td>
</tr>
<tr>
<td>3.3.3</td>
<td>DRAM 1-T cell</td>
<td>59</td>
</tr>
<tr>
<td>3.4</td>
<td>Layouts using only one n-type and one p-type transistor</td>
<td>61</td>
</tr>
<tr>
<td>3.4.1</td>
<td>SRAM 6-T cell</td>
<td>61</td>
</tr>
<tr>
<td>3.5</td>
<td>Conclusion</td>
<td>61</td>
</tr>
<tr>
<td>3.5.1</td>
<td>Standard CMOS Cells</td>
<td>62</td>
</tr>
<tr>
<td>3.5.2</td>
<td>Memory Cells</td>
<td>63</td>
</tr>
<tr>
<td>3.5.3</td>
<td>Domino Logic</td>
<td>64</td>
</tr>
<tr>
<td>4</td>
<td>Analysis</td>
<td>65</td>
</tr>
<tr>
<td>4.1</td>
<td>Capacitances</td>
<td>66</td>
</tr>
<tr>
<td>4.1.1</td>
<td>Gate Capacitance</td>
<td>66</td>
</tr>
<tr>
<td>4.1.2</td>
<td>Drain Capacitance</td>
<td>66</td>
</tr>
<tr>
<td>4.1.3</td>
<td>Wire Capacitance</td>
<td>67</td>
</tr>
<tr>
<td>4.1.4</td>
<td>MAGIC Technology File</td>
<td>68</td>
</tr>
<tr>
<td>4.1.5</td>
<td>Experiments</td>
<td>69</td>
</tr>
<tr>
<td>4.2</td>
<td>Switching Characteristics</td>
<td>72</td>
</tr>
<tr>
<td>4.2.1</td>
<td>Experiments</td>
<td>73</td>
</tr>
<tr>
<td>4.3</td>
<td>Power Dissipation</td>
<td>77</td>
</tr>
<tr>
<td>4.3.1</td>
<td>Static Dissipation</td>
<td>77</td>
</tr>
<tr>
<td>4.3.2</td>
<td>Dynamic Dissipation</td>
<td>78</td>
</tr>
</tbody>
</table>
# List of Figures

1-1  Sample NOR gate layout using Ocean: Sea of Gates System. .......... 14
1-2  Exemplary cell arrangement in cell-based array patent. ............... 16

2-1  Transistor circuit diagram for the nand2 gate. .......................... 23
2-2  Circuit diagram for a dual-rail XOR domino gate. ........................ 24
2-3  Gate-level diagram for a full-adder circuit. .............................. 25
2-4  Circuit diagram for a domino full-adder circuit. ........................ 25
2-5  A, B, and Carry inputs for domino full-adder ............................ 27
2-6  Clock, and Carry Out, Sum (dotted) outputs for domino full-adder .... 28
2-7  The SRAM block includes the 6-T cell, a sense amplifier and write drivers. ................................................................. 30
2-8  The SRAM transistor chain during a read consists of the load nFet, the access nFet and the inverter nFet. ................................. 30
2-9  Word input, Write input, and Bit lines for SRAM cell ..................... 31
2-10 The 3-Transistor DRAM cell consists of a read access, write access and a storage transistor. .................................................. 33
2-11 The 1-T DRAM system requires the cells, some dummy cells and a cross-coupled sense amplifier. .......................................... 35
2-12 Left bit-line and right bit-line(dotted) for 1-Transistor DRAM cell on a 0 and a 1 write/read ...................................................... 37

3-1  Diffusion pattern for transistors that share polysilicon gates .......... 45
3-2  6-T SRAM layout using only metal 1 ........................................... 47
3-3  6-T SRAM layout using two metals ............................................. 48
Chapter 1

Introduction

1.1 Problem Description

Today, both industry and educational institutions throughout the world use gate-array technologies to design digital systems. While the design time is minimal compared to companies who choose full-custom designs, gate-arrays have several drawbacks, often causing an inability to meet circuit timing and area specifications. Because typically gate-arrays only have a limited selection of transistors, circuits cannot benefit from having a selection of transistor sizes to create small computation circuits and larger drive circuits. Generally, gate-arrays have large transistors for all circuits; for many systems, using larger transistors will cause blocks to run slower than circuits with a wide selection of transistor sizes. In addition, using large transistors will consume much silicon, costing the designer both area on the silicon that could be used for other system blocks and money for the actual cost of the silicon.

Digital designers are seeking for a median between the two extremes, allowing for a fair balance between speed, area, and design time. In this thesis we discuss diffusion topologies for transistors that allow for efficient layout and routing of common cells used in digital design today.
1.2 Prior Work

1.2.1 Sea-of-Gates

Sea-of-gates is a specific type of approach to generalizing a digital design where a chip contains a continuous array of nFets and pFets. Unlike full-custom layout designs, sea-of-gates only require the wiring of the transistors to complete a design. However, with this limitation, the technology is still very generic. The approach to such a design begins with the formulation of a single block that consists of a small number of transistors, and then replicate it many times on the same piece of silicon.

Ocean: the Sea-of-Gates Design System is a working CAD tool that implements a sea-of-gates example; the fishbone layout, shown in Figure 1-1, is one of the simplest topologies available in a sea-of-gates system. It is a standard pattern with alternating rows of nFets and pFets.

![Figure 1-1: Sample NOR gate layout using Ocean: Sea of Gates System.](image)

The DIMES production facility in the Netherlands is capable of producing chips developed using the fishbone master image. Two other images, octagon and gatearray, serve as examples for the different features of the Ocean system.
Tools

Ocean: the Sea-of Gates Design System uses a number of different tools in its design process, listed in [3]. The next few paragraphs details a few of these tools.

**Seadali** is the main CAD tool used to generate layouts. This tool includes the fish, madonna, and trout programs. The first tool, fish, takes layouts and purifies them, creating a newer enhanced cell. Madonna is a partitioning based sea-of-gates placer that uses a netlist the designer provides. Although this program does not always generate the optimal layout, it can be a useful tool, especially in large systems. Trout is the sea-of-gates router. After the designer lays out and labels some cells, trout will route these cells together based on the SLS file linked to the project. If the designed feeds the router enough information, trout will respond successfully if there is a route available for the particular cell layout.

After generating a layout, it is necessary to perform an extraction of the layout in order to simulate the behavior of the circuit. Ghoti is a script that purifies sea-of-gates extracted circuits. It takes a layout and removes the unnecessary transistors from the circuits. Space performs the actual extraction and generates a file readable by simeye, the simulation tool provided by Ocean.

### 1.2.2 Sea-of-Gates Structure for Digital and Analog Applications

Phillipe Duchene and Michel Declercq, in [1], describe a generalized sea-of-gates structure that is suitable for a number of digital and analog applications. The paper gives a general description of the sea-of-gates concept, including the current limitations of a sea-of-gates implementation. The basic template for their structure includes 2 nFets and 1 pFet per transistor column.

In their new structure, they argue that a *random logic design* reduces the number of routing channels, since cells from a library would not be specific to the intended application. They also argue that optimization of regular circuits such as RAMs, ROMs or PLAs is possible. The article further describes the key features of their
design including their reasoning for a 2:1 nFet to pFet transistor ratio, as well as their transistor and grid spacing sizing.

The last sections of the article present the results of their experiments. It compares circuit density and electrical performance of different types of circuits. They compare between sea-of-gates, gate array, standard cell and full-custom designs. Full-custom layouts excluded, sea-of-gates consume the least amount of circuit area compared to other design styles; on the average, a typical circuit consumes twice the area as a symbolically generated full-custom layout.

### 1.2.3 Cell-Based Array Patent

![Exemplary cell arrangement in cell-based array patent.](image)

Iranmanesh et al. received a patent in January 2001 titled *Cell based array having compute drive ratios of N:1* [2]. This patent approaches a unique topology allowing for compact computation circuitry yet providing larger transistor sizes for the high
drive needed elsewhere.

The patent summarizes the invention by describing and providing diagrams for the topology. It features groups of small transistors that allows for computation, while providing a pairs of larger transistors necessary to drive higher capacitive loads. An analysis of the system describes how to improve performance, and explains the power distribution through the layout.

Figure 1-2 shows the cell arrangement for this particular invention. Notice that transistors are grouped into columns, and in each column of substrate, there are 4 pFet transistors are 4 nFet transistors. Looking at the last column of transistors to the right of the $M_4\ VDD$ node, we see from top to bottom and left to right: 2 large pFet transistors, 2 small pFet transistors, 2 large nFet transistors, and 2 small nFet transistors. The claim of this patent is that the smaller transistors are useful for compute transistors, and the larger ones are used for drive transistors.

For example, suppose a digital designer needed to design a 16-bit adder circuit. In this particular case, the computation circuit would be the different full-adders connected together. The nodes that represent the solution of the 16-bit adder would require drive transistors to be able to drive the next stage of the system. For the full-adders in these circuits, the smaller pFets and nFets are more suitable than the larger ones since these transistors will operate the fastest. Additionally, since these are in the middle of a computation circuit, there is no need to drive a huge capacitive load, so a small transistor would be the ideal size. However, if the system uses the same transistors for the output drive, then the adder would not be able to drive the next stage, and the system would suffer a large time delay.

There are a total of 8 transistors in each “transistor column.” In Figure 1-2, there are 4 transistor columns shown. Having these columns separate does have some benefits, although there are some disadvantages as well. The benefit is that there is an option of having current flow or not between the diffusions across the columns. However, the disadvantage is that if three transistors need to be in series, an additional transistor can fit in between the two columns; in this diffusion layout, multiple transistors in series can be less efficient than other layouts.
1.3 General Overview

The goal of this project is to design a diffusion topology that allows for both dense computation cells and dense memory cells. This topology should demonstrate increased efficiency in speed and area over a standard gate-array layout. The main steps towards discovering this optimal topology are \textit{HSPICE} simulations and \textit{Magic} layout and extraction. After a number of iterations, a final circuit extraction can provide data for a circuit analysis to find limitations on the diffusion technology tested.

The first stage towards designing an efficient diffusion topology is to consider the different type of cells that a digital designer would need in his toolkit. From simple NAND and NOR gates to more complicated domino gates, \textit{SPICE} simulations will verify correct sizing of the transistors to make ideal circuits when an unlimited number of transistors sizes are available. In addition to static and domino gates, it is important to look at six-transistor SRAM cells, three-transistor DRAM cells and one-transistor DRAM cells. Since most digital systems need some sort of memory elements, testing these cells will allow nearly full coverage of the type of cells a designer would need to design any arbitrary system.

The second stage begins to look at specific transistor topologies. To consider a number of different possible layouts, \textit{Magic} allows quick editing of the diffusion and polysilicon. The main steps in this stage is to lay a specific topology, test a few circuits and measure the results. These results would be either some timing constraints or space constraints. Space constraints are easily measurable by the area a sample circuit consumes in \textit{Magic}. Timing constraints on the other hand, require circuit extraction and a spice simulation. Therefore, testing on this second stage may require several iterations of \textit{SPICE} simulations.

The third stage analyzes a specific transistor topology and focuses on specific characteristics of different circuits. From the previous stages, this thesis already considers timing issues and efficient use of silicon area. However, in this analysis stage, more detail is necessary: for timing, this stages considers gate delay as well
as memory reads and writes. These delays are important, since they are the major factors that limits a system from operating at a higher frequency. To be more specific regarding efficient use of silicon area, gates/mm$^2$ is a measurable quantity that can compare and contrast different transistor topologies. However, there are other factors, such as different types of capacitances, power, and leakage current. These factors are all key issues towards designing a digital system.
Chapter 2

Circuit Types and Spice Simulations

This chapter presents a number of circuit types important for designing an arbitrary digital system. It outlines some of the advantages and disadvantages for certain types of circuits. *SPICE* simulations on these idealized circuits will provide a set of useful timing criteria. Iterations on different transistor sizes will help to choose useful transistor sizes based on comparisons to the best possible circuits given the technology system uses. These types of circuits will be referred to as “ideal” for the remainder of this chapter.

Each type of circuit has an “ideal” transistor configuration, as well as appropriate sizes for each transistor. Of course, there is a minimal number of transistors necessary for each circuit. However, more importantly, timing parameters are important to consider, since a large part of designing an ideal circuit is to optimize timing constraints. Balanced rise and fall times are best for gates; however, it is very important to consider shorter times. For memory circuits, transistor sizes are important because slight variations in size can make a large difference in voltage levels, and therefore, the amount of time necessary to perform a read operation.
2.1 Spice Simulations

Since the final diffusion topology should be general, the next step is to encode each circuit into SPICE to be able to determine the best set of transistor sizes to build the circuits described above. Though it will be impossible to replicate the circuits due to simulation limitations, the transistor sizes should be a compromise between all the different types of circuits so that no circuit is much more efficient as any other. Such a condition would lead to bottlenecks; a system with highly efficient computation circuits but weak memory circuits could result in a system that spends too much time performing memory accesses.

The best approach for determining transistor sizes is to perform an analysis on the different types of circuits. Dense computation circuits will require no large transistors, while memories do require some large transistors for the reads and writes. One other factor is that two medium-sized transistors can be equivalent to a larger transistor. Therefore, circuits with larger drivers can make use of medium-sized transistors; this allows some sort of flexibility.

A more detailed description on the SPICE file format is provided at the end of the chapter, and SPICE files used in this chapter are provided in Appendix A.

2.2 Ideal Circuits

A varied selection of CMOS circuits will allow for a generalized comparision to select the best transistor size set. The circuit types will include cell-based logic, domino logic, and several types of memory cells and sense amplifiers.

2.2.1 Cell-Based Logic

Cell-based logic is composed of several main types of logic: simple gates, synthesis gates, and drive inverters. Simple gates (inv, mux2, nand2, nand3, nand4, nor2, nor3, nor4, and xor2) are gates used by a synthesizer as well as a person to build circuits. Synthesis gates (aoi21, aoi22, aoi211, oai21, oai22, oai211) are larger pieces
of logic that a synthesizer would use to piece together logic equations. Inverters come in many different widths; the larger ones are drive inverters intended to drive large capacitances. Lastly, there are many types of special-purpose circuits such as multiple versions of d-flipflops and d-latches to use in different parts of a design.

![Image of transistor circuit diagram for the nand2 gate.](image)

Figure 2-1: Transistor circuit diagram for the nand2 gate.

For most elements in this library, the amount of current flowing through the pull-up and pull-down network determines each transistor size. For example, assuming that the electrons have a mobility twice that of holes, the nFets in Figure 2-1 should be the same size as the pFets. This is because the current must flow through two nFets when the circuit is pulled down; while at the minimum, current will flow through a single pFet. This will balance the pull-down and pull-up delay time.

2.2.2 Domino Logic

Domino logic works differently than standard logic because it relies on a clock to precharge and evaluate a cascaded set of dynamic logic blocks\[7\. The name “Domino Logic” is an analogy to falling dominos; when dominos are stacked in a row, knocking one domino down causes other dominos to fall. A gate that uses domino logic consists of a single pFet and the logic encoded into the nFet. When the clock is low, the pFet precharges the output node. When the clock is high, the evaluate nFet enables the pulldown path, allowing the nMOS transistors to conditionally discharge the output.
node.

Figure 2-2 demonstrates a simple dual-rail XOR domino gate. As mentioned, when \( CLK \) is high, the output node charges. When \( CLK \) is in the low phase, a pull-down path is available as long as \( A \cdot \overline{B} \) or \( \overline{A} \cdot B \) is true for the XOR, and the opposite for the \( \overline{XOR} \).

![Circuit diagram for a dual-rail XOR domino gate.](image)

Because domino circuits precharge and conditionally discharges the output node, it is necessary to have inputs rise monotonically during the evaluation phase. If an operation requires a non-monotonic function such as XOR, it will be necessary to use dual-rail domino logic. This way, a signal and its complement will stabilize and remain constant before the evaluation phase begins.

Figure 2-3 demonstrates a gate-level implementation of a full-adder. It consists of two XOR gates, two AND gates, and an OR gate. The XOR of the three inputs, \( A \), \( B \), and \( C_i \) generates the \( \text{Sum} \) output. The \( C_{out} \) is determined by a majority circuit based on the three inputs.

Without using dual-rail XORs, the \( \overline{X} \) input, shown in Figure 2-3, falls during evaluation. Therefore, it is necessary to implement both the \( X \) and \( \overline{X} \) using a dual-rail XOR as the first XOR to properly implement the full-adder. The circuit uses only positive logic, which is important, since each stage requires a buffer inverter output. It includes two XOR gates, a XNOR gate, two AND gates and an OR gate. The
full-adder takes advantage of dual-rail domino logic, since the second XOR requires both $A \oplus B$ and $\overline{A} \oplus B$ to implement the sum.

Figure 2-3: Gate-level diagram for a full-adder circuit.

Figure 2-4: Circuit diagram for a domino full-adder circuit.
Domino logic is advantageous over static logic because the circuits take up less area, and parasitic capacitances are smaller, resulting in faster gates because of the reduction in self-loading. Since timing constraints can focus specifically on the falling edges of output nodes, an output inverter can be designed to detect a falling transition earlier than cell-based logic. Because of this, glitch free operation is possible by design. In addition, the elimination of the pullup short-circuit current saves time and energy.

Figure 2-5 shows the input waveforms for the domino full-adder circuit shown in Figure 2-4. This demonstrates a number of test patters to show the functionality of the circuit. The three waveforms show the \( A, B, C_{in} \) inputs.

Figure 2-6 shows the relationship between the clock signal in (a) and the output signals \( C_{out} \) and \( S \) in (b). On the low phase of the clock, both output signals are set to low. Next, based on the logic's functionality, each output signal is conditionally set to high. After the first discharge that occurs at about \( t = 1.0 \) ns, the values of \( C_{out} \) in the next 6 clock cycles are 1, 1, 0, 1, 0, and 1. For \( S \), the output values are 0, 0, 0, 1, 1, and 1.
Figure 2-5: A, B, and Carry inputs for domino full-adder
2.2.3 SRAM Cell

Since memories often occupy large areas in a digital layout, SRAM cells are important in the consideration of a diffusion topology. The standard six-transistor SRAM cell stores a single bit of memory using cross-coupled inverters. While this loop is stable, it is easy to change the value by driving one of the nodes with a strong transistor. While one of the major issues is the size of the actual cell, the write driver and the sense amplifier are also important modules in the SRAM system. The two functions necessary to consider are the write and read cycle.

The SRAM undergoes a write cycle when the write signal is asserted. The write and write-data transistors will drive bit and $\overline{bit}$ to $V_{SS}$ and $V_{DD} - V_t$, depending on
the value of *write-data*. Since the write transistors will be larger than the transistors in the cross-coupled inverter, the write transistors will overpower the transistors in the loop, causing the memory cell to flip its state.

The SRAM’s read cycle occurs when the word line is asserted. Since the bit lines need to be either charged or discharged to attain a certain voltage, the bit-line pullups transistors server to provide this charge. These transistors can be permanently on, or turned on when reading or writing. Using a precharge, rather than keeping the transistors on will save power, since the circuit can be shut off when not in use.

At this time, the two access transistors will cause either the *bit* or *\overline{bit}* lines to change depending on the state of the inverters. The voltage pulled down in either bit line is dependent on the ratios of nFets in the pulldown path; the precharge, access, and one of the cross-coupled transistor pulldown transistor.

While this voltage change is too small for a standard gate to properly detect, a sense amplifier can amplify the small change to cover the full range. This sense amplifier is a differential amplifier that amplifies slight diverging changes in the *bit* and *\overline{bit}* line.
Figure 2-7: The SRAM block includes the 6-T cell, a sense amplifier and write drivers.

Figure 2-8: The SRAM transistor chain during a read consists of the load nFet, the access nFet and the inverter nFet.
Figure 2-9: Word input, Write input, and Bit lines for SRAM cell
When choosing transistor sizes for an SRAM, there are some key points to consider. First, the six transistors in the memory cell should be minimized, since the majority of the memory area will be these cells. Second, the bit and \( \overline{\text{bit}} \) lines will need to change quickly to decrease access time, and have a larger change in voltage to speed up the sensing. The last point is that a read operation should not flip the selected cell.

Figure 2-9 shows the word and write inputs for the SRAM cell. It also shows the two bit lines that change based on the current value stored in memory and the state of the inputs. On the first write pulse, the system writes a 0 into the memory. The second assertion of the word signal results in the bit line value of approximately 1.5 V. The second half of the simulation performs a similar task, but it writes a 1 into the memory. As a result, rather than the bit line dropping to about 1.5 V, the \( \overline{\text{bit}} \) lines drops.

2.2.4 DRAM cell

Dynamic RAM cells have an advantage over static RAM cells because they take up much less space. There are a number of useful DRAM cell configurations, usually denoted by the number of transistors in the cell. This next section describe a 3-transistor and a 1-transistor DRAM cell.

DRAM cells have a number of disadvantages, including the necessity to refresh bits, and the susceptibility of the memory system to noise. This is because DRAM cells rely on charge storing in the system to store data; charge storing leads to charge leakage, and therefore, a restorative procedure is necessary to retain memory for long periods of time. While some systems use trench capacitors as storage elements, a sea-of-gates system such as this one will need to use gates of transistors to serve as the storage capacitors.
Three-transistor cell

The three-transistor DRAM cell consists of a write access transistor, a memory transistor, and a read access transistor. Like the SRAM cell, there are two functions to consider: the read and the write cycle.

The write access transistor enables the system to “permanently” trap charge to turn on the memory transistor. This charge remains there until the system writes a 0, that is, draining the charge, and turning the transistor off. In a true system, there is charge leakage, and the stored charge will not remain there forever; therefore, the system will have to rewrite the data to retain the charge. The leakage occurs across the gate, as well as through the access transistor. Because of charge leakage, it is beneficial to have the largest capacitance possible to charge and discharge. For this reason, using any unused transistors as a storage element would be beneficial, as demonstrated in the next chapter’s description of the three-transistor cell.

![Figure 2-10: The 3-Transistor DRAM cell consists of a read access, write access and a storage transistor.](image)

The read access transistor determines the value stored in the cell based on whether the memory transistor currently has a pull-down path. If the pull-down path exists, then the read line will read low; if the pull-down path does not exist, then the read line will read $V_{DD} - V_t$. Note that the read and write values are actually opposites; if the system stores a 1, then the pull-down path is available, and the system will read a 0. Of course, this can easily be corrected with an inverter, either on the write bit or the read bit.
PMOS Version of Three-transistor Cell

Even though a logic step towards designing efficient transistor topologies is to start with SPICE simulations, and then move to layouts, thinking ahead to the layout phase is important towards reaching a well-designed system. Since the goal of this thesis is to modify a gate-array layout, the final topologies will likely have an even ratio of nMOS and pMOS transistors. For this reason, a pMOS version of a three-transistor cell will allow a more efficient use of an area in the digital design designated for memory.

However, using a pMOS version could cause a number of capacitances issues, such as the Miller effect, if all the pFets replace all the nFets in the three-transistor cell. Because of this, the three-transistor cell now creates a pullup path rather than a pulldown path. There is now a pulldown path in the read bit rather than a pullup path. This way, the bitline predischarges prior to read, and a read may cause the line to charge. One point to consider in the pulldown path is the voltage readouts for the read bit. Since it would be beneficial to not use a sense amplifier, it is necessary to modify the pulldown path slightly to attain the proper voltages. Placing multiple nFets in series with each allows a readout of a 1 to be near 2.0 V.

Lastly, since all of the nFets will change to pFets, and read and write signals will need to be inverted as well, since pFets take in the opposite polarity to be active.

Sense Amplifier for the Three-transistor Cell

While it is true that using a sense amplifier will decrease the read cycle timing for a three-transistor cell, a sense amplifier is not necessary for the three-transistor cell since the read-bit line is nearly rail-to-rail. A sense amplifier is also not needed since reads are not destructive, since the memory is actually stored in the gate of the memory transistor. However, reads also do not recharge the memory capacitance; in the next section, the one-transistor cell demonstrates a restorative read-cycle.
One-transistor Cell

A one-transistor DRAM cell is even more efficient in area than a three-transistor cell[4]. However, a one-transistor DRAM cell is somewhat of a misnomer in this thesis because a gate-array does not have the capability of placing trench capacitors in DRAM cells. Since trench capacitors are unavailable in a gate array, each cell will use the gate capacitances on other available transistors.

Figure 2-11: The 1-T DRAM system requires the cells, some dummy cells and a cross-coupled sense amplifier

Before describing the details on writes and reads in this system, a brief overview of this system will aid to better describe the write and read operations. In this system, shown in figure 2-11, the system accesses half of the bits via the lbit line, and half of the bits via the rbit line. This aids to balance the bit lines’ capacitances, which is important during the read cycle. The read cycle focuses on two main components. First, the cross-coupled inverter in the sense amplifier aids to sense minute differences in voltages in the bit lines. The amplifier also restores the stored charge which normally would remain discharged since reads from one-transistor cells are destructive. Lastly, there is a half-capacitance dummy cell on both bit lines. The read cycle description will further describe the reason for this cell.
The writes to this system are very similar to the writes of other memory systems. The system drives the write-data line with either a 1 or a 0 through a transistor. When write is 1, this causes whichever bit is selected to either charge or discharge the storage capacitor. Of course, in this case, the storage capacitor is a number of gate capacitors.

The reads to this system are more complicated than in other systems. There are a number of stages to a single read. First, the lbit and rbit lines are precharged using the three nFets in the cross-coupled sense amplifier. While two of these actually charge the bit lines, the third nFet connecting the two bit lines aid to equalize the voltages between the two bit lines. This is very important since slight variations in the voltages will make a difference.

In the second stage to a read, the system activates one access transistor, and the dummy access transistor on the opposing side. In figure 2-11, this corresponds to bit₀ and DSᵣ. Because the precharge phase drained the \( \frac{C}{2} \) capacitor in the dummy cell, the rbit line will decrease by the capacitance ratio of the rbit line and \( \frac{C}{2} \). There are two cases to consider for the lbit line. First, if the storage capacitor in bit₀ has no charge in it, lbit will decrease by twice the voltage drop on the rbit line, since the bit₀ storage capacitor has twice the capacitance of the dummy cell capacitor. Second, if the storage capacitor in bit₀ has charge in it, lbit will not change. In either case, lbit and rbit are slightly different values based on the charge stored on the bit₀ capacitor.

In the last stage, the system enables CS and CSᵣ, which causes the bit lines to rail either to \( V_{SS} \) or \( V_{DD} \) based on the initial slight difference. If \( v_{lbit} < v_{rbit} \), then lbit will drain to \( V_{SS} \) and rbit will charge up to \( V_{DD} \). In either case, the amplifier works because the cross-coupled inverters will amplify the slight voltage difference into a readable voltage.
Figure 2-12: Left bit-line and right bit-line (dotted) for 1-Transistor DRAM cell on a 0 and a 1 write/read
In the two waveform screenshots, the screenshots display both a write and a read of a 0 and a 1. The precharges occur when both bit lines increase and reach a peak of about 2.1 V. After the first precharge, the system writes a 0 into bit\textsubscript{0} for the top waveforms. For the bottom waveforms, the system writes a 1 into bit\textsubscript{0}. After both bit lines are recharged, the right bit-line charge-shares with the dummy cell, and the voltage decreases to about 1.9 V. Next, the left bit line charge-shares with the bit\textsubscript{0} cell. Depending on the value stored in the bit cell, the bit-line either drops to about 1.5 V or remains at about 2.1 V.

It is clear after this state, one bit-line’s voltage is greater than the other, based on the value of bit\textsubscript{0}. From this point, using the cross-coupled inverters as the sense-amp causes the two bit lines to quickly diverge. The value read from memory is the value of the left bit-line. Since the left bit-line has the original value stored in bit\textsubscript{0}, this is a self-restoring read.

**Basic System Size.** It is important to consider how many bits per DRAM block should store. There are a number of issues that limit the range, and a number of reasons that aid in choosing the ideal number of bits. One major issue to consider is the capacitance ratio between the bit lines and the storage capacitances. Another issue is the actual memory density given a specific layout.

Since the read relies completely on the voltage difference on the two bit lines, it is important to pay close attention to the ratio between the bit line capacitance and the storage capacitance. This determine the change in voltage on a read of either a high and a low. To minimize the read delay, maximizing the $\Delta V$ allows for the fastest read on the sense amplifier.

Minimizing the capacitance ratio will maximize the $\Delta V$. Suppose the bit line precharges to $V_{\text{precharge}}$, and the capacitances are $C_{\text{bit}}$ and $C_{\text{storage}}$. The $\Delta V$ can be approximated as $V_{\text{precharge}} \cdot \frac{C_{\text{storage}}}{C_{\text{bit}}}$.

Since the drain capacitances and the gate capacitances of a CMOS transistor are roughly equivalent, this limits the number of bits on a specific bit line. The more bits on the line, the larger the capacitance ratio, and the smaller the $\Delta V$. On the other
hand, if the bit line only holds a few bits, then the memory density suffers greatly, since every two bit lines require write drivers and sense amplifiers to read the memory quickly.

### 2.2.5 Spice Decks

*SPICE* decks for each circuit type presented in the previous section allow for an analysis to determine both the ideal transistor sizes for a full-custom layout, as well as transistor sizes when the system is limited to only a few sizes. Each deck contains information that provide *HSPICE* with a circuit type, input signals suitable for simulation, and transistor sizing information.

The appendix contains all of the *HSPICE* files used for simulation as described in this chapter.

**Inputs and Outputs**

Since *SPICE* inputs are not subject to loading capacitances and other factors, driving all inputs to testing systems (except the clock input for domino) through an inverter ensures that the inputs to the actual system is a typical signal in the middle of a larger system.

Likewise, in a typical system, outputs connect to other blocks; for this reason, simulations need to account for this. Each output connects to four different inverters, which end up loading the outputs with a certain amount of capacitance.

**Transistor Sizing**

The first step in creating these *SPICE* decks is to create ideal circuits, since this will serve as a reference when choosing transistor sizes from a limited set. The second step is to parameterize the transistor sizes and to iterate through a number of possible transistor size sets. Simulation results will provide useful data for testing new transistor topologies, since these results will provide a decent approximation to the performance characteristics of each circuit.
With the `.alter` and `.param` statements, *HSPICE* is able to run similar circuit simulations using a single file. A parameter for each ideal transistor size allows a dynamic assignment of each of these transistor widths. For example, the following parameters are listed at the top of the domino 16-bit adder, `16-bit.sp`:

```
.param length=0.25u
.param N0.5=0.50u
.param N1.0=1.00u
.param N1.5=1.50u
.param N8.0=8.00u
.param P1.0=1.00u
.param P2.0=2.00u
.param P3.0=3.00u
.param P16.0=16.00u
```

As shown above, these parameters easily substitute into the *SPICE* deck, and `.alter` statements at the end can reassign these parameters to analyze different transistor sizes. After each `.alter` statement, a `.param` statement redefines the parameter. For example, `.param N0.5=1.0u` redefines all transistors with an ideal width of 0.5μ with a width of 1.0μ. Redefining each transistor size allows for tests of different possible transistor size sets.

### 2.3 Conclusion

The main circuit types this thesis analyzes include standard CMOS, domino logic, and memory circuits. There are a number of factors to analyze that are important for the next stage where this thesis develops an actual transistor topology. It is important to look at both transistor ratios as well as transistor sizes.

#### 2.3.1 Transistor Ratios

From the *SPICE* design files, it is apparent that an unequal ratio N to P transistors would probably be optimal for most circuit types, standard CMOS excluded. The
main cells that require an unequal ratio of transistors are the memory cells and the domino logic circuits. While the only type of circuit with a certain ratio required is the SRAM and DRAM 3-T cells, the other cells require a fairly high ratio.

Specifically, the SRAM 6-T cell require a 4:2 ratio of N:P. The DRAM 3-T require either only pFets or nFets, depending on the polarity of the cell. Lastly, the DRAM 1-T cell requires almost entirely nFets, and any available gates for storage capacitance.

The last type of circuit that would benefit from an unequal ratio of N:P are domino logic circuits. However, for these types of circuits, it is impossible to tell what the ideal ratio is; for larger cells, a larger ratio is necessary, but for smaller cells, a smaller ratio would be satisfactory.

Even though this chapter does provide a basic framework to determine transistor ratios, figuring out the transistor utilization for each circuit is difficult since there will be transistors used solely for circuit isolation; otherwise, current would run between blocks. This is addressed in the next chapter.

2.3.2 Transistor Sizes

From the different circuit analyses it is possible to figure out the necessary transistor sizes to cover most of the circuit types presented in this chapter.

For example, standard CMOS cells will use the minimum transistor sizes for computation. It is not necessary to use larger transistor sizes, except for the inverters, which may be used to drive larger nodes.

For memory cells, access transistors in general need to be larger, other transistors are variable. The cross-coupled inverter should be minimum-sized for quick switching, while the transistors in the DRAM 3-T cell should be slightly larger. Lastly, the DRAM 1-T require large transistors for the storage capacitances.

Domino-logic circuits generally do not require transistor sizes that are too large, although it is important to properly size the output inverter, which could use a larger
pFet to speed up the transition. Additionally, the pullup pFet and the evaluate nFet could need appropriate sizing analyses based on the evaluation network.
Chapter 3

Diffusion Topology and Routing

This chapter presents a topology design for a 0.25\(\mu\) process. It discusses the CAD tool Magic, and the technology files used to test the new topology designs. This chapter also tours the different layouts for the important circuit types, especially the memory cells and some domino-logic circuits.

3.1 Magic

The first stage in generating the ideal topology is to use Magic to test out several different transistor topologies. Magic is the ideal tool because it the tool gives full control over the diffusion, polysilicon and metal layout, while providing design rule check feedback to ensure that the project still follows the specifications for the current technology. Lastly, circuit extraction to a SPICE file allows for circuit simulation with the additional capacitances from the layout.

3.1.1 Design Rules and Technology

Magic’s layout tools scale on the order of a \(\frac{1}{2}\lambda\). MOSIS provides a technology called SCN5M.DEEP.12; this technology provides the design rules necessary for laying out a
transistor layout given the 0.25μ technology.

Several design rules limit the transistor spacing and the metal spacing. For this reason, these rules structure the final transistor layout possible. The next few paragraphs explain the key technology rules that defines the minimum possible grid size, 4.5λ.

**Polysilicon spacing.** The technology has a spacing rule of $\frac{3}{2}\lambda$ between polysilicon. However, a different rule supersedes this rule and actually defines the grid size; polysilicon to a polysilicon contact must be at least $2\lambda$ apart.

**Contact widths.** Most contacts need to be $\frac{5}{2}\lambda$ by $\frac{5}{2}\lambda$. Since the pm12c contact that connects polysilicon, metal 1, and metal 2 needs to be $\frac{3}{2}\lambda$ wide, this length added to the $2\lambda$ from the preceding paragraph defines the 4.5λ grid pattern.

### 3.1.2 General Layout Issues

There are several factors to consider regarding layout that affect the final layout topology. First are the limitations imposed by the design rules in the previous section. However, another factor is the limitations set forth by a computer program that would route a circuit. While it may be possible for a person to route a circuit much more efficiently by hand, using a computer to route the same circuit saves time. For this reason, it is important to consider the limitations of the routing program.

### 3.1.3 Sharing Polysilicon Among Multiple Diffusion

Sharing polysilicon among multiple diffusions may seem beneficial because a design may call for either a small or medium sized transistor. By sharing polysilicon, the system allows for a choice between the two sizes, while saving silicon in between the two transistors. Using separate polysilicon would require additional spacing because
Figure 3-1: Diffusion pattern for transistors that share polysilicon gates

of the technology's spacing constraints.

Figure 3-1 shows the diffusion pattern for four n-type transistors. There are several benefits to this pattern. First, the technology only requires $2\lambda$ between each of the diffusion areas. This allows the design to be more compact. Also, creating a third transistor size from the two given sizes is as easy as tying two diffusions in parallel.

However, large drawback prevents the use of this particular pattern. Suppose the bottom-left transistor is part of a circuit. Unless the circuit requires another n-type transistor in series with the used transistor, the bottom-right transistor is useless. As with all other gate-array topologies, grounding the bottom-right transistor will prevent any current from flowing into other parts of the circuit. Lastly, the two upper transistors are also useless because of the lower transistors.

By breaking the polysilicon between the diffusion, circuits now can utilize both the lower and upper transistors. Of course, this does not avoid unused transistors between cells, since these transistors still need to prevent current flow between cells.
3.2 Layouts using equal number of p-type and n-type transistors

3.2.1 SRAM 6-T cell

Designing a topology that will fit the SRAM 6-T cell is important because memory can occupy a huge percentage of silicon in a finalized layout. SRAM is also one of the most popular memory designs used. Since this cell may need to be replicated many times, saving a single transistor column or row can save a lot of silicon. Therefore, the next few paragraphs show results from several different layouts.

**Single metal.** In general, cells should use up only metal 1. This allows the remaining metals to connect cells to other cells. If a specific cell uses too many metal layers, then this imposes a constraint on the area above the cell for routing.

Figure 3-2 shows an 6-T SRAM cell that limits the internal cell routing to metal 1. The boxes show the location of the relevant parts of the circuit. The bottom boxes denote the access nFets, and the top boxes show the four transistors used for the cross-coupled inverters. Actually, the cell uses a little metal 2, but only for the word lines; when building the actual SRAM, the words will connect together using metal 2 anyway. In this case the cell occupies seven columns of transistors. This seems very inefficient, since there are a total of 28 transistors in the seven columns.

**Two metals.** The next case shows the same circuit laid out on the same transistor topology, except it uses two metals. Since this circuit uses a second metal, it limits the number of routing metals. However, more likely than not, this limitation is justifiable since the using two metals will cut the total SRAM area by a factor of $\frac{7}{4}$. In addition, the additional metal 2 that optimizes the area does not conflict with the word line routing; therefore, there is little loss of metal.
Figure 3-2: 6-T SRAM layout using only metal 1
Figure 3-3: 6-T SRAM layout using two metals
3.2.2 Logic Cells

Full-Adder Domino Cell

The full-adder cell serves as a generic computation cell, consisting of 5 gates; two XORs, two ANDs, and an OR gate. Figure 3-4 shows the layout of the full-adder cell, separated into boxes for each domino-gate. The first gate on the left is a dual-rail XOR gate. The remaining 4 gates are laid side-by-side, and are labelled appropriately.

Figure 3-5 shows the dual-rail XOR gate as previously mentioned, but in its entirety. Since this is a domino implementation, the utilization ratio between nFets and pFets is very high; about half of the nFets are used for a gate that does not follow a special pattern. Of all the pFets in this array, only two of them are utilized, for the pull-ups necessary for domino logic. The rest of the area serves as routing paths for the signals to connect the dual-rail XOR to the second XOR and one of the AND gates.

![Figure 3-4: Layout of full-adder block](image)

The domino logic full-adder cell serves as a basic logic cell as a test for the 2:2 nFet:pFet layout pattern. A normal CMOS full-adder cell will have a very similar layout, except that the pMOS section would be a complement of the nMOS section, and that the cell would either have to occupy slightly more area to accommodate for the pFet routing, or the cell would need to utilize more area.
Figure 3-6 displays simulation results using SPICE after extraction of the layout for the domino full-adder. Comparing these results to the simulation in the previous chapter, it is clear that the signals are suffering from additional capacitances. The $C_{out}$ signal only peaks for a short time at $t = 2$ ns; this indicates that the system is approaching the maximum operable clock frequency.

### 3.2.3 DRAM 3-T cell

Like the SRAM cell, each DRAM cell occupies three transistor columns. However, as mentioned in the previous chapter, using both the nMOS and pMOS versions of the SRAM cell doubles the memory density.

Figure 3-7 shows a DRAM layout with an nMOS version of the 3-T DRAM cell.
Figure 3-5: Layout of dual-rail XOR gate
Figure 3-6: Clock, and Carry Out, Sum (dotted) outputs for domino full-adder (layout version)
Figure 3-7: 3-T DRAM layout
3.2.4 DRAM 1-T cell

When using gate arrays, the 1-transistor cell is somewhat of a misnomer. In reality, each cell contains either four or five transistors that are used solely as a storage capacitance. This storage capacitance is accessible through a single n-type transistor. To compare memory density to other cells, the DRAM 1-T cell density is two bits for every three gate-array columns.

The 1-T DRAM bit-slice is pictured in Figure 3-8; it is accurate to say that this memory cell density is two bits per three columns; however, there are a few points to consider. Since it is essential to increase the storage capacitance in order to decrease the read time, the system must use all unused gates in the memory array. After an initial assignment of a memory storage bit per column, one out of every three columns serve to separate the access transistors. However, in the column, three of the transistors in the gate array are unused. To increase the capacitance for the storage elements, assigning two of these transistors to the storage element to the left, and one of the transistors to the right allows an optimal use of the storage capacitors in the array. Of course, the total storage capacitance in each bit varies, since there are a different number of capacitors per bit; however, balancing the unused n-type and the smaller p-type transistor with the larger p-type allows the capacitances to be as close as possible. This way, between the two different capacitance measurements, the system can have a measured read cycle period.

It is just as important to also evaluate the area needed for the sense-amp and the write drivers for each 32-bit block for the 1-T DRAM cell. Figure 3-9 shows the entire block, divided into the three main parts: the memory grids, the sense-amp, and the write drivers.

Figures 3-10 and 3-11 show the 1-T DRAM restorative sense-amp, and the write driver, respectively. Of course, the sense-amp is necessary for the restorative read, and to speed up the read cycle. While the write drivers do take up a fairly small portion of
the entire area, the sense-amp does take a significant portion; this is mostly because of the wide-transistor current source necessary, as well as the cross-coupled inverters necessary to drive the two bit-lines away from each other. Even through the write drivers do not take up a significant amount of area, this will add up when considering large segments of DRAM.

Figure 3-8: 1-T DRAM bit-slice, showing 2 bits of storage per 3 columns
Figure 3-9: 1-T DRAM 32-bit block

Figure 3-10: 1-T DRAM restorative sense-amp, using a cross-coupled inverter
Figure 3-11: 1-T DRAM write driver, driving the left bit-line
3.3 Layouts using twice the number of n-type devices than p-type devices

The first section of this chapter details the layout issues for a layout topology that uses an equal ratio of n-type and p-type transistors. While using an equal ratio will yield a higher percentage of used transistors for simple logic blocks, it is clear that a fair number of pFets are unused in memory cells and domino gates. For this reason, it is best to explore topologies that feature an uneven ratio of nFets and pFets.

3.3.1 SRAM 6-T cell

The layout for the SRAM 6-T cell with a 2:1 ratio of N:P is very similar to the layout in the previous section. In fact, the layout is identical except that one row of p-diffusion is missing. Therefore, with respect to the memory cell itself, there should not be any behavioral changes. However, the rest of the system does change slightly, since the write drivers must use smaller transistors.

3.3.2 DRAM 3-T cell

Of the three main memory cells explored in this thesis, the DRAM 3-T cell suffers the most from the removal of a p-diffusion row. With an equal number of nFets and pFets, it is possible to utilize both the nFets and pFets to create a pFet DRAM cell for every nFet DRAM cell. However, with the intended "optimization", this is not possible anymore.

It is still possible to use a high percentage of transistors. As mentioned in the previous section, a DRAM 3-T cell occupies 3 transistor columns. It is possible to match a pFet DRAM cell for every two nFet DRAM cells, if the pFet cell uses only the single transistor size available. Despite the loss in performance, this can utilize a larger percentage of transistors.
One last method of laying out a DRAM 3-T cell is to lay it all out in a single transistor row. This actually improves the percentage of transistors used; however, the performance decreases slightly because the transistor sizes are all the same. Further, if there are two different transistor sizes, optimal performance would not be attainable since one memory cell would always outperform the other; therefore, a connecting sub-system would need to meet the slower, more inefficient memory cell’s specifications.

### 3.3.3 DRAM 1-T cell

In the DRAM 1-T memory system described in the previous chapter, there are very few pFets used to implement the system. In fact, the only pFets utilized are in the sense amplifier. However, almost all of the pFets’ gates served as additional storage capacitance for each storage bit. Therefore, the expected result from remove a row of
p-diffusion is a functioning system, will performance loss, especially in reads. Reads will suffer such a loss because maximizing the $\Delta V$ for a read of a 0 on the bit line will cause the cross-coupled inverters to react faster. Additionally, the system will be more susceptible to noise.

The only figure is the bit-slice, as previously shown in the 1-T DRAM cell. However, the minimum capacitance that a storage bit uses is a minimum-sized nFet gate and a minimum-sized pFet gate. The sense-amp and write drivers are similar in size and shape to the 2:2 ratio of pFets:nFets, since the extra pFets were not used efficiently.

Figure 3-13: 1-T DRAM bit-slice using 2:1 of N:P ratio, showing 2 bits of storage per 3 columns
3.4 Layouts using only one n-type and one p-type transistor

This section serves to present a set of “control” layouts to compare the two previously mentioned layout topologies. A layout with a single nFet and pFet per column is the typical topology used in gate arrays. The expectations from layouts in this section is lower percentage transistor used and decreased performance in circuit speed.

Since larger transistors are necessary to drive signals, it is necessary to select the larger nFet/pFet pair from the first layout presented in figure 3-1.

3.4.1 SRAM 6-T cell

As figure 3-14 shows, the SRAM 6-T cell using the traditional sea-of-gates pattern cannot use the transistors in a given cell efficiently. Six transistor columns are required to implement the cell, since the access nFets must be in the same row as the nFets used for the inverters. Lastly, because the SRAM cell needs to be regular, there is an additional unused pFet in each cell.

3.5 Conclusion

This chapter presents a number of transistor topologies. The main geometries tested include 3 different types. The first sampling features a small pFet/nFet pair and a large pFet/nFet pair. The second uses a single pFet size, and a choice between a small and a large nFet. Lastly, a single pFet/nFet pair serves as a control to make comparisons to a typical gate-array layout.

It is important to look at all the different circuit types when evaluating a specific transistor technology. For example, looking only at standard CMOS gates, there is no reason to have unequal N:P ratios. It would make sense to have either the 1:1
or the 2:2 ratio. However, when looking at domino and memory cells, then it makes perfect sense to have unequal ratios. Based on these two types of circuits, a 2:1 ratio is suitable. While it is true that domino logic would benefit from having a ratio higher than 2:1, other circuits would suffer.

### 3.5.1 Standard CMOS Cells

Since standard CMOS cells always require a N:P ratio of 1:1, any transistor layout with such a ratio will results in the highest utilization percentage. Because of this fact, 2:2 and 1:1 transistors layouts would fair better than the proposed 2:1 transistor layout.

However, because both many digital systems use memory and memories occupy large chunks of silicon, it is more important to consider the transistor utilization
of memory cells. Additionally since computation cells which typically use standard CMOS can use domino logic cells instead, a higher N:P ratio would benefit these subsystems as well.

3.5.2 Memory Cells

Table 3-1 shows the percentage of transistors for different types of memory cells. This is an accurate representation of the percentage of transistors when looking in an entire array of memory cells, since nFets and pFets in between cells need to be grounded to prevent unwanted current flow. Here, it is easy to compare the efficiency of each type of layout.

Unfortunately, this table does not provide all of the characteristics for each type of memory cell. For example, the SRAM 6-T cell percentages are fairly accurate and indisputable. However, for the DRAM 1-T cell, the slight difference in percentage is based on the reasoning that there is an additional transistor row for the 2:2 transistor ratio. Functionally, the two cells will perform almost identically; the 2:2 will only perform a little better because of the higher capacitance for each bit cell, attainable with the additional transistors in the extra pMOS row. For the DRAM 3-T cell, it is possible to lay out a cell in any single diffusion row. However, most optimally, two rows will allow for different transistor sizes. Using a single row will allow for a 50% utilization ratio, which using two rows will allow for 60% utilization. For the DRAM 3-T cell using the 2:1 ratio, creating a 2-row cell for the pMOS and a 1-row cell for the nMOS results in the reported 65% utilization percentage.
Table 3.1: Percent of used transistors per type of memory cell

<table>
<thead>
<tr>
<th>Memory Cell Type</th>
<th>Transistor ratio (N:P)</th>
<th>Percent Transistors Used</th>
</tr>
</thead>
<tbody>
<tr>
<td>SRAM 6-T Cell</td>
<td>2:2</td>
<td>37.5</td>
</tr>
<tr>
<td>SRAM 6-T Cell</td>
<td>2:1</td>
<td>50.0</td>
</tr>
<tr>
<td>SRAM 6-T Cell</td>
<td>1:1</td>
<td>50.0</td>
</tr>
<tr>
<td>DRAM 1-T Cell</td>
<td>2:2</td>
<td>83.0</td>
</tr>
<tr>
<td>DRAM 1-T Cell</td>
<td>2:1</td>
<td>77.8</td>
</tr>
<tr>
<td>DRAM 3-T Cell</td>
<td>2:2</td>
<td>50.0</td>
</tr>
<tr>
<td>DRAM 3-T Cell</td>
<td>2:1</td>
<td>65.0</td>
</tr>
</tbody>
</table>

3.5.3 Domino Logic

As previously mentioned, there are several benefits of using domino logic, especially in a transistor system with a higher than equal N:P ratio. First, the computation cycle for domino logic can be quicker than a computation cycle for standard CMOS. Second, since there is a higher N:P transistor ratio for this type of circuit, it makes sense that a 2:1 ratio would be better than an equal ratio of transistors. The only pFets necessary for these circuits are the pFets for the pullups and for the output inverters.
Chapter 4

Analysis

The last phase will be to use the newly generated topology and to test the set of circuits that prove that the topology cannot be improved further given the types of circuits proposed in the first stage. It is true that there will be more optimal topologies for specific types of circuits, but the new topology will be efficient when considering all different types of circuits at the same time.

An important part of the analysis is to determine the key attributes to consider. Weste summarizes the key attributes; they include transition times, dense memories, power dissipation, precharging, power supply, packing density and layout [7]. These main topics should be considered in the analysis. The most important attributes and analysis methods are detailed in this chapter.

For each key attribute, the section describes the appropriate analysis, whether it is a simple explanation or comparison, or an experiment using SPICE. The basic features of any circuit, such as capacitances, switching characteristics and power is fairly independent; therefore, these can be tested using very simple Magic layouts and SPICE simulations independent of any specific transistor topology. However, layout-specific attributes require comparisons of topology characteristics and or SPICE simulations extracted from Magic.
4.1 Capacitances

Switching characteristics for FETs are dependent on different capacitances in the system, including gate, drain, and routing capacitances. It is important to see how the current technology limits how fast circuits can switch, and how capacitances influence the behavior of the system.

4.1.1 Gate Capacitance

Gate capacitances can tell what the maximum fan-in for gates and cells should be, since for each additional input that a cell must drive, there is additional capacitance to charge. According to Weste[7], gate capacitance may be approximated by the equation

\[ C_{\text{gate}} = \frac{\varepsilon_{\text{SiO}_2}\varepsilon_0}{t_{\text{ox}}} A \]  

(4.1)

The MAGIC Maintainer’s Manual states that Magic should not calculate gate capacitance, since the size of the transistor’s gate is extracted, and SPICE will make the necessary calculations to calculate the gate’s capacitance.

4.1.2 Drain Capacitance

Drain capacitance refers to the capacitance between the active drain region and the substrate. There are two key parameters that Magic extracts called AD and PD. Respectively, they are the area and the perimeter of the drain region of the transistor. With these two parameters, the total area between the drain region and substrate provides the necessary information to calculate the drain capacitance.
4.1.3 Wire Capacitance

When using sub-micron technologies, wire capacitances can be a large factor in designing systems. Long bus wires can lead to a longer gate delay because of the large parasitic capacitances in routing wires. Kang describes in [4] that the “time of flight” time (determined by the speed of light) is much shorter than the rise/fall times, inductance can generally be ignored on VLSI chips. However, as gate delays decrease, inductance becomes more of an important factor.

![Figure 4-1: Different types of wire capacitances](image)

The different types of capacitance that affect wires are parallel-plate, fringing-field, and inter-wire capacitances. Figure 4-1 labels these three types of capacitances. Parallel-plate capacitances refer to the capacitance between the bottom plane of the wire to the substrate. This capacitance is approximated by the equation

\[
C = \frac{\varepsilon \cdot A}{t}
\]  

(4.2)

where \(\varepsilon = 8.854 \text{ pF/m}\), \(A\) is the cross-sectional area, and \(t\) is the distance between the substrate and the wire.

The importance of fringing-field capacitances is emphasized in [5]. The articles explains that depending on the geometry of the wires, it may be necessary to use two-
or three-dimensional representations of wires. According the the graphs, the fringing field-capacitance can be many orders of magnitude greater than the parallel-plate capacitance. The NTRS Roadmap for Interconnects states that the metal aspect ratio is approximately 2.1; this means that the height of a metal wire is roughly twice the width of the wire. Ruehli states that given this aspect ratio, the fringing-field capacitance is anywhere between eight and twenty times the parallel plate capacitance to substrate, depending on the length of the wire.

Inter-wire capacitances is a third factor in determining capacitances in a system. Veendrick, in [6] states that as the CMOS process size decreases, the cross-talk problem increases. Furthermore, a node in the high-impedance state is very sensitive to cross-talk from neighboring signals. The main design principle to follow is to decrease the area and number of signals crossings. This is especially important in dynamic circuits, since nodes are often left in the high-impedance state, so the voltage on these nodes can easily change.

4.1.4 MAGIC Technology File

The technology file for Magic contains all of the necessary information for extracting capacitances from a layout. By interpreting this information, estimates and predictions for circuit performance is attainable even before the implementation of a design.

Different types of capacitances defined in the tech file includes areacap, perim-cap, overlap, sidewall, and sideoverlap. The next few paragraphs briefly describes each of these keywords:

areacap

The areacap capacitance describes the capacitance to substrate; for example, the areacap for metall would describe the parallel-place capacitance, since the planar area parallel to substrate is the key parameter in determining this type of capacitance.
perimcap

The perimcap capacitance requires a perimeter parameter. This is one of the most important capacitances to consider, since this determines the fringing-field capacitances for wires.

overlap

While areacap determines the capacitance to substrate, there is often different layers inbetween the layer in question and the substrate. The overlap capacitance is intended to make the proper adjustment. Using this keyword, capacitance is deducted from the capacitance to substrate, and replaced by capacitance to the appropriate layer.

sidewall

The sidewall capacitance refers to the capacitance in a single layer; for example, this occurs when two metall wires run in parallel. This can be important in analysis because bus wires are often necessary to carry different signals for long distances.

sideoverlap

Lastly, sideoverlap corresponds to the capacitance between the sidewall of a particular layer and a different layer. This can be important since sidewall area is now a large factor given the aspect ratio, as previously noted.

4.1.5 Experiments

Capacitances will be dependant on the geometry of the devices and wires, as well as the Level 3 SPICE models for the fets. While some experiments can prove worthwhile using only SPICE models, most of the tests will provide more realistic results using Magic as well. The “SPICE-only” experiments provided in Chapter 2 served only as
a starting point to make educated initial designs for improvements to does and does not provide a reasonable substitution for layout simulations.

To measure wire capacitances, laying out two metal 1 wires next to each other in *Magic* using a technology file will yield estimates of wire capacitances. Comparing this estimate to the analysis provided earlier, it is verifiable that *Magic* is providing reasonable estimations for wires in the layouts demonstrated in the previous chapter.

Experiments that estimate gate and drain capacitances are provided in the Switching Characteristics section, since sampling different switching times can lead to the capacitances in question.

**Results**

While some of the experiments detailed in the next few paragraphs may not be very typical of metal geometries found in layouts, they still provide relevant information regarding wire capacitances. They demonstrate how parallel-plate capacitances and fringing-field capacitances balance with each other, and when each type of capacitance would not be a large factor in specific cases.

**Different sized rectangles.** The first experiment is a very basic metal 1 to substrate test. By creating some simple metal 1 geometries in MAGIC, using `ext2spice` after extracting capacitances from the layout will provide basic capacitance information and how metal layout will affect circuit behavior. Table 4-1 shows some results from some simple metal 1 rectangles and their corresponding capacitances. This demonstrates that is both area and perimeter are important factors when determining capacitances. In these extractions, MAGIC uses `areacap` and `perimcap` capacitances for metal 1 to substrate.
Table 4.1: Table of Simple Metal 1 rectangles and corresponding capacitance

<table>
<thead>
<tr>
<th>Dimensions (in μm)</th>
<th>Perimeter (in μm)</th>
<th>Area (in μm²)</th>
<th>Capacitance (in fF)</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.125 x 3.125</td>
<td>12.500</td>
<td>9.766</td>
<td>0.5</td>
</tr>
<tr>
<td>6.250 x 6.250</td>
<td>25.000</td>
<td>39.063</td>
<td>1.6</td>
</tr>
<tr>
<td>3.125 x 12.500</td>
<td>31.250</td>
<td>39.063</td>
<td>1.7</td>
</tr>
<tr>
<td>1.500 x 24.000</td>
<td>51.000</td>
<td>36.000</td>
<td>1.9</td>
</tr>
<tr>
<td>0.750 x 48.000</td>
<td>97.500</td>
<td>36.000</td>
<td>2.4</td>
</tr>
<tr>
<td>0.375 x 48.000</td>
<td>96.750</td>
<td>18.000</td>
<td>1.7</td>
</tr>
</tbody>
</table>

**Overlapping rectangles**  Magic analyzes capacitances based on parallel-plate and fringing-field capacitances. The next example tests two squares of metal; one large square of metal 1, and one large square of metal 2. By overlapping one quarter of their total area (each 96 by 96 squares), as shown in Figure 4-2, we can show the how the two layers of metal affect each other.

When determining the capacitances of each square individually, the metal 1 and metal 2 squares each has a capacitance of 6.0 fF and 2.1 fF respectively. The distance between each appropriate layer of metal and substrate is a large factor when determining the capacitance to substrate.

Since the metal 1 square overlaps the metal 2 square, the metal 2 square’s capacitance to substrate will decrease. Additionally, there will be a capacitance between
the two metal squares. After extracting the capacitances with the given squares, the metal 2 to substrate capacitance is only 1.6 fF, and the metal 1 to metal 2 capacitance is 1.8 fF. The metal 1 to substrate remains at 6.0 fF, as expected.

Retrying the experiment with 100% overlap yields reasonable results. In this case, the parallel-place capacitance is much higher since there is more overlap; the total capacitance between the two places is 6.1 fF. Also as expected, the metal 2 to substrate is only 0.2 fF, since this capacitance only results from the little bit of fringing-field capacitance.

Stacking rectangles. The last basic experiment is to stack five squares of metal using the same size as in the previous experiments. In this case, each layer of metal to its successive layer of metal should have roughly a capacitance of roughly 6.0 fF as measured in the previous test. Additionally, each metal to substrate capacitance will be fairly low, with the exception of metal 1 to substrate and metal 5 to substrate. Of course, metal 1 does not have any other metals between it and the substrate. Metal 5 has additional fringing-field capacitances from the top surface of the metal to substrate. This fringing-field capacitance is again diagrammed in Figure 4-1.

As expected, each capacitance between layers is approximately 6.0 fF. Similarly, each layer to substrate is about 0.2 fF, with the exception of metal 1 and 5. The additional fringing-field capacitance brings the metal 5 to substrate capacitance to 1.1 fF.

### 4.2 Switching Characteristics

Switching characteristics refer to the rise and fall times of a signal, as well as delay times from input to output. All of these times are important in determining how fast a specific digital system can run.

For timing analyses, experiments will assume that the rise time, \( t_R \) is the time for
a signal to rise from 10% to 90% of the steady-state value. Similarly, the fall time \( t_F \) is the time for a signal to fall from 90% to 10% of the steady state value. Lastly, the propagation delay \( t_{PD} \) is the time between the input and output transition, defined as the 50% point. Traditionally, the key point to consider is the switching threshold, determined by connecting the input to the output of a standard minimum sized inverter. However, for these experiments it is unnecessary to complicate such measurements. As the Results section explains, the switching threshold is fairly close to the 50% point, so it is a moot point.

Rise and fall times are highly dependent on \( C_{load} \), \( \beta \), and \( V_{DD} \). \( C_{load} \) represents the load capacitance of a fet, and includes the drain capacitance as well as the gate capacitances of successive stages. \( \beta \) is the MOS transistor gain factor. Clearly, \( \beta \) and \( V_{DD} \) are process-specific, while \( C_{load} \) changes depending on fan-out in a specific system.

### 4.2.1 Experiments

It is best to start analyzing switching characteristics from the simplest circuit to more complicated circuits. Even a circuit as simple as an inverter requires several stages to perform to analyze, from layout to extraction and simulation. At the simplest level, the technology speed parameter, \( \tau \) is determined by layout out three inverters, and measuring the propagation delay from the input and output of the middle inverter.

The input needs to be able to drive two fets, specifically the nfet and the pfet of the inverter. Since all inputs defined in SPICE are ideal, it will be necessary to put a SPICE input through another inverter to generate a digital system's "typical" signal. The output of this inverter will be a better signal to send to the inverter under test.

It is also important to consider the output of the inverter under test. In the simplest case, leaving the inverter unconnected will show how long it takes for a signal to propagate and switch with only the drain capacitance. However, a typical
node in a circuit connects to many gates; most analyses use F04 delays as a standard, meaning that the output connects to four inverters that represents the next stage.

![Figure 4-3: Minimum sized inverter layout](image)

**Results**

Using the minimum sized inverter as shown in Figure 4-3, connecting the input to the output determines the threshold voltage. *SPICE* calculates this steady-state voltage to be 1.12 volts. As mentioned earlier, 1.12 volts is close enough to 1.25 volts, the 50% threshold, so for purposes of propagation delay, using 50% point will be both convenient and less confusing.

In retrospect, if one were to design transistor sizes to have the switching threshold voltage exactly on the 50% point, the changes to the current design would only be to change the N:P size ratio. The size would be based on the electron and hole
mobility; however, in most cases, the mobility ratio is assumed to be close enough to 1:2, respectively.

Figure 4-4: Circuit diagram for HSpice inverter under test

Figure 4-4 shows the circuit used to test a typical inverter. Note that there are two inverters in the test circuit. The first inverter is only for signal shaping; the input to the second inverter is more typical of a signal that an inverter would see in the middle of a circuit. There are four voltage sources, $V_{DD}$, $V_{DD2}$, $V_{DSP}$, and $V_{DSN}$. The first two voltage sources are to ease the circuit setup so that the measurement for the current sourced from $V_{DD}$ will be solely from the inverter under test. The latter two voltage sources are set to 0 V and only serve as a measurement device for current. The current graphs are used in the next section to describe power dissipation.

Figure 4-5 shows characteristics of an 8/4 inverter under test. From measurements on the rising edge of the input, the $t_{PD}$ of the signal is 0.06 ns. The $t_R$ of the output is 0.07 ns. On a rising edge of the input, the $t_{PD}$ of the signal is 0.05 ns. The $t_F$ is 0.06 ns.
While the 8/4 inverter under test does give some initial measurements, the results is only a control towards typical inverters in a system. Since logic is typically only an intermediary stage in a system, this next experiment adds an F04 delay to the output of the inverter under test. This means that 4 inverters connect to the output to behave as a typical load.

Figures 4-6 again show the characteristics of an 8/4 inverter. However, the figure shows an increase in the propagation delay because of the additional load capacitance.

Figure 4-5: In (solid) and Out (dotted) port of a typical inverter of size 8/4 with no load

Figure 4-6: In (solid) and Out (dotted) port of a typical inverter of size 8/4 with F04 load
4.3 Power Dissipation

Today’s digital systems can consume a large amount of power because many systems operate at very higher frequencies, and leakage current increases as technology sizes decreases. Most of the power dissipated is a result of dynamic dissipation, although with sub-micron technologies, it is important to consider static dissipation as well.

By estimating the static and dynamic power dissipation, it is possible to estimate how much power a digital system would dissipate.

4.3.1 Static Dissipation

There are two main sources of static power dissipation. First, certain circuits such as sense amplifiers require constant current sources. Certainly, these elements can be switched on and off at appropriate times to conserve power.

Leakage current is also a factor in static dissipation; as transistors decrease in size, the leakage current is a larger factor to consider. Three cases to consider are the gate leakage, subthreshold leakage and diode leakage currents.

As gate oxides decrease in thickness, more electrons tunnel across the oxide. In today’s technology, gate oxides are only a few molecules thick.

Subthreshold leakage increases dramatically as transistor sizes decrease. Given a specific low voltage, an nFet should be switched off. However, since fets are in fact not true digital parts, there will be current leakage between the drain and the source of the fet.

While ideal p-n junctions have zero current flow when reverse-biased, there is a reverse saturation current $I_s$ that adds to the total static power dissipation.
4.3.2 Dynamic Dissipation

Dynamic power dissipation can be a large factor in the total power dissipation, especially if the operating frequency is very high. This includes drain, gate and wire capacitances.

Capacitive Load

The size of the load capacitance is one of the major factors towards how much power a gate dissipates. The total amount of energy required to charge $C_L$ is

$$E_{0 \rightarrow 1} = C_L V_{DD}^2$$  \hspace{1cm} (4.3)

Half of the energy dissipates as heat through the pFet when the load capacitor stores $\frac{1}{2}C_L V_{DD}^2$. The energy stored in the capacitor dissipates through the nFet when the nFet turns on.

Short Circuit. Since mosfets are not perfect switches, there is a short period of time on any transition from 1 to 0 or from 0 to 1 where both fets are conducting. This results in a short circuit current.

4.3.3 Experiments

Using the simple equation $P = VI$, we can determine how much average power a certain circuit cell consumes. Therefore, the product of $SPICE$'s current measurements and the supply voltage will generate the appropriate calculation for each cell.

For static power dissipation, the current measurable when no signals are changing in the circuit provides the necessary parameter to calculate the dissipation.

For dynamic power dissipation, it is more difficult to estimate the power usage. In a larger system, a particular subsystem may be active only during certain times.
Therefore, the best way to analyze such a circuit is to perform a typical calculation, and then estimate the percentage time that a circuit will be on. From these estimations, getting the average power dissipation at all times is simply a product of the average “ON” power, and the average percentage “ON”.

Results

The results here demonstrate measurements for static and dynamic power dissipation for a minimum-sized inverter.
Figure 4-7: Currents for typical inverter of size 8/4 with no load
Figure 4-7 and 4-8 show different current characteristics for the inverter under test in the circuit configuration figured in the previous section. From each pair of graphs, extracting the static and dynamic power characteristics requires simple analysis.

In terms of static power characteristics, it is important to consider the inverter when no switching occurs. Using the measurement tool on the graph, this will give a small positive current sourced from $V_{DD}$. The static power dissipation sourced for the inverter is minimal.

For dynamic power characteristics, analyzing the short circuit current requires extracting the total charged sourced from $V_{DD}$ each transition. Using the current characteristics, it is possible to calculate the area under the short circuit current curve. Then using the frequency of the system, it is possible to extrapolate the total power dissipation from short circuit current.

Dynamic dissipation accounts for the source and drain capacitances for the transistors in a circuit. Every switch of an inverter causes the output node to either charge or discharge its capacitor. As with short circuit current, calculating the area under the curve accounts for how much charge is required for a single minimum-sized inverter in this system.
Figure 4-8: Currents for typical inverter of size 8/4 with F04 load

(a) Current sourced from Vdd

(b) pFet and nFet drain current
4.4 Supply Voltage

Of course, choosing an ideal supply voltage is an important factor when considering a specific technology. It is important to realize that power and switching characteristics are all affected by this variable. As the supply voltage decreases, the delay rises because the supply voltage closely approaches the fet’s threshold voltage. However, the larger the supply voltage, the more energy is consumed.

4.4.1 Experiments

Measuring the ideal supply voltage for a certain technology requires determining the supply voltage versus delay time and power consumption for a system. The simplest way to test this is to use a simple gate, and extract the power consumption, propagation delay, rise and fall times. By repeating this extraction, and varying the
supply voltage, then determine the best power-delay time product, since both factors would ideally be minimized.

Results

Again using the same minimum-sized inverter setup in the previous sections, iterating between different voltages and analyzing delays and power will generate a function of delay versus supply voltage. From this function, weighing the characteristics as suitable by the designer allows a system to be low-power, or high-speed, or some median.

4.5 Layout

As described, Chapter 3 describes three different transistor topologies. First, it describes a 2:2 ratio of nFets to pFets. Next, it describes a similar topology but with only 1 pFet for every 2 nFets. Lastly, a typical 1 nFet to 1 pFet demonstrates a typical gate array layout. There are two main measurements that determine how successful a topology is over another: total circuit area and percentage of transistors used.

4.5.1 Total Circuit Area

The clearest metric to use to determine the efficiency of a layout is the total circuit area. It would be best to minimize the amount of area a circuit occupies, therefore saving silicon and cost. Circuit area certainly involves all of the transistors necessary to implement the circuit, but it also includes the transistors wasted due to routing, such as the pFets in the domino full-adder (Figure 3-4).
4.5.2 Circuit Overhead

Every gate array layout has a certain amount of “overhead” because of design rules, which limit the spacing between transistors and metals. Because of this, it is important to consider the efficiency in terms of the layout itself using the equation

\[
\text{Circuit overhead} = \frac{\text{gate area}}{\text{total area}}
\]  

(4.4)

While a sea-of-gate layout is very efficient since there is not a lot of empty space, the layout topology described by Iranmanesh et al. in [2] is not as efficient because groups of transistors are separated vertically as well as horizontally.

4.5.3 Percent of Transistors Used

While total circuit area gives an overall efficiency for a chip, it is also helpful to analyze the transistor usage in a system. This is important because if the transistor percentage is very low, then a layout can likely be improved to remove unnecessary transistors. This is clear from the 2:2 layout, where the SRAM cell never utilizes the top row of pFets, or when a domino logic block uses a very small percentage of the pFets just because domino logic is so nFet dependant.

4.6 Conclusion

This analysis chapter is important to consider when designing digital systems using the proposed transistor topology. It focuses on five different aspects of a digital system: capacitance, switching characteristics, power dissipation, supply voltage, and layout. Each of these characteristics will affect the performance of the system as well as the cost of the entire system.

The input and output capacitance of subsystems is important because it deter-
mines how to connect subsystems together. Based on these measurements, it may be necessary to add inverters give certain signals appropriate drive. Of course, entire system itself needs an input/output interface to the outside of the chip. At the very end, the capacitance of a system determines the switching characteristics of the system.

Of course, switching characteristics determine many different timing factors for a digital system. For any subsystem, the rise time, fall time, and propagation delay can provide useful information as to how the circuit will communicate between other subsystems. Lastly, by combining all the subsystems, the final characteristics of the system are determinable.

Power dissipation in today’s digital systems is very important. With smaller transistors and higher clock frequencies, static and dynamic power dissipation is an increasing problem. It is important to analyze certain parts of silicon for high power dissipation, since certain parts of a chip can burn up if overused. There are a number of ways to moderate the power to save energy and avoid any temperature problems. For each transistor technology, dynamic and static characteristics can determine the maximum frequency for a certain system and other limitations due to power.

Directly related to power is the supply voltage for a digital system. The higher the supply voltage, the higher the power dissipated. However, there is a second factor to consider when selecting supply voltage. Choosing a higher supply voltage results in a shorter delay time for signals using a certain supply. Choosing a smaller supply voltage will results in a longer delay time, since the supply voltage is closer to the threshold voltage for each transistor. Therefore, a specific balance will allow the best choice, weighing power dissipation and delay time appropriately.

Although the previous chapter does summarize layout characteristics and efficiency, it does makes sense to include layout in the analysis chapter of this thesis. As mentioned, the key points for layout include total circuit area, circuit overhead,
percentage transistors used. In general, the total circuit area is the most important characteristic, since this will provide the best estimate for the cost of the project. Circuit overhead includes the nFets tied to ground or the pFets tied to $V_{CC}$ as well as transistors not used due to routing tracks. However, routing tracks are not needed as much because the number of layers of metals in different technologies are enough to be overlap transistors and routing track. Lastly, the percentage transistor used relates to the first two characteristics, and this point is addressed in the previous chapter's conclusion.
Chapter 5

Conclusion

This thesis undergoes two main phases: First, the design of certain circuits, including basic gates, more complicated full-adder cells, domino cells, and different assortment of memory cells and circuits. The second phase is to implement layouts for each of these phases. An analysis chapter served to briefly tour the important concepts in both designing and evaluating a specific transistor technology.

5.1 Current State

This thesis offers a design for a transistor topology that would be suitable for a sea-of-gates design methodology. However, Magic does not have any place-and-route tools available; all layouts in this thesis used no extra routing tools.

Although it is true that the main design is presented, implementing the master images in Ocean would allow a designer to develop key cells and SLS files to describe his digital system. Using the extra tools would speed up the design process, and the end result would be an improvement in performance and cost compared to the standard gate-array master image provided with Ocean.
Table 5.1: Table of different transistor ratios and area usage for a 128x32 SRAM

<table>
<thead>
<tr>
<th>Transistor Ratio (N:P)</th>
<th>Dimensions (in ( \mu m ))</th>
<th>Area (in ( mm^2 ))</th>
</tr>
</thead>
<tbody>
<tr>
<td>1:1</td>
<td>552.96 x 622.08</td>
<td>0.344</td>
</tr>
<tr>
<td>2:2</td>
<td>829.44 x 380.16</td>
<td>0.315</td>
</tr>
<tr>
<td>2:1</td>
<td>552.96 x 414.72</td>
<td>0.229</td>
</tr>
</tbody>
</table>

5.2 Comparisons

5.2.1 Area and Performance Comparison

Area

Since this thesis's goal is to evaluate different transistor topologies, it is useful to make some comparisons between the three main transistor ratios sampled: 2:2, 2:1, and 1:1 for nFet:pFet.

While initially a 2:2 transistor ratio seemed to provide up to 3 transistor sizes for pFets and 3 transistor sizes for nFets, there were a couple problems with this system. First, circuits consumed less pFets than nFets. This lead to large areas of silicon wasted. Additionally, the availability of 3 pFet sizes was not useful.

A 2:1 transistor ratio optimizes the silicon usage for most types of cells, with the exception of standard CMOS. As you can see by the table for the three different topologies, the 2:1 ratio is most optimal by a decent margin.

The 1:1 ratio is primarily to provide a standard for comparison. Of course, the N:P ratio is equal, so this topology is still less optimal in terms of area usage.

Performance

Again comparing the three different transistor ratios, there is no significant difference between the read and the write timing. Between the 2:1 and 2:2 ratio, the timing
is virtually identical; this is mainly because the two nFet sizes are available in both cases. The read timing for these cases is approximately 170 ps.

For the 1:1 ratio, the timing is slightly different, but again, there is no great difference compared to the 2:1 and the 2:2 ratio. In this case, the read timing is approximately 210 ps. This can be attributed to the larger transistors available, rather than the smaller transistors used in the 2:1 and 2:2 ratio systems.

5.2.2 Different Transistor Topologies in a System

Chapter 3 samples a number of different N:P ratios, including 2:2, 2:1, and 1:1. Between these three, the 2:1 ratio would seem to fare better than the others. However, there is certainly an advantage in having an equal ratio when using standard CMOS. It is also possible that a higher ratio than 2:1 would be ideal for some applications of domino logic. Because of this, prefabricating chips with certain areas of the chip designated for certain functions may prove even more optimal than choosing a single ratio for the entire chip.

For example, in a typical processor, one may have a memory, and ALU, and random control logic. Of course, an SRAM memory benefits most from a 2:1 layout, as suggested by the layout analysis for memories in chapter 3. The ALU may benefit most from domino logic, and it may be most beneficial to have a larger ratio such as 3:1. Lastly, the control logic, comprised mostly of standard CMOS, would most optimally use an equal ratio of N:P.

While every digital project does use a different amount of RAM, contains different amount of processing and use certain amounts of random logic, advantages are certainly possible by estimating ratios between these needs.

However, there is a large disadvantage for allocating specific amounts of RAM on a piece of silicon. Unless a designer is sure the ratio will be nearly the same every design, it is probably better to use the generic 2:1 ratio of N:P, since each portion
would be generic enough so that any part of silicon could be allocated to whatever
the designer needs. If sections were preallocated, then either the performance would
degrade considerably due to ratio incompatibilities, or the area consumed would in-
crease drastically.

5.3 Predictions for Larger Systems

While this thesis does not cover larger digital systems which would be a more typical
goal in a design process, it is possible to make certain predictions about how a larger
system would behave. The following can be extrapolated from the analysis provided
in the previous chapters.

5.3.1 Routing Availability

In all of the sample layouts presented in this thesis, each layout utilized the minimum
number of metals layers necessary. This is important because other layers are neces-
sary to connect the cells together on a higher level of hierarchy. While this technology
had 6 layers in total, none of the cells required so many layers.

In today’s technology, some processes have as many as 9 metal layers. However,
each layer does have a significant cost, and avoiding the use of any layers possible will
generally yield a better design, in terms of both cost and performance.

Because the number of routing layers available, “routing tracks” which were nec-
essary for older technologies are not as necessary. Routing forced designers to leave
transistors unused so that routing wires would be able to connect subsystems together,
or even to route within a subsystem.
5.4 Further Work

5.4.1 Ocean

As Groeneveld and Stravers mentions in [3], their documentation for Ocean, the sea-of-gates strategy has several benefits. First, all chips are prefabricated, so the silicon foundry does not need to produce as many masks for the design. This saves a lot of time when attempting to produce the chip. Second, since transistors are already placed on the chip, the design time is minimized. Third, chips are cheap to produce because the chips are prefabricated.

Taking the next step to implement the desired transistor topology into a master image should be a fairly straightforward step. After providing the images and other relevant data, such as routing costs and metal layers available, a designer would be able to use Ocean to develop chips using this transistor topology.

Fabricating Chips with Ocean

The DIMES facility at the Delft University of Technology can produce chips based on the fishbone master image. In the early 1990’s, the facility producer many chips designed with the Ocean CAD tool, but they do not provide any support to produce chips with master images other than the fishbone array. Therefore, if there are to be any designs for this proposed transistor topology, the chips would need to be fabricated.
Appendix A

SPICE Design Files

A.1 Standard Gates

aoi2l.sp

* SPICE aoi2l.sp

.SUBCKT inv A Y Vss Vdd
m1 Vss A Y Vss cmosn l=0.25u w=0.50u
mp1 Vdd A Y Vdd cmosp l=0.25u w=1.00u
.ENDS

mn1 Vss A int0 Vss cmosn l=0.25u w=2.0u
mn2 int0 B Y Vss cmosn l=0.25u w=2.0u
mn3 Vss C Y Vss cmosn l=0.25u w=2.0u

mp1 Vdd A int1 Vdd cmosp l=0.25u w=4.0u
mp2 Vdd B int1 Vdd cmosp l=0.25u w=4.0u
mp3 int1 C Y Vdd cmosp l=0.25u w=4.0u

Xinv1 Y cap1 Vss Vdd inv
Xinv2 Y cap2 Vss Vdd inv
Xinv3 Y cap3 Vss Vdd inv
Xinv4 Y cap4 Vss Vdd inv

Xinv5 nainput a Vss Vdd inv
Xinv6 nbinput b Vss Vdd inv
Xinv7 ncinput c Vss Vdd inv

vVdd Vdd 0 2.5
vVss Vss 0 0

* A 1 1 1 0 1 1 0 1
* B 1 0 0 0 1 0 0 0
* C 1 1 0 0 1 1 0 0

vmainput nainput 0 pw1(0p 2.5v, 100p 0v, 1250p 0v, 1350p 0v, 2500p 0v,
2600p 0v, 3750p 0v, 3850p 2.5v, 5000p 2.5v, 5100p 0v, 6250p 0v, 6350p
0v, 7500p 0v, 7600p 2.5v, 8750p 2.5v, 8850p 0v)
vmainput nbinput 0 pw1(0p 0v, 100p 0v, 1250p 0v, 1350p 2.5v, 2500p
2.5v, 2600p 2.5v, 3750p 2.5v, 3850p 2.5v, 5000p 2.5v, 5100p 0v, 6250p
0v, 6350p 2.5v, 7500p 2.5v, 7600p 2.5v, 8750p 2.5v, 8850p 2.5v)
vncinput ncinput 0 pw1(0p 2.5v, 100p 0v, 1250p 0v, 1350p 0v, 2500p 0v, 2600p 2.5v, 3750p 2.5v, 3850p 2.5v, 5000p 2.5v, 5100p 0v, 6250p 0v, 6350p 0v, 7500p 0v, 7600p 2.5v, 8750p 2.5v, 8850p 2.5v)

.tran 1ps 10ns

.option post

*Mosfet models from www.mosis.org

*DATE: Jun 11/01
*LOT: T14Y WAF: O3
*DIE: N_Area_Fring DEV: N3740/10
*Temp= 27

.MODEL CMOSNMOS

+ TUS = 5.7E-9 NSUB = 1E17
+ PHI = 0.7 VTO = -0.5536085 DELTA = 0
+ UD = 425.6466519 ETA = 0 THEA = 0.1754054
+ KP = 2.501048E-4 VMAX = 8.287851E4 KAPPA = 0.1686779
+ RSH = 4.062439E-3 NFS = 1E12 TPG = 1
+ XJ = 3E-7 LD = 3.162278E-11 WD =
1.232881E-8
+ CGDO = 6.2E-10 CDSO = 6.2E-10 CGBO = 1E-10
+ CJ = 1.81211E-3 PB = 0.5 MJ = 0.3282553
+ CJSW = 5.341337E-10 MJSW = 0.5

.MODEL CMOSPPMOS

+ TUS = 5.7E-9 NSUB = 1E17
+ PHI = 0.7 VTO = -0.5536085 DELTA = 0
+ UD = 250 ETA = 0 THEA = 0.1573195
+ KP = 5.194153E-5 VMAX = 2.296325E5 KAPPA = 0.7448494
+ RSH = 30.0776952 NFS = 1E12 TPG = -1
+ XJ = 2E-7 LD = 9.968346E-13 WD =
5.475113E-9
+ CGDO = 6.66E-10 CDSO = 6.66E-10 CGBO = 1E-10
+ CJ = 1.893569E-3 PB = 0.9906013 MJ =
0.4664287
+ CJSW = 3.625544E-10 MJSW = 0.5

.end
* SPICE aoi211.sp

.SUBCKT inv A Y Vss Vdd
mn1 Vss A Y Vss cmosn l=0.25u w=0.50u
mp1 Vdd A Y Vdd_cmosp l=0.25u w=1.00u
.ENDS

mn1 Vss A int0 Vss cmosn l=0.25u w=1.00u
mn2 int0 B Y Vss cmosn l=0.25u w=1.00u
mn3 Vss C Y Vss cmosn l=0.25u w=1.00u
mn4 Vss D Y Vss cmosn l=0.25u w=1.00u

mp1 Vdd A int1 Vdd_cmosp l=0.25u w=6.00u
mp2 Vdd B int1 Vdd_cmosp l=0.25u w=6.00u
mp3 int1 C int2 Vdd_cmosp l=0.25u w=6.00u
mp4 int2 D Y Vdd_cmosp l=0.25u w=6.00u

+ xinv1 Y cap1 Vss Vdd inv
xinv2 Y cap2 Vss Vdd inv
xinv3 Y cap3 Vss Vdd inv
xinv4 Y cap4 Vss Vdd inv

xinv5 nainput a Vss Vdd inv
xinv6 nbinput b Vss Vdd inv
xinv7 ncinput c Vss Vdd inv
xinv8 ndinput d Vss Vdd inv

vVdd Vdd 0 2.5
vVss Vss 0 0

* A 1 1 1 0 1 1 0 1
* B 1 0 0 0 1 0 0 0
* C 1 1 0 0 1 1 0 0
* D 1 1 0 0 1 1 0 0

vninput nainput 0 pwl(0p 2.5v, 100p 0v, 1250p 0v, 1350p 0v, 2500p 0v, 2600p 0v, 3750p 0v, 3850p 2.5v, 5000p 2.5v, 5100p 0v, 6250p 0v, 6350p 0v, 7500p 0v, 7600p 2.5v, 8750p 2.5v, 8850p 0v)

vbininput nbinput 0 pwl(0p 0v, 100p 0v, 1250p 0v, 1350p 2.5v, 2500p 2.5v, 2600p 2.5v, 3750p 2.5v, 3850p 2.5v, 5000p 2.5v, 5100p 0v, 6250p 0v, 6350p 2.5v, 7500p 2.5v, 7600p 2.5v, 8750p 2.5v, 8850p 2.5v)

vncinput ncinput 0 pwl(0p 2.5v, 100p 0v, 1250p 0v, 1350p 0v, 2500p 0v, 2600p 2.5v, 3750p 2.5v, 3850p 2.5v, 5000p 2.5v, 5100p 0v, 6250p 0v, 6350p 0v, 7500p 0v, 7600p 2.5v, 8750p 2.5v, 8850p 2.5v)

vndinput ndinput 0 pwl(0p 2.5v, 100p 0v, 1250p 0v, 1350p 0v, 2500p 0v, 2600p 2.5v, 3750p 2.5v, 3850p 2.5v, 5000p 2.5v, 5100p 0v, 6250p 0v, 6350p 0v, 7500p 0v, 7600p 2.5v, 8750p 2.5v, 8850p 2.5v)

.tran 1ps 10ns

.option post

* Mosfet models from www.mosis.org

* DATE: Jun 11/01
* LOT: T14Y
* DIE: N_Drain_Fring
* WAF: 03
* DEV: N3740/10
* Temp= 27

.MODEL CMOSN NMOS (LEVEL = 3)
+ TOX = 5.7E-9 NSUB = .1E17 GAMMA = .4317311
+ PHI = 0.7 VTO = 0.4238252 DELTA = 0
+ U0 = 426.6466619 ETA = 0 THETA = 0.1754054
+ KP = 2.501048E-4  VMAX = 8.287851E4  KAPPA = 0.1686779
+ RSH = 4.062439E-3  NFS = 1E12  TPG = 1
+ XJ = 3E-7  LD = 3.162776E-11  WD = 1.232881E-8
+ CGDO = 6.2E-10  CGSD = 6.2E-10  CGBO = 1E-10
+ CJ = 1.81211E-3  PB = 0.5  MJ = 0.3282553
+ CJSW = 5.341337E-10  MJSW = 0.5  )

MODEL CMOSP PMOS (LEVEL = 3
+ TDX = 5.7E-9  NSUB = 1E17  GAMMA = 0.6348369
+ PHI = 0.7  VTO = -0.5536085  DELTA = 0
+ UO = 250  STA = 0  THETA = 0.1573195
+ KP = 5.194153E-5  VMAX = 2.295325E5  KAPPA = 0.7448494
+ RSH = 30.07769E2  NFS = 1E12  TPG = -1
+ XJ = 2E-7  LD = 9.968346E-13  WD = 5.475113E-9
+ CGDO = 6.66E-10  CGSD = 6.66E-10  CGBO = 1E-10
+ CJ = 1.893569E-3  PB = 0.9996013  MJ = 0.4664287
+ CJSW = 3.625544E-10  MJSW = 0.5  )
*
.end
aoi22

* SPICE aoi22.ep

.SUBCKT inv A Y Vss Vdd
mn1 Vss A Y Vss cmosn l=0.25u w=0.50u
mp1 Vdd A Y Vdd cmosp l=0.25u w=1.00u
.ENDS

mn1 Vss A int0 Vss cmosn l=0.25u w=1.00u
mn2 int0 B Y Vss cmosn l=0.25u w=1.00u
mn3 Vss C int1 Vss cmosn l=0.25u w=1.00u
mn4 int1 D Y Vss cmosn l=0.25u w=1.00u

mp1 Vdd A int2 Vdd cmosp l=0.25u w=6.00u
mp2 Vdd B int2 Vdd cmosp l=0.25u w=6.00u
mp3 int2 C Y Vdd cmosp l=0.25u w=6.00u
mp4 int2 D Y Vdd cmosp l=0.25u w=6.00u

xinv1 Y cap1 Vss Vdd inv
xinv2 Y cap2 Vss Vdd inv
xinv3 Y cap3 Vss Vdd inv
xinv4 Y cap4 Vss Vdd inv

xinv5 nainput a Vss Vdd inv
xinv6 ninput b Vss Vdd inv
xinv7 ncinput c Vss Vdd inv
xinv8 ncinput d Vss Vdd inv

vVdd Vdd 0 2.5
vVss Vss 0 0

* A 1 1 0 1 1 0 1
* B 1 0 0 0 0 0 0
* C 1 1 0 0 1 1 0 0
* D 1 1 0 0 1 1 0 0

vmaininput nainput 0 pw(0 2.5v, 100p 0v, 1250p 0v, 1350p 0v, 2500p 0v,
2600p 0v, 3750p 0v, 3850p 2.5v, 5000p 2.5v, 5100p 0v, 6250p 0v, 6350p 0v,
7500p 0v, 7600p 2.5v, 8750p 2.5v, 8850p 0v)
vmainput mbinput 0 pw(0 2.5v, 100p 0v, 1250p 0v, 1350p 2.5v, 2500p
2.5v, 2600p 2.5v, 3750p 2.5v, 3850p 2.5v, 5000p 2.5v, 5100p 0v, 6250p 0v,
6350p 2.5v, 7500p 2.5v, 7600p 2.5v, 8750p 2.5v, 8850p 2.5v)
vmainput mbinput 0 pw(0 2.5v, 100p 0v, 1250p 0v, 1350p 0v, 2500p 0v,
2600p 2.5v, 3750p 2.5v, 3850p 2.5v, 5000p 2.5v, 5100p 0v, 6250p 0v,
6350p 0v, 7500p 0v, 7600p 2.5v, 8750p 2.5v, 8850p 2.5v)
vmainput mbinput 0 pw(0 2.5v, 100p 0v, 1250p 0v, 1350p 0v, 2500p 0v,
2600p 2.5v, 3750p 2.5v, 3850p 2.5v, 5000p 2.5v, 5100p 0v, 6250p 0v,
6350p 0v, 7500p 0v, 7600p 2.5v, 8750p 2.5v, 8850p 2.5v)

.tran ips 10ns

.option post

*Mosfet models from www.mosis.org
*
* DATE: Jun 11/01
* LOT: TI4Y
* DIE: NRArea_Fring
* Temp= 27
.END

.MODEL CMOSN NMOS (LEVEL = 3)
+ TOX = 5.7E-9 NSUB = 1E17 GAMMA = 0.431731
+ PHI = 0.7 VTO = 0.4238262 DELTA = 0
+ UD = 426.6666819 ETA = 0 THETA = 0.178456
+ KP = 2.501048E-4 VMAX = 8.287851E4 KAPPA = 0.1688779
+ RSH = 4.062439E-3  NFS = 1E12  TPG = 1
+ XJ = 3E-7  LD = 3.162278E-11  WD =
+ 1.232881E-8
+ CGDO = 6.2E-10  CGSO = 6.2E-10  CGBO = 1E-10
+ CJ = 1.81211E-3  PB = 0.5  MJ = 0.3282553
+ CJSW = 5.341337E-10  MJSW = 0.5

.*

.MODCAL CMOSP PMOS

+ TOX = 5.7E-9  NSUB = 1E17  GAMMA = 0.6348369
+ PHI = 0.7  VTD = -0.5535085  DELTA = 0
+ UD = 260  ETA = 0  THETA = 0.1573195
+ KP = 5.194153E-5  VMAX = 2.2965325E5  KAPPA = 0.7448494
+ RSH = 30.0976892  NFS = 1E12  TPG = -1
+ XJ = 2E-7  LD = 9.968346E-13  WD =
+ 5.475113E-9
+ CGDO = 6.66E-10  CGSO = 6.66E-10  CGBO = 1E-10
+ CJ = 1.823569E-3  PB = 0.9906013  MJ =
+ 0.4664287
+ CJSW = 3.625544E-10  MJSW = 0.5

.*

.end
dlatch

* SPICE dlatch.sp

.SUBCKT inv A Y Vss Vdd
mn1 Vss A Y Vss cmosn l=0.25u w=0.50u
mp1 Vdd A Y Vdd cmosp l=0.25u w=1.00u
.ENDS

mn1 Vss D int0 Vss cmosn l=0.25u w=0.50u
mp1 Vdd D int0 Vdd cmosp l=0.25u w=1.00u

mn2 int0 clk int1 Vss cmosn l=0.25u w=0.50u
mp2 int1 nclk int0 Vdd cmosp l=0.25u w=1.00u

mn3 Vss int1 int2 Vss cmosn l=0.25u w=0.50u
mp3 Vdd int1 int2 Vdd cmosp l=0.25u w=1.00u

mn4 Vss nclk int4 Vss cmosn l=0.25u w=0.50u
mp4 Vdd nclk int4 Vdd cmosp l=0.25u w=1.00u

mn5 nclk int2 int1 Vss cmosn l=0.25u w=0.50u
mp5 nclk int3 int2 Vdd cmosp l=0.25u w=1.00u

mn6 Vss int3 int4 Vss cmosn l=0.25u w=0.50u
mp6 Vdd int3 int4 Vdd cmosp l=0.25u w=1.00u

xinv1 Q cap1 Vss Vdd inv
xinv2 Q cap2 Vss Vdd inv
xinv3 Q cap3 Vss Vdd inv
xinv4 Q cap4 Vss Vdd inv

xinv5 ndinput d Vss Vdd inv

.vWdd Vdd 0 2.5
.vVdd Vdd 0 2.5

* D I I I 1 0 1 1
* clk 1 0 1 0 1 1 1

.vdinput ndinput 0 pwl(0p 2.5v, 100p 0v, 1250p 0v, 1350p 0v, 2500p 0v,
2600p 0v, 3750p 0v, 3850p 2.5v, 5000p 2.5v, 5100p 0v, 6250p 0v, 6350p
2.5v, 7500p 2.5v, 7600p 0v)

.vnlclkinput nclk 0 pwl(0p 0v, 100p 0v, 1250p 0v, 1350p 2.5v, 2500p
2.5v, 2600p 0v, 3750p 0v, 3850p 2.5v, 5000p 2.5v, 5100p 0v, 6250p 0v,
6350p 0v, 7500p 0v, 7600p 0v)

.vclkinput clk 0 pwl(0p 2.5v, 100p 0v, 1250p 0v, 1350p 2.5v, 2500p
0v, 2600p 2.5v, 3750p 2.5v, 3850p 0v, 5000p 0v, 5100p 2.5v, 6250p
2.5v, 6350p 2.5v, 7500p 2.5v, 7600p 2.5v)

.tran 1ps 8ns

.option post

* Mosfet models from www.mosis.org
* 
* DATE: Jun 11/01
* LOT: T14Y WAF: 03
* DIE: N.Area_Fring DEV: N3740/10
* Temp= 27

.MODEL CMOSN NMOS (
+ TOX = 5.7E-9 NSUB = 1E17 LEVEL = 3
+ PHI = 0.7 VTO = 0.4238252 GAMMA = 0.4317311
+ UD = 425.6468519 ETA = 0 THETA = 0.1754054
+ KP = 2.501048E-4 VMAX = 8.287851E4 KAPPA = 0.1688779
+ RSH = 4.062439E-3 NFS = 1E12 TPG = 1
+ XJ = 3E-7 LD = 3.162278E-11 WD =

101
1.232881E-8
+ CGDO = 6.2E-10    CGSU = 6.2E-10    CGBO = 1E-10
+ CJ = 1.812111E-3   PB = 0.5        MJ = 0.3282553
+ CJSW = 5.341337E-10 MJSW = 0.5
)
.

.MODEL CMOSP PMOS (LEVEL = 3
+ TOX = 5.7E-9    NSUB = 1E17    GAMMA = 0.6348369
+ PHI = 0.7      VTO = -0.5536085  DELTA = 0
+ UD = 250       ETA = 0         THETA = 0.1573195
+ KP = 5.194153E-5 VMAX = 2.293325E5  KAPPA = 0.7448494
+ RSH = 30.0776952 NFS = 1E12   TPG = -1
+ XJ = 2E-7      LD = 9.968346E-13 WD =
5.475113E-9
+ CGDO = 6.66E-10    CGSU = 6.66E-10    CGBO = 1E-10
+ CJ = 1.893569E-3   PB = 0.9906013   MJ =
0.4664287
+ CJSW = 3.625644E-10 MJSW = 0.5
)
dreg

* SPICE dreg.sp

.SUBCKT inv A Y Vss Vdd
mn1 Vss A Y Vss cmosn l=0.25u w=0.50u
mp1 Vdd A Y Vdd cmosp l=0.25u w=1.00u
.ENDS

mn1 Vss D int0 Vss cmosn l=0.25u w=0.50u
mp1 Vdd D int0 Vdd cmosp l=0.25u w=1.00u

mn2 int0 nclk int1 Vss cmosn l=0.25u w=0.50u
mp2 int1 clk int0 Vdd cmosp l=0.25u w=1.00u

mn3 Vss int1 int2 Vss cmosn l=0.25u w=0.50u
mp3 Vdd int1 int2 Vdd cmosp l=0.25u w=1.00u

mn4 Vss clk int4 Vss cmosn l=0.25u w=0.50u
mp4 Vdd nclk int4 Vdd cmosp l=0.25u w=1.00u

mn5 int4 int2 int1 Vss cmosn l=0.25u w=0.50u
mp5 int2 int4 int1 Vdd cmosp l=0.25u w=1.00u

mn6 Vss int1 Qint Vss cmosn l=0.25u w=0.50u
mp6 Vdd int1 Qint Vdd cmosp l=0.25u w=1.00u

mn7 Vss Qint int5 Vss cmosn l=0.25u w=0.50u
mp7 Vdd Qint int5 Vdd cmosp l=0.25u w=1.00u

mn8 int5 clk int6 Vss cmosn l=0.25u w=0.50u
mp8 int6 nclk int5 Vdd cmosp l=0.25u w=1.00u

mn9 Vss int6 int7 Vss cmosn l=0.25u w=0.50u
mp9 Vdd int6 int7 Vdd cmosp l=0.25u w=1.00u

mn10 Vss nclk int9 Vss cmosn l=0.25u w=0.50u
mp10 Vdd clk int9 Vdd cmosp l=0.25u w=1.00u

mn11 int9 int7 int6 Vss cmosn l=0.25u w=0.50u
mp11 int6 int9 int7 Vdd cmosp l=0.25u w=1.00u

mn12 Vss int6 Q Vss cmosn l=0.25u w=0.50u
mp12 Vdd int6 Q Vdd cmosp l=0.25u w=1.00u

xin1v Q cap1 Vss Vdd inv
xin2v Q cap2 Vss Vdd inv
xin3v Q cap3 Vss Vdd inv
xin4v Q cap4 Vss Vdd inv
xin5v ndinput d Vss Vdd inv

vVdd Vdd 0 2.5
vVss Vss 0 0

* D 1 1 1 0 1 0 1
* clk 1 0 0 0 1 1

vndinput ndinput 0 pw1(0p 2.5v, 100p 0v, 1250p 0v, 1350p 0v, 2500p 0v, 2600p 0v, 3750p 0v, 3850p 2.5v, 5000p 2.5v, 6250p 0v, 6350p 2.5v, 7500p 2.5v, 7600p 0v)
vnclockinput nclk 0 pw1(0p 0v, 100p 0v, 1250p 0v, 1350p 2.5v, 2500p 2.5v, 2600p 0v, 3750p 0v, 3850p 2.5v, 5000p 2.5v, 6250p 0v, 6350p 0v, 7500p 0v, 7600p 0v)
vclkinput clk 0 pw1(0p 2.5v, 100p 0v, 1250p 0v, 1350p 2.5v, 2500p 2.5v, 2600p 2.5v, 3750p 2.5v, 3850p 0v, 5000p 0v, 5100p 2.5v, 6250p 0v, 6350p 2.5v, 7500p 2.5v, 7600p 2.5v)
.tran 1ps 8ns

.option post

* Mosfet models from www.mosis.org

* DATE: Jun 11/01
* LOT: T14Y
* DIE: N_Area_Fring
* Temp= 27

.MODEL CMOSN NMOS ( LEVEL = 3
+ TOX = 5.7E-9 NSUB = 1E17 GAMMA = 0.4317311
+ PHI = 0.7 VTO = 0.4238252 DELTA = 0
+ UO = 425.646519 ETA = 0 THETA = 0.1754054
+ KP = 2.501048E-4 VMAX = 8.287851E4 XAPPA = 0.1696779
+ RSH = 4.062439E-3 NFS = 1E12 TPG = 1
+ XJ = 3E-7 LD = 3.162278E-11 WD =
  1.232881E-8
+ CGDO = 6.2E-10 CGSO = 6.2E-10 CGB0 = 1E-10
+ CJ = 1.81211E-3 PB = 0.5 MJ = 0.3282553
+ CJSW = 5.341337E-10 MJSW = 0.5 )

.MODEL CMOSP PMOS ( LEVEL = 3
+ TOX = 5.7E-9 NSUB = 1E17 GAMMA = 0.6348369
+ PHI = 0.7 VTO = -0.5536085 DELTA = 0
+ UO = 250 ETA = 0 THETA = 0.1573195
+ KP = 5.194153E-5 VMAX = 2.295325S5 KAPPA = 0.7448494
+ RSH = 30.0776952 NFS = 1E12 TPG = -1
+ XJ = 2E-7 LD = 9.968346E-13 WD =
  5.475113E-9
+ CGDO = 6.66E-10 CGSO = 6.66E-10 CGB0 = 1E-10
+ CJ = 1.893569E-3 PB = 0.9906013 MJ =
  0.4664287
+ CJSW = 3.62554E-10 MJSW = 0.5 )

* .end
inverter1x

* SPICE inverter1x.sp

.param length=0.25u
.param N0.5=0.50u
.param N1.0=1.00u
.param N1.5=1.50u
.param N8.0=8.00u
.param P1.0=1.00u
.param P2.0=2.00u
.param P3.0=3.00u
.param P16.0=16.00u

.SUBCKT inv A Y Vss Vdd
   m1 Vss A Y Vss cmosn l=0.25u w=N0.5
   mpi Vdd A Y Vdd cmosp l=0.25u w=P1.0
   .ENDS

   m1 Vss A Y Vss cmosn l=0.25u w=N0.5
   mpi Vdd A Y Vdd cmosp l=0.25u w=P1.0

xinv1 Y cap1 Vss Vdd inv
xinv2 Y cap2 Vss Vdd inv
xinv3 Y cap3 Vss Vdd inv
xinv4 Y cap4 Vss Vdd inv

xinv5 nainput a Vss Vdd inv

vVdd Vdd 0 2.5
vVss Vss 0 0

* A I I I O I I O I

vnainput nainput 0 pwl(Ov - 2.5v, 100p - 0v, 1250p 0v,
2500p 0v, 5000p 2.5v, 5100p 0v, 6250p 0v, 6350p
0v, 7500p 0v, 7600p 2.5v, 8750p 2.5v, 8850p 0v)

.tran Ips 8ns

.option post

* Mosfet models from www.mosis.org
*
* DATE: Jun 11/01 WAF: 03
* LOT: T14Y
* DIE: N_Area_Fring DEV: N3740/10
* Temp= 27

.MODEL CMOSN NMOS ( LEVEL = 3
+ TOX = 5.7E-9 NSUB = 1E17 GAMMA = 0.4317311
+ PHI = 0.7 VTO = 0.4238252 DELTA = 0
+ UD = 425.6466519 ETA = 0 THETA = 0.1754054
+ KP = 2.501048E-4 VMAX = 8.287851E4 KAPPA = 0.1866779
+ RSH = 4.062439E-3 NFS = 1E12 TPG = 1
+ XJ = 3E-7 LD = 3.162278E-11 WD = 1.232881E-8
+ CGDO = 6.2E-10 CGSO = 6.2E-10 CBGO = 1E-10
+ CJ = 1.81211E-3 PB = 0.5 MJ = 0.3282553
+ CJSW = 5.341337E-10 MJSW = 0.5 )

.MODEL CMOSP PMOS ( LEVEL = 3
+ TOX = 5.7E-9 NSUB = 1E17 GAMMA = 0.6348369
+ PHI = 0.7 VTO = -0.5536085 DELTA = 0
+ UD = 250 ETA = 0 THETA = 0.1573195
+ KP = 5.194153E-5 VMAX = 2.295325E5 KAPPA = 0.7446494
+ RSH = 30.0776952 NFS = 1E12 TPG = -1

105
+ XJ = 2E-7   LD = 9.968348E-13   WD = 5.475113E-9
+ CGSO = 6.66E-10  CGSO = 6.66E-10  CGSO = 1E-10
+ CJ = 1.893569E-3   PB = 0.9906013   MJ = 0.4664287
+ CJSW = 3.625444E-10  MJ_SW = 0.5 )
* 
.alter * N = 1, 2, 4  P = 4, 8 
.param N0.5=1.00u
.param N1.0=1.00u
.param N1.5=2.00u
.param N8.0=7.00u
.param P1.0=4.00u
.param P2.0=4.00u
.param P3.0=4.00u
.param P16.0=16.00u

.alter * N = 1, 2, 4  P = 2, 4, 8 
.param N0.5=1.00u
.param N1.0=1.00u
.param N1.5=2.00u
.param N8.0=7.00u
.param P1.0=2.00u
.param P2.0=2.00u
.param P3.0=2.00u
.param P16.0=12.00u

.alter * N = 0.5, 1, 4  P = 2, 4, 10 
.param N0.5=2.00u
.param N1.0=2.00u
.param N1.5=2.00u
.param N8.0=6.00u
.param P1.0=2.00u
.param P2.0=2.00u
.param P3.0=2.00u
.param P16.0=16.00u
.end
inverter2x

* SPICE inverter2x.sp

.param length=0.25u
.param N0.5=0.50u
.param N1.0=1.00u
.param N1.5=1.50u
.param N9.0=8.00u
.param P1.0=1.00u
.param P2.0=2.00u
.param P3.0=3.00u
.param P16.0=16.00u

.SUBCKT inv A Y Vss Vdd
mn1 Vss A Y Vss cmosn l=0.25u w=N0.5
mp1 Vdd A Y Vdd cmosp l=0.25u w=P1.0
.ENDS

mn1 Vss A Y Vss cmosn l=0.25u w=N2.0
mp1 Vdd A Y Vdd cmosp l=0.25u w=P4.0

xin1 Y cap1 Vss Vdd inv
xin2 Y cap2 Vss Vdd inv
xin3 Y cap3 Vss Vdd inv
xin4 Y cap4 Vss Vdd inv
xin5 ninput a Vss Vdd inv
vVdd Vdd 0 2.5
vVss Vss 0 0

* A 1 1 1 0 1 1 0 1

vmainput ninput 0 pwl(0 2.5v, 100p 0v, 1250p 0v, 1350p 0v, 2600p 0v, 3750p 0v, 3850p 2.5v, 5000p 2.5v, 5100p 0v, 6250p 0v, 6350p 0v, 7500p 0v, 7600p 2.5v, 8750p 2.5v, 8850p 0v)

.dc vA start=0 stop=2.5 step=0.01
.tran ips 8ns
.option post

* Mosfet models from www.mosis.org

* DATE: Jun 11/01
* LOT: T14Y
* DIE: N_Area_Fring
* Temp= 27

.MODEL CMOSN NMOS (LEVEL = 3
+ TOX = 5.7E-9 NSUB = 1E17 GAMMA = 0.431731
+ PHI = 0.7 VTO = 0.4238252 DELTA = 0
+ UD = 425.6466519 ETA = 0 THETA = 0.1754054
+ KP = 2.501048E-4 VMAX = 8.287851E4 KAPPA = 0.1686779
+ RSH = 4.062439E-3 NFS = 1E12 TPG = 1
+ XJ = 3E-7 LD = 3.162278E-11 WD = 1.223881E-8
+ CGDO = 6.2E-10 CGSO = 6.2E-10 CGBO = 1E-10
+ CJ = 1.81211E-3 PB = 0.5 MJ = 0.3282553
+ CJSW = 5.341337E-10 MJSW = 0.5 )

.MODEL CMOSP PMOS (LEVEL = 3
+ TOX = 5.7E-9 NSUB = 1E17 GAMMA = 0.6348369
+ PHI = 0.7 VTO = -0.5536085 DELTA = 0
+ UD = 250 ETA = 0 THETA = 0.1573195

107
+ KP = 5.194153E-5  VMAX = 2.295325E5  KAPPA = 0.7448494
+ RSH = 30.0776952  NFS = 1E12  TPG = -1
+ XJ = 2E-7  LD = 9.968346E-13  WD = 5.475113E-9
+ CGDO = 6.66E-10  CGSO = 6.66E-10  CGBO = 1E-10
+ CJ = 1.893569E-3  PB = 0.9906013  MJ = 0.4664287
+ CJSW = 3.625544E-10  MJSW = 0.5

* .alter * N = 1, 2, 4  P = 4, 8
.param NO.5=1.00u
.param N1.0=1.00u
.param N1.5=2.00u
.param N8.0=7.00u
.param P1.0=4.00u
.param P2.0=4.00u
.param P3.0=4.00u
.param P16.0=16.00u

.alter * N = 1, 2, 4  P = 2, 4, 8
.param NO.5=1.00u
.param N1.0=1.00u
.param N1.5=2.00u
.param N8.0=7.00u
.param P1.0=2.00u
.param P2.0=2.00u
.param P3.0=2.00u
.param P16.0=12.00u

.alter * N = 0.5, 1, 4  P = 2, 4, 10
.param NO.5=2.00u
.param N1.0=2.00u
.param N1.5=2.00u
.param N8.0=8.00u
.param P1.0=2.00u
.param P2.0=2.00u
.param P3.0=2.00u
.param P16.0=16.00u
.end
inverter4x

* SPICE inverter4x.sp

.param length=0.25u
.param N0.5=0.50u
.param N1.0=1.00u
.param N1.5=1.50u
.param N8.0=8.00u
.param P1.0=1.00u
.param P2.0=2.00u
.param P3.0=3.00u
.param P16.0=16.00u

.SUBCKT inv A Y Vss Vdd
mn1 Vss A Y Vss cmosn l=0.25u w=N0.5
mp1 Vdd A Y Vdd cmosp l=0.25u w=P1.0
.ENDS

mn1 Vss A Y Vss cmosn l=0.25u w=N4.0
mp1 Vdd A Y Vdd cmosp l=0.25u w=P8.0

xinv1 Y cap1 Vss Vdd inv
xinv2 Y cap2 Vss Vdd inv
xinv3 Y cap3 Vss Vdd inv
xinv4 Y cap4 Vss Vdd inv
xinv5 nainput a Vss Vdd inv

cap1 Vss Vdd 0 2.5
vVdd Vdd 0 2.5
vVss Vss 0 0

* A 1 1 0 1 1 0 1

vnanput nainput 0 pwl(Op 2.5v, 100p Ov, 1250p 0v, 1350p 0v, 2500p 0v, 2600p 0v, 3750p 0v, 3850p 2.5v, 5000p 2.5v, 5100p 0v, 6250p 0v, 6350p 0v, 7500p 0v, 7600p 2.5v, 8750p 2.5v, 8850p 0v)

.dc vA start=0 stop=2.5 step=0.01
.tran 1ps 8ns
.option post

* Mosfet models from www.mosis.org
*
* DATE: Jun 11/01
* LOT: Ti4Y
* DIE: N.Area_Fring
* Temp = 27

.MODEL CMOSN NMOS ( LEVEL = 3
+ TOX = 5.7E-9 NSUB = 1E17 GAMMA = 0.4317311
+ PHI = 0.7 VTO = 0.4238252 DELTA = 0
+ UD = 425.6466519 ETA = 0 THETA = 0.1754054
+ KP = 2.501048E-4 VMAX = 8.287851E4 KAPPA = 0.1868779
+ RSH = 4.062439E-3 NFS = 1E12 TPG = 1
+ XJ = 3E-7 LD = 3.162278E-11 WD =
1.232881E-8
+ CGDO = 6.2E-10 CGSO = 6.2E-10 CGB0 = 1E-10
+ CJ = 1.81211E-3 PB = 0.5 MJ = 0.3282553
+ CJSW = 5.341337E-10 MJSW = 0.5 )

.MODEL CMOSP PMOS ( LEVEL = 3
+ TOX = 5.7E-9 NSUB = 1E17 GAMMA = 0.6348369
+ PHI = 0.7 VTO = -0.5536085 DELTA = 0
+ UD = 250 ETA = 0 THETA = 0.1573195
+ KP = 5.194153E-5 VMAX = 2.296325E5 KAPPA = 0.7448494

109
+ RSH = 30.0776952  NFS = 1E12  TPG = -1
+ XJ = 2E-7  LD = 9.968346E-13  WD =
5.475113E-9
+ CGDO = 6.66E-10  CGSO = 6.66E-10  CGBO = 1E-10
+ CJ = 1.893569E-3  PB = 0.9995013  MJ =
0.4664287
+ CJSW = 3.625544E-10  MJSW = 0.5 )
*
.*alter * N = 1, 2, 4  P = 4, 8
.param N0.5=1.00u
.param N1.0=1.00u
.param N1.5=2.00u
.param N8.0=7.00u
.param P1.0=4.00u
.param P2.0=4.00u
.param P3.0=4.00u
.param P16.0=16.00u
.*alter * N = 1, 2, 4  P = 2, 4, 8
.param N0.5=1.00u
.param N1.0=1.00u
.param N1.5=2.00u
.param N8.0=7.00u
.param P1.0=2.00u
.param P2.0=2.00u
.param P3.0=2.00u
.param P16.0=12.00u
.*alter * N = 0.5, 1, 4  P = 2, 4, 10
.param N0.5=2.00u
.param N1.0=2.00u
.param N1.5=2.00u
.param N8.0=8.00u
.param P1.0=2.00u
.param P2.0=2.00u
.param P3.0=2.00u
.param P16.0=16.00u
.end
MUX2

* SPICE mux2.sp

.SUBCKT inv A Y Vss Vdd
m1 Vss A Y Vss cmosn l=0.25u w=0.50u
mp1 Vdd A Y Vdd cmosp l=0.25u w=1.00u
.ENDS

m1 Vss S nS Vss cmosn l=0.25u w=0.50u
mp1 Vdd S nS Vdd cmosp l=0.25u w=1.00u

mn2 Vss nS int0 Vss cmosn l=0.25u w=1.00u
mn3 int0 A nY Vss cmosn l=0.25u w=1.00u
mn4 Vss int1 Vss cmosn l=0.25u w=1.00u
mn5 int1 B nY Vss cmosn l=0.25u w=1.00u

mp2 Vdd S int2 Vdd cmosp l=0.25u w=2.00u
mp3 int2 A nY Vdd cmosp l=0.25u w=2.00u
mp4 Vdd nS int3 Vdd cmosp l=0.25u w=2.00u
mp5 int3 B nY Vdd cmosp l=0.25u w=2.00u

mn6 Vss nY Y Vss cmosn l=0.25u w=0.50u
mp7 Vdd nY Y Vdd cmosp l=0.25u w=1.00u

xinv1 Y cap1 Vss Vdd inv
xinv2 Y cap2 Vss Vdd inv
xinv3 Y cap3 Vss Vdd inv
xinv4 Y cap4 Vss Vdd inv
xinv5 nainput a Vss Vdd inv
xinv6 nbinaryput b Vss Vdd inv
xinv7 nsinput s Vss Vdd inv

vVdd Vdd 0 2.5
vVss Vss 0 0

* A 1 1 i 0 1 1 1
* B 1 0 1 0 1 1 1
* S 1 0 i 0 1 0 1

vninput nainput 0 pwl(0p 2.5v, 100p 0v, 1250p 0v, 1350p 0v, 2500p 0v,
2600p 0v, 3750p 0v, 3850p 2.5v, 5000p 2.5v, 5100p 0v, 6250p 0v, 6350p 0v,
7500p 0v, 7600p 0v)

vbinput nbinaryput 0 pwl(0p 0v, 100p 0v, 1250p 0v, 1350p 2.5v, 2500p
2.5v, 2600p 0v, 3750p 0v, 3850p 2.5v, 5000p 2.5v, 5100p 0v, 6250p 0v,
6350p 0v, 7500p 0v, 7600p 0v)

vnsinput nsinput 0 pwl(0p 0v, 100p 0v, 1250p 0v, 1350p 2.5v, 2500p
2.5v, 2600p 0v, 3750p 0v, 3850p 2.5v, 5000p 2.5v, 5100p 0v, 6250p 0v,
6350p 2.5v, 7500p 2.5v, 7600p 0v)

.tran 1ps 8ns

.option post

*Mosfet models from www.mosis.org
*

* DATE: Jun 11/01
* LOT: T14F
* DIE: N.Area_Fring
* Temp= 27

.Model CMOSN NMOS (LEVEL = 3
+ TOX = 5.7E-9 NSUB = 1E17 GAMMA = 0.431731
+ PHI = 0.7 VTO = 0.4238252 DELTA = 0
+ UD = 426.8466519 ETA = 0 THETA = 0.1754054
+ KP = 2.501048E-4 VMAX = 8.287851E4 KAPPA = 0.1886779

111
* RSH = 4.062439E-3  NFS = 1E12  TPG = 1
* XJ = 3E-7  LD = 3.162278E-11  WD = 1.232881E-8
* CGDO = 6.2E-10  CGSO = 6.2E-10  CBGO = 1E-10
* CJ = 1.81211E-3  PB = 0.5  MJ = 0.3282553
* CJSW = 5.341337E-10  MJSW = 0.5

+ RSH = 4.062439E-3  NFS = 1E12  TPG = 1
+ XJ = 3E-7  LD = 3.162278E-11  WD = 1.232881E-8
+ CGDO = 6.2E-10  CGSO = 6.2E-10  CBGO = 1E-10
+ CJ = 1.81211E-3  PB = 0.5  MJ = 0.3282553
+ CJSW = 5.341337E-10  MJSW = 0.5

.MODEL CMOSP PMOS (LEVEL = 3)
+ TOX = 5.7E-9  NSUB = 1E17  GAMMA = 0.6348369
+ PHI = 0.7  VTO = -0.5536085  DELTA = 0
+ UO = 250  ETA = 0  THETA = 0.1573195
+ XP = 5.194153E-5  VMAX = 2.295325E5  KAPPA = 0.7446494
+ RSH = 30.0776952  NFS = 1E12  TPG = -1
+ XJ = 2E-7  LD = 9.968346E-13  WD = 5.475113E-9
+ CGDO = 6.66E-10  CGSO = 6.66E-10  CBGO = 1E-10
+ CJ = 1.893566E-3  PB = 0.9906013  MJ = 0.5
+ CJSW = 3.625444E-10  MJSW = 0.5

* .end
+ CJSW  =  5.341337E-10  MJSW  =  0.5  )

.model cmosp pmos ( level = 3
+ tox  =  5.7E-9  nsub  =  1e17  gamma  =  0.6348369
+ phi  =  0.7  vto  =  -0.5536085  delta  =  0
+ uo  =  250  eta  =  0  theta  =  0.1573195
+ kp  =  5.194153E-5  vmax  =  2.293325E5  kappa  =  0.7468494
+ rsh  =  30.0776952  nfs  =  1e12  tcg  =  -1
+ xj  =  2E-7  ld  =  9.966346e-13  wd  =
  5.475113e-9
+ cgdo  =  6.66e-10  cgso  =  6.66e-10  cgb0  =  1e-10
+ cj  =  1.893569e-3  fb  =  0.9906013  mj  =
  0.4664287
+ cjsw  =  3.625544e-10  mjsw  =  0.5  )
*
  .alter * n = 1, 2, 4  p = 4, 8
  .param n0.5=1.00u
  .param n1.0=1.00u
  .param n1.5=2.00u
  .param n5.0=7.00u
  .param p1.0=4.00u
  .param p2.0=4.00u
  .param p3.0=4.00u
  .param p16.0=16.00u

  .alter * n = 1, 2, 4  p = 2, 4, 8
  .param n0.5=1.00u
  .param n1.0=1.00u
  .param n1.5=2.00u
  .param n5.0=7.00u
  .param p1.0=2.00u
  .param p2.0=2.00u
  .param p3.0=2.00u
  .param p16.0=12.00u

  .alter * n = 0.5, 1, 4  p = 2, 4, 10
  .param n0.5=2.00u
  .param n1.0=2.00u
  .param n1.5=2.00u
  .param n5.0=8.00u
  .param p1.0=2.00u
  .param p2.0=2.00u
  .param p3.0=2.00u
  .param p16.0=16.00u
.end
nand3

* SPICE nand3.sp

.param length=0.25u
.param NO.5=0.50u
.param N1.0=1.00u
.param N1.5=1.50u
.param N8.0=8.00u
.param P1.0=1.00u
.param P2.0=2.00u
.param P3.0=3.00u
.param P16.0=16.00u

.SUBCKT inv A Y Vss Vdd
m1 Vss A Y Vss cmosn l=length w=N0.5
mpl Vdd A Y Vdd cmosp l=length w=P1.0
.ENDS

m1 Vss A int0 Vss cmosn l=length w=N1.5
mn2 int0 B int1 Vss cmosn l=length w=N1.5
mn3 int1 C Y Vss cmosn l=length w=N1.5

mpl Vdd A Y Vdd cmosp l=length w=P1.0
mp2 Vdd B Y Vdd cmosp l=length w=P1.0
mp3 Vdd C Y Vdd cmosp l=length w=P1.0

xinv1 y cap1 Vss Vdd inv
xinv2 y cap2 Vss Vdd inv
xinv3 y cap3 Vss Vdd inv
xinv4 y cap4 Vss Vdd inv

xinv5 m ainput a Vss Vdd inv
xinv6 n binput b Vss Vdd inv
xinv7 n cinput c Vss Vdd inv

vVdd Vdd 0 2.5
vVss Vss 0 0

* A 1 1 1 0 1 1 1
* B 1 0 1 0 1 1 1
* C 1 1 0 1 0 1 1

vmainput mainput 0 pwl(0p 2.5v, 100p 0v, 1250p 0v, 1350p 0v, 2500p 0v,
2600p 0v, 3750p 0v, 3850p 2.5v, 5000p 2.5v, 5100p 0v, 6250p 0v, 6350p
0v, 7500p 0v, 7600p 0v)

vmainput 0 pwl(0p 0v, 100p 0v, 1250p 0v, 1350p 2.5v, 2500p
2.5v, 2600p 0v, 3750p 0v, 3850p 2.5v, 5000p 2.5v, 5100p 0v, 6250p 0v, 6350p
0v, 7500p 0v, 7600p 0v)

vmainput 0 pwl(0p 0v, 100p 0v, 1250p 0v, 1350p 2.5v, 2500p
2.5v, 2600p 0v, 3750p 0v, 3850p 2.5v, 5000p 2.5v, 5100p 0v, 6250p 0v, 6350p
2.5v, 7500p 2.5v, 7600p 0v)

.tran 1ps 8ns

.option post

* Mosfet models from www.mosis.org
  *
  * DATE: Jun 11/01
  * LOT: T14Y
  * DIE: N.Area_Fring
  * Dev: N3740/10
  * Temp: 27
  .MODEL CMOSN NMOS
  LEVEL = 3
  + TOX = 5.7E-9
  NSUB = 1E17
  GAMMA = 0.4317311
\[ \begin{align*}
+ PHI &= 0.7 & VTO &= 0.4238252 & DELTA &= 0 \\
+ UO &= 425.6466519 & ETA &= 0 & THETA &= 0.1754054 \\
+ KP &= 2.501048E-4 & VMAX &= 8.28781E4 & KAPPA &= 0.1686779 \\
+ RSH &= 4.052439E-3 & NFS &= 1E12 & TPG &= 1 \\
+ XJ &= 3E-7 & LD &= 3.1622E8-11 & WD &= \\
\text{1.232881E-8} \\
+ CGDO &= 6.2E-10 & CGSO &= 6.2E-10 & CGBO &= 1E-10 \\
+ CJ &= 1.8121E-3 & PB &= 0.5 & MJ &= 0.3282553 \\
+ CJSW &= 5.3413E-10 & MJSW &= 0.5 & \\
\end{align*} \]

```
.MODEL CMOSP PMOS 
LEVEL = 3 
+ TOX = 5.7E-9 & NSUB = 1E17 & GAMMA = 0.6348369 \\
+ PHI = 0.7 & VTO = -0.5536085 & DELTA = 0 \\
+ UO = 250 & ETA = 0 & THETA = 0.1573195 \\
+ KP = 5.194153E-5 & VMAX = 2.296325E5 & KAPPA = 0.7448494 \\
+ RSH = 30.0776952 & NFS = 1E12 & TPG = -1 \\
+ XJ = 2E-7 & LD = 9.968346E-13 & WD = \\
\text{5.47513E-9} \\
+ CGDO = 6.66E-10 & CGSO = 6.66E-10 & CGBO = 1E-10 \\
+ CJ = 1.893569E-3 & PB = 0.9906013 & MJ = \\
\text{0.4664287} \\
+ CJSW = 3.625544E-10 & MJSW = 0.5 & \\
```

```
* 
.alter * N = 1, 2, 4  P = 4, 8 
.param NO.5=1.00u 
.param N1.0=1.00u 
.param N1.5=2.00u 
.param N8.0=7.00u 
.param P1.0=4.00u 
.param P1.0=4.00u 
.param P3.0=4.00u 
.param P16.0=16.00u 

.alter * N = 1, 2, 2  P = 2, 4, 8 
.param NO.5=1.00u 
.param N1.0=1.00u 
.param N1.5=2.00u 
.param N8.0=7.00u 
.param P1.0=2.00u 
.param P2.0=2.00u 
.param P3.0=2.00u 
.param P16.0=12.00u 

.alter * N = 0.5, 1, 4  P = 2, 4, 10 
.param NO.5=2.00u 
.param N1.0=2.00u 
.param N1.5=2.00u 
.param N8.0=8.00u 
.param P1.0=2.00u 
.param P2.0=2.00u 
.param P3.0=2.00u 
.param P16.0=16.00u 
.end
```
nand4

* SPICE nand4.sp

.param length=0.25u
.param N0.5=0.50u
.param N1.0=1.00u
.param N1.5=1.50u
.param N2.0=2.00u
.param N8.0=8.00u
.param P1.0=1.00u
.param P2.0=2.00u
.param P3.0=3.00u
.param P16.0=16.00u

.SUBCKT inv A Y Vss Vdd
mn1 Vss A Y Vss cmosn l=length w=N0.5
mp1 Vdd A Y Vdd cmosp l=length w=P1.0
.ENDS

mn1 Vss A int0 Vss cmosn l=length w=N2.0
mn2 int0 B int1 Vss cmosn l=length w=N2.0
mn3 int1 C int2 Vss cmosn l=length w=N2.0
mn4 int2 D Y Vss cmosn l=length w=N2.0

mp1 Vdd A Y Vdd cmosp l=length w=P1.0
mp2 Vdd B Y Vdd cmosp l=length w=P1.0
mp3 Vdd C Y Vdd cmosp l=length w=P1.0
mp4 Vdd D Y Vdd cmosp l=length w=P1.0

xinv1 Y cap1 Vss Vdd inv
xinv2 Y cap2 Vss Vdd inv
xinv3 Y cap3 Vss Vdd inv
xinv4 Y cap4 Vss Vdd inv

xinv5 nainput a Vss Vdd inv
xinv6 nbinput b Vss Vdd inv
xinv7 ncinput c Vss Vdd inv
xinv8 ndinput d Vss Vdd inv

vVdd Vdd 0 2.5
vVss Vss 0 0

• A 1 0 1 0 1 0 1 0 1
• B 1 1 1 0 1 0 1 0 1
• C 1 1 1 1 1 0 1 0 1
• D 1 1 1 1 1 1 1 0 1

vainput nainput 0 pwl(Op 2.5v, 100p 0v, 1250p 0v, 1350p 2.5v, 2500p 2.5v, 2600p 0v, 3750p 0v, 3850p 2.5v, 5000p 2.5v, 5100p 0v, 6250p 0v, 6350p 2.5v, 7500p 2.5v, 7600p 0v, 8750p 0v, 8850p 2.5v, 10000p 2.5v, 10100p 0v)

vbinput nbinput 0 pwl(Op 0v, 100p 0v, 1250p 0v, 1350p 0v, 2500p 0v, 2600p 0v, 3750p 0v, 3850p 2.5v, 5000p 2.5v, 5100p 0v, 6250p 0v, 6350p 2.5v, 7500p 2.5v, 7600p 0v, 8750p 0v, 8850p 2.5v, 10000p 2.5v, 10100p 0v)

vncinput ncinput 0 pwl(Op 0v, 100p 0v, 1250p 0v, 1350p 0v, 2500p 0v, 2600p 0v, 3750p 0v, 3850p 0v, 5000p 0v, 5100p 0v, 6250p 0v, 6350p 2.5v, 7500p 2.5v, 7600p 0v, 8750p 0v, 8850p 2.5v, 10000p 2.5v, 10100p 0v)

vndinput ndinput 0 pwl(Op 0v, 100p 0v, 1250p 0v, 1350p 0v, 2500p 0v, 2600p 0v, 3750p 0v, 3850p 0v, 5000p 0v, 5100p 0v, 6250p 0v, 6350p 0v, 7500p 0v, 7600p 0v, 8750p 0v, 8850p 2.5v, 10000p 2.5v, 10100p 0v)
* Mosfet models from www.mosis.org *

```
* DATE: Jun 11/01
* LOT: T14Y
* DIE: N_Area_Fring
* Temp= 27

.MODEL CMOSN NMOS (LEVEL = 3)
+ TOX = 5.7E-9 NSUB = 1E17 GAMMA = 0.4317311
+ PHI = 0.7 VTO = 0.4238252 DELTA = 0
+ UO = 425.6466519 ETA = 0 THETA = 0.1754054
+ RSH = 4.062439E-3 NFS = 1E12 TPG = 1
+ XJ = 3E-7 LD = 3.162278E-11 WD = 1.232881E-8
+ CGDO = 6.2E-10 CGSO = 6.2E-10 CGBO = 1E-10
+ CJ = 1.81211E-3 PB = 0.5 MJ = 0.3282553
+ CJSW = 5.341337E-10 MJSW = 0.5 )

.MODEL CMOSP PMOS (LEVEL = 3)
+ TOX = 5.7E-9 NSUB = 1E17 GAMMA = 0.6348369
+ PHI = 0.7 VTO = -0.5536085 DELTA = 0
+ UO = 250 ETA = 0 THETA = 0.1573195
+ RSH = 5.194153E-5 VMAX = 2.295325E5 KAPPA = 0.7448949
+ RSH = 30.0776952 NFS = 1E12 TPG = -1
+ XJ = 2E-7 LD = 9.988346E-13 WD = 6.475113E-9
+ CGDO = 6.66E-10 CGSO = 6.66E-10 CGBO = 1E-10
+ CJ = 1.893569E-3 PB = 0.9906013 MJ = 0.4664287
+ CJSW = 3.628544E-10 MJSW = 0.5 )

* .alter * N = 1, 2, 4 P = 4, 8
  .param N0.5=1.00u
  .param N1.0=1.00u
  .param N1.5=2.00u
  .param N2.0=2.00u
  .param N8.0=7.00u
  .param N8.0=7.00u
  .param P1.0=4.00u
  .param P2.0=4.00u
  .param P3.0=4.00u
  .param P16.0=16.00u

* .alter * N = 1, 2, 4 P = 2, 4, 8
  .param N0.5=1.00u
  .param N1.0=1.00u
  .param N1.5=2.00u
  .param N2.0=2.00u
  .param N8.0=7.00u
  .param P1.0=2.00u
  .param P2.0=2.00u
  .param P3.0=2.00u
  .param P16.0=12.00u

* .alter * N = 0.5, 1, 4 P = 2, 4, 10
  .param N0.5=2.00u
  .param N1.0=2.00u
  .param N1.5=2.00u
  .param P1.0=2.00u
  .param P2.0=2.00u
  .param P3.0=2.00u

118
```
.param P16.0=16.00u
.end
nor2

* SPICE nor2.sp

.SUBCKT inv A Y Vss Vdd
mn1 Vss A Y Vss_cmos 1=0.25u w=0.50u
mp1 Vdd A Y Vdd_cmos 1=0.25u w=1.00u
.ENDS

mn1 Vss A Y Vss_cmos 1=0.25u w=0.50u
mn2 Vss B Y Vss_cmos 1=0.25u w=0.50u

mp1 Vdd A int0 Vdd_cmos 1=0.25u w=2.00u
mp2 int0 B Y Vdd_cmos 1=0.25u w=2.00u

xinv1 Y cap1 Vss Vdd inv
xinv2 Y cap2 Vss Vdd inv
xinv3 Y cap3 Vss Vdd inv
xinv4 Y cap4 Vss Vdd inv

xinv5 maininput a Vss Vdd inv
xinv6 maininput b Vss Vdd inv

vVdd Vdd 0 2.5
vVss Vss 0 0

* A 1 0 1 0 1 0
* B 1 0 0 0 1 0

vmaininput maininput 0 pwl(0p 2.5v, 100p 0v, 1250p 0v, 1350p 2.5v, 2500p
2.5v, 2600p 0v, 3750p 0v, 3850p 2.5v, 5000p 2.5v, 5100p 0v, 6250p 0v,
6350p 0v, 7600p 0v, 7600p 2.5v)
vnbinput ninput 0 pwl(0p 2.5v, 100p 0v, 1250p 0v, 1350p 2.5v, 2500p
2.5v, 2600p 2.5v, 3750p 2.5v, 3850p 2.5v, 5000p 2.5v, 5100p 0v, 6250p
0v, 6350p 2.5v, 7500p 2.5v, 7600p 2.5v)

.tran 1ps 8ns

.option post

* Mosfet models from www.mosis.org

* DATE: Jun 11/01
* LOT: TI4Y
* DIE: AreaFring
* Temp = 27

.MODEL CMOSN NMOS
+ TOX = 5.7E-9
+ PHI = 0.7
+ UH = 425.645519
+ KP = 2501048E-4
+ RSH = 5.194153E-5
+ RSH = 1.232881E-8
+ XJ = 3E-7
+ CGSO = 6.2E-10
+ CJ = 1.81211E-3
+ CJSW = 5.341337E-10

.MODEL CMOSP PMOS
+ TOX = 5.7E-9
+ PHI = 0.7
+ UH = 425.645519
+ KP = 5.194153E-5
+ RSH = 30.09759E2
+ XJ = 2E-7
+ CGSO = 6.2E-10
+ CJ = 1.81211E-3
+ CJSW = 5.341337E-10

120
CGDO = 6.66E-10  CGSO = 6.66E-10  CGRO = 1E-10
+ CJ = 1.893569E-3  PB = 0.9906013  MJ = 0.4664287
+ CJSW = 3.625544E-10  MJSW = 0.5  )
* 
.end
nor3

* SPICE nor3.sp

.SUBCKT inv A Y Vss Vdd
m1 Vss A Y Vss cmosn l=0.25u w=0.50u
mp1 Vdd A Y Vdd cmosp l=0.25u w=1.00u
.ENDS

m1 Vss A Y Vss cmosn l=0.25u w=0.50u
m2 Vss B Y Vss cmosn l=0.25u w=0.50u
m3 Vss C Y Vss cmosn l=0.25u w=0.50u

mp1 Vdd A int0 Vdd cmosp l=0.25u w=3.0u
mp2 int0 B int1 Vdd cmosp l=0.25u w=3.0u
mp3 int1 C Y Vdd cmosp l=0.25u w=3.0u

xinv1 Y cap1 Vss Vdd inv
xinv2 Y cap2 Vss Vdd inv
xinv3 Y cap3 Vss Vdd inv
xinv4 Y cap4 Vss Vdd inv

xinv5 nainput a Vss Vdd inv
xinv6 nbinput b Vss Vdd inv
xinv7 ncinput c Vss Vdd inv

vVdd Vdd 0 2.5
evVss Vss 0 0

* A 1 0 1 0 1 1 0 1
* B 1 0 0 0 1 0 0 0
* C 1 0 0 0 1 1 0 1

vmainput nainput 0 pw1(0p 2.5v, 100p 0v, 1250p 0v, 1350p 2.5v, 2500p
2.5v, 2600p 0v, 3750p 0v, 3850p 2.5v, 5000p 2.5v, 5100p 0v, 6250p 0v,
6350p 0v, 7600p 0v, 7600p 2.5v, 8750p 2.5v, 8850p 0v)

vnbinput nbinput 0 pw1(0p 0v, 100p 0v, 1250p 0v, 1360p 2.5v, 2500p
2.5v, 2600p 2.5v, 3750p 2.5v, 3850p 2.5v, 5000p 2.5v, 5100p 0v, 6250p 0v,
6350p 2.5v, 7600p 2.5v, 8750p 2.5v, 8850p 2.5v)

vncinput ncinput 0 pw1(0p 2.5v, 100p 0v, 1250p 0v, 1350p 2.5v, 2500p
2.5v, 2600p 2.5v, 3750p 2.5v, 3850p 2.5v, 5000p 2.5v, 5100p 0v, 6250p 0v,
6350p 0v, 7600p 2.5v, 8750p 2.5v, 8850p 0v)

.tran 1ps 10ns

.option post

#Mosfet models from www.mosis.org

* DATE: Jun 11/01
* LOT: T14Y WAF: 03
* DIE: N_Area_Fring DEV: N3740/10
* Temp= 27

.MODEL CMOSN NMOS (LEVEL = 3
+ TOX = 5.75E-9 NSUB = 1E17
+ PHI = 0.7 VTO = 0.4238252
+ UO = 425.6466519 ETA = 0
+ KP = 2.501048E-4 VMAX = 8.287851E4
+ RSH = 4.062439E-3 NFS = 1E12
+ RJ = 3E-7 LD = 3.162278E-11
1.232881E-8
+ CGDO = 6.2E-10 CGSO = 6.2E-10 CGRG = 1E-10
+ C1 = 1.81211E-3 PB = 0.5
+ CJSW = 5.341337E-10 MJSW = 0.5
)
.MODEL CMOSP PMOS (LEVEL = 3
+ TOX = 5.7E-9 NSUB = 1E17 GAMMA = 0.6348369
+ PHI = 0.7 VTO = -0.5536085 DELTA = 0
+ UD = 250 ETA = 0 THETA = 0.1573195
+ KP = 5.194153E-5 VMAX = 2.29532E5 KAPPA = 0.7448494
+ RSH = 30.0776952 NFS = 1E12 TPG = -1
+ XJ = 2E-7 LD = 9.968346E-13 WD =
5.475113E-9
+ CGDO = 6.66E-10 CGSO = 6.66E-10 CGBO = 1E-10
+ CJ = 1.893669E-3 PB = 0.9906013 MJ =
0.4664287
+ CJSW = 3.625544E-10 MJSW = 0.5 )
*
.end
nor4

* SPICE nor4.sp

.SUBCKT inv A Y Vss Vdd
  mn1 Vss A Y Vss cmosn l=0.25u w=0.50u
  mp1 Vdd A Y Vdd cmosp l=0.25u w=1.00u
.ENDS

  mn1 Vss A Y Vss cmosn l=0.25u w=1.00u
  mn2 Vss B Y Vss cmosn l=0.25u w=1.00u
  mn3 Vss C Y Vss cmosn l=0.25u w=1.00u
  mn4 Vss D Y Vss cmosn l=0.25u w=1.00u

  mp1 Vdd A int0 Vdd cmosp l=0.25u w=4.00u
  mp2 int0 B int1 Vdd cmosp l=0.25u w=4.00u
  mp3 int1 C int2 Vdd cmosp l=0.25u w=4.00u
  mp4 int2 D Y Vdd cmosp l=0.25u w=4.00u

  xinv1 Y cap1 Vss Vdd inv
  xinv2 Y cap2 Vss Vdd inv
  xinv3 Y cap3 Vss Vdd inv
  xinv4 Y cap4 Vss Vdd inv

  xinv5 nainput a Vss Vdd inv
  xinv6 nbinput b Vss Vdd inv
  xinv7 ncinput c Vss Vdd inv
  xinv8 ndinput d Vss Vdd inv

  vVdd Vdd 0 2.5
  vVes Vss 0 0

  * A 0 0 0 0 0 0 0 1 0
  * B 0 0 0 0 0 1 0 1 0
  * C 0 0 0 1 0 1 0 1 0
  * D 0 1 0 1 0 1 0 1 0

  vinainput nainput 0 pwl(Op 2.5v, 100p 2.5v, 1250p 2.5v, 1350p 2.5v, 2500p 2.5v, 2600p 2.5v, 2850p 2.5v, 3850p 2.5v, 5000p 2.5v, 5100p 2.5v, 6250p 2.5v, 6350p 2.5v, 7500p 2.5v, 7600p 2.5v, 8750p 2.5v, 8850p 0v, 10000p 0v, 10100p 2.5v)

  vbinput nbinput 0 pwl(0p 2.5v, 100p 2.5v, 1250p 2.5v, 1350p 2.5v, 2500p 2.5v, 2600p 2.5v, 2850p 2.5v, 3850p 2.5v, 5000p 2.5v, 5100p 2.5v, 6250p 2.5v, 6350p 0v, 7500p 2.5v, 7600p 2.5v, 8750p 2.5v, 8850p 0v, 10000p 0v, 10100p 2.5v)

  vcinput ncinput 0 pwl(0p 2.5v, 100p 2.5v, 1250p 2.5v, 1350p 2.5v, 2500p 2.5v, 2600p 2.5v, 2850p 2.5v, 3850p 0v, 5000p 0v, 5100p 2.5v, 6250p 2.5v, 6350p 0v, 7500p 2.5v, 7600p 2.5v, 8750p 2.5v, 8850p 0v, 10000p 0v, 10100p 2.5v)

  vndinput ndinput 0 pwl(0p 2.5v, 100p 2.5v, 1250p 2.5v, 1350p 0v, 2500p 0v, 2600p 2.5v, 2750p 2.5v, 3750p 2.5v, 3850p 0v, 5000p 0v, 5100p 2.5v, 6250p 2.5v, 6350p 0v, 7500p 2.5v, 7600p 2.5v, 8750p 2.5v, 8850p 0v, 10000p 0v, 10100p 2.5v)

.tran ips 12ns

.option post

* Mosfet models from www.mosis.org

* DATE: Jun 11/01
* LOT: T14Y
* D1E: N.Area_Fring
* Temp= 27
.MODEL CMOSN NMOS ( LEVEL = 3

124
+ TOX = 5.7E-9 NSUB = 1E17 GAMMA = 0.4317311
+ PHI = 0.7 VT0 = 0.4238252 DELTA = 0
+ UD = 425.6466519 ETA = 0 THETA = 0.1754054
+ KP = 2.501048E-4 VMAX = 8.287851E4 RAPPA = 0.1686779
+ RSH = 4.062439E-3 NF = 1E12 TPG = 1
+ XJ = 3E-7 LD = 3.162278E-11 WD = 1.232881E-8
+ CGDO = 6.62E-10 CGSO = 6.2E-10 CGBO = 1E-10
+ CJ = 1.81211E-3 PB = 0.5 MJ = 0.3282553
+ CJSW = 5.341337E-10 MJSW = 0.5 )

.CODE CMOUS PMS ( LEVEL = 3
+ TOX = 5.7E-9 NSUB = 1E17 GAMMA = 0.6348369
+ PHI = 0.7 VT0 = -0.5536085 DELTA = 0
+ UD = 250 ETA = 0 THETA = 0.1573196
+ KP = 5.194153E-5 VMAX = 2.295325E5 RAPPA = 0.7448494
+ RSH = 30.0776952 NF = 1E12 TPG = -1
+ XJ = 2E-7 LD = 9.963466E-13 WD = 5.475113E-9
+ CGDO = 6.66E-10 CGSO = 6.66E-10 CGBO = 1E-10
+ CJ = 1.893569E-3 PB = 0.9906013 MJ = 0.4664287
+ CJSW = 3.625544E-10 MJSW = 0.5 )

.end
.

.SUBCKT inv A Y Vss Vdd
mm1 Vss A Y Vss cmosn l=0.25u w=0.50u
mp1 Vdd A Y Vdd cmosp l=0.25u w=1.00u
.ENDS

mm1 Vss A int0 Vss cmosn l=0.25u w=2.00u
mm2 Vss B int0 Vss cmosn l=0.25u w=2.00u
mm3 int0 C Y Vss cmosn l=0.25u w=2.00u

mp1 Vdd A int1 Vdd cmosp l=0.25u w=4.00u
mp2 int1 B Y Vdd cmosp l=0.25u w=4.00u
mp3 Vdd C Y Vdd cmosp l=0.25u w=2.00u

vVdd Vdd 0 2.5
vVss Vss 0 0

xinv1 Y cap1 Vss Vdd inv
xinv2 Y cap2 Vss Vdd inv
xinv3 Y cap3 Vss Vdd inv
xinv4 Y cap4 Vss Vdd inv
xinv5 minput a Vss Vdd inv
xinv6 nbinput b Vss Vdd inv
xinv7 ncinput c Vss Vdd inv

* A 1 1 1 0 1 1 0 1
* B 1 0 0 0 1 0 0 0
* C 1 1 0 0 1 1 0 0

vminput minput a 0 pw1(0 2.5v, 100p 0v, 1250p 0v, 1350p 0v, 2500p 0v, 2600p 0v, 3750p 0v, 3850p 2.5v, 5000p 2.5v, 5100p 0v, 6250p 0v, 6350p 0v, 7500p 0v, 7650p 2.5v, 7850p 2.5v, 8850p 0v)

vminput nbinput b 0 pw1(0 2.5v, 100p 0v, 1250p 0v, 1350p 2.5v, 2500p 2.5v, 2600p 2.5v, 3750p 2.5v, 3850p 2.5v, 5000p 2.5v, 5100p 0v, 6250p 0v, 6350p 2.5v, 7500p 2.5v, 7600p 2.5v, 8750p 2.5v, 8850p 2.5v)

vncinput ncinput c 0 pw1(0 2.5v, 100p 0v, 1250p 0v, 1350p 0v, 2500p 0v, 2600p 2.5v, 3750p 2.5v, 3850p 2.5v, 5000p 2.5v, 5100p 0v, 6250p 0v, 6350p 0v, 7500p 0v, 7650p 2.5v, 8750p 2.5v, 8850p 2.5v)

.tran lps lOns

.option post

* Mosfet models from www.mosis.org

* DATE: Jun 11/01
* LOT: T14Y
* DIE: N_Area_Fring
* Temp= 27

.MODEL CMOSN NMOS LEVEL = 3
+ TOX = 5.7E-9 NSUB = 1E17 GAMMA = 0.431731
+ PHI = 0.7 VTD = 0.4238252 DELTA = 0
+ U0 = 425.646619 ETA = 0 THETA = 0.1754064
+ KP = 2.50104BE-4 VMAX = 8.287851E4 KAPPA = 0.1686779
+ RSH = 4.062439E-3 NFS = 1E12 TPG = 1
+ XJ = 3E-7 LD = 3.162278E-11 WD = 1.2328818E-8
+ CGDO = 6.2E-10 CGS0 = 6.2E-10 CGBO = 1E-10
+ CJ = 1.81211E-3 PB = 0.5 MJ = 0.3282563
+ CJSW = 5.341397E-10 MJSW = 0.5

126
.MODEL CMOSP PMOS
+ TOX = 5.7E-9  NSUB = 1E17  LEVEL = 3
+ PHI = 0.7  VTO = -0.5536085  GAMMA = 0.6348369
+ UD = 250  ETA = 0  DELTA = 0
+ KP = 5.194153E-5  VMAX = 2.295325E5  THETA = 0.1573195
+ RSH = 30.0776952  NFS = 1E12  KAPPA = 0.7448494
+ XJ = 2E-7  LD = 9.968346E-13  TPG = 0
+ CDDO = 6.66E-10  CGSO = 6.66E-10  WD = 5.475113E-9
+ CJ = 1.893669E-3  PB = 0.9906013  CGBO = 1E-10
+ CJSW = 3.626544E-10  MJ = 0.4664287
+ CJSW = 3.626544E-10  MJ = 0.5
*
.end
**SPICE oai211.sp**

`.SUBCKT inv A Y Vss Vdd
mn1 Vss A Y Vss cmosn l=0.25u w=0.50u
mp1 Vdd A Y Vdd cmosp l=0.25u w=1.00u
.ENDS

mn1 Vss A int0 Vss cmosn l=0.25u w=1.00u
mn2 Vss B int0 Vss cmosn l=0.25u w=1.00u
mn3 int0 C int1 Vss cmosn l=0.25u w=1.00u
mn4 int1 D Y Vss cmosn l=0.25u w=1.00u

mp1 Vdd A int2 Vdd cmosp l=0.25u w=6.00u
mp2 int2 B Y Vdd cmosp l=0.25u w=6.00u
mp3 Vdd C Y Vdd cmosp l=0.25u w=6.00u
mp4 Vdd D Y Vdd cmosp l=0.25u w=6.00u

xinv1 Y cap1 Vss Vdd inv
xinv2 Y cap2 Vss Vdd inv
xinv3 Y cap3 Vss Vdd inv
xinv4 Y cap4 Vss Vdd inv

xinv5 nainput a Vss Vdd inv
xinv6 nbinput b Vss Vdd inv
xinv7 ncinput c Vss Vdd inv
xinv8 ndinput d Vss Vdd inv

vVdd Vdd 0 2.5
vVss Vss 0 0

* A 1 1 1 0 1 1 0 1
* B 1 0 0 0 1 0 0 0
* C 1 1 0 0 1 1 0 0
* D 1 1 0 0 1 1 0 0

vmainput nainput 0 pwl(0p 2.5v, 100p 0v, 1250p 0v, 1350p 0v, 2500p 0v, 2600p 0v, 3750p 0v, 3850p 2.5v, 5000p 2.5v, 5100p 0v, 6250p 0v, 6350p 0v, 7500p 0v, 7600p 2.5v, 8750p 2.5v, 8850p 0v)

vnbinput nbinput 0 pwl(0p 2.5v, 100p 0v, 1250p 0v, 1350p 2.5v, 2500p 2.5v, 2600p 2.5v, 3750p 2.5v, 3850p 2.5v, 5000p 2.5v, 5100p 0v, 6250p 0v, 6350p 2.5v, 7500p 2.5v, 7600p 2.5v, 8750p 2.5v, 8850p 2.5v)

vncinput ncinput 0 pwl(0p 2.5v, 100p 0v, 1250p 0v, 1350p 0v, 2500p 0v, 2600p 2.5v, 3750p 2.5v, 3850p 2.5v, 5000p 2.5v, 5100p 0v, 6250p 0v, 6350p 0v, 7500p 0v, 7600p 2.5v, 8750p 2.5v, 8850p 2.5v)

vndinput ndinput 0 pwl(0p 2.5v, 100p 0v, 1250p 0v, 1350p 0v, 2500p 0v, 2600p 2.5v, 3750p 2.5v, 3850p 2.5v, 5000p 2.5v, 5100p 0v, 6250p 0v, 6350p 0v, 7500p 0v, 7600p 2.5v, 8750p 2.5v, 8850p 2.5v)

.tran 1ps 10ns

.option post

*Mosfet models from www.mosis.org
*
* DATE: Jun 11/01
* LOT: T14Y WAF: 03
* DIE: N_Area_Fring DEV: N3740/10
* Temp= 27

.MODEL CMOSN NMOS (LEVEL = 3
+ TOX = 5.7E-9 NSUB = 1E17 GAMMA = 0.4317311
+ PHI = 0.7 VTO = 0.4238252 DELTA = 0
+ UD = 0.6466519 ETA = 0 THETA = 0.1754054
+ XP = 2.501048E-4 VMAX = 8.287851E4 KAPPA = 0.1688779
+ RSH = 4.062439E-3  NFS = 1E12  TPG = 1
+ XJ  = 3E-7  LD = 3.162278E-11  WD =
  1.232881E-8
+ CGDO = 6.2E-10  CGSO = 6.2E-10  CGBO = 1E-10
+ CJ  = 1.81211E-3  PB = 0.8  MJ = 0.3282553
+ CJSW = 5.341337E-10  MJSW = 0.5  )

.MODEL CMOSP PMOS (LEVEL = 3
+ TOX = 5.7E-9  NSUB = 1E17  GAMMA = 0.6348369
+ PHI  = 0.7  VTO = -0.5536085  DELTA = 0
+ UO  = 250  ETA = 0  THETA = 0.1573195
+ KP = 5.194153E-5  VMAX = 2.295325E5  KAPPA = 0.7448494
+ RSH = 30.0776952  NFS = 1E12  TPG = -1
+ XJ  = 2E-7  LD = 9.968346E-13  WD =
  5.475113E-9
+ CGDO = 6.66E-10  CGSO = 6.66E-10  CGBO = 1E-10
+ CJ  = 1.693669E-3  PB = 0.9906013  MJ =
  0.4664287
+ CJSW = 3.625564E-10  MJSW = 0.5  )
*
.end
* SPICE oai22.sp

.SUBCKT inv A Y Vss Vdd
mn1 Vss A Y Vss cmosn l=0.25u w=0.50u
mp1 Vdd A Y Vdd cmosp l=0.25u w=1.00u
.ENDS

mn1 Vss A int0 Vss cmosn l=0.25u w=1.00u
mn2 Vss B int0 Vss cmosn l=0.25u w=1.00u
mn3 int0 C Y Vss cmosn l=0.25u w=1.00u
mn4 int0 D Y Vss cmosn l=0.25u w=1.00u

mp1 Vdd A int1 Vdd cmosp l=0.25u w=6.00u
mp2 int1 B Y Vdd cmosp l=0.25u w=6.00u
mp3 Vdd C int2 Vdd cmosp l=0.25u w=6.00u
mp4 int2 D Y Vdd cmosp l=0.25u w=6.00u

xinv1 Y cap1 Vss Vdd inv
xinv2 Y cap2 Vss Vdd inv
xinv3 Y cap3 Vss Vdd inv
xinv4 Y cap4 Vss Vdd inv
xinv5 nainput a Vss Vdd inv
xinv6 nbinput b Vss Vdd inv
xinv7 ncinput c Vss Vdd inv
xinv8 ndinput d Vss Vdd inv

vVdd Vdd 0 2.5
vVss Vss 0 0

* A 1 1 1 0 1 1 0 1
* B 1 0 0 0 1 0 0 0
* C 1 1 0 0 1 1 0 0
* D 1 1 0 0 1 1 0 0

vnainput nainput 0 pwl(0p 2.5v, 100p 0v, 1250p 0v, 1350p 0v, 2600p 0v, 2600p 0v, 3750p 0v, 3850p 2.5v, 5000p 2.5v, 5100p 0v, 6250p 0v, 6350p 0v, 7500p 0v, 7600p 2.5v, 8750p 2.5v, 8850p 0v)

vninput nbinput 0 pwl(0p 2.5v, 100p 0v, 1250p 0v, 1350p 2.5v, 2500p 2.5v, 2600p 2.5v, 3750p 2.5v, 3850p 2.5v, 5000p 2.5v, 5100p 0v, 6250p 0v, 6350p 2.5v, 7500p 2.5v, 7600p 2.5v, 8750p 2.5v, 8850p 2.5v)

vncinput ncinput 0 pwl(0p 2.5v, 100p 0v, 1250p 0v, 1350p 0v, 2600p 2.5v, 3750p 2.5v, 3850p 2.5v, 5000p 2.5v, 5100p 0v, 6250p 0v, 6350p 0v, 7500p 0v, 7600p 2.5v, 8750p 2.5v, 8850p 2.5v)

vndinput ndinput 0 pwl(0p 2.5v, 100p 0v, 1250p 0v, 1350p 0v, 2600p 2.5v, 3750p 2.5v, 3850p 2.5v, 5000p 2.5v, 5100p 0v, 6250p 0v, 6350p 0v, 7500p 0v, 7600p 2.5v, 8750p 2.5v, 8850p 2.5v)

.tran 1ps 10ns
.option post

*Mosfet models from www.mosis.org
*
* DATE: Jun 11/01
* LOT: T14Y     WAF: 03
* DIE: N_Area_Fring    DEV: N3740/10
* Temp= 27

.MODEL CMOSN NMOS (LEVEL = 3
+ TOX = 5.7E-9  NSUB = 1E17  GAMMA = 0.4317311
+ PHI = 0.7  VTO = 0.4238252  DELTA = 0
+ U0 = 425.6466519  ETA = 0  THETA = 0.1754054

130
xor2

* SPICE xor2.sp

.SUBCKT inv A Y Vss Vdd
mn1 Vss A Y Vss cmosn l=0.25u w=0.50u
mp1 Vdd A Y Vdd cmosp l=0.25u w=1.00u
.ENDS

mn1 Vss A nA Vss cmosn l=0.25u w=0.50u
mp1 Vdd A nA Vdd cmosp l=0.25u w=1.00u

mn2 Vss B nB Vss cmosn l=0.25u w=0.50u
mp2 Vdd B nB Vdd cmosp l=0.25u w=1.00u

mn3 Vss B int0 Vss cmosn l=0.25u w=1.00u
mn4 Vss nB int1 Vss cmosn l=0.25u w=1.00u
mn5 int0 A Y Vss cmosn l=0.25u w=1.00u
mn6 int1 nA Y Vss cmosn l=0.25u w=1.00u

mp3 Vdd B int2 Vdd cmosp l=0.25u w=2.00u
mp4 Vdd nB int3 Vdd cmosp l=0.25u w=2.00u
mp5 int2 nA Y Vdd cmosp l=0.25u w=2.00u
mp6 int3 A Y Vdd cmosp l=0.25u w=2.00u

xinv1 Y cap1 Vss Vdd inv
xinv2 Y cap2 Vss Vdd inv
xinv3 Y cap3 Vss Vdd inv
xinv4 Y cap4 Vss Vdd inv
xinv5 nainput a Vss Vdd inv
xinv6 nbinput b Vss Vdd inv

vVdd Vdd 0 2.5
vVss Vss 0 0

* A 1 1 1 0 1 0 1
* B 1 0 1 0 1 1 1

vnainput nainput 0 pwl(0,2.5v,100p,0v,1250p,0v,1350p,0v,2500p,0v,2600p,0v,3750p,0v,3850p,2.5v,5000p,2.5v,5100p,0v,6250p,0v,6350p,2.5v,7500p,2.5v,7600p,0v)
vnbinput nbinput 0 pwl(0,2.5v,100p,0v,1250p,0v,1350p,2.5v,2500p,2.5v,2600p,0v,3750p,0v,3850p,2.5v,5000p,2.5v,5100p,0v,6250p,0v,6350p,0v,7500p,0v,7600p,0v)

.tran ips 8ns

.option post

* Mosfet models from www.mosis.org

* DATE: Jun 11/01
* LOT: T14Y WAF: 03
* DIR: N_Area_Fring DEV: N3740/10
* Temp= 27

.MODEL CMOSN NMOS ( LEVEL = 3
+ TOX = 5.7E-9 NSUB = 1E17 GAMMA = 0.4317311
+ PHI = 0.7 VTO = 0.4238252 DELTA = 0
+ UD = 425.6466519 ETA = 0 THETA = 0.1754054
+ AP = 2.501048E-4 VMAX = 8.287851E4 KAPPA = 0.1686779
+ RSH = 4.062439E-3 NFS = 1E12 TPC = 1
+ XJ = 3E-7 LD = 3.162278E-11 WD =
1.232881E-8
+ CGSO = 6.2E-10 CGSS0 = 6.2E-10 CGSO = 1E-10
+ CJ = 1.81211E-3 PB = 0.5 MJ = 0.3282553
+ CJSW = 5.341337E-10 MJSW = 0.5 )

132
.MODEL CMOSP PMOS (LEVEL = 3
+ TOX = 5.7E-9  NSUB = 1E17  GAMMA = 0.6348369
+ PHI = 0.7  VTU = -0.5536085  DELTA = 0
+ UD = 250  ETA = 0  THETA = 0.1573195
+ KP = 5.194153E-5  VMAX = 2.295325E5  KAPPA = 0.7448494
+ RSH = 30.0776952  WPS = 1E12  TPG = -1
+ XJ = 2E-7  LD = 9.968346E-13  WD = 5.475113E-9
+ CGDO = 6.66E-10  CGSO = 6.66E-10  CGBO = 1E-10
+ CJ = 1.893569E-3  PB = 0.9906013  MJ = 0.4664287
+ CJSW = 3.625544E-10  MJSW = 0.5 )
*
.end
Memory

DRAM 1-T

* SPICE dram.sp

.param length=0.25u
.param NO.5=0.50u
.param N1.0=1.00u
.param N1.5=1.50u
.param N2.0=2.00u
.param N4.0=4.00u
.param N8.0=8.00u
.param P1.0=1.00u
.param P2.0=2.00u
.param P3.0=3.00u
.param P4.0=4.00u
.param P16.0=16.00u

* INPUT SPICE dram.sp

.param length=0.25u
.param NO.5=0.50u
.param N1.0=1.00u
.param N1.5=1.50u
.param N2.0=2.00u
.param N4.0=4.00u
.param N8.0=8.00u
.param P1.0=1.00u
.param P2.0=2.00u
.param P3.0=3.00u
.param P4.0=4.00u
.param P16.0=16.00u

* wdata
* word
* vr
* OUTPUT
* bit
* nbit
*

******* DUMMY CELL *******

mn1 Vss Vss lbit Vss cmosn l=length w=NO.5

*actual dummy cell used for read
mn2 Vss prec bitdr Vss cmosn l=length w=NO.5
mn3 bitdr drsel rbit Vss cmosn l=length w=NO.5
*cap
mn5 Vss bitdr Vss Vss cmosn l=length w=NO.5

******* CELLS *******
*actual cell for test
mn7 bit0 bit0sel lbit Vss cmosn l=length w=NO.5
*cap
mn8 Vss bit0 Vss Vss cmosn l=length w=NO.5
mn9 Vss bit0 Vss Vss cmosn l=length w=NO.5
mn10 Vss bit0 Vss Vss cmosn l=length w=NO.5
mn11 Vss bit0 Vss Vss cmosn l=length w=NO.5

**** Other bits on left
mn12 Vss Vss lbit Vss cmosn l=length w=NO.5
mn13 Vss Vss lbit Vss cmosn l=length w=NO.5
mn14 Vss Vss lbit Vss cmosn l=length w=NO.5
mn15 Vss Vss lbit Vss cmosn l=length w=NO.5

mn16 Vss Vss lbit Vss cmosn l=length w=NO.5
mn17 Vss Vss lbit Vss cmosn l=length w=NO.5
mn18 Vss Vss lbit Vss cmosn l=length w=NO.5
mn19 Vss Vss lbit Vss cmosn l=length w=NO.5

mn20 Vss Vss lbit Vss cmosn l=length w=NO.5
mn21 Vss Vss lbit Vss cmosn l=length w=NO.5
mn22 Vss Vss lbit Vss cmosn l=length w=NO.5
mn23 Vss Vss lbit Vss cmosn l=length w=NO.5

mn24 Vss Vss lbit Vss cmosn l=length w=NO.5
mn25 Vss Vss lbit Vss cmosn l=length w=NO.5
mn26 Vss Vss lbit Vss cmosn l=length w=NO.5

**** Other bits on right
mn27 Vss Vss rbit Vss cmosn l=length w=NO.5
mn28 Vss Vss rbit Vss cmosn l=length w=NO.5
mn29 Vss Vss rbit Vss cmosn l=length w=NO.5
mn30 Vss Vss rbit Vss cmosn l=length w=NO.5

mn31 Vss Vss rbit Vss cmosn l=length w=NO.5
mn32 Vss Vss rbit Vss cmosn l=length w=NO.5
mn33 Vss Vss rbit Vss cmosn l=length w=NO.5
mn34 Vss Vss rbit Vss cmosn l=length w=NO.5

mn35 Vss Vss rbit Vss cmosn l=length w=NO.5
mn36 Vss Vss rbit Vss cmosn l=length w=NO.5
mn37 Vss Vss rbit Vss cmosn l=length w=NO.5
mn38 Vss Vss rbit Vss cmosn l=length w=NO.5

mn39 Vss Vss rbit Vss cmosn l=length w=NO.5
mn40 Vss Vss rbit Vss cmosn l=length w=NO.5
mn41 Vss Vss rbit Vss cmosn l=length w=NO.5
mn42 Vss Vss rbit Vss cmosn l=length w=NO.5

****** WRITE INTERFACE ******

mn43 wdata write lbit Vss cmosn l=length w=N8.0
mn44 Vss Vss rbit Vss cmosn l=length w=N8.0

****** WDATA INVERTERS ******

mp1 Vdd wdatain intwdata Vdd cmosp l=length w=P16.0
mn45 Vss wdatain intwdata Vss cmosn l=length w=N8.0

mp2 Vdd intwdata wdata Vdd cmosp l=length w=P16.0
mn46 Vss intwdata wdata Vss cmosn l=length w=N8.0

****** SENSE AMP ******

mn47 rbit prec Vdd Vss cmosn l=length w=N1.0
mn48 lbit prec Vdd Vss cmosn l=length w=N1.0
mn49 lbit prec rbit Vss cmosn l=length w=N1.0
mn50 int0 lbit rbit Vss cmosn l=length w=N4.0
mn51 int0 rbit lbit Vss cmosn l=length w=N4.0

mn52 Vss cs int0 Vss cmosn l=1.0u w=N1.0

****** Restoring ******

mn53 lbit rbit int1 Vdd cmosp l=length w=P2.0
mn54 rbit lbit int1 Vdd cmosp l=length w=P2.0
mn55 int1 ncs Vdd Vdd cmosp l=1.0u w=P2.0

vVdd Vdd 0 2.5
.tran 1ps 30ns
.option post

* Mosfet models from www.mosis.org
* DATE: Jun 11/01
* LOT: T4Y                     WAF: 03
* DIE: N_Area_Fring            DEV: N3740/10
* Temp = 27

.MODEL CMOSN NMOS (LEVEL = 3)
tox = 5.7E-9          ns = 1E17          gamma = 0.4317311
phi = 0.7          vto = 0.4238252          delta = 0
ud = 425.6466519        eta = 0          theta = 0.1754054
kp = 2.501048E-4      vmax = 8.287851E4      kappa = 0.1686779
rsh = 4.062439E-3     nfs = 1E12          tpg = 1
xj = 3E-7            ld = 3.162278E-11      wd =
1.232881E-8
ccdo = 6.2E-10      cgsd = 6.2E-10      cbdo = 1E-10
cj = 1.81211E-3      pb = 0.5          mj = 0.3282653
cjsw = 5.341337E-10  mjsw = 0.5

.MODEL CMOSP PMOS (LEVEL = 3)
tox = 5.7E-9          ns = 1E17          gamma = 0.6348369
phi = 0.7          vto = -0.5536085         delta = 0
ud = 250           eta = 0          theta = 0.1673196
kp = 5.194153E-5      vmax = 2.295325E5      kappa = 0.7448494
rsh = 30.0776962     nfs = 1E12          tpg = -1
xj = 2E-7            ld = 9.368346E-13      wd =
5.475113E-9
ccdo = 6.66E-10      cgsd = 6.66E-10      cbdo = 1E-10
 cj = 1.893569E-3      pb = 0.9906013      mj = 0.4664287
 cjsw = 3.625544E-10  mjsw = 0.5

*alter * n = 1, 2, 4    p = 4, 8
*.param n0.5=1.00u
*.param n1.0=1.00u
*.param n1.5=2.00u
*.param N2.0=2.00u
*.param N4.0=4.00u
*.param N8.0=7.00u
*.param P1.0=4.00u
*.param P2.0=4.00u
*.param P3.0=4.00u
*.param P4.0=4.00u
*.param P16.0=16.00u
*.alter N = 1, 2, 4
*.param N0.5=1.00u
*.param N1.0=1.00u
*.param N1.5=2.00u
*.param N2.0=2.00u
*.param N4.0=4.00u
*.param N8.0=7.00u
*.param P1.0=2.00u
*.param P2.0=2.00u
*.param P3.0=2.00u
*.param P4.0=4.00u
*.param P16.0=16.00u
*.alter N = 2, 8
*.param N0.5=2.00u
*.param N1.0=2.00u
*.param N1.5=2.00u
*.param N2.0=2.00u
*.param N4.0=2.00u
*.param N8.0=8.00u
*.param P1.0=2.00u
*.param P2.0=2.00u
*.param P3.0=2.00u
*.param P4.0=4.00u
*.param P16.0=16.00u
*.end
DRAM 3-T nMos Version

* SPICE dram.sp

.param length=0.25u
.param N0.5=0.50u
.param N1.0=1.00u
.param N1.5=1.50u
.param N2.0=2.00u
.param N4.0=4.00u
.param N8.0=8.00u
.param P1.0=1.00u
.param P2.0=2.00u
.param P3.0=3.00u
.param P4.0=4.00u
.param P16.0=16.00u

* * INPUT
* wdata
* word
* w
* * OUTPUT
* bit
* nbit
*
******* 3 T DRAM CELL *******
* Write access
mn1 wdata write mem Vss cmosn l=length w=N1.0
 *
* Storage
mn2 Vss mem rint Vss cmosn l=length w=NO.5
 *
* Read access
mn3 rint read rdata Vss cmosn l=length w=N1.0

******* WDATA INVERTERS *******
mp4 Vdd wdatain intwdata Vdd cmosp l=length w=P16.0
mn9 Vss wdatain intwdata Vss cmosn l=length w=N8.0

mp5 Vdd intwdata wdata Vdd cmosp l=length w=P16.0
mn10 Vss intwdata wdata Vss cmosn l=length w=N8.0

******* PULLUPS *******
mn5 wdata Vdd Vdd Vss cmosn l=length w=N2.0

******* COLUMN DECODER *******

mn11 rdata Vdd cd0 Vss cmosn l=length w=N1.0
mn12 Vss Vss cd0 Vss cmosn l=length w=N1.0
mn13 cd0 Vdd cd1 Vss cmosn l=length w=N1.0
mn14 Vss Vss cd1 Vss cmosn l=length w=N1.0
mn15 cd1 Vdd cd2 Vss cmosn l=length w=N1.0
mn16 Vss Vss cd2 Vss cmosn l=length w=N1.0
mn17 cd2 Vdd cd3 Vss cmosn l=length w=N1.0
mn18 Vss Vss cd3 Vss cmosn l=length w=N1.0
mn19 cd3 Vdd int1 Vss cmosn l=length w=N1.0
mn20 Vss Vss int1 Vss cmosn l=length w=N1.0

******* SENSE AMP *******

* Vref
mp1 Vdd ref ref Vdd cmosp l=length w=P4.0
mn6 Vss Vdd ref Vss cmosn l=length w=NO.5
mp2 Vdd ref int0 Vdd cmosp l=length w=P1.0
mp7 int1 ref int0 Vss cmosn l=length w=N4.0

mp3 Vdd int0 rout Vdd cmosp l=length w=P3.0
mn8 Vss int1 rout Vss cmosn l=length w=N0.5

vVdd Vdd 0 2.5
vVss Vss 0 0

vWr write 0 pwl(0000p 0.0v, 1150p 0.0v, 1250p 2.5v, 7400p 2.5v, 7500p 0.0v, 17400p 0.0v, 17500p 2.5v, 22400p 2.5v, 22500p 0.0v)

vWdata wdatain 0 pwl(0000p 0.0v, 9900p 0.0v, 10000p 2.5v)

*vselect this bit
vWord read 0 pwl(0000p 0.0v, 12400p 0.0v, 12500p 2.5v, 16150p 2.5v, 16250p 0.0v, 26000p 0.0v, 26100p 2.5v, 29000p 2.5v, 29100p 0.0v)

.tran 1ps 30ns
.option post

*Mosfet models from www.mosis.org
*
* DATE: Jun 11/01
* LOT: T14Y WAF: 03
* DIE: N_Area_Fring DEV: N3740/10
* Temp= 27

.Model CMOSN NMOS ( LEVEL = 3 )
+ TOX = 5.7E-9 NSUB = 1E17 GAMMA = 0.4317311
+ PHI = 0.7 VTO = 0.4238252 DELTA = 0
+ UD = 425.6466519 ETA = 0 THETA = 0.1754054
+ KP = 2.501048E-4 VMAX = 8.287651E4 KAPPA = 0.1686779
+ RSH = 4.062439E-3 NFS = 1E12 TPG = 1
+ XJ = 3E-7 LD = 3.162278E-11 WD = 1.232881E-8
+ CGDO = 6.2E-10 CGSO = 6.2E-10 CGBO = 1E-10
+ CJ = 1.81211E-3 PB = 0.5 MJ = 0.3282553
+ CJSW = 5.341337E-10 MJSW = 0.5
)

.Model CMOSP PMOS ( LEVEL = 3 )
+ TOX = 5.7E-9 NSUB = 1E17 GAMMA = 0.6348369
+ PHI = 0.7 VTO = -0.5536085 DELTA = 0
+ UD = 250 ETA = 0 THETA = 0.1573195
+ KP = 5.194153E-6 VMAX = 2.295325E5 KAPPA = 0.7446494
+ RSH = 30.0769962 NFS = 1E12 TPG = -1
+ XJ = 2E-7 LD = 9.968346E-13 WD = 5.475113E-9
+ CGDO = 6.66E-10 CGSO = 6.66E-10 CGBO = 1E-10
+ CJ = 1.893569E-3 PB = 0.9906013 MJ = 0.4664287
+ CJSW = 3.625544E-10 MJSW = 0.5
)
.alter * N = 1, 2, 4 P = 4, 8
.param NO.5=1.00u
.param N1.0=1.00u
.param N1.5=2.00u
.param N2.0=2.00u
.param N4.0=4.00u
.param NO.6=7.00u
.param P1.0=4.00u
.param P2.0=4.00u

139
.param P3.0=4.00u
.param P4.0=4.00u
.param P16.0=16.00u

.alter * N = 1, 2, 4   P = 2, 4, 8
.param N0.5=1.00u
.param N1.0=1.00u
.param N1.5=2.00u
.param N2.0=2.00u
.param N4.0=4.00u
.param N8.0=7.00u
.param P1.0=2.00u
.param P2.0=2.00u
.param P3.0=2.00u
.param P4.0=4.00u
.param P16.0=12.00u

.alter * N = 2, 8   P = 2, 4, 10
.param N0.5=2.00u
.param N1.0=2.00u
.param N1.5=2.00u
.param N2.0=2.00u
.param N4.0=2.00u
.param N8.0=8.00u
.param P1.0=2.00u
.param P2.0=2.00u
.param P3.0=2.00u
.param P4.0=4.00u
.param P16.0=16.00u
.end
DRAM 3-T pMOS Version

* SPICE dram.sp

.param length=0.25u
.param N0.5=0.50u
.param N1.0=1.00u
.param N1.5=1.50u
.param N2.0=2.00u
.param N4.0=4.00u
.param N8.0=8.00u
.param P1.0=1.00u
.param P2.0=2.00u
.param P3.0=3.00u
.param P4.0=4.00u
.param P16.0=16.00u

* INPUT
* wdata
* word
* wr
* OUTPUT
* bit
* nbit
* 
****** 3 T DRAM CELL ******
* Write access
mn1 wdata write mem Vdd cmosp l=length w=P1.0

* Storage
mn2 Vdd mem rint Vdd cmosp l=length w=N3u

* Read access
mn3 rint read rdata Vdd cmosp l=length w=P2.0

****** WDATA INVERTERS ******
mp4 Vdd wdatain intwdata Vdd cmosp l=length w=P16.0
mn9 Vdd wdatain intwdata Vss cmosn l=length w=N8.0

mp5 Vdd intwdata wdata Vdd cmosp l=length w=P16.0
mn10 Vdd intwdata wdata Vss cmosn l=length w=N8.0

****** PREDISCHARGE ******
mn5 wdata pred Vss Vss cmosn l=length w=10u

****** COLUMN DECODER ******
*mn11 rdata Vdd cd0 Vss cmosn l=length w=N1.0
mn12 Vss Vss cd0 Vss cmosn l=length w=N1.0
mn13 cd0 Vdd cd1 Vss cmosn l=length w=N1.0
mn14 Vss Vss cd1 Vss cmosn l=length w=N1.0
mn15 cd1 Vdd cd2 Vss cmosn l=length w=N1.0
mn16 Vss Vss cd2 Vss cmosn l=length w=N1.0
mn17 cd2 Vdd cd3 Vss cmosn l=length w=N1.0
mn18 Vss Vss cd3 Vss cmosn l=length w=N1.0
mn19 cd3 Vdd int1 Vss cmosn l=length w=N1.0
mn20 Vss Vss int1 Vss cmosn l=length w=N1.0

****** SENSE AMP ******
* Vref
mp1 Vdd ref ref Vdd cmosp l=length w=P4.0
mn6 Vss Vdd ref ref Vss cmosn l=length w=N0.5
mp2 Vdd ref int0 Vdd cmosp l=length w=P1.0
mn7 int1 ref int0 Vss cmosn l=length w=N4.0

mp3 Vdd int0 rout Vdd cmosp l=length w=P3.0
mn8 int1 rout Vss cmosn l=length w=N0.5

vVdd 0 2.5
vVss 0 0

vWr write 0 pwl (0000p 2.5v, 1150p 2.5v, 1250p 0.0v, 7400p 0.0v, 7500p 2.5v, 17400p 2.5v, 17500p 0.0v, 22400p 0.0v, 22500p 2.5v)

vWdata wdatain 0 pwl (0000p 0.0v, 9900p 0.0v, 10000p 2.5v)

*select this bit
vWord read 0 pwl (0000p 2.5v, 12400p 2.5v, 12500p 0.0v, 16150p 0.0v, 16250p 2.5v, 26000p 2.5v, 26100p 0.0v, 29000p 0.0v, 29100p 2.5v)

*.IC V(mem)=2.5
.tran 1ps 30ns
.option post

*Mosfet models from www.mosis.org
* DATE: Jun 11/01
* LOT: T14Y
* DIE: N.Area_Fring
* Temp= 27
.
LEVEL = 3
.
.MODEL CMOSN NMOS (LEVEL = 3)
+ TOX = 5.7E-9
+ PHI = 0.7
+ UD = 425.6466519
+ KP = 2.501048E-4
+ RSH = 4.062439E-3
+ XJ = 3E-7
+ CGDO = 6.2E-10
+ RSH = 4.062439E-3
+ XJ = 3E-7
+ CGDO = 6.66E-10
+ RSH = 3.625544E-10
.
.
.MODEL CMOSP PMOS (LEVEL = 3)
+ TOX = 5.7E-9
+ PHI = 0.7
+ UD = 260
+ KP = 5.194153E-5
+ RSH = 30.077851E4
+ XJ = 3E-7
+ CGDO = 6.2E-10
+ RSH = 4.062439E-3
+ XJ = 3E-7
+ CGDO = 6.66E-10

.end
SRAM 6-T

* SPICE sram.sp

.param length=0.25u
.param NO.5=0.50u
.param N1.0=1.00u
.param N1.5=1.50u
.param NO.0=8.00u
.param P1.0=1.00u
.param P2.0=2.00u
.param P3.0=3.00u
.param P16.0=16.00u

* INPUT
* wdata
* word
* wr
*
* OUTPUT
* bit
* nbit

******* 6 T SRAM CELL *******

* Forward inverter
mp1 Vdd mem0 nmem1 Vdd cmosP l=length w=P1.0
mn1 Vss mem0 nmem1 Vss cmosN l=length w=NO.5

* Backward inverter
mp2 Vdd mem1 nmem0 Vdd cmosP l=length w=P1.0
mn2 Vss mem1 nmem0 Vss cmosN l=length w=NO.5

* Access cmosN
mn3 mem0 word nbit Vss cmosN l=length w=N1.5
mn4 mem1 word bit Vss cmosN l=length w=N1.5

******* RESISTORS *******

mn5 nbit Vdd Vdd Vss cmosN l=length w=N1.0
mn6 bit Vdd Vdd Vss cmosN l=length w=N1.0

******* WRITE ACCESS *******

mn7 nwdtata wr nbit Vss cmosN l=length w=N8.0
mn8 intwdata wr bit Vss cmosN l=length w=N8.0

******* WDATA INVERTERS *******

mp3 Vdd wdata nwdtata Vdd cmosP l=length w=P16.0
mn9 Vss wdata nwdtata Vss cmosN l=length w=N8.0

mp4 Vdd nwdtata intwdata Vdd cmosP l=length w=P16.0
mn10 Vss nwdtata intwdata Vss cmosN l=length w=N8.0

******* SENSE AMP *******

mp5 Vdd mirror nrdtata Vdd cmosP l=length w=P2.0
mn11 intO nbit nrdtata Vss cmosN l=length w=N1.0

mp6 Vdd mirror mirror Vdd cmosP l=length w=P2.0
mn12 intO bit mirror Vss cmosN l=length w=N1.0

mn13 Vss Vdd intO Vss cmosN l=2.0u w=0.5u

mp7 Vdd nrdtata rdata Vdd cmosP l=length w=P2.0
mn14 Vss nrdtata rdata Vss cmosN l=length w=N1.0

*mn100 Vss word Vss Vss cmosN l=length w=N1.5
*mn101 Vss word Vss Vss cmosN l=length w=N1.5
mn102 \text{Vss word Vss cmosn l=1-length w=N1.5}
mn103 \text{Vss word Vss Vss cmosn l=1-length w=N1.5}
mn104 \text{Vss word Vss Vss cmosn l=1-length w=N1.5}
mn105 \text{Vss word Vss Vss cmosn l=1-length w=N1.5}
mn106 \text{Vss word Vss Vss cmosn l=1-length w=N1.5}
mn107 \text{Vss word Vss cmosn l=1-length w=N1.5}
mn108 \text{Vss word Vss Vss cmosn l=1-length w=N1.5}
mn109 \text{Vss word Vss Vss cmosn l=1-length w=N1.5}
mn110 \text{Vss word Vss Vss cmosn l=1-length w=N1.5}
mn111 \text{Vss word Vss Vss cmosn l=1-length w=N1.5}
mn112 \text{Vss word Vss Vss cmosn l=1-length w=N1.5}
mn113 \text{Vss word Vss Vss cmosn l=1-length w=N1.5}
mn114 \text{Vss word Vss Vss cmosn l=1-length w=N1.5}
mn115 \text{Vss word Vss Vss cmosn l=1-length w=N1.5}
mn116 \text{Vss word Vss Vss cmosn l=1-length w=N1.5}
mn117 \text{Vss word Vss Vss cmosn l=1-length w=N1.5}
mn118 \text{Vss word Vss Vss cmosn l=1-length w=N1.5}
mn119 \text{Vss word Vss Vss cmosn l=1-length w=N1.5}
mn120 \text{Vss word Vss Vss cmosn l=1-length w=N1.5}
mn121 \text{Vss word Vss Vss cmosn l=1-length w=N1.5}
mn122 \text{Vss word Vss Vss cmosn l=1-length w=N1.5}
mn123 \text{Vss word Vss Vss cmosn l=1-length w=N1.5}
mn124 \text{Vss word Vss Vss cmosn l=1-length w=N1.5}
mn125 \text{Vss word Vss Vss cmosn l=1-length w=N1.5}
mn126 \text{Vss word Vss Vss cmosn l=1-length w=N1.5}
mn127 \text{Vss word Vss Vss cmosn l=1-length w=N1.5}
mn128 \text{Vss word Vss Vss cmosn l=1-length w=N1.5}
mn129 \text{Vss word Vss Vss cmosn l=1-length w=N1.5}
mn130 \text{Vss word Vss Vss cmosn l=1-length w=N1.5}
mn131 \text{Vss word Vss Vss cmosn l=1-length w=N1.5}
mn200 \text{Vss Vss bit cmosn l=1-length w=N1.5}
mn300 \text{Vss Vss bit cmosn l=1-length w=N1.5}
mn201 \text{Vss Vss bit cmosn l=1-length w=N1.5}
mn301 \text{Vss Vss bit cmosn l=1-length w=N1.5}
mn202 \text{Vss Vss bit cmosn l=1-length w=N1.5}
mn302 \text{Vss Vss bit cmosn l=1-length w=N1.5}
mn203 \text{Vss Vss bit cmosn l=1-length w=N1.5}
mn303 \text{Vss Vss bit cmosn l=1-length w=N1.5}
mn204 \text{Vss Vss bit cmosn l=1-length w=N1.5}
mn304 \text{Vss Vss bit cmosn l=1-length w=N1.5}
mn205 \text{Vss Vss bit cmosn l=1-length w=N1.5}
mn305 \text{Vss Vss bit cmosn l=1-length w=N1.5}
mn206 \text{Vss Vss bit cmosn l=1-length w=N1.5}
mn306 \text{Vss Vss bit cmosn l=1-length w=N1.5}
mn207 \text{Vss Vss bit cmosn l=1-length w=N1.5}
mn307 \text{Vss Vss bit cmosn l=1-length w=N1.5}
mn208 \text{Vss Vss bit cmosn l=1-length w=N1.5}
mn308 \text{Vss Vss bit cmosn l=1-length w=N1.5}
mn209 \text{Vss Vss bit cmosn l=1-length w=N1.5}
mn309 \text{Vss Vss bit cmosn l=1-length w=N1.5}
mn210 \text{Vss Vss bit cmosn l=1-length w=N1.5}
mn310 \text{Vss Vss bit cmosn l=1-length w=N1.5}
mn211 \text{Vss Vss bit cmosn l=1-length w=N1.5}
mn311 \text{Vss Vss bit cmosn l=1-length w=N1.5}
mn212 \text{Vss Vss bit cmosn l=1-length w=N1.5}
mn312 \text{Vss Vss bit cmosn l=1-length w=N1.5}
mn213 \text{Vss Vss bit cmosn l=1-length w=N1.5}
mn313 \text{Vss Vss bit cmosn l=1-length w=N1.5}
mn214 \text{Vss Vss bit cmosn l=1-length w=N1.5}
mn314 \text{Vss Vss bit cmosn l=1-length w=N1.5}
mn215 \text{Vss Vss bit cmosn l=1-length w=N1.5}
mn315 \text{Vss Vss bit cmosn l=1-length w=N1.5}
mn216 \text{Vss Vss bit cmosn l=1-length w=N1.5}
mn316 \text{Vss Vss bit cmosn l=1-length w=N1.5}
mn217 \text{Vss Vss bit cmosn l=1-length w=N1.5}
mn317 \text{Vss Vss bit cmosn l=1-length w=N1.5}
mn218 \text{Vss Vss bit cmosn l=1-length w=N1.5}
mn318 Vss Vss nbit cmosn 1=length w=N1.5
mn219 Vss Vss bit cmosn 1=length w=N1.5
mn319 Vss Vss nbit cmosn 1=length w=N1.5
mn220 Vss Vss bit cmosn 1=length w=N1.5
mn320 Vss Vss nbit cmosn 1=length w=N1.5
mn221 Vss Vss bit cmosn 1=length w=N1.5
mn321 Vss Vss nbit cmosn 1=length w=N1.5
mn222 Vss Vss bit cmosn 1=length w=N1.5
mn322 Vss Vss nbit cmosn 1=length w=N1.5
mn223 Vss Vss bit cmosn 1=length w=N1.5
mn323 Vss Vss nbit cmosn 1=length w=N1.5
mn224 Vss Vss bit cmosn 1=length w=N1.5
mn324 Vss Vss nbit cmosn 1=length w=N1.5
mn225 Vss Vss bit cmosn 1=length w=N1.5
mn325 Vss Vss nbit cmosn 1=length w=N1.5
mn226 Vss Vss bit cmosn 1=length w=N1.5
mn326 Vss Vss nbit cmosn 1=length w=N1.5
mn227 Vss Vss bit cmosn 1=length w=N1.5
mn327 Vss Vss nbit cmosn 1=length w=N1.5
mn228 Vss Vss bit cmosn 1=length w=N1.5
mn328 Vss Vss nbit cmosn 1=length w=N1.5
mn229 Vss Vss bit cmosn 1=length w=N1.5
mn329 Vss Vss nbit cmosn 1=length w=N1.5
mn230 Vss Vss bit cmosn 1=length w=N1.5
mn330 Vss Vss nbit cmosn 1=length w=N1.5
mn231 Vss Vss bit cmosn 1=length w=N1.5
mn331 Vss Vss nbit cmosn 1=length w=N1.5
mn232 Vss Vss bit cmosn 1=length w=N1.5
mn332 Vss Vss nbit cmosn 1=length w=N1.5

vVdd Vdd 0 2.5
vVss Vss 0 0

vWr wr 0 pwl(0000p 0.0v, 1150p 0.0v, 1250p 2.5v, 7400p 2.5v, 7500p 0.0v, 17400p 0.0v, 17500p 2.5v, 22400p 2.5v, 22500p 0.0v)

vWdata wdata 0 pwl(0000p 0.0v, 9900p 0.0v, 10000p 2.5v)

*select this bit
vWord word 0 pwl(0000p 0.0v, 1150p 0.0v, 1250p 2.5v, 7400p 2.5v, 7500p 0.0v, 12400p 0.0v, 12500p 2.5v, 16150p 2.5v, 16250p 0.0v, 17400p 0.0v, 17500p 2.5v, 22400p 2.5v, 22500p 0.0v, 26100p 2.5v, 29000p 2.5v, 29100p 0.0v)

.tran 1ps 30ns
.option post

* Mosfet models from www.mosis.org

* DATE: Jun 11/01
* LOT: T14Y
* DIE: N_Area_Fring
* Temp= 27

.MODEL CMOSN NMOS (LEVEL = 3
+ TOX = 5.7E-9 NSUB = 1E17 GAMMA = 0.4317311
+ PHI = 0.7 VTO = 0.4238262 DELTA = 0
+ UD = 426.6466519 ETA = 0 THETA = 0.1754054
+ KP = 2.601046E-4 VMAX = 8.287851E4 KAPPA = 0.1606779
+ RSH = 4.062439E-3 NFS = 1E12 TPG = 1
+ XJ = 3E-7 LD = 3.162278E-11 WD =
1.232881E-8
+ CGDO = 6.2E-10 CGSU = 6.2E-10 CGBO = 1E-10
+ CJ = 1.81211E-3 PB = 0.8 MJ = 0.3282553
+ CJSW = 5.341337E-10 MJSW = 0.5 )
.MODEL CMOSP PMOS (LEVEL = 3
+ TOX = 5.7E-9 NSUB = 1E17 GAMMA = 0.5348369
+ PHI = 0.7 VT0 = -0.5536085 DELTA = 0
+ U0 = 250 ETA = 0 THETA = 0.1573185
+ KP = 5.194153E-5 VMAX = 2.295326E5 KAPPA = 0.7448494
+ BSH = 30.0776952 NFS = 1E12 TPG = -1
+ XJ = 2E-7 LD = 9.968346E-13 WD =
5.475113E-9
+ CGDO = 6.66E-10 CGSO = 6.66E-10 CGBO = 1E-10
+ CJ = 1.893569E-3 PB = 0.9906013 MJ =
0.4664287
+ CJSW = 3.625844E-10 MJSW = 0.5 )
*
.alter * N = 1, 2, 4 P = 4, 8
.param NO.5=1.00u
.param N1.0=1.00u
.param N1.5=2.00u
.param N8.0=7.00u
.param P1.0=4.00u
.param P2.0=4.00u
.param P3.0=4.00u
.param P16.0=16.00u
.alter * N = 1, 2, 4 P = 2, 4, 8
.param NO.5=1.00u
.param N1.0=1.00u
.param N1.5=2.00u
.param N8.0=7.00u
.param P1.0=2.00u
.param P2.0=2.00u
.param P3.0=2.00u
.param P16.0=12.00u
.alter * N = 2, 8 P = 2, 4, 10
.param NO.5=2.00u
.param N1.0=2.00u
.param N1.5=2.00u
.param N8.0=8.00u
.param P1.0=2.00u
.param P2.0=2.00u
.param P3.0=2.00u
.param P16.0=16.00u
.alter * N = 8 P = 16
.param NO.5=8.00u
.param N1.0=8.00u
.param N1.5=8.00u
.param N8.0=8.00u
.param P1.0=16.00u
.param P2.0=16.00u
.param P3.0=16.00u
.param P16.0=16.00u
.end
Domino Logic

16-bit Adder

* SPICE fulladder.sp

.param length=0.25u
.param NO.5=0.50u
.param N1.0=1.00u
.param N1.5=1.50u
.param N8.0=8.00u
.param P1.0=1.00u
.param P2.0=2.00u
.param P3.0=3.00u
.param P16.0=16.00u

.SUBCKT inv A Y Vss Vdd
mni Vss A Y Vss cmosn l=length w=NO.5
mp1 Vdd A Y Vdd cmosp l=length w=P1.0
.ENDS

.SUBCKT xor a b na nb out nout clk Vss Vdd
+ XOR
mp1 Vdd clk xnorO Vdd cmosp l=length w=P1.0

mn1 Vss clk evalO Vss cmosn l=length w=N1.0
mn2 evalO a intO Vss cmosn l=length w=N1.0
mn3 evalO na int1 Vss cmosn l=length w=N1.0
mn4 intO nb xnorO Vss cmosn l=length w=N1.0
mn5 int1 b xnorO Vss cmosn l=length w=N1.0

mp2 Vdd xnorO out Vdd cmosp l=length w=P3.0
mn6 Vss xnorO out Vss cmosn l=length w=NO.5

+ XNOR
mp3 Vdd clk xorO Vdd cmosp l=length w=P1.0

mn7 Vss clk eval1 Vss cmosn l=length w=N1.0
mn8 eval1 a int2 Vss cmosn l=length w=N1.0
mn9 eval1 na int3 Vss cmosn l=length w=N1.0
mn10 int2 b xorO Vss cmosn l=length w=N1.0
mn11 int3 nb xorO Vss cmosn l=length w=N1.0

mp4 Vdd xorO out Vdd cmosp l=length w=P3.0
mn12 Vss xorO out Vss cmosn l=length w=NO.5
.ENDS

.SUBCKT and a b out clk Vss Vdd
mp1 Vdd clk nand Vdd cmosp l=length w=P1.0

mn1 Vss clk evalO Vss cmosn l=length w=N1.0
mn2 evalO a intO Vss cmosn l=length w=N1.0
mn3 intO b nand Vss cmosn l=length w=N1.0

mp2 Vdd nand out Vdd cmosp l=length w=P3.0
mn4 Vss nand out Vss cmosn l=length w=NO.5
.ENDS

.SUBCKT or a b out clk Vss Vdd
mp1 Vdd clk nor Vdd cmosp l=length w=P1.0

mn1 Vss clk evalO Vss cmosn l=length w=N1.0
mn2 evalO a nor Vss cmosn l=length w=N1.0
mn3 evalO b nor Vss cmosn l=length w=N1.0

mp2 Vdd nor out Vdd cmosp l=length w=P3.0
mn4 Vss nor out Vss cmosn l=length w=NO.5
.SUBCKT pgblock a b na nb np np g ng clk Vss Vdd
xxor1 a b na nb np np g ng clk Vss Vdd xor
xand1 a b g clk Vss Vdd and
xand2 na nb ng clk Vss Vdd or
.ENDS

xc0 c0 p0 p1 p2 p3 g0 g1 g2 g3 pg0 pg1 pg2 pg3 c1 c2 c3 clk Vss Vdd carry
xc1 c4 p4 p5 p6 p7 g4 g5 g6 g7 pg4 pg5 pg6 pg7 c5 c6 c7 clk Vss Vdd carry
xc2 c8 p8 p9 p10 p11 g8 g9 g10 g11 pg8 pg9 pg10 pg11 c9 c10 c11 clk Vss Vdd carry
xc3 c12 p12 p13 p14 p15 g12 g13 g14 g15 pg12 pg13 pg14 pg15 c13 c14 c15 clk Vss Vdd carry

xcn0 nc0 p1 p2 np0 np1 np2 np3 ng0 ng1 ng2 ng3 npg0 npg1 npg2 npg3 c1 c2 c3 clk Vss Vdd ncarry
xcn1 nc4 p4 p5 p6 p7 ng4 ng5 ng6 ng7 npg4 npg5 npg6 npg7 c5 c6 c7 clk Vss Vdd ncarry
xcn2 nc8 p8 p9 p10 p11 ng8 ng9 ng10 ng11 npg8 npg9 npg10 npg11 c9 c10 c11 clk Vss Vdd ncarry
xcn3 nc12 p12 p13 p14 p15 np12 np13 np14 np15 ng12 ng13 ng14 ng15 npg12 npg13 npg14 npg15 c13 c14 c15 clk Vss Vdd ncarry

xcc0 c0 pg0 pg1 pg2 pg3 gg0 gg1 gg2 gg3 pout15 gout15 c4 c8 c12 clk Vss Vdd carry

xcc0 nc0 pg1 pg2 npg0 npg1 npg2 npg3 ngg0 ngg1 ngg2 ngg3 pout15 nc4 nc8 nc12 clk Vss Vdd ncarry

xpgb0 a0 b0 na0 nb0 p0 np0 g0 ng0 clk Vss Vdd pgblock
xpgb1 a1 b1 na1 nb1 p1 np1 g1 ng1 clk Vss Vdd pgblock
xpgb2 a2 b2 na2 nb2 p2 np2 g2 ng2 clk Vss Vdd pgblock
xpgb3 a3 b3 na3 nb3 p3 np3 g3 ng3 clk Vss Vdd pgblock
xpgb4 a4 b4 na4 nb4 p4 np4 g4 ng4 clk Vss Vdd pgblock
xpgb5 a5 b5 na5 nb5 p5 np5 g5 ng5 clk Vss Vdd pgblock
xpgb6 a6 b6 na6 nb6 p6 np6 g6 ng6 clk Vss Vdd pgblock
xpgb7 a7 b7 na7 nb7 p7 np7 g7 ng7 clk Vss Vdd pgblock
xpgb8 a8 b8 na8 nb8 p8 np8 g8 ng8 clk Vss Vdd pgblock
xpgb9 a9 b9 na9 nb9 p9 np9 g9 ng9 clk Vss Vdd pgblock
xpgb10 a10 b10 na10 nb10 p10 np10 g10 ng10 clk Vss Vdd pgblock
xpgb11 a11 b11 na11 nb11 p11 np11 g11 ng11 clk Vss Vdd pgblock
xpgb12 a12 b12 na12 nb12 p12 np12 g12 ng12 clk Vss Vdd pgblock
xpgb13 a13 b13 na13 nb13 p13 np13 g13 ng13 clk Vss Vdd pgblock
<table>
<thead>
<tr>
<th>Input</th>
<th>Voltage</th>
</tr>
</thead>
<tbody>
<tr>
<td>a1</td>
<td>0 V, 100000 pV</td>
</tr>
<tr>
<td>b1</td>
<td>0 pV, 0 V</td>
</tr>
<tr>
<td>a2</td>
<td>100 pV, 2.5 V</td>
</tr>
<tr>
<td>b2</td>
<td>25000 pV, 2.5 V, 25100 pV, 2.5 V, 37500 pV, 2.5 V, 37600 pV, 2.5 V, 50000 pV, 2.5 V, 50100 pV</td>
</tr>
<tr>
<td>b3</td>
<td>2.5 V, 50100 pV, 2.5 V, 62600 pV, 2.5 V, 75000 pV, 2.5 V, 75100 pV, 2.5 V, 87500 pV</td>
</tr>
<tr>
<td>a4</td>
<td>2.5 V, 87600 pV, 2.5 V, 100000 pV, 2.5 V, 101000 pV, 2.5 V</td>
</tr>
<tr>
<td>b4</td>
<td>0 pV, 0 V, 125000 pV, 25000 pV, 25100 pV, 0 V, 37500 pV, 2.5 V, 37600 pV, 2.5 V, 50000 pV, 2.5 V, 50100 pV</td>
</tr>
<tr>
<td>a5</td>
<td>2.5 V, 50100 pV, 2.5 V, 62600 pV, 2.5 V, 75000 pV, 2.5 V, 75100 pV, 2.5 V, 87500 pV</td>
</tr>
<tr>
<td>b5</td>
<td>2.5 V, 87600 pV, 2.5 V, 100000 pV, 2.5 V, 101000 pV, 2.5 V</td>
</tr>
<tr>
<td>a6</td>
<td>0 pV, 0 V, 125000 pV, 25000 pV, 25100 pV, 0 V, 37500 pV, 2.5 V, 37600 pV, 2.5 V, 50000 pV, 2.5 V, 50100 pV</td>
</tr>
<tr>
<td>b6</td>
<td>2.5 V, 50100 pV, 2.5 V, 62600 pV, 2.5 V, 75000 pV, 2.5 V, 75100 pV, 2.5 V, 87500 pV</td>
</tr>
<tr>
<td>a7</td>
<td>2.5 V, 87600 pV, 2.5 V, 100000 pV, 2.5 V, 101000 pV, 2.5 V</td>
</tr>
<tr>
<td>b7</td>
<td>0 pV, 0 V, 125000 pV, 25000 pV, 25100 pV, 0 V, 37500 pV, 2.5 V, 37600 pV, 2.5 V, 50000 pV, 2.5 V, 50100 pV</td>
</tr>
<tr>
<td>a8</td>
<td>2.5 V, 50100 pV, 2.5 V, 62600 pV, 2.5 V, 75000 pV, 2.5 V, 75100 pV, 2.5 V, 87500 pV</td>
</tr>
<tr>
<td>b8</td>
<td>2.5 V, 87600 pV, 2.5 V, 100000 pV, 2.5 V, 101000 pV, 2.5 V</td>
</tr>
</tbody>
</table>

pwl(Op 2.5 V, 0 V, 100000 pV, 0 V)
<table>
<thead>
<tr>
<th>Voltage</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.5v, 87600p</td>
<td>2.5v, 100000p, 2.5v, 100100p</td>
</tr>
<tr>
<td>vb8input</td>
<td>b8input</td>
</tr>
<tr>
<td>va9input</td>
<td>na9input</td>
</tr>
<tr>
<td>va9input</td>
<td>na9input</td>
</tr>
<tr>
<td>va9input</td>
<td>na9input</td>
</tr>
<tr>
<td>vb9input</td>
<td>b9input</td>
</tr>
<tr>
<td>va10input</td>
<td>na10input</td>
</tr>
<tr>
<td>va10input</td>
<td>na10input</td>
</tr>
<tr>
<td>vb10input</td>
<td>nb10input</td>
</tr>
<tr>
<td>va11input</td>
<td>na11input</td>
</tr>
<tr>
<td>vb11input</td>
<td>nb11input</td>
</tr>
<tr>
<td>va12input</td>
<td>na12input</td>
</tr>
<tr>
<td>vb12input</td>
<td>nb12input</td>
</tr>
</tbody>
</table>
2.5v, 100100p 0v)
vna1input nai1input 0 pwl(0p 2.5v, 100p 2.5v, 12500p 2.5v, 12600p 0v, 25000p 0v, 25100p 2.5v, 37500p 2.5v, 37600p 2.5v, 50000p 2.5v, 50100p 2.5v, 62500p 2.5v, 62600p 2.5v, 75000p 2.5v, 75100p 2.5v, 87500p 2.5v, 87600p 0v, 100000p 0v, 100100p 2.6v)
vai1input ali1input 0 pwl(0p 0v, 100p 0v, 12500p 0v, 12600p 2.5v, 25000p 2.5v, 25100p 0v, 37500p 0v, 37600p 0v, 50000p 0v, 50100p 0v, 62500p 0v, 62600p 0v, 75000p 0v, 75100p 0v, 87500p 0v, 87600p 2.5v, 100000p 2.5v, 100100p 0v)
vnb1input nbi1input 0 pwl(0p 2.5v, 100p 2.5v, 12500p 2.5v, 12600p 0v, 25000p 0v, 25100p 2.5v, 37500p 2.5v, 37600p 2.5v, 50000p 2.5v, 50100p 2.5v, 62500p 2.5v, 62600p 2.5v, 75000p 2.5v, 75100p 2.5v, 87500p 2.5v, 87600p 0v, 100000p 0v, 100100p 2.5v)
vai1input ali1input 0 pwl(0p 0v, 100p 0v, 12500p 0v, 12600p 2.5v, 25000p 2.5v, 25100p 0v, 37500p 0v, 37600p 0v, 50000p 0v, 50100p 0v, 62500p 0v, 62600p 0v, 75000p 0v, 75100p 0v, 87500p 0v, 87600p 0v, 100000p 0v, 100100p 0v)
vnb1input nbi1input 0 pwl(0p 2.5v, 100p 2.5v, 12500p 2.5v, 12600p 0v, 25000p 0v, 25100p 2.5v, 37500p 2.5v, 37600p 2.5v, 50000p 2.5v, 50100p 2.5v, 62500p 2.5v, 62600p 2.5v, 75000p 2.5v, 75100p 2.5v, 87500p 2.5v, 87600p 0v, 100000p 0v, 100100p 2.5v)
vai1input ali1input 0 pwl(0p 0v, 100p 0v, 12500p 0v, 12600p 0v, 25000p 0v, 25100p 0v, 37500p 0v, 37600p 0v, 50000p 0v, 50100p 0v, 62500p 0v, 62600p 0v, 75000p 0v, 75100p 0v, 87500p 0v, 87600p 0v, 100000p 0v, 100100p 0v)
vnb1input nbi1input 0 pwl(0p 2.5v, 100p 2.5v, 12500p 2.5v, 12600p 0v, 25000p 0v, 25100p 2.5v, 37500p 2.5v, 37600p 2.5v, 50000p 2.5v, 50100p 2.5v, 62500p 2.5v, 62600p 2.5v, 75000p 2.5v, 75100p 2.5v, 87500p 2.5v, 87600p 2.5v, 100000p 2.5v, 100100p 2.5v)
vai1input ali1input 0 pwl(0p 0v, 100p 0v, 12500p 0v, 12600p 0v, 25000p 0v, 25100p 0v, 37500p 0v, 37600p 0v, 50000p 0v, 50100p 0v, 62500p 0v, 62600p 0v, 75000p 0v, 75100p 0v, 87500p 0v, 87600p 0v, 100000p 0v, 100100p 0v)
vnb1input nbi1input 0 pwl(0p 2.5v, 100p 2.5v, 12500p 2.5v, 12600p 0v, 25000p 0v, 25100p 2.5v, 37500p 2.5v, 37600p 2.5v, 50000p 2.5v, 50100p 2.5v, 62500p 2.5v, 62600p 2.5v, 75000p 2.5v, 75100p 2.5v, 87500p 2.5v, 87600p 2.5v, 100000p 2.5v, 100100p 2.5v)
vai1input ali1input 0 pwl(0p 0v, 100p 0v, 12500p 0v, 12600p 0v, 25000p 0v, 25100p 0v, 37500p 0v, 37600p 0v, 50000p 0v, 50100p 0v, 62500p 0v, 62600p 0v, 75000p 0v, 75100p 0v, 87500p 0v, 87600p 0v, 100000p 0v, 100100p 0v)

* .dc v4 start=0 step=2.5 step=0.01
* .dc v8 start=0 step=2.5 step=0.01

.tran 1ps 100ns

.option post

* Mosfet models from www.mosis.org
*
* DATE: Jun 11/01
* LOT: T14Y WAF: 03
* DIE: N_Area_Fring DEV: N3740/10
* Temp= 27

.MODEL CMOSN NMOS (
+ TOX = 5.75E-9 NSUB = 1E17 LEVEL = 3
+ PHI = 0.7 VTO = 0.4238252 DELTA = 0
+ UD = 425.6466519 ETA = 0 THETA = 0.1754054
+ KP = 2.501048E-4 VMAX = 8.287851E4 KAPPA = 0.1686779

156
+ RSH = 4.062439E-3 NFS = 1E12 TPG = 1
+ XJ = 3E-7 LD = 3.162278E-11 WD =
1.232881E-8
+ CGDO = 6.2E-10 CQSO = 6.2E-10 CGBD = 1E-10
+ CJ = 1.81211E-3 PB = 0.5 MJSW = 0.3282653
+ CJSW = 5.341337E-10 MJSW = 0.5 )

.MOSP CMOSP PMOS ( LEVEL = 3
+ TOX = 5.7E-9 NSUB = 1E17 GAMMA = 0.6348369
+ PHI = 0.7 VTO = -0.5536085 DELTA = 0
+ UD = 250 ETA = 0 THETA = 0.1573195
+ KP = 5.194153E-5 VMAX = 2.295325E5 KAPPA = 0.7448494
+ RSH = 30.0776952 NFS = 1E12 TPG = -1
+ XJ = 2E-7 LD = 9.968346E-13 WD =
5.475113E-9
+ CGDO = 6.66E-10 CQSO = 6.66E-10 CGBD = 1E-10
+ CJ = 1.893569E-3 PB = 0.9906013 MJSW =
0.4664287
+ CJSW = 3.625544E-10 MJSW = 0.5 )
*
* .alter * N = 1, 2, 4 P = 4, 8
.param NO.5=1.00u
.param M1.0=1.00u
.param M1.5=2.00u
.param N8.0=7.00u
.param P1.0=4.00u
.param P2.0=4.00u
.param P3.0=4.00u
.param P16.0=16.00u

.alter * N = 1, 2, 4 P = 2, 4, 8
.param NO.5=1.00u
.param M1.0=1.00u
.param M1.5=2.00u
.param N8.0=7.00u
.param P1.0=2.00u
.param P2.0=2.00u
.param P3.0=2.00u
.param P16.0=12.00u

.alter * N = 0.5, 1, 4 P = 2, 4, 10
.param NO.5=2.00u
.param M1.0=2.00u
.param M1.5=2.00u
.param N8.0=8.00u
.param P1.0=2.00u
.param P2.0=2.00u
.param P3.0=2.00u
.param P16.0=16.00u
.end
Full Adder

* SPICE fulladder.sp

* XOR
mp1 Vdd clk xnor1 Vdd cmosn l=0.25u w=1.00u

mn1 Vss cik eval1 Vss cmosn l=0.25u w=1.00u
mn2 eval1 A int0 Vss cmosn l=0.25u w=1.00u
mn3 eval1 nA int1 Vss cmosn l=0.25u w=1.00u
mn4 int0 nB xnor1 Vss cmosn l=0.25u w=1.00u
mn5 int1 B xnor1 Vss cmosn l=0.25u w=1.00u

mp2 Vdd xnor1 outxor1 Vdd cmosp l=0.25u w=3.00u
mn6 Vss xnor1 outxor1 Vss cmosn l=0.25u w=0.50u

* XNOR
mp3 Vdd clk xor1 Vdd cmosp l=0.25u w=1.00u

mn7 Vss clk eval2 Vss cmosn l=0.25u w=1.00u
mn8 eval2 A int2 Vss cmosn l=0.25u w=1.00u
mn9 eval2 nA int3 Vss cmosn l=0.25u w=1.00u
mn10 int2 B xor1 Vss cmosn l=0.25u w=1.00u
mn11 int3 nB xor1 Vss cmosn l=0.25u w=1.00u

mp4 Vdd xor1 outxnor1 Vdd cmosp l=0.25u w=3.00u
mn12 Vss xor1 outxnor1 Vss cmosn l=0.25u w=0.50u

* AND
mp5 Vdd clk xnor2 Vdd cmosp l=0.25u w=1.00u

mn13 Vss clk eval3 Vss cmosn l=0.25u w=1.00u
mn14 eval3 C int4 Vss cmosn l=0.25u w=1.00u
mn15 eval3 nC int5 Vss cmosn l=0.25u w=1.00u
mn16 int4 outxnor1 xnor2 Vss cmosn l=0.25u w=1.00u
mn17 int5 outxor1 xnor2 Vss cmosn l=0.25u w=1.00u

mp6 Vdd xnor2 S Vdd cmosp l=0.25u w=3.00u
mn18 Vss xnor2 S Vss cmosn l=0.25u w=0.50u

* OR
mp7 Vdd clk nand1 Vdd cmosp l=0.25u w=1.00u

mn19 Vss clk eval4 Vss cmosn l=0.25u w=1.00u
mn20 eval4 A int6 Vss cmosn l=0.25u w=1.00u
mn21 int6 B nand1 Vss cmosn l=0.25u w=1.00u

mp8 Vdd nand1 and1 Vdd cmosp l=0.25u w=3.00u
mn22 Vss nand1 and1 Vss cmosn l=0.25u w=0.50u

* AND
mp9 Vdd clk nand2 Vdd cmosp l=0.25u w=1.00u

mn23 Vss clk eval5 Vss cmosn l=0.25u w=1.00u
mn24 eval5 outxor1 int7 Vss cmosn l=0.25u w=1.00u
mn25 int7 C nand2 Vss cmosn l=0.25u w=1.00u

mp10 Vdd nand2 and2 Vdd cmosp l=0.25u w=3.00u
mn26 Vss nand2 and2 Vss cmosn l=0.25u w=0.50u

* OR
mp11 Vdd clk nor Vdd cmosp l=0.25u w=1.00u

mn27 Vss clk eval6 Vss cmosn l=0.25u w=1.00u
mn28 eval6 and1 nor Vss cmosn l=0.25u w=1.00u
mn29 eval6 and2 nor Vss cmosn l=0.25u w=1.00u
mp12 Vdd nor Cout Vdd cmosp l=0.25u w=3.00u
mn30 Vss nor Cout Vss cmosn l=0.25u w=0.50u

cLoad1 S Vss 25fF
*cLoad2 Cout Vss 25fF

vDD Vdd 0 2.5
vSS Vss 0 0
* A 1 1 1 0 1 0 1
* B 1 0 1 0 1 1 1
* C 1 1 1 0 1 0 1

vClk clk 0 pw1(0000p 2.5v, 525p 2.5v, 625p 0.0v, 1150p 0.0v, 1250p 2.5v, 1775p 2.5v, 1875p 0.0v, 2400p 0.0v, 2500p 2.5v, 3025p 2.5v, 3125p 0.0v, 3650p 2.5v, 3750p 2.5v, 4275p 2.5v, 4375p 0.0v, 4900p 0.0v, 5000p 2.5v, 5525p 2.5v, 5625p 0.0v, 6150p 2.5v, 6250p 2.5v, 6775p 2.5v, 6875p 0.0v, 7400p 0.0v, 7500p 2.5v)

vA A 0 pw1(0000p 2.5v, 1150p 2.5v, 1250p 2.5v, 2400p 2.5v, 2500p 2.5v, 2500p 2.5v, 3650p 2.5v, 3750p 0.0v, 4900p 0.0v, 5000p 2.5v, 6150p 2.5v, 6250p 0.0v, 7400p 0.0v, 7500p 2.5v)

vB B 0 pw1(0000p 2.5v, 1150p 2.5v, 1250p 0.0v, 2400p 0.0v, 2500p 2.5v, 2500p 2.5v, 3650p 2.5v, 3750p 0.0v, 4900p 0.0v, 5000p 2.5v, 6150p 2.5v, 6250p 2.5v, 7400p 2.5v, 7500p 2.5v)

vC C 0 pw1(0000p 2.5v, 1150p 2.5v, 1250p 2.5v, 2400p 2.5v, 2500p 0.0v, 2500p 0.0v, 3650p 0.0v, 3750p 0.0v, 4900p 0.0v, 5000p 2.5v, 6150p 2.5v, 6250p 0.0v, 7400p 0.0v, 7500p 2.5v)

vMA mA 0 pw1(0000p 0.0v, 1150p 0.0v, 1250p 0.0v, 2400p 0.0v, 2500p 0.0v, 3650p 0.0v, 3750p 2.5v, 4900p 2.5v, 5000p 0.0v, 6150p 0.0v, 6250p 2.5v, 7400p 2.5v, 7500p 0.0v)

vMB mB 0 pw1(0000p 0.0v, 1150p 0.0v, 1250p 2.5v, 2400p 2.5v, 2500p 2.5v, 2500p 2.5v, 3650p 0.0v, 3750p 2.5v, 4900p 2.5v, 5000p 0.0v, 6150p 0.0v, 6250p 0.0v, 7400p 0.0v, 7500p 0.0v)

vMC mC 0 pw1(0000p 0.0v, 1150p 0.0v, 1250p 0.0v, 2400p 0.0v, 2500p 2.5v, 2500p 2.5v, 2500p 2.5v, 3650p 2.5v, 3750p 2.5v, 4900p 2.5v, 5000p 0.0v, 6150p 0.0v, 6250p 2.5v, 7400p 2.5v, 7500p 0.0v)

*.dc vA start=0 stop=2.5 step=0.01
*.dc vB start=0 stop=2.5 step=0.01
.tran 1ps 8ns
.option post

* Mosfet models from www.mosis.org

* DATE: Jun 11/01
* LOT: T14Y
* DIE: N_Area_Fring
* Temp= 27
*.MODEL CMOSN NMOS (LEVEL = 3
 + TOX = 5.7E-9 NSUB = 1E17 GAMMA = 0.4317311
 + PHI = 0.7 VTO = 0.4238252 DELTA = 0
 + UC = 425.6466519 ETA = 0 THETA = 0.1754054
 + KP = 2.501048E-4 VMAX = 8.28786154 KAPPA = 0.1867797
 + RSH = 4.062439E-3 NFS = 1E12 TPG = 1
 + XJ = 3E-7 LD = 3.162278E-11 WD =
 + 1.232881E-8
 + CGDO = 6.3E-10 CGSO = 6.2E-10 CGBO = 1E-10
 + CJ = 1.81211E-3 PB = 0.5 MJ = 0.3282553
 + CJSW = 5.341337E-10 MJSW = 0.5)
.MODEL CMOSP PMOS ( LEVEL = 3
+ TOX = 5.7E-9 NSUB = 1E17 GAMMA = 0.6348369
+ PHI = 0.7 VTO = -0.5536085 DELTA = 0
+ UD = 250 ETA = 0 THETA = 0.1573195
+ KP = 5.194153E-5 VMAX = 2.295325E5 KAPPA = 0.7448494
+ RSH = 30.0776952 NFS = 1E12 TPG = -1
+ XJ = 2E-7 LD = 9.968346E-13 WD =
 5.475113E-9
+ CGDO = 6.66E-10 CGSO = 6.66E-10 CGBD = 1E-10
+ CJ = 1.893569E-3 PB = 0.9906013 MJ =
 0.4664287
+ CJSW = 3.625544E-10 MJSW = 0.5 )
*
.end
Bibliography


