Table of Contents

Part 2 - The Logic-Electronics Interface

Part 3 - The Design of VLSI Systems

An Overview of VLSI

1

VLSI is an acronym that stands for **very-large-scale integration**. This somewhat nebulous term is used to collectively refer to the many fields of electrical and computer engineering that deal with the analysis and design of very dense electronic integrated circuits. Although a strict definition is difficult to come by, one commonly used metric is to say that a VLSI contains more than a million (10^6) or so switching devices or logic gates. Early in the first decade of the 21st century, the actual number of **transistors** (the switching devices) has exceeded 100 million (10^8) for the more complex designs on a piece of silicon (a **chip**), which is typically about 1 centimeter on a side.

This book has been written to provide an understanding of the basics of **digital VLSI chip design**. Emphasis is placed on presenting the details of translating a system specification to a small piece of silicon. The treatment is very technical with many details. Some statements and analyses will appear immediately obvious, while others may not make sense until later chapters. This occurs because the field of VLSI engineering encompasses several distinct "areas of specialization" that mesh together in a unique manner. The most difficult aspect of learning VLSI is seeing the common theme that links the areas together. Once this is accomplished, you are on your way to understanding one of the most fascinating fields of modern times.

1.1 Complexity and Design

Engineering a VLSI chip is an extremely complex task. When attempting to describe the field to a non-technical group, the idea of the "VLSI design funnel" shown in Figure 1.1 helps break the ice. This views the process as one where we provide the basic necessities such as money, an idea, and

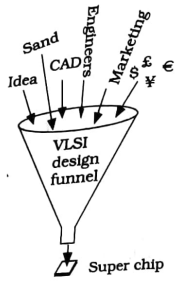


Figure 1.1 The VLSI design funnel

marketing information and dump them all into a "magic technology funnel." Adding a pile of sand as a raw material produces the super chip at the bottom that will sell millions of units and hopefully revolutionize the world. And maybe make someone rich. Of course, engineers and scientists are needed somewhere in the process, but they just put the things together. Unfortunately, the process is slightly more complicated than portrayed in this example.

Any system that is composed of millions of elements is inherently difficult to understand. One human mind cannot process information of the complexity that is required for the design and implementation. Creating a **design team** provides a realistic approach to approaching a VLSI project as it allows each person to study small sections of the system. In a modern design, hundreds of engineers, scientists, and technicians may be working different parts of the design. However, since the team is working on a single project, it is important that each team member have some understanding of where their work falls within the overall scheme. This is accomplished by means of the **design hierarchy**, where the chip is viewed at many different "levels" from the abstract to the physical implementation. Every level is important, and each has subdivisions that can evolve into a lifetime career.

In our treatment of VLSI, we will continually stress the fact that the field is inherently multidisciplinary in nature. Specialists in an uncountable number of areas are needed to produce a working functional design. Computer architects must interact with code writers and logic designers, and they must be able to comprehend some of the problems of circuit design and silicon processing. Electronics experts must move beyond circuits to see how their units will affect the system. And everyone depends upon the computer-aided design tools and the support groups that perform the 10,000 or so other tasks not described here. If this description

makes the field sound complicated, that's because it is. VLSI is not a simple discipline to understand. But it is possible to learn the basics in a reasonable amount of time. Persons who end up working in the area usually gravitate there because one or more aspects catch their interest and fall within their background.

Now that we have an appreciation of what is involved, let us move to a better description of the design process. An overview with the major steps in the sequence is shown in Figure 1.2. The starting point of a VLSI design is the system specification. At this point, the product is defined in both general and specific terms that provide design targets such as functions, speed, size, etc., for the entire project. This is the "Top" level of the design hierarchy. The system specifications are used to create an abstract, high-level model. Digital design is usually based on some type of **hardware description language** (HDL) that allows abstract modeling of the operation. VHDL and Verilog™ are the most common HDLs in practice, but several others (including C and C++) are used. The abstract model contains information on the behavior of each block and the interaction among the blocks in the system. The model is subjected to extensive verification steps where the design is checked and rechecked to ensure that it is correct.

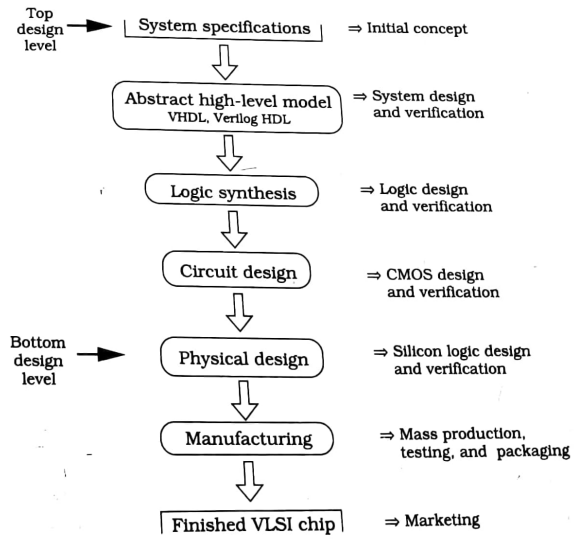


Figure 1.2 General overview of the design hierarchy

The next step in the process is called **synthesis**. The abstract log
model is used to provide the **logical design** of the network by specifyin
the primitive gates and units needed to build each unit. This then form
the basis for transferring the design to the **electronic circuit leve**
where transistors are used as switches and Boolean variables are treate
as varying voltage signals. To create a transistor, we move down anothe
level to that of **physical design**. At this level, the network is built on
tiny area on a slice of silicon using a complex mapping scheme tha
translated transistors and wires into extremely fine-line patterns of me
als and other materials. The physical design level constitutes the Botto
of the design hierarchy. After the design process is completed, the projec
moves on to the manufacturing line. The final result is a finished elec
tronic VLSI chip.

When we start at the system level specification, the design process i
called the **top-down** approach. The initial work is quite abstract and the
oretical and there is no direct connection to silicon until many steps hav
been completed. The reverse approach starts at the silicon or circuit leve
and builds primitive units such as logic gates, adders, and registers a
the first steps. These are combined to obtain larger and more comple
logic blocks, which are then used as building blocks in even large
designs. This **bottom-up** approach is acceptable for small projects, bu
the complexity of modern VLSI designs makes it impractical; it is
extremely difficult to design a functional 64-bit microprocessor by start
ing with single bits.

A bottom-up study of the various aspects of VLSI does work well fo
learning the basics of the field. This approach has therefore been chosen
for the first half of the book. We will start simple and evolve into higher
levels of complexity and abstraction. Our goal is to present a coherent
understanding of the field as a single entity made up of many different
areas. Even if a discussion seems overly specialized, it will be linked to
other concepts later. Once we have achieved an understanding of the
basics, we are in a position to study the problem from a higher level. The
second half of the book introduces the system aspects of VLSI to complete
the picture.

1.1.1 Design Flow Example

As an example of a design hierarchy, let us determine what would be
entailed in the design of a basic microprocessor. The initial conception
could be at the system level where the instruction set and components are
defined. An *instruction* is a primitive operation (such as adding two binary
numbers) that the microprocessor is designed to execute; the instruction
set is the group of all instructions for a particular processor. A component
is a digital logic unit that provides a specific function (such as addition).
The field of computer architecture is concerned with the units that make
up the computer and how they are connected together.

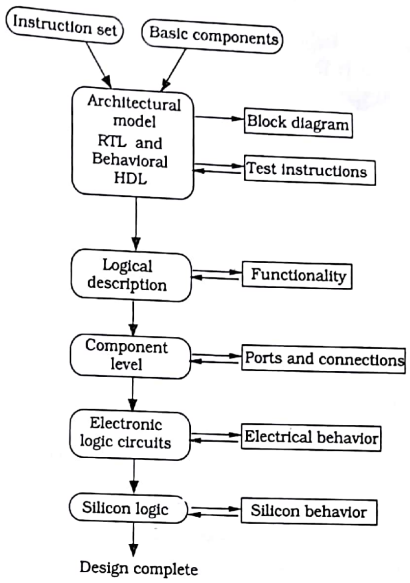


Figure 1.3 A simple design flow for a microprocessor

A basic design flow for the problem is shown in Figure 1.3. The instruction set and component group can be used to construct a high-level model of the architecture. At this level, the behavior of the system is described in an abstract manner that ignores the low-level details needed to actually build the network. For example, we may define an addition event by writing

Register_X ←A + B

which is translated as saying that the sum of A and B is transferred to a storage device named Register_X. High-level abstractions of this type can be used to define the processor architecture, and are commonly known as the register-transfer level (RTL) description. RTL models describe the operation of the system without reference to specific components. When written with an HDL, it can be used to test instructions and verify the architectural behavior Abstract design allows us to construct a block diagram for the system.

RTL code can be translated to an equivalent description that contains more detail about the operation and behavior components. The operation of each block can be summarized at the HDL **behavioral level**, where the emphasis is on the large-scale behavior of the blocks as they interact with

other sections. Behavioral modeling at this stage is extremely critical as it is used to verify the architecture; any problems must be solved before progressing further.

The next stage of the design process involves translating the system blocks into a logic model that is based upon Boolean equations and logic gates. This takes the abstract design to a more tangible level, and is the first step toward realizing a hardware design. Two approaches can be used for this stage: **automated design and synthesis**, or **custom design** of the logic circuit. Automated design is based on a set of CAD (computer-aided design) tools that run on high-performance workstations. A synthesis tool usually accepts HDL code and creates the corresponding logic network with a predefined set of rules. Properly written HDL code can produce logic designs very quickly, and automated synthesis is used for all noncritical sections. Custom design is used when special problems arise and the synthesis solution does not meet the necessary specifications. Various logic equations and networks are derived and tested as a means of solving the problem at hand. This is an intense, time-consuming process, so it is reserved for critical sections.

The logic model produces functional components, which are then translated to electronics. Characteristics of the silicon circuits become important at this stage of the design process. Given a large-scale function, one can usually find several equivalent logic expressions; all produce the same output, but will use different equations and gates. Recall, for example, how a Karnaugh map is used to simplify logic equations. Silicon VLSI is complicated by the fact that each type of logic gate or circuit has distinct characteristics, and we must often search for circuits that are faster or smaller than what can be obtained using an obvious solution. A sophisticated synthesis tool can take HDL code and provide suggested designs for both the logic gates and the silicon circuits. However, the toolsets have not yet reached the level where they are powerful enough to produce the "best" design, whatever that may mean.

After the logic network and circuits have been designed, the next step is to use the information to produce an integrated circuit at the physical design level. This is accomplished in a series of steps where transistors are defined as 3-dimensional structures on a chip of silicon, and are then placed and wired using another set of graphical CAD tools. Once this is accomplished, the designs are tested and verified, and then used to create a database that allows the manufacturing line to actually build the electronic chip. Fabricating a VLSI chip is itself a complex specialized field. Once started, it may take several weeks to produce the final circuits.

The specifics of the procedures are much more complicated than what is portrayed in the simple flowchart. It does, however, illustrate the essence of a top-down design flow. VLSI design is concerned with filling in the details needed to produce a manufactured chip that functions as

designed with high reliability and a long lifetime. And can be sold at a profit, of course.

1.1.2 VLSI Chip Types

At the engineering level, digital VLSI chips are classified by the approach used to implement and build the circuit. A **full-custom** design is one where every circuit is custom designed for the project. This is an extremely tedious and time-consuming process that makes it impractical for designing an entire system.

Application-specific integrated circuits (ASICs) allow digital designers to create ICs for a particular application. ASICs are very popular for prototyping or low-volume production runs. They are designed using an extensive suite of CAD tools that portray the system design in terms of standard digital logic constructs: state diagrams, function tables, and logic diagrams. Usually, an ASIC designer does not need any knowledge of the underlying electronics or the actual structure of the silicon chip. Design automation CAD tools are responsible for taking the logic design and building most of the chip. One drawback of ASIC design is that all characteristics such as speed are set by the architectural design; the designer does not have access to the electronics, so delay times cannot be changed. Modern ASICs have evolved to a high level of sophistication, and are generally capable of providing solutions to a large class of problems.

A **semi-custom** design is in between that of a full-custom and an ASIC-type circuit. The majority of the chip is designed using a group of primitive predefined cells as building blocks. Each cell provides a basic function, such as a logic operation or a storage circuit, and the master design resides in a database collection called a library. A cell entry contains all of the information needed to create the circuit on silicon. If it is not possible to meet the system specifications using the cell library, then the semi-custom approach permits the designer to engineer a solution by creating alternate silicon circuits that have the desired characteristics. These are used only in small sections where the problems occur. For example, floating point circuits in microprocessors can be extremely complex, so that some sections may require custom design to meet the clocking budget. Variations of semi-custom design are used for most high-performance chips.

1.2 Basic Concepts

The objective of this book is to present the field of VLSI in its entirety. Overall, VLSI design is a system design discipline. Many aspects can be taught without any reference to the underlying silicon circuits. System solutions can be generated using the CAD tools, and the necessary data turned over to the manufacturing group for production. While this

approach produces functional solutions, it makes many of the details invisible to the designer. Simplifying the design process is important. However, many of the most powerful techniques and ideas of VLSI reside at lower levels and are therefore lost. Circuits work, but they are not as fast or as small as they could have been.

VLSI should be thought of as a single discipline that deals with the conception, design, and manufacture of complex integrated circuits. Many system-level concepts are based on the characteristics of electronic circuits that are made at the silicon level. When Carver Mead of Caltech pioneered the field in the 1970's, one of the most important foundations for VLSI arose from his observation that digital electronic integrated circuits could be viewed as a set of geometrical patterns on the surface of a silicon chip. Groups of patterns represented different logic functions and were repeated many times in the system. Complexity could thus be dealt with using the concept of repeated patterns that were fitted together in a structured manner. Signal flow and data movement could be followed by tracing the paths of the metallic "lines" that carried electricity. It was possible to write Boolean expressions that could be directly translated to geometrical patterns on silicon in a well-defined manner. The microphotograph of a CMOS chip in Figure 1.4 shows many of these features in a finished device. Note in particular the repeated patterns and ordered placement of rectangular lines, polygons, and groups of geometric patterns.[1] Mead's observation (and a huge amount of work) has structured VLSI into the important field it is today. The importance of gaining an overall unified view of VLSI becomes clear.

VLSI design encompasses many practical aspects of digital system design. One is the fact that even the most powerful **system on a chip** (SOC) must be interfaced to other components to create an operational unit. This is achieved by placing the silicon circuitry in the center of a rectangular piece of material, and then providing some type of scheme that allows external wires to contact it. Figure 1.5 shows the use of **bonding pads**, which are square metal sections where wires can be bonded and connected to the package in which the chip is mounted. A more advanced technique developed by IBM is called the C4 technology; it allows metal "bumps" to be located across the surface area. Contact to the package is established by "flipping" the chip so that the bumps are on the bottom and can be aligned to a wiring grid. Regardless of the approach, there is a limit on the actual size of the chip.

In an ideal world, we could make the chip as large as desired. This would allow increasingly complex systems to be designed without any bounds. Unfortunately, it is not possible to manufacture a functional

[1] This is a section of a binary adder network designed at Georgia Tech. Each group of patterns adds two bits and produces sum and carry outputs.

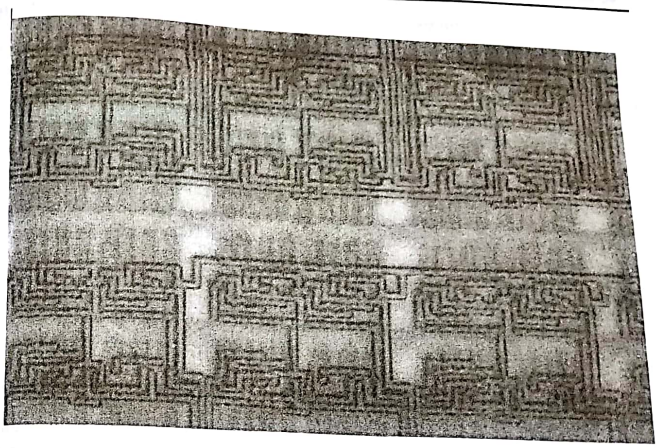


Figure 1.4 Micrograph of a section of a digital CMOS integrated circuit

design because of defects in the silicon crystal structure that cannot be avoided. The larger the area of the circuit, the higher the probability that a defect will occur. Even a single bad transistor or connection renders the circuit nonfunctional, so we attempt to keep the overall size of the chip small. We note in passing that other problems in manufacturing also limit the size of the chip.

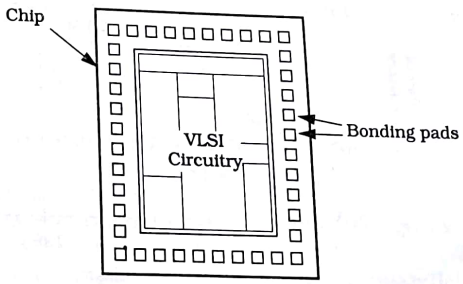


Figure 1.5 Bonding pad frame for interfacing

To overcome this limit we have adopted the philosophy that shrinkin
the size of a transistor will allow more devices to be placed in a given ch
area. Technologically, this is a very difficult problem. Precision design an
manufacturing techniques give devices where the smallest dimension
around 1.3×10^{-7} meters. At this level, we change our measurement me
ric to the micrometer (μm), or **micron** for short, such that 1 μm = 10^{-6}
$= 10^{-4}$ cm, and refer to the technology as a 0.13-μm process.

One of the classical predictions in VLSI transistor densities is know
as **Moore's Law**. Gordon Moore, one of the cofounders of the Intel Corp
ration, visualized in the 1970's that chip building technology wou
improve very quickly. He projected that the number of transistors on
chip would double about every 18 months. Although there have been va
ations due to technological problems or economic slowdowns, Moore
Law has proved amazingly close to actual trends. Figure 1.6 shows a pl
of device count as a function of year for a group of randomly selecte
microprocessor chips from major vendors. There have always bee
debates as to how long the transistor count can continue to increase
this rate due to technological limitations in reducing the size. Regardle
of the actual slope, however, it seems clear that VLSI design will remain
powerful force for many years to come.

This short introduction to some of the problems of VLSI illustrates th
vast nature of the field itself. The role of a VLSI design group is to create
large, complex system on a tiny piece of silicon. The group faces co
straints at every level, from the abstract modeling and timing down
building a chip with millions of transistors. Project status presentation
engineering summaries, and critical deadlines are always present.

Welcome to the exciting world of VLSI!

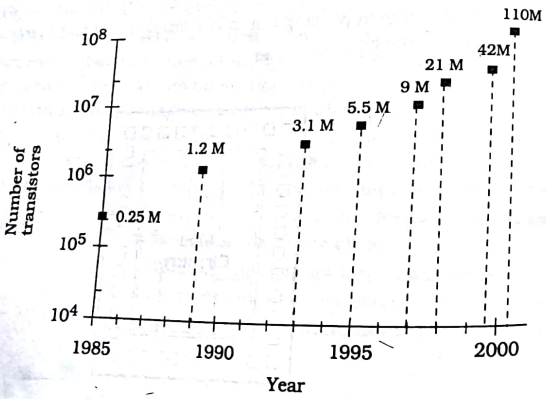


Figure 1.6 Device count by year

1.3 Plan of the Book

The book has been divided into three main sections. For self-study, it is best to follow them sequentially, although it is not necessary to read every section in a first reading.

Part 1 is entitled *Silicon Logic*, and includes Chapters 2 through 5. The material examines the techniques for designing logic networks in silicon. It concentrates on introducing transistor logic circuits and how they translate to patterns on a silicon chip. Details of the CMOS processing sequence are presented, and applied to realistic chip design. After completing Part 1, the reader will be able to design a myriad of CMOS logic gates at both the circuit and silicon levels.

The electronics of VLSI are covered in Part 2, which is entitled *The Logic-Electronics Interface*. Part 2 includes Chapters 6 through 9, with the more advanced concepts in Chapters 8 and 9. Transistor switching characteristics are presented and then used to analyze digital electronic logic gates. The treatment is quite detailed, but it concentrates on the important basics that affect system performance and switching speeds. Completing this section of the book will provide a solid understanding of the relationship between logic design and electrical characteristics.

System-level problems are addressed in Part 3, *The Design of VLSI Systems*, which includes Chapters 10 through 16. The basics of Verilog® HDL are presented as the vehicle for system-level modeling. Many VLSI logic components such a multiplexors, adders, and memories are studied in Chapters 11 through 13. Large-scale chip design issues are addressed in Chapters 14 and 15. The book concludes with an introduction to digital testing in the final chapter.

An effort has been made to present the material in a readable, coherent manner that concentrates on explaining the details. This is particularly true in Part 1, where the reader is exposed to subject matter that is not typically found in other courses.

So, without further delay, let us begin our trek into the world of very-large scale integration.

1.4 General References

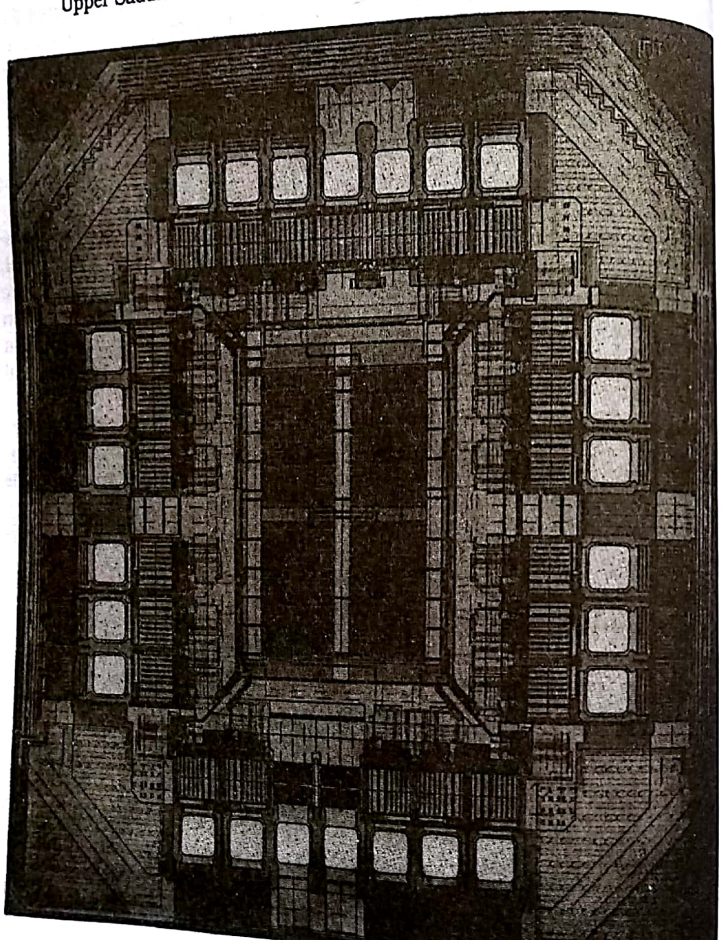
[1] Dan Clein, **CMOS IC Layout**, Newnes Publishing Co., Boston, 2000.

[2] Randy H. Katz, **Contemporary Logic Design**, Benjamin-Cummings Publishing Co., Redwood City, CA, 1994.

[3] Ken Martin, **Digital Integrated Circuit Design**, Oxford University Press, New York, 2000.

[4] Jan Rabaey, **Digital Integrated Circuits**, Prentice-Hall, Upper Saddle River, NJ, 1996.

[5] Michael John Sebastian Smith, **Application-Specific Integrated Cu**
cuits, Addison-Wesley Longman Inc., Reading, MA, 1997.
[6] John P. Uyemura, **A First Course in Digital Systems Desig**
Brooks-Cole Publishers, Pacific Grove, CA, 2000.
[7] John P. Uyemura, **CMOS Logic Circuit Design**, Kluwer Academ
Press, Norwell, MA, 1999.
[8] John P. Uyemura, **Physical Design of CMOS Integrated Circuit**
Using L-Edit®, PWS /Brooks-Cole Publishers, Pacific Grove, CA, 1995.
[9] M. Michael Vai, **VLSI Design**, CRC Press. Boca Raton, FL, 2001.
[10] Neil H.E. Weste and Kamran Eshraghian, **Principles of CMOS VLS**
Design, 2nd ed., Addison-Wesley Publishing Co., Reading, MA, 1993.
[11] Wayne Wolf, **Modern VLSI Design**, 2nd ed., Prentice-Hall PT
Upper Saddle River, NJ, 1998.



Part 1

Silicon Logic

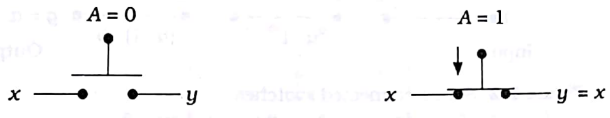
Logic Design with MOSFETs 2

CMOS integrated circuits use bi-directional devices called MOSFETs as logic switches. This chapter examines the logical characteristics of MOSFETs and develops techniques for building digital networks.

2.1 Ideal Switches and Boolean Operations

All digital designs are based on primitive logic operations. The first task in our study of VLSI will be to create electronic logic gates that can be used as building blocks in complex switching networks.

Logic gates are created by using sets of **controlled switches**. The characteristics of an **assert-high** controlled switch are illustrated by the drawings in Figure 2.1. In this idealized situation, the state of the switch (**open** or **closed**) is determined by the value of the control variable A. In Figure 2.1(a), the control bit has the value of $A = 0$ which is defined to give an open switch. This means that there is no relationship between the two variables x and y as represented by the gap between the left and right sides. The opposite case is a closed switch where we visualize the top portion of the switch being "pushed down" as shown in Figure 2.1(b). This condition occurs when $A = 1$ and connects the two sides of the switches



(a) Open (b) Closed

Figure 2.1 Behavior of an assert-high switch

so that

$$y = x \tag{2.}$$

is valid. If we interpret the left side variable x as the input and the rig side as the output, then we can say that the condition $A = 1$ allows th input variable to flow through the switch and establish the value of th output. This is called an "assert-high" switch because a high control bit $A = 1$ is required to close the circuit.

A different approach to characterizing the behavior of the switch is write the logic equation[1]

$$y = x \cdot A \quad \text{iff} \quad A = 1 \tag{2.}$$

By itself, the relationship between x and y is undefined if $A = 0$. Althoug this appears to be a serious deficiency, in practice we will avoid the prob lem by using additional switches to define the value of y for this case.

Let us now proceed to create a logic network by combining the concep of an ideal switch with a voltage source. Suppose that we take tw switches that are controlled by the independent variables a and b an connect them as shown by the diagram in Figure 2.2. The two switche are said to be **in series** with each other. As we trace the signal pat through the first switch, equation (2.2) shows that the output (direct after the switch) is given by $a \cdot 1$ as indicated on the drawing. This acts a the input to the second switch, so that applying equation (2.2) agai yields

$$g = (a \cdot 1) \cdot b = a \cdot b \tag{2.}$$

for the output. This is easy to interpret using a qualitative analysis: bot switches must be closed with $a = 1$ AND $b = 1$ to allow the input 1 t reach the output and result in $g = 1$. The circuit appears to provide th AND2 operation.[2] However, note that equation (2.2) is valid only if th control bit has a value of 1; it if is 0, then there is no direct relationshi between the left and right sides of the switch. There are three other poss

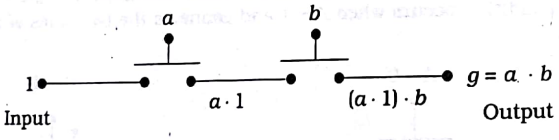


Figure 2.2 Series-connected switches

[1] We use the shorthand mathematical notation "iff" to mean "if, and only if".

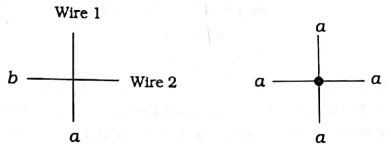
[2] We denote a 2-input AND operation as an AND2. This type of notation will be used for all gates. For exam ple, an OR2 operation implies a 2-input OR.

bilities for the two inputs:

$$(a, b) = (1, 0), (0, 1), (0, 0) \tag{2.4}$$

Any of these input combinations should result in a logical output of $g = 0$ but the logic equations say that g is undefined.

Before proceeding any further, let us clarify our approach to portraying logic networks. In general, the switch drawings will be called **schematic diagrams** since they show the "scheme" used in the wiring. We extend this terminology to include diagrams that contain electronic devices. Since we want to keep the drawings relatively compact and neat looking, wiring lines will often cross one another in the drawing. When this occurs, we will adopt the convention shown in Figure 2.3. In Figure 2.3(a), Wire 1 and Wire 2 are assumed to be totally separate. The signal a on Wire 1 has no relationship to the signal b on Wire 2. If we wish to create a connection, we will use a "dot" as in Figure 2.3(b). In this case, the two wires are connected so that placing a signal a on one of the lines results in the same value on all points of both lines.



(a) No connection (b) Connection

Figure 2.3 Connection convention used in schematic diagrams

Let us examine another circuit that has the same problem. Figure 2.4 shows two switches that are controlled by the independent variables a and b, but the two are wired **in parallel** with one another; this means that the left (input) sides are connected together and the right (output) sides are connected together. The output f can be constructed by recog-

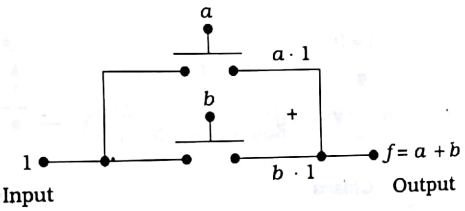


Figure 2.4 Parallel-connected switches

nizing that according to equation (2.2), the top switch produces an outp
of $a \cdot 1$ iff $a = 1$, while the lower switch produces an output of $b \cdot 1$ iff b
1. Both expressions are shown at the appropriate points in the diagra
We conclude that if either $a = 1$ OR $b = 1$ (or both), then the output
described by the single expression

$$g = a + b \qquad (2.$$

which appears to be the OR2 operation at this point in the analysis. Pa
allel-connected switches can thus be used to OR variables together; this
indicated on the diagram by including the "+" between the switches. No
however, that if $a = 0$ and $b = 0$ at the same time, then the output g of t
switching network is undefined. It thus fails to provide the entire O
function.

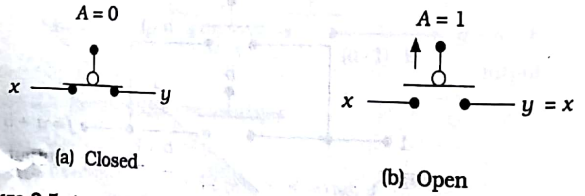
The preceding examples illustrate that switches have characteristi
that can be used as a basis for implementing logic operations. Howev
since the logic equation (2.2) is valid only if the switch is closed, we we
not able to obtain complete AND and OR gates as neither network cou
produce a logic 0 output.

It is useful at this point to introduce another type of switch th
behaves in the exact opposite manner. This is called an **assert-low** swit
and is defined to have the characteristics illustrated in Figure 2.5. W
have added a logic "bubble" to the top of the symbol to distinguish it fro
an assert-high switch. By definition, an assert-low switch is closed wh
the control bit is at a value of $A = 0$ as shown in Figure 2.5(a). To open t
switch we must apply a value of $A = 1$ to the device as in Figure 2.5(
This behavior can be described by the logic equation

$$y = x \cdot \bar{A} \qquad \text{iff } A = 0 \qquad (2.$$

In this case, the value of y is not defined if $A = 1$. Comparing the two typ
of switches we see that they behave in a complementary manner.

As an example of how this type of switch can be used, consider th
series-connected pair in Figure 2.6. Tracing the signal path from th
input through the first switch gives an output of $\bar{a} \cdot 1$ which is valid iff a
0. This acts as the input to the second switch so that the output of th



(a) Closed (b) Open

Figure 2.5 An assert-low switch

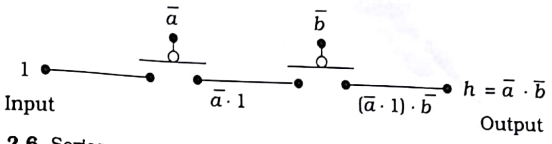


Figure 2.6 Series-connected complementary switches

series chain is given by

$$\begin{aligned} h &= (\bar{a} \cdot 1) \cdot \bar{b} \\ &= \overline{a + b} \end{aligned} \qquad (2.7)$$

where we have used the DeMorgan relation to write the second line. This looks like the NOR2 operation. However, since the second switch must be closed with $b = 0$, this result is correct only if both $a = 0$ and $b = 0$. If either a or b is a 1, then g is undefined. We thus have the same type of problem experienced in our earlier examples.

Let us now progress to the idea of using both types of switches in a single network. We will provide both logic 1 and logic 0 inputs in an effort to produce an output that is defined for all possible input combinations. In Figure 2.7, the assert-high switch SW1 is used to connect a logic 0 input to the output y, while the assert-low switch SW2 connects a logic 1 input to y. The variable a controls both switches. Since the two are in parallel, we may write the OR relation between the upper and lower branches to give the output in the form

$$y = \bar{a} \cdot 1 + a \cdot 0 \qquad (2.8)$$

The operation of the circuit can be understood by specifying a value for a. If $a = 0$, then SW1 is open and SW2 is closed which gives

$$y = \bar{0} \cdot 1 + 0 \cdot 0 = 1 \qquad (2.9)$$

If $a = 1$, then SW1 is closed and SW2 is open. Substituting into the expression we have

$$y = \bar{1} \cdot 1 + 1 \cdot 0 = 0 \qquad (2.10)$$

This circuit thus eliminates the problem of an undefined voltage. Moreover, since logically $a \cdot 0 = 0$, the expression reduces to

$$y = \bar{a} \qquad (2.11)$$

In other words, this circuit implements the NOT operation.

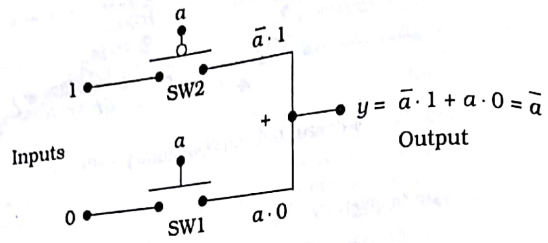


Figure 2.7 A switch-based NOT gate

$$y = \text{NOT}(a) = \bar{a} \tag{2.12}$$

This demonstrates that using two switches with opposite characteristics allows us to build a network with well-defined results.

The NOT circuit in Figure 2.7 is based on the behavior of a 2:1 multiplexor as shown in Figure 2.8. The MUX uses the input a to select between input 0 (that has a "1" applied to it) when $a = 0$, or input 1 (that has a "0" applied to it) when $a = 1$. The output is given by the expression

$$y = \bar{a} \cdot 1 + a \cdot 0 \tag{2.13}$$

which reduces to $y = \bar{a}$. A close examination of the switching circuit in Figure 2.7 verifies that there is a one-to-one correspondence with the 2:1 multiplexor.

2.2 MOSFETs as Switches

MOSFETs are electronic devices that are used to direct and control logic signals in high-density digital IC design.[3] The acronym "MOSFET" stands

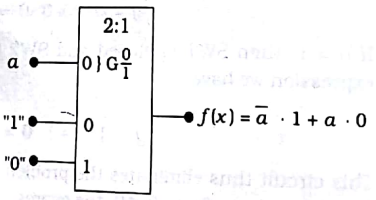


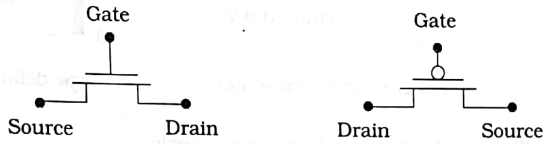
Figure 2.8 A MUX-based NOT gate

[3] "MOSFET" is pronounced as *moss-fet*.

for **metal-oxide-semiconductor field-effect transistor**, but we will not worry about the details just yet. In many ways, MOSFETs behave like the idealized switches introduced in the previous section. There are important differences that must be taken into account before they are used. These arise from the fact that MOSFETs must obey circuit equations and their ultimate performance is limited by the laws of physics. In this section we will concentrate on creating switching models for the devices. The more complicated aspects of current flow will be discussed in later chapters.

Complementary MOS (**CMOS**) uses two types of MOSFETs to create logic networks. One type is called an n-channel MOSFET (or nFET for short), and uses negatively charged electrons for electrical current flow. The circuit symbol for an nFET is shown in Figure 2.9(a). The **gate** terminal acts as the control electrode for the device. Applying a voltage on the gate electrode determines the current flow between the **drain** and **source** terminals. The other type of transistor is called a p-channel MOSFET or pFET. It uses positive charges for current flow and has the circuit symbol drawn in Figure 2.9(b). The only graphical difference between the nFET and pFET symbols is the inversion bubble at the gate. As with the nFET, the voltage applied to the gate determines the current flow between the source and drain terminals. Do not confuse the **gate terminal** of a MOSFET with a **logic gate**, as the two "gates" are not related. The context of the discussion always helps clarify the usage.

MOSFETs are intrinsically electronic devices. To use them as logic-controlled switches, we must first define how to translate between Boolean values and electrical parameters. This is accomplished by using voltages that exist on the chip when we apply an external power supply. In the most general case, two power supply voltages V_{DD} and V_{SS} are defined as shown in Figure 2.10. The reference terminal is taken to be the **ground** connection (which is at 0 volts) between the two sources so that the chip receives both a positive power voltage V_{DD} and a negative power supply voltage V_{SS}. Early generations of silicon MOS logic circuits used both positive and negative supply voltages. However, modern designs require only a single positive voltage V_{DD} and the ground connection; common values are $V_{DD} = 5$ V and 3.3 V or lower. The remaining source is set to a value of



(a) nFET symbol (b) pFET symbol

Figure 2.9 Symbols used for nFETs and pFETs

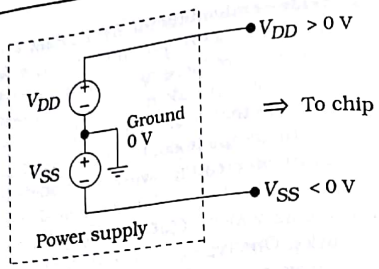


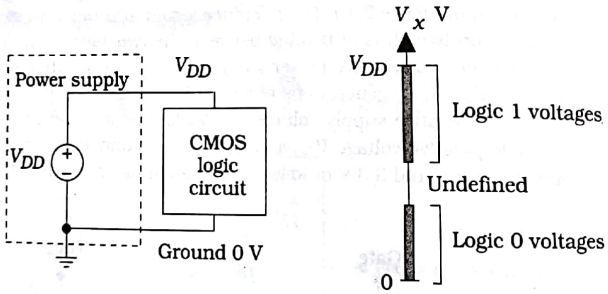
Figure 2.10 Dual power supply voltages

$V_{SS} = 0$ V, which results in the power supply network portrayed in Fig 2.11(a).[4] We will assume that all of our circuits use only a single posit voltage source V_{DD}. In practice it is still common to use V_{SS} to denote lowest voltage in the circuit such that V_{SS} has an implied value of 0 V.

We can now define the relationship between logic variables and v ages. Recall that Boolean variables are discrete; a binary variable x have the value of $x = 0$ or $x = 1$ only. At the circuit level we represent variable x using a voltage V_x such that

$$0 \le V_x \le V_{DD} \qquad (2.$$

gives the normal range of values with a power supply providing 0 V a V_{DD} directly to the circuit. The **positive logic convention** then defi the **ideal** logic 0 and logic 1 voltages as



(a) Power supply connection (b) Logic definitions

Figure 2.11 Single voltage power supply

[4] The unit of volt is denoted by V in the text.

$x = 0$ means that $V_x = 0$ V
$x = 1$ means that $V_x = V_{DD}$ (2.15)

Realistic circuits are more lenient and allow us to use a *range* of voltages for both logic 0s and logic 1s as portrayed in Figure 2.11(b). In general,

- Low voltages correspond to logic 0 values
- High voltages correspond to logic 1 values

The transition region between the highest logic 0 voltage and the lowest logic 1 voltage is undefined in that it does not represent either a 0 or a 1. The actual extent of both voltage ranges is determined by the characteristics of the logic circuits and will be dealt with later.

With the logic-voltage conversion defined, let us now examine the switching characteristics of MOSFETs. Ideally, an nFET behaves like an assert-high switch. This is shown in Figure 2.12 where A is the logic variable applied to the gate. If $A = 0$ corresponding to a low voltage, then the nFET acts like an open switch and there is no relationship between the left and right sides; this is illustrated in Figure 2.12(a). Increasing the gate voltage to a high value is the same as changing to $A = 1$. This results in a closed switch as shown in Figure 2.12(b). As with the assert-high switch, this can be described by the logic equation

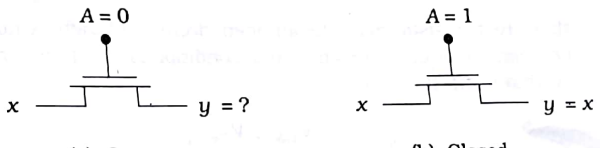
$$y = x \cdot A \qquad (2.16)$$

which is valid iff $A = 1$.

The pFET is exactly opposite in that it behaves like an assert-low switch. In Figure 2.13(a), the signal applied to the gate has a logical value of $A = 1$ corresponding to a high voltage. This gives an open circuit and there is no direct relationship between x and y. If the gate voltage is reduced to give $A = 0$, then the pFET acts as a closed switch. This allows us to write the ideal relationship

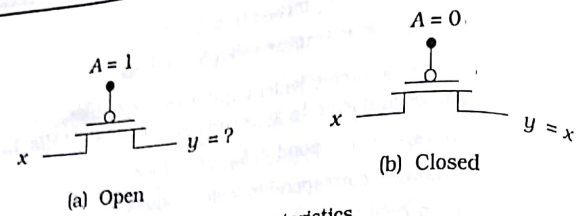
$$y = x \cdot \bar{A} \qquad (2.17)$$

which is valid so long as $A = 0$ is true; this condition is shown in Figure 2.13(b).



(a) Open (b) Closed

Figure 2.12 nFET switching characteristics



(a) Open (b) Closed

Figure 2.13 pFET switching characteristics

MOSFETs allow us to design logic circuits using the techniques assert-high and assert-low switching networks. However, FETs are phy cal devices that do not behave exactly like the ideal switch models abo This is not a severe problem so long as we understand the differences a learn the limitations.

FET Threshold Voltages

The switching equations assume that the binary variable A applied to t gate of a FET is either a 0 or a 1. The corresponding voltage V_A is a phy cal quantity and does not behave in such a discrete manner. Moreover, want to define a range of voltages for both cases of $A = 0$ and $A = 1$ to a in the design of working circuits. Every MOSFET has a characteriz parameter called the **threshold voltage** V_T that helps us define t important gate voltage ranges. The specific value of V_T is established du ing the manufacturing process, and is thus taken to be a given value the VLSI designer. One complicating factor is that nFETs and pFETs ha different threshold voltages.

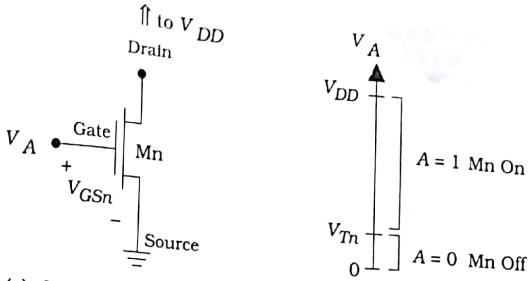
An nFET is characterized by a threshold voltage V_{Tn} that is a positi number with values around $V_{Tn} = 0.5$ V to 0.7 V being typical. The mea ing of V_{Tn} can be understood by referring to the parameters shown in F ure 2.14(a). First note that the drain terminal has been identified as t one closest to the power supply V_{DD}, while the source terminal has be connected to ground (0 V). The gate-source voltage V_{GSn} shown in t drawing is the important parameter that determines whether the nFE acts as an open or closed switch. In particular, if

$$V_{GSn} \le V_{Tn} \qquad (2.1$$

then the transistor acts like an open circuit and there is no current flo between the drain and source; this condition is said to describe a transi tor that is **off**. If instead

$$V_{GSn} \ge V_{Tn} \qquad (2.1$$

then the nFET drain and source are connected and the equivalent switc is closed. A transistor that conducts current is said to be **on**. This beha



(a) Gate-source voltage (b) Logic translation

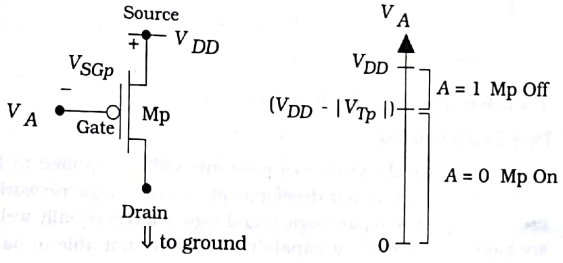
Figure 2.14 Threshold voltage of an nFET

ior allows us to create the voltage diagram shown in Figure 2.14(b) to define the voltage V_A that is associated with the binary variable A. In particular, we note that

$$V_A = V_{GSn} \qquad (2.20)$$

This shows that $A = 0$ corresponds to values of $V_A \le V_{Tn}$, while $A = 1$ implies that $V_A \ge V_{Tn}$. These relations establish the voltage ranges needed to control the nFET.

A pFET behaves in a **complementary** manner. Consider the transistor shown in Figure 2.15(a). For the pFET, the source terminal has been connected to the power supply V_{DD} while the drain is the side closest to ground; this is opposite to that used for the nFET. In this device, the source-gate voltage V_{SGp} is the important applied voltage. By convention, the pFET threshold voltage V_{Tp} is referenced to the gate-source voltage



(a) Source-gate voltage (b) Logic translation

Figure 2.15 pFET threshold voltage

V_{GSp} and is a negative number with typical values in the range of ab
V_{Tp} = - 0.5 V to V_{Tp} = - 0.8 V. In this book we will describe pFETs by us
V_{SGp} = - V_{GSp} as this allows us to use the absolute value $|V_{Tp}|$ for
threshold voltage. The meaning of the threshold voltage is as follows. I

$$V_{SGp} \leq |V_{Tp}| \qquad (2.$$

then the pFET is off and it acts as an open switch. Conversely, a la
source-gate voltage of

$$V_{SGp} \geq |V_{Tp}| \qquad (2.$$

turns the pFET on and it behaves as a closed switch. To relate this beh
ior to the applied voltage V_A we first sum voltages to write

$$V_A + V_{SGp} = V_{DD} \qquad (2.$$

Thus,

$$V_A = V_{DD} - V_{SGp} \qquad (2.$$

shows that a low value of V_A implies a large V_{SGp} and the pFET is on. S
ilarly, if V_A is large then V_{SGp} is small and the pFET is off. This gives
the logic 0 and logic 1 ranges summarized in Figure 2.15(b). Note that
transition between a logic 0 and a logic 1 is at

$$V_{DD} - |V_{Tp}| \qquad (2.$$

since this corresponds to the source-gate voltage where the device tu
on.

It is important to note that the logic 0 and logic 1 voltage ranges of
are different for the two types of FETs. One way around this problem is
note that there are regions of overlap for both $A = 0$ and $A = 1$ values th
can be used if a uniform definition is needed. The ideal values of

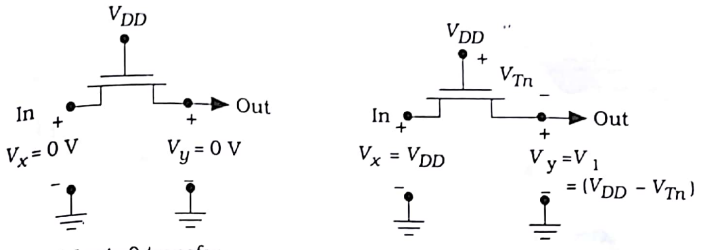
$$V_A = 0 \text{ V}$$
$$V_A = V_{DD} \qquad (2.$$

are, however, valid for both devices.

Pass Characteristics

An ideal electrical switch can pass any voltage applied to it. This w
implicitly assumed in our development of switch logic networks where
used the switches to pass logic 0 and logic 1 levels equally well. MOSFE
are more limited in their capabilities and are not able to pass arbitra
voltages from source to drain or vice versa.

Let us examine the pass characteristics of nFETs first. Figure 2.
summarizes the behavior of the device when we attempt to use it to pas



(a) Logic 0 transfer (b) Logic 1 transfer

Figure 2.16 nFET pass characteristics

voltage from left to right. Applying V_{DD} to the gate insures that the nFET is on and the device acts like a closed switch. In Figure 2.16(a) a logic 0 voltage of $V_x = 0$ V is applied to the left side. This results in an output voltage of $V_y = 0$ V as desired. Increasing the input voltage results in that value being transmitted to the output side. However, a problem occurs if we apply an ideal logic 1 input voltage of $V_x = V_{DD}$ as shown in Figure 2.16(b). In this case, the output voltage V_y is reduced to a value

$$V_1 = V_{DD} - V_{Tn} \qquad (2.27)$$

which is less than the input voltage V_{DD}. This is referred to as a **threshold voltage loss**. It arises from the fact that the minimum value of the gate-source voltage need to maintain an on state is

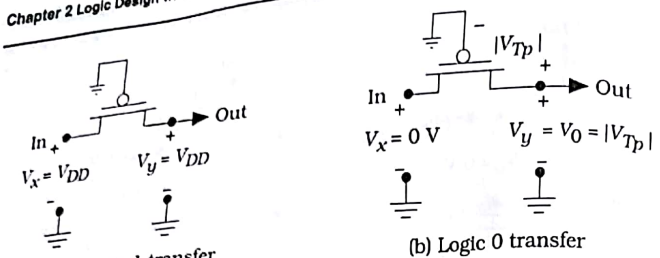
$$V_{GSn} = V_{Tn} \qquad (2.28)$$

Using the Kirchhoff voltage law, this subtracts from the voltage V_{DD} that is applied to the *gate* as shown in the drawing.[5] Since the transmitted voltage V_y is less than the ideal logic 1 value of V_{DD}, we say that the nFET can only pass a **weak** logic 1. In the same terminology, the nFET is said to pass a **strong** logic 0 since it is capable of producing an output voltage of $V_y = 0$ V without any problems. In general, the nFET can pass a voltage in the range $[0, V_1]$, but nothing above V_1.

A pFET has opposite pass characteristics. To examine the pFET properties we apply a logic 0 to the gate by grounding it. Figure 2.17 shows the circuits for both input values. Figure 2.17(a) portrays the case where $V_x = V_{DD}$ corresponding to a logic 1 input. The output voltage is

$$V_y = V_{DD} \qquad (2.29)$$

[5] Kirchhoff's voltage law (KVL) says that the algebraic sum of voltages around a closed loop is 0.



(a) Logic 1 transfer (b) Logic 0 transfer

Figure 2.17 pFET pass characteristics

which is an ideal logic 1 level. The pFET is therefore capable of passing strong logic 1 voltage. The problem arises when we attempt to pass a ideal logic 0 voltage of $V_y = 0$ V and is presented in Figure 2.17(b). In th case, the transmitted voltage can only drop to a minimum value of

$$V_y = |V_{Tp}| \qquad (2.3($$

This is also due to a threshold effect. In order to keep the pFET o requires a minimum source-gate voltage of

$$V_{SGp} = |V_{Tp}| \qquad (2.3|$$

as shown in the drawing. Since the gate is at 0 V, this represents a rise t a voltage of $|V_{Tp}|$, which is turn affects the output. Obviously, the pFE transmits a weak logic 0 voltage. In summary, a pFET can pass a voltag in the range $[V_{DD}, V_0]$, but nothing below V_0.

Let us restate the results of the above discussion:

- nFETs pass strong logic 0 voltages, but weak logic 1 values
- pFETs pass strong logic 1 voltages, but weak logic 0 levels

Complementary MOS (CMOS) circuits are designed to account for th transmission levels. In particular, we can write down the following rule as a basis for our design:

1. Use pFETs to pass logic 1 voltages of V_{DD}
2. Use nFETs to pass logic 0 voltages of $V_{SS} = 0$ V

These allow us to build circuits that can pass the ideal logic voltages 0 V and V_{DD} to the output terminal. We will find, however, that ideal levels are not always needed in practice.

2.3 Basic Logic Gates in CMOS

The concept of a general CMOS digital logic gate can be understood by referring to the drawing in Figure 2.18. In this example, *a*, *b*, and *c* are the

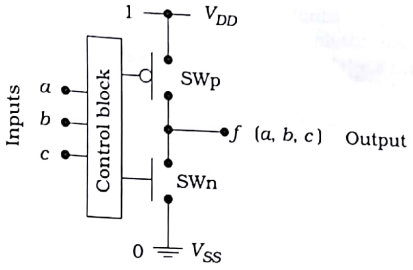
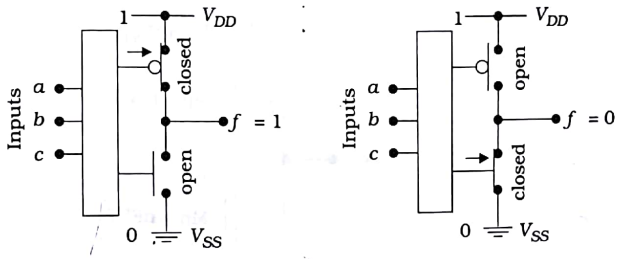


Figure 2.18 General CMOS logic gate

input bits that combine to give the output function bit $f(a, b, c)$. Since this is by definition a digital circuit, all of the quantities are restricted to values of 0 or 1. Digital logic circuits are nonlinear networks that use transistors as electronic switches to divert *one* of the supply voltages V_{DD} or 0 V to the output. This corresponds to a logical result of $f = 1$ or $f = 0$. Internally, we may view the output network of the gate as consisting of two switches SWp (an assert-low device) and SWn (an assert-high device) as shown. These are wired in to insure that one switch is closed while the other switch is open.

The operation of the general logic gate is shown in Figure 2.19 for both output possibilities. In Figure 2.19(a), the upper switch is closed while the lower switch is open. This connects the output to the power supply and yields a value of $f = 1$. The opposite situation is shown in Figure 2.19(b): the upper switch is open and the lower switch is closed. Because the output is now connected to $V_{SS} = 0$ V, the logical result is $f = 0$. Although this



(a) $f = 1$ output (b) $f = 0$ output

Figure 2.19 Operation of a CMOS logic gate

view is quite simplified, it does illustrate how CMOS logic circuits oper
The only missing feature in this model is the method used to control
output switches using the input variables. This is accomplished w
MOSFETs.

The Complementary Pair

CMOS logic circuits are based on the concept of using **complement**
pairs of transistors for switching. A complementary pair consists o
pFET and an nFET that have their gate terminals connected together
shown in Figure 2.20. The input signal x simultaneously controls t
conduction through both FETs. Note that the top of the pFET Mp
assumed to be close to the power supply voltage V_{DD}, while the nFET M
is close to the ground (V_{SS}). The behavior of the complementary pair
easily understood by observing the state of each FET for the two possib
input values as in Figure 2.21. An input of $x = 0$ turns Mp on while t
nFET Mn is off and acts like an open switch; this is shown in Figu
2.21(a). The opposite case shown in Figure 2.21(b) is where $x = 1$. Now t
pFET MP is off while Mn is on. The name "complementary" is derived fro
this operation: when one FET is on, the other is off. The important aspe
of this behavior is that the nFET and pFET are electrical opposites, whic
translates directly into a coherent switching scheme.

Now that we have seen the overall structure of CMOS logic gates an
the idea of a complementary pair, we have all of the concepts needed
create and analyze basic logic gate circuits.

2.3.1 The NOT Gate

The NOT or INVERT function is often considered the simplest Boolea
operation. It has an input x and produces an output $f(x)$ of

$$f(x) = \text{NOT}(x) = \bar{x} \qquad (2.3$$

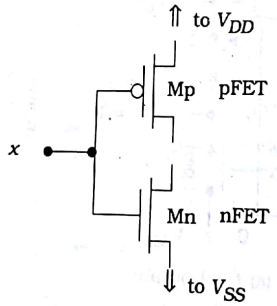
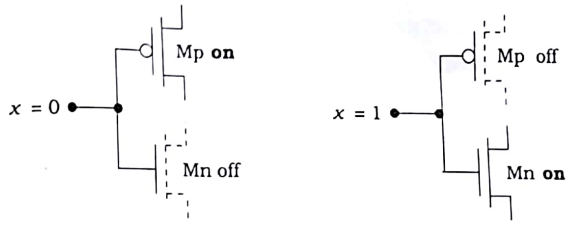


Figure 2.20 A CMOS complementary pair



(a) $x = 0$ input (b) $x = 1$ input

Figure 2.21 Operation of the complementary pair

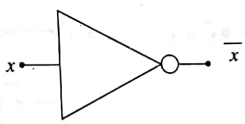
such that

$$\begin{aligned} &\text{If } x = 0 \quad \text{then } \bar{x} = 1 \\ &\text{If } x = 1 \quad \text{then } \bar{x} = 0 \end{aligned} \qquad (2.33)$$

defines the notation. The logic symbol and truth table are provided in Figure 2.22 for future reference.

A CMOS NOT gate is shown in Figure 2.23. This has been constructed using the same idea as for the switch-based circuit discussed earlier in the context of Figure 2.7. The circuit uses a complementary pair of MOSFETs such that the input variable x controls both transistors.

The operation follows directly from the properties of the complementary pair. If the input x has a value of 0, then pFET MP is on and the nFET Mn is off. As shown in Figure 2.24(a), this connects the output node to the power supply voltage V_{DD}, giving an output of $\bar{x} = 1$. Conversely, if $x = 1$ then Mp is off and Mn is on. The output is then connected to the ground node and gives $\bar{x} = 0$ as verified by the circuit in Figure 2.24(b). It is clear that this simple circuit does indeed provide the NOT operation. This can be verified analytically by applying the FET logic rules to write the output f as



(a) Logic symbol

x	$\bar{x}$
0	1
1	0

(b) Truth table

Figure 2.22 NOT gate

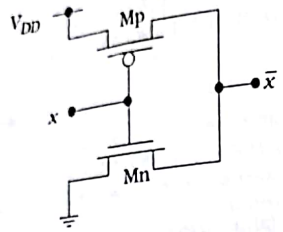


Figure 2.23 CMOS NOT gate

$$f = \bar{x} \cdot 1 + x \cdot 0 \qquad (2.$$

where the first term describes Mp and the second term is due to Mn. Si plifying gives

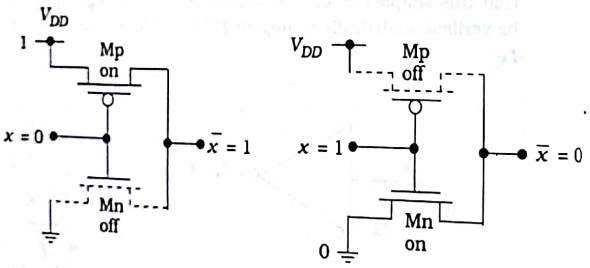
$$f = \bar{x} \qquad (2.$$

as expected.

One of the most important characteristics of the CMOS NOT gate is t manner in which the complementary FET pair insures that, for a giv input logic state of $x = 0$ or 1, the output is connected to either V_{DD} ground and gives a well-defined value. This circuit specifically avoids t possibilities where (i) both FETs are off at the same time, or (ii) both FE are on at the same time; either situation would give an ill-defined outpu

2.3.2 The CMOS NOR Gate

Now that we have seen the basic NOT gate, let us extend the concepts t create a 2-input NOR gate using the same principles. These are



(a) $x = 0$ input (b) $x = 1$ input

Figure 2.24 Operation of the CMOS NOT gate

- Use a complementary nFET/pFET pair for each input
- Connect the output node to the power supply V_{DD} through pFETs
- Connect the output node to ground through nFETs, and
- Insure that the output is always a well-defined high or low voltage

This set of guidelines helps us design logic circuits that have input and output characteristics which are compatible with the NOT gate.

The logic symbol and truth table for the NOR2 gate are provided in Figure 2.25.[6] With input variables x and y, the NOR2 produces the output

$$g(x, y) = \overline{x + y} \qquad (2.36)$$

such that a 1 at either input gives $g = 0$. Only the input combination $(x, y) = (1,1)$ yields an output of $g = 1$.

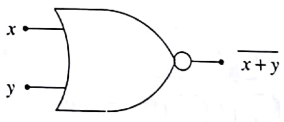
One way to synthesize the NOR2 operation at the logic design level is to use a 4:1 MUX as shown in Figure 2.26(a). Path selection is obtained using the input pair (x, y) such that every combination gives either a 1 or a 0 to the output. The Boolean expression for the output of the MUX is

$$g(x, y) = \bar{x} \cdot \bar{y} \cdot 1 + \bar{x} \cdot y \cdot 0 + x \cdot \bar{y} \cdot 0 + x \cdot y \cdot 0 \qquad (2.37)$$

which reduces to the desired form

$$g(x, y) = \overline{x + y} \qquad (2.38)$$

using the DeMorgan theorem. A voltage-equivalent circuit is obtained by replacing the binary quantities with voltages, and results in the circuit shown in Figure 2.26(b). In the notation of the drawing, V_x and V_y respectively represent the Boolean variables x and y. This information provides the basis for constructing the CMOS NOR2 circuit.



(a) Logic symbol

x	y	$\overline{x+y}$
0	0	1
0	1	0
1	0	0
1	1	0

(b) Truth table

Figure 2.25 NOR logic gate

[6] The terminology "NOR2" means a 2-input NOR gate.

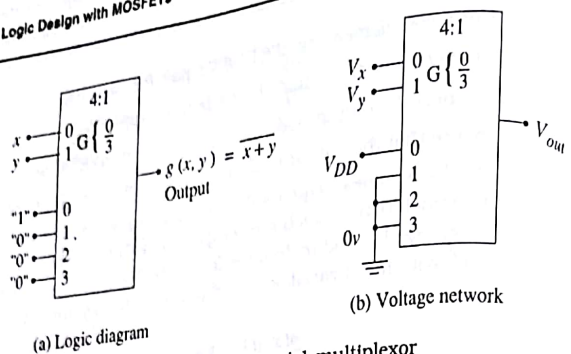


Figure 2.26 NOR2 operation using a 4:1 multiplexor

One approach to building the logic gate is to use the Karnaugh m drawn in Figure 2.27. CMOS generally produces **inverting** logic becau our gates are constructed using the NOT circuit as a basis. This creat the situation where we are generally interested in the occurrence of bo 1's and 0's when dealing with K-maps. In particular, note that we ha created two 0-groupings in the drawing. The map allows us to write t logic expression in the form

$$g(x, y) = \bar{x} \cdot \bar{y} \cdot 1 + x \cdot 0 + y \cdot 0 \qquad (2.3$$

and work backward to construct the circuit. Each term represents a Fl path to the output. The first term connects the output to 1 (the pow supply V_{DD}) and is controlled by complements of the input variables in series-connected AND arrangement. The second and third terms repr sent two independent nFET paths between the output and 0 (groun Combining these statements results in the CMOS NOR2 circuit shown

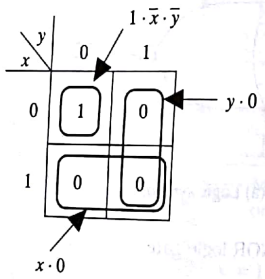


Figure 2.27 NOR2 gate Karnaugh map

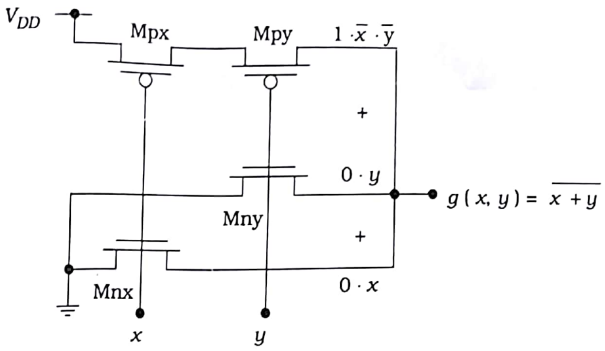


Figure 2.28 CMOS NOR2 gate

Figure 2.28; the one-to-one correspondence between each line in the equation and the circuit is obvious.

To verify that this circuit does have the proper electrical behavior, we may construct the table shown in Figure 2.29. This shows the state (on or off) of every FET for each of the four input possibilities. Tracing the output connections for each possibility easily shows that the switching circuit is consistent with the truth table.

x	y	Mpx	Mpy	Mnx	Mny	g
0	0	on	on	off	off	1
0	1	on	off	off	on	0
1	0	off	on	on	off	0
1	1	off	off	on	on	0

Figure 2.29 Operational summary of the NOR2 gate

The electrical structure of the NOR2 gate also illustrates another important point in the manner in which the FETs are wired together. Note that the two pFETs Mpx and Mpy are connected in series such that both must be on to establish a conducting path from V_{DD} to the output. The nFETs Mnx and Mny, on the other hand, are wired in parallel so that a connection between the output and ground is created if either nFET is on. This is called a **series-parallel** transistor arrangement; the principle allows us to design more complex gates.

As an example, let us construct a 3-input NOR (NOR3) gate using the NOR2 topology as a guideline. Let us label the inputs as x, y, and z. Each input is connected to the gate of a complementary nFET/pFET pair. The logical output expression for the gate is given by

$$f = \overline{x+y+z} \qquad (2.4$$

This says that the output has a value $f = 0$ if one or more of the inputs
a logic 1. Since output 0's are controlled by the nFETs, placing the thr
nFETs in parallel gives the proper functional behavior. If we apply th
principle of series-parallel structuring, then the pFETs should be in seri
with one another. Figure 2.30 shows the logic circuit constructed in th
manner; note the similarity with the NOR2 circuit in Figure 2.28. We ca
verify the operation of the NOR3 logic gate by inspection: if any input is
1, then the output is connected to ground giving $f = 0$. The only case tha
yields an output of $f = 1$ is if all three inputs are 0; this turns on all thre
pFETs while simultaneously turning the nFETs off.

Another approach to verifying the logic is to use the equations of FE
switches and derive the MUX equations. The top branch in Figure 2.30
through a series group of three pFETs as described by the term

$$1 \cdot \bar{x} \cdot \bar{y} \cdot \bar{z} \qquad (2.4$$

where we recognize that the power supply voltage V_{DD} is equivalent to
logic 1. Each of the three nFET branches consists of a single FET passin
the ground to the output. Since a ground is a logic 0, we can OR the fou
branches together to give a complete output expression of

$$f = 1 \cdot \bar{x} \cdot \bar{y} \cdot \bar{z} + 0 \cdot x + 0 \cdot y + 0 \cdot z \qquad (2.42)$$

The nFET terms insure that the output voltage of the circuit is 0 V when-
ever one or more of the inputs is 1. Logically, however, they evaluate to
leaving the final form

$$f = 1 \cdot \bar{x} \cdot \bar{y} \cdot \bar{z} = \overline{x+y+z} \qquad (2.43)$$

where we have use a DeMorgan relation in the reduction. This shows tha

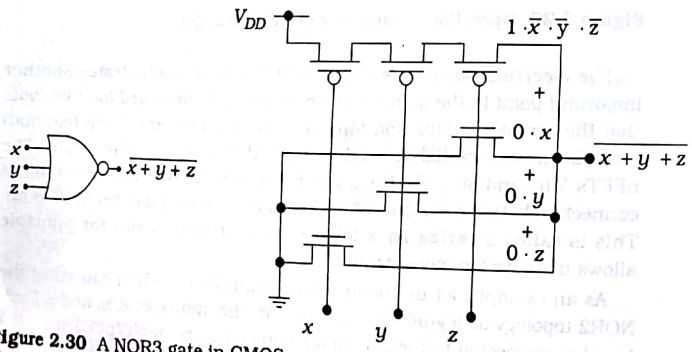


Figure 2.30 A NOR3 gate in CMOS

the circuit does in fact provide the NOR3 operation.

In principle, one may use the same arguments to construct multiple input NOR gates in CMOS such as a NOR4 or a NOR6. This technique is easy to apply and yields functional logic circuits. For VLSI applications, however, the choice of logic circuits is based on more than just the ability to provide a logic operation. Hardware characteristics such as switching speed and area consumption on the silicon chip must be taken into account. In this chapter, we will concentrate solely on forming logic functions through the circuit topology. More detailed considerations will be discussed later in the text.

2.3.3 The CMOS NAND Gate

Let us next construct the CMOS circuit for the NAND2 gate with the logical symbol and behavior summarized in Figure 2.31. This gate is characterized by an output that is 0 unless both of the inputs are 1. The truth table can be used to build the 4:1 MUX implementation drawn in Figure 2.32(a) such that the output is described by

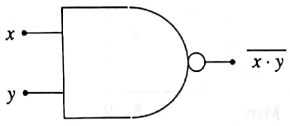
$$h(x, y) = \bar{x} \cdot \bar{y} \cdot 1 + \bar{x} \cdot y \cdot 1 + x \cdot \bar{y} \cdot 1 + x \cdot y \cdot 0 \qquad (2.44)$$

The voltage-equivalent network in Figure 2.32(b) is now somewhat obvious.

As with the NOR2 gate, it is useful to examine the Karnaugh map for the NAND2 function. Figure 2.33 shows the map along with two groupings that simplify the cases where $h = 1$. Using these reductions, our expression can be rewritten to read

$$h(x, y) = \bar{x} \cdot 1 + \bar{y} \cdot 1 + x \cdot y \cdot 0 \qquad (2.45)$$

Translating each term to FET groups yields the CMOS circuit shown in Figure 2.34. This gives the NAND2 function as can be verified by the oper-

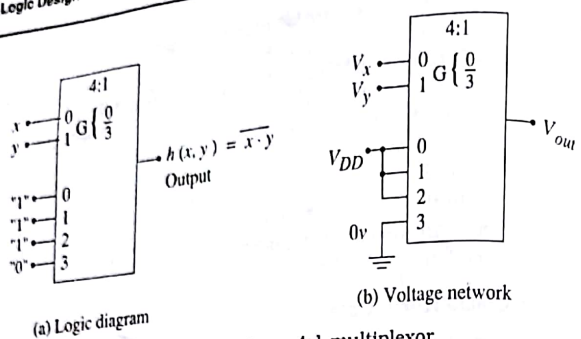


(a) Logic symbol

x	y	$\overline{x \cdot y}$
0	0	1
0	1	1
1	0	1
1	1	0

(b) Truth table

Figure 2.31 NAND2 logic gate



(a) Logic diagram (b) Voltage network

Figure 2.32 NAND2 operation using a 4:1 multiplexor

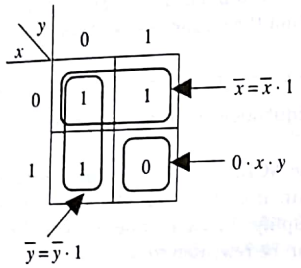


Figure 2.33 NAND2 K-map

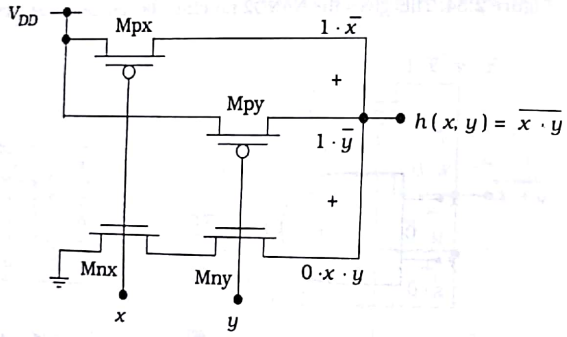


Figure 2.34 CMOS NAND2 logic circuit

x	y	Mpx	Mpy	Mnx	Mny	h
0	0	on	on	off	off	1
0	1	on	off	off	on	1
1	0	off	on	on	off	1
1	1	off	off	on	on	0

Figure 2.35 Operational summary of the NAND2 circuit

ation summarized in the table of Figure 2.35. An important characteristic of the NAND2 gate is that it uses two parallel-connected pFETs, while the nFETs are in series. This is exactly opposite to the structure of the NOR2 gate.

A NAND3 gate can be created using the same topology. It requires three sets of complementary pairs, each driven by a separate input. The nFETs are placed in series, while the pFETs are wired in parallel. This gives the gate shown in Figure 2.36. To verify the operation of the circuit, note that all three inputs must be 1's to provide a conduction path between the output and ground. If any one (or more) of the inputs is a 0, then the corresponding nFET is off while the pFET it drives acts like a closed switch; this gives a logic 1 voltage of V_{DD} at the output.

Switch logic analysis can also be applied by treating the circuit as a multiplexor. The series-connected nFET chain at the bottom of the circuit is described by the logic term

$$0 \cdot x \cdot y \cdot z \tag{2.46}$$

Each pFET branch consists of a single transistor that acts like a closed switch when a 0 is applied. Performing the OR operation among the four

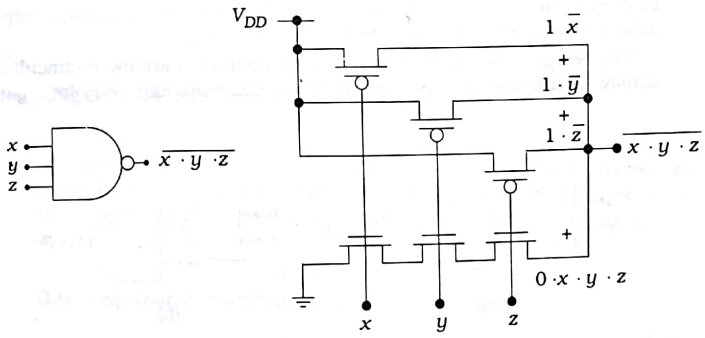


Figure 2.36 A NAND3 logic gate in CMOS

branches gives

$$0 \cdot x \cdot y \cdot z + 1 \cdot \bar{x} + 1 \cdot \bar{y} + 1 \cdot \bar{z} \qquad (2.47)$$

Eliminating the 0 terms and using DeMorgan reduction gives the output function as

$$\overline{x \cdot y \cdot z} \qquad (2.48)$$

which is the NAND3 function. This technique can be extended to design the CMOS circuitry for NAND gates with more inputs.

2.4 Complex Logic Gates in CMOS

One of the most powerful aspects of building logic circuits in CMOS is the ability to create a single circuit that provides several primitive operations (NOT, AND, and OR) in an integrated manner. These will be called **complex** or **combinational logic gates** in our discussion. Complex logic gates are very useful in VLSI system-level design.

To illustrate the main idea of a complex logic gate, consider the Boolean expression

$$F(a, b, c) = \overline{a \cdot (b + c)} \qquad (2.49)$$

The simplest way to construct the logic network for this function is to use one OR-gate, one AND-gate, and one NOT-gate as shown in Figure 2.37(a). Alternately, we might simplify the network to that in Figure 2.37(b) if a NAND2 gate is available. If we build the electronic equivalent of either implementation, then the traditional approach would be to use a one-to-one map: each gate requires one electronic logic circuit. For the first case (a), this would require three separate gates, while (b) reduces the gate count to two. For many applications, this method is perfectly acceptable; it is intuitive and straightforward to implement.

The design constraints on a VLSI implementation are more difficult to satisfy. Transistors occupy area on the silicon chip, and every logic gate

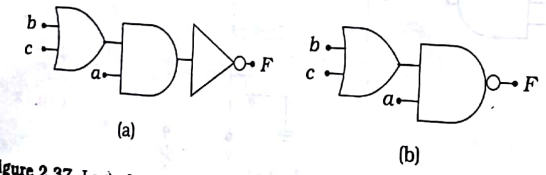


Figure 2.37 Logic function example

uses transistors. Since the gate count on a VLSI chip can easily exceed several hundred thousand, we often look for techniques that reduce the number of gates and/or FETs while still performing the required logic. In the present discussion, we will achieve this objective by building a single logic gate that implements the entire function.

Let us investigate the characteristics of the function F in more detail by applying DeMorgan expansions to the function to write

$$\begin{aligned} F &= \overline{a \cdot (b + c)} \\ &= \bar{a} + \overline{(b + c)} \\ &= [\bar{a} + (\bar{b} \cdot \bar{c})] \cdot 1 \end{aligned} \qquad (2.50)$$

The last step is simply ANDing the result with a logical 1. Expanding gives

$$F = \bar{a} \cdot 1 + (\bar{b} \cdot \bar{c}) \cdot 1 \qquad (2.51)$$

which is in a form that can be used to build the pFET switching circuit shown in Figure 2.38. The correspondence can be verified by checking each term. The first term implies a pFET connected between the power supply (V_{DD}) and the output that is controlled by the input a. The second term is identical in form to that encountered for the NOR2 gate. It represents two series-connected pFETs (with control variables b and c) that connect the power supply to the output.

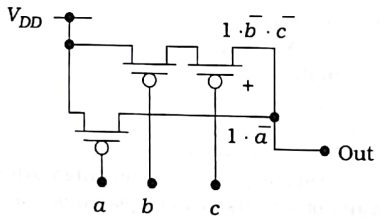


Figure 2.38 pFET circuit for F from equation (2.51)

The pFET circuit alone is not sufficient to create a functional electronic network. We must add an nFET array that gives $F = 0$ when necessary. The original form of the function in equation (2.49) shows that $F = 0$ occurs when

$$a = 1 \text{ AND } (b + c) = 1$$

This is equivalent to writing the output expression

$$0 \cdot [a \cdot (b + c)] \qquad (2.52)$$

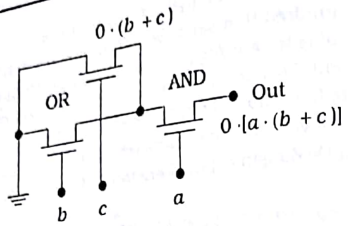


Figure 2.39 nFET logic circuit for F

which can in turn be used to describe the nFET array shown in Fig. 2.39. Two parallel-connected nFETs that are controlled by b and c g the OR operation. This group is in series with the a-input nFET to p duce the AND. The logic can be verified by the Karnaugh map groupi shown in Figure 2.40. Simplification to using a single a-input nF occurs because of the common term encompassed by the two groups.

The completed CMOS logic gate is built by combining the nFET a pFET circuits, and results in the circuit of Figure 2.41. We have rota the orientation of the FETs by 90 degrees to arrive at the finished sc matic. This is the most common way to draw CMOS logic circuits sinc makes series and parallel-connected FETs more obvious. The equivale of the circuit can be verified by tracing out each branch and comparin with the simpler circuits developed above.

This example illustrates that a complex function can be implemen with a single CMOS logic circuit that replaces a cascade made up of t or more primitive gates. Complex logic gate circuits can be more effici in VLSI design since they simplify the circuit requirements and the lo flows. One powerful aspect of a CMOS technology is that it allows us design logic networks using several different techniques, such as comp logic gates. This helps increase the **integration density**, which measu the amount of logic that can be placed on the silicon chip.

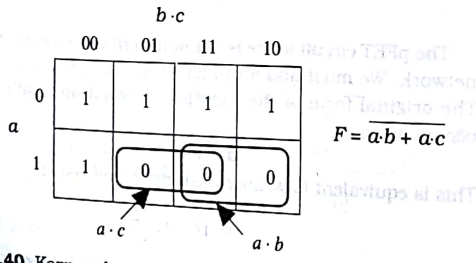


Figure 2.40 Karnaugh map grouping for the nFET circuit

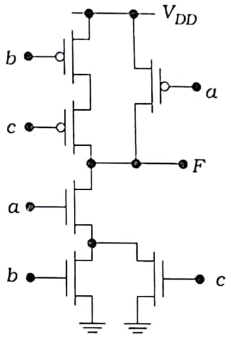


Figure 2.41 Finished complex CMOS logic gate circuit

2.4.1 Structured Logic Design

A structured approach to designing complex logic gates can be developed by focusing on the circuit characteristics. CMOS logic gates are intrinsically **inverting**; this means that the output always produces a NOT operation acting on the input variables. The simple inverter in Figure 2.42 illustrates the origin of this property. If the input a is a logic 1, then the nFET is ON and the pFET is OFF. The nFET passes the logic 0 (ground) to the output, giving $\bar{a}$ there. This characteristic was also observed in the NAND and NOR circuits.

The inverting nature of CMOS logic circuits allows us to construct logic circuits for AOI and OAI logic expressions using a structured approach. An AOI logic function is one that implements the operations in the order AND then OR then NOT (invert). For example,

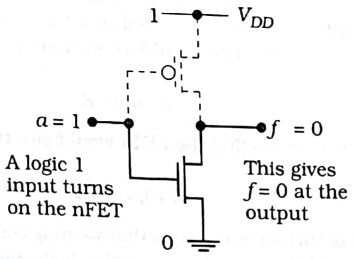


Figure 2.42 Origin of the inverting characteristic of CMOS gates

$$g(a, b, c, d) = \overline{a \cdot b + c \cdot d}$$

has an implied operational order of finding

(a AND b) and (c AND d)

and then perform the OR operation such that the final result is

g = NOT [(a AND b) OR (c AND d)]

Another example is found using the preceding CMOS gate examp
expanding the function to read

$$f(a, b, c) = \overline{a \cdot (b + c)} = \overline{a \cdot b + a \cdot c}$$

(2

The operational order A-O-I is seen after distributing the terms. An a
nate description of an AOI function is to say that it is an inverted sum
products (SOP). An OR-AND-INVERT (OAI) function reverses the ord
the AND and OR operations. An example of an OAI form is

$$h(x, y, z, w) = \overline{(x + y) \cdot (z + w)}$$

(2

since this implies that we first calculate

(x OR y) along with (w OR z)

and then

h = NOT [(x OR y) AND (w OR z)]

to evaluate the value of h. An OAI form is equivalent to an inverted p
uct-of-sums (POS) expression.

CMOS switching characteristics provide a natural means for im
menting inverting logic forms such as AOI and OAI. The techniqu
based on using nFET and pFET arrays in a consistent manner. Com
logic gates of this type allow the designer to compress three or more pr
itive operations into a single logic gate. Consider first the logic forma
properties of nFETs. From the NAND analysis, we learned that nFET
series provide AND-INVERT logic; this is shown in Figure 2.43(a). Si
larly, the NOR gate analysis showed us that parallel connected nF
produce the OR-INVERT operations as summarized in Figure 2.43
These results may be generalized to a larger number of transistors.
example, 4 series-connected nFETs with inputs a, b, c, d would produ

$$\overline{a \cdot b \cdot c \cdot d}$$

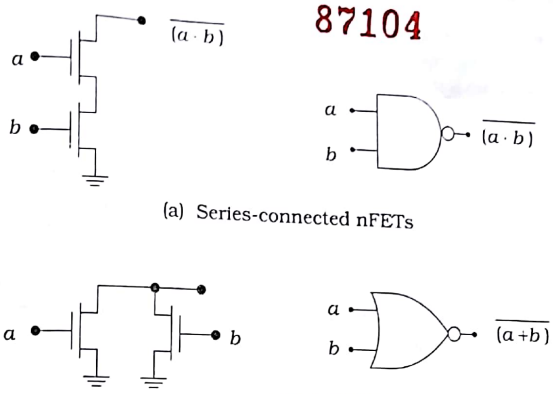
(2

while parallel-connecting the FETs would give the OR-INVERT operatio

$$\overline{a + b + c + d}$$

(2

The power of this observation is that we may combine series- and paral
connected nFETs to produce complex logic gates. An example of this
shown in Figure 2.44. This array consists of parallel-connected grou



(a) Series-connected nFETs

(b) Parallel-connected nFETs

Figure 2.43 nFET logic formation

with each group made of 2 series-connected nFETs. The transistors on the left side form the AND operation ($a \cdot b$) while the right group of nFETs yields ($c \cdot d$); the parallel connection of the two groups gives the OR operation, while the final output from the gate yields the NOT. We thus see that the function is described by

$$X = \overline{(a \cdot b) + (c \cdot d)} \qquad (2.58)$$

which is an AOI expression that is represented by the logic circuit shown next to the circuit. It is important to note that the NOT operation is viewed at the exit point of the logic (i.e., only for the function X). The AND operation is provided by series-connected nFETs, while the OR is accomplished by using a parallel-connected group. Although this approach is based on

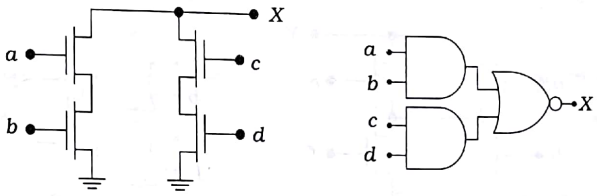


Figure 2.44 nFET AOI circuit

visually tracing the formation of the logic, we may use the formalism switching equations to verify the result. Applying the nFET equation gives the output as

$$0 \cdot [(a \cdot b) + (c \cdot d)] \qquad (2.$$

which is equivalent to the stated form for X.

Figure 2.45 illustrates a modified circuit. Comparing this with Fig 2.43 shows that a connection has been added such that now the up two transistors (with inputs a and e) are in parallel with one another. Si ilarly, the nFETs with inputs b and f are in parallel. Both parallel grou implement the OR operations giving the terms $(a + e)$ and $(b + f)$. Sen connecting the parallel groups gives the AND operation, so that inver the output results in

$$Y = \overline{(a+e)\cdot(b+f)} \qquad (2.$$

which has OAI form. To verify this result, use the switch-level equati to write

$$0 \cdot [(a+e)\cdot(b+f)] \qquad (2.$$

This is equivalent to the expression in equation (2.60) for Y.

Now recall that a CMOS logic gate uses nFETs to pass a 0 to the o put, and pFETs to pass a logic 1. Since pFETs complement nFETs, we c construct the logic formation characteristics summarized in Figure 2. The parallel-connected pFETs shown in Figure 2.46(a) are described the logic equation

$$1 \cdot (\bar{x} + \bar{y}) = 1 \cdot \overline{(x \cdot y)} \qquad (2.$$

which is the AND-NOT operation sequence. To obtain the OR-NOT ope tion, we must use series-connected pFETs as in Figure 2.46(b). In t case, the logic is formed from switching equations as

$$1 \cdot \bar{x} \cdot \bar{y} = 1 \cdot \overline{(x+y)} \qquad (2.$$

which verifies the statement.

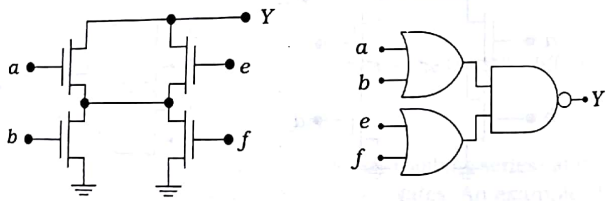
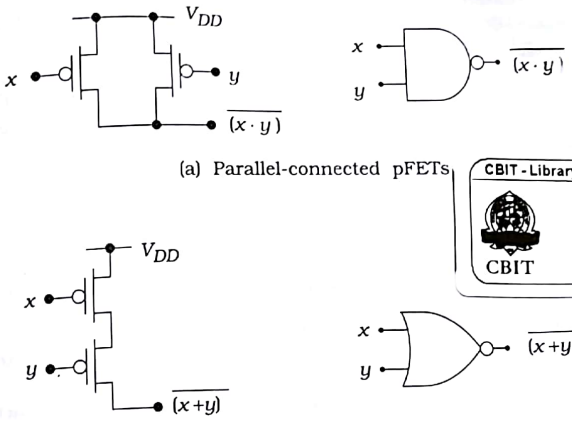


Figure 2.45 nFET OAI network



(a) Parallel-connected pFETs

(b) Series-connected pFETs

Figure 2.46 pFET logic formation

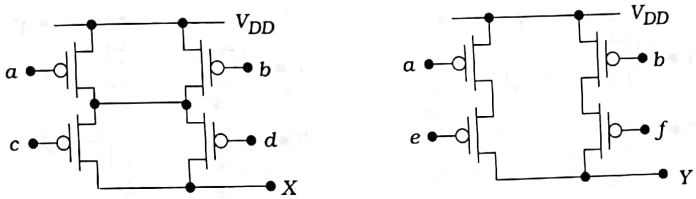
Let us examine the pFET array needed for the AOI function

$$X = \overline{(a \cdot b) + (c \cdot d)} \qquad (2.64)$$

discussed earlier for the nFET circuit shown in Figure 2.44. Using the pFET rules results in the network illustrated in Figure 2.47(a). Similarly, the OAI function

$$Y = \overline{(a+e)\cdot(b+f)} \qquad (2.65)$$

yields the pFET array in Figure 2.47(b).



(a) pFET AOI circuit

(b) pFET OAI circuit

Figure 2.47 pFET arrays for AOI and OAI gates

This discussion shows that nFET and pFET groups behave in differe ways. Parallel-connected nFETs yielded the OR-NOT operations while pa allel-connected pFETs give the AND-NOT sequence. Series-connecte nFETs provide AND-NOT, but series pFETs give us OR-NOT. We may u these results to state that equivalently wired groups of nFETs and pFE are logical **duals** of one another. In other words, if an nFET group yield function of the form

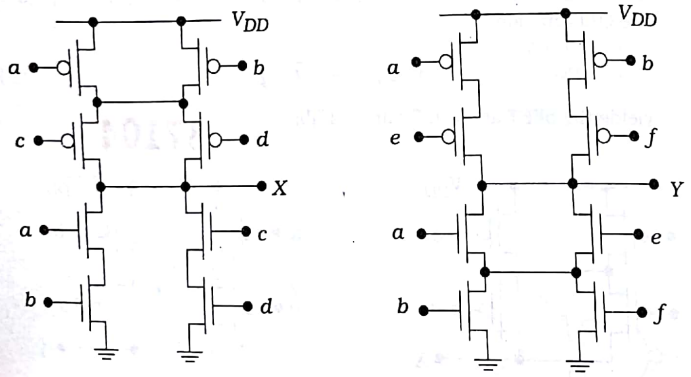
$$g = \overline{a \cdot (b + c)} \qquad (2.6$$

then an identically-wired pFET array gives the dual function

$$G = \overline{a + (b \cdot c)} \qquad (2.6$$

where the AND and OR operations have been interchanged. This is a interesting property of nFET-pFET logic that can be exploited in som CMOS designs.

The most important aspect of these examples is seen by constructi the complete CMOS circuit for each; both are shown in Figure 2.48. Co sider first the AOI circuit in Figure 2.48(a). The nFETs with inputs a and are in series, while the corresponding pFETs are wired in parallel. Thi scheme is also applied to the FETs with input variables c and d. Finall the nFET group with inputs (a, b) is in parallel with the input group (c, d so the corresponding pFET groups are in series. This is another exampl of series-parallel structuring of the nFET-pFET arrays. The OAI circuit Figure 2.48(b) exhibits the same features. In this case, the nFETs wit



(a) AOI circuit

(b) OAI circuit

e 2.48 Complete CMOS AOI and OAI circuits

inputs a and e are in parallel, as are the nFETs with inputs b and f. The pFET group with inputs a and e are wired in series; the same comment holds for the pFETs driven by b and f. Finally, since the nFET (a, e) group is in series with the (b, f) group, the corresponding pFETs groups are in parallel. This may be used to construct any AOI or OAI circuit in CMOS.

Example 2.1

Consider the complex function

$$X = \overline{a + b \cdot (c + d)} \qquad (2.68)$$

The nFET circuit can be constructed by using the following arrangements:

Group 1: nFETs with inputs c and d are in parallel;

Group 2: an nFET with input b is in series with Group 1;

Group 3: an nFET with input a is in parallel with the Group 1-Group 2 circuit.

The circuit in Figure 2.49 shows each group explicitly. The pFETs are arranged using series-parallel structuring. Each group of pFETs can be associated with the nFET group that has the same inputs such that

Group 1: pFETs with inputs c and d are in series;

Group 2: a pFET with input b is in parallel with Group 1 pFETs;

Group 3: a pFET with input a is in series with the Group 1-Group 2 pFETs.

The equivalent logic diagram for the circuit is shown in Figure 2.50.

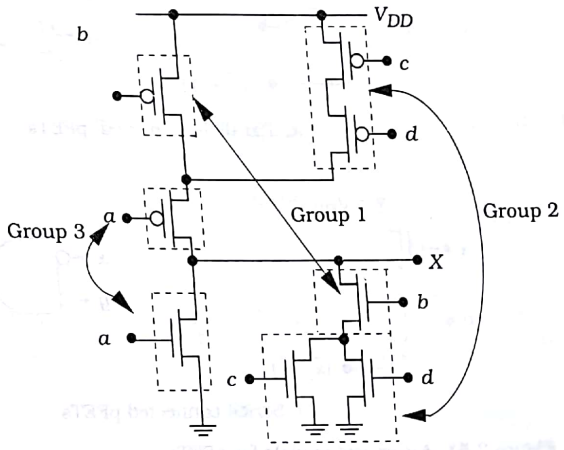


Figure 2.49 AOI circuit for Example 2.1

Tracing the data flow from the inputs to the output shows that the has OAOI structuring. This is just an AOI circuit with an additional input.

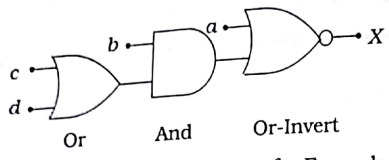
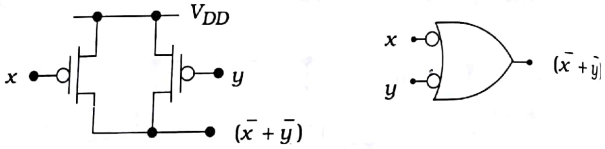


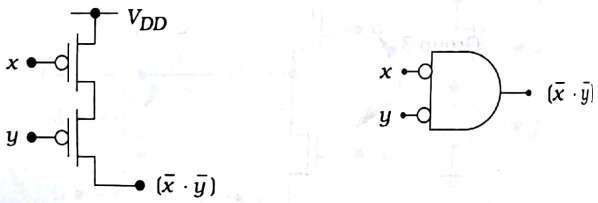
Figure 2.50 Equivalent logic diagram for Example 2.1

Bubble Pushing

Series-parallel wiring of complex CMOS logic circuits can be desig using an approach that is based on logic diagrams. The procedur obtained by applying the DeMorgan rules to the pFET relations illustra in Figure 2.46. Recall that pFETs can be modeled as assert-low switch Let us therefore model pFET groups as logic gates that have assert-l inputs. This leads to the modified logic associations shown in Figure 2. In Figure 2.51(a), we apply the DeMorgan rule to write



(a) Parallel-connected pFETs



(b) Series-connected pFETs

Figure 2.51 Assert-low models for pFETs

$$1 \cdot (\overline{x \cdot y}) = 1 \cdot (\bar{x} + \bar{y}) \tag{2.69}$$

so that parallel-connected pFETs may be viewed as an OR operation with assert-low (bubbled) inputs. In the same manner, the series-connected pFETs in Figure 2.51(b) provide the AND operation with assert-low inputs as verified by the identity

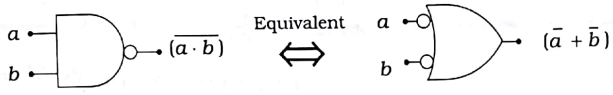
$$1 \cdot (\overline{x + y}) = 1 \cdot (\bar{x} \cdot \bar{y}) \tag{2.70}$$

Both operations can be represented graphically by the operations shown in Figure 2.52 where we visualize pushing the bubble backward through the gate to the inputs to create the dual operation with assert-low input ports.

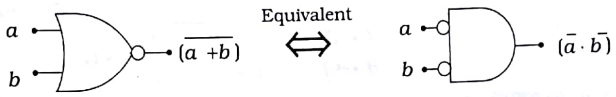
The procedure for designing the transistor circuitry for a CMOS logic gate can be summarized by the following steps.

- Construct the logic diagram using basic AOI or OAI structuring. Deeper nesting, such a OAOI and AOAI, is allowed.
- Use the gate-nFET relations summarized in Figure 2.43 to construct the nFET logic circuit between the output and ground.
- To obtain the topology of the pFET array, start with the original logic diagram and push the bubble back toward the inputs using the DeMorgan rules. Continue the backward pushing until every input is bubbled. The pFET circuitry between the output and VDD is then obtained using the rules in Figure 2.51.

Note that both the nFETs and the pFETs are wired such that parallel-connected transistors give the OR operation, while series-connected FETs provide the AND operation. The only difference between the two is that nFETs are assert-high devices while pFETs are assert-low (bubbled-input) switches.



(a) NAND - OR



(b) NOR - AND

Figure 2.52 Bubble pushing using DeMorgan rules

Example 2.2

Consider the logic diagram shown in Figure 2.53. This provides us wi
map for building the nFET logic array. We see that the nFETs with inp
a and b are in series (due to the AND gate), as are the nFETs with inp
c and d. These series-connected groups are in parallel with an nFET
has the input e since they are OR'ed at the output. The NOT operation
the NOR gate) is automatic in the nFET array.

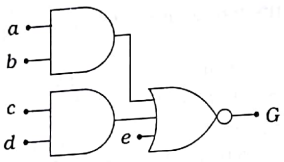
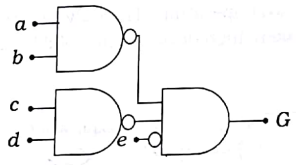
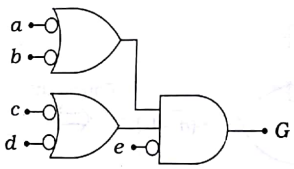


Figure 2.53 AOI logic diagram for bubble-pushing example

To obtain the wiring of the pFETs, we push the bubble back as sho
in Figure 2.54. The first step is to transform the output NOR gate into
AND gate with assert-low inputs; this results in the intermediate diagr
drawn in Figure 2.54(a). Pushing the bubbles back through the A
gates gives assert-low OR gates as in Figure 2.54(b). This shows that



(a) First transformation



(b) Final form

Figure 2.54 Bubble pushing to obtain the topology of the pFET array

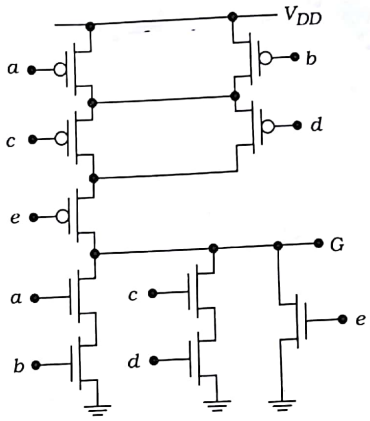


Figure 2.55 Final circuit for the bubble-pushing example

pFET array consists of

- Two pFETs with inputs a and b wired in parallel
- Two pFETs with inputs c and d wired in parallel
- One pFET with an input e that is in series with the two groups above.

The final circuit is drawn in Figure 2.55. It is worth the effort to trace through the construction procedure. And, it is important to remember that the CMOS logic gate implements the entire function G portrayed in the logic diagram. It is not possible to break the circuit down into more primitive logic.

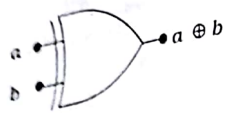
2.4.2 XOR and XNOR Gates

An important example of using an AOI circuit is constructing Exclusive-OR (XOR) and Exclusive-NOR circuits. These often-used gates are constructed from logic primitives. Figure 2.56 gives the circuit symbol and truth table for the XOR. Reading the logic 1 outputs gives the standard SOP equation

$$a \oplus b = \bar{a} \cdot b + a \cdot \bar{b} \tag{2.71}$$

from the second and third lines. This is not in AOI form. However, if we read the 0 output lines, then the XNOR expression is

$$\overline{a \oplus b} = a \cdot b + \bar{a} \cdot \bar{b} \tag{2.72}$$



a	b	a ⊕ b
0	0	0
0	1	1
1	0	1
1	1	0

Figure 2.56 Exclusive-OR (XOR) symbol and truth table

The XOR can thus be expressed as

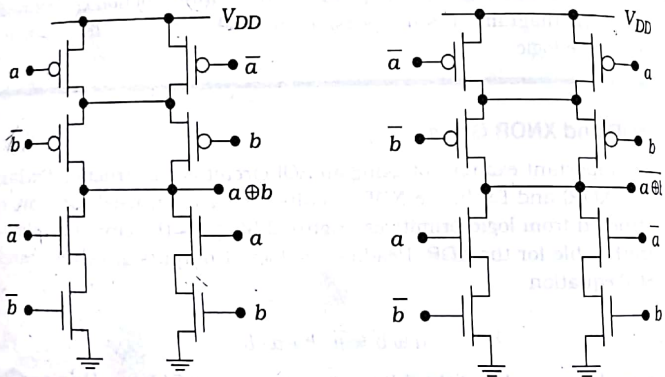
$$a \oplus b = \overline{(\overline{a \oplus b})} = \overline{a \cdot b + \bar{a} \cdot \bar{b}}$$

which has AOI structure. Using the circuit in Figure 2.48(a) gives basic AOI XOR circuit shown in Figure 2.57(a). Since the XOR gate inputs of (a, b) only, two inverters are needed to provide the 4-inpu $(a, b, \bar{a}, \bar{b})$ in this circuit.

To obtain an XNOR circuit, we just complement the XOR SOP equ to write

$$\overline{a \oplus b} = \overline{\bar{a} \cdot b + a \cdot \bar{b}}$$

Interchanging a and $\bar{a}$ in the XOR circuit thus gives the XNOR gate in ure 2.57(b). Switching the b and $\bar{b}$ variables would have given the result.

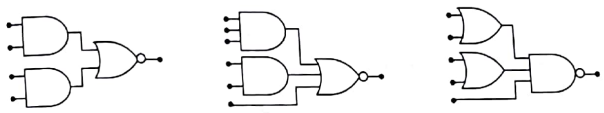


(a) Exclusive-OR (b) Exclusive-NOR

ure 2.57 AOI XOR and XNOR gates

.4.3 Generalized AOI and OAI Logic Gates

Standard logic design is often simplified using generalized multiple-input AOI and OAI logic gates. This is particularly true in ASIC-type circuits that rely on predesigned logic circuits. A straightforward nomenclature for distinguishing among various input configurations is developed in Figure 2.58. The network in Figure 2.58(a) has an AOI pattern with 2 inputs to each AND gate; it is therefore called an AOI22 gate. Similarly, the logic pattern in Figure 2.58(b) is called an AOI 321, where a "1" label implies an input that bypasses the AND gates and is connected directly to an OR gate. The third example, shown in Figure 2.58(c), is termed an OAI221 gate using the same convention. The CMOS circuits are easily designed using series-parallel wiring or bubble pushing.



(a) AOI22 gate (b) AOI321 gate (c) OAI221 gate

Figure 2.58 General naming convention

Generalized complex logic gates provide a uniform basis for creating different logic operations using a generic gate. As a simple example, consider the AOI22 gate shown in Figure 2.59(a). This provides an output of

$$\text{AOI22}(a, b, c, d) = \overline{a \cdot b + c \cdot d} \tag{2.75}$$

To create an XOR circuit, we can define the inputs as shown in Figure 2.59(b), which allows us to write

$$a \oplus b = \text{AOI22}(a, b, \bar{a}, \bar{b}) \tag{2.76}$$

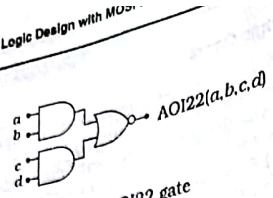
Using the same reasoning, the XNOR function can be obtained using

$$\overline{a \oplus b} = \text{AOI22}(a, \bar{b}, \bar{a}, b) \tag{2.77}$$

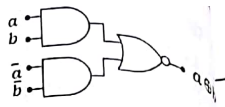
This illustrates how generic logic gates can be used in random logic design.

2.5 Transmission Gate Circuits

A CMOS **transmission gate** is created by connecting an nFET and pFET in parallel as shown in Figure 2.60(a). The nFET Mn is controlled by the signal s, while the pFET Mp is controlled by the complement $\bar{s}$. When wired in this manner, the pair acts as a good electrical switch between the



(a) General AOI22 gate



(b) XOR

Figure 2.59 Application of an AOI22 gate

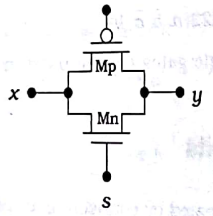
input and the output variables x and y, respectively.

The operation of the switch can be understood by analyzing the cases for s. If $s = 0$, the nFET is OFF; since $\bar{s} = 1$, the pFET is also OFF that the TG acts as an open switch. In this case, there is no relation between x and y. For the opposite case where $s = 1$ and $\bar{s} = 0$, both are on, and the TG provides a good conducting path between x and y. ically, this is identical to the switching of an nFET so that we may wri

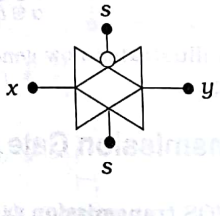
$$y = x \cdot s \quad \text{iff} \quad s = 1 \tag{2}$$

This assumes that x is the input and y is the output. However, the TG classified as a **bi-directional** switch. The TG symbol in Figure 2.60b based on this observation. It is created using two back-to-back arr indicating that the data can flow in either direction. Control is achieved s and $\bar{s}$; the bubble indicates the connection to the pFET gate.

Transmission gates are useful because they can transmit the en voltage range $[0, V_{DD}]$ from left to right (or vice versa). This is due to parallel connection of the transistors. Zero voltage levels are transmit by the nFET, while the pFET is responsible for transmitting the pos supply voltage V_{DD}. The main drawback of using TGs in modern VLS that they require two FETs and an implied inverter that takes s and p duces $\bar{s}$.



(a) Transistor circuit



(b) Symbol

Figure 2.60 Transmission gate (TG)

2.5.1 Logic Design

Transmission gate logic design has been used extensively in CMOS design for many years. The simplicity of the switching and the ability to transmit the entire range of voltages made it attractive for many applications. TG circuits are found in many ASIC structures, making them worth studying in more detail.

Multiplexors

The ideal-switch characteristics of TGs make them useful for creating some rather unique circuits. An example is the 2-to-1 MUX shown in Figure 2.61. The operation of the circuit is summarized in the table. When the selector signal has a value s = 0, TG0 is closed and TG1 is open, so that P_0 is transmitted to the output. If s = 1, the situation is reversed with TG0 open and TG1 closed; in this case, $F = P_1$. Combining these results gives

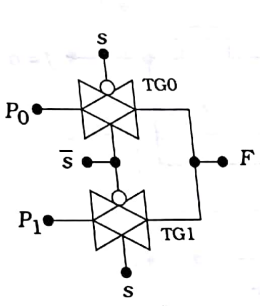
$$F = P_0 \cdot \bar{s} + P_1 \cdot s \tag{2.79}$$

which is the required equation. Note that the use of a pair of TGs eliminates the possibility of having a floating (disconnected) output since one TG is always closed while the other will be open. The 2-to-1 architecture can be extended to a 4:1 network by using the 2-bit selector word $(s_1\ s_0)$ that has values of (0 0), (0 1), (1 0), and (1 1). Each input line (P_0, P_1, P_2, P_3) will have two TGs in its path such that the output is

$$F = P_0 \cdot \bar{s}_1 \cdot \bar{s}_0 + P_1 \cdot \bar{s}_1 \cdot s_0 + P_2 \cdot s_1 \cdot \bar{s}_0 + P_3 \cdot s_1 \cdot s_0 \tag{2.80}$$

For example, the P_0 path will have TGs that are closed with $(s_1\ s_0) = (0\ 0)$. The construction of the network is left as an exercise for the reader.

The 2:1 MUX can be modified to produce other useful functions. One illustrated in Figure 2.62(a). The input to the top TG is a; this is inverte so that $\bar{a}$ enters the lower TG. Variable b and its complement are used control the TGs. When $b = 0$, the upper TG is closed and a is passed to t



s	TG0	TG1	F
0	Closed	Open	P_0
1	Open	Closed	P_1

Figure 2.61 A TG-based 2-to-1 multiplexor

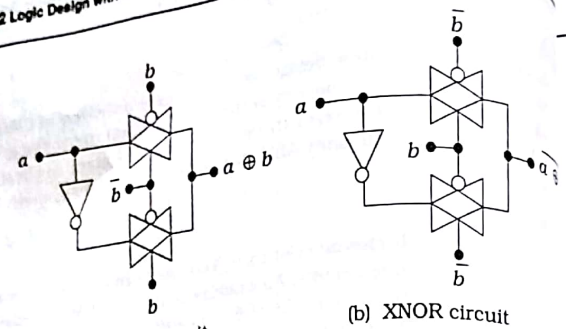


Figure 2.62 TG-based exclusive-OR and exclusive-NOR circuits

output, while $b = 1$ closes the lower TG and steers $\bar{a}$ to the output gives

$$a \cdot \bar{b} + \bar{a} \cdot b = a \oplus b$$

i.e., the circuit provides the XOR (exclusive-OR) function. The expression can be verified using the 2:1 MUX result. An XNOR function

$$\overline{a \oplus b} = a \cdot b + \bar{a} \cdot \bar{b}$$

is obtained if we interchange b and $\bar{b}$. The circuit for this simple modification is shown in Figure 2.62(b).

OR Gate

Transmission gate characteristics can be used to create the simple circuit shown in Figure 2.63; this is useful since complementary CMOS gates can only provide the NOR operation. The operation of the circuit

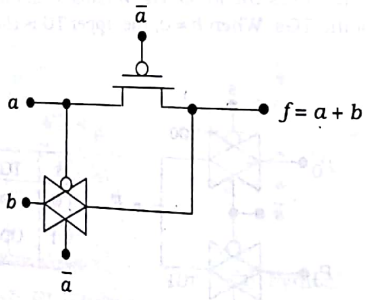


Figure 2.63 A TG-based OR gate

be understood by examining the effect that a has on the switches. If $a = 0$, then the pFET is OFF (since $\bar{a} = 1$ drives it into cutoff) while the TG acts as a closed switch. This gives an output of $f = b$. If $a = 1$, then the pFET is ON and the value of $f = a = 1$ is transmitted to the output. Thus, the output is $f = 1$ if either input is a 1, which establishes the OR operation. We can alternately use logic equations for the TG and the pFET to write the output as

$$\begin{aligned} f &= a \cdot (\overline{\bar{a}}) + \bar{a} \cdot b \\ &= a + \bar{a} \cdot b \\ &= a + b \end{aligned} \tag{2.83}$$

where the last step follows by absorption. This verifies the simpler bit-by-bit analysis.

Alternate XOR/XNOR Circuits

Mixing TGs and FETs as in the OR gate circuit gives rise to many variations for the design of basic logic gates. Many of these designs are for exclusive-OR and equivalence (XNOR) functions due to their importance in adders and error detection/correction algorithms.

An example of this type of circuit is the XNOR network in Figure 2.64. This uses the input pair $(b, \bar{b})$ to control the transmission gate. To understand the operation, remember that the output of an XNOR gate is 1 if and only if the inputs are equal. Suppose that $b = 1$; the TG acts as a closed switch and a is transmitted to the output to give $g = a$. For this case, the output is a 1 iff $a = 1$. The circuit operates differently if $b = 0$. Now, the TG is off and a is directed toward the gates of the Mp/Mn pair. Since $b = 0$ is applied to the source of the nFET Mn and $\bar{b} = 1$ is connected to the source (upper side) of the pFET, the $(b, \bar{b}) = (0, V_{DD})$ pair provides power to the FETs, resulting in an inverter! For this case, the output is $g = \bar{a}$, so that g is 1 iff $a = 0$. This establishes the circuit as an XNOR gate as stated. Interchanging b and $\bar{b}$ gives an XOR gate.

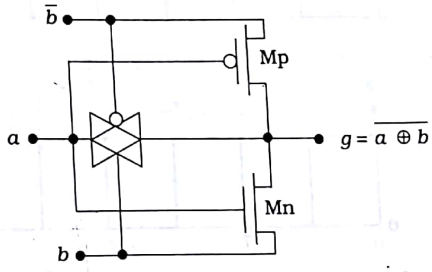


Figure 2.64 An XNOR gate that uses both TGs and FETs

2.6 Clocking and Dataflow Control

Synchronous digital design relies on the ability to control the flow o using a clocking signal ϕ. The switching characteristics of TGs c used to provide a simple approach to system clocking. Since compl tary signals are required to switch a TG, both ϕ and $\bar{\phi}$ are used in thi of design; waveforms are shown in Figure 2.65. The period T is the le seconds needed for one complete cycle. The frequency f is defined by

$$f = \frac{1}{T}$$

and has units of Hertz [Hz] = [1/sec], where 1 hertz means one cy completed in 1 second. We will assume that the clock is at a logic 1 of one-half of the period, and at a logic 0 value for the remaining h the period.

Let us examine the effect of applying the complementary clocks transmission gate. Figure 2.66(a) shows that when a value of ϕ applied to the nFET and $\bar{\phi} = 0$ to the pFET, the TG is On and acts closed switch. Reversing these values as in Figure 2.66(b) gives an switch. Under static conditions, the value of y would not be known the switch is opened. However, the electrical characteristics of allow us to temporarily hold the value of $y = x$ for a very short time typically, t_{hold} is less than 1 second. If we use a high-frequency clock the periodic open-closed change occurs at every half clock cycle. The can hold the previous value so long as $(T/2) < t_{hold}$. This provide accurate time base for controlling data flow in a complex network.

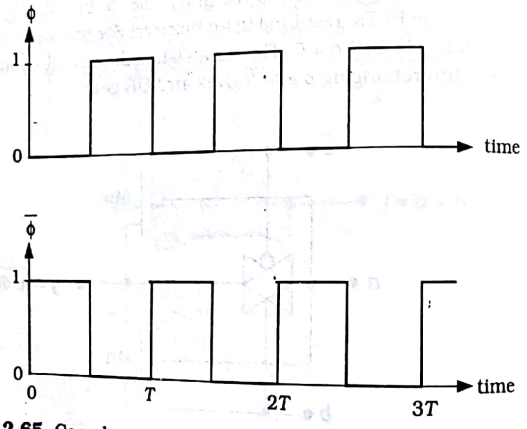
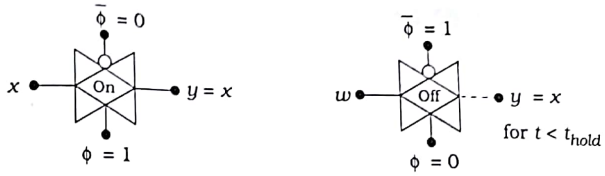


Figure 2.65 Complementary clocking signals



(a) Closed switch (b) Open switch

Figure 2.66 Behavior of a clocked TG

To use clocked TGs for data flow control, we place oppositely phased TGs at the inputs and outputs of logic blocks. A gate-level example is shown in Figure 2.67. The inputs on the left side are admitted when the clock is high with $\phi = 1$; the first group of logic gates evaluates the input bits and produce outputs f and g during this time. Since the output TGs are off, the outputs are held until the clock changes to $\phi = 0$. When this happens, f and g are allowed to enter the next group of logic gates, which results in F, G, and H. These are held at the outputs until they are transferred out when the clock returns to the value $\phi = 1$. This shows how data flow through the system is synchronized by the clocked TG.

Data flow can be visualized using system level block timing diagrams as in Figure 2.68. Each clock plane is shown graphically by a dashed line with either ϕ or $\bar{\phi}$ next to it. These represent a clock-controlled TG at every input. When the variable is true (equal to 1), then data is allowed to pass the plane from one side to the other. Otherwise the data is held on the left side until a clock transition takes place. With the labeling shown, this

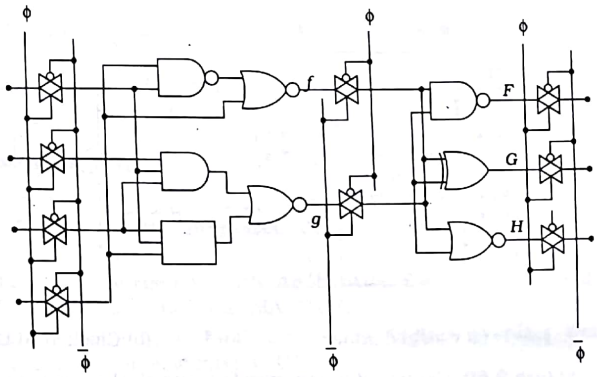


Figure 2.67 Data synchronization using transmission gates

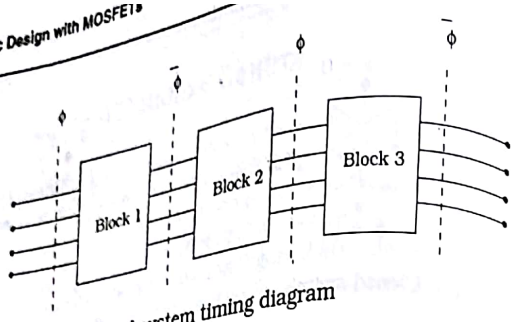
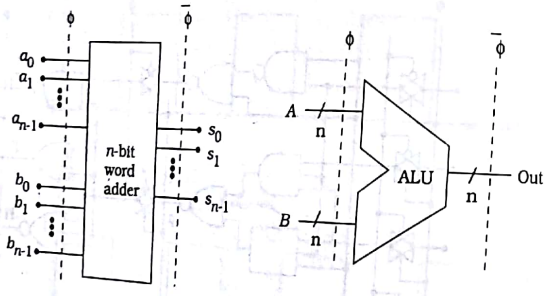


Figure 2.68 Block-level system timing diagram

says that a clock of $\phi = 1$ allows inputs into logic block 1. The outputs transferred to logic block 2 when the clock changes to $\phi = 0$ and so on this scheme, data moves through a logic block every half cycle. Since logic blocks are arbitrary, it can be used as the basis for building complex logic chains. It also allows us to synchronize the operations formed on each bit of an n-bit binary word.

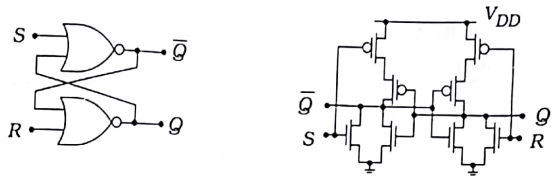
A synchronized word adder is illustrated in Figure 2.69(a). The in words $a_{n-1}...a_0$ and $b_{n-1}...b_0$ are controlled by the ϕ-clock plane, while sum $s_{n-1}...s_0$ is transferred to the output when $\phi = 0$. Every bit in a w is transmitted from one point to another at the same time, which allo us to track the data flow through the system. This is extended to a lar scale with the ALU (arithmetic and logic unit) example in Figure 2.69 Inputs A and B are "gated" into the ALU by the ϕ -plane control; the res word Out is transferred to the next stage when $\bar{\phi} = 1$, i.e., $\phi = 0$. This illu trates the power of using clocked data transfer in VLSI design.

Clocked transmission gates synchronize the flow of signals, but t lines themselves cannot store the values for times longer than t_{hold}, whi is very small. A storage element such as a latch is needed to obtain lon



(a) Clocked adder (b) Clocked ALU

Figure 2.69 Control of binary words using clocking planes

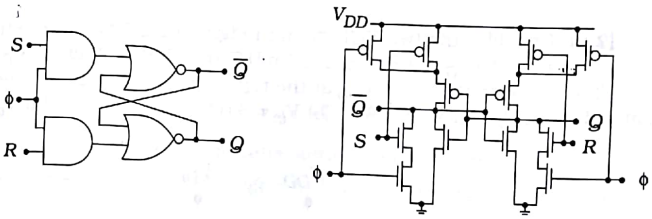


(a) Logic diagram (b) CMOS circuit

Figure 2.70 SR latch

term storage of a data bit. Figure 2.70(a) shows the logic diagram for a simple NOR-based SR-latch. The CMOS circuit in Figure 2.70(b) is obtained by wiring two NOR2 gates together.

Clock control can be added to the circuit by inserting AND gates at the inputs to arrive at the modified logic diagram in Figure 2.71(a). This only allows changes in the inputs when $\phi = 1$. A compact CMOS circuit can be obtained by observing that two identical CMOS AOI circuits can be used to create the circuit in Figure 2.71(b). Designing a CMOS circuit using a logic diagram as a starting point thus becomes a straightforward process. This makes CMOS easy to adapt to. The challenge arises in making the circuits as fast and as compact as possible.



(a) Logic diagram (b) CMOS circuit

Figure 2.71 Clocked SR latch

2.7 Further Reading

[1] Ken Martin, **Digital Integrated Circuit Design**, Oxford University Press, New York, 2000.

[2] Michael John Sebastian Smith, **Application Specific Integrated Circuits**, Addison-Wesley, Reading, MA, 1997.

[3] John P. Uyemura, **A First Co... in Digit... tems Design**, Brooks-Cole Publishers, Monterey

[4] John P. Uyemura, **CMOS Logic Circuit Design**, Kluwer Acad
Publishers, Norwell, MA, 1999.
[5] M. Michael Vai, **VLSI Design**, CRC Press, Boca Raton, FL, 2001.
[6] Neil H. E. Weste and Kamran Eshraghian, **Principles of CMOS**
Design, 2nd ed., Addison-Wesley, Reading, MA, 1993.
[7] Wayne Wolf, **Modern VLSI Design**, 2nd ed., Prentice-Hall PTR, U
Saddle River, NJ, 1998.

2.8 Problems

[2.1] Suppose that $V_{DD} = 5$ V and $V_{Tn} = 0.7$ V. Find the output vol
V_{out} of the nFET in Figure P2.1 for the following input voltage values
$V_{in} = 2$ V; (b) $V_{in} = 4.5$ V; (c) $V_{in} = 3.5$ V; (d) $V_{in} = 0.7$ V.

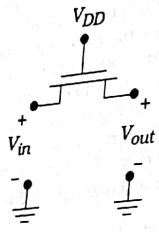


Figure P2.1

[2.2] Consider the two-FET chain in Figure P2.2. The power supply is
to a value of $V_{DD} = 3.3$ V and the nFET threshold voltage is $V_{Tn} = 0.5$
Find the output voltage V_{DD} at the right side of the chain for the follow
values of V_{in}: (a) $V_{in} = 2.9$ V; (b) $V_{in} = 3.0$ V; (c) $V_{in} = 1.4$ V; (d) $V_{in} = 3.1$

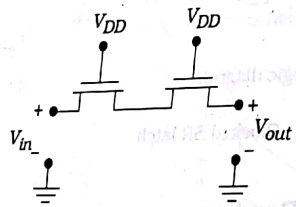


Figure P2.2

[2.3] The output of an nFET is used to drive the gate of another nFET
shown in Figure P2.3. Assume that $V_{DD} = 3.3$ V and $V_{Tn} = 0.60$ V. Find
output voltage V_{out} when the input voltages are at the following values:
(a) $V_a = 3.3$ V and $V_b = 3.3$ V; (b) $V_a = 0.5$ V and $V_b = 3.0$ V; (c) $V_a = 2.0$
and $V_b = 2.5$ V; (d) $V_a = 3.3$ V and $V_b = 1.8$ V.

[2.4] Design a NAND3 gate using an 8:1 MUX.

[2.5] Design a NOR3 gate using an 8:1 MUX as a basis.

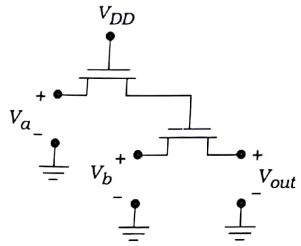


Figure P2.3

[2.6] Consider the 2-input XOR function $a \oplus b$.
(a) Design an XOR gate using a 4:1 MUX.
(b) Modify the circuit in (a) to produce a 2-input XNOR.
(c) A full adder accepts inputs a, b, and c and calculates the sum bit

$$s = a \oplus b \oplus c \tag{2.85}$$

Use your MUX-based gates to design a circuit with this output.

[2.7] Design a CMOS logic gate for the function

$$f = \overline{a \cdot b + a \cdot c + b \cdot d} \tag{2.86}$$

using the smallest number of transistors.

[2.8] Design a CMOS circuit for the OAI expression

$$h = \overline{(a+b) \cdot (a+c) \cdot (b+d)} \tag{2.87}$$

Use the smallest number of transistors in your design.

[2.9] Construct the CMOS logic gate for the function

$$g = \overline{x \cdot (y+z) + y} \tag{2.88}$$

Start with the minimum-transistor nFET network, and then apply bubble pushing to find the pFET wiring.

[2.10] Design a CMOS logic gate circuit that implements

$$F = \overline{a+b \cdot c + a \cdot b \cdot c} \tag{2.89}$$

using series-parallel logic. The objective is to minimize the transistor count.

[2.11] Consider the logic described by the diagram in Figure P2.4. A single, complex logic CMOS gate is to be designed for F.

(a) Construct the nFET array using the logic diagram.

(b) Apply bubble pushing to obtain the pFET logic. Then construct the pFET array using the rules.

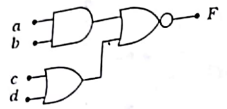


Figure P2.4

[2.12] An AOAI logic gate is described by the schematic in Figure P2.
(a) Construct the nFET array using the logic diagram.
(b) Apply bubble pushing to obtain the pFET logic. Use the diagra
construct the pFET array using the pFET rules.

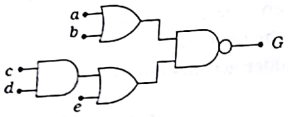


Figure P2.5

[2.13] A pFET logic array is shown in Figure P2.6. Construct the l
diagram using the pFET logic equations. Then construct the nFET cir

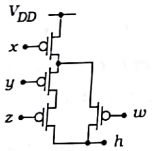


Figure P2.6

[2.14] Design the 4:1 multiplexor circuit that implements the functio
equation (2.80) by using TG switches.

[2.15] Use an AOI22 gate to design a 2:1 MUX. Inverters are permitte
your design.

[2.16] Design a 4:1 MUX using three 2:1 TG multiplexors.

[2.17] A CPU clock ϕ has a frequency 2.1 GHz. What is the period T?

[2.18] Suppose that the hold time for a TG is given as t_{hold} = 120 millis
onds (ms). What is the smallest clock frequency that can be used to cl
the data flow using a scheme such as that shown in Figure 2.67?

Physical Structure of CMOS Integrated Circuits

3

CMOS integrated circuits are electronic switching networks that are created on small area of a silicon wafer using a complex set of physical and chemical processes. A primary task of the VLSI designer is to translate circuit schematics into silicon form. This process is called **physical design** and is one aspect that separates the field of VLSI from general digital engineering. In this chapter we will examine the structure of a CMOS integrated circuit as seen at the microscopic silicon level in the design hierarchy.

3.1 Integrated Circuit Layers

A silicon integrated circuit can be viewed as a collection of patterned material layers, with each layer having specific conduction properties. The layers may be **metals** that conduct current very well, or they may be **insulators** that block the flow of current. Another material used to create layers is the element **silicon**. It is classified as a **semiconductor**, which means that it is a "partial" conductor. We sometimes refer to both metals and silicon as "conductors," but it is important to distinguish between the two.

An integrated circuit is made by stacking different layers of materials in a specific order to form three-dimensional structures that collectively act as an electronic switching network. Each layer has a predefined pattern that is specified in the system design process. The idea can be understood by referring to Figure 3.1, which portrays two separated layers. The bottom layer is a "sheet" of insulator on a base material called the "substrate." Above this is a patterned material layer of metal that is labeled "Layer M1." The pattern consists of two parallel **lines** of material which are to be placed on top of the insulator in the positions indicated by the

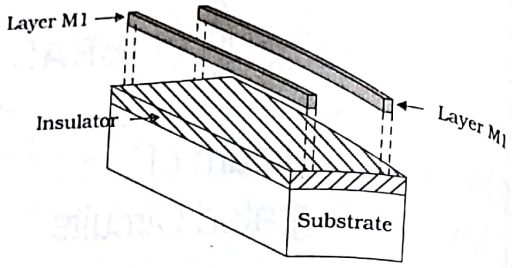
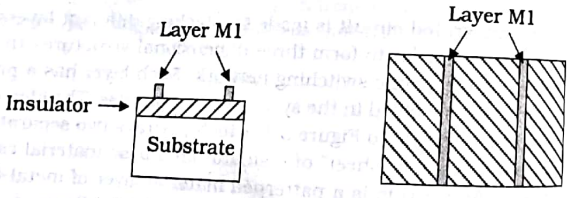


Figure 3.1 Two separate material layers

dashed lines. Figure 3.2 shows the structure after the stacking ope is finished. The end view is illustrated in Figure 3.2(a) and shows th lines of Layer M1 on top of the insulating layer. Figure 3.2(b) provide top view which shows that the two lines are parallel. The insula shown explicitly in this drawing, but we often take its existence as implied and omit it from top views. Physically this is acceptable bec the insulator itself is usually a layer of **silicon dioxide** (SiO_2), whi generically known as **quartz glass** and is visually transparent. Alth quite simple, this example provides one main feature of layering in con integrated circuit (IC): patterned conducting layers on top of a insulator. Complex VLSI chips employ several conducting layers of al num or copper with this type of structuring.

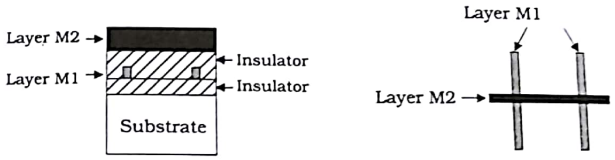
The concepts introduced in the preceding drawings can be extende adding more layers. Suppose that we want to place another metal pat on top of the structure shown in Figure 3.2. First we coat the surface another layer of insulating glass to keep it from coming in contact Layer M1, and then subject it to a **chemical-mechanical planariza** (CMP) sequence. In CMP, the surface is etched and "sanded" to prov



(a) Side view (b) Top view

Figure 3.2 Layers after the stacking process is completed

flat surface for the next layer. Next, we coat the surface with the second metal (Layer M2) to arrive at the structure drawn in Figure 3.3. The side view in Figure 3.3(a) shows the added insulator that covers Layer M1 and the second metal Layer M2 on top. This illustrates the stacking order of the various layers but does not show that the two metal layers have different patterns. The top view in Figure 3.3(b) provides the distinguishing features of the patterning. In particular, we see that the Layer M2 pattern is a single metal line that is perpendicular to the parallel lines of Layer M1. The line has been drawn so that it covers Layer M1 patterns when the two cross. Also note that we have not shown any of the insulating layers explicitly in the top view, but it is important to remember that the two do not touch.



(a) Side view (b) Top view

Figure 3.3 Addition of another insulator and a second metal layer

Combining the top and end views of an integrated circuit allows us to visualize the three-dimensional structure. Some important points that arise from this example are

- The side view illustrates the order of the stacking
- Insulating layers separate the two metal layers so that they are electrically distinct
- The patterning of each layer is shown by a top view perspective

The stacking order is established in the manufacturing process and cannot be altered by the VLSI designer. However, creating the pattern for each layer is a critical part of the chip design sequence as it defines the locations and sizes of all MOSFETs and specifies how the transistors are connected together.

3.1.1 Interconnect Resistance and Capacitance

Logic gates communicate with each other by signal flow paths from one point to another. At the integrated circuit level, this is accomplished by using patterned metal lines as wires to conduct electrical currents. These lines are generically referred to as **interconnects**. While this seems like a straightforward translation, the level of electric current flow is governed

by the physical characteristics of the material and the dimensions line. This implies that the signal transfer speed is directly affected physical implementation of the wiring, making it a very important of chip design.

Applying a voltage V (in units of volts V) to a patterned metal li ates a flow of current I (in amperes A) through it. For a simple con such as a metal, the relationship between the voltage and the cur given by Ohm's law

$$V = IR$$

where R is a constant of proportionality called the **resistance**. The resistance is the **Ohm** and is denoted by the Greek uppercase omega: Ω. It has fundamental unit of volts/amperes. Ohm's law only for simple devices that are called **resistors** in electronics. The s used for a resistor is shown in Figure 3.4. The jagged line is meant t cate that the device impedes the flow of electrical current. The sym used only for a "linear" resistor where the voltage is proportional current.

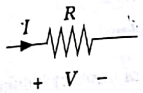


Figure 3.4 Symbol for a linear resistor

Now recall that a Boolean variable x is represented by a voltage V_x CMOS circuit. When applied to a patterned metal line, this voltage ca a current I_x to flow; the actual value of I_x is determined by the **line r tance** R_{line} and a few other electrical parameters. R_{line} is measure units of ohms and is classified as a **parasitic** (unwanted) electrical ment that cannot be avoided. Resistance impedes the flow of elect signals, so that the value of R_{line} should be kept as small as possible.

The value of R_{line} for a given line can be calculated using the geom shown in Figure 3.5. The length of the line is denoted by l and is m sured in units of centimeters [cm]. The cross-sectional area A (with u of cm^2) is the product of the width w and the thickness t of the layer.

$$A = wt$$

The **conductivity** σ shown in the drawing is a characteristic of the m rial used in the layer. It is measured in units of $[\Omega\text{-cm}]^{-1}$ and repres how easily current flows: a large value of σ means that the layer condu very well. Metals have large conductivities, while insulators have small values of σ; we assume that the numerical value of σ is known.

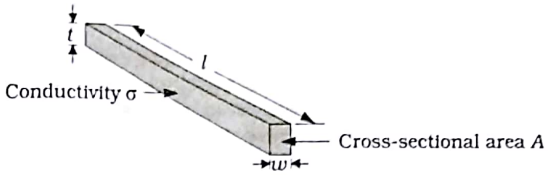


Figure 3.5 Geometry of a conducting line

these parameters, the line resistance in units of ohms is calculated from the expression

$$R_{line} = \frac{l}{\sigma A} \tag{3.3}$$

This shows the important relations that R_{line} is proportional to the length l of the line, and inversely proportional to the cross-sectional area A.

The **resistivity** ρ is the inverse of the conductivity such that

$$\rho = \frac{1}{\sigma} \tag{3.4}$$

ρ has units of [Ω-cm]. A high resistivity implies a low conductivity. The formula for the line resistance becomes

$$R_{line} = \rho \frac{l}{A} \tag{3.5}$$

by simple substitution.

A VLSI designer cannot control the values of t or σ, as these are established by the manufacturing process. Because of this, it is useful to rewrite the equation in the form

$$R_{line} = \left(\frac{1}{\sigma t}\right)\left(\frac{l}{w}\right) \tag{3.6}$$

so that the process-related terms are grouped together. This can be used to define the **sheet resistance** R_s of the line as

$$R_s = \frac{1}{\sigma t} = \frac{\rho}{t} \tag{3.7}$$

It is easily verified that R_s has units of ohms [Ω]. The sheet resistance of a layer is very useful because of two reasons. First, we find that it can be directly measured in the laboratory without knowing the actual values of σ or t. The second reason is due to the observation that a line with a length of $l = w$ has a line resistance of

$$R_{line} = R_s\left(\frac{w}{w}\right) = R_s \qquad (3$$

In other words, R_s represents the resistance of a square region with dimensions ($w \times w$). Because of this R_s is sometimes given units of "Ω square." This interpretation of the sheet resistance can be used to deri simple technique for calculating the value of the line resistance.

Consider the top-view geometry illustrated in Figure 3.6(a), whi identifies one square. By definition, the square has an end-to-end re tance of R_s. The square can be used to construct the equivalent line ill trated in Figure 3.6(b) by stringing many squares in a linear fashion. calculate the total end-to-end resistance R_{line}, we note that each squ has a resistance of R_s and that a string of resistors in series is equival to a single resistor with a value equal to the sum of the individual re tances. If there are a total of n squares, then we may write that

$$R_{line} = R_s n \qquad (3$$

where

$$n = \frac{l}{w} \qquad (3.$$

gives the total number of squares from one end to the other. Note that n not restricted to integer values; fractional contributions are permitted shown on the right-hand side of the line.

This analysis demonstrates that, for a given layer, the line resistan depends upon the ratio (l/w) of the patterned line. The importance of th result is based on the qualitative observation that the speed of a transm ted signal along a patterned line is affected by the value of R_{line}. A sm

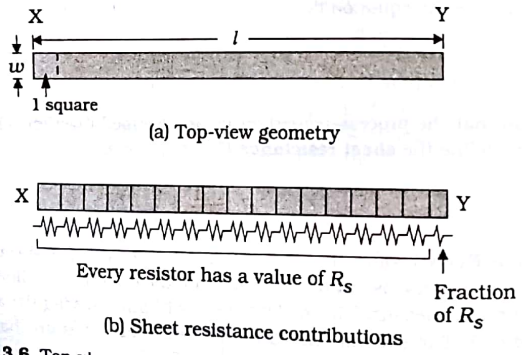


Figure 3.6 Top-view geometry of a patterned line

value of R_{line} allows for a high level of current flow, and is desirable for high-speed designs. We will quantify these statements later.

Interconnect lines also exhibit the property of **capacitance**, which is the ability to store electric charge and energy. In electronics, the element that stores charge is called a **capacitor**, and has the circuit symbol shown in Figure 3.7. It is characterized by the capacitance value C such that the charge Q on the positive side of the device is given by

$$Q = CV \qquad (3.11)$$

where V is the voltage; this is balanced by a negative charge $-Q$ on the other plate. The unit of capacitance is the **farad** [F] where 1 F is defined as 1 coulomb/volt. Since electric current is defined by the time derivative $I = (dQ/dt)$, differentiating gives the I-V equation

$$I = C\frac{dV}{dt} \qquad (3.12)$$

for the device.

Capacitance exists between any two conducting bodies that are electrically separated. For the interconnect line, the conductor is isolated from the semiconductor substrate by an insulating layer of silicon dioxide glass. The capacitance depends on the geometry of the line. Consider the structure shown in Figure 3.8 where T_{ox} is the thickness of the oxide between the interconnect line and the substrate in units of cm. Using basic physics, the line capacitance is given by the *parallel-plate* formula

$$C_{line} = \frac{\varepsilon_{ox} wl}{T_{ox}} \qquad (3.13)$$

and is measured in units of farads. In this equation, wl is the area of the interconnect in cm^2 as seen from the top. The parameter ε_{ox} is the permittivity of the insulating oxide with units of F/cm; ε_{ox} is determined by the composition of the oxide.

Capacitance will be examined in more detail later in this chapter. For our present purposes, it is sufficient to note that the interconnect line exhibits both parasitic resistance R_{line} [Ω] and capacitance C_{line} [F]. Forming the product of these two quantities gives

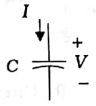


Figure 3.7 Circuit symbol for a capacitor

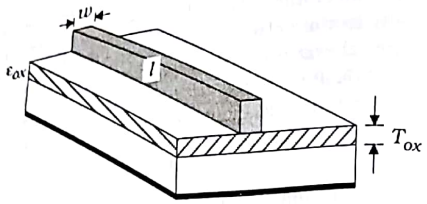
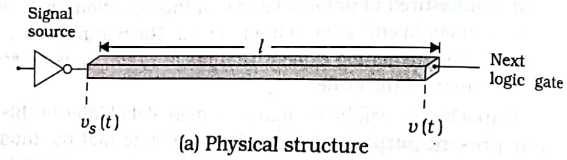


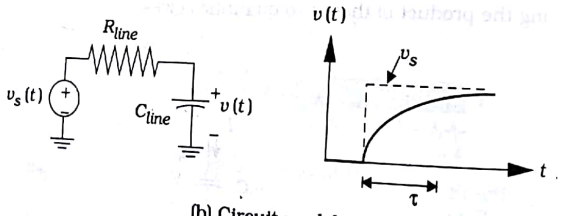
Figure 3.8 Geometry for calculating the line capacitance

$$\tau = R_{line}C_{line} \quad (3.$$

where τ has units of seconds [s] and is called a **time constant**. In hi speed digital circuits, signals on an interconnect line are delayed b which places a limiting factor on the speed of the network. This is illu trated in Figure 3.9. In the physical layout of Figure 3.9(a), the output s nal $v_s(t)$ from a NOT gate is connected to an interconnect line leading the next gates in a logic chain. The voltage at the end of the interconn is labeled as $v(t)$. The parasitic elements R_{line} and C_{line} are used to mod the interconnect circuit as shown in Figure 3.9(b). With this simple c cuit, $v(t)$ is the voltage across the capacitor. If the output voltage $v_s(t)$ NOT gate makes a voltage transition from a 0 to a 1 level as shown in t waveform plot, then $v(t)$ also rises in the same manner. However, t capacitor voltage is delayed by a time constant τ and the shape of t



(a) Physical structure



(b) Circuit model

Figure 3.9 Time delay due to the interconnect time constant

waveform is not as sharp as the source.

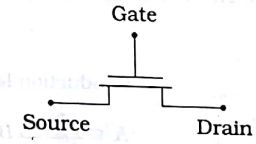
Many aspects of VLSI processing are directed toward minimizing both R_{line} and C_{line}; circuit designers are then faced with creating the fastest possible switching network within the limits of the interconnect delay. This simple discussion gives us a first look at the signal transmission characteristics of interconnect lines. The problems associated with interconnect delays are critically important in high-density VLSI chip designs.

3.2 MOSFETs

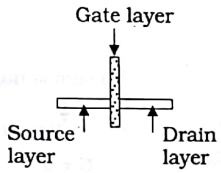
Our discussion in the previous chapter emphasized the technique of designing logic gates from MOSFET switches. To build the circuit on silicon, we need to first understand what a MOSFET looks like at the physical level. We may then proceed to study how logic gates can be designed.

An integrated MOSFET is a small area set of two basic patterned layers that together act like a controlled switch. To determine what the layering scheme should look like, recall the circuit symbol for an nFET shown in Figure 3.10(a). This schematic symbol was designed to resemble the physical structure of the FET itself. Each terminal provides an electrical "entry point" to a patterned feature on one of the layers that makes up the transistor at the chip level. The terminals have been labeled as the gate, the source, and the drain, and each provides access to the device. From the analysis in the previous chapter, we know that the gate electrode acts as the control terminal in that the voltage applied to it determines whether the switch is open or closed. In electrical terms, the voltage applied to the gate determines the electrical current flow between the source and drain terminals.

Our task at this point is to use the concept of integrated circuit layers to create a silicon FET. In Figure 3.10(b) we have drawn a simple representation of the nFET using conducting layers. The vertical line represents the gate layer and divides another layer into source and drain regions that correspond to the schematic symbol. This simplified view is



(a) nFET symbol



(b) nFET layers

Figure 3.10 nFET circuit symbol and layer equivalents

sufficient to understand the physical structure and operation of an
grated transistor.

We may use this drawing to determine the operational characte
needed in the physical structure. Let us assume that a signal G is ap
to the gate and study the behavior of the nFET. If $G = 0$ the sourc
drain are not connected electrically. This is shown in Figure 3.11(a) w
we have removed the gate layer to more clearly illustrate the behav
the device. In this case an open circuit exists so that the two sides
electrically separate; this means that there is no relationship betwe
and B. If we instead apply a gate signal of $G = 1$, then the nFET acts
closed switch and the source and drain sides are electrically conne
This is illustrated in Figure 3.11(b). A conduction layer that bridge
gap has been formed, yielding the logic expression

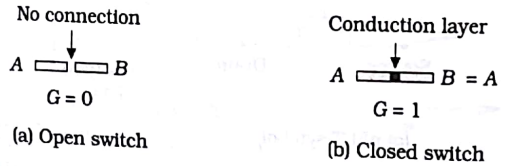
$$B = A$$

Assuming that the drain and source are formed on the same layer,
this behavior can be used to deduce that

- The gate signal G is responsible for the absence or presence of t
  conducting region between the drain and source regions

This is, in fact, how a MOSFET works. The voltage V_G applied to the
is used to electrically create a conduction path that allows current to
between the drain and source sections of the transistor.

Now that we have seen how IC layers can be used to create a MOS
let us examine the physical structure of the transistor in more detail.
ure 3.12 shows the layers involved in creating a generic FET. The d
and source regions are patterned into a silicon wafer; the wafer is equ
lent to the substrate introduced previously in Figure 3.1. Although
drain and source regions are on the same layer, they are physically se
rated from one another by a distance L; L has units of centimeters
and is called the **channel length** of the FET. The width W of the drain
source regions is called the **channel width** and also has units of centi
ters. The **aspect ratio** of the FET is defined as (W/L) and is the m
important parameter to the VLSI designer. The gate layer is separa
from the silicon wafer by a silicon dioxide (glass) layer that acts as



(a) Open switch (b) Closed switch

Figure 3.11 Simplified operational view of an nFET

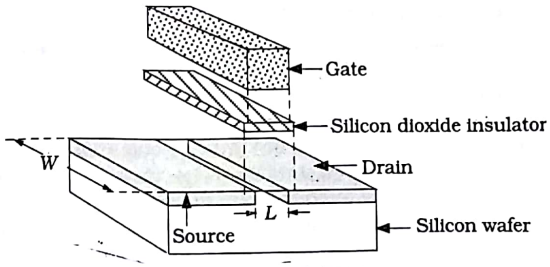
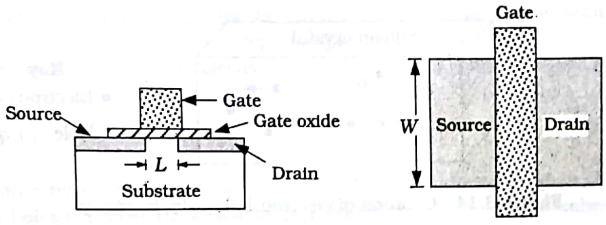


Figure 3.12 Layers used to create a MOSFET

insulator. The dashed vertical lines in the drawing show the alignment of the layers after the stacking process is completed.

Stacking the layers results in the 3-dimensional structure characterized by the drawing in Figure 3.13. The view of Figure 3.13(a) shows a cross-section of the layering scheme. The silicon dioxide layer has been renamed as the **gate oxide** as it resides directly underneath the gate region. The channel length L is shown explicitly in the drawing. The top view in Figure 3.13(b) is identical in form to the simple FET drawing we created in Figure 3.10(b). It shows the drain and source layer being separated by the gate pattern. The only major difference is that the simple drawing was concerned with layers and conduction paths and did not specify sizes.

The basic structure of nFETs and pFETs is the same as that portrayed in Figure 3.13. The difference between the two devices is in the nature of the layers used for the drain and source regions. Both use patterned layers in silicon, but the nFET layer is made to have an excess number of negatively charged electrons, while the pFET drain-source layer has an excess number of positive charges in it. Let us take a short excursion into the world of silicon physics to see how this is accomplished.



(a) Side view (b) Top view

Figure 3.13 Views of a MOSFET

3.2.1 Electrical Conduction in Silicon

In its pure crystalline form, silicon is a relatively poor conductor of elec-
tricity. It is formally called a **semiconductor** because it can conduct
small amounts of electrical current, making it a "partial" conductor. The
atomic density of a silicon crystal is about $N_{Si} \approx 5\times10^{22}$ atoms per cubic
centimeter (units of cm^{-3}), but there are only a small number of electrons
that are available to conduct electricity. These are due to **thermal excita-
tions** where some electrons gain thermal energy and break away from
their host silicon atoms. A sample of pure silicon crystal is said to be
intrinsic material. The number of electrons per cubic centimeter that are
free to carry current is denoted by the symbol n_i and is called the **intrin-
sic carrier density**; the term "carrier" is short for "charge carrier," mean-
ing that the particle has charge. The value of n_i is a function of the
temperature T. At **room temperature** (T = 27° C = 300 K), the intrinsic
density is given by

$$n_i \approx 1.45\times10^{10} \text{ cm}^{-3} \tag{3.16}$$

so that only a small fraction of the electrons in crystal are available for
conduction. The value of n_i increases with increasing temperature since
more thermal energy is added to the structure. However, the number of
free electrons remains small compared to that in a metal.

If we analyze the bonding structure of pure crystal silicon we find that
most of the electrons are confined to orbits around the atomic nuclei of
the atoms. When an electron gains sufficient thermal energy to break
away from its host atom, it may move around in the crystal as a **free** (or
mobile) electron. When an electron leaves its atomic site, it leaves behind
an empty covalent bond that is called a **hole**; this is illustrated in Figure
3.14. The hole represents the absence of an electron, and may be treated
as a "particle" with properties that are opposite to those of electrons. In
particular, since the electron has a negative charge of $-q$ associated with
it, the hole carries a positive charge of value $+q$ that allows it to participate
in the current flow process.[1] Although the particles are independent of

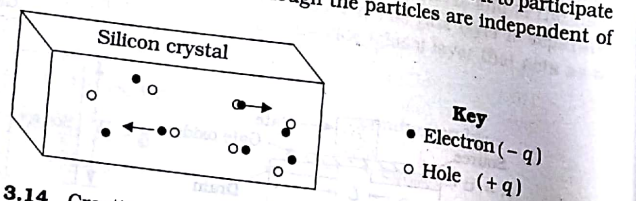


Figure 3.14 Creation of electron-hole pairs in silicon

[1] The numerical value of the fundamental charge unit is $q = 1.602 \times 10^{-19}$ coulombs.

one another, they constitute an **electron-hole pair** when they are created.

The ability of a material to conduct electrical current depends upon the number of freely moving charged particles that are available. Let us introduce two variables that provide this information. We define n to be the number of free electrons per cubic centimeter and p to be the number of free holes per cubic centimeters; both n and p have units of cm^{-3}. In a sample of pure silicon, the only way that a hole is created is by freeing an electron from its host atom. We thus see that

$$n = p = n_i \tag{3.17}$$

holds for our sample. The product of the two values gives

$$np = n_i^2 \tag{3.18}$$

which is a statement of the **mass-action law** that governs the relative numbers of electrons and holes if no currents are flowing. This is valid for any semiconductor in thermal equilibrium, which is equivalent to having zero current flow.

Pure silicon does not conduct current very well, but this may be changed by purposely adding small amounts of impurity atoms, called **dopants**, to create a **doped** sample. The idea is to enhance either the number of electrons, or the number of holes, to aid in the current flow process. The population of free electrons can be increased by adding arsenic (As) or phosphorus (P) atoms to the crystal. The resulting sample is called **n-type** material because it has an excess of negatively charged electrons. When used as dopants, both arsenic and phosphorus "donate" free electrons to the crystal, and are said to act as **donor atoms** or simply **donors**. The number of donors added to one cubic centimeter is given the symbol N_d, with a typical range of values for N_d of around 10^{16} to 10^{19} cm^{-3}. Each donor atoms adds a free electron to the crystal so that we may compute the electron density from

$$n_n \approx N_d \text{ cm}^{-3} \tag{3.19}$$

where the notation n_n means the electron density in an n-type sample. The number of holes in the n-type sample, which we will denote by p_n, is given by the mass-action law as

$$p_n \approx \frac{n_i^2}{N_d} \text{ cm}^{-3} \tag{3.20}$$

In an n-type sample, the electrons are called the **majority carriers** while the holes are called the **minority carriers** due to their relative numbers.

3.2.1 Electrical Conduction in Silicon

In its pure crystalline form, silicon is a relatively poor conductor of elec-tricity. It is formally called a **semiconductor** because it can conduct small amounts of electrical current, making it a "partial" conductor. The atomic density of a silicon crystal is about $N_{Si} \approx 5\times10^{22}$ atoms per cubic centimeter (units of cm^{-3}), but there are only a small number of electrons that are available to conduct electricity. These are due to **thermal excitations** where some electrons gain thermal energy and break away from their host silicon atoms. A sample of pure silicon crystal is said to be **intrinsic** material. The number of electrons per cubic centimeter that are free to carry current is denoted by the symbol n_i and is called the **intrinsic carrier density**; the term "carrier" is short for "charge carrier," meaning that the particle has charge. The value of n_i is a function of the temperature T. At **room temperature** ($T = 27°$ C = 300 K), the intrinsic density is given by

$$n_i \approx 1.45\times10^{10} \text{ cm}^{-3} \tag{3.16}$$

so that only a small fraction of the electrons in crystal are available for conduction. The value of n_i increases with increasing temperature since more thermal energy is added to the structure. However, the number of free electrons remains small compared to that in a metal.

If we analyze the bonding structure of pure crystal silicon we find that most of the electrons are confined to orbits around the atomic nuclei of the atoms. When an electron gains sufficient thermal energy to break away from its host atom, it may move around in the crystal as a **free** (or mobile) electron. When an electron leaves its atomic site, it leaves behind an empty covalent bond that is called a **hole**; this is illustrated in Figure 3.14. The hole represents the absence of an electron, and may be treated as a "particle" with properties that are opposite to those of electrons. In particular, since the electron has a negative charge of $-q$ associated with it, the hole carries a positive charge of value $+q$ that allows it to participate in the current flow process.[1] Although the particles are independent of one another, they constitute an **electron-hole pair** when they are created.

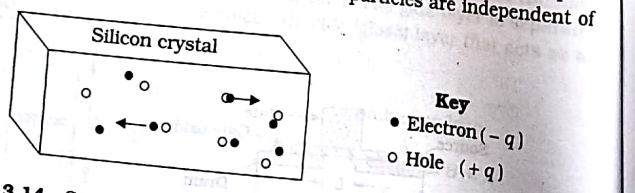


Figure 3.14 Creation of electron-hole pairs in silicon

[1] The numerical value of the fundamental charge unit is $q = 1.602 \times 10^{-19}$ coulombs.

The ability of a material to conduct electrical current depends upon the number of freely moving charged particles that are available. Let us introduce two variables that provide this information. We define n to be the number of free electrons per cubic centimeter and p to be the number of free holes per cubic centimeters; both n and p have units of cm^{-3}. In a sample of pure silicon, the only way that a hole is created is by freeing an electron from its host atom. We thus see that

$$n = p = n_i \tag{3.17}$$

holds for our sample. The product of the two values gives

$$np = n_i^2 \tag{3.18}$$

which is a statement of the **mass-action law** that governs the relative numbers of electrons and holes if no currents are flowing. This is valid for any semiconductor in thermal equilibrium, which is equivalent to having zero current flow.

Pure silicon does not conduct current very well, but this may be changed by purposely adding small amounts of impurity atoms, called **dopants**, to create a **doped** sample. The idea is to enhance either the number of electrons, or the number of holes, to aid in the current flow process. The population of free electrons can be increased by adding arsenic (As) or phosphorus (P) atoms to the crystal. The resulting sample is called **n-type** material because it has an excess of negatively charged electrons. When used as dopants, both arsenic and phosphorus "donate" free electrons to the crystal, and are said to act as **donor atoms** or simply **donors**. The number of donors added to one cubic centimeter is given the symbol N_d, with a typical range of values for N_d of around 10^{16} to 10^{19} cm^{-3}. Each donor atoms adds a free electron to the crystal so that we may compute the electron density from

$$n_n \approx N_d \text{ cm}^{-3} \tag{3.19}$$

where the notation n_n means the electron density in an n-type sample. The number of holes in the n-type sample, which we will denote by p_n, is given by the mass-action law as

$$p_n \approx \frac{n_i^2}{N_d} \text{ cm}^{-3} \tag{3.20}$$

In an n-type sample, the electrons are called the **majority carriers** while the holes are called the **minority carriers** due to their relative numbers.

Example 3.1

Suppose that the donor doping density is $N_d = 2 \times 10^{17}$ cm^{-3}. The ele... density is

$$n_n \approx N_d = 2 \times 10^{17} \text{ cm}^{-3}$$

while the hole concentration is

$$p_n \approx \frac{n_i^2}{N_d} = \frac{(1.45 \times 10^{10})^2}{2 \times 10^{17}}$$

which gives

$$p_n \approx 1 \times 10^3 \text{ cm}^{-3}$$

Obviously, $n_n \gg p_p$ holds for the sample.

The opposite-polarity material is called **p-type** and is created by a... ing boron (B) atoms to the crystal. A p-type material has more positi... charged holes than negatively charged electrons. Boron is used beca... every impurity atom induces a free hole into the bonding scheme. Sinc... hole can "accept" an electron, boron is called an **acceptor** dopant. ... the number of acceptors added per cubic centimeter is denoted by ... symbol N_a. The acceptor density has approximately the same range... that state for donors (about 10^{14} to 10^{19} cm^{-3}), but the effect is exac... opposite: adding boron enhances the concentration of holes p_p in the... type semiconductor. To calculate the carrier densities we use

$$p_p \approx N_a \qquad n_p \approx \frac{n_i^2}{N_a}$$

and refer to the holes as the majority carriers, while the electrons are t... minority carriers since $p_p > n_p$. Both p_p and n_p have units of cm^{-3}.

The conductivity σ of a semiconductor region with carrier densities... and p is given by

$$\sigma = q(\mu_n n + \mu_p p)$$

where μ_n and μ_p are called the electron and hole **mobilities**, respectively... with units of cm^2/V-sec. Qualitatively, the mobilities are parameters tha... indicate "how mobile" a particle is. A small value of μ indicates that it i... difficult for the particle to move, while a large value of μ implies relatively... free motion. For intrinsic silicon, the room temperature mobilities are

$$\mu_n = 1360 \qquad \mu_p = 480$$

which gives a conductivity of $\sigma \approx 4.27 \times 10^{-6}$ [Ω-cm]$^{-1}$ or $\rho \approx 2.34 \times 10^5$ [Ω-cm]. For comparison purposes, we note that quartz glass, which is an excellent insulator, has a resistivity ρ of about 10^{12} [Ω-cm].

If we specialize to an n-type sample where $n_n >> p_n$, then we may usually approximate the conductivity as

$$\sigma \approx q\mu_n n_n \tag{3.27}$$

Similarly, the conductivity of a p-type region is often estimated by

$$\sigma \approx q\mu_p p_p \tag{3.28}$$

For the present discussion, however, the most important point to remember is that an n-type region is dominated by negatively charged electrons, while a p-type region has mostly positively charged holes.

Example 3.2

Consider a sample of silicon that is doped p-type with boron added at a density of 10^{15} cm^{-3}. The majority charge carriers are holes with a density of

$$p_p = 10^{15} \text{ cm}^{-3} \tag{3.29}$$

while the minority carrier electron density is

$$n_p \approx \frac{(1.45 \times 10^{10})^2}{10^{15}} = 2.2 \times 10^5 \text{ cm}^{-3} \tag{3.30}$$

For this sample, the mobilities are given by $\mu_n \approx 1350$ cm^2/V-sec and $\mu_p \approx 450$ cm^2/V-sec. The conductivity is

$$\sigma = (1.6 \times 10^{-19})[(1350)(2.2 \times 10^5) + (450)(10^{15})] = 0.072 \quad [\Omega\text{-cm}]^{-1} \tag{3.31}$$

which is equivalent to a resistivity of

$$\rho = \frac{1}{0.08} = 13.9[\Omega\text{-cm}] \tag{3.32}$$

A quick check on the values shows that $\mu_p p_p >> \mu_n n_p$ for this example. In general, the resistivity of silicon samples is on the order of 1 to 10 Ω-cm.

This example shows that the doping level is the most important factor in determining the conductivity in n-type or p-type silicon. Increasing the doping density creates more charged particles that aid in the conduction process. However, a large number of impurity atoms creates more barriers

that the particles must pass, making them less mobile. This is called **impurity scattering**, and is described by writing the mobility μ as a function of the total doping density N. In general, $\mu(N)$ decreases with increasing N. An empirical equation for this effect is (see Reference [3])

$$\mu = \mu_1 + \frac{\mu_2 - \mu_1}{1 + \left(\frac{N}{N_{ref}}\right)^{\alpha}} \tag{3.34}$$

where μ_1, μ_2, N_{ref}, and α are constants. For electrons, the room temperature silicon values are approximately $\mu_1 = 92$ cm^2/V-sec, $\mu_2 = 1380$ cm^2/V-sec, $N_{ref} = 1.3 \times 10^{17}$ cm^{-3}, and $\alpha = 0.91$. The corresponding hole values are $\mu_1 = 47.7$ cm^2/V-sec, $\mu_2 = 495$ cm^2/V-sec, $N_{ref} = 6.3 \times 10^{16}$ cm^{-3}, and $\alpha = 0.76$. The decrease in mobility with increasing doping is called a **second-order effect** in device physics. Although it is tempting to ignore impurity scattering in simple calculations, doing so can introduce significant errors. A final comment is that for a given doping level N,

$$\mu_n > \mu_p \tag{3.35}$$

This means the electrons can move more easily than holes. Physically this can be visualized by assuming that electrons are true particles in the classical sense, while holes are seen as the "absence of particles."

The above analysis assumes that only donors N_d *or* acceptors N_a will be present in the sample. In CMOS processing, however, most doped regions have both donors *and* acceptors. The polarity is established by the dominant species. To create an n-type region, we need $N_d > N_a$ so that the donors outnumber the acceptors. The carriers are computed by

$$n_n \approx N_d - N_a \qquad p_n \approx \frac{n_i^2}{(N_d - N_a)} \tag{3.35}$$

with the electrons in the majority. For a p-type region, we need $N_a > N_d$ such that the carrier densities are given by

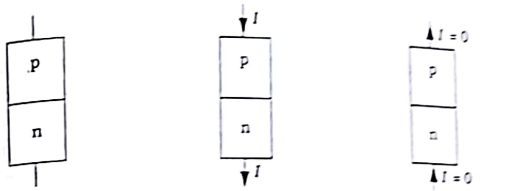
$$p_p \approx N_a - N_d \qquad n_p \approx \frac{n_i^2}{(N_a - N_d)} \tag{3.36}$$

for the majority carrier holes p_p and minority carrier electrons n_p. To calculate the mobility, we use the total doping density $N = N_a + N_d$ in equation (3.33). The conductivity is still calculated from

$$\sigma = q(\mu_n n + \mu_p p) \tag{3.37}$$

since only the values are altered. One special case is where $N_d = N_a$. Since every electron released by a donor is matched by a hole in an acceptor, the material looks like an intrinsic same with $n = p = n_i$. This is called **total compensation**. Note that the mobility will be smaller than the intrinsic value since the number of dopants is not zero.

When an n-type region touches a p-type region, a very special interface is formed. This **pn junction** allows electrical conduction in only one direction, from the p-side to the n-side. If we attempt to force current from the n-side to the p-side, the junction blocks it and acts like an open switch. The properties of a pn junction are summarized in Figure 3.15. In electronics, this feature is used to make a device called a **diode**. The characteristic of allowing current to flow in only one direction is called **rectification**.



(a) A pn junction (b) Forward current (b) Reverse blocking

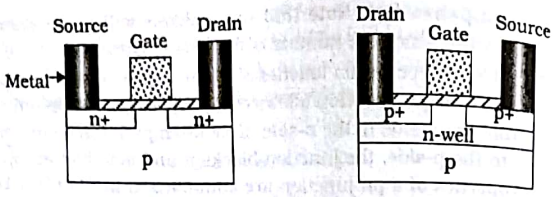
Figure 3.15 Formation and characteristics of a pn junction

3.2.2 nFETs and pFETs

With the distinction between n-type and p-type regions established, we can now define the structures of nFET and pFETs. This is a very simple task: the polarity of a FET (n or p) is determined by the polarity of the drain and source regions. The device is designed so that the conducting layer shown in Figure 3.11(b) has the same polarity as the drain and source regions when the device is conducting. An nFET uses n-type drain and source regions, while a pFET has p-type drain and source regions. These are shown in Figure 3.16(a) and (b), respectively. Metal contacts have been added to illustrate how we can connect the drain and source regions to other parts of the circuit.

Let us examine the nFET first. The drain and source regions are labeled as "n+" to indicate that they are **heavily doped**. This means that the donor doping density N_d is relatively large, with a typical value around $N_d = 10^{19}$ cm^{-3}. The substrate layer (at the bottom) is now specified to be p-type with an implied boron doping density of N_a; a reasonable value for the acceptor doping would be $N_a = 10^{15}$ cm^{-3}. Note that pn junctions are formed between the n+ regions and the p-type substrate. These are used to block the current flow between the substrate and the top n+ layers of the device as discussed in the context of Figure 3.15.

The pFET has the same structure as the nFET but the polarities are



(a) nFET cross-section (b) pFET cross-section

Figure 3.16 nFET and pFET layers

reversed. Source and drain regions are p+ sections that are embedded in an n-type "well" layer; the n-well itself resides on top of the p-type substrate. There are several pn junctions formed in this device; all are used to prevent current flow between adjacent layers. The layering scheme is more complicated because CMOS design uses both nFETs and pFETs that are built in a single silicon wafer. If we choose the wafer to be p-type, then the nFETs can be created as in Figure 3.16(a) by just adding n+ regions. However, if we add pFET p+ regions directly to the p-substrate, then we lose the needed structure of the layering: p+ into an n-region. Since no pn junction is formed, we cannot control the current flow. To correct for this problem, the n-well layer is used to build the pFET as shown. This assures us that the transistors have opposite electrical characteristics.

3.2.3 Current Flow in a FET

MOSFETs are used as voltage controlled switches in CMOS logic circuits. Applying a signal to the gate electrode results in either an open or closed switch as we saw previously in Figure 3.11 for an nFET. The creation of the conducting layer underneath the gate is due to the property of the capacitance that is built into the gate region of the MOSFET itself. A simple parallel-plate capacitor from basic physics is shown in Figure 3.17.

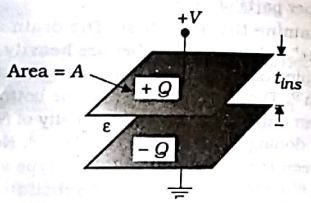


Figure 3.17 A parallel-plate capacitor

This device consists of two identical metal plates that are separated by an insulator with a thickness t_{ins} cm. The plates have an area A in units of cm^2. A capacitor stores electric charge Q on the plates as indicated in the drawing. With a voltage difference of V applied across the plates, the charge is given by

$$Q = CV \tag{3.38}$$

where C is the capacitance. For a parallel-plate structure, the basic formula for the capacitance is well known as

$$C = \frac{\varepsilon A}{t_{ins}} \tag{3.39}$$

where ε is the permittivity of the insulator in units of F/cm. The value of ε depends upon the material used to separate the plate.[2] The most important observation to be made is that applying a positive voltage V to the upper plate **induces** a negative charge $-Q$ on the lower plate.

Let us examine an nFET in more detail by referring to the drawing in Figure 3.18. The central region of the device is designed to be a capacitor. The gate oxide layer is the insulating glass between the gate (which acts as the upper plate) and the p-type substrate (which acts as the lower plate). In the early days of MOS, the gate was made out of aluminum (Al), which is a metal. The layering thus gave rise to the acronym "MOS" for metal-oxide-semiconductor. In modern processing, the gate material is **polycrystal silicon**, which is usually called **polysilicon** or just **poly**.[3] Although the gate material is no longer a metal, the acronym has never been changed with any success and still remains in use today.[4]

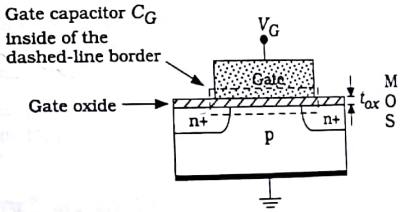


Figure 3.18 The gate capacitance in an n-channel MOSFET

[2] Physically, the permittivity is a measure of the electric energy storage capacity of the material.

[3] Polycrystal consists of small regions of silicon crystals, called crystallites; the material is discussed in more detail in Chapter 4.

[4] The most common substitute is the IGFET, which stands for insulated-gate FET.

To describe the MOS structure we introduce the **oxide capacitance**

$$C_{ox} = \frac{\varepsilon_{ox}}{t_{ox}} \quad (3.$$

which has units of F/cm^2. Comparing this to the parallel-plate formula equation (3.39) shows that the area has been left out. This is done on p pose so that C_{ox} can be applied to any device in the circuit. If the gate h an area A_G cm^2, then the total **gate capacitance** of the FET is

$$C_G = C_{ox}A_G \quad (3.$$

which has units of farads. In this formula, ε_{ox} is the permittivity of t glass insulating layer. In modern technology, silicon dioxide (SiO_2) is us for almost all silicon MOSFETs and the permittivity is given by

$$\varepsilon_{ox} = 3.9\varepsilon_0 \quad (3.4$$

where $\varepsilon_0 \approx 8.854 \times 10^{-14}$ F/cm is the permittivity of free space. The oxid thickness t_{ox} is a critical parameter in CMOS. For reasons that will b seen later, a thin oxide (small t_{ox}) is desirable. Modern processing line have $t_{ox} \leq 10$ nm = 100 Å with advanced fabrication facilities providing th capabilities to create an oxide with less than half this thickness.[5]

Example 3.3

Consider a gate oxide that has a thickness of $t_{ox} = 50$ Å $= 50 \times 10^{-8}$ cm. The oxide capacitance per unit area is

$$C_{ox} = \frac{(3.9)(8.854\times10^{-14})}{50\times10^{-8}} = 6.91\times10^{-7} \quad \text{F/cm}^2 \quad (3.43)$$

which is a typical value. Suppose that the gate of a FET has an area

$$A_G = (1\times10^{-4}\text{cm})\times(0.4\times10^{-4}\text{cm}) = 4\times10^{-9}\text{cm}^2 \quad (3.44)$$

We note that 10^{-4} cm = 10^{-6} m = 1 μm (micrometer), which is often called 1 **micron** and is the metric we use to describe FET dimensions. For this example, the gate capacitance would be

$$C_G = (6.91\times10^{-7})(4\times10^{-9}) = 2.76\times10^{-15} \text{ F} \quad (3.45)$$

Defining 1 **femtofarad** (fF) as 1 fF = 10^{-15} F, the gate capacitance is

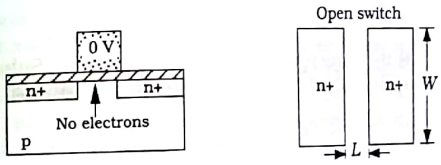
$$C_G = 2.76 \text{ fF} \quad (3.46)$$

5 One Angstrom (Å) is 10^{-10} cm = 10^{-8} m.

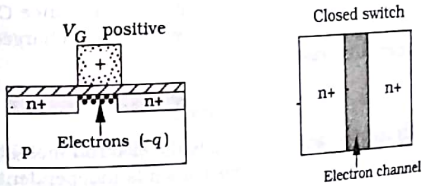
which is typical for a modern device. The electronics expert will notice that this is much smaller than the typical capacitance values encountered in the everyday world.

The above discussion illustrates that the gate of a MOSFET is really one side of an MOS capacitor. Applying a voltage on the gate causes a charge layer of the opposite polarity to form on the opposite side of the capacitor, i.e., in the silicon region directly beneath the gate oxide. If we apply a positive voltage to the gate, then a negative electron layer is created in the silicon. Conversely, using a voltage that is negative with respect to the rest of the device creates a positively charged layer of holes in the silicon. The formation of thin layers of charge in the silicon is possible because it is a semiconductor material where the number of charge carriers depends upon the local electrical conditions. Armed with this observation, the mechanism of current flow becomes easy to visualize.

Consider first an nFET as shown in Figure 3.19. The drain and source regions are n-type, but they are physically separated by a section of the p-type substrate. In Figure 3.19(a) the gate voltage has the value of 0 V so that no charge is induced underneath the gate oxide. The top view of the silicon provided on the right side shows that the drain and source are separated, so that no current can flow between them. This is due to the current-blocking properties of the pn junctions, and is analogous to having an open switch. Applying a positive voltage to the gate as in Figure 3.19(b), the capacitive MOS structure induces a layer of negatively



(a) Zero gate voltage



(b) Positive gate voltage

Figure 3.19 Controlling current flow in an nFET

charged electrons underneath the gate oxide. The electron layer est... lishes an electrical connection between the drain and source regions... shown. The electrons form a "channel" for current to flow through... nFET, and the device acts like a closed switch. The formation of the la... requires that the gate voltage be larger than the **threshold voltage**... that was introduced in Chapter 2. A typical value for the nFET thresh... voltage is $V_{Tn} = 0.70$ V. This parameter is established by the fabricat... sequence, and is always assumed to be a known value at the VLSI desi... level.

The channel charge in units of coulombs is given by

$$Q_c = -C_G(V_G - V_{Tn}) \tag{3.4}$$

where V_G is the gate voltage and C_G is the gate capacitance. The voltag... difference $(V_G - V_{Tn})$ is used because no charge forms until V_G reache... V_{Tn}. The negative sign in the equation indicates that the channel consis... of negatively charged electrons. The current I flowing through the chann... can be written as

$$I = \frac{|Q_c|}{\tau_t} \text{ C/sec} \tag{3.48}$$

where we have introduced τ_t as the channel **transit time** in units of sec... onds. Physically, τ_t is the average time needed for an electron to move... from one n+ region to the other, and can be calculated from

$$\tau_t = \frac{L}{v} \tag{3.49}$$

where v is the particle velocity in units of cm/sec. Substituting into the... current equation (3.48) gives

$$\begin{aligned} I &= \frac{C_G}{(L/v)}(V_G - V_{Tn}) \\ &= vC_{ox}W(V_G - V_{Tn}) \end{aligned} \tag{3.50}$$

We have used the definition of the gate capacitance C_G from equation (3.41) in writing the second line. The velocity of a charged particle moving in a FET can be estimated as

$$v = \mu_n E \tag{3.51}$$

where E is the electric field and μ_n is the electron mobility. With a voltage V applied between the n+ regions (which is independent of the gate voltage), the electric field is approximately

$$E = \frac{V}{L}$$

with units of volts/cm. Substituting these relations into equation (3.50) gives

$$I = \mu_n C_{ox}\left(\frac{W}{L}\right)(V_G - V_{Tn})V \tag{3.53}$$

as our first approximation for the current. The linear resistance R_n of the device can be calculated by taking the ratio

$$R_n = \frac{V}{I} = \frac{1}{\beta_n(V_G - V_{Tn})} \tag{3.54}$$

where we have defined the parameter

$$\beta_n = \mu_n C_{ox}\left(\frac{W}{L}\right) \tag{3.55}$$

called the **device transconductance**[6] that has units of A/V^2. With this model we can view the nFET as a device that acts either as an open switch with no channel and $R \to \infty$, or a closed switch with a resistance of R_n between the drain and source sides.

One fine point that needs to be mentioned is that the mobility μ_n used in the MOSFET analysis is the value at the "surface" of the silicon, and is therefore called the **surface mobility**. This is different from that calculated using equation (3.33) which is valid for the **bulk mobilities**, i.e., the value inside the material. A simple estimate is that the surface mobility is about (1/2) the bulk mobility value. In practice, measured values from the laboratory are used to design circuits.

A deeper analysis will show that MOSFETs are intrinsically **non-linear devices** in that the current I through a FET is a non-linear function of the voltage V across it. This relationship will be examined in more detail in Chapter 6. For simple modeling, however, we often treat the transistor as a linear resistor with a value

$$R_n = R_{c,n}\left(\frac{L}{W}\right) \tag{3.56}$$

where

$$R_{c,n} = \frac{1}{\mu_n C_{ox}(V_G - V_{Tn})} \tag{3.57}$$

is the equivalent sheet resistance for the electron current flow channel.

A pFET behaves in a similar manner, except that all of the polarities

[6] In general, a transconductance parameter has units of amps over volt with squares, cubes, etc. A transresistance has basic units of volts/amps.

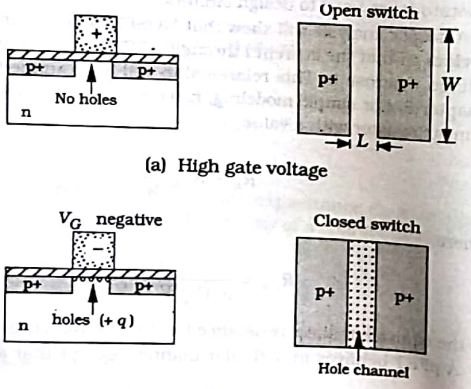
are reversed. The operation is summarized in Figure 3.20. If the gate
age is made positive [see Figure 3.20(a)] then only negative charge
the n-type layer exists underneath the gate oxide. Since the drain
source regions are both p-type, they are electrically separated from
other by the n-type region. The transistor thus acts like an open sw
If, on the other hand, we apply a negative voltage on the gate, then a
of positively charged holes can form underneath the gate oxide as
trated in Figure 3.20(b). To create the hole layer, the difference in vo
between the highest voltage p+ region and the gate must be greater
the magnitude of the pFET voltage $|V_{Tp}|$. This gives an electrical con
tion channel between the p-type source and drain regions so cur
flows through the transistor. The pFET is then similar to a closed sw
Like the nFET, the pFET also exhibits a resistance that is estimated b

$$R_p = \frac{1}{\beta_p(V_G - |V_{Tp}|)} \qquad (3.$$

By convention, V_{Tp} is a negative number, so we use $|V_{Tp}|$ to make the f
mula have the same form as the nFET expression. In this equation,

$$\beta_p = \mu_p C_{ox}\left(\frac{W}{L}\right) \qquad (3.$$

is the device transconductance for the pFET, with μ_p the hole mobility.
Although our first look at conduction through nFETs and pFETs h



(a) High gate voltage

(b) Negative gate voltage

Figure 3.20 Switching behavior of a pFET

been highly simplified, it does relay on a useful visualization for the VLSI designer. Oftentimes a simple model is more useful than a complex one.

3.2.4 Driving the Gate Capacitance

Let us dig a little deeper into the behavior of the MOS capacitor system. It is fundamental to the operation of the FET, but the presence of any capacitance in a CMOS integrated circuit causes signal delays. Figure 3.21 shows the circuit symbol for a capacitor with a capacitance C. The drawing defines a positive current i as flowing into the side of the capacitor that has a positive voltage on it. Note that positive charge $+Q$ is stored on the top plate, while the bottom plate holds a negative charge with a value of $-Q$. The current i flowing into the capacitor as a function of time t is the time rate-of-change of the charge

$$i(t) = \frac{dQ}{dt} \qquad (3.60)$$

Since $Q = CV$, we may substitute for the charge and obtain the I-V relation for a capacitor

$$i = C\,\frac{dV}{dt} \qquad (3.61)$$

This tells us several things about how voltage signals behave in a CMOS circuit. First, it is not possible to change the voltage $V(t)$ across a capacitor in an instantaneous manner, i.e., with $dt \to 0$; this is because (dV/dt) would have to be infinite in value, but it is physically impossible to have an infinite current i. If we apply this conclusion to the gate capacitance C_G of a FET as shown in Figure 3.22, we conclude that the gate voltage V_G cannot be changed without experiencing a delay. This corresponds to the time required to transfer the charge

$$Q = C_G V_G \qquad (3.62)$$

on to, or off of, the gate electrode. When combined with the mechanism of switching a FET to open and closed switching states, this implies that the

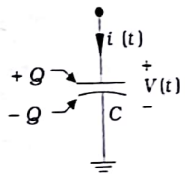


Figure 3.21 Voltage and current in a capacitor

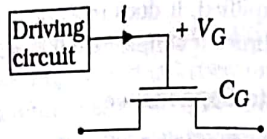


Figure 3.22 Driving the gate of a FET

transistor itself introduces signal delays. The value of the capacitance determines the amount of charge necessary to change the voltage; a large capacitance implies a long delay.

The second important observation is that capacitors store electric energy. Charging and discharging C_G corresponds to changing the energy stored in the device, so that switching a transistor on or off requires that we transfer energy from one point to another in a circuit. The power, in units of **watts** [W] is related to the energy E by

$$P = \frac{dE}{dt} \tag{3.6}$$

where E has units of **joules** [J]. By definition, 1 watt of power means that 1 joule of energy has been transferred in 1 second. For an electrical device with voltage V that has a current i flowing into it, the power is given by the product $P = Vi$. Using equation (3.61) for a capacitor gives

$$P = V\left(C\frac{dV}{dt}\right) = \frac{d}{dt}\left(C\frac{V^2}{2}\right) \tag{3.64}$$

so that the electric energy E_e stored in a capacitor with a voltage V is

$$E_e = \frac{1}{2}CV^2 \tag{3.65}$$

When applied to a CMOS switching network, this means that changing the gate voltage of a FET from 0 V to V_{DD} requires energy of

$$E_e = \frac{1}{2}C_G V_{DD}^2 \tag{3.66}$$

for *every* transistor in the circuit.

Now note that the driving circuit must transmit current through an interconnect wire that has a resistance R_{line}. Resistors do not store electric energy. Instead, they **dissipate** power by changing it to heat. The power P_R dissipated by a resistance R is calculated from

$$P_R = Vi = i^2R \tag{3.67}$$

where we have used Ohm's law. This illustrates that flowing currents induce localized heating effects. This applies to every electrical device in

the circuit, not just interconnect lines.

These simple observations bring out some critical aspects of VLSI circuit design that will be examined throughout the book. Two immediate considerations that arise are

- Switching delays are due to the physical characteristics of the devices and interconnects.
- Every switching event requires energy transfer in the circuit. This implies that power dissipation will occur within the circuit.

The first consideration implies that the designer must understand the nature of switching delays in order to design a fast digital network. The characteristics of both the FETs and the interconnects influence the overall system speed, so VLSI design deals with the network as a whole. The second statement is more practical. Excessive localized heating may be so severe that it melts the silicon crystal and destroys the chip. This, of course, must be avoided by proper design and the use of heat-removal techniques. If the chip is for a portable unit that uses batteries for a power supply, then the design must reduce the power requirements to extend battery life.

3.3 CMOS Layers

Now that we have seen how patterned layers of materials are used to create nFETs and pFETs, let us move to a higher level and examine the entire structure of a CMOS integrated circuit.

CMOS provides the economic basis for a huge segment of the world computing industry. Many companies compete in the marketplace, with each attempting to provide a more advanced technological base than the other. Because of the rapid evolution of high-density circuit manufacturing techniques in the first years of the 21st century, countless variations in CMOS have been introduced. We will choose a rather simple process to study here and purposely avoid advanced (and, hence, complicated) techniques. In particular, we will concentrate on an **n-well process** as being typical.

First of all, let us define what a "CMOS fabrication process" is. In simplest terms, this refers to the sequence of steps that we use to take a bare "wafer" of silicon to the finished form of an electronic integrated circuit. The details of the fabrication process will be discussed in Chapter 5. For the moment, we will only be concerned with the final structure.

The n-well process starts with a p-type substrate wafer that is used as a base layer for building all transistors. nFETs can be fabricated directly in the p-type substrate, while n-well regions are added to accommodate pFETs. The cross-sectional view in Figure 3.23 illustrates the nFET and pFET structures after they have been fabricated on the sub-

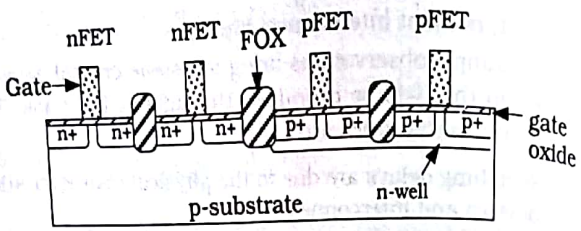


Figure 3.23 MOSFET layers in an n-well process

strate. From this drawing, we can identify the following layer types:

- p-substrate
- n-well
- n+ (nFET drain/source)
- p+ (pFET drain/source)
- gate oxide
- gate (polysilicon)

Note that the term "layer" implies a region with distinct electrical characteristics, even though it may be physically at the same geometrical level as another layer (such as the n+ and p+ layers). The drawing also shows regions labeled as "FOX" which defines **field oxide** sections. Field **regions** are simply recessed insulating glass (silicon dioxide) sections that are inserted in between adjacent FETs to provide **electrical isolation**. The glass acts to insure that there is no current flowing between the two transistors, keeping them electrically separate. Another point that is worth mentioning again is that junctions between n-type and p-type regions have the ability to block current flow. It is therefore assumed that n-regions and p-regions are electrically isolated.[7]

The top view patterning for this example is shown in Figure 3.24. In this drawing, the only layers that are shown explicitly are n-well, n+ (nFET drain/source), p+ (pFET drain/source), and gate (polysilicon). The p-substrate is implied, as are the oxide layers. Note that FOX surrounds every transistor so there is an implied field region everywhere except at transistor sites. FOX regions are rarely shown explicitly, so it is important to remember that every device in the wafer is automatically isolated from every other device.

Once the base transistor layers have been defined, we add conductive metal layers separated by glass insulators to allow for wiring. Modern processes tend to allow for five or more metal interconnect layers to ease the problem of massive wiring in complex circuits. The example in Figure

[7] The ability to block current flow requires that the voltage on the n-side be higher than the voltage on the p-side.

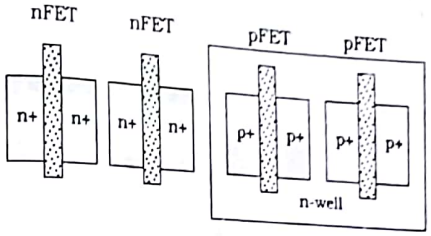


Figure 3.24 Top view FET patterning

3.25 shows two metal layers to illustrate the main points. After the FETs are formed, an oxide layer (Ox1) is deposited over the wafer surface and planarized. A "hole" (called a contact cut) is then etched in the oxide to allow electrical access to drain/source regions. This is shown as the Active contact in the drawing; it is filled with a conducting metal such as tungsten. Metal1 is then deposited on top, followed by another insulating oxide layer (Ox2). We note that Metal1 can also be connected to the gate layer by using an etch hole in Ox1. The second metal layer (Metal2) is then deposited on top of Ox2. Electrical contact between Metal1 and Metal2 is accomplished using a Via, which is a hole etched in Ox2 and filled with a conducting metallic "plug" as shown.

Now that we have seen how metal interconnect layers can be added to the CMOS process, it is important to make the following observations:

- Metal layers are electrically isolated from each other and the transistors by glass
- Electrical contact between adjacent conducting layers requires that we create **contact cuts** and **vias** in the oxide between them

These imply that we can "cross" conducting layers without creating an electrical path between them. Examples of these rules are provided by the layout (top view) drawing in Figure 3.26. Metal2 lines can cross over every other layer; a via is needed to contact Metal1. Metal1 can be connected to

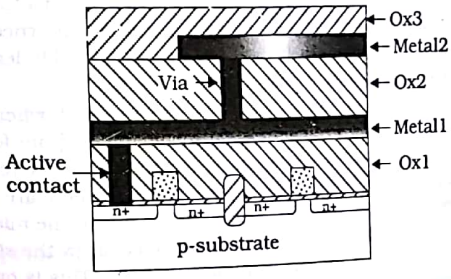


Figure 3.25 Metal interconnect layers

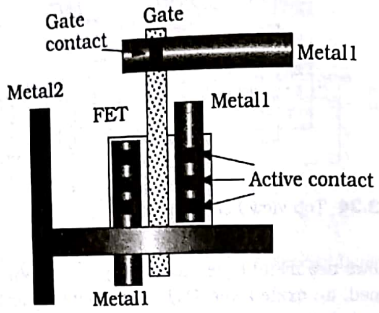


Figure 3.26 Interconnect layout example

the gate using the Gate contact, while an Active contact is used to provide an electrical connection between Metal1 and the drain/source regions of the FET.

CMOS circuits are designed by creating nFETs and pFETs in silicon, and then wiring them together using interconnect lines formed on the conducting layers. One interesting point that may be obvious is that digital CMOS logic circuits consist only of transistors and wires. No other devices are needed, regardless of the complexity of the system. Once we learn how to design basic FETs and add the interconnect wiring, then we will understand the basics of CMOS VLSI!

3.4 Designing FET Arrays

CMOS logic gates are switching networks that are controlled by the input variables. These switching arrays use FETs that are wired together in series and parallel groups in a manner that allows us to create the desired functions. In Chapter 2 we learned how to build logic gates using FETs at the schematic level. These must eventually be translated to silicon patterns for the final design. In VLSI, the patterns themselves become the circuits. Tracing signals and voltages using patterned polygon shapes may seem a bit strange at first, but you will quickly learn to "read" the logic flow and operations that they represent.

Let us start with the simplest case of an n-stack where two nFETs are in series. Figure 3.27(a) shows the schematic diagram for this case. The signals *A* and *B* are applied to the gate terminals of the respective transistors. To construct the silicon pattern, note that there are really only three n+ regions that are needed: one on the left, one in the middle, and one on the right. This simple observation allows us to draw the silicon pattern for the 2-transistor group shown in Figure 3.27(b). This is our first lesson in high-density integration:

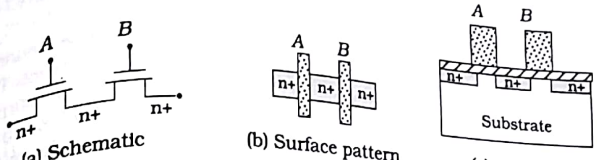


Figure 3.27 Silicon patterning for two series-connected nFETs

- Devices can share patterned regions, which may reduce the layout area or complexity.

In the present case, this says that it is not necessary to first build individual devices and then wire them together. A more efficient design results if we combine n+ regions. The side view shown in Figure 3.27(c) shows that this technique does indeed create a signal path that consists of two transistors. Conduction from left to right occurs only if both transistors are conducting.

This technique can be applied to any group of series-connected FETs. A 3-FET chain is shown in Figure 3.28. Instead of labeling every region in the surface layout drawing, it is usually more convenient to provide a key that associates each fill pattern with a specific material. Color coding the layers is even easier, and is the preferred technique used in computer-based design aids. Metal lines have been added on the left and right sides, along with active contacts (that connect the metal to the n+ regions). These define electrical connections to the nodes x and y shown in the drawing and are required to connect the transistor group to other parts of the circuit. This pattern also shows the channel width *W* for the three transistors and thus provides more information than the simpler surface patterning scheme used in Figure 3.27. In the initial design stages, the width is not always shown explicitly. At that point in the design, we are usually more interested in the signal flow path and the circuit topology than the details of the transistors. In other words, we want

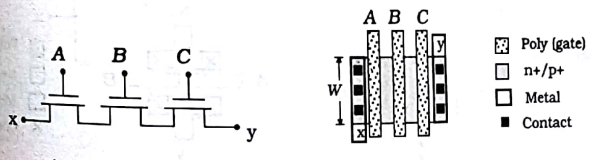
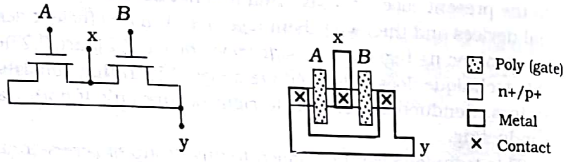


Figure 3.28 Three series-connected nFETs

to design and verify the logic before tackling the details involved in ch
ing the actual sizes of the transistors.

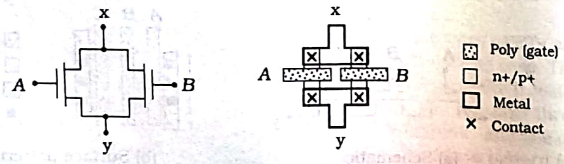
Parallel-connected FETs can be patterned in the same manner. In F
ure 3.29, two nFETs are wired in parallel using metal patterns. The par
lel connection can be understood by noting that the drain/source regio
of both transistors are connected between the nodes labeled x and
which implies that they are in parallel. The scheme is shown in Figu
3.29(a), while Figure 3.29(b) illustrates the transistor patterns and wir
scheme. This approach to surface patterning maintains the orientation
the transistor patterns that were used for series-connections groups. Th
may be desirable in that a uniform layout philosophy may lead to a high
packing density on the silicon surface.



(a) Schematic (b) Surface pattern

Figure 3.29 Parallel-connected FET patterning

An alternate layout strategy for parallel FETs is shown in Figure 3.30. This uses vertical drain-source orientations for the transistors. In this approach, two FETs are created with separate n+ regions. The parallel connection is accomplished using metal interconnects to give the nodes x and y shown in the drawing. While these two techniques maintain the same orientation for both FETs (horizontal or vertical), this is not mandatory. Only the wiring and the resulting electrical connections are important. Separated transistors usually require more area than those that share drain/source regions, so this type of scheme is restricted to special situations.



(a) Schematic (b) Surface pattern

Figure 3.30 Alternate layout strategy for parallel FETs

3.4.1 Basic Gate Designs

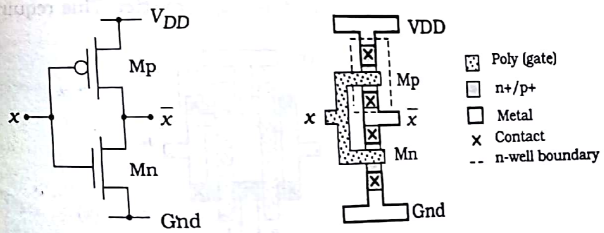
Now that we have seen the basic ideas involved in CMOS layout, let us examine the surface patterns used for CMOS logic gates in silicon. In the simplified view used in this section, patterned lines on conducting layers are viewed as paths that "steer" electrical current and establish voltages. The widths of the lines are not important at this level; only the topology of the network is needed to trace the logic. This approach is very useful in the initial stages of creating a CMOS layout design as it allows one to play with the location and orientation of the devices to see how well they pack together.

Consider first a NOT gate. Figure 3.31(a) shows how the circuit is wired using transistors Mn and Mp as a complementary pair. The silicon implementation is shown in Figure 3.31(b). The layout has been structured so that there is a visual one-to-one correspondence with the circuit. Some of the important aspects are that

- Both the power supply (VDD) and ground (Gnd) are routed using the Metal layer
- n+ and p+ regions are denoted using the same fill pattern. The difference is that pFETs are embedded within an n-well boundary
- Contacts are needed from Metal to n+ or p+ since they are at different levels in the structure

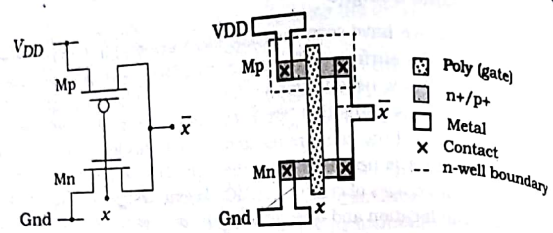
The ability to trace the logic operation on the layout is a useful skill to develop. In this case, the input x controls the poly gate. When $x = 0$, Mp acts like a closed switch while Mn is open, giving an output of VDD, i.e., $\bar{x} = 1$. Conversely, an input of $x = 1$ forces Mn into conduction while Mp is open. This connects Gnd to the output, and is equivalent to $\bar{x} = 0$.

An alternate layout is shown in Figure 3.32. In this case, the NOT gate has been drawn like a 2:1 multiplexor. While the operation is entirely equivalent, a one-to-one translation results in FETs that are at right



(a) Circuit (b) Layer patterning

Figure 3.31 Translating a NOT gate circuit to silicon



(a) Circuit (b) Layer patterning

Figure 3.32 Alternate layout for a NOT gate

angles to those in Figure 3.31. This illustrates the fact that different geometrical layouts can be used to implement CMOS circuits. Variations in the layout strategy are not important until the actual sizes of the patterns are taken into account. This aspect of physical design is discussed later in the book.

One goal of physical design is to minimize the area of the overall chip. This can be accomplished at many levels using various techniques. One example is shown in Figure 3.33 where two NOT circuits share the VDD and Gnd connections. The left inverter has an input of a and produces $\bar{a}$, while the right circuit inverts b to $\bar{b}$. It is easy to visualize the area savings over the brute-force approach that uses two separate circuits. Of course, the design must have the need for two inverters close together in the logic chain. The same layout may be used as a basis for creating the non-inverting buffer in Figure 3.34. This uses two series-connected inverters as shown in Figure 3.34(a) to provide the logic. Although an input of a produces the same Boolean logical value for a, the buffer provides electrical reshaping of the signal and provides additional "drive strength" for large fan-outs. The layout scheme in Figure 3.34(b) uses the output of the left inverter to feed the input of the right inverter. This requires a metal-

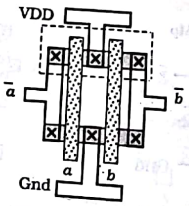
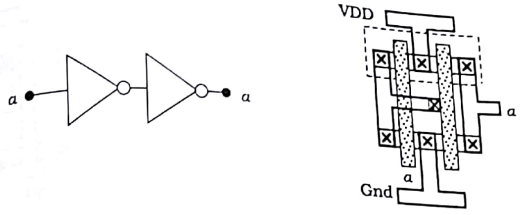


Figure 3.33 Two NOT gates that share power supply and ground



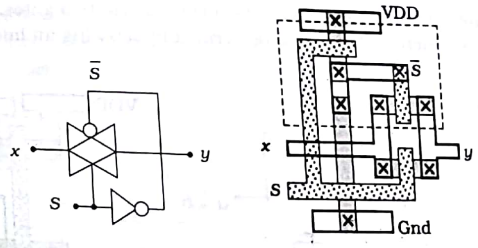
(a) Logic diagram (b) Layout

Figure 3.34 Non-inverting buffer

to-poly contact between the two stages as shown.[8] In addition, the scheme makes use of the fact that metal can cross over the input poly gate without creating an electrical connection.

The transmission gate problem illustrates some of the interconnect routing problems that arise in layout. The logic diagram in Figure 3.35(a) shows a TG with an input x and an output y. Since a transmission gate has only two FETs, it is very simple to design at the physical level. The complicating factor is the inverter that takes the switching signal S and must produce $\bar{S}$ to drive the pFET side of the TG. The NOT gate must be connected to the power supply and ground, but the nFET and pFET of the TG may be located as needed. One solution is shown in Figure 3.35(b). This uses inverter FETs with a tall n+ region so that metal TG input line carrying x may be crossed over it. The complementary switching signal $\bar{S}$ is taken directly from the inverter and fed to the TG pFET.

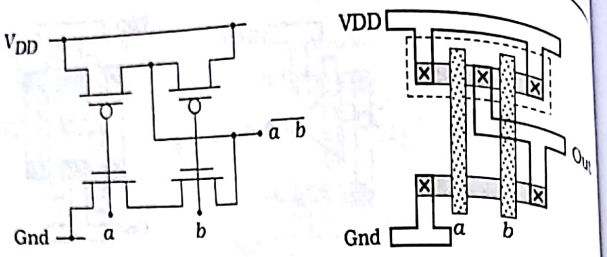
Once we have laid the foundation for simple layout, it may be used for



(a) Logic diagram (b) Layout

Figure 3.35 Layout of a transmission gate with a driver

[8] This is called a **poly contact** and defines an oxide cut.

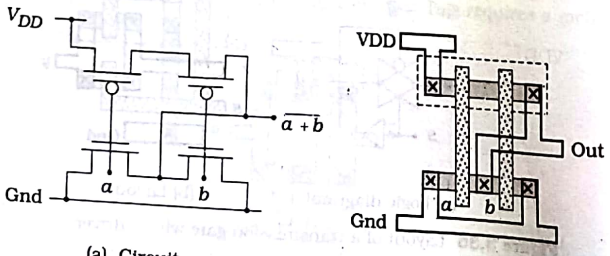


(a) Circuit (b) Layer design

Figure 3.36 NAND2 layout

more complex gates. Figure 3.36(a) shows a NAND2 circuit that has been drawn in a manner that leads to the patterning in Figure 3.36(b). The two nFETs are in series and can be laid out using the method shown in Figure 3.27. Since the gates (with inputs *a* and *b*) run in a vertical direction, the parallel-connected pFETs can be added using the technique introduced earlier in Figure 3.29 which achieves the parallel connection by Metal wiring. This allows us to maintain simple gate poly lines shown. The same approach may be used to construct the NOR2 gate. As shown in Figure 3.37(a), the FET arrangement is opposite with the nFETs in parallel and the pFETs in series. The resulting layout in Figure 3.37(b) follows the same philosophy as for the NAND2 gate wiring.

The similarity between the NAND2 and NOR2 layouts can be seen by decomposing the structures into transistors and wiring. The basic FET arrangement for both gates is shown in Figure 3.38(a). To obtain a NAND2 gate, we use the metal wiring pattern provided in Figure 3.38(b); the NOR2 gate is obtained using the wiring in Figure 3.38(c). If you take a moment to study the metal patterns for the two gates, you will see that they are identical! This can be verified by drawing an imaginary horizontal



(a) Circuit (b) Layer design

Figure 3.37 NOR2 gate design

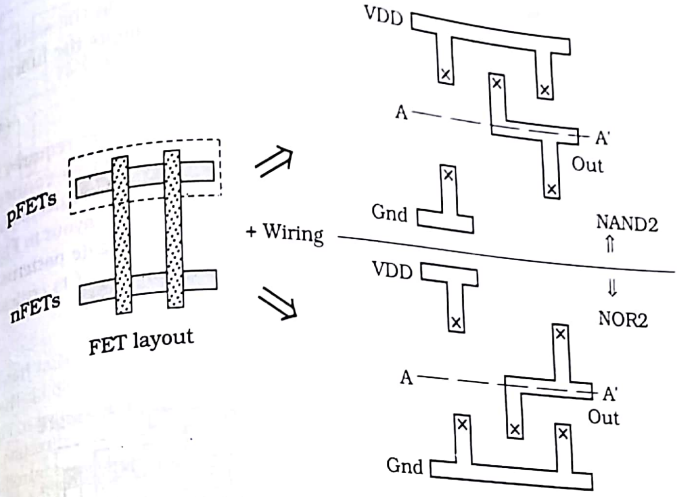
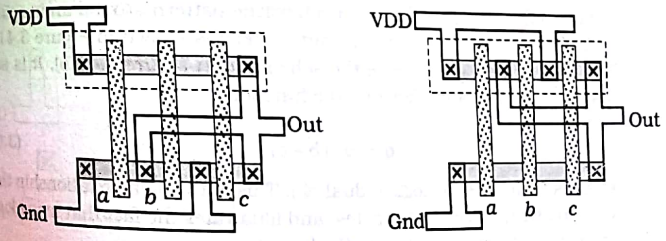


Figure 3.38 NAND2-NOR2 layout comparison

line through the center of one, and then rotating the pattern around it (i.e., flip it vertically). This illustrates how the AND-OR property of duality translates into a layout symmetry.

These layout techniques can be extended to gates with 3 or more inputs. A NOR3 gate is shown in Figure 3.39(a). This uses 3 series-connected pFETs and 3 parallel-connected nFETs. If we "flip" the metal pattern, then we obtain the NAND3 circuit in Figure 3.39(b). On paper, 4-input gates can also be designed in the same manner. However, the electrical switching time of 4-input NAND and NOR gates is relatively slow, which often precludes their usage.



(a) NOR3 (b) NAND3

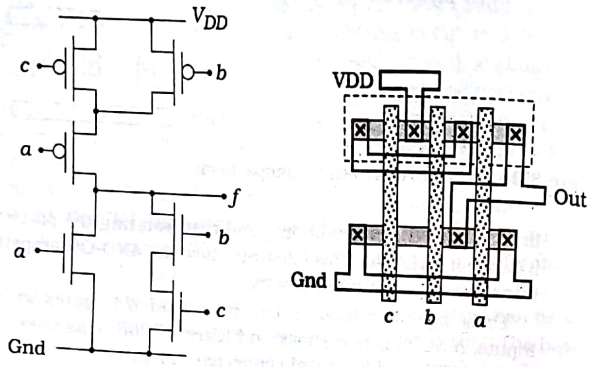
Figure 3.39 Layout for 3-input gates

3.4.2 Complex Logic Gates

The layout of complex logic gates can be accomplished in the same manner. Consider the circuit in Figure 3.40(a) that implements the function

$$f = \overline{a + b \cdot c} \tag{3.68}$$

as can be verified using the standard analysis. The circuit requires that an nFET be placed in parallel with a group of two series-connected nFETs. The pFET array consists of a group of two parallel-connected transistors that are wired in series with one other device. The layout in Figure 3.40(b) provides the correct wiring and uses single poly gate patterns for each input. Note, however, that the signal placement order is critical to obtaining the logic output.



(a) Circuit (b) Patterning

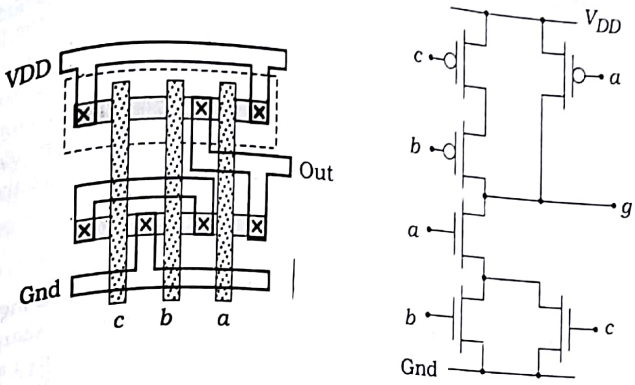
Figure 3.40 Extension of layout technique to a complex logic gate

An interesting variation of the layout demonstrates another important point. Suppose that we flip the metal wiring pattern around an imaginary horizontal line. The resulting layout pattern is shown in Figure 3.41(a). Tracing out the circuit yields the schematic in Figure 3.41(b). It is seen that the new circuit implements the function

$$g = \overline{a \cdot (b + c)} \tag{3.69}$$

which is seen to be the logical dual of f. This is the same relationship that we found for the NOR-NAND gates, and illustrates the fact that many logic symmetries directly translate to the layout.

Unfortunately, not all gate layouts are as simple as these examples. Many require much thought and may involve trial-and-error sketches to



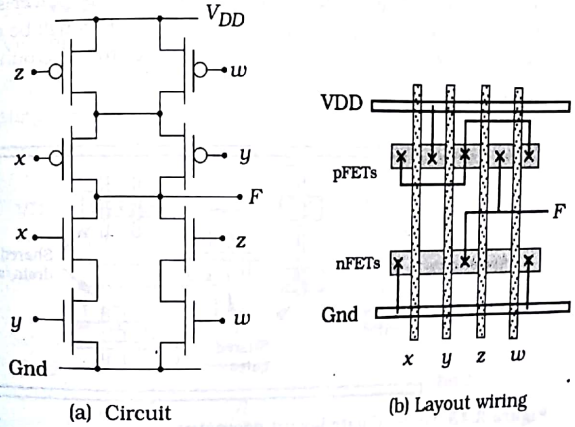
(a) Pattern (b) Circuit

Figure 3.41 Creation of the dual network

accomplish the finished design. Consider the general AOI expression

$$F = \overline{x \cdot y + z \cdot w} \tag{3.70}$$

that can be implemented using the circuit in Figure 3.42(a). If we want to maintain the layout strategy where we use a vertical-running poly line for each input, then we start with 4 gate lines with VDD and Gnd lines. To minimize the area, we would like to share n+ and p+ regions. The nFET patterning is easy since it consists of two groups in parallel, with each



(a) Circuit (b) Layout wiring

Figure 3.42 A general 4-input AOI gate

group containing 2 nFETs. These are shown in the layout of Figure 3.42(b). We have used thick lines to denote the metal wiring, since only the routing is important for the initial design. Once we place the nFETs, then the pFETs must be properly wired as required by the circuit. For this gate, the pFET wiring shown in the layout is a valid solution. Note that the circuit diagram shows the z and w pFETs touching the power supply while the layout uses the x and y group at VDD. The two provide the same switching characteristics between VDD and the output F, so the layout is acceptable as shown.

3.4.3 General Discussion

These examples illustrate some basic techniques for creating gate-level layouts. In the basic gates examined, it was possible to share n+ or p+ regions among several transistors, which reduces the area and wiring complexity. This is not always possible, especially in complicated arrangements. Various approaches to handling FET placement and wiring have been developed over the years, and are worth discussing here.

Consider the general problem of placing transistors into a CMOS circuit. Experience has shown that regular patterns and arrays will yield the best packing density, and randomly placed polygons should be avoided when possible. In general, every logic gate requires a power supply (VDD) and ground (VSS) connection, which will run as horizontal metal lines in our examples without loss of generality. This leads to the basic framework illustrated in Figure 3.43. All FETs are placed in between the two power rails. In the drawing, transistors are shown as individual devices, groups with shared gate poly lines, and groups with shared drain/source regions. The latter case is the most area-efficient placement, but it may not always be possible to link transistors. The drawing also shows that gate lines can run perpendicular or parallel to the power supply rails. Although not shown explicitly in the drawing, pFETs will be embedded in n-wells around VDD, while the nFETs are closer to the ground rail.

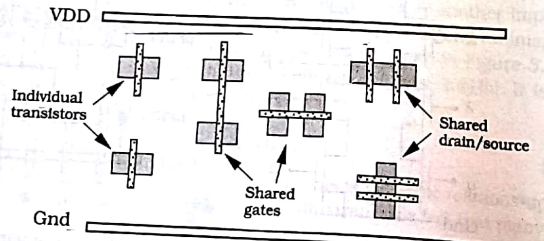


Figure 3.43 General gate layout geometry

One approach to layout is based on the concept of simple **stick diagrams** where each layer is represented by a distinct color, and the routing consists of colored lines that obey the rules of chip formation. A simple example of a stick diagram is shown in Figure 3.44. To save printing costs and keep the price of the book as low as possible, the drawing is monochrome and layers have been represented by different line features such as varying the linewidth or using a dashed line. The key is used to translate the lines into corresponding layers. The most commonly used colors for each layer are listed in the drawing. They are

- Polysilicon (gates): Red
- Doped n+/p+ (active): Green
- N-Well: Yellow (varies)
- Metal1: Blue
- Metal2: Grey (varies)
- Contacts: Black X's

Armed with a set of colored pencils, the layout designer can easily create and verify trial layouts for eventual transferal to silicon. Some of the simple rules associated with colored stick diagrams are as follows.

- A red line crossing a green line creates a transistor
- Red over green inside a yellow border region is a pFET; otherwise it is an nFET
- Red may cross blue or grey
- Blue may cross red, green, or gray
- Grey may cross red, green, or blue
- Transistor contacts must be placed from blue to green
- Vias must be specified to contact blue to grey
- A (poly) contact must be used to connect blue to red

This simple set of rules provides the basics of stick diagram layout. The

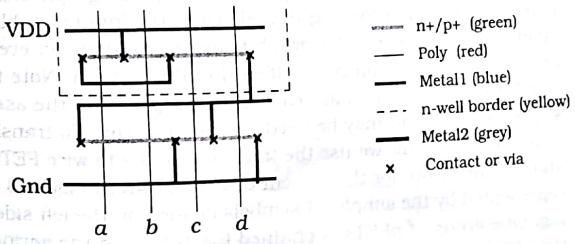


Figure 3.44 Basic stick layout diagram

accompanying CD provides a more detailed discussion of stick diagrams using a color on-screen presentation. Stick diagrams are often used to perform quick layouts or to study large complex routing problems.

A more structured technique is to apply graph theory to the problem of transistor placement and logic gate layout. Figure 3.45 defines the basic components of a graph element that represents a FET. In this approach, the drain and source nodes *x* and *y* of the transistor translate to connection nodes called **vertices**. The transistor itself is represented by an **edge** that corresponds to the signal flow path. Any CMOS circuit can be translated into an equivalent graph consisting of edges and vertices.

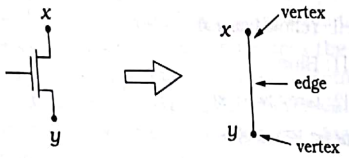
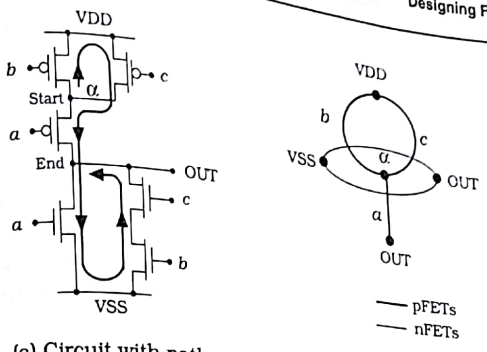


Figure 3.45 Representation of a FET in graph theory

Euler graphs aid in the placement and wiring of circuits where the transistors have shared drain/source regions. To construct an Euler graph, start with the CMOS circuit diagram and select a starting vertex (node). If it is possible to trace the entire graph without passing over an edge more than once, then it is possible to use common n+/p+ regions for nFETs/pFETs. The resulting graph can then be used directly to create the layout strategy.

An example of this process is shown in Figure 3.46. The circuit in Figure 3.46(a) can be traced as shown. The path starts at the vertex shown and follows the arrow to the end vertex, and passes over every edge only once; this defines an **Eulerian path**. Since the path exists, we can use it to construct the Euler graph in Figure 3.46(b). The graph consists of intersecting pFET and nFET graphs. The pFET graph links VDD to the node α, and then to the output node OUT; the input variables are used to label each edge. The nFET graph is drawn to intersect every pFET edge once; the resulting path specifies the nFET chain. Note that both the nFET and pFET graphs are closed; this represents the assertion that a single n+/p+ region may be used for each polarity. To translate the Euler graph to the layout, we use the transistor paths to wire FETs in the order shown. The layout for the present case is shown in Figure 3.47. FETs are represented by the simplified symbols defined on the left side of the drawing. One group of pFETs is chained together, and the wiring specified in the pFET part of the Euler graph is transferred to the drawing. Similarly,



(a) Circuit with path (b) Euler graph

Figure 3.46 Construction of an Euler graph

we use a common n+ region for the nFETs and follow the wiring in the nFET portion of the Euler graph. This may then be translated into the final layout.

If an Euler path cannot be found, then it means that it is not possible to use FET chains to build the circuit. Two or more groups of transistors will be needed, and the layout is much more complex. Design automation tools have been developed to help in some aspects of gate layout, but an experienced layout designer is still considered necessary in critical applications. Many layout specialists have backgrounds in graphics or art, and are able to produce amazingly compact designs that are not obvious to the rest of the group.

3.4.4 Summary

In this chapter we have seen the basics of translating FET logic circuits to silicon. The layout considerations presented here are sufficient to create complex logic networks using a set of standard MOSFETs as building blocks. In many designs it is possible to simply place reasonably sized transistors as specified by the layout, and wire them together. If done

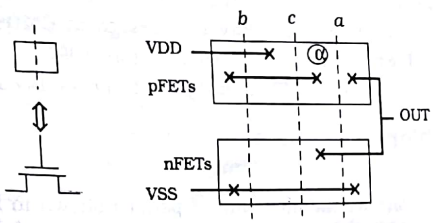


Figure 3.47 Layout using an Euler graph

correctly, it will result in a logically functional circuit. The switching speed of the circuit, however, may not be as fast as desired.

The key to high-speed VLSI design is to create switching networks that perform the required operations as fast as possible. This means that we must start to be concerned about signal delays that are induced by transistor switching times and parasitic resistance and capacitance elements. This takes us into the unique world of the VLSI designer. We are not content to simply obtain a functional network; it must also be fast!

The details of the CMOS fabrication process and how it affects the electrical performance provide the missing link between the discussion in this chapter and high-performance system design. The concepts introduced in the next few chapters reinforce and expand on the material here. The relationship among CMOS logic circuits, layout, transistors, and systems design is quite natural.

3.5 References for Further Reading

[1] H. B. Bakoglu, **Circuits, Interconnections, and Packaging for VLSI**, Addison-Wesley, Reading, MA, 1990.

[2] Dan Clein, **CMOS IC Layout**, Newnes, Woburn, MA, 2000.

[3] Richard S. Muller and Theodore I. Kamins, **Device Electronics for Integrated Circuits**, 2nd. ed., John Wiley & Sons, New York, 1986.

[4] Robert F. Pierret, **Semiconductor Device Fundamentals**, Addison-Wesley, Reading, MA, 1996.

[5] Bryan Preas and Michael Lorenzetti (eds.), **Physical Design Automation of VLSI Systems**, Benjamin/Cummings Publishing Company, Menlo Park, CA, 1988.

[6] M. Sarrafzadeh and C. K. Wong, **An Introduction to VLSI Physical Design**, McGraw-Hill, New York, 1996.

[7] Naveed Sherwani, **Algorithms for VLSI Physical Design Automation**, Kluwer Academic Publishers, Norwell, MA, 1993.

[8] Jasprit Singh, **Semiconductor Devices**, John Wiley & Sons, New York, 2001.

[9] Ben G. Streetman and Sanhay Banerjee, **Solid State Electronic Devices**, 5th ed., Prentice Hall, Upper Saddle River, NJ, 1998.

[10] John P. Uyemura, **Physical Design of CMOS Integrated Circuits Using L-Edit™**, PWS Publishers, Boston, 1995.

[11] M. Michael Vai, **VLSI Design**, CRC Press, Boca Raton, FL, 2001.

3.6 Problems

[3.1] Consider the interconnect pattern shown in Figure P3.1. The line has a width of 1 unit, and the sheet resistance is $R_s = 25\ \Omega$. Find the resistance from A to B if each corner square contributes a factor of 0.625 of a "straight-path" square.

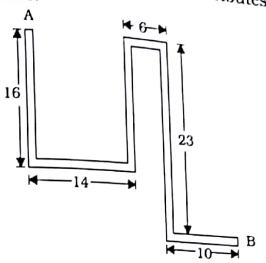


Figure P3.1

[3.2] An interconnect line can be made in either of two layers. If a gate polysilicon layer is selected, the sheet resistance is 25 Ω; for this case, the interconnect will have a width of 0.5 μm and a length of 27.5 μm. A metal layer can also be used. It has a sheet resistance of 0.08 Ω. The metal line has a width of 0.8 μm but requires a different routing length of 32.4 μm.

Calculate the line resistance R_{line} for each case and determine the lower resistance alternate. What is the percentage increase in resistance if the larger resistance line is used instead?

[3.3] An interconnect line is made from a material that has a resistivity of $\rho = 4\ \mu\Omega$-cm. The interconnect is 1200 Å thick, where 1 Angstrom (Å) is 10^{-8} cm. The line has a width of 0.6 μm.

(a) Calculate the sheet resistance R_s of the line.

(b) Find the line resistance for a line that is 125 μm long.

[3.4] Consider equation (3.14) for the interconnect time constant τ. Prove that τ has units of seconds by expressing ohms and farads in fundamental MKS units and reducing.

[3.5] An interconnect line runs over an insulating oxide layer that is 10,000 Å thick. The line has a width of 0.5 μm and is 40 μm long. The sheet resistance is known to be 25 Ω.

(a) Find the line resistance R_{line}.

(b) Find the line capacitance C_{line}. Use $\varepsilon_{ox} = 3.453 \times 10^{-13}$ F/cm, and express your answer in femtofarads (fF) where $1\ \text{fF} = 10^{-15}$ F.

(c) Find the time constant τ for the line in units of picoseconds (ps) where $1\ \text{ps} = 10^{-12}$ sec.

[3.6] A sample of silicon is doped with arsenic with $N_d = 4 \times 10^{17}\ \text{cm}^{-3}$.

(a) Find the majority carrier density.

(b) Find the minority carrier density.

(c) Calculate the electron and hole mobilities and then find the conductivity of the sample.

[3.7] A region of silicon is doped with both phosphorus and boron. The doping is $N_d = 2 \times 10^{16}$ cm^{-3} while the B-doping level is $N_a = 6 \times 10^{1}$. Determine the polarity (n or p) of the region, and find the carrier densities.

[3.8] A sample of silicon is doped with boron atoms at an acceptor density of $N_a = 4 \times 10^{14}$ cm^{-3}.

(a) Find the majority and minority carrier densities.

(b) Find the resistivity ρ of the sample.

(c) Suppose that the region has dimensions of 2 μm × 0.5 μm × 100 Find the largest resistance of an end-to-end block of the region.

[3.9] Consider a doped semiconductor where

$$\sigma = q(\mu_n n + \mu_p p) \tag{3}$$

and $np = n_i^2$. Suppose we wish to minimize the conductivity.

(a) Use the mass-action law to write in terms of p only.

(b) Compute the derivative $(d\sigma/dp)$ and set it equal to 0 to find the concentration that minimizes σ.

(c) Noting that $\mu_n > \mu_p$, what polarity (n-type or p-type) is required the highest resistivity? Then use your equations to find the doping and density that give the highest resistivity.

[3.10] An n-channel MOSFET has a mobility value of $\mu_n = 560$ cm^2/V- and uses a gate oxide with a thickness of $t_{ox} = 90$ Å. The gate voltage given as $V_G = 2.5$ V, and the threshold voltage is 0.65 V.

(a) Calculate the value of C_{ox} in units of F/cm^2.

(b) Find the process transconductance k'_n.

(c) Find the device transconductance β_n if the FET has a chann length of 0.25 μm and a channel width of 2 μm.

[3.11] Use equation (3.57) for R_n to find the units of the electron mobili μ_n. Then suppose that $\mu_n = 500$ cm^2/V-sec and $(V_G - V_{Tn}) = (3.3 - 0.7)$ V known.

(a) Find the nFET resistance if $W = 10$ μm, $L = 0.5$ μm, and $t_{ox} = 10$ nm

(b) Find R_n if the channel width is increased to a value of $W = 22$ μ while the channel length remains the same.

[3.12] A pFET is described by $\mu_p = 220$ cm^2/V-sec and $(V_G - |V_{Tp}|) = (3.3 - 0.8)$ V, $W = 14$ μm, $L = 0.5$ μm, and $t_{ox} = 11.5$ nm. Find the pFET resistance R_p of the device.

[3.13] Consider a process that has an oxide thickness of $t_{ox} = 9.5$ nm. The particle mobilities are given as $\mu_n = 540$ and $\mu_p = 220$ cm^2/V-sec. An nFET and a pFET are made, both with $W = 12$ μm, $L = 0.35$ μm. Both have gate voltages of $V_G = 3.3$ V, while the threshold voltages are $V_{Tn} = 0.65$ V and $V_{Tp} = -0.74$ V.

(a) Find the values of R_n and R_p for the two transistors.

(b) Suppose that we want to keep the nFET the same size, but increase

the width of the pFET to the point where $R_p = 0.8\ R_n$. Find the required width of the pFET.

[3.14] Design a CMOS logic gate that provides the function

$$Out = \overline{x \cdot (y \cdot z + z \cdot w)} \tag{3.72}$$

Then perform the basic layout of circuit.

[3.15] Design the circuit and layout for a CMOS gate that implements the function

$$F = \overline{a \cdot b \cdot c + a \cdot d} \tag{3.73}$$

using the fewest number of transistors and a compact layout style.

[3.16] Consider the OAI logic function

$$g = \overline{(a + b) \cdot (c + d) \cdot e} \tag{3.74}$$

Design the CMOS logic gate and then construct a basic layout for the circuit.

[3.17] Expand the function g given in equation (3.74) [Problem 3.16 above] into AOI form. Then design the CMOS logic circuit and layout.

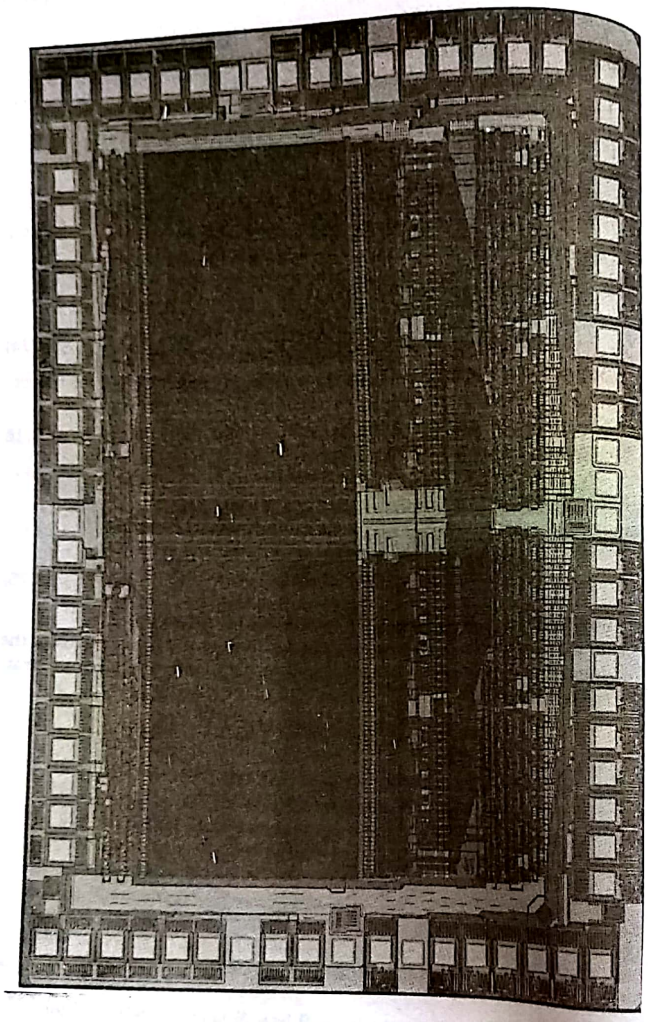
[3.18] Examine the stick diagram in Figure 3.44. Is this a functional logic gate? If so, determine the logic operation it provides.

[3.19] Consider the logic function

$$g = \overline{a \cdot b \cdot c + d} \tag{3.75}$$

(a) Design the CMOS logic gate that provides this function.

(b) Is it possible to find an Euler graph for the circuit? If so, construct the graph and use it to perform a stick-level layout. If not, find a layout strategy for the gate.



Fabrication of CMOS Integrated Circuits

4

An integrated circuit consists of several patterned layers of materials that are used to form transistors and provide electrical interconnections for the circuit. In a modern process, the minimum feature size is less than about 0.12 μm, which allows for a tremendous packing density. Individual chips with more than 100 million FETs are becoming commonplace. The techniques needed to fabricate silicon chips of this sophistication have been developed over several decades at an enormous cost. In fact, silicon has been characterized as being the most studied element on earth!

Now that we have an understanding of the physical structure of CMOS integrated circuits we may progress to studying how the circuits are fabricated in the manufacturing process. Our treatment will focus on those aspects of silicon chip fabrication that are important to VLSI design.

4.1 Overview of Silicon Processing

Silicon integrated circuits are created on larger circular sheets of silicon called wafers. They are typically 100–300 mm in diameter, and about 0.4–0.7 mm thick. A large silicon circuit is about 1 cm on a side so that many individual circuits can be made on a single wafer. The location of a circuit is called a die site, with the number of sites per wafer depending upon the size of each site and relative to the overall surface area of the wafer. Figure 4.1 portrays a wafer with individual sites. The flat is used as a reference plane to form an imaginary grid that is used to place the individual sites. Some wafers will have additional flats that are coded to provide information about the crystal orientation at a glance.

Starting with a bare polished surface, the wafer is subjected to thousands of individual steps in the manufacturing processes. The most

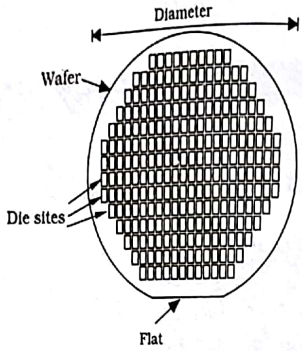


Figure 4.1 Silicon wafer showing die sites

important steps in the sequence are for creating and patterning the layers of materials needed in the CMOS structures. Most of the remaining steps are cleaning and rinsing of the wafer. The manufacturing capacity of a chip factory is usually measured by the number of **wafer starts** per week, i.e., how many fresh wafers are introduced into the fabrication sequence. Wafers are processed in groups, and it takes several weeks for a lot to make it through the entire processing line.

Unfortunately, not every site on the wafer turns out to be a functional circuit. This is due to many factors that may arise in the manufacturing line that are inherent due to the complexity of the processing sequence. To describe this problem, we introduce the concept of the fabrication **yield** Y such that

$$Y = \frac{N_G}{N_T} \times 100\% \tag{4.1}$$

where N_G is the number of good (functional) sites and N_T is the total number of sites. A yield of $Y = 85\%$ means that 85% of the chips operate as they should and can be sold to customers. High yield values are, of course, desirable to help insure the economic stability of the company. However, yield enhancement is a complex problem that requires countless hours of thinking and experimenting with the process line.

Yield analysis is based on predicting the yield of a particular process, and requires a thorough understanding of all aspects of the silicon processing sequence. One working in this area is faced with increasing the value of Y for a given design. One variable that is critically important to increasing the yield is the area A_{die} of the die. The number of total die sites N_T on a wafer of diameter d is estimated from

$$N_T = \pi \frac{(d - d_e)^2}{4A_{die}} \tag{4.2}$$

where d_e is the wasted edge distance that arises from placing rectangular sites onto a round wafer. Empirical analysis shows that large area die are plagued by smaller yields. A simple expression that describes this is

$$Y = e^{-\sqrt{DA}} \tag{4.3}$$

where A is the area of the die. The parameter D is the **defect density** in units of cm^{-2} and is the average number of defects per cm^2 on the wafer. D represents the "limit of perfection" that can be expected of the silicon wafer; this is due to the fact that every crystal wafer has random imperfections that cannot be eliminated. In modern technology, $D = 1\ cm^{-2}$ is a reasonable value for the defect density limit.

Some physical defects tend to occur in clusters on the wafer. An estimate for the yield when these dominate is given by

$$Y = \left(1 - \frac{A_{die}D}{c}\right)^c \tag{4.4}$$

where c is an empirical parameter that accounts for the clusters. This idea has been used to a write a binomial equation of the form

$$Y = \frac{1}{\left(1 + \frac{A_{die}D}{c}\right)^2} \tag{4.5}$$

Alternately, when several die fail in a large area A_{fail} of the wafer, then the yield has been approximated using the expression

$$Y \approx (1 - g)e^{-A_{die}D} \tag{4.6}$$

where

$$g = \frac{A_{fail}}{A_{wafer}} \tag{4.7}$$

is the fractional area where the defects exist.

Yield analysis is a very specialized aspect of VLSI manufacturing. Persons working in the area tend to have strong backgrounds in physics, general and physical chemistry, mathematics, statistics, or engineering (chemical, materials, or electrical), and groups work closely with the manufacturing line and the wafer analysis groups to maximize yield values. It is not possible to solve a problem until it is discovered and defined. The "design of experiments" that can pinpoint problems and lead to solutions becomes very critical.

Economics 101

It is worth examining some important economic factors that deal with the

design, manufacture, and marketing of VLSI circuits. Let C_{chip} be the cost of manufacturing a chip, and C_{sell} the selling price. The profit per chip is then given by

$$\text{Profit} = C_{sell} - C_{chip}$$

To survive, a product line must result in a value where

$$\text{Profit} > 0$$

While this may seem blatantly obvious, the VLSI designer must recognize that neither C_{chip} nor C_{sell} is easy to compute. The cost of manufacturing the chip includes the materials and salaries of all personnel (design, manufacturing, testing, etc.) plus overhead (electricity, water, taxes, ...). Increasing the yield reduces the overall costs per unit, so the importance of yield analysis becomes obvious. These factors and many more contribute to C_{chip}.

In modern VLSI, the cost of a state-of-the art chip manufacturing facility is somewhere between \$1–3 billion (USD). This includes the land, building, equipment, and start-up costs, but not materials or everyday operations. The cost of the facility must be amortized over the product lifetime of the plant.

The selling price C_{sell} of every product must include all direct and indirect costs, plus a fraction of the plant debt. The laws of supply and demand also enter the picture: C_{sell} must be at a level that the customers are willing to pay. If a product is in great demand, then C_{sell} may be set above the costs and the design produces a large income. In this case chips (and products in general) may be sold at a price that is determined entirely by demand; a common phrase that describes this is *whatever the market will bear*. On the other hand, even a great engineering design may fail to gain a following of users, and it will be eventually withdrawn; in this case we encounter the unwanted result that Profit < 0.

Another complication is that C_{sell} tends to decrease in time. Even the "hottest" new microprocessor eventually becomes a cheap bargain basement item. This is not a major problem so long as we have repaid the investment made in engineering costs. Complex VLSI chips are very difficult to design, and the original design can be very expensive. Another helpful factor is that as time progresses,

$$C_{chip} \rightarrow C_{materials} \tag{4.1}$$

where $C_{materials}$ is the cost of materials. Silicon has the advantage of being very cheap, especially when compared to alternates such as gallium arsenide (GaAs). Keeping a product line active for many years greatly enhances the profitability.

This short introduction is designed to help the aspiring VLSI designer understand the overall structure of the industry. Producing a silicon chip

with 100 million transistors is much more complicated than starting a "dot-com" web site. It requires financial backing, strong technological support, innovative engineering, and a reliable sales force. The fabrication process is considered to be a major expense in the Profit equation, so we have chosen to study it in some detail in this chapter. Design and engineering costs are almost as high, and are discussed in the rest of the book.

4.1.1 Outline of the Chapter

We have introduced the viewpoint that a silicon chip is a set of patterned material layers. When the layers are properly stacked, the resulting three-dimensional structures are controlled switches (transistors) that are wired together to implement logic operations.

In this chapter, we first examine the most important material layers that are used in silicon processing. This includes oxides, doped silicon regions, and metals. A few chemical reactions are presented along with a brief description of how the layer is actually grown or deposited. We then move on and study how a layer is physically patterned to have the proper shape and size needed for wire and transistors. This allows us to progress to the steps used to fabricate a basic CMOS circuit.

The main objective of the treatment is to provide an understanding of the basics and how they relate to the physical design of a VLSI circuit. Many persons with strong science backgrounds become fascinated with silicon processing, and establish an entire career in the field.

4.2 Material Growth and Deposition

An integrated circuit is created by stacking layers of various materials in a prespecified sequence. Both the electrical properties of the material and the geometrical patterns of the layer are important in establishing the characteristics of devices and networks.

Most layers are created first, and then patterned using the lithographic sequence described in the next section. Doped silicon layers are the exception to this rule, as they are created with the desired shapes by using the lithographic process to define where the dopants can enter the silicon. In this section we will examine some basic processing steps used in silicon VLSI processing.

4.2.1 Silicon Dioxide

Silicon dioxide (SiO_2) is a critically important material in IC processing because

- It is an excellent electrical insulator
- It adheres well to most materials
- It can be "grown" on a silicon wafer or deposited on top of the wafer

SiO_2 is generically known as quartz glass, or simply "glass," and is used for the gate oxide in a MOSFET, in addition to numerous other applications.

There are two types of SiO_2 layers found in VLSI circuits, with the distinction being how they are created. A **thermal oxide** is formed by the reaction

$$Si + O_2 \rightarrow SiO_2 \tag{4.11}$$

using heat as a catalyst. The unique aspect of a thermal oxide is that the silicon (Si) required for the reaction is obtained from the silicon wafer itself. This is illustrated in Figure 4.2(a) where oxygen molecules O_2 are passed over the surface of the wafer where the reaction takes place. This literally "grows" the glass layer with the results shown in Figure 4.2(b). The final thickness of the oxide is denoted as x_{ox} in the drawing, and depends on the temperature, crystal orientation, and growth time. Since silicon atoms from the surface of the wafer are used by the reaction, a layer of silicon with a thickness

$$x_{Si} \approx 0.46\ x_{ox} \tag{4.12}$$

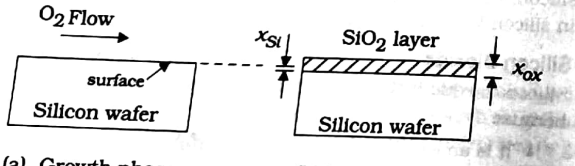
is consumed. An equivalent (and useful) viewpoint is that the surface of the silicon is "recessed" from its original location.

Although pure oxygen yields high-quality oxide layers, it is relatively slow. A faster growth rate is obtained using water (H_2O) in the form of steam via the reaction

$$Si + 2H_2O \rightarrow SiO_2 + 2H_2 \tag{4.13}$$

which is called "wet oxidation." In practice, mixtures of O_2 and steam are used, along with nitrogen as a carrier gas and other chemicals such as chlorine (Cl).

Thermal oxide is a form of a **native oxide**, i.e., one that is created when the surface is exposed to an oxygenated atmosphere. If you take a bare silicon wafer and place it in air, a thin native oxide layer will form. Increasing the temperature enhances the growth rate. Silicon oxidation temperatures are typically in the range of about 850–1100 °C.



(a) Growth phase (b) Final structure

Figure 4.2 Thermal oxide growth

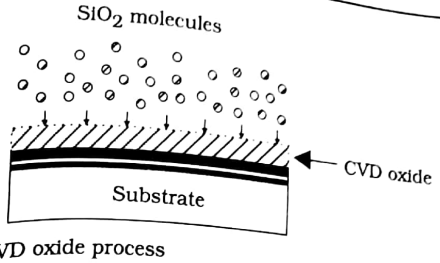


Figure 4.3 CVD oxide process

Most oxide layers in VLSI circuits are well above the wafer surface and no silicon is available for thermal oxide growth. In this case, we create SiO_2 molecules using gaseous reactions, and then **deposit** them onto the surface to provide an oxide coating. The process is shown schematically in Figure 4.3. A chemical reaction using silane (SiH_4) such as

$$SiH_4\ (gas) + 2O_2\ (gas) \rightarrow SiO_2(solid) + 2H_2O\ (gas) \tag{4.14}$$

can be used to produce the SiO_2 molecules above the wafer. This technique is called **chemical vapor deposition** (CVD) and the resulting layers are often called **CVD oxides**. The thickness of the oxide layer is controlled using the growth rate and deposition time. It is possible to perform the deposition at low temperatures, giving rise to the name **LTO** (low-temperature oxides). Also, it is sometimes advantageous to dope the glass. For example, phosphorus doping yields "P-doped glass" which helps certain types of planarization steps.

4.2.2 Silicon Nitride

Another useful material is silicon nitride Si_3N_4, which is often just called "nitride" when the context is clear. The reaction

$$3SiH_4(gas) + 4NH_3(gas) \rightarrow Si_3N_4\ (solid) + 12H_2(gas) \tag{4.15}$$

illustrates one technique. Nitrides are unique in that they act as strong barriers to most atoms. This makes them ideal for use as an **overglass** layer, which is a final protective coating on a chip, since it keeps contaminants from reaching the sensitive silicon circuits. Silicon nitride is used in a fabrication sequence that electrically isolates adjacent FETs (as will be discussed later). And, they have a relatively high dielectric constant $\varepsilon_N \approx 7.8\ \varepsilon_0$, which makes them candidates for insulating ON (oxide-nitride) "sandwich" insulators in various capacitor structures such as those used in DRAM (dynamic random-access memory) cells.

4.2.3 Polycrystal Silicon

If we deposit silicon atoms on top of an amorphous SiO_2 layer, the silicon attempts to crystallize but can't find a crystal structure for reference. This

results in the formation of small **crystallites**, which are small regions of silicon crystal. The material is then called **polycrystal silicon** or **polysilicon**, or just **poly** for short. Polysilicon is universally used as the gate material in FETs. It has the desirable characteristics that it can be doped, it adheres well to silicon dioxide, and it can be "coated" with a high-melting-temperature (refractory) metal such as Ti or Pt to reduce the sheet resistance. Poly provides an excellent basis for building MOSFETs in CMOS integrated circuits.

A basic reaction using silane is

$$SiH_4 \rightarrow Si + 2H_2 \tag{4.16}$$

which is performed at a temperature around 500–600° C. Poly deposition techniques have evolved during recent years in the fabrication of **stacked capacitors** used in advanced dynamic random-access memory (DRAM) cells. These are examined in Section 13.3 of Chapter 13.

4.2.4 Metals

Aluminum (Al) is the most common metal used for interconnect wiring in integrated circuits. It can be evaporated by heating in a vacuum chamber with the resulting flux used to coat the wafer. Al has good adhesion characteristics and is easy to pattern. Its popularity is understandable.

Aluminum has a bulk resistivity of about $\rho = 2.65\ \mu\Omega$-cm. An aluminum interconnect line that is 0.1 μm thick has a sheet resistance of about

$$R_s = \frac{\rho}{t} = \frac{2.56\times10^{-6}}{10^{-5}} = 0.265\Omega \tag{4.17}$$

However, aluminum exhibits a problem called **electromigration**. High current flow densities tend to literally move atoms from one end of an interconnect line, creating pits called **voids**. The atoms pile up at the other end in microscopic structures called **hillocks**. These are illustrated schematically in Figure 4.4. Hillocks and voids can lead to failure and

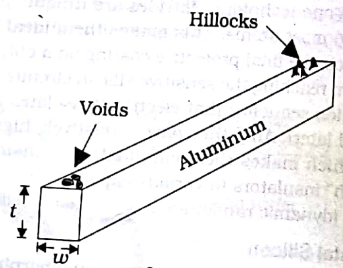


Figure 4.4 Visualization of electromigration effects in alum[illegible]n

much research has been devoted toward studying this problem. A common solution is to mix copper with the aluminum during the metal deposition step. This reduces electromigration effects, but increases the resistivity to values around $\rho = 3.5\ \mu\Omega$-cm. The sheet resistance is increased proportionately.

At the physical design level, we avoid excessive formation of hillocks and voids by controlling the current density J A/cm^2 flowing in the interconnect. For an interconnect line with thickness t and width w, the current density is given by

$$J = \frac{I}{A} \tag{4.18}$$

where I is the current in amperes, and $A = wt$ is the cross-sectional area in units of square centimeters. Layout designers cannot alter the thickness t of the layer since it is established in the processing line. Electromigration is thus controlled by specifying the minimum linewidth w needed to keep J below a maximum value J_{max}. This is our first example of a layout **design rule** that specifies a minimum dimension of a feature for a particular situation. We will investigate design rules more thoroughly in the later sections of this chapter.

MOS had its beginnings in metal-gate technology where the "M" truly stood for metal, and aluminum was the choice for the gate layer. The drawback of using Al for a transistor gate is that its low melting temperature prohibits the use of high-temperature processing steps once it is deposited on the wafer. As processing technology continued to improve with increasingly complex processing sequences, this became a limiting factor. Transistors using polysilicon gates were developed and are now standard in CMOS. A significant problem with silicon gates is that even heavily doped poly has a high sheet resistance with values around R_s = 25–50 ohms. To overcome this, the poly is coated with a thin layer of a **refractory** (high-temperature) metal such as titanium (Ti), tungsten (W), or platinum (Pt). This combination is called a **silicide** and the poly-metal mixture is usually treated as a single layer in the design. This will be shown explicitly in the CMOS processing sequence described later. Tungsten is also commonly used for plugs in vias to connect metal layers.

Copper (Cu) has recently been introduced as a replacement to aluminum. Since its resistivity is about one-half the value of Al, it gives smaller sheet resistances. At the device level, the difference is not important. However, the reduction in sheet resistance is significant when copper is used for long, system-level interconnect lines. The improvement in technology does not come easily. Standard patterning techniques cannot be used on copper layers; specialized techniques had to be developed. The use of copper will be discussed in Section 4.4.1.

4.2.5 Doped Silicon Layers

The silicon wafer is the starting point for the CMOS fabrication process. It is defined to be n-type or p-type during the crystal growth and acts as the basis substrate for the entire circuit structure. By our definition, a doped silicon layer is a patterned n- or p-type section of the wafer surface. Even though silicon layers don't always "stack" in the usual sense, we will maintain this terminology to be consistent.

The key to creating doped layers in the substrate is to introduce donor or acceptor atoms into the wafer that can be eventually incorporated into the silicon crystal. In modern CMOS, this is accomplished by a technique called **ion implantation** where the atoms are first ionized in a chamber, then accelerated to high energies in a particle accelerator. The beam is passed through a mass separation unit that selects the desired charge species using a magnetic field. The overall system is shown in Figure 4.5.

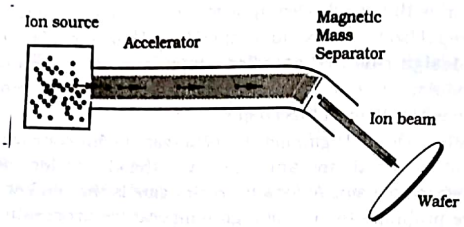


Figure 4.5 Basic sections of an ion implanter

The fast moving ions are literally smashed into the substrate at typical energies around 100–200 keV. The ions come to rest after several collisions with electrons and nuclei in the silicon wafer. This is illustrated schematically in Figure 4.6. The slowing mechanism damages the crystal

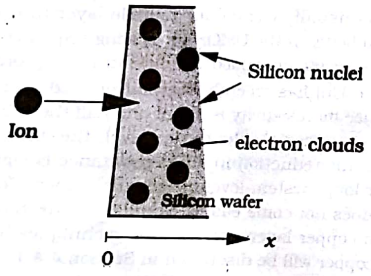


Figure 4.6 The ion stopping process

and leaves the dopants in random locations. To heal the crystal and set the dopants into proper locations within the crystal lattice, the wafer is heated in an **anneal** step. The dopants redistribute a little during the annealing step because of a process known as **particle diffusion**; diffusion is simply the collective heat-induced motion of particles that are concentrated in a small region that makes the particle spread out.

The ion distribution into the silicon can be approximated to first order using the Gaussian form

$$N_{ion}(x) = N_p e^{-\frac{1}{2}\left(\frac{x-R_p}{\Delta R_p}\right)^2} \tag{4.19}$$

with units of cm^{-3}; the surface of the wafer is defined by $x = 0$. This function is shown in Figure 4.7. The quantity R_p is called the **projected range**, and is the average depth of an implanted ion. The value of R_p depends on the incident energy, the species, and the crystal orientation, and can range from about 0.1 μm to as deep as 1 μm. The peak density N_p occurs at $x = R_p$. The standard deviation is denoted as the **straggle** ΔR_p; this represents the variation in the stopping depth of the individual ions due to the statistical nature of the energy loss process. More accurate models of the implant profile employ Pearson Type IV distributions and numerical simulations.

The number of implanted ions is usually described by the implant dose D_I defined by

$$D_I = \int_{\text{All x}} N_{ion}(x)dx \tag{4.20}$$

which has units of ions per cm^2 (or just cm^{-2}). This can be very accurately measured using charge counters. The dose is often used when analyzing the macroscopic electrical characteristics of MOS capacitors.

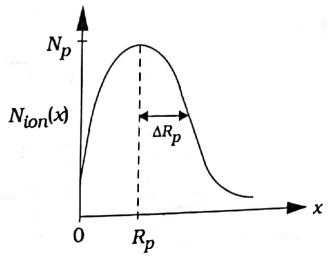


Figure 4.7 Gaussian implant profile

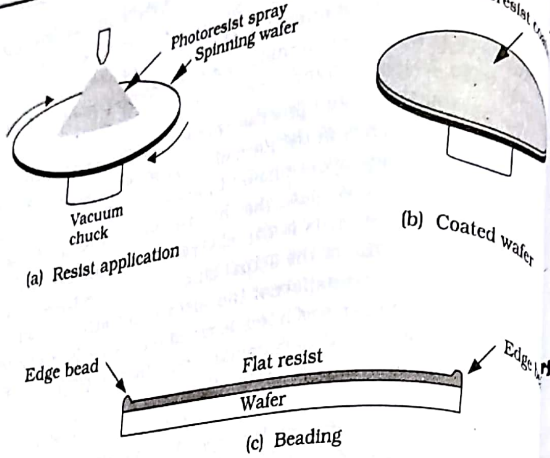


Figure 4.10 Photoresist application

where the regions that are shielded from the light are hardened
development process, while regions that were exposed to the ligh
rinsed away. The characteristics of a positive resist are shown in F
4.12. The exposure step in Figure 4.12(a) defines the light and d
regions in the reticle shadow. After the resist is developed, hardened
ers remain in the regions that were shielded from the light; this is il
trated in Figure 4.12(b). Negative photoresist has opposite characteris
illuminated regions harden while shielded regions are soluble and
rinsed away.

The hardened resist layer is used to protect underlying regions fr

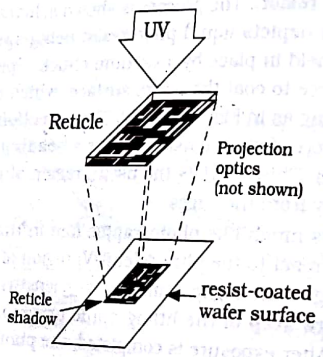


Figure 4.11 Exposure step

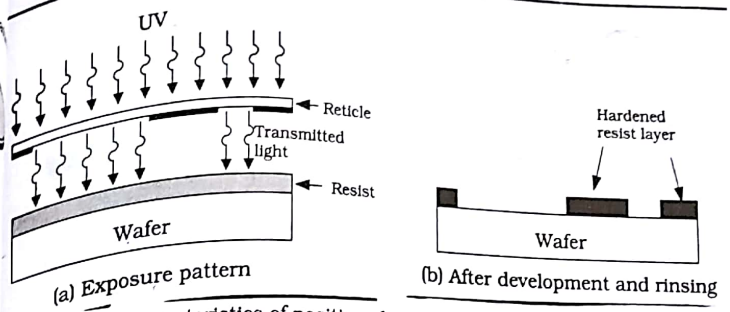


Figure 4.12 Characteristics of positive photoresist

the **etching** process. This is where the surface of the wafer is subjected to a gaseous plasma that is formed from an inert gas such as argon (Ar) and has reactant chemicals in it; overall, this is called a **reactive-ion etch** (RIE). The chemicals and plasma are chosen to attack and remove the material layer not shielded by the hardened photoresist. The resist itself can withstand the etchant mixture for the duration of the process. An example is shown in Figure 4.13. In Figure 4.13(a), a resist pattern is created on top of an oxide layer. The etching step removes oxide in the unprotected regions, so that the oxide has the same pattern as the resist; this is illustrated in Figure 4.13(b). This technique can be used to pattern any material layer above the wafer surface, including polysilicon, CVD oxides, and metals.[1] It allows us to transfer patterns from a computer layout design to the physical silicon level, thus creating the physical implementation of a logic network.

Doped silicon regions are also patterned using the lithographic process but the sequence is different. In this case, we grow an oxide layer on the wafer and then use lithography to etch down to the silicon surface; this is identical to the cross-section that was shown as Figure 4.13(b). The

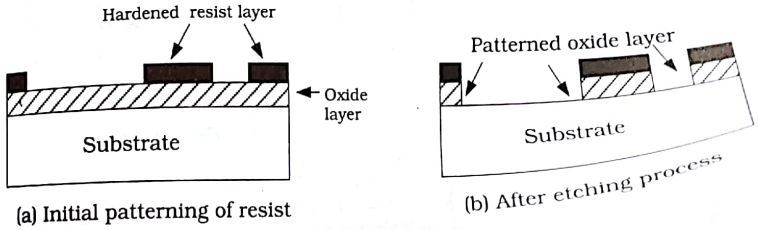
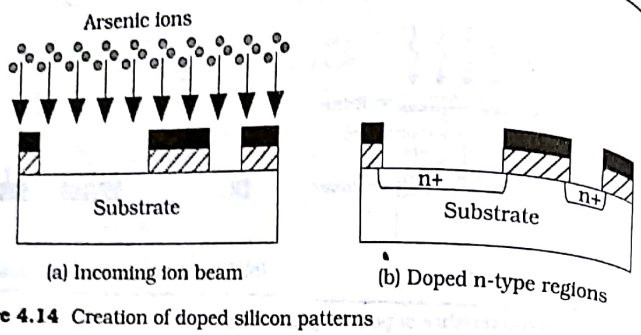


Figure 4.13 Etching of an oxide layer

[1] Copper is an exception as it is patterned using a different tech

ove the
This is called a



(a) Incoming ion beam (b) Doped n-type regions

Figure 4.14 Creation of doped silicon patterns

resist-oxide layers are then used to shield the silicon from an ion implantation step. Figure 4.14(a) shows that an incoming beam of arsenic ions covers the entire surface, but the dopants can enter the silicon only where the oxide has been etched away. The resulting n+ regions are thus defined by the oxide openings. Note that the widths of the n+ patterns are slightly larger than the oxide openings. This is due to an effect called **lateral doping** that arises from dopant diffusion during the annealing step. Lateral effects can limit the resolution of a narrow-line printing system.

Although we have shown only a single pattern in our examples, the manufacturing processes use larger wafers that accommodate many individual chip sites. Each site is individually exposed using a **step-and-repeat process**; a **wafer stepper** is an apparatus that holds the wafer and allows accurate movement to align the optics to each site, one at a time. After a site is exposed, the mechanism "steps" the wafer to the next site. This sequence produces a wafer with a large number of identical sites as illustrated in Figure 4.15. The **test site** locations contain various

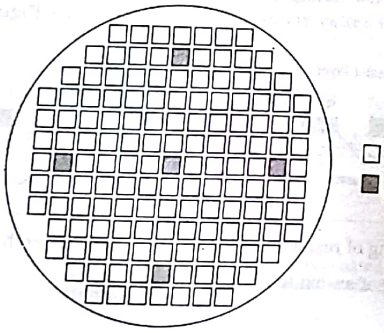


Figure 4.15 Wafer sites

test structures and circuits, such as MOS capacitors, doped regions of silicon, MOSFETs, and simple circuits. These are included to allow the wafer to be electrically tested during various phases of the manufacturing sequence. **Wafer probes** are sets of very small metallic probes that can contact regions on the wafer to allow these tests. The readings provide information on how well the manufacturing flow is progressing and also give critical data on electrical parameters needed for circuit design. It is important to include several test sites that are representative of all regions of the wafer, as nonuniform temperatures, gas flow densities, and other parameters vary across the wafer and affect the electrical characteristics.

The lithographic sequence is repeated for every masking step needed to build the integrated circuit. It is important to note that the first masking step defines the basic outline of the chip patterns; subsequent masking steps must pattern layers that have correct spacing relative to the features already created on the substrate. Correct alignment of a mask with the patterns on other masks is critical to the yield. Mask misalignment can cause the entire chip to be nonfunctional. Accurate alignment is achieved using **registration targets**, which are geometrical patterns that are created on a base layer solely to help align later masking steps. As the layers build, more sets of registration marks are required.

4.3.1 Clean Rooms

The lithographic process is very sensitive to dust particles. If a speck of dust lands on the photoresist, it will interfere with the exposure and development and may lead to a defect. Similarly, if a dust particle lands on the reticle in the focal plane of the optics, it will be imaged down to the wafer site. Events such as these decrease the yield, and are especially critical in submicron geometries.

Many procedures have been developed to deal with these problems. Lithography is performed in a **clean room** environment that uses HEPA (high-efficiency particulate air) filters to remove dust particles. HEPA filters must be able to be 99.97% effective in removing particles with diameters of 0.5 μm or larger. A **Class X** clean room means that there are less than X particles per cubic foot with diameters greater than 0.5 microns; modern facilities have a Class 1 or better rating in critical work areas. To insure this level of cleanliness, workers must take air showers and wear special suits that cover all parts of the body before entering the area; these are generically referred to as "bunny suits" because of their appearance. Alternately, the entire flow may be automated and all movement performed by robots.

Lithographic areas are lighted by yellow light since it does not affect the UV-sensitive photoresist. To keep dust particles on the reticle from ruining the image, a thin layer of transparent plastic is placed above the reticle to catch dust and keep it off of the reticle surface. This is called a

pellicle, and is placed far enough above the reticle to keep the dust out of the image plane of the projection optics.

Many other features of the processing environment are included to insure that functional chips can be produced. Many scientists, engineers, and technicians are required to design, maintain, and update the processing areas. Touring an advanced chip fabrication facility is usually an overwhelming show of VLSI technology.

4.4 The CMOS Process Flow

Modern CMOS processing is, by all definitions, a "technological marvel." Starting literally with sand, the manufacturing line produces tiny rectangular slices that provide the computing power for the world. Semiconductor manufacturing companies have developed highly advanced processing techniques, and the details of their process flows are highly proprietary. Since a new manufacturing plant costs in excess of a billion dollars, it is no wonder that companies must remain secretive.

In this section we will study the main steps in a "standard" silicon CMOS process. The level of presentation has been chosen to insure that the main points are discussed without going into excessive details. Understanding CMOS processing is important to every VLSI designer, some more so than others. It depends on the task that the engineer is currently involved with. Device and circuit engineers view processing parameters as the fundamental limit to how fast their transistors and circuits can switch. The system architect understands that logic blocks need to be created in silicon, and that the processing dictates area allocations, interconnect levels, delays, clock speeds, and dozens of other system-level considerations. Everyone involved in the design of a VLSI chip is affected.

The initial steps are illustrated in Figure 4.16. It should be noted that the features, especially in the vertical directions, are not drawn to scale as this would obscure some of the important details. The starting point in Figure 4.16(a) is a p+ wafer with a thin p-type **epitaxial layer** of silicon grown on top. The epitaxial layer is created by dropping silicon atoms onto a heated wafer to form a high-quality crystal layer for transistors. The wafer itself acts as the substrate for building the chip, and is not shown explicitly in any of the remaining drawings.

The next step shown in Figure 4.16(b) is the formation of n-well regions using a masking step. This defines the locations of pFETs. In general, every transistor (nFET or pFET) is built in an **active area** of the wafer surface. Active areas are defined by a masking step that patterns a layer of silicon nitride that rests on a thin layer of thermal oxide that is used to relieve the mechanical stress of the crystal surface. Figure 4.16(c) shows the details after the patterning. Active areas are introduced as part of the **electrical isolation** scheme that prevents electrical conduction between

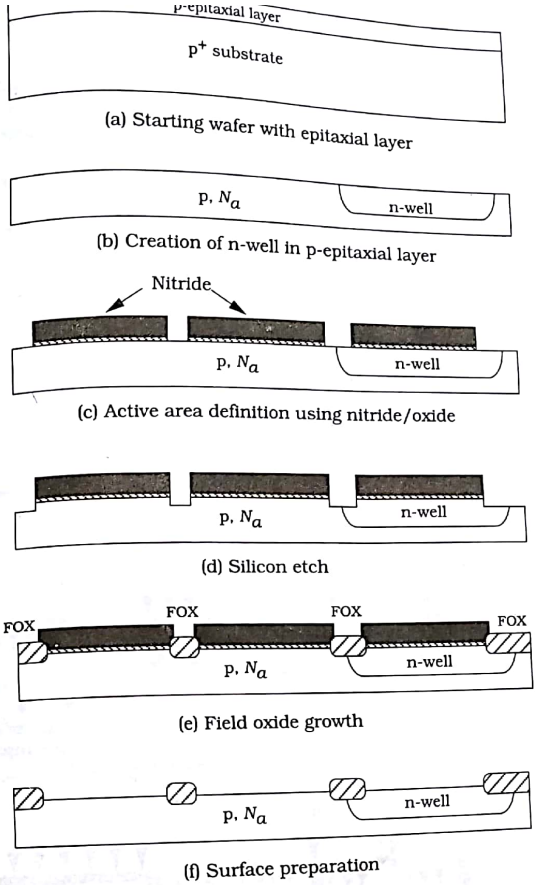
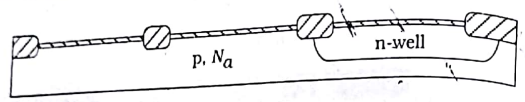


Figure 4.16 Initial sequences in the CMOS fabrication sequence

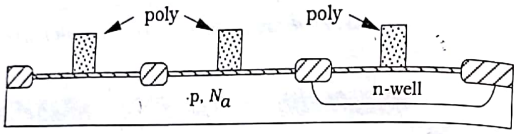
neighboring devices using recessed regions of glass (oxide) as an insulator. To achieve isolation, the nitride pattern is used to define silicon etched regions shown in Figure 4.16(d). Oxide is then grown or deposited in the etched regions as in Figure 4.16(e). Glass insulation between active areas defines the **field** regions, and the oxide there is called **field oxide** or

FOX. Once the FOX is grown, the layers are removed to expose the silicon surface. The wafer illustrated by the cross-sectional view in Figure 4.16(c) is now ready for the transistor fabrication process.

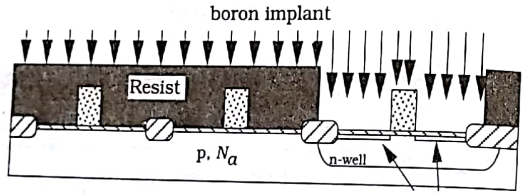
FETs are formed by using a **self-aligned gate process**. In this technique, the gates are created first and then used as implant masks to define the n+ or p+ drain/source regions. The starting point is the growth of the gate oxide shown in Figure 4.17(a). The value of t_{ox} is established during this step. Next, the polysilicon layer is deposited and patterned to form transistor gates. The resulting structure in Figure 4.17(b) shows the cross-sectional view at this point. To form transistors, we need to create



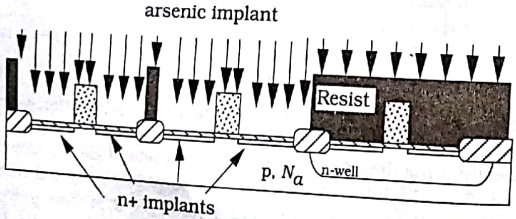
(a) Gate oxide growth



(b) Poly gate deposition and patterning



(c) pSelect mask and implant



(d) nSelect mask and implant

Figure 4.17 Formation of nFETs and pFETs

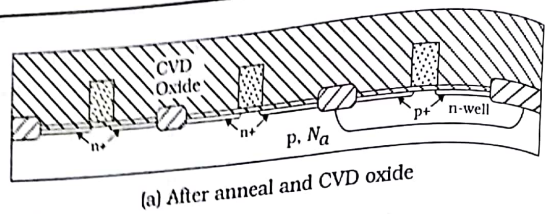
doped drain and source regions in the silicon. A pFET is created using a **pSelect** masking pattern with a boron ion implant. As illustrated in Figure 4.17(c), the pSelect mask creates a hardened photoresist layer that blocks the implant over nFET locations, but allows the ion beam to hit the pFET region. Ions are absorbed by the gate poly layer, but easily make it through the thin oxide layer to reach the silicon. The term "self-aligned gate" arises from this step. nFETs are formed in a similar manner. An **nSelect** mask is used to block an n-type ion implant from reaching pFET sites. Ions are permitted to bombard nFET locations, creating the n+ regions shown in Figure 4.17(d). At this point, all transistors have been built. Silicided gates can be created by layering a refractory metal on the poly. This lowers the sheet resistance of the poly lines. The remaining steps in the process flow are used to create interconnect layers.

The basic sequence for adding interconnect layers is illustrated in Figure 4.18 for the first layer of metal. CVD oxide is used to coat the surface as in Figure 4.18(a). Electrical contact with the n+ and p+ regions is established by etching holes in the oxide using an **Active Contact** mask. After the cuts are made, they are filled with a metal plug material such as tungsten (W). The resulting structure is shown in Figure 4.18(b). The first layer of metal is deposited and patterned with a Metal1 mask. This mask defines the first level of metal interconnect used to wire the circuit together. The drawing in Figure 4.18(c) illustrates the final view after the first metal has been patterned. Additional layers of metal are added in the same manner. Current processing lines have 5 or more metal interconnect layers (separated by oxide) to aid in complex wiring.

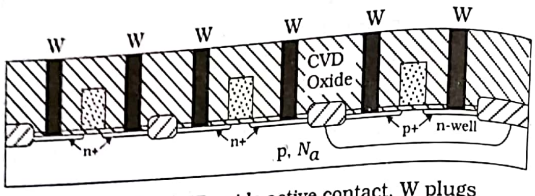
After all of the metal layers have been added, the entire chip is covered with the overglass layer that protects the surface from external contaminants. Silicon nitride is the most common overglass material since it is a dense dielectric that prevents diffusion of unwanted atoms and has good adhesion to metals. It is an insulator, so a via must be etched to gain electrical access to the chip; this requires another masking step. The simplest way to interface the silicon circuitry with the outside world is to use a **pad frame** arrangement where large metal **bonding pads** surround the central chip core area. Wires are attached between the pads and the output pins on the package. Figure 4.19 illustrates the basic idea. The top view in Figure 4.19(a) shows the metal pad (solid line) and an overglass cut (dashed line). The bonding pad itself may be quite large, with 100 μm × 100 μm being used in some processes. The side view in Figure 4.19(b) shows the details of the bond itself. A robotic apparatus is used to place the bond on the pad accurately and string a wire from the chip to the specified pin on the package frame.

4.4.1 Variations

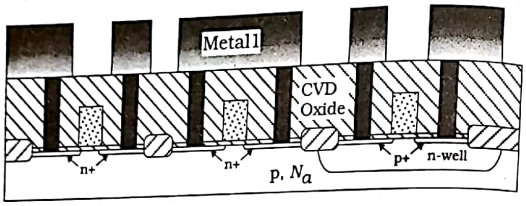
Modern CMOS processing lines use a large number of enhancements to the basic flow described above. These are usually included to provide



(a) After anneal and CVD oxide

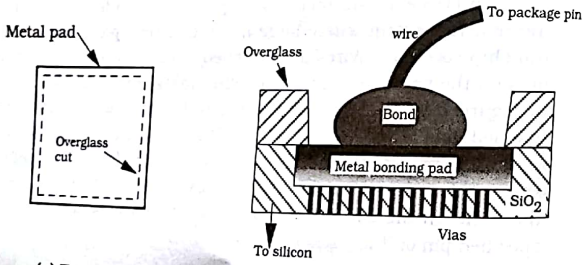


(b) After CVD oxide active contact, W plugs



(c) Metal1 coating and patterning

Figure 4.18 First metal interconnect layer



(a) Top view

(b) Side view

Figure 4.19 Bonding pad structure

better electrical characteristics, combat small device or high-density problems, or to enhance the yield. We will examine two additional steps that are now standard: lightly doped drain (LDD) FETs and silicides. In addition, we will take a brief look at how copper interconnect patterns are created.

A lightly doped drain MOSFET is designed to reduce the electric fields in the channel region by providing n- (lightly doped) drain and source regions instead of the usual n+ regions. Theoretically, this reduces the maximum electric field intensity, which in turn increases the reliability of the transistors.[2] LDD structures can be created without an additional mask, so that their presence is usually transparent to the layout designer.

A sequence for creating an LDD FET is portrayed in Figure 4.20. The starting point is shown in Figure 4.20(a). To create an n-channel MOSFET, we start with a low-dose donor doping to create the n- (lightly doped) drain and source regions shown. The next step [illustrated in Figure 4.20(b)] is to deposit an oxide layer over the surface. Note that an oxide layer coats the side (vertical) wall of the poly gate feature. After this, the wafer is subjected to an oxide etching step. When viewed from the top, the sidewall oxides are thicker than the oxide covering the flat portions of the surface. This results in the **sidewall spacers** shown in the drawing of Figure 4.20(c). The spacers are used to block the heavy n+ donor implant in Figure 4.20(d), which keeps the drain and source regions closest to the channel at lightly doped levels. The lateral (horizontal) width of the spacers determines the extent of the n-regions.

Figure 4.21(a) provides an expanded view of a finished LDD nFET with the details of the doped regions shown. This can be used as the basis for studying silicides, which is the second variation from the basic CMOS flow that we will examine. Even heavily doped polysilicon exhibits a sheet resistance of about 25 Ω or more, limiting its use as an interconnect material. To overcome this problem, a refractory metal such as titanium or platinum can be coated over silicon or polysilicon as in Figure 4.21(b). The resulting silicide reduces the sheet resistance of the poly layer without affecting the electrical characteristics of the MOS gate structure; a typical order of magnitude for a silicided poly is $R_s \approx 10$ mΩ. The drain-source n+ silicides reduce the contact resistance when a tungsten plug is used as an active contact. Owing to this fact, silicides have become very common in high-frequency processes. We note in passing that neither Pt nor W by themselves can be used to replace the polysilicon gate (and form a true MOS structure) as they do not adhere to the silicon dioxide insulating layer, but simply "slide off."

The last variation that we will examine is the use of copper (Cu) as an

[2] This is accomplished by using LDD FET to reduce the **hot-electron effects** found in short-channel devices.

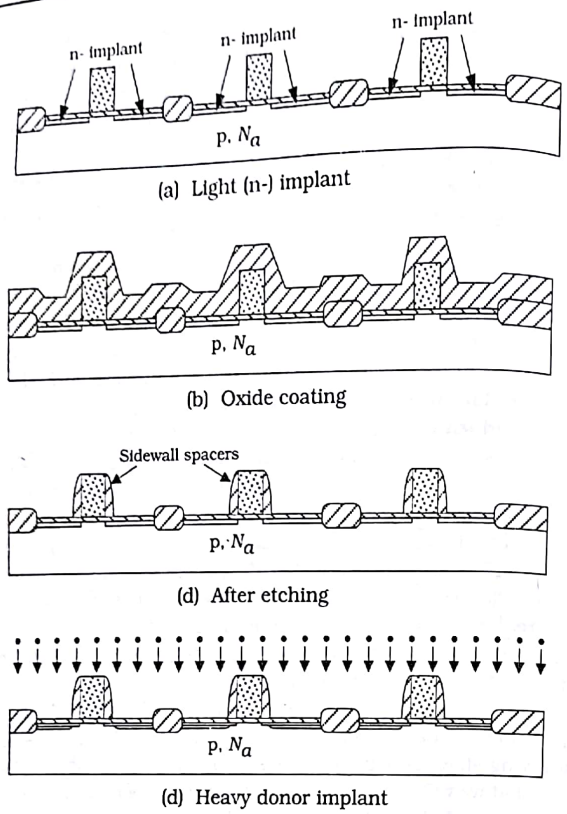


Figure 4.20 Sequence for creating a lightly doped drain nFET

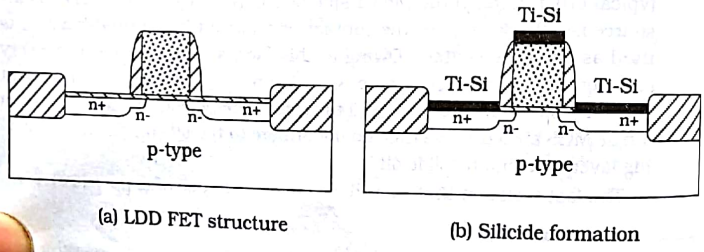


Figure 4.21 LDD nFET with silicided gate and contacts

interconnect material instead of aluminum. It is a well-known fact that the bulk resistivity of copper is $\rho = 1.67\ \mu\Omega$-cm, which is about one-half that of Al. When used as an interconnect line material, the sheet resistance would be about one-half that of an aluminum line with the same thickness. Copper, however, has proved difficult to introduce into the processing line. It cannot be patterned using the standard sequence of deposition followed by a lithographic step because it is very difficult to etch using standard RIE techniques. Copper diffuses very rapidly through silicon and can alter the electrical characteristics, so it cannot be directly deposited on top of any silicon regions. It also diffuses through silicon dioxide, making the problem even more difficult to deal with. Much research has been directed toward the development of techniques to replace aluminum with low-resistivity interconnect metals. At the present time, copper is being introduced into the majority of new high-speed CMOS lines, making it of interest to the VLSI designer. One of the first VLSI chips to use copper technology was an advanced generation Power PC microprocessor design.

Let us first examine how copper patterns are produced. As mentioned above, dry-etching techniques do not etch copper. Even trace amounts of copper from Al-Cu mixtures are difficult to remove from a chip surface. To get around this problem, we use the **Damascene** process based on the method used in ancient times to inlay gold or silver into an iron sword. The name is taken from the city of Damascus whose artisans were well known for their work. In this technique, the copper pattern is first etched into a silicon dioxide layer; copper is then deposited (using, for example, electroplating) on the surface. The sequence is shown in Figures 4.22(a) and (b). To avoid the etching problem, we subject the

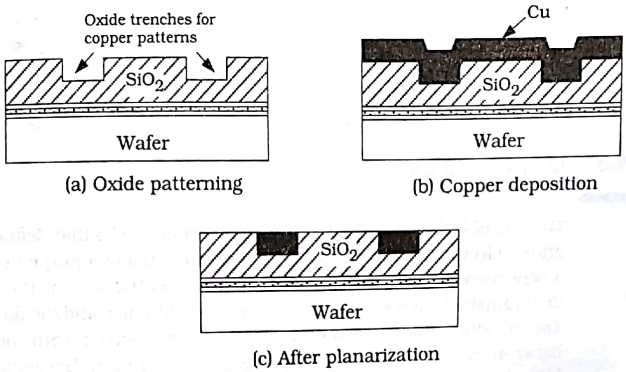


Figure 4.22 Copper patterning using the Damascene process

wafer to a chemical-mechanical polishing (CMP) step that planarizes the surface and removes copper not in a oxide trench. This results in the structure shown in Figure 4.22(c).

Dual-Damascene processes that allow copper vias to be created have also been developed. The basic sequence is the same, except that two oxide etch steps are used to give the general structure portrayed in Figure 4.23. Copper vias have a lower resistance than tungsten, and also avoid the contact resistance introduced by a standard Al-W interface.

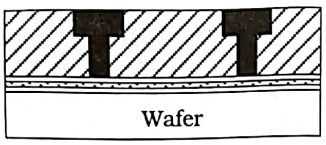


Figure 4.23 Dual-Damascene structure with copper vias

The other major problem with using copper as an interconnect material is the need to prevent it from diffusing into silicon regions. This is achieved by using thin barrier layers to contain the material. Copper has relatively poor adhesion properties, so providing a barrier layer around the copper feature can help. Various barrier materials have been tried with results published in the literature. Included in this group are W, Ti, TiN, Ta, TaN, and TaN_x (where x is the mole fraction of nitrogen). The choice of materials affects the resistivity and sheet resistance of the interconnect, so many trade-offs exist. Moreover, reliability issues and long-term effects still need to be studied in more detail.

The fabrication problems introduced by using copper illustrate the complexity of a modern silicon processing line. Even a small perturbation is significant, but a major change can require years of research before it reaches the manufacturing line. Ion implantation was a research technique until it was studied and restructured for the production line. Although the investment in time and money is large, these examples show that the rewards can be great.

4.5 Design Rules

The role of physical design is to create a set of masks that define the integrated circuit. The layout itself is performed using a graphics CAD tool where every polygon on every layer is drawn on the screen. The layers are distinguished from each other using different color and/or fill patterns. The drawing area is based on a reference grid pattern, with the distance between each grid point representing a specific length. Designing the patterns for a silicon chip is much like drawing boxes on a piece of grid paper using a set of colored pencils. However, just because we can draw something does not mean that it can be fabricated. Every piece of fabrication equipment used in the IC manufacturing process has limited accuracy. A lithographic stepper unit that is designed to image linewidths of 0.25 μm will not operate at 0.18 μm. The same is true for an etching system. Physical limitations at the silicon level also restrict what can be fabricated in the microworld of silicon circuitry.

Topological **design rules** (DRs) are a set of geometrical specifications that dictate the design of the layout masks. A design rule set provides numerical values for minimum dimensions, line spacings, and other geometrical quantities that are derived from the limits of a specific processing line. The design rules must be followed to insure functional structures on the fabricated chip. An example of a design rule specification is shown in Figure 4.24 for the case of two closely spaced polysilicon lines. This drawing is used to show the two parameters

w_p = minimum width of a polysilicon line

$s_{p\text{-}p}$ = minimum poly-to-poly spacing

These are given numerical values in the DR listing; violating these values may lead to a failure. Every layer in the process will have similar quantities assign to it. In our notation,

w = minimum width specification

s = minimum spacing value

d = generic minimum distance

with subscript used to denote the relevant layers. For example,

w_{m1} = minimum width of a metal1 line

$s_{m1\text{-}m1}$ = minimum spacing between metal1 lines

This convention makes it easy to understand each rule as it is introduced and used in layout drawings. In practice, layers are numbered and design rules are usually assigned identifiers that are associated with the layer numbers.

All design rule specifications such as w and s have units of length, with the micron (μm) being the most common metric. For example, a process might specify polysilicon features with minimum width and spacing

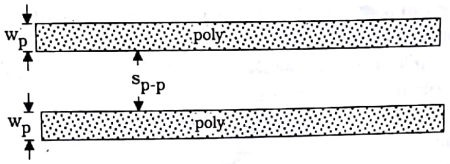


Figure 4.24 Design rule limits for two polysilicon lines

values of $w_p = 0.25\ \mu m$, $s_{p\text{-}p} = 0.425\ \mu m$

The layout grid in the CAD system is usually calibrated to accomm
the necessary resolution. These values are obtained by a careful ar
of relevant parts of the manufacturing line and vary with the pro
set of topological design rules may require a document listing
pages or more, and takes some time to learn. This level of detail is
sary to produce chips with the highest possible packing density.

Design rules change with technological advances, so we have mad
decision not to include a specific set in this text. In the United State
government-sponsored MOSIS group provides foundry access to un
ties and small companies.[3] The reader is directed to the MOSIS web
at **www.mosis.org** where up-to-date sets can be viewed and downlo
for immediate usage. In this spirit, our discussion will be kept gene
nature.

The popularity of modern VLSI applications has introduced the
cept of the **silicon foundry**. A foundry provides access to a chip manu
turing process on a pay-by-use basis. On a more global scale, found
such as TSMC are used by large corporations down to well-funded
viduals to create their designs.[4] A foundry allows designers to sub
designs using a state-of-the-art process. Since the customer base is
and varied, most foundry operations allow the submission of desi
using a simpler set of design rules that can be easily scaled to diffe
processes. These are called **lambda design rules**.

Lambda design rules are based on a reference metric λ that has u
of μm. All widths, spacings, and distances are written in the form

$$\text{Value} = m\lambda \qquad (4.2$$

where m is scaling multipler. For example, we might stipulate that $w =$
and $s = 3\lambda$ for the minimum width and spacing on a layer. The numeri
values of w and s are not known until λ itself is specified. If $\lambda = 0.15\ \mu$
then these would specify that

$$w = 2(0.15) = 0.30\ \mu m$$
$$s = 3(0.15) = 0.45\ \mu m$$

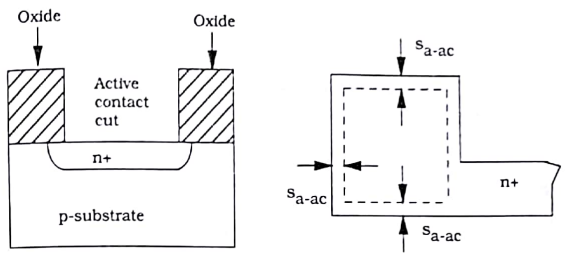
for the design. If the layout is based on a λ-grid, then submitting t
design to a different process just means that the numerical value o
must be changed. The relative dimensions remain the same. The ma
drawback of using a **scalable** design rule set of this type is that it is n
possible to achieve the highest packing density using integer values of m

[3] MOSIS stands for MOS Implementation Service.

[4] TSMC stands for Taiwan Semiconductor Manufacturing Corporation.

Design rules can be classified into four main types: minimum width, minimum spacing, surround, and extension. We have already seen minimum width and minimum spacing examples. A surround rule is enforced when a feature must be placed inside of an existing feature on the chip surface. An extension rule is similar in that it requires that portion of the pattern be extended beyond the edge of an existing border.

Let us consider the placement of an active contact as an example of a surround rule. As shown in Figure 4.25(a), the oxide contact cut must be aligned so that it is over the existing (active) n+ region. The corresponding design rule is shown in Figure 4.25(b). The surround spacing $s_{a\text{-}ac}$ between the active area (n+) and active contact edge must be maintained to guard against a misaligned contact cut pattern during the lithographic exposure step.



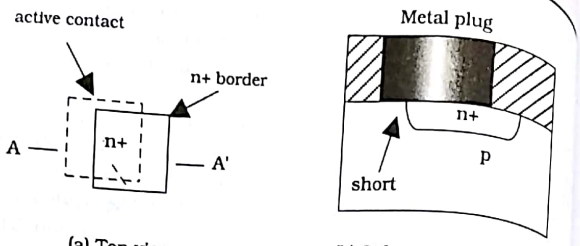
(a) Side view (b) Surround rule

Figure 4.25 Example of a surround design rule

Misalignment problems must be included in the design rule set because it is not possible to project the reticle image to the chip surface with an arbitrary degree of accuracy. The registration marks are in the form of geometrical target patterns on some layers during the processing. The targets are used to align several subsequent patterning steps. When an opaque material layer is deposited, a new set of marks must be introduced. Surround rules are included to compensate for the alignment tolerance of the stepper.

Figure 4.26 illustrates the potential problem with the active contact. Suppose that the contact cut is not aligned to fall within the n+ active region as seen in Figure 4.26(a). After the contact is made and the metal plug added, the cross-sectional view in Figure 4.26(b) shows the existence of a metal-substrate short. This will render the chip nonfunctional.

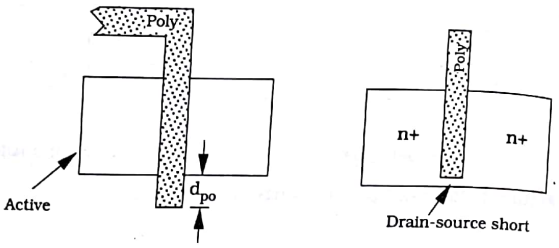
Extension-type design rules also tend to be based on misalignment problems. Consider the formation of a self-aligned nFET as an example.



(a) Top view (b) Side view along A-A'

Figure 4.26 Misalignment-induced defect

The polysilicon gate is used as a dopant mask for the n-type ion implant that defines the drain and source regions. In Figure 4.27(a), the extension distance d_{po} (for poly overhang) is included to insure functional p structures. If we do not provide the overhang distance, then a misalign poly mask may result in the situation shown in Figure 4.27(b). In t case, the poly edge did not traverse the entire active area, so that the implant creates a short between the drain and source sides.



(a) Gate overhang DR (b) Misalignment failure

Figure 4.27 Example of an extend (gate overhang) design rule

4.5.1 Physical Limitations

Some geometrical design rules originate from physical considerations. These enter into the formulation of the design rule set, and may or may not be obvious.

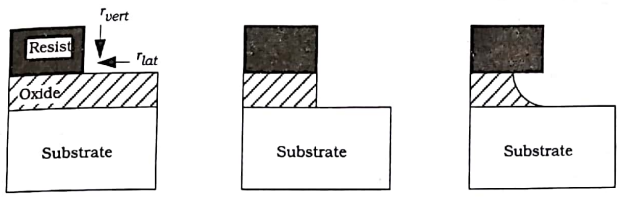
An important aspect is the linewidth limitation of an imaging system. The reticle shadow projected to the surface of the photoresist does not have sharp edges due to optical diffraction. As a simple rule of thumb, a lightwave with an optical wavelength of λ cannot accurately image a feature size much less than that value. UV-sensitive positive photoresists are used because the short wavelengths of the ultraviolet light allow for better

resolution of fine linewidths, and positive resists have better development properties than negative resists. In addition, the structure of a reticle is much more complicated than we have alluded to; advanced optical techniques such as phase-shifting structures are used to enhance the resolution.

The etching process introduces another type of problem. When we remove material around a resist edge, both vertical (perpendicular to the wafer surface) and lateral (parallel to the surface) etching occurs. We can characterize the respective etch rates of the two by r_{vert} [μm/min] and r_{lat} [μm/min] and define the **degree of anisotropy** A by

$$A = 1 - \frac{r_{lat}}{r_{vert}} \tag{4.22}$$

The presence of lateral etching in r_{vert} limits the resolution that can be achieved. Figure 4.28(a) shows an oxide layer that is to be patterned by the resist layer on top of it. A pure anisotropic etch profile is shown in Figure 4.28(b). This is characterized by $r_{lat} = 0$ which gives vertical walls and $A = 1$. The result of a pure isotropic etch with $r_{lat} = r_{vert}$ is shown in Figure 4.28(c). Undercutting of the resist due to the lateral etching decreases the resolution that can be used in the design. Another factor that enters the problem is the absorption profile of light by the resist layer itself; this results in the resist edges having finite slopes instead of well-defined vertical shapes.



(a) Resist pattern (b) Pure anisotropic etch (c) Isotropic etch

Figure 4.28 Etching profiles

Semiconductor effects in silicon also influence the formulation of design rules. Any time a pn junction is formed it gives rise to what is known as a **depletion region** at the interface. By definition, the depletion region is "depleted" of free electrons and holes because of an electric field that originates from the dopants and forces the charges out. If the depletion regions of adjacent pn junctions touch, then the current blocking characteristics are altered and current can flow between the two. This limits the spacing rule $s_{n\text{-}n}$ shown in Figure 4.29. The drawing also

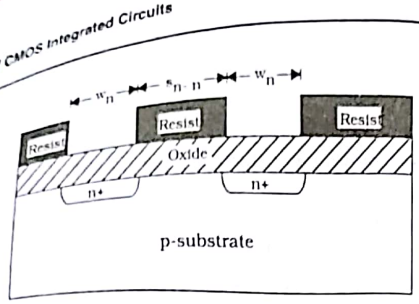


Figure 4.29 Limits on n+ spacings

shows that the minimum linewidth parameter w_n must account for la
doping and isotropic etching effects.

Another physical problem is the electrical capacitive coupling
occurs between closely spaced conducting lines. This leads to a pro
called crosstalk in which a portion of the electrical energy from one lo
coupled to another, causing an unwanted perturbation called "noi
This can lead to errors, and is a major problem in high-density desig
Memory chips are particularly sensitive to induced noise of this ty
Crosstalk considerations can lead to design rule spacing values that
much larger than the minimum value that can be achieved by the lith
raphy. The problem of crosstalk itself is treated in more detail in Chap
14.

4.5.2 Electrical Rules

In addition to topological design rules, a CMOS process provides electr
layout rules. These tend to be in the form of changes to the basic des
rule values when certain electrical conditions occur. Electrical rules m
be provided directly in the general set, or as an addendum.

An example of an electrical rule is the allowed width of a metal int
connect line. To avoid electromigration effects, the design rule set m
stipulate the maximum current flow level permitted for a given linewid
Larger currents require wider lines.

4.6 Further Reading

[1] Stephen A. Campbell, **The Science and Engineering of Microele
tronic Fabrication**, Oxford University Press, New York, 1996.

[2] C.Y. Chang and S.M. Sze, **ULSI Technology**, McGraw-Hill, New Yo
1996.

[3] James D. Plummer, Michael Deal, and Peter B. Griffin, **Silicon VL
Technology**, Prentice Hall, Upper Saddle River, NJ, 2000.

Elements of Physical Design 5

In the previous chapter we examined the basic fabrication sequence for manufacturing CMOS integrated circuits. In this chapter we will study the details of translating logic circuits into silicon, which is called **physical design**. Details such as the minimum size specifications allowed for a patterned region become critical. However, the most important lessons in the physical design of VLSI chips revolve around the use of CAD tools and database structures that describe the silicon masks. These give the needed information for creating the chip, and provide the basis for the hierarchical design of large complex logic networks.

5.1 Basic Concepts

Physical design is the actual process of creating circuits on silicon. During this phase of the VLSI design process, schematic diagrams are carefully translated into sets of geometric patterns that are used to define the on-chip physical structures. Every layer in the CMOS fabrication sequence is defined by a distinct pattern. A patterned layer consists of a group of geometrical objects that are generically referred to as **polygons**. This naturally includes rectangles and squares, but allows us to include arbitrarily complex *n*-vertex shapes with specific dimensions. Examples of the types of polygons that occur in a CMOS design are shown in Figure 5.1 where several are superposed to form the overall layout. When stacked into three-dimensional structures, the layers are electrically equivalent to the circuit diagram.

Our study to this point shows that the **topology** of the transistor network establishes the logic function. In other words, the details of how the FETs are wired together (series, parallel, etc.) are sufficient to determine the binary operations of the circuit. Another aspect of logic is that of

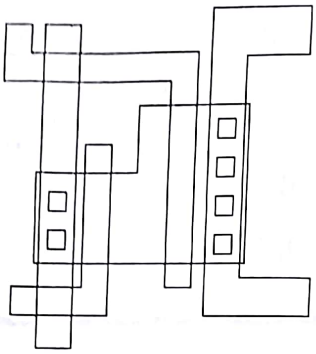


Figure 5.1 Examples of polygons in physical design

switching speed. This is more complicated to analyze, but is crucia
important to modern chip design. Although the details will be discus
later, we can summarize the main points here. For a given set of proce
ing parameters, we will find that the electrical characteristics of a lo
gate depend on the aspect ratios of the transistors. This is due to both
current flow levels and the parasitic resistance and capacitance of
devices. Physical design must address both of these areas. The patter
must be created to correctly implement the signal flow network, as w
discussed in Chapter 3. The complicating factor is that the dimensions
every feature affect the electrical performance of the circuit. In a VL
chip, the switching speed of some gates will be critical, especially those
long complex logic paths. We will concentrate on studying the basics
circuit layout in this chapter, and delve into the intricacies of high-spee
circuits in Part 2 of the book.

The process of physical design is performed using a computer to
called a **layout editor**. This is a graphics program that allows th
designer to specify the shape, dimensions, and placement of every poly
gon on every layer of the chip. Complexity issues are attacked by firs
designing simple gates and storing their descriptive files in a **library** sub
directory or folder; the gates constitute **cells** in the library. Library cells
are used as building blocks by creating copies of the basic cells to con
struct a larger more complex circuit. This process is called **instancing** o
the cells, while a copy of a cell is called an **instance**.

The layout must be an accurate representation of the logic network
but much of the designer's work is directed toward the goal of obtaining a
fast circuit in the minimum amount of area. Small changes in the shapes
or areas of a polygon will affect the resulting electrical characteristics of
the circuit. However, the changes may or may not be significant for the
logic chain. An experienced layout designer gains a certain level of intu
ition that often helps find trouble spots. Circuit simulations also help

insure that the layout is accurate and provides a network that meets specifications.

5.1.1 CAD Toolsets

Physical design is based on the use of CAD tools that simplify the procedure and aid in the verification process. The most powerful toolsets are collections of programs that are combined together into an integrated suite environment. There are several packages available, each with its own strengths.

Let us examine what constitutes a basic chip design toolset by listing some of its features. For the physical design process the primary tool is the layout editor described above. This is a graphical interface to a database that allows the user to draw transistors and wiring patterns made up of polygons. Each layer has a distinct color or fill pattern on the screen. The overlap of the boxes or polygons on each layer becomes our view of the transistor. The layout editor creates a database for each layer that describes the patterning on a universal grid. This is eventually used to create the masks needed to pattern the layers in the fabrication sequence.

After a layout is completed, we must run several secondary programs that use database information to determine if our layout is valid. The electrical behavior of the design is simulated by first using an **extraction** routine that translates the polygon patterns and layers into an equivalent electrical network. The output of an extraction routine is a netlist file that can be used in a circuit simulation program; SPICE formats are the most common. Extraction programs provide important geometrical parameters such as the drawn channel width and length for each FET. They also specify how the transistors are wired together. Process-dependent electrical parameters are added to the extract output file to form a complete basis for simulation. Circuit simulation codes such as SPICE are usually included within the toolset (or within a related subdirectory) for easy access. This allows the designer to immediately perform simulations as needed.

A related program that is usually included in the design environment is called **layout versus schematic**, or **LVS** for short. As implied by its name, this program checks the layout against the schematic diagram. This is important to verify that the layout corresponds to the intended circuit. LVS can be performed using either logic diagrams or electronic circuit schematics.

The **design rule checker** (**DRC**) is a program that uses the layout database and checks every occurrence of the design rule list on the layout. This means, for example, that the width and spacing of every metal line in the layout are checked to insure that they do not violate the minimum specified values. Passing a DRC insures that the design can be

fabricated within the limitations of the manufacturing process.

Other tools are provided to help in large designs. **Place and route** routines help the layout designer by automatically finding viable wiring routes between two specified points. This is useful when trying to connect two complex units together. Electrical continuity can be seen using an **electrical rule checker (ERC)** which highlights connecting paths.

This short description of a chip design environment provides us with a starting point for a more detailed discussion of the layout and design of VLSI silicon networks. Our approach will be to stress the fundamental ideas and procedures without going into the details of using any specific set of CAD tools. Once the techniques are understood and mastered, they may be applied to any environment.

5.2 Layout of Basic Structures

Let us start with the sequences that are used to define chip regions. We will base our discussion on the p-substrate (n-well) technology described in Chapter 4.[1] The masking sequence was established as

0. Start with p-type substrate
1. nWell
2. Active
3. Poly
4. pSelect
5. nSelect
6. Active contact
7. Poly contact
8. Metal1
9. Via
10. Metal2
11. Overglass

It is important to remember that oxides are grown or deposited between conducting layers above the substrate. The details needed for chip layout vary with the sequence. However, only minor modifications are needed to extend the ideas here to arbitrary processing lines.

In this section we will study how to design basic structures on the chip such as n+ and p+ regions and MOSFETs using the basic masking sequence. Relevant design rules are introduced for each structure. It is worth remembering that the features on every level have design rule specifications for the minimum width w of a line, and a minimum edge-to-edge spacing s between adjacent polygons. These are illustrated in Figure 5.2.

[1] These techniques are quite general, and may be easily extended to other technologies with minor changes.

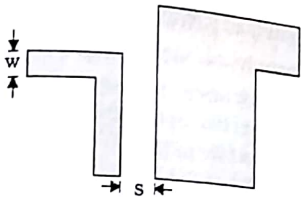


Figure 5.2 Minimum line width and spacing

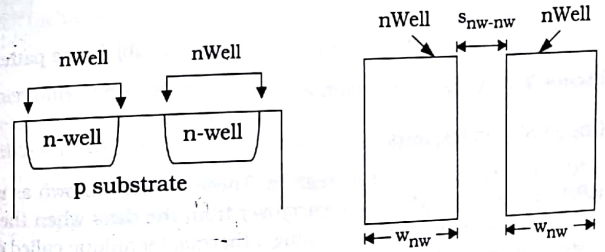
The actual values for w and s depend on the layer. Design rules apply only to the features on the *mask* (a generic term for the reticle) for that layer. The actual fabricated structure on the chip will have different dimensions For that reason, we will sometimes refer to the layout sizes as the **drawn** values, while the resulting sizes on the finished chip have **effective** or **final** values. This is particularly important in designing FETs.

Our discussion will consider only **Manhattan** geometries where all turns are multiples of 90°. Right-angle layout is the most straightforward to learn, but does not always give the best packing density. Many layout editors allow you to select the angles in an arbitrary manner, but you must be sure that the structures are supported by the fabrication process.

As mentioned in the previous chapter, our discussion will be generic in nature. Detailed and up-do-date sets of design rules for various processes can be obtained from the MOSIS web site at **www.mosis.org**.

5.2.1 n-Wells

An n-well is required at every location where a pFET is to be made. We define these using the nWell mask on which closed polygons represent the placement of the wells. Figure 5.3(a) shows the cross-sectional view of two adjacent n-well regions. The polygons in Figure 5.3(b) constitute the mask set for this part of the chip. The drawing illustrates the two design rules



(a) Cross-section (b) Mask set

Figure 5.3 n-well structure and mask

w_{nw} = minimum width of an n-well mask feature

$s_{nw\text{-}nw}$ = minimum edge-to-edge spacing of adjacent n-wells

It is often possible to merge adjacent n-wells together into one. Al must be remembered that an n-well must have a connection to the p supply V_{DD} when used for pFETs.

5.2.2 Active Areas

Silicon devices are built on active areas of the substrate. Figure 5. illustrates the cross-sectional view of an active section. After the isola (field) oxide is grown, an active area is flat and provides access to the of the silicon wafer. The field oxide (FOX) exists everywhere else on wafer. Active regions are defined by closed polygons on the **Active** ma The set of polygons required to define the patterns in Figure 5.4(shown in Figure 5.4(b). The relevant design rule spacings are denoted

w_a = minimum width of an Active feature

$s_{a\text{-}a}$ = minimum edge-to-edge spacing of Active mask polygons

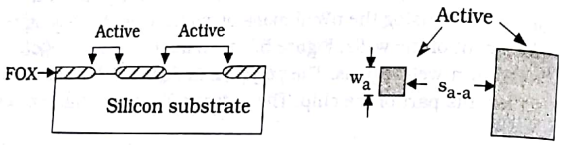
These are minimum values that must be observed in maximum den designs. The field oxide regions can be derived from the Active mask the expression

$$FOX = NOT\ (\ Active\ ) \qquad (5.$$

This is a symbolic expression that is based on the observation

$$FOX + Active = Surface \qquad (5.$$

In other words, if a region is not Active, then it is FOX by default.

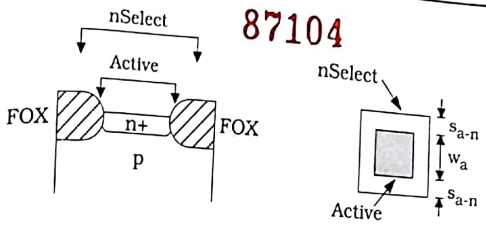


(a) Cross-section (b) Active patterns

Figure 5.4 Active area definition

5.2.3 Doped Silicon Regions

Next, let us create n+ and p+ regions. These are also known as **ndiff** and **pdiff**, respectively, which is a carryover from the days when the dopants were introduced into the wafer using a thermal technique called **diffusion** instead of ion implantation. An n+ region is shown in Figure 5.5(a). It is created by ion implanting arsenic or phosphorus ions into the substrate in areas described by the **nSelect** mask. Since this is done after the isolation process, the nSelect mask defines regions that cover Active areas.



(a) Cross-section (b) Mask set

Figure 5.5 Design of n+ regions

The mask set shown in Figure 5.5(b) shows that both the nSelect and Active areas are needed to create the n+ region. Let us use the notation (Mask_name) to imply the set of all polygons on that layer. If only the nSelect and Active masks are included, we may express an n+ region by[2]

$$n+ = (\ nSelect\ ) \cap (\ Active\ ) \qquad (5.3)$$

This says that n+ regions are created whenever the Active and nSelect masks intersect, as denoted by the intersection operator ∩. Two design rules are illustrated in the drawing. These are

w_a = minimum width of an Active area

$s_{a\text{-}n}$ = minimum Active-to-nSelect spacing

where it is implied that spacing distances are measured from edge to edge; this convention will be followed throughout the book. Design rules are usually invariant with respect to direction, so that the same values also apply to the horizontal dimensions.

A p+ region is obtained by ion implanting boron into an active area opening on the wafer. The cross-sectional view in Figure 5.6(a) shows that p+ regions are created in n-well areas in this technology. The active region is made p-type by using the implant defined by the **pSelect** mask. The required mask set is shown in Figure 5.6(b) where the **nWell** mask has been included for completeness. The expression for a p+ region is

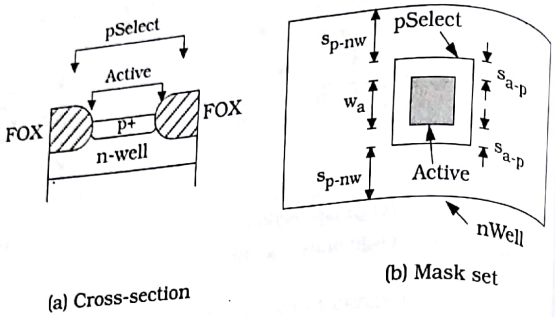
$$p+ = (\ pSelect\ ) \cap (\ Active\ ) \cap (\ nWell\ ) \qquad (5.4)$$

when only these three masks are considered. This says that p+ regions are created whenever regions on the pSelect and Active masks overlap within an nWell region. The important design rule spacings are shown as

w_a = minimum Active area width

$s_{a\text{-}p}$ = minimum Active-to-pSelect spacing

[2] In the Magic layout editor, **ndiff** and **pdiff** are drawn by a single command so that separate Active and nSelect patterns are not necessary. Note, however, that there are no such things as ndiff and pdiff masks in the process.



(a) Cross-section (b) Mask set

Figure 5.6 Design of a p+ region

$s_{p\text{-}nw}$ = minimum pSelect-to-nWell spacing

Again, these are specified by the process design rule set.

5.2.4 MOSFETs

Self-aligned MOSFET structures exist every time a poly gate line co pletely crosses an n+ or p+ region. Physically, the poly line is depos before the ion implant, and acts to block dopants from entering the s con. FETs thus require the use of polygons on the **Poly** mask layer. T basic design rules for Poly features are

w_p = minimum poly width

$s_{p\text{-}p}$ = minimum poly-to-poly spacing

The minimum poly linewidth w_p is the same as the drawn channel leng for a FET.

Let us construct an nFET first. In Figure 5.7(a) the cross-sectional vie shows the n+ and poly layers; the gate oxide between the gate and sub strate is not shown explicitly. The top view in Figure 5.7(b) shows th drawn values of the channel length L and the channel width W of th transistor. To construct the mask set, we just add a polygon to the Po mask that separates the n+ into two regions. This results in the mask shown in Figure 5.8. The implied design rule is

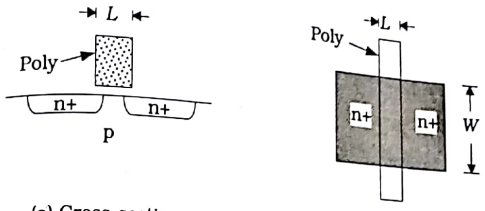
$L = w_p$ = minimum width of a Poly line

The other design rule shown is

d_{po} = minimum extension of Poly beyond Active

which is required to insure the formation of the self-aligned FET if a smal registration error occurs in the lithography. This is known as the **gate overhang** distance. Using the figure allows us to write the definition of th central part of the nFET as

$$\text{nFET} = (\text{nSelect}) \cap (\text{Active}) \cap (\text{Poly}) \quad (5.5$$



(a) Cross-section (b) Layout view

Figure 5.7 nFET structure

since this is where the channel is formed. The n+ regions are defined by

$$\text{n+} = (\text{nSelect}) \cap (\text{Active}) \cap (\text{NOT [Poly]}) \quad (5.6)$$

This is more precise than the limited definition given previously in equation (5.3) which ignored the existence of the Poly mask.

A pFET is created in the same manner. Figure 5.9(a) shows the cross-sectional view of the device, while the top view in Figure 5.9(b) provides the important channel dimensions L and W as drawn on the masks. It is worth mentioning that the n-well region is surrounded by implied p-substrate; this is shown explicitly in the top-view drawing. The pFET mask set shown in Figure 5.10 has the same basic features as the nFET group. The differences are only in the polarity of the implant (pSelect instead of nSelect) and the presence of nWell surrounding the transistor. The drawn channel length L corresponds to the minimum Poly linewidth, while d_{po} is the gate overhang design rule. The other design rules implied by the drawing have already been discussed. A simple expression for the central pFET region is

$$\text{pFET} = (\text{pSelect}) \cap (\text{Active}) \cap (\text{Poly}) \cap (\text{nWell}) \quad (5.7)$$

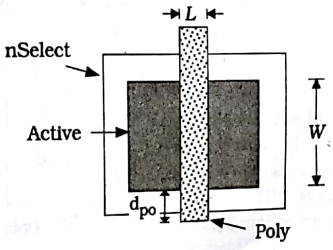
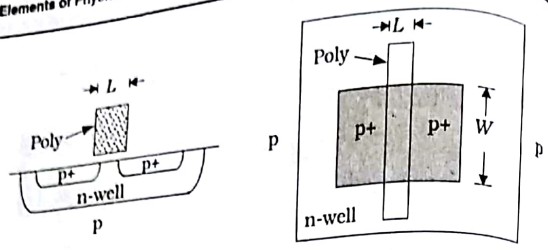


Figure 5.8 Masks for the nFET



(a) Cross-section (b) Layout view

Figure 5.9 pFET structure

which expresses the device as the overlap of the four masks. A p+ region is then described by

$$p+ = (\text{pSelect}) \cap (\text{Active}) \cap (\text{nWell}) \cap (\text{NOT [Poly]}) \quad (5.8)$$

i.e., sections in the device where no poly has been created. This is more precise than the simpler expression in equation (5.4) which ignored the Poly mask.

Drawn and Effective Values in MOSFETs

The critical dimensions of a MOSFET are the channel length L and the channel width W. As we have seen, L is established by the width of the poly gate line. Tracing the fabrication sequence shows that the channel width W is set by the appropriate edge measurement of the Active transistor area, since that region defines where the drain/source ion implant penetrates into the silicon. As mentioned in the previous chapter, design rules dictate the mask layout and represent the drawn dimensions. The final values measured on the chip will be slightly different. The exact relationship is particularly important in the electrical analysis of transistors.

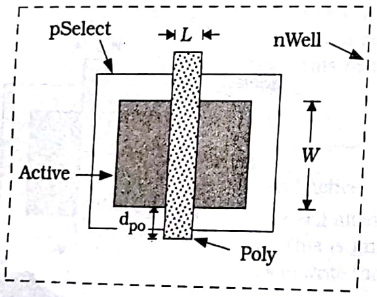
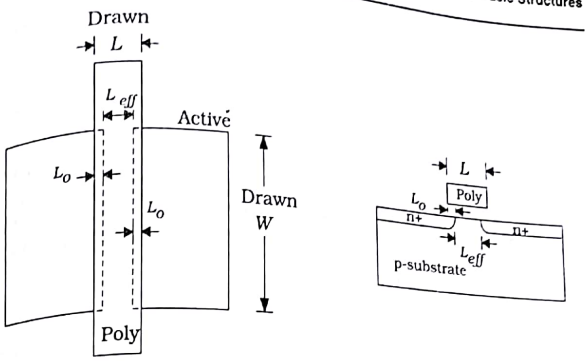


Figure 5.10 pFET mask set



(a) Drawn layout (b) Finished view

Figure 5.11 Drawn and effective dimensions of a MOSFET

Let us examine the FET layout geometry shown in Figure 5.11(a); this general layout applies to both nFETs and pFETs. Consider first the channel length of the device. The drawn value L is the width of the polysilicon line. However, the distance between the n+ regions in the final structure is smaller than L due to lateral doping during the implant annealing step. When the wafer is heated, the dopants on opposite sides move toward one another. The overlap effect is symmetrical and results in the overlap distance L_o on both sides. In the electrical analysis of the transistor, the important distance is the final value between the two n+ regions. When a distinction needs to be made, the final value is referred to as the **electrical** or **effective channel length**. Denoting the effective value by L_{eff} we see that

$$L_{eff} = L - 2L_o \quad (5.9)$$

gives the numerical value. A more general form is

$$L_{eff} = L - \Delta L \quad (5.10)$$

where ΔL is the total reduction in channel length due to overlap and other effects.

The channel width is also smaller than the drawn value due to a reduction of active area by the field oxide growth. This is called active area **encroachment**, and leads to an effective channel width of the form

$$W_{eff} = W - \Delta W \quad (5.11)$$

where W is the drawn value and ΔW is the total reduction in channel

length from all effects. The aspect ratio of the transistor that is u
the electrical characterization is *always* the ratio of effective values

$$\frac{W_{eff}}{L_{eff}} \qquad (5$$

not the drawn value of (W/L). This is important to remember when a
ing formulas for quantities such as the nFET resistance R_n. If a ch
simulation CAD tool is being used, we tend to use the drawn values
let the program calculate the effective values. This is discussed lat
the book in the context of SPICE.

5.2.5 Active Contacts

An active contact is a cut in the oxide Ox1 that allows the first lay
metal to contact an active n+ or p+ region. This is shown in the cross-s
tional view of Figure 5.12(a). These are defined by the **Active Co**
Mask with the general overlay shown in Figure 5.12(b). Since the con
is placed to fall inside of an n+ or p+ region, it is subject to the surro
design rule

$s_{a\text{-}ac}$ = minimum spacing between Active and Active Contact

The dimensions of the contact are given by

$d_{ac,\,v}$ = vertical size of the contact

$d_{ac,\,h}$ = horizontal size of the contact

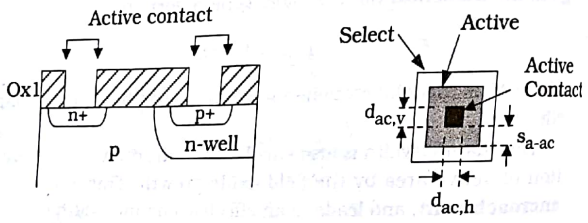
which are exact specifications. A square contact is obtained if

$$d_{ac,\,v} = d_{ac,\,v} = d_{ac} \qquad (5.$$

but it is not uncommon to have aspect ratios other than 1:1.

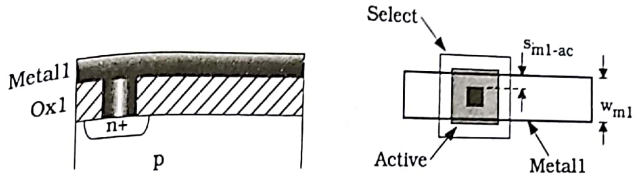
5.2.6 Metal1

Metal1 is applied to the wafer after the Ox1 oxide. It is used as interco
nect for signals and also for power supply distribution. Figure 5.13
shows the cross-sectional view of a first-layer metal line with an acti



(a) Cross-section (b) General mask set

Figure 5.12 Active contact formation



(a) Cross-section (b) General mask set

Figure 5.13 Metal1 line with Active Contact

contact to an n+ region. The contact cut through the oxide has been filled with a plug as described in the previous chapter. The mask set for this arrangement is drawn in Figure 5.13(b) with the **Metal1** mask feature overlapping the Active Contact to attain the electrical connection. The two design rules indicated in the drawing are

w_{m1} = minimum width of a Metal1 line

and

$s_{m1\text{-}ac}$ = minimum spacing from Metal1 to Active Contact

In addition, the metal has a minimum spacing rule value of $s_{m1\text{-}m1}$ which is not shown.

Every contact is characterized by a resistance

R_c = contact resistance Ω

due to the metal connections. To limit the overall resistance, it is common to use as many contacts as the design rules permit. An example is shown in Figure 5.14. Since the contacts are all in parallel, the effective resistance of the Metal1-Active connection with N contacts is reduced to

$$R_{c,\,eff} = \frac{1}{N} R_c \qquad (5.14)$$

In the example, $N = 16$ so that the effective resistance of the connection is (1/16) the value of a single contact. These also spread the current flow.

Metal1 allows access to the active regions of MOSFETs using the Active

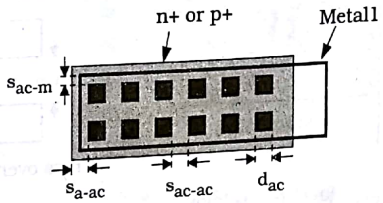
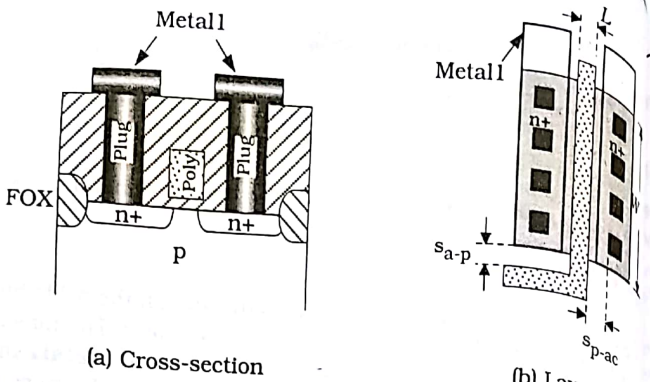


Figure 5.14 Multiple contacts to reduce contact resistance



(a) Cross-section (b) Layout

Figure 5.15 Drain and source FET terminals using Metal1

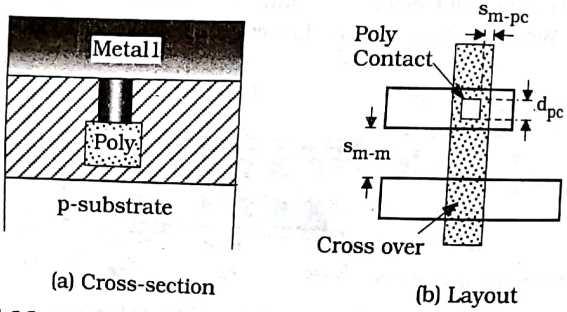
Contact oxide cut. Drain and source terminals are usually at the Me level as shown for an nFET in Figure 5.15(a). The corresponding layo provided in Figure 5.15(b) where we have included the design rules

$s_{p\text{-}ac}$ = minimum spacing from Poly to Active Contact

$s_{a\text{-}p}$ = minimum spacing from Active to Poly

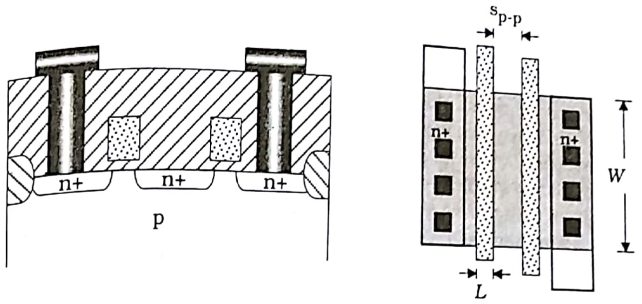
The first parameter is a surround-type specification to insure that Active Contact does not destroy any of the polysilicon gate. The sec spacing distance $s_{a\text{-}p}$ is due to the self-aligned FET sequence; it insu that the FET has the proper dimensions even if the Poly mask is not p fectly registered with the existing Active pattern on the wafer.

A Poly Contact mask is used to allow electrical connections betwe Metal1 and the polysilicon gate. Figure 5.16(a) is a cross-sectional view the contact between the two layers. The Poly Contact mask defines t oxide cut as indicated by the "empty" square shown in the upper part



(a) Cross-section (b) Layout

Figure 5.16 Poly Contact



(a) Cross-section (b) Layout

Figure 5.17 Series-connected FETs

the layout in Figure 5.16(b). In the lower portion of the layout, the Metal1 and Poly layers are not connected. This "crossover" characteristic is useful for routing the wiring.

As a final example, let us construct a pair of series-connected FETs. Figure 5.17(a) shows the cross-sectional view for two nFETs. Series wiring is achieved by sharing the central n+ region; since n+ is a reasonable conductor, no additional wiring is needed between the two devices. The layout in Figure 5.17(b) uses this observation: the series transistors are created by parallel Poly lines. The important design rule spacing is

$s_{p\text{-}p}$ = minimum Poly-to-Poly spacing

To obtain a pair of parallel-connected FETs, we add the contacts shown in Figure 5.18. The spacing $s_{g\text{-}g}$ shown in the drawing is the distance

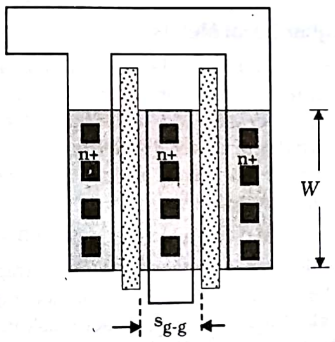


Figure 5.18 Parallel-connected nFETs

between the two gates. It is not a design rule, but can be written in of the basic design rules presented thus far as

$$s_{g\text{-}g} = d_{ac} + 2\, s_{p\text{-}ac}$$

since we must allow for the size of the contact itself, plus two u poly-active spacing. This is not a general rule that can be applied to process. In some submicron design sets, the poly-to-poly spacin applies regardless of the situation; contacts can be added between th gates without increasing the separation.

Another design rule enters the picture when we use a common area to create FETs with different values of *W*. This is shown in F 5.19 for two series-connected nFETs with channel widths $W_2 > W_1$ poly-active spacing $s_{p\text{-}a}$ is between the edge of a gate and a change active border. It must be enforced twice in this design since both FE the change.

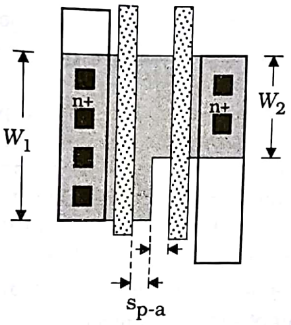


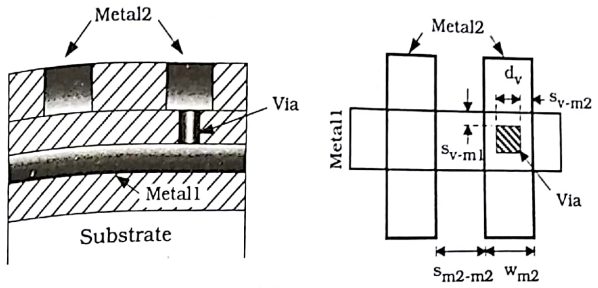
Figure 5.19 Different channel widths using the same active region

5.2.7 Vias and Higher Level Metals

Although simple circuits can be created in a single-poly, single-metal p cess, interconnect routing becomes very difficult in complex netwo Modern CMOS processes add several additional layers of metal that be used for signal and power distribution. We will label the layers acco ing to the order in which they are added. For example, in a 4-metal p cess the layering sequence would be

$$\text{Metal1} \rightarrow \text{Metal2} \rightarrow \text{Metal3} \rightarrow \text{Metal4}$$

CVD oxide is deposited between layers making each electrically distin Connection between adjacent layers is accomplished using a Via ma This is equivalent to an Active Contact mask in that it defines the locati of oxide cuts; the cuts are filled with a plug material that gives an elec cal contact between the two metals.



(a) Cross-section (b) Layout

Figure 5.20 Metal1-Metal2 connection using a Via mask

Figure 5.20(a) illustrates the use of a via to connect Metal1 to Metal2. Without a Via (as on the left side of the drawing), the two metal layers are electrically separate. The Via on the right side of the cross-sectional view provides the connection between the two layers. The mask layout is shown in Figure 5.20(b). The new design rule quantities shown are

d_v = dimension of a Via (may be different for vertical direction)

w_{m2} = minimum width of Metal2 feature

$s_{m2\text{-}m2}$ = minimum spacing between adjacent Metal2 features

$s_{v\text{-}m1}$ = minimum spacing between Via and Metal1 edges

$s_{v\text{-}m2}$ = minimum spacing between Via and Metal2 edges

Vias between other metal layers are similar. We note that the values of w_{mj} and $s_{mj\text{-}mj}$ for the j-th metal layer vary for $j > 1$ as the topology and roughness of the wafer surface often dictate that wider lines be used.

5.2.8 Latch-up Prevention

Latch-up is a condition that can occur in a circuit fabricated in a bulk CMOS technology. When a chip is in a state of latch-up it draws a large current from the power supply but does not function in response to input stimuli. A chip may be operating normally and then enter a state of latch-up; in this case, removing and the reconnecting the power supply may restore operations. In the worst-case situation, the chip may enter latch-up when power is applied and never be functional. If the current flow is too large, heat dissipation will destroy the die.

Figure 5.21 shows the current flow path when the chip is in latch-up. Under proper conditions, the path has a very low resistance and can allow large currents to flow. The key to understanding latch-up is noting that the bulk technology gives a **4-layer pnpn** structure between the power supply VDD and ground. This structure, shown in Figure 5.22(a), has the

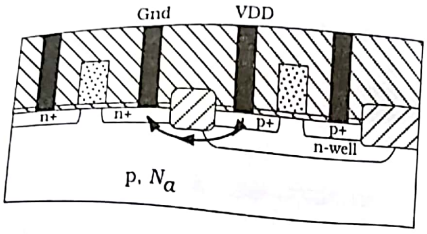


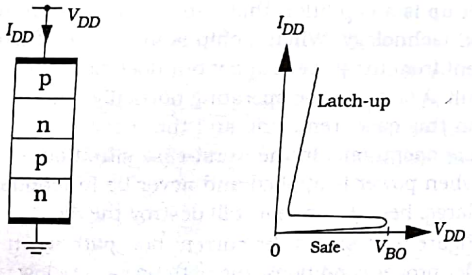
Figure 5.21 Latch-up current flow path

current-voltage dependence shown in Figure 5.22(b). For small voltages V_{DD}, the current I_{DD} is small because of the blocking characteristics of the pn junctions. However, if V_{DD} reaches the **breakover voltage** V_{BO}, the blocking is overwhelmed by internal electric fields. This admits large currents as shown in the drawing, indicating that the chip has entered latch-up state.

Latch-up prevention starts at the physical design level with various rules used to avoid the formation of the current flow path. One idea is quite simple. Since the current must flow through the n-well and the substrate, we can place VDD and ground connections at many different points to steer the current out of the "bad" path. This gives us the general rules

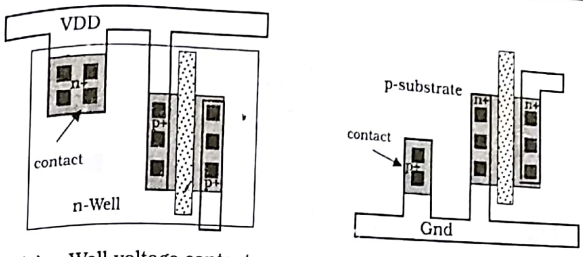
- Include an n-Well contact every time a pFET is connected to the power supply V_{DD}, and
- Include a p-substrate contact every time an nFET is connected to a ground rail.

Since the electrical connections must be made anyway, it is a simple matter to remember to include them. These are illustrated in Figure 5.23, and are very effective for avoiding latch-up. Other techniques have been



(a) Structure (b) Behavior

Figure 5.22 Characteristics of a 4-layer pnpn device



(a) n-Well voltage contact (b) Substrate ground contact

Figure 5.23 n-Well and substrate contacts for latch-up prevention

developed, and one should always check the design rule guidelines on how latch-up is to be avoided.

Non-bulk CMOS technologies that do not build the transistors directly on a silicon substrate avoid latch-up problems by not having the pnpn layering. This is true of silicon-on-insulator (SOI) designs. Alternately, using two separate wells for FETs, an n-well for pFETs and a p-well for nFETs, helps resist the formation of the current flow path. These **twin-tub** technologies are popular in advanced processing lines.

Since latch-up is induced by a high voltage, one must exercise special caution when designing circuits that have high levels of induced electrical "noise" such as a data receiver circuit. Information on avoiding these types of problem is also included in the design rule set. A new designer doesn't always worry about latch-up until a chip fails because of it; from that point on, the problem receives the respect it deserves!

5.2.9 Layout Editors

Several important aspects of layout have been presented in this section. The more critical items are summarized below for future reference.

- n+ is formed whenever Active is surrounded by nSelect; this is also called ndiff.
- p+ is formed whenever Active is surrounded by pSelect; this is also called pdiff.
- an nFET is formed whenever Poly cuts an n+ region into two separate segments.
- a pFET is formed whenever Poly cuts a p+ region into two separate segments.
- No electrical current path exists between conducting layers (n+, p+, Poly, Metal, etc.) unless a contact cut (Active Contact, Poly Contact, or Via) is provided.

These simple observations provide the basis for most of the layout problems we will encounter.

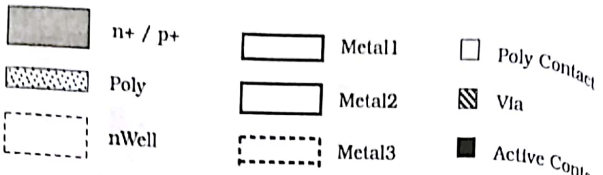


Figure 5.24 Layer key for layout drawings in this book

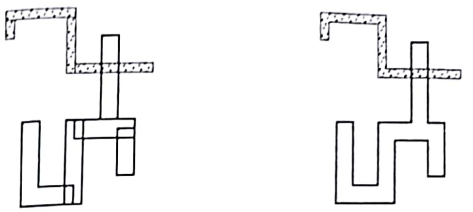
A layout editor visually distinguishes among the layers by defining d ferent colors and/or fill patterns for each. We have opted to use sim gray-scale and linewidth variations here to save the cost involved in pr ing the book in color.[3] Figure 5.24 shows the outlines that we will us identify layers in the book. Note that n+ and p+ regions have the sa shading, so that the polarity of a region is implied by where it is located will be a p+ layer in an nWell, an n+ section otherwise.

Every layout editor operates in a slightly different manner, but all ha the same basic features. In general,

- One enters a polygon by first choosing the desired layer of material and then using the drawing tools to shape the object as needed.
- Layout editors provide a background grid. The distance between eac grid point is a specified distance.
- The layers may be drawn in any order, so long as each polygon is properly identified by layer color/name/pattern. The database auto matically keeps track of the polygons drawn on each layer.
- The layout pattern is used to create the mask set for the process, and constitute the drawn dimensions.
- Design rules must be obeyed and the spacing must be checked before the drawing is complete.
- Polygons on a given layer may be drawn to touch or overlap. Only the outline is important. This is illustrated in Figure 5.25. The entire lay out in Figure 5.25(a) is drawn using rectangles, but results in the fin ished masks shown in Figure 5.25(b). This simplifies the overall layout process.

Always save your designs in a timely fashion! When the chip is completed it is usually put into a standard format for transmission to the processing line. Keeping in the spirit of the pioneers of chip design, the process is called **tape-out** because the files were transferred to the fabrication group on magnetic tape. The most common format used is probably the GDS standard which was a standard of one of the early minicomputer-based

[3] Which would quadruple the cost of the book!



(a) Multiple polygons (b) Mask pattern

Figure 5.25 Drawing complex polygons using rectangles

CAD systems. Academic users often produce files in CIF (Caltech Intermediate Form) format which was developed in the 1970's.

5.3 Cell Concepts

Digital VLSI chips are based on the idea of hierarchical design. Individual transistors are used to build gates, which are then used to create logic cascades and functional blocks, which in turn are used as the basis for even larger units. The basic building blocks in physical design are called **cells**. A cell may be as simple as a FET, or as complex as an arithmetic logic unit (ALU). Regardless of the internal complexity, every cell acts in the same manner: it may be used as a component to create a larger logic network.

The main idea of **cell-based** design is straightforward to visualize. Suppose that we start with a set of CMOS logic gates (NOT, NAND2, NOR2) and design the physical circuit layout for each. At the basic level, we concentrate on placing polygons for each layer with the required sizes. We then "step back" and view the gates as portrayed in Figure 5.26; each block is an independent cell. At this level in the design hierarchy, we do not care about the internal details. Only the external characteristics of a

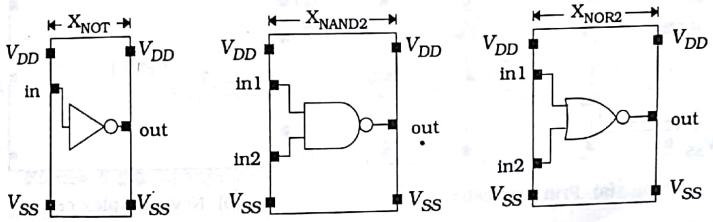


Figure 5.26 Logic gates as basic cells

gate are important, so we have replaced all of the layout by an equiva
logic symbol. In the examples shown, input and output terminals a
shown as **ports** into the cell. A port allows access to the interior circuit
Also note that a cell needs power supply ports for VDD and VSS that
chosen to be at the same locations for every cell. Finally, the width of ea
cell is shown as X_{NOT}, X_{NAND2}, and X_{NOR2} for the NOT, NAND2, a
NOR2, respectively. The numerical values depend on the transistor siz
and wiring used at the physical level.

Once a set of cells are defined, they may be used to create more co
plex networks. Suppose we want a cell that provides the function

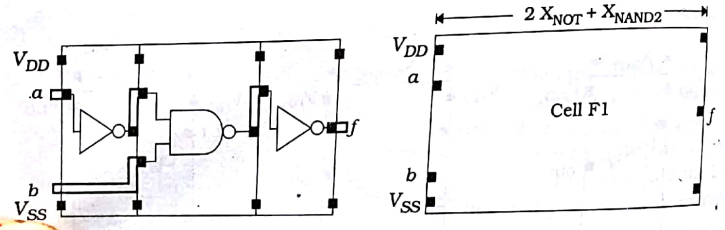
$$f = \bar{a} \cdot b \tag{5.16}$$

This can be created using the simple cascade of two NOT gates and on
NAND2 gate in Figure 5.27(a). Metal1 lines have been used to wire th
ports of the cells as needed. For example, the output of the first NOT gat
is wired to In1 of the NAND2 gate. Once the cascade has been created, we
can define a new cell F1 as on Figure 5.27(b). This cell has a total width o

$$2X_{NOT} + X_{NAND2} \tag{5.17}$$

which is just the sum of the widths of the three cells used to construct it.
Once defined, the new cell F1 can be used as a building block without
decomposing it into the primitive cells that were used to create it. It
becomes as basic as the NOT, NAND2, and NOR2 circuits. Using this hier-
archical design approach allows us to design and construct extremely
complex logic networks. It is, in fact, one of the most important tech-
niques to learn in VLSI.

Let us now turn our attention to the problem of creating a basic collec-
tion of cells at the physical level. The first item that we should investigate
is the placement of the power supply lines VDD and VSS. The problem is
shown in Figure 5.28. Both are shown on the Metal1 layer. The spacing
between the two lines is shown as



(a) Primitive cells (b) New complex cell

Figure 5.27 Creation of a new cell using basic units

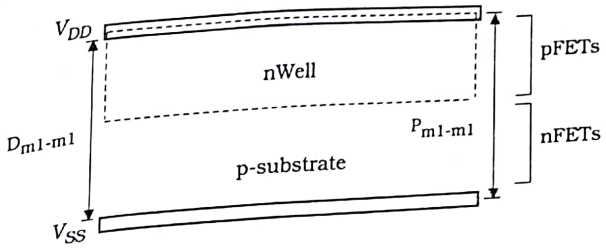


Figure 5.28 VDD and VSS power supply lines

$D_{m1\text{-}m1}$ = Edge-to-edge distance between VDD and VSS

and the **pitch**

$P_{m1\text{-}m1}$ = Distance between the middle of the VDD and VSS lines

The two are related by

$$P_{m1\text{-}m1} = D_{m1\text{-}m1} + w_{DD} \tag{5.18}$$

where w_{DD} is the width of the power supply lines.[4] Fabrication specialists often use the pitch specification, while the actual distance D between the edges is more useful for circuit layout. The nWell region that is used for pFETs is placed about the VDD line as shown. The region around VSS is kept as p-substrate since nFETs are connected to it.

Once we have established the VDD and VSS lines, we can proceed to place FETs between them. Figure 5.29 shows two different approaches to transistor orientation. The FETs on the left side of the drawing are ori-

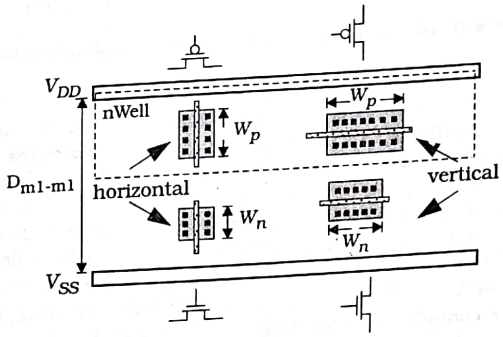
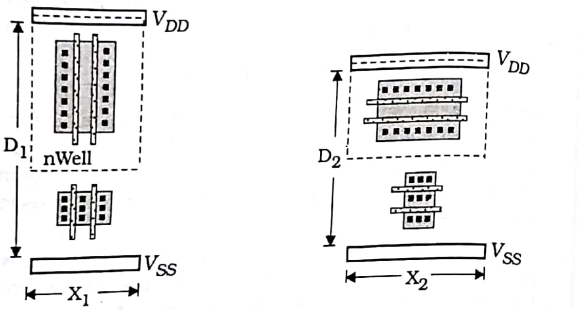


Figure 5.29 MOSFET orientation

[4] Note that w_{DD} may be larger than the minimum design rule width w_{m1} allowed for a Metal1 line.

ented with the drain and source running in the horizontal direction. In this case, the FET channel widths W_n and W_p are limited by $D_{m1\text{-}m1}$ and the n-well size. If the FETs are rotated 90 degrees to the vertical orientation shown on the right side, then the channel widths W_n and W_p may be chosen to be any size needed. However, the width of the cell may get large. Since we want to choose a set value of $D_{m1\text{-}m1}$ that is used for every cell, we should investigate the effect of the FET placement on the cell dimensions.

The trade-offs are shown in Figure 5.30. Horizontally oriented transistors are used in Figure 5.30(a). In this case, we would want to make D large enough to accommodate the most complex logic gate needed. Using vertical FETs, the value of D_2 shown in Figure 5.30(b) can be made smaller than D_1. The difference is in the horizontal widths of the cell. In general, we would expect X_2 to be greater than X_1 for a given circuit.



(a) Horizontal FETs (b) Vertical FETs

Figure 5.30 Effect of FET orientation on cell dimensions

The shape of the cells affects how the cells fit together in logic cascades and determines what the more complex units may look like. Piecing the cells together is called **tiling** since the cells themselves look like non-uniform **tiles**. Figure 5.31(a) illustrates a simple cascade created out of four tiles for a large value of D. This gives an overall cell grouping that is relatively narrow compared to that shown in Figure 5.31(b) for a smaller value of D. In that case, the grouping is short, but quite wide.

Interconnect routing considerations are also important considerations for the VDD-VSS spacing. In complex digital systems, the wiring is often more complicated than designing the transistor arrays. One approach to this problem is to place rows of logic cells in parallel and allocate space in between the rows for wiring. The general idea is portrayed in Figure 5.32. Metal1 lines running parallel to the logic rows can be used to route signals as required. Since Metal2 lines can cross over Metal1, vertical lines

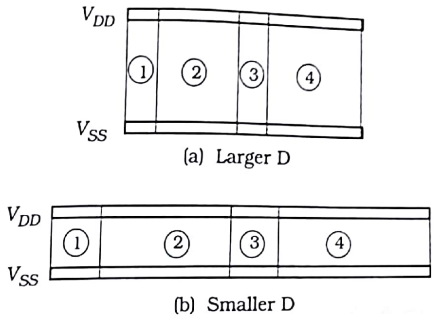


Figure 5.31 Effect of tile shapes on larger cells

can be used to connect logic cells to the Metal1 interconnect as shown. This technique is often found in ASIC designs because it allows a significant amount of freedom for different designs. The main drawback is that the logic density is relatively low compared to close-packed layouts.

An alternate high-density technique is to alternate VDD and VSS power lines and share them with cells above and below. This results in the **Weinberger image** shown in Figure 5.33. The "Inverted logic cells" are defined to be flipped in relation to the rows of "Logic cells" above or below. This is because they have VSS at the top and VDD at the bottom. The

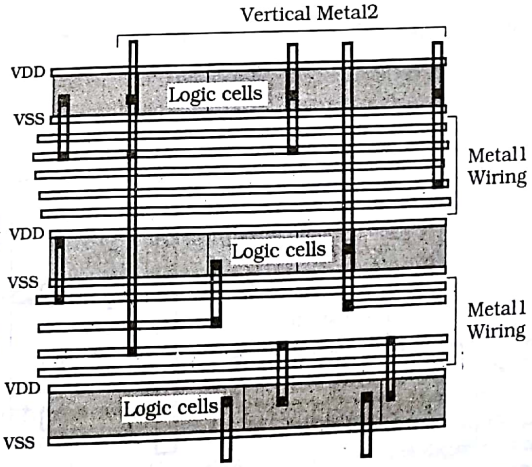


Figure 5.32 Wiring channels

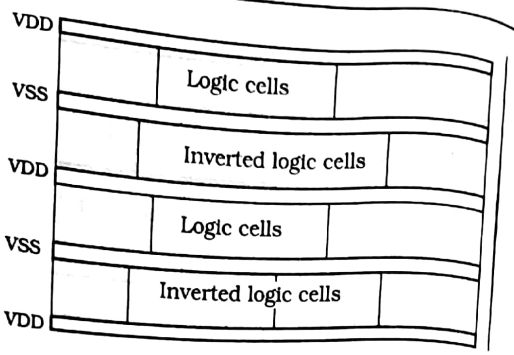


Figure 5.33 Weinberger image array

details of FET placement in a Weinberger image are provided by the cl up in Figure 5.34. The nWell regions surround the VDD rails and al pFETs to be created above or below the power lines. The nFETs placed on both sides of the VSS line. Since no space is automatic reserved for wiring, this scheme allows for high-density placement of cells. The main drawback is that the connections between rows must accomplished by using Metal2 or higher, since Metal1 is already de nated for the power supplies. It may be possible to use horizontal Me interconnect lines within a row if there is sufficient room.

Port Placement

The input and output ports of a cell must be placed at convenient poi to facilitate the interconnect wiring. At the basic level, we view logic c cuit inputs as being to the gate terminals of MOSFETs, while the outpu

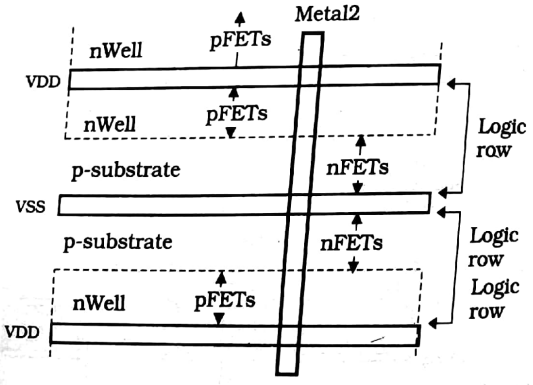


Figure 5.34 FET placement in a Weinberger array

are metal interconnect lines. Since FET gates are at the polysilicon level, we must provide a poly contact to connect the output of a cell to the input of another cell.

Figure 5.35 shows the case where the ports are placed around the periphery of a cell. With this simple view, the input poly lines are on the left side and include a Metal1 pad and poly contact. The output on the right side is at the Metal1 level, which allows cell interconnects to be completed on the same level. Vertical poly inputs are also shown. These are useful if the layout uses wiring channels between cell rows as in Figure 5.32.

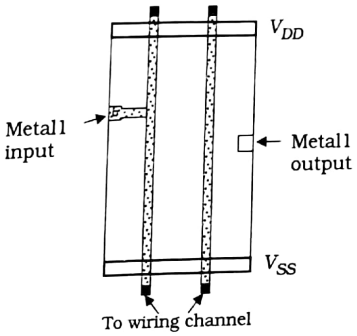


Figure 5.35 Port placement in a cell

There are no *a priori* constraints on the placement of cell ports, and interior ports are also used in practice. The most important factor is to insure that the cells can be wired together as needed in a complex design. Wiring problems have a tendency of appearing at critical times. Careful cell planning and a reliable CAD tool set helps to solve them more efficiently.

Now that we have learned the basics of logic cells, let us study the details of designing a set of CMOS gates at the silicon level. Once we have a reasonable set of gates, we can progress into the next hierarchical design level where we build up more complex units.

5.4 FET Sizing and the Unit Transistor

Field-effect transistors are specified by the aspect ratio (W/L) where W is the channel width and L is the channel length. In modern VLSI, both are on the order of microns [μm], with specific numerical values established in the layout of the masks. These dimensions combine with the processing parameters to give the electrical characteristics of the transistor.

Consider the basic FET drawn in Figure 5.36. The drawn value
channel length and width are shown explicitly. We may estimate
the layout-dependent electrical properties of the transistor by using
simple formulas. First, the area A_G of the gate is defined to be the
of the poly that is over the channel region. The drawing shows th
area A_G of the gate is given by $A_G = LW$. The gate capacitance C_G
looking into the gate terminal (labeled as G in the drawing) is then
by

$$C_G = C_{ox}WL \tag{5}$$

where we recall that C_{ox} is the oxide capacitance per unit area.

Now let us examine the current flow through the device from the d
(D in the drawing) to the source (labeled S). The current into the dra
denoted by I_D, while the current out of the source is I_S such that

$$I_D \approx I_S \tag{5}$$

is a reasonable approximation. This says that the current flows fr
drain to source using the channel region, which is underneath the ga
The channel itself has a resistance R_{chan} [Ω] that impedes the flow of c
rent. If the channel were modeled as a simple rectangular block, then
resistance could be approximated as

$$R_{chan} = R_{s,c}\left(\frac{L}{W}\right) \tag{5.2}$$

where $R_{s,c}$ is the sheet resistance of the channel region. Unfortunate
FETs are not that simple and computing the drain-to-source resistance
more complicated. The equation does, however, agree with the more rig
ous analysis in that it predicts that R_{chan} is inversely proportional to t
channel width W:

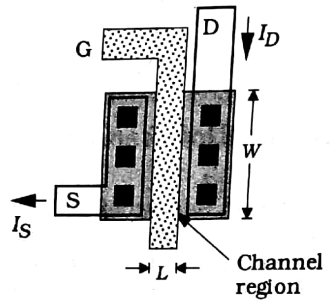


Figure 5.36 Basic geometry of a FET

$$R_{chan} \propto \frac{1}{W} \tag{5.22}$$

This says that increasing W decreases the resistance, which allows more current to flow. The channel dimensions thus establish the resistance and capacitance of a FET.

One other feature is worth mentioning. The primary difference between an nFET and a pFET is the polarity of charge that gives the current. An nFET uses negatively charged electrons, while a pFET relies on positively charged holes. Recall, however, that electrons can move more easily than holes. This is expressed by the relation

$$\mu_n > \mu_p \tag{5.23}$$

that was introduced earlier in Section 3.2 of Chapter 3. In this equation, μ_n and μ_p are the electron and hole mobilities, respectively. A high value of mobility implies that the particle is "more mobile" than a low-mobility particle. Suppose we design an nFET and a pFET with the same aspect ratio (W/L). Since electrons have a higher mobility, the nFET resistance R_n would be smaller than the pFET resistance R_p. Let us define the **mobility ratio** r by

$$r = \frac{\mu_n}{\mu_p} \tag{5.24}$$

In modern CMOS processing, the mobility ratio $r > 1$ is usually between 2 and 3, with the actual value set by the doping densities and other physical considerations. The resistance is inversely proportional to the conductivity, which is proportional to the mobility. We can thus conclude that R_n and R_p for equal size FETs are related by

$$\frac{R_p}{R_n} = r \tag{5.25}$$

This is often stated in the literature by saying that pFETs don't conduct as well as nFETs. Alternately, since electrons travel faster than holes, we conclude that nFETs are faster than pFETs. Both statements assume that the transistors being compared are the same size.

The resistance of a FET can be adjusted by changing the channel width W. Suppose that we have an nFET with an aspect ratio of $(W/L)_n$ that gives a resistance R_n. To design a pFET with the same resistance value $R_p = R_n$, we use an aspect ratio of $(W/L)_p > (W/L)_n$ that compensates for the differences in mobilities. This is accomplished by selecting

$$\left(\frac{W}{L}\right)_p = r\left(\frac{W}{L}\right)_n \tag{5.26}$$

With this design, the resistances are equal. Note, however, that the gate areas are different with $A_{Gp} > A_{Gn}$ due to the increased channel width of the pFET. Assuming that the channel lengths are the same, this gives different gate capacitances such that

$$C_{Gp} = rC_{Gn} \tag{5.}$$

since the areas are proportional to W.

Example 5.1

Consider an nFET with an aspect ratio of $(W/L)_n = 4$ that is constructed in a process where $r = 2.4$. To create a pFET with the same resistance, we must select

$$\left(\frac{W}{L}\right)_p = 2.4(4) = 9.6 \tag{5.}$$

In practice, we might use the nearest integer value $(W/L)_p = 10$. The gate capacitance of the pFET would be larger than that of the nFET by the same ratio

$$C_{Gp} = 2.4C_{Gn} \tag{5.}$$

It is also worth mentioning the obvious fact that the pFET will consume more surface area than the nFET.

The electrical characteristics of transistors determine the switching speed of a VLSI circuit. At the physical level, this translates to selecting the aspect ratios $(W/L)_n$ and $(W/L)_p$ for every FET in the circuit. Once the sizes have been determined, the physical design problem revolves around designing the silicon circuit using the specified aspect ratios. Let us concentrate on the physical design problem for now. Many of the remaining sections of this book are concerned with how to choose the transistor sizes for high-speed logic networks.

A useful starting point for circuit layout is to define a **unit transistor**. This is a FET with a specified aspect ratio (W/L) that can be replicated as needed in the layout. Since it only needs to be drawn once, layouts can be completed much more quickly than if the designer had to construct every transistor. Moreover, since the electrical characteristics of device will be known, the switching performance analysis will be straightforward.

One choice for a unit transistor is the **minimum-size MOSFET**. As implied by its name, a minimum-size FET is the smallest transistor that can be created using the design rule set. An example is shown in Figure 5.37. The drawn channel length L is the minimum allowed poly width w_p while the drawn channel length W is the minimum width w_a allowed for a

feature on the Active mask. The aspect ratio for the device is thus

$$\left(\frac{W}{L}\right)_{min} = \frac{w_a}{w_p} \tag{5.30}$$

as can be verified by inspection. The gate capacitance is set as

$$C_G = C_{ox}w_a w_p \tag{5.31}$$

since the gate area is just $A_G = w_a\, w_p$. The minimum-size device is the smallest transistor, so that in theory it allows the highest packing density. However, it does have the largest resistance of any FET, so it may not be the best choice for every circuits.

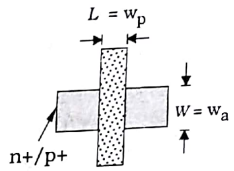


Figure 5.37 Geometry of a minimum-size FET

The basic minimum-size FET shown in Figure 5.37 does not have any contacts. If we add Active Contacts to allow Metal1 connections, then our dimensions may change. Consider the modified layout in Figure 5.38(a). The channel length is still given by $L = w_p$. However, since we have used Active Contact cuts in the oxide, the design rules

d_c = dimension of the contact

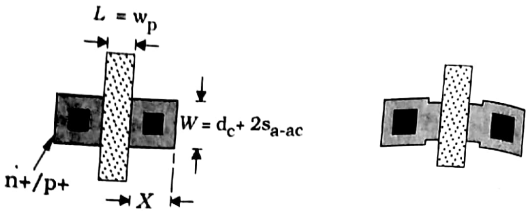
$s_{a\text{-}ac}$ = spacing between Active and Active Contact

must be applied. As shown in the drawing, the minimum width is now

$$W = d_c + 2s_{a\text{-}ac} \tag{5.32}$$

In some processes, this value may be the same as $W = w_a$. If not, then the Active region can be enlarged to accommodate the contact as in Figure 5.38(b). This allows us to have $W = w_a < d_c + 2s_{a\text{-}ac}$. Although minimum-size FETs are slow due to their high resistance, they can be useful in situations where slow switching is not a critical concern.

Once a unit FET has been selected, it is useful to allow it to be **scaled** in size. In Figure 5.39, the 1X transistor is used as the reference basis. Larger transistors are obtained by multiplying the width; 2X and 4X versions are shown in the drawing. Altering the size of the transistor changes its resistance and capacitance. Let us denote the resistance and gate capacitance of the 1X device by R_{1X} and C_{1X}, respectively. If the width of



(a) Active contact (b) Small w_a

Figure 5.38 Minimum-size FETs with Active Contact features

the 1X device is W_{1X}, then we can create larger FETs using a **scaling fa**
tor $S \geq 1$ such that

$$W_{SX} = SW_{1X} \tag{5.3}$$

For example, setting $S = 4$ gives

$$W_{4X} = 4W_{1X} \tag{5.3}$$

which describes the 4X transistor. The resistance and capacitance of
scaled FET are changed because they depend upon the size of the devic
Applying the scaling transformations gives the general values

$$R_{SX} = \frac{R_{1X}}{S} \qquad C_{SX} = SC_{1X} \tag{5.3}$$

For example, the 2X FET has

$$R_{2X} = \frac{R_{1X}}{2} \qquad C_{2X} = 2C_{1X} \tag{5.3}$$

which is easy to remember. Since pFETs have different conduction cha

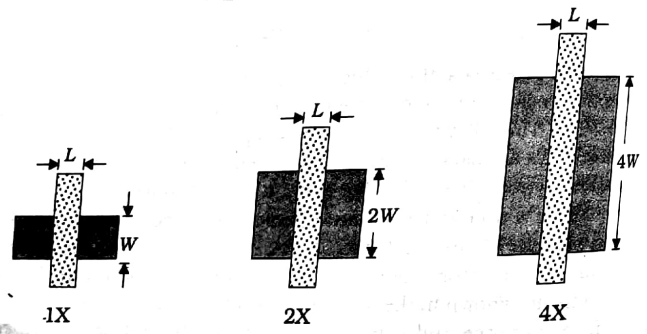


Figure 5.39 Scaling of the unit transistor

acteristics than nFETs, it is common to introduce unit transistors for each type. The scaling relations remain the same regardless of the polarity.

Unit devices are not restricted to individual transistors. It may be useful to define series and parallel groups of FETs as 1X units, and then scale using the same technique. Figure 5.40 shows an example of a series-connected 2-FET chain at 1X and 2X sizes. Since each transistor has been scaled in the same manner, the resistance and capacitance relations are still valid. It is important to note, however, that the total resistance of the series-connected transistors is the sum of the individual resistances. If the resistance of one 1X FET is R_{1X}, then the series group has a resistance of $2R_{1X}$. Since each FET in the 2X circuit has a resistance given by $(R_{1X}/2)$, the resistance of the scaled 2X series pair is

$$2\,(R_{1X}/2) = R_{1X} \tag{5.37}$$

by just adding. Series-connected FETs are usually made larger than individual FETs to reduce the overall end-to-end resistance.

Large transistors often require a bit more thought. There are occasions when aspect ratios reach 100 or greater. A single device with a large channel width W will have a long rectangular shape and may not easily fit into the overall layout. Or, the resistance of the gate material may slow down the signal.

The most common solution is to use a group of parallel-connected transistors. Figure 5.41 shows a group of transistors that is based on a channel width of W. The four gate lines are all connected together, and the wiring is routed to give an effective channel length of $4W$ between sides A and B. One advantage of this approach is that the overall layout geometry can be adjusted to square or nearly square shapes.

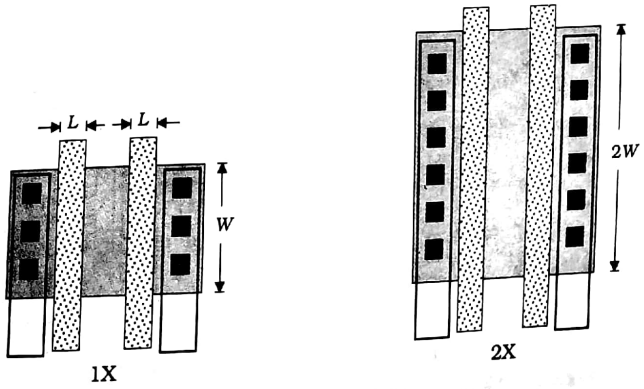


Figure 5.40 Scaling of series-connected FET chain

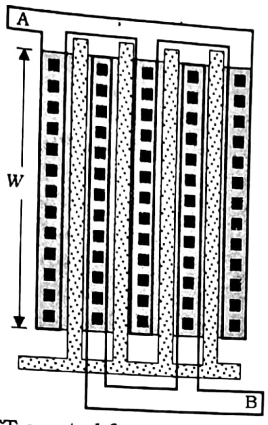


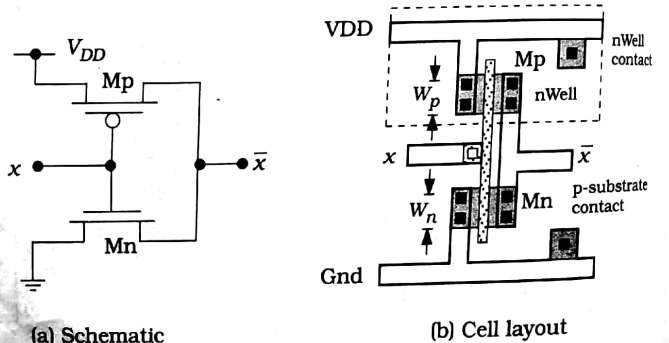
Figure 5.41 A large FET created from parallel-connected transistors

5.5 Physical Design of Logic Gates

Let us now apply the basics of the physical design process to the problem of constructing a set of layouts for basic CMOS logic gates. Each gate is classified as an individual cell. We will concentrate on a unit cell design since larger cells can be obtained by scaling.

5.5.1 The NOT Cell

The simplest CMOS logic circuit is the inverter that provides the NOT operation. Consider the schematic shown in Figure 5.42(a); the horizontal orientation can be directly translated to the layout in Figure 5.42(b). The layout shows the channel widths W_p and W_n of the FETs. It also shows the important connection of VDD to the nWell (which is the pFET bulk

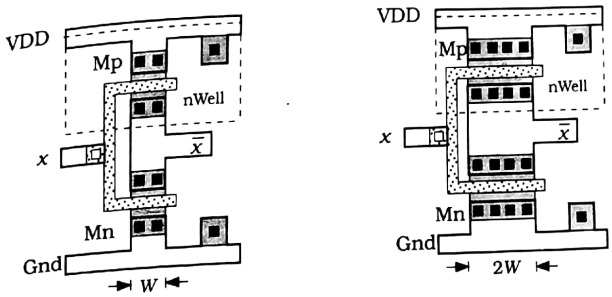


(a) Schematic (b) Cell layout

Figure 5.42 NOT gate with horizontal FETs

and Gnd to the p-substrate (which is the nFET bulk). These connections will not always be shown explicitly in our drawings, but must be included in every cell to produce a functional circuit.

Although this simple example illustrates the basic aspects of the layout, the small spacing of the VDD-Gnd Metal1 lines makes it difficult to scale. If we rotate the FETs by 90 degrees, then it is easier to increase the channel widths of the FETs. This is illustrated by the example in Figure 5.43. In Figure 5.43(a), the unit NOT design has aspect ratios of (W/L) for both Mn and Mp. The 2X cell in Figure 5.43(b) uses the same VDD-Gnd pitch, but provides transistors with aspect ratios of $2(W/L)$ by stretching the FETs in the horizontal direction.



(a) Basic cell (b) 2X cell

Figure 5.43 NOT layout using vertical FETs

Another example is shown in Figure 5.44. In this design, the pFET is larger than the nFET by a mobility ratio $r = 2.5$. This equalizes the resistance between the output $\bar{x}$ and the two power supply rails VDD and Gnd. Since this has equal nFET and pFET resistances, it is called a **symmetric inverter** (even though it is not geometrically symmetric). The significance of this design will be discussed in Chapter 7 in the context of the electrical design of CMOS logic gates.

5.5.2 NAND and NOR Cells

We can apply the same techniques to design the layout for a NAND gate. Vertical FETs are used in the NAND2 layout shown in Figure 5.45(a). In this design, all transistors have the same aspect ratio. They can be resized as needed, as can the cell itself. If more inputs are used, e.g., as in a NAND3 gate, then the sizing of the nFETs becomes more critical. In this case, the value of W_n should be increased to reduce the series resistance from the output to ground.

A NOR gate may be created in the same manner. The NOR2 layout in

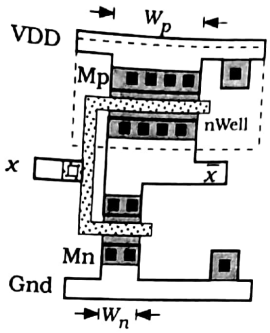
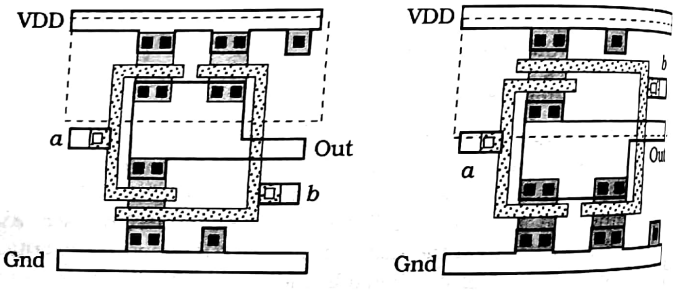


Figure 5.44 Layout for an electrically symmetric NOT gate

Figure 5.45(b) has been obtained by simply flipping the NAND2 la and redefining the FET polarities and power supply lines. This desig uses the same size FETs throughout. However, since pFETs have tively high resistance values, the series-connected pFET chain fro output to VDD may cause excessive switching delays. The delay ca shortened by using larger values for W_p.

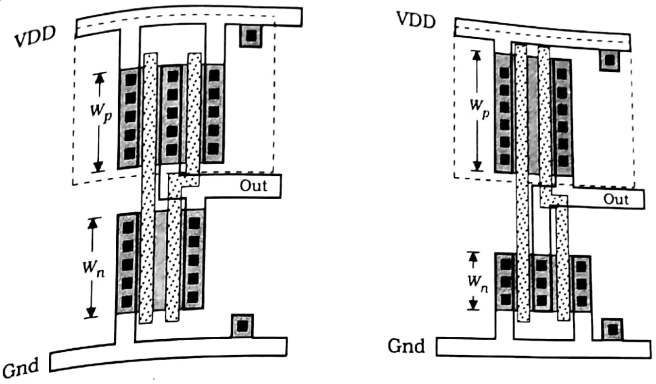
Alternate layouts for the NAND2 and NOR2 gates using vertically ning gate patterns are shown in Figure 5.46; the wiring for this appr was examined in Chapter 3. However, these drawings are more det in that they show the FET sizes. Both increase the channel width series-connected transistors in order to reduce the resistance. The nF in the NAND2 gate in Figure 5.46(a) are made wider than the parallel nected nFETs used in the NOR2 gate of Figure 5.46(b). Similarly, series pFETs in the NOR2 are wider than the parallel pFETs of NAND2.

The actual numerical values of W_n and W_p determine the electri



(a) NAND2 gate (b) NOR2 gate

Figure 5.45 NAND2 and NOR2 layouts using vertical FETs



(a) NAND2 (b) NOR2

Figure 5.46 Alternate NAND2 and NOR2 cells

characteristics of the gate. In many designs, conveniently sized FETs are used in the layout. The circuits are then simulated to determine their electrical response, and the sizes adjusted if needed. In critical data paths, the values are more important and initial design work concentrates on finding acceptable values. The electrical aspects of logic gate design are discussed in later chapters of the book.

5.5.3 Complex Logic Gates

Complex logic gate layout progresses in the same manner. The routing techniques presented in Chapter 3 give the device placement. In the physical design stage, every transistor size is specified and the FET structures are placed between the VDD and Gnd rails. Series-connected FETs are generally made wider than individual transistors unless they share the same Active area or other considerations are important.

An example of a complex logic gate is shown in Figure 5.47. Since both the nFET and pFET arrays share drain/source regions, single values of W_n and W_p are used for simplicity in layout. Note that the pFETs can be made wider within the given VDD-Gnd spacing to compensate for their higher resistance values.

5.5.4 Generalized Comments on Layout

These examples illustrate the basics for the physical design of logic gates using the following sequence:

- Design the MOSFET logic circuit;

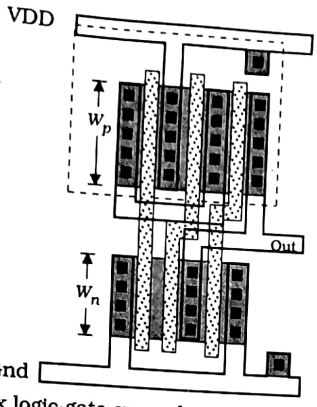


Figure 5.47 Complex logic gate example

- Use the transistor circuit to create a routing diagram where only wi ing paths and levels are important;
- Use the routing diagram as the basis for the final physical design o the gate that includes proper sizes for all features and adheres to th design rules.

The final aspect of gate design can be time consuming for the neophyt Some complex circuits are tricky even for an experienced designer. After layout is completed, it must be extracted and simulated using the proces parameters. Final acceptance of a cell depends on its satisfying all electri cal and size specifications.

VLSI designers attempt to place as much circuitry on a given area a possible. At the engineering level, this is accomplished by using the idea of regular, repeated patterns and mastering the CAD tools. The lessons learned from designing simple cells provide the basis for increasingly complex networks. Design automation tools are becoming quite powerfu and more intelligent, and are helping pave the way for designs of incredi ble complexity.

5.6 Design Hierarchies

VLSI systems are created using the concept of design hierarchies where simple building blocks are used to design more complex units. This nesting continues until the entire chip is complete. The code for a layout editor is structured to provide this type of environment for the chip designer. The key to creating the hierarchy lies in the concept of cells. We define a cell to be a collection of objects that is treated as a single entity. The characteristics of the objects themselves provide the hierarchical viewpoint.

The simplest cells consist of only polygons. The logic gates such as the NOT and NAND2 examples in the previous section fall into this category. A cell with this property is said to be **flat**; this means that every object is independent and not related to any other object. In a flat cell, we can alter any polygon without affecting anything else. To initiate the design process, we create a large number of flat cells and store them in a library. The most primitive library entries are chosen to be transistors and logic gates that can be used as building blocks in more complex designs. Figure 5.48 illustrates the idea. In this simple example, three gate-level designs nor2, nand2, and not are created at the polygon level. Each is then stored as a separate cell in the library. Each cell is independent of the others.

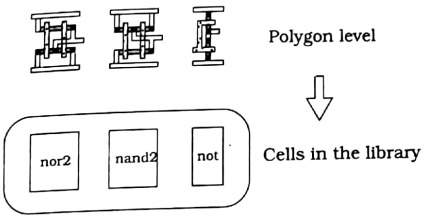


Figure 5.48 Primitive polygon-level library entries

Once the initial library is established, we can use the cell entries in our design by instancing them into our layout. An instance is a copy of the cell in the library. An instanced object cannot be altered in the new layout, as it is always an exact replica of the library entry. The only way to change the characteristics of an instance is to change the library entry. The most important concept to grasp is that the new layout will be a more complex object that can itself be stored as a cell in the library. In Figure 5.49, two new cells named cell_1 and cell_2 are designed using instances of the Primitive Library, plus polygons of their own. We may save the new cells and create a larger library group (Library 1) for use in more complex designs. This process may be repeated as needed. Useful functions are designed into new cells that become a part of the library, and are used to build other cells. The final cell collection chosen for the library should contain the great majority of the cells needed for the design projects.

The concept of cell hierarchy is based on the building of the cell library. Figure 5.50 provides visualization of the scheme. At the most primitive level, the cells consist only of polygons representing the material layers. This is designated as Level_1 in the hierarchy. Level_2 cells consist of polygons and instances of Level_1 cells. The next group is designated as Level_3 cells. These consist of polygons and may contain instances from Level_1 and Level_2 entries. The last group shown in the drawing are the

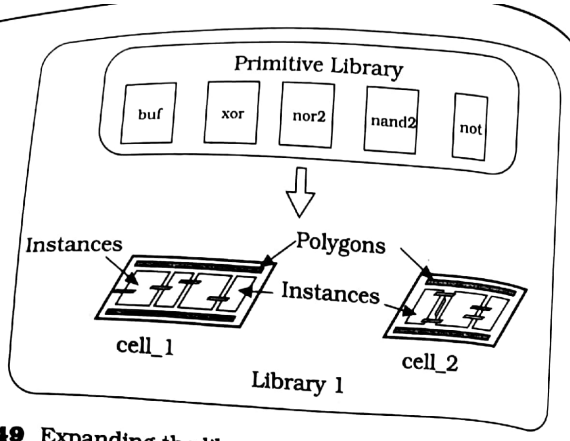


Figure 5.49 Expanding the library with more complex cells

Level_4 cells. They are made up of polygons and instances of any cells fr
Level_1 to Level_3.

It is important to remember that an instance is only a copy of a simp
entity and its internal structure cannot be altered at a higher level. F
example, if a Level_2 cell is instanced into a Level_4 cell, the Level_
design treats it as being invariant. To alter the Level_2 cell, one mu
return to the original Level_2 design. Any changes in the cell will prop
gate to all higher levels where the cell was instanced. In practice, a
library is used by a large number of designers, but most users do n

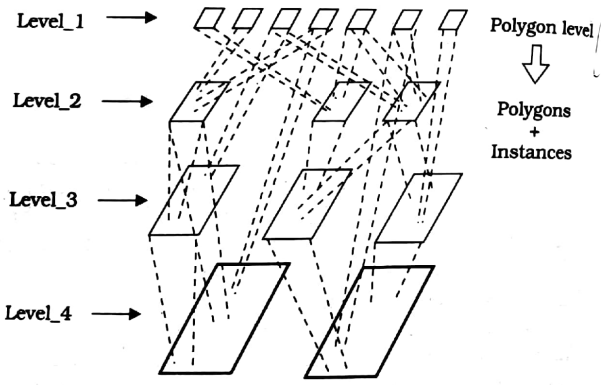


Figure 5.50 Cell hierarchy

have access privileges to the central group of cells. This prevents someone from changing a characteristic that may be critical in another's design.

Although the contents of an instance can be changed, it is possible to decompose it into polygons by the **flatten** command. After a cell is flattened, all references to the original cell are lost and individual features of the circuit can be modified. Figure 5.51 illustrates the effect of the flatten operation. A flattened cell cannot be restored to its original instanced form.

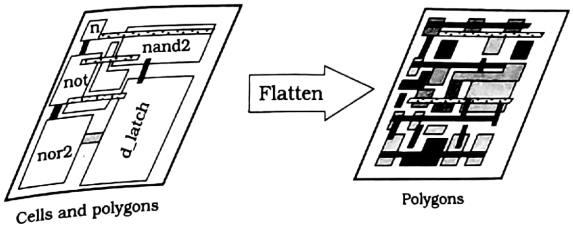


Figure 5.51 Effect of the flatten operation

The concept of design hierarchies is indispensable in VLSI engineering. It allows us to build up complex networks by starting at primitive levels and adding cells as deemed useful. In this manner, various libraries can be built and maintained for use in many different projects. Complex systems are broken down into manageable sections, and the concept of building chips with millions of transistors becomes a reality.

As a closing comment, note that the layout is process dependent. This means that a new library must be built every time a new fabrication plant goes on-line. Unless the new process is radically different, the old cells may be used as a starting point for the new group. Sometimes it is possible to simply scale the dimensions, which is the basis for the concept of cell **reuse**. This helps reduce the time needed for the new chip. Many current designs are created with reuse in mind.

5.7 References for Further Reading

[1] R. Jacob Baker, Harry W. Li, and David E. Boyce **CMOS Circuit Design, Layout and Simulation**, IEEE Press, Piscataway, NJ, 1998.

[2] H. B. Bakoglu, **Circuits, Interconnections, and Packaging for VLSI**, Addison-Wesley, Reading, MA, 1990.

[3] Kerry Bernstein, et al., **High-Speed CMOS Design Styles**, Kluwer Academic Publishers, Norwell, MA, 1998.

[4] Dan Clein, **CMOS IC Layout**, Newnes Publishing Co., Woburn, MA, 2000.

[5] Robert F. Pierret, **Semiconductor Device Fundamentals**, Addison-Wesley, Reading, MA, 1996.

[6] Bryan Preas and Michael Lorenzetti (eds.), **Physical Design Automation of VLSI Systems**, Benjamin/Cummings Publishing Co., Menlo Park, CA, 1988.

[7] M. Sarrafzadeh and C.K. Wong, **An Introduction to VLSI Physical Design**, McGraw-Hill, New York, 1996.

[8] Jasprit Singh, **Semiconductor Devices**, John Wiley & Sons, New York, 2001.

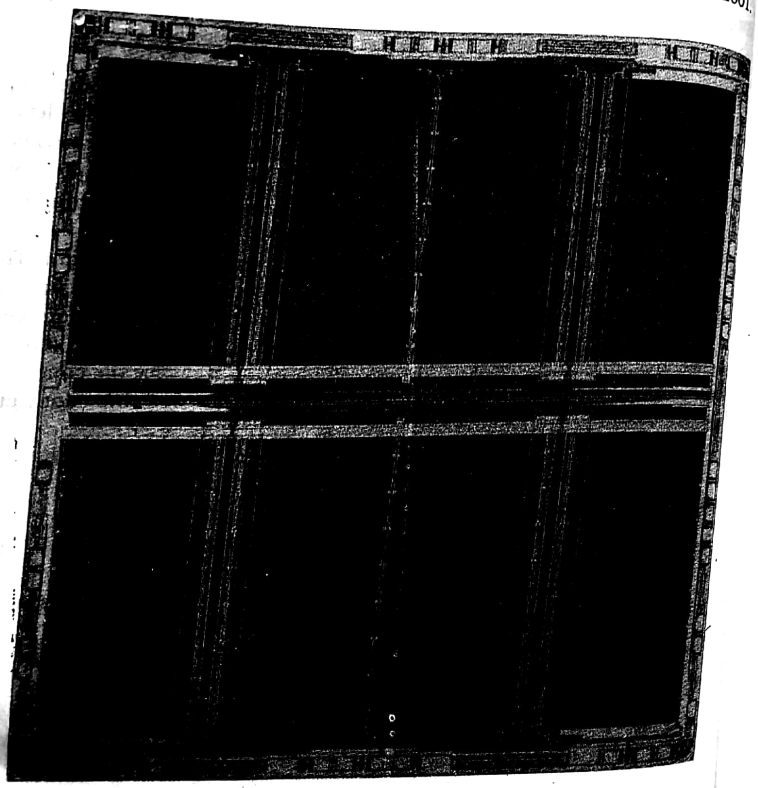
[9] Ben G. Streetman and Sanhay Banerjee, **Solid State Electronic Devices**, 5th ed., Prentice Hall, Upper Saddle River, NJ, 1998.

[10] R. R. Troutman, **Latchup in CMOS Technology**, Kluwer Academic Publishers, Norwell, MA, 1986.

[11] John P. Uyemura, **CMOS Logic Circuit Design**, Kluwer Academic Publishers, Norwell, MA, 1999.

[12] John P. Uyemura, **Physical Design of CMOS Integrated Circuits Using L-Edit™**, PWS Publishing Company, Boston, 1995.

[13] M. Michael Vai, **VLSI Design**, CRC Press, Boca Raton, FL, 2001.



Part 2

The Logic-Electronics Interface

Electrical Characteristics of MOSFETs

6

This chapter centers on MOSFET characteristics and initiates the "electronics" side of VLSI where electrical currents and voltages are the most important quantities. The emphasis, however, is not on studying electronics for its own sake, but to emphasize the link between physical design and logic networks.

6.1 MOS Physics

MOSFETs conduct electrical current by using an applied voltage to move charge from the source side to the drain side of the device. Since the drain and source are physically separate, the flow of charge underneath the gate can occur only if a conduction path, or **channel**, has been created.

Consider the nFET schematic symbol shown in Figure 6.1. The drain current I_{Dn} is controlled by the voltages applied to the device. The primary voltages are identified in the drawing as the gate-source voltage V_{GSn} and the drain-source voltage V_{DSn}. It is important to determine the current versus voltage (I-V) relation

$$I_{Dn} = I_{Dn}(V_{GSn}, V_{DSn}) \tag{6.1}$$

to obtain models for the device operation. Once this is accomplished, we

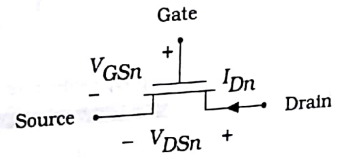


Figure 6.1 nFET current and voltages

will have both a physical understanding and a mathematical mod
analyzing and designing CMOS switching networks.

The starting point for our study is the simple MOS structure show
Figure 6.2; remember that the acronym "MOS" is used to describe
conductor-oxide-semiconductor layering even if the top layer is n
metal. In the present situation, the gate layer is the top conducting l
This drawing represents the central region of an nFET and provides
physics of how the conduction layer is formed between the drain a
source regions. The voltage applied to the gate is denoted by V_G and
assumed to be a positive value with the polarity shown. The oxide laye
taken to be silicon dioxide (SiO_2), which acts as an insulator between
gate and substrate. This gives the oxide capacitance per unit area
units of F/cm^2) as

$$C_{ox} = \frac{\varepsilon_{ox}}{t_{ox}} \qquad (6.$$

where t_{ox} is the thickness of the oxide in cm. We recall from Chapte
that the permittivity for silicon dioxide is $\varepsilon_{ox} = 3.9\ \varepsilon_o$ such that $\varepsilon_o = 8.85$
$\times 10^{-14}$ F/cm is the permittivity of free space. Oxide layers in moder
CMOS processing are very thin with $t_{ox} < 10$ nm $= 10^{-6}$ cm being typical

The value of C_{ox} determines the amount of electrical coupling tha
exists between the gate electrode and the p-type silicon region. The effe
is most pronounced at the **silicon surface**, i.e., the top of the silico
region. The coupling is described by an electric field E (with units of V/
cm) that is created in the insulating oxide layer when a voltage is applie
to the gate. The electric field induces charge in the semiconductor and
allows us to control the current flow through the FET by varying the gat
voltage V_G. This is the origin of the terminology *field-effect*.

To describe the field-effect, we introduce the concept of a surface
charge density Q_S that has units of coulombs/square centimeter [C/cm^2]

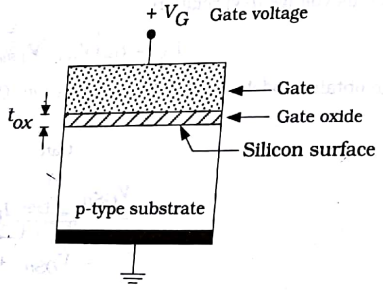


Figure 6.2 Structure of the MOS system

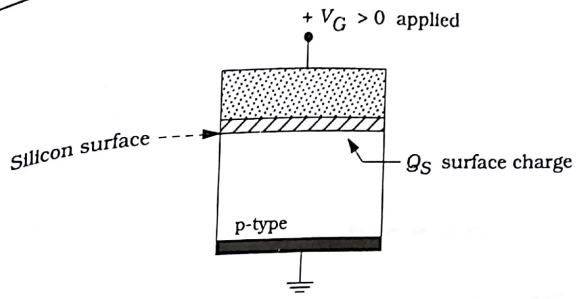


Figure 6.3 Surface charge density Q_S

This is related to the gate voltage by means of

$$Q_S = -C_{ox} V_G \qquad (6.3)$$

The concept of the surface charge can be understood using the drawing of Figure 6.3. Physically, Q_S represents the charge density that is seen looking down "into" the semiconductor from the oxide layer. The minus sign is included because a positive V_G induces a negative surface charge density. Although this is a simple-looking equation, MOS physics is complicated by the fact that Q_S represents all of the charge at the semiconductor surface, and the characteristics of the charge depend upon the value of the applied gate voltage.

At the circuit level, the threshold voltage is obtained by applying Kirchhoff's Voltage Law[1] (KVL) to the MOS system shown in Figure 6.4. Assuming that the gate voltage V_G has the polarity shown, KVL gives the expression

$$V_G = V_{ox} + \phi_S \qquad (6.4)$$

where V_{ox} is the voltage drop across the oxide layer and ϕ_S is the **surface potential** that represents the voltage at the top of the silicon. The voltages in the MOS system can be plotted as shown, with V_G at the gate and ϕ_S at the silicon surface. The oxide voltage V_{ox} is the difference (V_G - ϕ_S) and is the result of a decreasing electric potential inside the oxide as illustrated in the plot. Also note that the voltage in the semiconductor decreases from a value of ϕ_S to $\phi = 0$ in a more gradual manner.

The electric fields in the MOS system are illustrated in Figure 6.5 where we have expanded the vertical dimensions of the oxide to allow us

[1] KVL says that the sum of the voltage rises must equal the sum of the voltage drops when a circuit is traced around a closed loop.

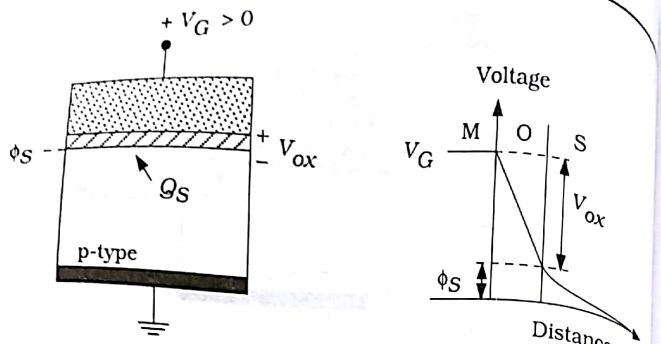


Figure 6.4 Voltages in the MOS system

to see more details. This shows the **oxide electric field** E_{ox} in the insul... tor pointing away from the higher potential gate electrode. The **surf... electric field** E_S also points in the same direction (toward the grou... connection), and is the field that controls the surface charge density Q_S ... the surface of the semiconductor. This is due to the fact that an elect... field exerts a force on a charged particle according to the Lorentz law

$$F = Q_{particle}E \tag{6.}$$

where $Q_{particle}$ is the charge on the particle with the appropriate sign. Po... itively charged holes have a charge of $+q$ and the force equation

$$F_h = +qE \tag{6.}$$

indicates that holes experience a force in the *same direction* as the electr... field.[2] Conversely, electrons have a negative charge $-q$ so they experien...

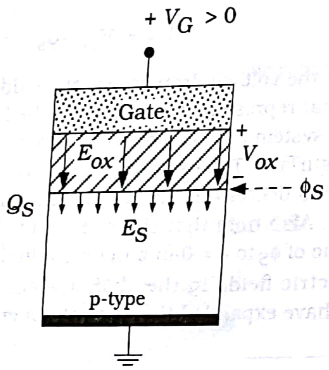


Figure 6.5 MOS electric fields

[2] Recall that the numerical value of the fundamental ... 1.6×10^{-19} C

a force of

$$F_e = -qE \tag{6.7}$$

In this case, the minus sign says that electrons are forced in a direction *opposite* to that of the electric field. With the surface field E_S pointing downward as shown in Figure 6.5, positive charges are forced *away* from the surface while negative charges are attracted *toward* the surface. This explains why the surface charge density consists of negative charge and Q_S itself is a negative number.

The nature of the surface charge depends upon the magnitude of the applied gate voltage. Suppose that V_G starts at 0 V and is then increased to a small positive value, say $V_G = 0.1$ V. The surface field attracts electrons toward the surface while pushing holes downward. This results in a negative charge on the semiconductor surface that is called the **bulk charge** density $Q_B < 0$ with units of C/cm^2. Bulk charge is due to the presence of boron atoms in the p-type substrate. Since a boron acts as an acceptor, it can capture and hold a negatively charged electron. When this happens, it becomes an **ionized dopant** with a net negative charge. Bulk charge is immobile since these ions cannot move. An analysis of the physics gives that

$$Q_B = -\sqrt{2q\varepsilon_{Si}N_a\phi_S} \tag{6.8}$$

where ε_{Si} is the silicon permittivity $\varepsilon_{Si} = 11.8\ \varepsilon_o$. For this case the oxide voltage is related to the bulk charge by

$$Q_B = -C_{ox}V_{ox} \tag{6.9}$$

Bulk charge is shown in Figure 6.6, where it is represented by circles with enclosed minus signs. The section from the silicon surface to the bottom of the bulk charge layer is called the **depletion region** because it is "depleted" of free electrons and holes: the holes have been forced away while the electrons have been "absorbed" by the boron dopant atoms. The depth x_d of the depletion layer increases with the applied voltage. This situation defines the "depletion mode of operation" in an MOS system. A depleted MOS structure cannot support the flow of electrical current since bulk charge is trapped by the silicon crystal lattice and cannot move.

If we increase the gate voltage to a special value called the **threshold voltage** V_{Tn}, then we observe a change in the charge properties. As implied by its name, the threshold voltage is the border between two different phenomena. For $V_G < V_{Tn}$, the charge is immobile bulk charge and

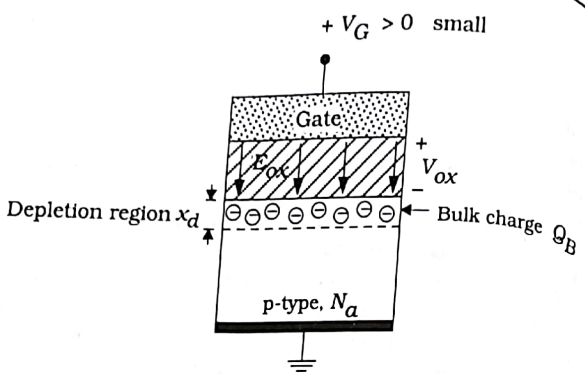


Figure 6.6 Bulk (depletion) charge in the MOS system

$Q_S = Q_B$. However, for $V_G > V_{Tn}$, the charge is made up of two distinct components such that

$$Q_S = Q_B + Q_e < 0 \qquad (6.10)$$

where Q_B is the bulk charge but now we observe an electron charge layer that is described by the quantity Q_e C/cm^2. The two components of the surface charge are shown in Figure 6.7. The important point is that electrons are mobile and can move in a lateral direction (parallel to the surface). The electron layer can thus be used as a channel region to construct a MOSFET. The threshold voltage $V_G = V_{Tn}$ represents the value of the gate voltage where Q_e just starts to form. This means that $Q_e = 0$ for $V_G = V_{Tn}$, but Q_e increases for $V_G > V_{Tn}$ according to the capacitor relation

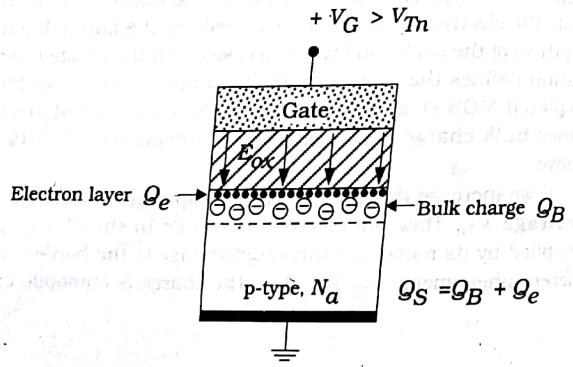


Figure 6.7 Formation of the electron charge layer

$$Q_e = -C_{ox}(V_G - V_{Tn}) \qquad (6.11)$$

We must subtract the threshold voltage from V_G to obtain the effective voltage across the insulator after the electron layer has formed. Note that this implies that the bulk charge Q_B does not increase for gate voltages that satisfy $V_G > V_{Tn}$. The negative sign is included to indicate that the electron charge is negative.

The numerical value of the threshold voltage is set in the fabrication process. Typically, it ranges from about $V_{Tn} = 0.5$ V to $V_{Tn} = 0.8$ V depending upon the intended type of application for the circuits. In VLSI system design, we assume the V_{Tn} has a value that is specified in the electrical parameters list.

6.1.1 Derivation of the Threshold Voltage[3]

It is not difficult to obtain an approximate expression that illustrates the origin of the numerical value. Recall that KVL gave us the voltage equation

$$V_G = V_{ox} + \phi_S \qquad (6.12)$$

A deeper study of the MOS system shows that the electron layer just starts to form when the surface potential reaches a value of

$$\phi_S = 2|\phi_F| \qquad (6.13)$$

where $|\phi_F|$ is called the **bulk Fermi potential** which is set by the acceptor doping density N_a of boron in the p-type semiconductor. The analysis gives

$$|\phi_F| = \left(\frac{kT}{q}\right) \ln\left(\frac{N_a}{n_i}\right) \qquad (6.14)$$

where k is Boltzmann's constant and T is the temperature in Kelvin. The parameter group (kT/q) is also known as the **thermal voltage** V_{th}, and has a numerical value of $(kT/q) \approx 0.026$ V at room temperature ($T = 27°$ C = 300 K).

With this established, we may write the KVL equation $V_G = V_{Tn}$ as

$$V_{Tn} = V_{ox}\big|_{\phi_S = 2|\phi_F|} + 2|\phi_F| \qquad (6.15)$$

Recalling equations (6.8) and (6.9) for Q_B then gives

$$V_{Tn} = \frac{1}{C_{ox}}\sqrt{2q\varepsilon_{Si}N_a(2|\phi_F|)} + 2|\phi_F| \qquad (6.16)$$

[3] This subsection may be skipped without loss of continuity in the discussion.

This is the threshold voltage for an **ideal MOS** structure in which the oxide is free of all stray charge and the gate and semiconductor materials are identical. A general expression that accounts for a more realistic situation is

$$V_{Tn} = \frac{1}{C_{ox}}\sqrt{2q\varepsilon_{Si}N_a(2|\phi_F|)} + 2|\phi_F| + V_{FB} \tag{6.17}$$

where V_{FB} is called the **flatband voltage** and accounts for both charge in the oxide and different gate and substrate materials.[4] In most modern CMOS processes, V_{FB} is a negative number that gives $V_{Tn} < 0$. Owing to the fact that most CMOS circuits operate with a positive power supply, it is desirable to have a positive threshold voltage with $V_{Tn} > 0$. This is accomplished by introducing another processing step where additional boron ions are implanted into the surface of the region. This alters the threshold voltage equation to read

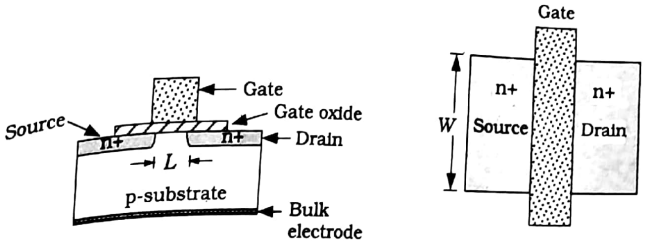
$$V_{Tn} = \frac{1}{C_{ox}}\sqrt{2q\varepsilon_{Si}N_a(2|\phi_F|)} + 2|\phi_F| + V_{FB} + \frac{qD_I}{C_{ox}} \tag{6.18}$$

where D_I is the implant dose that gives the number of ions implanted per square centimeter; D_I has units of cm^{-2}. The threshold voltage may thus be set by adjusting the implant dose. In some processes, it is also possible to alter the threshold voltage by changing the doping of the gate, which modifies the flatband voltage V_{FB}.

6.2 nFET Current-Voltage Equations

Let us now direct our interest toward the *I-V* characteristics for an n-channel MOSFET. These are determined by the physical structure of the device itself. The nFET consists of an MOS capacitor with n^+ regions added on both sides. The cross-sectional view in Figure 6.8(a) shows how the source and drain n^+ regions are placed with respect to the MOS (gate-oxide-substrate) capacitor. The distance between the edges of the n^+ regions is denoted by L, which is known as the (electrical) **channel length** of the device. L has units of length, and is the smallest feature size in the FET. The labels for the drain and the source are purely arbitrary at this point as the distinction between the two cannot be determined until the voltages are applied. For future reference we will note that in an nFET, the drain is the n^+ side with the higher voltage. A top view of the nFET is provided in Figure 6.8(b). This defines the (electrical) **channel width** W which also has units of length. The dimensionless quantity (W/L) is the

[4] The name *flatband voltage* arises from an energy band diagram analysis of the system which is beyond the scope of the present treatment.



(a) Side view (b) Top view

Figure 6.8 Details of the nFET structure

aspect ratio that is used to specify the relative size of a transistor with respect to others in the circuit.

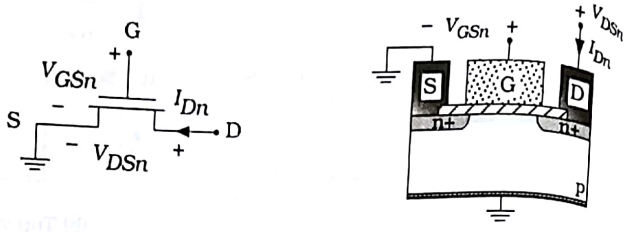
It is important to note that the values used for W and L in this chapter are the electrical or "effective" values, not the drawn values introduced in the previous chapter. This notational convention is used in device physics treatments, and is worthwhile to maintain when the discussion is at the electronics level. To avoid confusion, we will denote the drawn values (used for layout) as L' and W' such that

$$L = L' - \Delta L \tag{6.19}$$
$$W = W' - \Delta W$$

give the relationship between the electrical and drawn values; ΔL and ΔW are reduction factors from the processing. All of the equations in this chapter use electrical values W and L, and we maintain this association for the remainder of the book. The actual usage of the two in SPICE will be discussed later to provide final clarification.

The current flow characteristics are found by applying voltages to the physical structure and then analyzing the physics. As shown in Figure 6.9, there is a one-to-one correlation between the voltages represented in the symbol [see Figure 6.9(a)] and those applied to the integrated structure [see Figure 6.9(b)]. The source has been grounded for simplicity. This does not affect the generality of the results, since only relative voltages V_{GSn} and V_{DSn} are used. The program at this point is to determine the dependence of the current I_{Dn} on the voltages.

The key to understanding current flow in an n-channel MOSFET is to note that the MOS structure allows one to control the creation of the electron charge layer Q_e under the gate oxide by using the gate-source voltage V_{GSn}. This is illustrated in Figure 6.10. If $V_{GSn} < V_{Tn}$ then $Q_e = 0$ as illustrated in Figure 6.10(a). Since no electron layer exists, the two n+ regions are physically separated from each other, and no direct current flow path exists between them. From the outside world, an open circuit exists



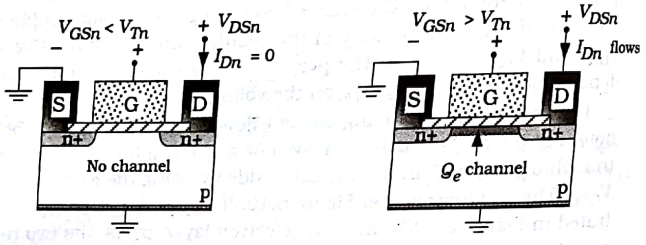
(a) Symbol (b) Structure

Figure 6.9 Currents and voltages for an nFET

between the drain and source terminals, which then says that the curren I_{Dn} must be 0. This state of operation is called **cutoff**, and is defined by having $V_{GSn} < V_{Tn}$ with $I_{Dn} = 0$. A cutoff transistor is equivalent to having an open switch between the drain and source terminals.

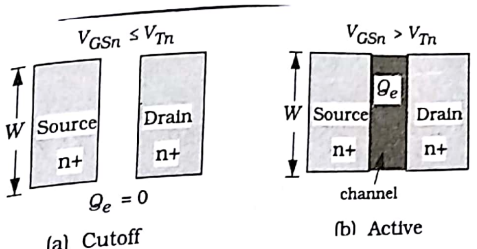
If, on the other hand, the gate-source voltage is increased to a value $V_{GSn} > V_{Tn}$, the situation changes dramatically. An electron charge layer Q_e is created underneath the gate oxide as shown in Figure 6.10(b). The layer provides an electrical **channel** between the electrons in the drain and source n+ regions and allows current to flow between the two. The presence of a channel defines the **active** mode of operation of the transistor. The numerical value of the current I_{Dn} depends upon both V_{GSn} and V_{DSn}.

Figure 6.11 shows the operational modes of the FET from the viewpoint of layers. These drawings illustrate the device operation as it appears at the silicon surface, i.e., if the gate layer is made transparent. Cutoff with $V_{GSn} < V_{Tn}$ is shown in Figure 6.11(a); since $Q_e = 0$, no channel exists between the drain and source regions and the device acts like an open switch. Figure 6.11(b) illustrates the opposite case where the gate-source voltage satisfies $V_{GSn} > V_{Tn}$ and results in the formation of an



(a) Cutoff (b) Active bias

Figure 6.10 Controlling the channel in an nFET



(a) Cutoff (b) Active

Figure 6.11 Channel formation in an nFET

electron charge layer Q_e. This defines the active mode of operation. The charge layer acts as a conducting channel between the two n+ regions, allowing charge transport between the two.

The behavior described in the preceding paragraphs almost justifies the modeling of an nFET as an assert-high switch that is OPEN with a small gate voltage ($V_{GSn} < V_{Tn}$) and CLOSED with a large gate voltage ($V_{GSn} > V_{Tn}$). In VLSI, the switch model is sufficient for designing logic gates as was demonstrated in Chapter 2. However, the electrical characteristics of FETs deviate substantially from those of an ideal switch. While this consideration does not affect the rules for logic formation with FETs, it does establish fundamental upper limits on the transient response of a CMOS network. Since switching speed is critical in modern chip design, it is worth the effort to dig deeper into the operation of MOSFETs to provide a complete picture of the VLSI design environment. The sizing of transistors provides the link between the physical design and the electronic operation of a logic gate.

To characterize an nFET at the device level, we will adopt the simple procedure where the current is plotted as a function of a voltage. Since there are two voltages (V_{GSn} and V_{DSn}), we will hold one constant while varying the other, and perform two separate experiments to obtain the overall behavior. The first is shown in Figure 6.12, where we have set the drain-source voltage to be the power supply value ($V_{DSn} = V_{DD}$) while we increase V_{GSn} in a positive direction from 0 V. This results in the plot of I_{Dn} vs. V_{GSn} shown. For voltages $V_{GSn} < V_{Tn}$, the transistor is in cutoff and $I_{Dn} = 0$. Increasing the gate-source voltage to values $V_{GSn} > V_{Tn}$ biases the nFET into the active region of operation by forming the electron charge layer Q_e. The drain-source voltage V_{DSn} provides the difference in potential needed to move the charge, which results in the current I_{Dn} flowing through the device. Mathematically, the current can be approximated by the equation

$$I_{Dn} = \frac{\beta_n}{2}(V_{GSn} - V_{Tn})^2 \qquad (6.20)$$

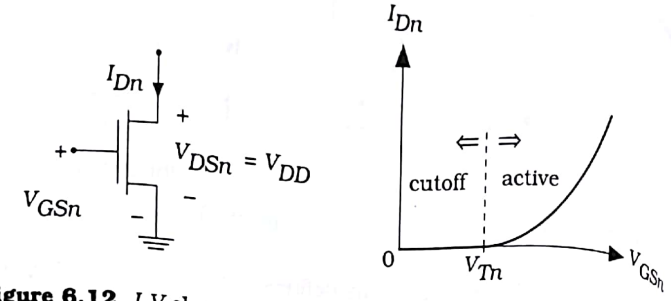


Figure 6.12 *I-V* characteristics as a function of V_{GSn}

which shows a quadratic dependence on the voltage. This defines the **square-law model** of a FET. Although it is only an approximation, it is very useful for calculating the behavior of complex CMOS networks. The factor β_n multiplying the voltage factor is the device transconductance parameter with units of A/V^2. Every nFET has a distinct value of β_n that is determined by its aspect ratio through

$$\beta_n = k'_n\left(\frac{W}{L}\right) \tag{6.2}$$

In this equation, k'_n is the process transconductance parameter that is calculated from

$$k'_n = \mu_n C_{ox} \tag{6.2}$$

It cannot be changed by the VLSI designer. In this equation, μ_n is the electron mobility at the silicon surface. In a silicon MOSFET at room temperature, μ_n is typically around 500 to 580 $cm^2/V\text{-sec}$ and is a characteristic of the material.

Note that the process transconductance is proportional to the oxide capacitance per unit area

$$C_{ox} = \frac{\varepsilon_{ox}}{t_{ox}} \tag{6.23}$$

Substituting gives

$$k'_n = \frac{\mu_n \varepsilon_{ox}}{t_{ox}} \tag{6.24}$$

so that a thin oxide (small t_{ox}) gives a large value for k'_n. This increases the sensitivity of the device with respect to the gate voltage, and helps the device switch faster. From the physical viewpoint it can be seen that decreasing t_{ox} increases C_{ox}, which in turn enhances the field-effect.

Example 6.1

Consider an nFET that has a gate oxide thickness of $t_{ox} = 12$ nm and an electron mobility of $\mu_n = 540\ cm^2/V\text{-sec}$. The oxide capacitance per cm^2 is

$$C_{ox} = \frac{(3.9)(8.854\times10^{-14})}{1.2\times10^{-6}} = 2.88\times10^{-7}\ \ F/cm^2 \tag{6.25}$$

where we have used $t_{ox} = 12\ nm = 1.2\times10^{-6}$ cm since the permittivity is in units of F/cm. The process transconductance is computed from

$$\begin{aligned} k'_n &= \mu_n C_{ox} \\ &= (540)(2.88\times10^{-7}) \\ &= 1.55\times10^{-4}\ \ A/V^2 \end{aligned} \tag{6.26}$$

or,

$$k'_n = 155\ \mu A/V^2 \tag{6.27}$$

If the oxide is reduced to a thickness of $t_{ox} = 8$ nm, then the process transconductance increases to a value of

$$k'_n = 233\ \mu A/V^2 \tag{6.28}$$

indicating a more sensitive device.

Let us now change the voltages to the situation shown in Figure 6.13. In this case, we apply a constant gate-source voltage $V_{GSn} > V_{Tn}$ to the nFET and vary the drain-source voltage V_{DSn}. This gives the plot of I_{Dn} vs. V_{DSn} shown. For small values of V_{DSn}, the current can be estimated by the equation

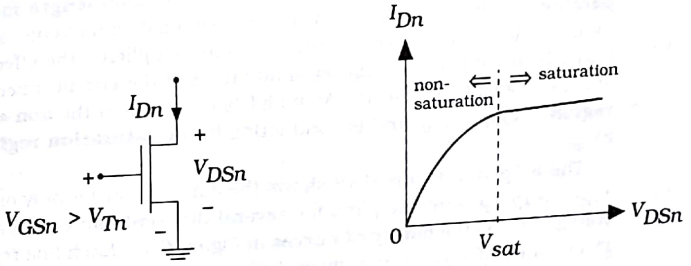


Figure 6.13 *I-V* characteristics as a function of V_{DSn}

$$I_{Dn} = \frac{\beta_n}{2}[2(V_{GSn} - V_{Tn})V_{DSn} - V_{DSn}^2]$$

which describes a parabola. The peak occurs at the point where

$$\frac{\partial I_{Dn}}{\partial V_{DSn}} = 0$$

Evaluating the derivative and equating the result to 0 gives

$$\frac{\partial}{\partial V_{DSn}}[2(V_{GSn} - V_{Tn})V_{DSn} - V_{DSn}^2] = 2(V_{GSn} - V_{Tn}) - 2V_{DSn} = 0$$

The solution to this equation defines a special value of V_{DSn} called the **saturation voltage**

$$V_{sat} = V_{DSn}|_{\text{peak current}} = V_{GSn} - V_{Tn}$$

that is shown in the plot. For larger drain-source voltages that satisfy $V_{DSn} \geq V_{sat}$, the current is approximately independent of V_{DSn} and is given by

$$I_{Dn} = \frac{\beta_n}{2}(V_{GSn} - V_{Tn})^2$$

This is identical to that given in equation (6.20) and is called the **saturation current** since it is the largest value of I_{Dn} that can flow for a given value of V_{GSn}. A more detailed analysis shows that the saturation current does increase slightly for $V_{DSn} \geq V_{sat}$. This is often modeled by the equation

$$I_{Dn} = \frac{\beta_n}{2}(V_{GSn} - V_{Tn})^2[1 + \lambda(V_{DSn} - V_{sat})]$$

where λ is an empirical quantity called the **channel-length modulation parameter** with units of V^{-1}. When performing digital circuit calculations by hand, we usually assume that $\lambda = 0$ for simplicity; the effect of λ can easily be included in computer simulations of the circuit if necessary. In general, we will say that the MOSFET is operating in the **non-saturation region** if $V_{DSn} \leq V_{sat}$, and is conducting in the **saturation region** if $V_{DSn} \geq V_{sat}$.

The *I-V* plot in Figure 6.13 shows the current flow for only one value of V_{GSn}. Superposing the plots for several different values of gate-source voltages yields the **family of curves** in Figure 6.14. Each line represents a given value of V_{GSn}. For a given drain-source voltage V_{DSn}, the current increases with V_{GSn} as indicated. The separation between the non-saturated and saturated operational regions is given by the saturation current

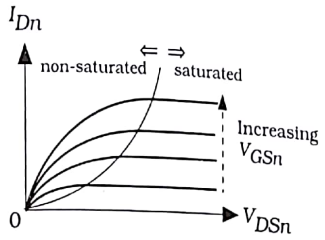


Figure 6.14 nFET family of curves

$$I_{Dn} = \frac{\beta_n}{2}V_{sat}^2 \tag{6.35}$$

where $V_{sat} = (V_{GSn} - V_{Tn})$ depends upon the value of the gate-source voltage. This set of equations allows us to find the drain current I_{Dn} once we know the voltages.

Example 6.2

Consider an n-channel MOSFET with the following characteristics:

t_{ox}=10 nm, μ_n = 520 cm^2/V-s, (W/L) = 8, V_{Tn} = + 0.70 V

This information allows us to find the device equations. We will start by finding the oxide capacitance using

$$C_{ox} = \frac{\varepsilon_{ox}}{t_{ox}} = \frac{(3.9)(8.854\times10^{-14})}{10\times10^{-7}} = 3.45\times10^{-7}\text{F/cm}^2 \tag{6.36}$$

The process transconductance is found from

$$k'_n = \mu_n C_{ox} = (520)(3.45\times10^{-7}) = 1.79\times10^{-4}\text{ A/V}^2 \tag{6.37}$$

or, k'_n = 179 μA/V^2. The device transconductance may now be calculated from

$$\beta'_n = k'_n\left(\frac{W}{L}\right) = 179(8) = 1.435\text{ mA/V}^2 \tag{6.38}$$

Let us now calculate the drain current for different voltage combinations.

Suppose that we apply voltages of V_{GSn} = 2 V and V_{DSn} = 2 V to the nFET. The first task is to determine the state of conduction, i.e., is the transistor operating in the saturated or non-saturated region? Once this is known, we can use the appropriate equation. The saturation voltage is

$$V_{sat} = V_{GSn} - V_{Tn}$$
$$= 2 - 0.7$$
$$= 1.3 \text{ V}$$

Since $V_{DS} = 2$ V > V_{sat}, the nFET is saturated such that

$$I_{Dn} = \frac{\beta_n}{2}(V_{GSn} - V_{Tn})^2$$
$$= \left(\frac{1.435}{2}\right)(2 - 0.7)^2$$
$$= 1.213 \text{ mA}$$

Now let us lower the drain-source voltage to $V_{DSn} = 1.2$ V while maintaining $V_{GSn} = 2$ V. The saturation voltage is still given by

$$V_{sat} = V_{GSn} - V_{Tn} = 1.3 \text{ V}$$

but now $V_{DSn} = 1.2$ V < V_{sat}, which says that the transistor is non-saturated. The current is then computed from

$$I_{Dn} = \frac{\beta_n}{2}[2(V_{GSn} - V_{Tn})V_{DSn} - V_{DSn}^2]$$
$$= \left(\frac{1.435}{2}\right)[2(1.3)(1.2) - (1.2)^2]$$
$$= 1.21 \text{ mA}$$

This set of calculations illustrates the general current characteristics of a MOSFET.

6.2.1 SPICE Level 1 Equations

Channel-length modulation effects are easily included in SPICE simulations, but tend to be somewhat cumbersome for hand calculations that use the equation set above. An alternate set of MOSFET equations that follows SPICE LEVEL 1 models is to write the non-saturation current which is valid for $V_{DSn} \le V_{sat}$ in the form

$$I_{Dn} = \frac{\beta_n}{2}[2(V_{GSn} - V_{Tn})V_{DSn} - V_{DSn}^2](1 + \lambda V_{DSn}) \quad (6.43)$$

This provides a continuous transition to the saturation current

$$I_{Dn} = \frac{\beta_n}{2}(V_{GSn} - V_{Tn})^2(1 + \lambda V_{DSn}) \quad (6.44)$$

that is valid for $V_{DSn} \ge V_{sat}$. This is not consistent with a physical analysis since channel-length modulation occurs only in a saturated device. However, it makes circuit analysis easier. These forms are quite common in analog CMOS design. However, channel-length modulation effects do not affect hand calculations of digital circuits enough to justify the increased algebraic complexity, so they are rarely used in hand calculations here.

6.2.2 Body-Bias Effects

Up to this point we have ignored the presence of the p-type substrate. In reality, the MOSFET is a four-terminal device with the substrate being the **bulk** (B) terminal of the device. **Body-bias effects** occur when a voltage V_{SBn} exists between the source and bulk terminals of a nFET as in Figure 6.15. The body-bias V_{SBn} voltage increases the threshold voltage of the device such that

$$V_{Tn} = V_{T0n} + \gamma(\sqrt{2|\phi_F| + V_{SBn}} - \sqrt{2|\phi_F|}) \quad (6.45)$$

where γ is the body-bias coefficient with units of $V^{1/2}$ and $2|\phi_F|$ is the bulk Fermi potential term from equation (6.14). The term V_{T0n} is the zero body-bias threshold voltage

$$V_{T0n} = V_{Tn}|_{V_{SBn}=0} \quad (6.46)$$

and is the value quoted in a set of processing specifications. The body-bias coefficient can be estimated by

$$\gamma = \frac{\sqrt{2q\varepsilon_{Si}N_a}}{C_{ox}} \quad (6.47)$$

where $q = 1.6 \times 10^{-19}$ C is the fundamental charge unit, $\varepsilon_{Si} = 11.8\varepsilon_o$ is the permittivity of silicon, and N_a is the acceptor doping in the p-type substrate. The value of γ is usually quoted in the process specification. Note that thin oxides decrease the value of γ.

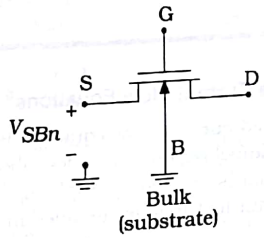


Figure 6.15 Bulk electrode and body-bias voltage

Example 6.3

Consider an nFET where $V_{T0n} = 0.7$ V, $\gamma = 0.08$ $V^{1/2}$, and $2|\phi_F| = 0.$
The threshold voltage depends on the body-bias voltage V_{SBn} accordi

$$V_{Tn} = 0.70 + 0.08(\sqrt{0.58 + V_{SBn}} - \sqrt{0.58})$$

Some values can be computed as follows:

V_{SBn} (V)	V_{Tn} (V)
0	0.70
1	0.74
2	0.77
3	0.79

The function is plotted in Figure 6.16, which illustrates the characteri
square root dependence.

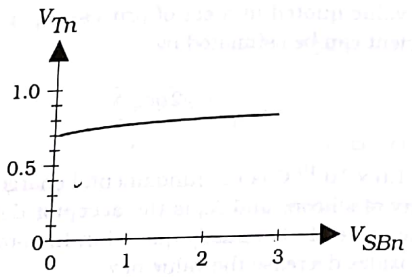


Figure 6.16 Body-bias effect

6.2.3 Derivation of the Current Flow Equations[5]

The non-saturated current flow equation is obtained by analyzing the physics of the channel region that is described of the electron charge den sity Q_e C/cm^2 that is created by applying a gate-source voltage V_{GSn} V_{Tn}. The important features are detailed in Figure 6.17. Physically, the

[5] This section may be skipped without loss of continuity. The reader may jump to Section 6.3 where the main discussion is resumed.

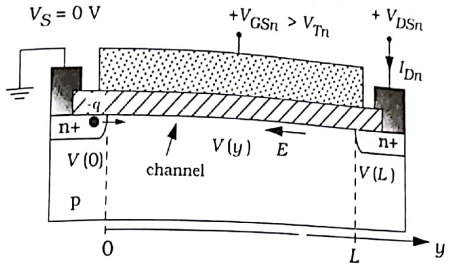


Figure 6.17 Channel voltage in a MOSFET

drain-source voltage V_{DSn} applied across the device induces an electric field E that points from the drain to the source (remember that, by definition, the drain is the side at the higher voltage). Since electrons have a negative charge $-q$, they experience a force in a direction opposite to that of the electric field. The electrons thus move *from* the source and flow through the channel *to* the drain; this is the origin of the electrode names. In electronics, we usually deal with **conventional current** which moves in the direction of positive charge; current flows in a direction that is opposite to the direction of electron motion. Applying this to the nFET shows that the current flows from the drain to the source as shown.

Now that the qualitative aspects of the physics have been discussed, let us analyze the situation in greater depth. From electromagnetic theory we know that electric fields are conservative. This means that there exists an electrostatic potential (or voltage) $V(y)$ such that

$$E(y) = -\frac{dV}{dy} \tag{6.49}$$

where y is a coordinate that is defined as shown in the drawing. $V(y)$ is called the **channel voltage** and is due to the applied drain-source voltage V_{DSn}. At the ends of the channel, it has the known values of

$$V(0) = 0 \qquad V(L) = V_{DSn} \tag{6.50}$$

which act as boundary conditions on the problem and indicate that $V(y)$ decreases from the drain to the source. The existence of the channel voltage alters the charge in the channel and makes Q_e a function of the coordinate y. To understand this, recall the electron charge density in a simple MOS structure (not a FET) is given by

$$Q_e = -C_{ox}(V_{GSn} - V_{Tn}) \qquad \text{(MOS value)} \tag{6.51}$$

where $(V_{GSn} - V_{Tn})$ is the effective voltage across the insulating oxide l... For the nFET, however, the situation changes because of the chann... age $V(y)$ underneath the oxide. A moment's reflection will verify that ... opposes the applied gate-source voltage V_{GSn} since it is a positive nu... The nFET channel charge equation is thus given by

$$Q_e(y) = -C_{ox}[V_{GSn} - V_{Tn} - V(y)] \qquad \text{(MOSFET)}$$

which shows that Q_e varies in the channel. The minimum value is on... drain side where

$$Q_e(L) = -C_{ox}[V_{GSn} - V_{Tn} - V_{DSn}]$$

while the maximum charge density is found at the source with

$$Q_e(0) = -C_{ox}[V_{GSn} - V_{Tn}]$$

The functional dependence $Q_e(y)$ is significant because it means that ... charge density is nonuniform. This in turn implies that the I-V relati... ship will be non-linear.

The equation for I_{Dn} can be obtained by applying the above obser... tions to the channel geometry illustrated in Figure 6.18. To handle ... varying charge density, let us start with the differential channel segm... that has a length dy as shown. The current I_{Dn} flows through this s... ment and causes a voltage drop

$$dV = I_{Dn}dR$$

where dR is the differential resistance

$$dR = \frac{dy}{\sigma_n A_n}$$

of the segment. In this equation, σ_n is the conductivity and A_n is t... cross-sectional area. Since the conductivity of an n-type region is given ...

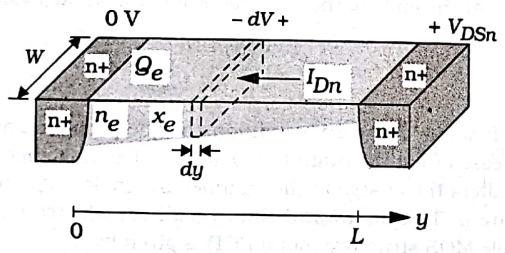


Figure 6.18 Channel geometry

$\sigma_n = q\mu n$, we may rewrite the denominator in the form

$$\sigma_n A_n = q\mu_n n_e W x_e \qquad (6.57)$$

where n_e is the electron density in units of cm^{-3} and x_e is the thickness of the channel at that point. The channel charge density is equivalent to

$$Q_e = -qn_e x_e \qquad (6.58)$$

This can be seen by noting the units of Q_e are C/cm^2 and that the given quantities combine on physical grounds; the negative sign is due to the fact that Q_e is defined as a negative number. Substituting this into resistance equation then yields

$$dV = -\frac{I_{Dn}dy}{\mu_n W Q_e} = \frac{I_{Dn}dy}{\mu_n W C_{ox}(V_{GSn} - V_{Tn} - V)} \qquad (6.59)$$

using the expression for Q_e from equation (6.52). This can be rearranged and integrated to read

$$I_{Dn}\int_0^L dy = \mu_n W C_{ox}\int_0^{V_{DSn}} [(V_{GSn} - V_{Tn}) - V]dV \qquad (6.60)$$

The limits of integration have been chosen as $y = 0$ to $y = L$ to include the entire channel. The voltage integral on the right-hand side uses the equivalent channel voltages at these points, i.e., $V(0) = 0$ V and $V(L) = V_{DSn}$ Assuming that the term $(V_{GSn}-V_{Tn})$ on the right side is independent of the channel voltage V gives

$$I_{Dn}L = \mu_n W C_{ox}[(V_{GSn} - V_{Tn})V_{DSn} - V_{DSn}^2] \qquad (6.61)$$

so that

$$I_{Dn} = \mu_n C_{ox}\left(\frac{W}{L}\right)[(V_{GSn} - V_{Tn})V_{DSn} - V_{DSn}^2] \qquad (6.62)$$

This is the same as the non-saturated current expression given earlier in equation (6.29).

One interesting point concerning the channel arises when we extend the analysis to the saturation voltage $V_{sat} = (V_{GSn} - V_{Tn})$. Equation (6.53) gives the channel charge at the drain side. Substituting the saturation voltage $V_{DSn} = V_{sat}$ gives

$$Q_e(L) = -C_{ox}[V_{GSn} - V_{Tn} - V_{sat}] = 0 \qquad (6.63)$$

i.e., the charge density appears to fall to 0 when at the saturation voltage. A more detailed analysis shows that the charge does not really fall to zero, but is in fact small. This corresponds to a phenomenon known

as **channel pinch-off** in the FET. Formally, it is the border between saturation and non-saturation regions of operation. For $V_{DSn} > V_{sat}$, the pinch-off of the charge limits the current flow (hence the term **saturation**) and the pinch-off effect itself decreases the effective length of the channel (hence the **channel-length modulation** factor λ).

6.3 The FET RC Model

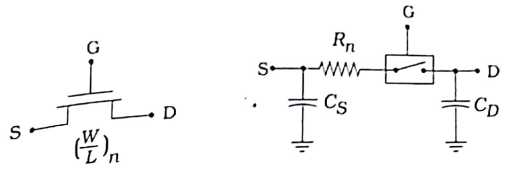
The equations of current flow above illustrate that the nFET exhibits **non-linear** I-V characteristics. This property makes it difficult to analyze electrical circuits that use FETs because the circuit equations themselves become non-linear; hand calculations thus become quite tedious. The solution, of course, is to use a CAD tool such as SPICE to perform the difficult analyses. But this does not solve the problem that VLSI designers face: they must **create** circuits that have the proper electrical characteristics. This pinpoints the difference between **analysis** and **design**: analysis deals with studying a new network that has resulted from the design process. Designers are true problem solvers in that they use existing knowledge as a basis for building new systems.

There are two approaches to dealing with the problem of messy transistor equations. The first is to let circuit specialists deal with the issues introduced by the non-linear devices. Skilled electronic designers are indispensable in the chip design process. VLSI system design, on the other hand, is based on logic and digital architectures; engineers working at the systems level also need to understand FET circuitry. This provides the basis of the second approach: create a simplified **linear model** of the device that is useful at the logic and system level. By its very nature, the model will ignore most of the details of the current flow. It will, however, be much simpler to use for tracing signal flows in complex networks at the system level. If we can work at least some of the important transistor characteristics into the model, then it can be used to provide a basis for the first design phase. Simplified linear models also allow us to develop techniques that compare various algorithms for choosing the most efficient VLSI approach.

The linear model that will be used in our treatment is shown in Figure 6.19. This simplifies the nFET to a resistor R_n, two capacitors (C_S and C_D), and an assert-high logic-controlled switch. The values of the linear components depend on the aspect ratio $(W/L)_n$ of the nFET in a manner that will be developed in the next two subsections.

6.3.1 Drain-Source FET Resistance

Field-effect transistors are inherently non-linear, so we must be careful about the concept of using a linear resistor with a fixed value of R_n to model the current flow through an nFET.



(a) nFET Symbol (b) Linear model for nFET

Figure 6.19 RC model of an nFET

Consider the situation shown in Figure 6.20. In Figure 6.20(a), the gate-source voltage is assumed to be set with a value $V_{GSn} > V_{Tn}$, to make the nFET active. The current I_{Dn} is then a function of the drain-source voltage V_{DSn} as plotted in Figure 6.20(b). The drain-source resistance at any point on the curve is then given by

$$R_n = \frac{V_{DSn}}{I_{Dn}} \tag{6.64}$$

The non-linear effects are due to the fact that I_{Dn} varies with V_{DSn}, which makes R_n itself a function of V_{DSn}.

The effects of this dependence can be seen by writing the resistance equations for the three points labeled 'a', 'b', and 'c' shown in the drawing. For small values of V_{DSn} (point 'a'), the current is approximated by

$$I_{Dn} \approx \beta_n (V_{GSn} - V_{Tn}) V_{DSn} \tag{6.65}$$

by ignoring the squared term V_{DSn}^2 in the non-saturated current flow equation (6.29). The resistance is then

$$R_n \approx \frac{1}{\beta_n (V_{GSn} - V_{Tn})} \tag{6.66}$$

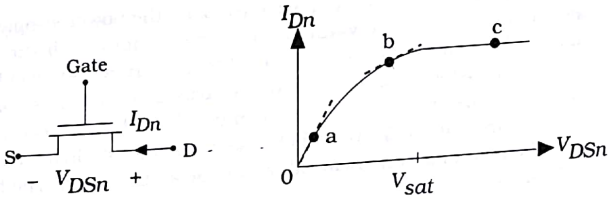


Figure 6.20 Determining the nFET resistance.

so that R_n varies with V_{GSn}. At point 'b', the full non-saturated current equations must be used so that

$$R_n = \frac{2}{\beta_n[2(V_{GSn} - V_{Tn}) - V_{DSn}]} \quad (6.6$$

showing that R_n is a function of both V_{GSn} and V_{DSn}. When the device is saturated as at point 'c', the resistance becomes

$$R_n = \frac{2V_{DSn}}{\beta_n(V_{GSn} - V_{Tn})^2} \quad (6.6$$

by using equation (6.20) which ignores channel-length modulation. Once again, the resistance varies with both V_{GSn} and V_{DSn}.

These equations illustrate that it is not possible to define a constant value for R_n and still maintain the correct current-flow behavior. Note, however, that in all cases, R_n is inversely proportional to β_n, i.e.,

$$R_n \propto \frac{1}{\beta_n} \quad (6.6$$

This is simply a statement that a device with a large β_n conducts more current than one with a small β_n. Using the definition

$$\beta_n = k'_n\left(\frac{W}{L}\right)_n \quad (6.70)$$

shows that the important parameter is the device aspect ratio $(W/L)_n$. Qualitatively, increasing the width W of the nFET decreases the resistance.

With this in mind, we will introduce a simple equation for modeling the resistance as a function of the aspect ratio (or, width) of the transistor by writing

$$R_n = \frac{\eta}{\beta_n(V_{DD} - V_{Tn})} \quad (6.71)$$

In constructing this equation, we have used the power supply voltage V_{DD} as the largest possible value for V_{GSn} by analogy with the expressions above. The factor η has been included to account for some of the variation as the transistor is switched through various operating regions; it has no physical basis. In the literature, the multiplying factor tends to range from $\eta = 1$ to around $\eta = 6$. We will choose $\eta = 1$ for simplicity, acknowledging that the resulting numerical values will be a little small. The formula then reduces to

$$R_n = \frac{1}{\beta_n(V_{DD} - V_{Tn})} \ \Omega \quad (6.72)$$

which is the final form. The unit of the resistance R_n is ohms, which is consistent with the units established by the denominator.

Example 6.4

Consider an nFET that has a channel width $W = 8$ μm, a channel length of $L = 0.5$ μm, and is made in a process where $k'_n = 180$ μA/V^2, $V_{Tn} = 0.70$ V, and $V_{DD} = 3.3$ V. The linearized drain-source resistance is computed as

$$R_n = \frac{1}{\beta_n(V_{DD} - V_{Tn})} \quad (6.73)$$

so that substituting the values gives

$$R_n = \frac{1}{(180\times10^{-6})\left(\frac{8}{0.5}\right)(3.3 - 0.7)} = 133.5\ \Omega \quad (6.74)$$

If we shrink the channel width to $W = 5$ μm while keeping all other quantities the same, the resistance increases to

$$R_n = 133.5\left(\frac{8}{5}\right) = 213.6\ \Omega \quad (6.75)$$

where we have simply scaled the value by noting that R_n is inversely proportional to the channel width. It is important to remember that these values are not actual values for the nFET resistance, but are used only for simplified modeling.

6.3.2 FET Capacitances

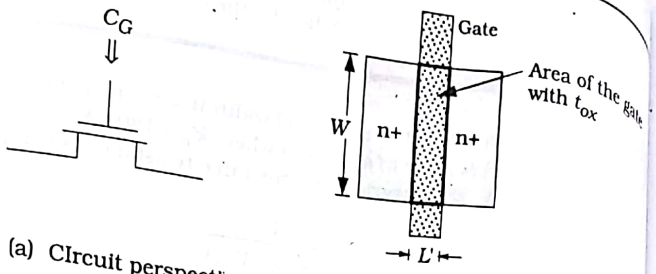
A MOSFET has several parasitic capacitances that must be included in the simplified switching model. As we will see in later developments, the maximum switching speed of a CMOS circuit is determined by the capacitances.

MOS Capacitances

The metal-oxide-semiconductor layering scheme is intrinsically a capacitor, so let us analyze its value first. Figure 6.21(a) shows the circuit model. If we look into the gate terminal of the FET, we see the **gate capacitance** C_G that is due to the MOS structure. Since this is the region that has a gate oxide thickness of t_{ox}, it is described by the oxide capacitance per unit area C_{ox}. Denoting the area of the gate region by A_G gives us

$$C_G = C_{ox}A_G \quad (6.76)$$

in farads, which is taken to be the capacitance between the gate terminal and ground. For the simple geometry shown in Figure 6.21(b) the gate



(a) CIrcuit perspective (b) Physical origin

Figure 6.21 Gate capacitance in a FET

area is $A_G = WL'$ where W is the channel width and L' is the drawn chan-
nel length. L' is just the channel length that is defined by the extent of the
gate region when viewed from the top of the layout drawing. Thus,

$$C_G = C_{ox}WL' \tag{6.77}$$

gives the important result that the gate capacitance is proportional to the
width W of the channel.

We also describe the MOS contributions using the gate-source capaci-
tance C_{GS} and the gate-drain capacitance C_{GD} shown in Figure 6.22.
These two parasitics are complicated because their values change with
the voltages due to the changing shape of the channel region. When we
have $C = C(V)$, the capacitance is said to be **non-linear**. In VLSI system
design, we will usually employ a circuit simulation program such as
SPICE to handle the detailed calculations. For our purposes, we will sim-
ply estimate the values by writing

$$C_{GS} \approx \frac{1}{2}C_G \approx C_{GD} \tag{6.78}$$

In other words, we will just divide the gate capacitance by 2 and split it
equally between C_{GS} and C_{GD}. Although this isn't extremely accurate, it

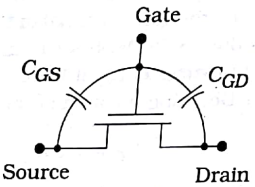


Figure 6.22 Gate-source and gate-drain capacitance

allows us to focus on the large-scale characteristics. Proper use of a CAD
tool suite will provide final verification.

Example 6.5

Consider a FET with an oxide capacitance of $C_{ox} = 3.45 \times 10^{-7}$ F/cm^2 and a gate with dimensions $W = 8$ μm and $L' = 0.5$ μm. The gate capacitance formula gives

$$C_G = (3.45\times10^{-7})(8\times10^{-4})(0.5\times10^{-4}) \tag{6.79}$$

While this is a simple calculation, let us reduce it even further by noting that $C_{ox} = 3.45 \times 10^{-7} = 3.45$ fF/μm^2 where we recall that 1 fF = 10^{-15} F. Then

$$C_G = 3.45(8)(0.5) = 13.8 \text{ fF} \tag{6.80}$$

The gate-source and gate-drain contributions are then estimated by

$$C_{GS} \approx \frac{1}{2}C_G = 6.9 \text{ fF} = C_{GS} \tag{6.81}$$

These are typical orders of magnitude for FET capacitances. It is important to keep in mind that we are always dealing with device capacitances that are on the order of a few fF.

Junction Capacitance

Semiconductor physics reveals that a pn junction automatically exhibits capacitance due to the opposite polarity charges involved. This is called **junction** or **depletion** capacitance and is found at every drain or source region of a FET. Figure 6.23 illustrates the presence of the pn junctions and the associated capacitances C_{SB} (source-bulk) and C_{DB} (drain-bulk). We usually characterize this capacitance by introducing a parameter C_J with units of F/cm^2 such that the total capacitance is

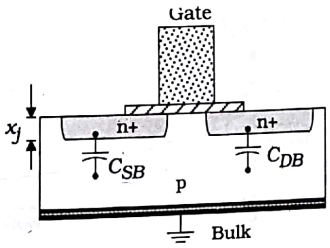


Figure 6.23 Junction capacitances in a MOSFET

$$C_0 = C_j A_{pn} \text{ F}$$

where A_{pn} is the area of the junction in units of cm^2. The value of C_j determined by the processing, and varies with doping levels.

There are two complications in applying this formula to the nFET. The first is that this capacitance also varies with the voltage. With a reverse bias voltage of V_R applied, this is usually modeled by an equation of the form

$$C = \frac{C_0}{\left(1 + \frac{V_R}{\phi_o}\right)^{m_j}}$$

where C_0 is the **zero-bias capacitance** (with $V_R = 0$), ϕ_o is the **built-in potential** of the junction, and m_j is called the **grading coefficient** of the junction. Both ϕ_o and m_j are determined by the doping characteristics. A special case is that of an **abrupt** or **step** junction where the doping changes from a constant acceptor density N_a to a constant donor density N_d. In this case, $m_j = 1/2$ and the built-in voltage is computed from

$$\phi_o = \left(\frac{kT}{q}\right) \ln\left[\frac{N_d N_a}{n_i^2}\right]$$

Another simple model is the **linearly graded junction** where the doping transition is a linear function of position. This gives a grading coefficient of $m_j = 1/3$, while the built-in potential ϕ_o can be calculated if the details of the doping are known. For our purposes, we will always assume that C_j, ϕ_o, and m_j are known parameters. In general, the maximum value of the capacitance is $C = C_0$ when $V_R = 0$; increasing the reverse voltage across the junction causes C to decrease as illustrated in Figure 6.24. We will use the zero-bias values as estimates in hand calculations, and turn to the use of CAD tools when more accurate values are needed.

The second complication that we need to consider in calculating the pn junction capacitance is the geometry of the pn junctions. The cross-sectional view shown in Figure 6.23 shows that the n+ regions are "embedded" a depth x_j (called the **junction depth**) within the p-substrate. When computing the area A_{pn} of the pn junction, we must be careful to include both the bottom and the side contributions. Figure 6.25 illustrates the geometry. The top view of the FET in Figure 6.25(a) defines the channel width W of the transistor, and the extent X (away from the gate) of the n+ region. The 3-dimensional aspects of the pn junction area calculation are illustrated in Figure 6.25(b). Since the n+ region may be visualized as an "open box" structure, it is possible to decompose the boundaries into the bottom and sidewall sections shown. The area of the bottom region is easily seen to be

$$A_{bot} = XW \tag{6.85}$$

which is equal to the area of the n+ region seen in the top view. Then, denoting the zero-bias junction capacitance per unit area of this region by C_j with unit of F/cm^2, the capacitance due to the bottom section is

$$C_{bot} = C_j XW \tag{6.86}$$

To compute the sidewall capacitance C_{sw}, we note that the total sidewall area is obtained by adding the four contributions. Each sidewall section has a height equal to the junction depth x_j. Sidewall sections 1 and 2 have areas of $(W \times x_j)$, while Sidewall sections 3 and 4 have areas of $(X \times x_j)$. Adding the terms gives

$$\begin{aligned} A_{sw} &= 2(W \times x_j) + 2(X \times x_j) \\ &= x_j P_{sw} \end{aligned} \tag{6.87}$$

where P_{sw} is the **sidewall perimeter** in units of cm such that

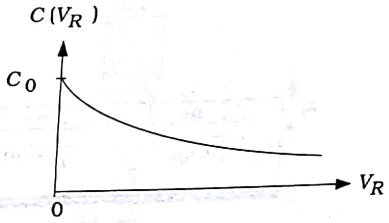
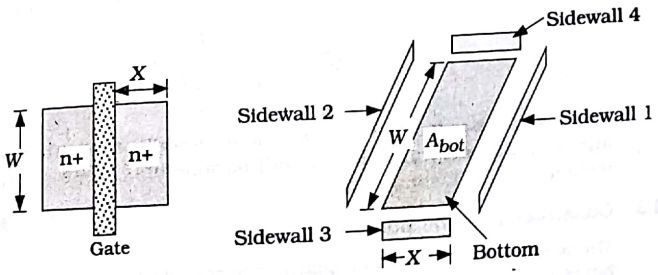


Figure 6.24 Junction capacitance variation with reverse voltage



(a) Top view (b) Geometry

Figure 6.25 Calculation of the FET junction capacitance

$$P_{sw} = 2(W + X)$$

for the rectangular geometry shown in the example. The sidewall capacitance is obtained by multiplying by the junction capacitance per unit area. This is usually modified to the form

$$C_{sw} = C_{jsw} P_{sw} \text{ farads}$$

where

$$C_{jsw} = C_J x_j \text{ F/cm}$$

is the **sidewall capacitance per unit perimeter**. This is convenient to use because the perimeter P_{sw} can be found directly from the layout drawing. In practice, C_{jsw} is specified as a processing parameter which automatically refers to the bottom capacitance.

These formulas ignore the gate overlap L_o of the n+ regions underneath the gate. For hand calculations, these should be included by changing

$$X \rightarrow (X + L_o)$$

everywhere. In a SPICE simulation, the drawn values of L and W are used to describe the circuit and gate overlap (and other) correction factors are included through the modeling information.

The total zero-bias capacitance of the n+ region is given by adding the bottom and sidewall contributions:

$$\begin{aligned} C_n &= C_{bot} + C_{sw} \\ &= C_J A_{bot} + C_{jsw} P_{sw} \end{aligned} \tag{6.92}$$

which can be used to compute both C_{SB} and C_{DB}. It is worthwhile to note that the non-linear characteristics of the bottom and sidewall junctions are usually distinct. This gives a non-linear variation of the form

$$C_n = \frac{C_J A_{bot}}{\left(1 + \frac{V}{\phi_o}\right)^{m_j}} + \frac{C_{jsw} P_{sw}}{\left(1 + \frac{V}{\phi_{osw}}\right)^{m_{jsw}}} \tag{6.93}$$

where V is the reverse voltage, m_j and ϕ_o describe the bottom junction, and m_{jsw} and ϕ_{osw} are the sidewall parameters. These are routinely included in SPICE simulations.

6.3.3 Construction of the Model

The parasitic resistance and capacitance contributions may now be combined to construct the simple RC model of the nFET. A layout drawing is useful to aid our visualization of the model. Figure 6.26 shows the top view of an nFET with the capacitance contributions. The p-substrate surrounding the transistor is at ground potential. A signal entering from

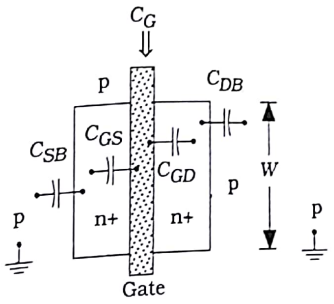


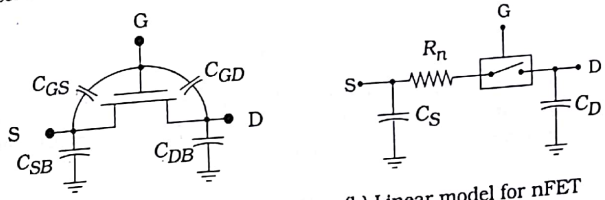
Figure 6.26 Physical visualization of FET capacitances

either side sees both an MOS term (C_{GS} or C_{GD}) and a junction parasitic (C_{SB} or C_{DB}).

The physical layout forms the basis for the schematic-level circuit in Figure 6.27(a), where the capacitors are divided into source and drain components. The simplest approach is to write

$$\begin{aligned} C_S &= C_{GS} + C_{SB} \\ C_D &= C_{GD} + C_{DB} \end{aligned} \tag{6.94}$$

which approximates the total capacitance by summing all contributions that touch a given node. Moreover, we will use zero-bias values for simplicity in all hand calculations. It is important to note that the resistance R_n is inversely proportional to the aspect ratio $(W/L)_n$, while the capacitances increase with the channel width W.



(a) nFET (b) Linear model for nFET

Figure 6.27 Final construction of the nFET RC model

Example 6.6

Let us create a switch model for the nFET shown in Figure 6.28; the measurements are given in units of microns (μm). First, since we are given

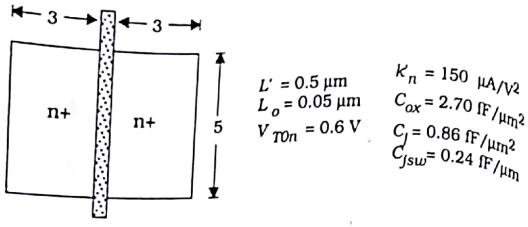


Figure 6.28 FET geometry for modeling example

that the overlap distance is $L_o = 0.05$ μm, the electrical channel length = 0.5-2(0.05) = 0.4 μm. The channel width is shown as $W = 5$ μm. Assuming a power supply voltage of $V_{DD} = 3.3$ V, the linear resistance is given

$$R_n = \frac{1}{\left(\frac{5}{0.4}\right)(150\times10^{-6})(3.3-0.6)} = 197.5 \ \Omega \qquad (6.$$

If the sheet resistance of the n+ regions were known, the parasitic resistance could be found and added to this value.

The gate capacitance is

$$C_G = (2.7)(5)(0.5) = 6.75 \ \text{fF} \qquad (6.$$

so that

$$C_{GS} = C_{GD} = 3.375 \ \text{fF} \qquad (6.$$

by taking one-half of the gate value. The junction capacitance for either side is

$$C_n = (0.86)A_{bot} + (0.24)P_{sw} \qquad (6.9$$

With the overlap L_o =0.05 μm both the area and the perimeter are larger than the drawn values of (3 × 5) μm² and 16 μm, respectively. Including this observation in the formula gives

$$C_n = (0.86)(5)(3.05) + (0.24)(2)(5+3.05) = 16.98 \ \text{fF} \qquad (6.9$$

The final drain and source capacitances are then

$$C_D = C_S = 16.98 + 3.375 = 20.36 \ \text{fF} \qquad (6.100$$

which completes the calculations.

This simple model provides a reasonable basis for design estimates. To use it in a circuit problem, we just substitute the model for the transistor and then apply standard linear circuit techniques. Since it ignores the inherent non-linearities of the FET, the analysis will have limited accuracy. Increased precision is obtained from computer simulations that are performed after the initial design has produced a candidate circuit. Simplified device modeling is an important part of the VLSI design process as it allows us to create basic networks very quickly. However, these must always be checked and "fine-tuned" using CAD tools.

6.4 pFET Characteristics

A p-channel MOSFET is the electrical complement of an nFET. This was seen in Chapter 2 where the nFET was modeled as an assert-high switch while the pFET behaved as an assert-low switch. The complementary characteristics are even more evident at the device level. Suppose that we start with an nFET and want to modify it to form a pFET. All that needs to be done to the structure is

- Change all n-type regions to p-type regions
- Change all p-type regions to n-type regions

and the resulting device will in fact be a pFET. This shown in Figure 6.29. We have chosen a p-type substrate for both devices, which then makes it necessary to include the n-well region to embed the pFET in. Both devices are assumed to have the same oxide thickness of t_{ox} so that

$$C_{ox} = \frac{\varepsilon_{ox}}{t_{ox}} \qquad (6.101)$$

describes both nFETs and pFETs. This means that the basic mechanism of the field effect is identical to that discussed for the nFET. However, since the polarities of the regions have been reversed, both the direction of the electric fields and the polarities of the charges will be opposite.

The structural details of a pFET are provided in Figure 6.30. The channel length L is defined as the distance between the edges of the source

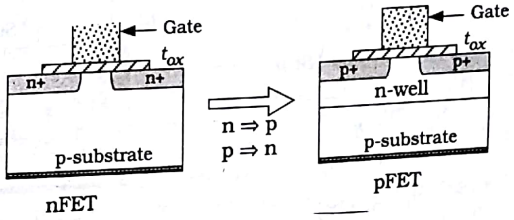
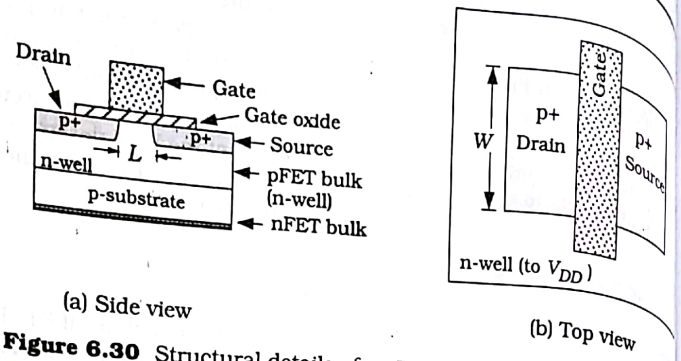


Figure 6.29 Transforming an nFET to a pFET

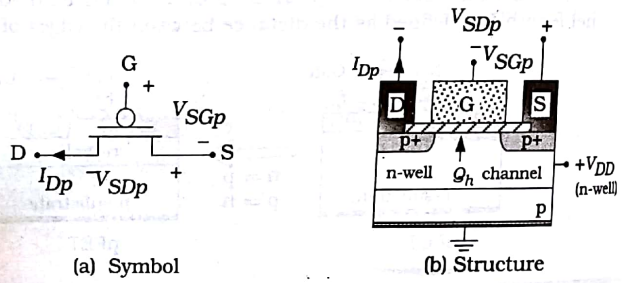


(a) Side view (b) Top view

Figure 6.30 Structural details of a pFET

and drain p+ regions as shown in the side view of Figure 6.30(a), while
channel width W is defined by the extent of the p+ regions as in the
view of Figure 6.30(b); these feature sizes are the same as those used
define the nFET. The presence of the n-well is shown in both drawing
and is an important region of a pFET since it acts as the bulk electrode
the device. Electrically, the n-well is tied to the positive power supply vo
age V_{DD} which acts to insure that the voltage is well defined. As with
nFET, the naming of the source and drain terminals requires that
know the relative voltage levels. The pFET, however, uses definitions th
are exactly opposite to those used for an nFET. This means that the p
side at the higher voltage is the source, while the remaining side (at
lower voltage) is the drain.

A p-channel MOSFET uses positively charged holes for current flo
The pFET current I_{Dp} and the device voltages are defined in Figure 6.3
where it has been assumed that the right side of the device is the sour
in both drawings. First note that the schematic symbol in Figure 6.31(



(a) Symbol (b) Structure

Figure 6.31 Current and voltages in a pFET

shows the current I_{Dp} as flowing out of the drain electrode. This is because positive charge moves from the source to the drain, which gives current in that direction. The pFET voltages are referenced to the source and are denoted by V_{SGp} (the source-gate voltage) and V_{SDp} (the source-drain voltage); note that these are opposite in polarity from the analogous nFET quantities V_{GSn} and V_{DSn}. The structural view in Figure 6.31(b) includes the fact that the n-well layer is electrically connected to the power supply voltage V_{DD}.

The conduction through the pFET is governed by the source-gate voltage V_{SGp}. The MOS structure consisting of the gate, the oxide, and the n-well layers is characterized by a pFET threshold voltage V_{Tp}. By convention, V_{Tp} is a negative number with typical values of V_{Tp} = -0.5 V to about V_{Tp} = -1.0 V. From the physical viewpoint, the value of V_{SGp} determines whether the gate is sufficiently negative with respect to the source to create a layer of holes under the gate oxide and thus establish a positive hole charge density of Q_h C/cm^2 that can be used as a channel between the source and drain. This is summarized by the statements

$$Q_h = 0 \text{ for } (V_{SGp} < |V_{Tp}|)$$
$$Q_h \text{ exists for } (V_{SGp} > |V_{Tp}|) \qquad (6.102)$$

where we have used the absolute value $|V_{Tp}|$ of the threshold voltage. The first line corresponds to the situation where the gate voltage is not sufficiently negative to induce the formation of a hole conduction layer in the n-well, while the second case is where V_{SGp} is large enough to insure that the gate voltage can attract the holes and form the channel. The role of the source-drain voltage V_{SDp} is to move the charge from the source to the drain if the channel exists.

The pFET threshold voltage can be computed from

$$V_{Tp} = -\frac{1}{C_{ox}}\sqrt{2q\varepsilon_{Si}N_d(2\phi_{Fp})} - 2\phi_{Fp} + V_{FBp} \mp \frac{qD_I}{C_{ox}} \qquad (6.103)$$

where N_d is the donor doping in the n-well,

$$2\phi_{Fp} = 2\left(\frac{kT}{q}\right)\ln\left(\frac{N_d}{n_i}\right) \qquad (6.104)$$

is the surface potential needed to create the hole layer in the pFET, V_{FBp} is flatband voltage for the pFET MOS structure, and the last term represents the threshold adjustment ion implant step that has an ion dose of D_I. The minus sign '-' is used if donors are implanted, while the plus sign '+' corresponds to the case where acceptors are used.

The conduction modes for a pFET are summarized in Figure 6.32. The cutoff condition is portrayed in Figure 6.32(a). In this situation, V_{SGp} is less than $|V_{Tp}|$ so that Q_h = 0 and no channel exists. This gives I_{Dp} = 0

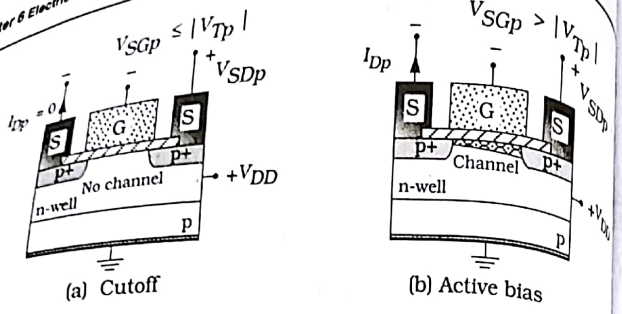


Figure 6.32 Conduction modes of a pFET

which can be modeled as an open switch. Active operation is shown in Figure 6.32(b) and is defined by $V_{SGp} \ge |V_{Tp}|$. The hole conduction layer forms and gives rise to the channel as shown. Since the electric field points from the right side to the left side, the positively charged holes originate from the source (right) and flow to the drain (left). The pFET current I_{Dp} thus flows out of the drain electrode as shown.

The current-voltage characteristics of a pFET can described using the same approach as that introduced for nFETs. In Figure 6.33, the source-drain voltage V_{SDp} is specified to be V_{DD} (the power supply value) while the source-gate voltage V_{SGp} is increased. For $V_{SGp} \le |V_{Tp}|$, the device is in cutoff with $I_{Dp} = 0$ since no channel exists. When V_{SGp} is elevated above $|V_{Tp}|$, the charge layer Q_h is formed and the device is active. The current can be approximated by the square-law expression

$$I_{Dp} = \frac{\beta_p}{2}(V_{SGp} - |V_{Tp}|)^2 \qquad (6.105)$$

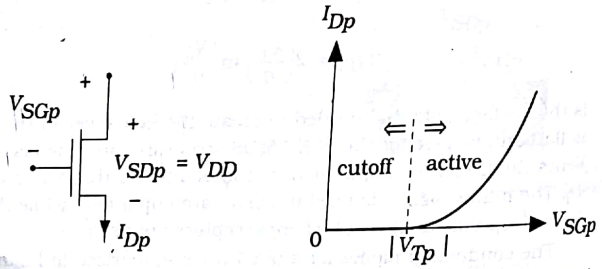


Figure 6.33 Gate-controlled pFET current-voltage characteristics

where

$$\beta_p = k'_p\left(\frac{W}{L}\right)_p \qquad (6.106)$$

is the pFET device transconductance with units of A/V^2. The quantity $(W/L)_p$ is the pFET aspect ratio, and k'_p is the pFET process transconductance parameter

$$k'_p = \mu_p C_{ox} \qquad (6.107)$$

with units of A/V^2. In this equation, μ_p is the hole mobility. These definitions are identical to the nFET parameters except that μ_p must be used to describe the motion of holes in silicon. A typical value for the surface hole mobility in silicon at room temperature is $\mu_p = 220$ cm^2/V-sec; this is noticeably lower than the electron value (around 550 cm^2/V-sec) quoted earlier. A typical ratio is

$$r = \frac{\mu_n}{\mu_p} \approx 2 - 3 \qquad (6.108)$$

Note that the important multiplying factors in the FET currents are transconductance parameters

$$\beta_n = k'_n\left(\frac{W}{L}\right)_n \qquad (6.109)$$

$$\beta_p = k'_p\left(\frac{W}{L}\right)_p$$

The difference between k'_p and k'_n can lead to some unique design choices for $(W/L)_n$ and $(W/L)_p$ when nFETs and pFETs are used in the same circuit.

Figure 6.34 shows the more general case where V_{SGp} is held constant while V_{SDp} is increased. Each value of V_{SGp} gives a distinct plot of I_{Dp} vs. V_{SDp} which results in the family of curves shown. The saturation voltage

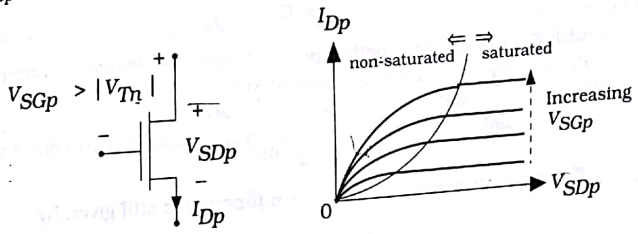


Figure 6.34 pFET *I-V* family of curves

for a pFET is defined by

$$V_{sat} = V_{SGp} - |V_{Tp}| \tag{6.}$$

such that non-saturated conduction occurs for $V_{SDp} \leq V_{sat}$ and described by

$$I_{Dp} = \frac{\beta_p}{2}[2(V_{SGp} - |V_{Tp}|)V_{SDp} - V_{SDp}^2] \tag{6.}$$

while saturation occurs for $V_{SDp} \geq V_{sat}$ with

$$I_{Dp} = \frac{\beta_p}{2}(V_{SGp} - |V_{Tp}|)^2 \tag{6.}$$

Saturated conduction was portrayed previously in Figure 6.33; a FET can be recognized as being saturated if the voltage between the source and drain is large (compared to V_{sat}).

6.4.1 pFET Parasitics

The parasitic resistance and capacitances of the pFET are calculated in the same manner as for the nFET. A linearized pFET resistance can be introduced as

$$R_p = \frac{1}{\beta_p(V_{DD} - |V_{Tp}|)} \tag{6.}$$

which illustrates the dependence

$$R_p \propto \frac{1}{\beta_p} = \frac{1}{k_p'(W/L)_p} \tag{6.}$$

Large aspect ratios thus give small resistances that allow for larger current flows.

The capacitances are computed using the same equations as for nFETs. For example, the input gate capacitance is given by

$$C_{Gp} = C_{ox}(WL)_p \tag{6.}$$

with C_{ox} the same for both types of transistors. The gate-source and gate-drain capacitances are approximated by

$$C_{GS} \approx \frac{1}{2}C_{Gp} \approx C_{GD} \tag{6.}$$

The junction capacitance of a p+-n junction is still given by

$$C_p = C_j A_{bot} + C_{jsw} P \tag{6.}$$

but it is important to remember that the numerical values of C_j and C_{jsw}

are different for nFETs and pFETs because of differences in doping. Linear RC modeling of a pFET is identical to that shown in Figure 6.27 for the nFET, except that pFET values and an assert-low switch are used.

6.5 Modeling of Small MOSFETs

The equations presented in this chapter are simplified models that are useful for initial design estimates. They are reasonably accurate in **long-channel** MOSFETs where L is larger than about 20–30 μm; these are still found in discrete (separate individual) devices. Modern IC technology has reduced the channel length of production-line VLSI transistors to $L = 0.13$ μm, and this value is still shrinking. The physics of submicron sized devices is quite complicated. It is not possible to find closed form expressions that accurately describe these transistors. At the circuit design level, we turn instead to two levels of modeling: scaling theory and computer models.

6.5.1 Scaling Theory

Scaling theory deals with the "incredible shrinking transistor" and directs us toward the behavior of a device when its dimensions are reduced in a structured manner.

Consider a transistor that has a channel width W and a channel length L. We wish to find out how the main electrical characteristics change when both dimensions are reduced by a scaling factor $s > 1$ such that the new (scaled) transistor has sizes

$$\tilde{W} = \frac{W}{s} \qquad \tilde{L} = \frac{L}{s} \tag{6.118}$$

We note that the original transistor has a gate area of $A = WL$ while the scaled FET occupies

$$\tilde{A} = \frac{A}{s^2} \tag{6.119}$$

For example, $s = 2$ implies that the scaled device occupies only 25% of the area of the original. This provides ample motivation for continuing to improve the lithographic process.

Now let us consider the device transconductance. Since both W and L are scaled by the same factor, the aspect ratio is invariant:

$$\left(\frac{W}{L}\right) = \left(\frac{\tilde{W}}{\tilde{L}}\right) \tag{6.120}$$

The oxide capacitance is given by

$$C_{ox} = \frac{\varepsilon_{ox}}{t_{ox}} \tag{6.121}$$

where t_{ox} is the thickness of the gate oxide. If the new FET has a thinner oxide that is decreased as

$$\tilde{t}_{ox} = \frac{t_{ox}}{s} \tag{6.122}$$

then the scaled device has

$$\tilde{C}_{ox} = \frac{\varepsilon_{ox}}{\left(\frac{t_{ox}}{s}\right)} = sC_{ox} \tag{6.123}$$

i.e., it is increased by a factor of s. Since the process transconductance is given by $k' = \mu C_{ox}$, the device transconductance $\beta = k'(W/L)$ is increased in the scaled device to

$$\tilde{\beta} = sk'\left(\frac{W}{L}\right) = s\beta \tag{6.124}$$

Note, however, that the ability to scale L and W by s does not imply that the oxide thickness can be reduced by the same factor, so one must be careful when applying this relation. If it does hold, then the FET resistance

$$R = \frac{1}{\beta(V_{DD} - V_T)} \tag{6.125}$$

has a scaled value

$$\tilde{R} = \frac{1}{s\beta(V_{DD} - V_T)} \tag{6.126}$$

If we do not alter the voltages applied to the reduced-size FET, then the resistance is decreased according to

$$\tilde{R} = \frac{R}{s} \tag{6.127}$$

On the other hand, if we can scale the voltages in the small device to new values of

$$\tilde{V}_{DD} = \frac{V_{DD}}{s}, \qquad \tilde{V}_T = \frac{V_T}{s} \tag{6.128}$$

then the resistance of the scaled FET would be unchanged with

$$\tilde{R} = R \tag{6.129}$$

This provides the basis of **voltage scaling** where we reduce the voltages as the device dimensions decrease.

To see the effects of scaling the voltage, consider a scaled MOSFET with reduced voltages of

$$\tilde{V}_{GS} = \frac{V_{DS}}{s}, \qquad \tilde{V}_{GS} = \frac{V_{GS}}{s} \tag{6.130}$$

such that the non-saturated current of the original device is given by

$$I_D = \frac{\beta}{2}[2(V_{GS} - V_T)V_{DS} - V_{DS}^2] \tag{6.131}$$

Applying the scaling formulas gives the current in the scaled FET as

$$\tilde{I}_D = \frac{s\beta}{2}\left[2\left(\frac{V_{GS}}{s} - \frac{V_T}{s}\right)\frac{V_{DS}}{s} - \frac{V_{DS}^2}{s^2}\right] = \frac{I_D}{s} \tag{6.132}$$

The power dissipation of the transistor is

$$\tilde{P} = \tilde{V}_{DS}\tilde{I}_D = \frac{V_{DS}I_D}{s^2} \tag{6.133}$$

i.e., it is reduced by a factor of $1/s^2$. This is a motivating factor for reducing the power supply voltage as the size of FETs decreases.

The actual value of the power supply voltage V_{DD} is a system-level decision that is often used to reduce the power dissipation of the circuit. The value of the threshold voltage V_T is controlled in the processing. Although some changes in the operating voltages can be made, the reduction is usually different from the geometric scaling specified by s. This does, however, provide a general set of guidelines on what to expect.

6.5.2 Small-Device Effects

As the size of MOSFETs decreased in the 1980's and 1990's, the natural approach was to provide corrections to the current flow equations that would account for newly observed effects. Many new types of phenomena were discovered and studied, and much of the jargon and terminology remains in use today.

The most important geometrical parameter in a VLSI FET is the channel length L. Since the aspect ratio (W/L) determines the maximum current flow through the transistor, reducing L allows us to simultaneously reduce W while still maintaining the same aspect ratio. In the next chapter we will demonstrate that the aspect ratio is the primary circuit design parameter. The scaled circuit thus consumes less area, but still maintains some of the important circuit characteristics.

When the channel length is reduced below about 20 μm, it is found

that the threshold voltage decreases from its long-channel value $V_{T,long}$. This is called a **geometrical short-channel effect** (SCE) and is expressed by an equation of the form

$$V_T = V_{T,long} - (\Delta V_T)_{SCE} \tag{6.134}$$

where $(\Delta V_T)_{SCE}$ increases with decreasing L. The reduction can be calculated by accounting for the charge more accurately than is done in long-channel derivation. The **narrow-width effect** (NWE) is also a geometrical correction that increases the threshold voltage as W decreases. This can be expressed in the form

$$V_T = V_{T,long} + (\Delta V_T)_{NWE} \tag{6.135}$$

and is due to fringing electric fields that are ignored in the long-channel analysis. Minimum-size devices may exhibit both SCEs and NWEs.

Reducing L also causes a change in the channel conduction characteristics. Consider a FET with a drain-source voltage V_{DS} applied. The channel electric field may be estimated by

$$E \approx \frac{V_{DS}}{L} \tag{6.136}$$

which shows that E increases as L decreases. The velocity of a charged particle in silicon is observed to follow the dependence illustrated in Figure 6.35. For small values of E, the velocity increases linearly as expressed by

$$v = \mu E \tag{6.137}$$

which defines the mobility μ used in the derivation of the FET current. However, as the electric field intensity is increased, $v(E)$ enters the nonlinear region and the mobility is no longer a constant. The value eventually hits the **saturation velocity** v_s, which is about 10^7 cm/sec for electrons in silicon at room temperature. The simplified equations are therefore no longer valid, and must be modified. Modern short-channel FETs routinely enter the velocity saturation region.

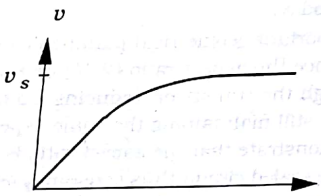


Figure 6.35 Velocity-field relation for charged particles in silicon

The velocity is also useful for estimating the transit time τ_t, which is the time required for a charge to traverse the channel. This is viewed as the fundamental limit on how fast a FET can switch. To obtain a simple expression, we use the velocity v to write

$$\tau_t = \frac{v}{L} \tag{6.138}$$

In the linear region of the v-E plot,

$$\tau_t = \frac{\mu E}{L} = \frac{\mu V_{DS}}{L^2} \tag{6.139}$$

This gives another reason for decreasing the channel length: the value of τ_t is decreased, indicating faster switching. Once the particle velocity saturates, the transit time goes to the constant

$$\tau_t = \frac{v_s}{L} \tag{6.140}$$

so the effect is muted somewhat. Another effect of short channels is that fewer electrons are involved in the current flow. Many assumptions involving the statistically derived charge concentrations start to become invalid.

Many other effects have been observed in small-geometry MOSFETs, and research continues as the device dimensions continue to shrink and new transistors are proposed and developed. The interested reader is directed toward the current literature for more details.

6.5.3 SPICE Modeling

Over the years we have learned that it is not possible to derive closed-form equations that accurately describe modern transistors. Luckily, the development of sophisticated CAD tools allows us to perform accurate simulations at the device and circuit level. Device simulators are beyond the scope of this book. Circuit simulation, on the other hand, is a routine procedure in the design of VLSI circuits. The design flow is to first create the logic circuit using FET switching theory, then estimate electrical characteristics using simplified equations. The circuit is then simulated, and the results are used to refine the electrical design. The most widely used circuit simulation engine is SPICE.[6] This program was conceived and written at the University of California, Berkeley, to aid in the design of integrated circuits. Since it is considered a standard in the industry, we will center our discussion around it. Several implementations of SPICE are available, but they are all similar in operation.

[6] This is an acronym for Simulation Program with Integrated Circuit Emphasis.

MOSFETs in SPICE are entered into a circuit listing using an element statement of the form

Mname ND NG NS NB model_name L=length W=width <AS, PS, AD, PD>

where

- Mname is the name of the FET, such as M1 or Mn_out
- ND, NG, NS, NB are the node numbers (or names, if allowed) of the drain, gate, source, and bulk, respectively.
- model_name is the name of the .model line that provide the process parameter listing.
- AS, PS, AD, PD are the (optional) area and perimeter of the drain (AD, PD) and source (AS, PS) for the device. Areas are in units of m^2, and perimeters must be specified in units of m.

The important electrical parameters are included in the .model line which has the form

.model < listing >

where < listing > is a list of numerical values. There are many different MOSFET models in use. They are distinguished using the

Level = N

statement in < listing > where the value of N defines the equation set. The original SPICE allowed Level = 1, 2, 3 where Level 1 is based on a modified form of the equation set derived in Section 6.2.3. The Level 2 model, also called the bulk-charge equations, is more accurate, while Level 3 is an empirical model. Levels 1 and 2 lose accuracy when applied to modern submicron devices, but are often used for initial estimates because they allow very quick simulations.

We usually enter drawn values for all dimensions in the element statement. For example, the transistor in Example 6.6 would be described by

MExa6_6 10 20 30 0 nFET L=0.4U W=5U AD=15P PD=16U AS=15P PS=16U

where P is the pico scaler, and U is the micro scaler. The difference between the drawn values and the effective (electrical) values is computed from information supplied in the

.MODEL nFET <parameters>

listing supplied for the process. This makes the translation from layout to simulation file much easier.

In modern CMOS, the BSIM model set provides the most accurate SPICE simulations.[7] Unfortunately, the parameter set itself is somewhat cryptic and the values do not always have a direct relationship to the simple analytic expressions. A detailed treatment of the BSIM model can be found in Reference [2]. In VLSI design, we generally interpret the model as a given set of parameters that can be used in a CAD tool suite. Extracting the netlist from a layout allows an electrical simulation to be run.

[7] BSIM stands for Berkeley Submicron IGFET Model, where IGFET stands for Insulated Gate FET. In everyday usage, IGFET and MOSFET are used interchangeably.

6.6 References for Further Reading

[1] R. Jacob Baker, Harry Li, and David E. Boyce, **CMOS Circuit Design, Layout and, Simulation**, IEEE Press, Piscataway, NJ, 1998.

[2] Yuhua Cheng and Chenming Hu, **MOSFET Modeling and BSIM3 User's Guide**, Kluwer Academic Press, Norwell, MA, 1999.

[3] Richard S. Muller and Theodore I. Kamins, **Device Electronics for Integrated Circuits**, 2nd ed., John Wiley & Sons, New York, 1992.

[4] Robert F. Pierret, **Semiconductor Device Fundamentals**, Addison-Wesley, Reading, MA, 1996.

[5] Ben G. Streetman and Sanjay Banerjee, **Solid State Electronic Devices**, 5th ed., Prentice Hall, Upper Saddle River, NJ, 1999.

[6] Jasprit Singh, **Semiconductor Devices**, John Wiley & Sons, New York, 2001.

[7] S. M. Sze, **Semiconductor Devices**, 2nd ed., Wiley-Interscience, New York, 1981.

[8] John P. Uyemura, **CMOS Logic Circuit Design**, Kluwer Academic Publishers, Norwell, MA, 1999.

[9] Edward S. Yang, **Microelectronic Devices**, McGraw-Hill, New York, 1988.

6.7 Problems

[6.1] A CMOS process produces gate oxides with a thickness of t_{ox} = 100 Å. The FET carrier mobility values are given as μ_n = 550 cm^2/V-sec and μ_p = 210 cm^2/V-sec.

(a) Calculate the oxide capacitance per unit area in units of fF/μm^2.

(b) Find the process transconductance values for nFETs and pFETs. Place your answer in units of $\mu A/V^2$.

[6.2] An nFET with W = 10 μm and L = 0.35 μm is built in a process where k'_n = 110 $\mu A/V^2$ and V_{Tn} = 0.70 V. Assume V_{SBn} = 0 V.

(a) Find the current if the voltages are set to V_{GSn} = 2 V, V_{DSn} = 1.0 V.

(b) Find the current if the voltages are set to V_{GSn} = 2 V, V_{DSn} = 2 V.

[6.3] An nFET has a device transconductance of β_n = 2.3 mA/V^2 and a threshold voltage of 0.76 V. Assume V_{SBn} = 0 V.

(a) Find the current if the voltages are set to V_{GSn} = 1 V, V_{DSn} = 2.5 V.

(b) Find the current if the voltages are set to V_{GSn} = 2 V, V_{DSn} = 2.5 V.

(c) Find the current if the voltages are set to V_{GSn} = 3 V, V_{DSn} = 2.5 V.

[6.4] Consider a pFET that has a gate oxide thickness of t_{ox} = 60 Å. The hole mobility is measured to be 220 cm^2/V-sec, and the aspect ratio is $(W/L) = (12/1)$. Assume that V_{DD} = 3.3 V and $|V_{Tp}|$ = 0.7 V.

(a) Calculate the process transconductance k'_p in units of mA/V².

(b) Find the device transconductance β_p and the resistance R_p.

[6.5] An nFET has a gate oxide with a thickness of t_{ox} = 120 Å. The p-type bulk region is doped with boron at a density of $N_a = 8 \times 10^{14}$ cm^{-3}. Find V_{tn} given that V_{T0n} = 0.55 V and (W/L) = 10.

(a) Calculate the body bias coefficient γ.

(b) What is the device threshold voltage if a body bias voltage of V_{SBn} = 2 V is applied?

(c) The electron mobility is μ_n = 540 cm²/V-sec. Calculate the drain current with bias voltages of V_{GSn} = 3 V, V_{DSn} = 3 V, and V_{SBn} = 3 V applied to the device.

[6.6] Construct the RC switch model for the FET layout in Figure P6.1. Assume a power supply voltage of 3 V and that the dimensions are in units of microns.

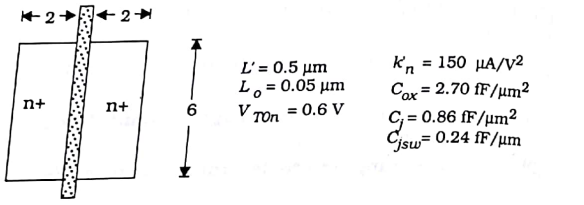


Figure P6.1 Transistor layout geometry for Problem 6.6

[6.7] Write a SPICE description of the nFET in Figure P6.1. Use your listing to obtain the family of I_D versus V_{DS} curves.

[6.8] Consider the FET geometry shown in Figure P6.1 where the sheet resistance of the n+ regions is 30 Ω, and the poly gate has a sheet resistance of 26 Ω. Compute the parasitic resistances R_{n+} and R_{poly} associated with these parameters by determining the appropriate geometry that applies for each. How would these parasitics affect the device operation?

[6.9] An nFET with W = 20 μm and L = 0.5 μm is built in a process where k'_n = 120 μA/V² and V_{Tn} = 0.65 V. The voltages are set to a value of V_{GSn} = V_{DSn} = V_{DD} = 5 V.

(a) Is the transistor saturated or non-saturated?

(b) Calculate the drain-source resistance using the proper equation for the transistor.

(c) Compare your value in (b) with that found using equation (6.71) with a value of η = 1.

[6.10] An nFET with L = 0.5 μm is built in a process where k'_n = 100 μA/V² and V_{Tn} = 0.70 V. The gate-source voltage is set to a value of V_{GSn} = V_{DD} = 3.3 V. Calculate the required channel width to obtain a resistance of R_n = 950 Ω using equation (6.71) with for a value of η = 1.

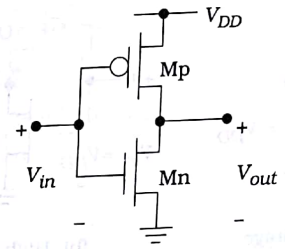
Electronic Analysis of CMOS Logic Gates

7

In the previous chapter we examined the electrical characteristics of MOSFETs. This sets the foundation for analyzing the behavior of transistors in CMOS logic circuits in this chapter. The treatment centers on the important areas of switching speed and layout design, and provides the foundation for much of modern chip design.

7.1 DC Characteristics of the CMOS Inverter

The CMOS inverter gives the basis for calculating the electrical characteristics of logic gates. Consider the circuit shown in Figure 7.1. The input voltage V_{in} determines the conduction states of the two FETs Mn and Mp. This produces the output voltage V_{out} of the gate. Two types of calculations are needed to characterize a digital logic circuit. A **DC analysis** determines V_{out} for a given value of V_{in}. In this type of calculation, it is assumed that V_{in} is changed very slowly, and that V_{out} is allowed to stabilize before a measurement is made. A DC analysis provides a direct mapping of the input to the output, which in turn tells us the voltage ranges



pFET: $V_{Tp} < 0$

$\beta_p = k'_p \left(\frac{W}{L}\right)_p$

nFET: $V_{Tn} > 0$

$\beta_n = k'_n \left(\frac{W}{L}\right)_n$

Figure 7.1 The CMOS inverter circuit

that define Boolean logic 0 and logic 1 values. The second type of cha... terization is called a **transient analysis**. In this case, the input voltage... an explicit function of time $V_{in}(t)$ corresponding to a changing logic val... The response of the circuit is contained in $V_{out}(t)$. The delay betwee... change in the input and the corresponding change at the output is th... fundamental limiting factor for high-speed design. In this section we... concentrate on the DC analysis. The transient response is analyzed in th... next section.

The DC characteristics of the inverter are portrayed in the **voltag... transfer characteristic (VTC)**, which is a plot of V_{out} as a function of V_{in}. This is obtained by varying the input voltage V_{in} in the range from 0 V to V_{DD} and finding the output voltage V_{out}. The end point values are easily found with the aid of the circuits in Figure 7.2. If V_{in} is equal to 0 V as in Figure 7.2(a), Mn is off while Mp is on. Since the pFET is on, it connects the output to the power supply and gives $V_{out} = V_{DD}$. This defines the **output high voltage** of the circuit as

$$V_{OH} = V_{DD} \tag{7.1}$$

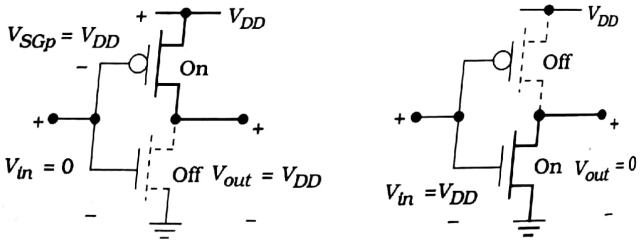
i.e., the highest output voltage is the value of the power supply V_{DD}. The opposite case with $V_{in} = V_{DD}$ is illustrated in Figure 7.2(b). This turns on Mn while Mp is in cutoff. The output node is then connected to 0 V (ground) through the nFET, defining the **output low voltage**

$$V_{OL} = 0 \text{ V} \tag{7.2}$$

The **logic swing** at the output is

$$V_L = V_{OH} - V_{OL} = V_{DD} \tag{7.3}$$

Since this is equal to the full value of the power supply, this is called a **full-rail output**.



(a) Low input voltage (b) High input voltage

Figure 7.2 V_{OH} and V_{OL} for the inverter circuit

The VTC for the circuit is obtained by starting with an input voltage of $V_{in} = 0$ V and then increasing it up to a value of $V_{in} = V_{DD}$. This results in the plot shown in Figure 7.3. The details can be understood by writing the device voltages in terms of the input and output voltages:

$$V_{GSn} = V_{in}$$
$$V_{SGp} = V_{DD} - V_{in} \tag{7.4}$$

Mn is in cutoff so long as $V_{in} \leq V_{Tn}$. Since the output voltage is high with a value $V_{out} = V_{DD}$, any input voltage in the range labeled as "0" can be interpreted as a logic 0 input. Increasing V_{in} causes a downward transition in the VTC. This is because the input voltage turns the nFET on while the pFET is still conducting. Note, however, that increasing V_{in} decreases V_{SGp}, so the pFET becomes a less efficient conductor and the output voltage falls. Mp goes into cutoff when

$$V_{in} = V_{DD} - |V_{Tp}| \tag{7.5}$$

For V_{in} greater than this value, $V_{out} = 0$ V since only the nFET is active. This shows that there is a range of input voltages that act as logic 1 input values as indicated by the "1" on the VTC.

The logic 0 and 1 voltage ranges are defined by the changing slope of the VTC. Point 'a' in the drawing is where the slope has a value of -1, and defines the **input low voltage** V_{IL}. By definition, a logic 0 input voltage is defined by

$$0 \leq V_{in} \leq V_{IL} \tag{7.6}$$

The second -1 slope point is labeled as 'b' and defines the **input high voltage** V_{IH}. This is used to define a logic 1 input voltage as

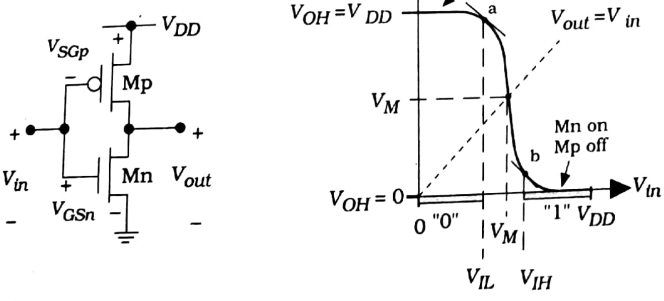


Figure 7.3 Voltage transfer curve for the NOT gate

$$V_{IH} \le V_{in} \le V_{DD}$$

The **voltage noise margins** are

$$VNM_H = V_{OH} - V_{IH} \quad (7.$$
$$VNM_L = V_{IL} - V_{OL} \quad (7.$$

for high and low states, respectively. The noise margins give a quantita... measure of how stable the inputs are with respect to coupled electromag... netic signal interference.

While it is possible to calculate the exact voltages that define logic ... and 1 input voltages, it is simpler to introduce the midpoint voltage ... shown in the VTC. This is defined as the point where the VTC intersec... the unity gain line that is defined by $V_{out} = V_{in} = V_M$. A value of $V_{in} = V_M$ ... in the transition region and does not represent a Boolean quantity. How... ever, for V_{in} less than V_M the input voltage is toward the logic 0 value, while $V_{in} > V_M$ indicates that the input is on the logic 1 side. Knowing the value of V_M thus tells us the center point for input transitions.

To calculate the midpoint voltage we set $V_{in} = V_{out} = V_M$ as shown in Figure 7.4. Equating the drain currents of the FETs gives

$$I_{Dn} = I_{Dp} \quad (7.9$$

but we need to find the operating region (saturation or non-saturation) of each FET before we can use the expression. Consider first the nFET and recall that the saturation voltage is given by

$$V_{sat} = V_{GSn} - V_{Tn} = V_M - V_{Tn} \quad (7.10)$$

where we have used $V_{in} = V_{GSn} = V_M$ in the second line. The drain-source voltage is $V_{DSn} = V_{out} = V_M$. Since V_{Tn} is a positive number,

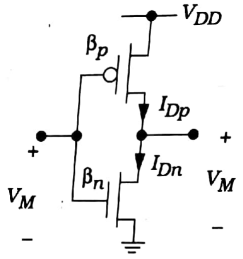


Figure 7.4 Inverter voltages for V_M calculation

$$V_{DSn} > V_{sat} = V_M - V_{Tn} \quad (7.11)$$

which says that Mn must be saturated. The same arguments can be applied to the pFET Mp since $V_{SGp} = V_{SDp}$. Using the saturation current equations from Chapter 6 gives

$$\frac{\beta_n}{2}(V_M - V_{Tn})^2 = \frac{\beta_p}{2}(V_{DD} - V_M - |V_{Tp}|)^2 \quad (7.12)$$

Dividing by β_p and taking the square root gives

$$\sqrt{\frac{\beta_n}{\beta_p}}(V_M - V_{Tn}) = V_{DD} - V_M - |V_{Tp}| \quad (7.13)$$

Simple algebra then gives the midpoint voltage as

$$V_M = \frac{V_{DD} - |V_{Tp}| + \sqrt{\frac{\beta_n}{\beta_p}}V_{Tn}}{1 + \sqrt{\frac{\beta_n}{\beta_p}}} \quad (7.14)$$

This equation shows that V_M is set by the nFET-to-pFET ratio

$$\frac{\beta_n}{\beta_p} = \frac{k'_n\left(\frac{W}{L}\right)_n}{k'_p\left(\frac{W}{L}\right)_p} \quad (7.15)$$

Since k'_n and k'_p are set in the processing, the ratio of the FET sizes establishes the switching point. It is important to remember that nFETs and pFETs have different mobility factors with a typical ratio of

$$\frac{k'_n}{k'_p} \approx 2 \text{ to } 3 \quad (7.16)$$

depending upon the details of the processing. This fact has a significant effect on the choices we make in both the sizing of individual transistors, and the types of circuits that are used in advanced VLSI designs. Note that, since C_{ox} is approximately the same for both FET types.

$$\frac{k'_n}{k'_p} = \frac{\mu_n}{\mu_p} = r \quad (7.17)$$

where r is the mobility ratio introduced in Chapter 5.

A **symmetrical inverter** VTC is one that has equal "0" and "1" input voltage ranges. This can be achieved by choosing

$$V_M = \frac{1}{2}V_{DD}$$

in equation (7.12). Rearranging gives us the design equation

$$\frac{\beta_n}{\beta_p} = \left(\frac{\frac{1}{2}V_{DD} - |V_{Tp}|}{\frac{1}{2}V_{DD} - V_{Tn}}\right)^2$$

This allows us to compute the transistor sizes for this particular choice V_M. Note that if $V_{Tn} = |V_{Tp}|$, then a symmetric design requires that

$$\beta_n = \beta_p$$

i.e., the device transconductance values of the two FETs are equal. It important to remember that β is proportional to the aspect ratio (W/L a MOSFET, and that (W/L) is the actual design variable.

Example 7.1

Consider a CMOS process with the following parameters

$$k'_n = 140\ \mu A/V^2 \qquad V_{Tn} = +0.70\ V$$
$$k'_p = 60\ \mu A/V^2 \qquad V_{Tp} = -0.70\ V$$

with $V_{DD} = 3.0$ V.

Consider the case where $\beta_n = \beta_p$. We can verify that this is a symmetrical design by calculating

$$V_M = \frac{3 - 0.7 + \sqrt{1}(0.7)}{1 + \sqrt{1}} = 1.5\ V$$

so that V_M is one-half the value of the power supply voltage. To achieve this design, we must choose the device aspect ratios such that

$$\frac{\beta_n}{\beta_p} = \frac{k'_n\left(\frac{W}{L}\right)_n}{k'_p\left(\frac{W}{L}\right)_p} = 1$$

where we recall that the process transconductance parameters k' are given by $k' = \mu_n C_{ox}$, and are set by the processing. For the present case we rearrange the expression to read

$$\left(\frac{W}{L}\right)_p = \frac{k'_n}{k'_p}\left(\frac{W}{L}\right)_n$$

so that

$$\left(\frac{W}{L}\right)_p = \left(\frac{140}{60}\right)\left(\frac{W}{L}\right)_n = 2.33\left(\frac{W}{L}\right)_n \tag{7.25}$$

This shows that the pFET must be about 2.33 times larger than the nFET.

Let us now examine the case where the nFET and the pFET have the same aspect ratio: $(W/L)_n = (W/L)_p$. With the values provided in the problem statement,

$$\frac{\beta_n}{\beta_p} = \frac{k'_n}{k'_p} = 2.33 \tag{7.26}$$

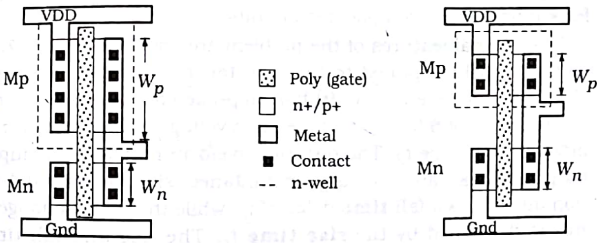
so that the midpoint voltage is given by

$$V_M = \frac{3 - 0.7 + \sqrt{2.33}\ (0.7)}{1 + \sqrt{2.33}} = 1.33\ V \tag{7.27}$$

This choice shifts V_M to a value that is smaller than $(V_{DD}/2)$.

Figure 7.5 illustrates the difference in the layout between an inverter that uses the two design styles. The channel length is the same for both transistors in the inverter, leaving the channel widths W_p and W_n as the design variables. In Figure 7.5(a), the pFET has a width of about $W_p \approx 2$ W_n which gives V_M of about $(V_{DD}/2)$. Equal size transistors are used in the layout of Figure 7.5(b), so that the circuit has $V_M < (V_{DD}/2)$. It is important to remember that we are only dealing with the DC characteristics at the moment. As we will see in the next section, the switching properties of the two designs are also affected by the aspect ratios.

The derivation and examples above illustrate the importance of the FET aspect ratios in the DC behavior of the logic gate. At the physical level, the relative device sizes contained in the ratio (β_n/β_p) determine the switching points. In general, increasing (β_n/β_p) decreases the value of the midpoint voltage V_M. This dependence is illustrated in the plot of Figure



(a) Larger pFET design (b) Equal aspect ratios

Figure 7.5 Comparison of the layouts for Example 7.1

7.6. With the parameters shown, a symmetrical design with $\beta_n = \beta_p$ gives $V_M = (V_{DD}/2) = 1.5$ V. Increasing the ratio to $(\beta_n/\beta_p) = 1.5$ gives $V_M = 1.$ V, while $(\beta_n/\beta_p) = 2.5$ decreases the midpoint voltage to $V_M = 1.31$ V. It is also possible to use a ratio of $(\beta_n/\beta_p) < 1$, which shifts the VTC toward the right, i.e., $V_M > (V_{DD}/2)$. However, this is rarely used since the p aspect ratios get quite large.

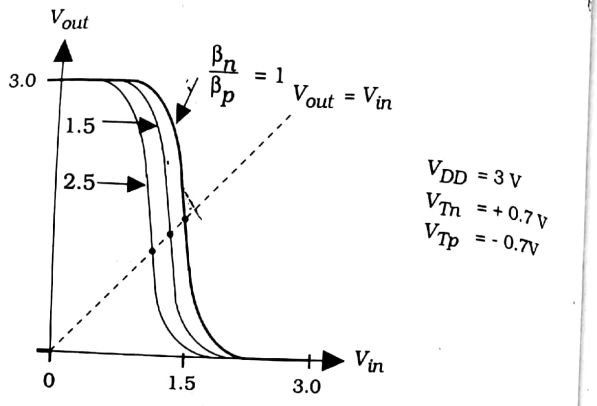


Figure 7.6 Dependence of V_M on the device ratio

7.2 Inverter Switching Characteristics

High-speed digital system design is based on the ability to perform calcu lations very quickly. This requires that logic gates introduce a minimum amount of time delay when the inputs change. Designing fast logic cir cuits is one of the more challenging (but critical) aspects of VLSI physical design. As with the DC analysis, analyzing the NOT gate provides a basis for studying more complicated circuits.

The general features of the problem are shown in Figure 7.7. An input voltage $V_{in}(t)$ is applied to the inverter, resulting in an output voltage $V_{out}(t)$. We assume that $V_{in}(t)$ has step-like characteristics and makes an abrupt transition from 0 to 1 (i.e., to a voltage of V_{DD}) at time t_1, and back down to 0 at time t_2. The output waveform reacts to the input, but the output voltage cannot change instantaneously. The output 1-to-0 transition introduces a **fall time** delay of t_f, while the 0-to-1 change at the output is described by the **rise time** t_r. The rise and fall times can be calculated by analyzing the electronic transitions of the circuits.

The rise and fall time delays are due to the parasitic resistance and

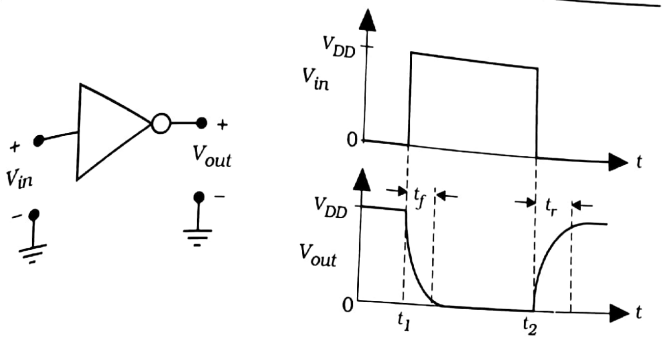
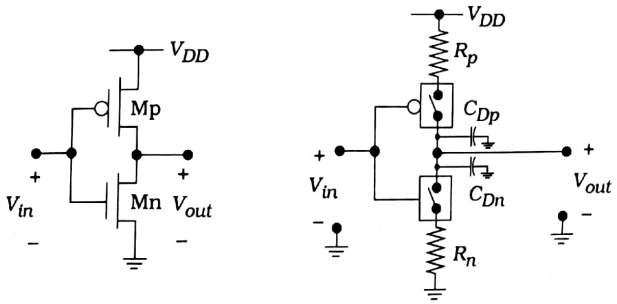


Figure 7.7 General switching waveforms

capacitances of the transistors. Consider the NOT circuit shown in Figure 7.8(a). Both FETs can be replaced by their switch equivalents, which results in the simplified RC model in Figure 7.8(b). It is worth recalling that the actual values of the components depend upon the device dimensions. Once we specify the aspect ratios $(W/L)_n$ and $(W/L)_p$, we can calculate R_n and R_p using

$$R_n = \frac{1}{\beta_n(V_{DD} - V_{Tn})}$$
$$R_p = \frac{1}{\beta_p(V_{DD} - |V_{Tp}|)} \qquad (7.28)$$

Knowing the layout dimensions of each FET allows us to find the capacitances C_{Dn} and C_{Dp} at the output node. The formulas are given by



(a) FET circuit (b) RC switch model equivalent

Figure 7.8 RC switch model equivalent for the CMOS inverter

$$C_{Dn} = C_{GSn} + C_{DBn} = \frac{1}{2}C_{ox}L'W_n + C_{jn}A_n + C_{jswn}P_n$$

$$C_{Dp} = C_{GSp} + C_{DBp} = \frac{1}{2}C_{ox}L'W_p + C_{jp}A_p + C_{jswp}P_p \qquad (7.$$

where we have added n and p subscripts to specify the nFET or pF quantities, respectively.[1] It is significant to remember that increasing t channel width of a FET increases the parasitic capacitance values.

There is one more important point that needs to be included before obtain a complete model. In a logic chain, every logic gate must dri another gate, or set of gates, to be useful. The number of gates is specifie by the **fan-out** (FO) of the circuit. The fan-out gates act as a **load** to th driving circuit because of their **input capacitance** C_{in}. Consider th inverter shown in Figure 7.9(a). The input capacitance of the inverter just the sum of the FET capacitances

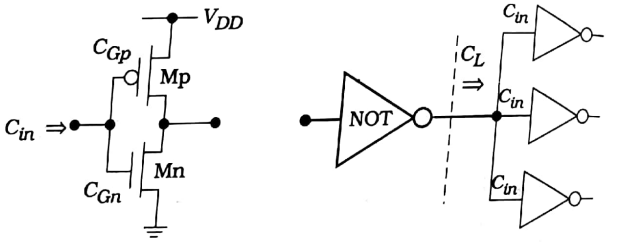
$$C_{in} = C_{Gp} + C_{Gn} \qquad (7.3$$

Figure 7.8(b) shows the effect of input capacitance for a fan-out of FO = 3 The input capacitance to each gate acts as an **external load capacitance** C_L to the driving gate. In this example, it is easily seen that

$$C_L = 3C_{in} \qquad (7.3$$

is the value of the load presented to the NOT gate.

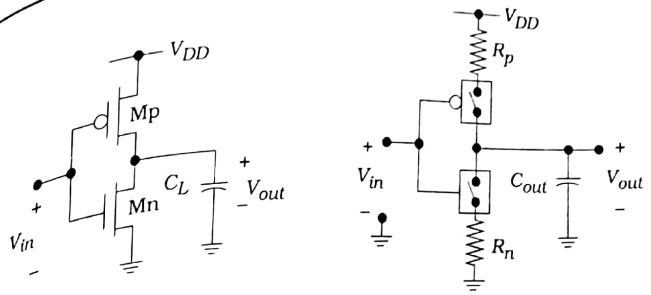
We may now calculate the switching times of the inverter. Figure 7.10 illustrates the general problem. A CMOS NOT gate is used to drive an external load capacitance C_L as in Figure 7.10(a). This gives the complete



(a) Single stage (b) Loading due to fan-out

Figure 7.9 Input capacitance and load effects

[1] Note that the source capacitances C_{Sp} and C_{Sn} do not enter the problem as they are at the power supply and ground, respectively, and have constant voltages.



(a) External load (b) Complete switching model

Figure 7.10 Evolution of the inverter switching model

switching model shown in Figure 7.10(b) where the total output capacitance is defined as

$$C_{out} = C_{FET} + C_L \qquad (7.32)$$

The FET capacitances shown earlier in Figure 7.8 have been merged into the single term

$$C_{FET} = C_{Dn} + C_{Dp} \qquad (7.33)$$

and are the parasitic internal contributions that cannot be eliminated. These add with C_L since all elements are in parallel. The total output capacitance C_{out} is the load that the gate must drive; the numerical value varies with the load.

Example 7.2

Let us apply this analysis to find the capacitances in the NOT gate shown in Figure 7.11. It is assumed that all dimensions have units of microns (μm).

First we will find the gate capacitances using

$$C_{Gp} = (2.70)(1)(8) = 21.6 \text{ fF} \qquad (7.34)$$
$$C_{Gn} = (2.70)(1)(4) = 10.8 \text{ fF}$$

Next, note that the overlap distance L_o is specified as 0.1 μm, which should be included in the area and perimeter factors in the junction capacitances. For the pFET, the p+ capacitance is

$$C_p = C_J A_{bot} + C_{jsw} P_{sw} \qquad (7.35)$$

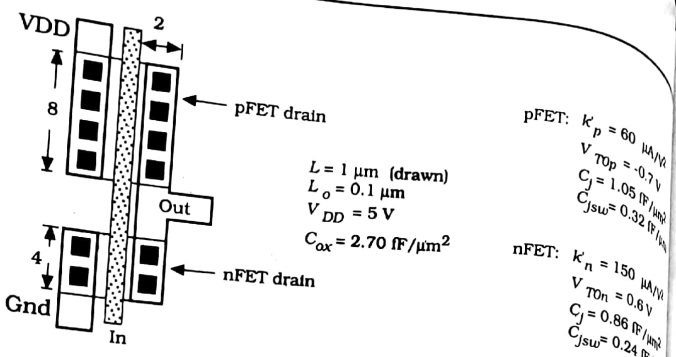


Figure 7.11 Example of capacitance calculations

so

$$C_p = (1.05)(8)(2.1) + (0.32)2(8+2.1) = 24.10 \text{ fF} \tag{7.36}$$

The total capacitance at the pFET drain is therefore given by

$$C_{Dp} = \frac{21.6}{2} + 24.10 = 34.9 \text{ fF} \tag{7.37}$$

The nFET drain is analyzed using the same approach. The n+ junction capacitance is

$$C_n = (0.86)(4)(2.1) + (0.24)(2)(4+2.1) = 10.15 \text{ fF} \tag{7.38}$$

so that

$$C_{Dn} = \frac{10.8}{2} + 10.15 = 15.55 \text{ fF} \tag{7.39}$$

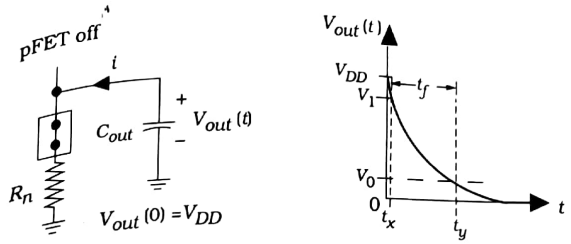
is the total capacitance at the drain of the nFET. Adding gives

$$\begin{aligned} C_{FET} &= C_{Dp} + C_{Dn} \\ &= 34.9 + 15.55 \\ &= 50.45 \text{ fF} \end{aligned} \tag{7.40}$$

as the total internal FET capacitance. The total capacitance at the output is

$$C_{out} = 50.45 + C_L \tag{7.41}$$

in fF, where C_L is the external load (also in fF).



(a) Discharge circuit (b) Output waveform

Figure 7.12 Discharge circuit for the fall time calculation

7.2.1 Fall Time Calculation

Let us start by calculating the output fall time t_f. We will shift the time origin such that V_{in} changes from 0 to V_{DD} at time $t = 0$. The initial condition at the output is $V_{out}(0) = V_{DD}$. When the input is switched, the nFET goes active while the pFET is driven into cutoff. In terms of the switch models, the nFET switch is closed and the pFET switch is open. This gives us the simplified discharge circuit shown in Figure 7.12(a). The capacitor C_{out} is initially charged to a voltage V_{DD}, and is allowed to discharge to 0 V through the nFET resistance R_n. The current leaving the capacitor is

$$i = -C_{out}\frac{dV_{out}}{dt} = \frac{V_{out}}{R_n} \tag{7.42}$$

which gives the differential equation for the discharge event. Solving with the initial condition $V_{out}(0) = V_{DD}$ results in the well-known form

$$V_{out}(t) = V_{DD}e^{-t/\tau_n} \tag{7.43}$$

where

$$\tau_n = R_n C_{out} \tag{7.44}$$

is the nFET **time constant** with units of seconds. The function is plotted in Figure 7.12(b).

The fall time is traditionally defined to be the time interval from $V_1 = 0.9\,V_{DD}$ to $V_0 = 0.1\,V_{DD}$, which are respectively known as the 90% and the 10% voltages as referenced to the full rail swing of V_{DD}. Rearranging the solution to the form

$$t = \tau_n \ln\left(\frac{V_{DD}}{V_{out}}\right) \tag{7.45}$$

allows us to calculate the time t needed to fall to a particular voltage. From the drawing we see that

$$\begin{aligned} t_f &= t_y - t_x \\ &= \tau_n \ln\left(\frac{V_{DD}}{0.1V_{DD}}\right) - \tau_n \ln\left(\frac{V_{DD}}{0.9V_{DD}}\right) \\ &= \tau_n \ln(9) \end{aligned}$$

where we have used the identity

$$\ln(a) - \ln(b) = \ln\left(\frac{a}{b}\right)$$

in the last step. Approximating $\ln(9) \approx 2.2$ gives the final result

$$t_f \approx 2.2\tau_n$$

as the fall time for the circuit. The output fall time in a generic digital gate is usually called the output **high-to-low time** t_{HL} and is identical the value computed here:

$$t_{HL} = t_f$$

The two symbols will be used interchangeably in the discussion.

7.2.2 The Rise Time

The rise time calculation follows in the same manner. Initially, the input voltage is at $V_{in} = V_{DD}$ and is switched to $V_{in} = 0$ V; we time shift this event to occur at $t = 0$ for simplicity. This turns on the pFET while simultaneously driving the nFET into cutoff, so that the simplified charging circuit of Figure 7.13(a) is valid. The output voltage at $t = 0$ is given by $V_{out}(0) = 0$ V.

The charging current is given by

$$i = C_{out}\frac{dV_{out}}{dt} = \frac{V_{DD} - V_{out}}{R_p}$$

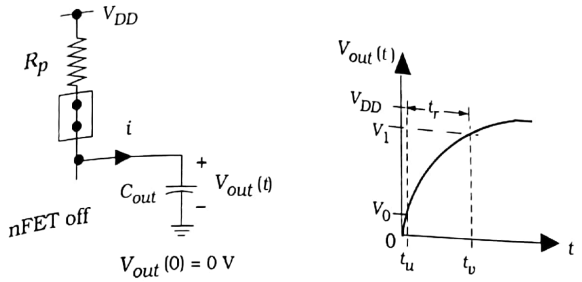
Solving and applying the initial condition gives the exponential form

$$V_{out}(t) = V_{DD}[1 - e^{-t/\tau_p}]$$

where the pFET time constant is defined by

$$\tau_p = R_p C_{out}$$

Figure 7.13(b) shows the output voltage as a function of time. The rise time is taken between 10% and 90% points such that



(a) Charge circuit (b) Output waveform

Figure 7.13 Rise time calculation

$$t_r = t_v - t_u$$ (7.53)

A little algebra yields the expression

$$t_r = \ln(9)\tau_p \approx 2.2\tau_p$$ (7.54)

for the rise time t_r. This has the same form as the fall time t_f because of the symmetry of the charge and discharge circuits. The rise time is identical to the output **low-to-high time** t_{LH}; the symbols will be used interchangeably.

The low-to-high time t_{LH} and the high-to-low time t_{HL} represent the shortest amount of time needed for the output to change from a logic 0 to logic 1 voltage, or from a logic 1 to a logic 0 voltage, respectively. Let us assume that the input is a square wave with a period of T sec such that the voltage is 0 for $(T/2)$ and V_{DD} for a $(T/2)$ time interval.[2] We then define the **maximum signal frequency** as

$$f_{max} = \frac{1}{t_{HL} + t_{LH}} = \frac{1}{t_r + t_f}$$ (7.55)

since this is the largest frequency that can be applied to the gate and still allow the output to settle to a definable state.[3] If the signal frequency exceeds f_{max}, the output voltage of the gate will not have sufficient time to stabilize to the correct value.

[2] This defines what is known as a 50% duty cycle.

[3] This definition assumes that t_{HL} and t_{LH} have the same order of magnitude to be useful.

Example 7.3

Consider an inverter circuit that has FET aspect ratios of $(W/L)_n = 6$, $(W/L)_p = 8$ in a process where

$$k'_n = 150\ \mu A/V^2 \qquad V_{Tn} = +0.70\ V$$
$$k'_p = 62\ \mu A/V^2 \qquad V_{Tp} = -0.85\ V$$

and uses a power supply voltage of $V_{DD} = 3.3$ V. The total output capacitance is estimated to be $C_{out} = 150$ fF. Let us compute the rise and fall times using the equations derived above.

Consider first the fall time. The pFET resistance is given by

$$R_p = \frac{1}{\beta_p(V_{DD} - |V_{Tp}|)} = \frac{1}{(62\times10^{-6})(8)(3.3-0.85)} = 822.9\ \Omega$$

The time constant for the charging event is computed using the RC product R_pC_{out} to find

$$\tau_p = (822.9)(150\times10^{-15}) = 123.43\ \text{ps}$$

where 1 ps (picosecond) is 10^{-12} sec. The rise time is

$$t_r = 2.2\tau_p = 271.55\ \text{ps}$$

The fall time is calculated in a similar manner. First, we find the nFET resistance

$$R_n = \frac{1}{\beta_n(V_{DD} - V_{Tn})} = \frac{1}{(150\times10^{-6})(6)(3.3-0.70)} = 427.35\ \Omega$$

so that the discharge time constant is

$$\tau_p = (427.35)(150\times10^{-15}) = 64.1\ \text{ps}$$

The fall time is

$$t_f = 2.2\tau_n = 141.0\ \text{ps}$$

Combining these results, the maximum signal frequency is

$$f_{max} = \frac{1}{t_r + t_f} = \frac{1}{(271.55 + 141.0)\times10^{-12}} = 2.42\ \text{GHz} \qquad (7.63)$$

where 1 GHz = 10^9 Hz. Although this is a very high frequency, it is important to remember that this refers only to a single inverter.

7.2.3 The Propagation Delay

The propagation delay time t_p is often used to estimate the "reaction" delay time from input to output. When we use step-like input voltages, the propagation delay is defined by the simple average of the two time intervals shown in Figure 7.14 by

$$t_p = \frac{(t_{pf} + t_{pr})}{2} \qquad (7.64)$$

In this expression, t_{pf} is the output fall time from the maximum level to the "50%" voltage line, i.e., from V_{DD} to $(V_{DD}/2)$; t_{pr} is the propagation rise time from 0 V to $(V_{DD}/2)$. Using the exponential equations for V_{out} we obtain

$$t_{pf} = \ln(2)\tau_n \qquad t_{pr} = \ln(2)\tau_p \qquad (7.65)$$

Approximating ln(2) ≈ 0.693 then gives

$$t_p \approx 0.35(\tau_n + \tau_p) \qquad (7.66)$$

The propagation delay time is a useful estimate of the basic delay, but does not provide detailed information on the rise and fall times as individual quantities. Propagation delays are commonly used in basic logic simulation programs.

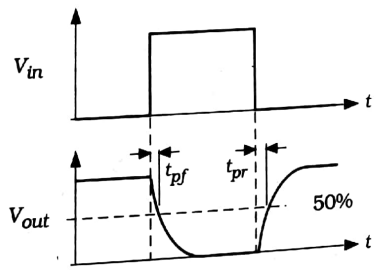


Figure 7.14 Propagation time definitions

7.2.4 General Analysis

The rise and fall time equations provide the basis for high speed CMOS design. We can manipulate them to show us how to design single logic gates and then characterize the behavior of the gates when used in logic networks.

To see the important factors, recall that the total output capacitance consists of two terms such that

$$C_{out} = C_{FET} + C_L \tag{7.67}$$

C_{FET} represents the parasitic capacitances of the transistors, while C_L is the external load. The layout geometry establishes the value of C_{FET}, but the load capacitance C_L varies with the application. Substituting this expression into the rise and fall time equations gives

$$t_r = 2.2R_p(C_{FET} + C_L)$$
$$t_f = 2.2R_n(C_{FET} + C_L) \tag{7.68}$$

which can be cast into the forms

$$t_r = t_{r0} + \alpha_p C_L$$
$$t_f = t_{f0} + \alpha_n C_L \tag{7.69}$$

These show that the rise and fall times are linear functions of the load capacitance C_L. The general behavior of both quantities is shown in Figure 7.15. Under zero-load conditions ($C_L = 0$), the inverter drives its own capacitances such that

$$t_r = t_{r0} = 2.2R_pC_{FET}$$
$$t_f = t_{f0} = 2.2R_nC_{FET} \tag{7.70}$$

are determined solely from the inverter parameters. When an external

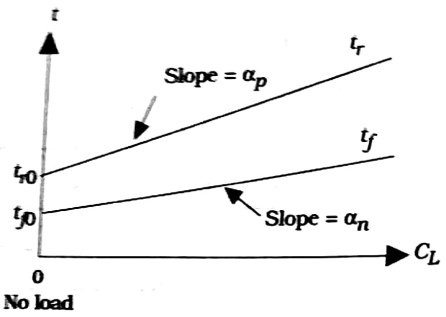


Figure 7.15 General behavior of the rise and fall times

load C_L is added, the switching times increase in a linear fashion. Large capacitive loads may cause problems because of longer delays. The dependence is described by the slope values

$$\alpha_p = 2.2R_p = \frac{2.2}{\beta_p(V_{DD} - |V_{Tp}|)} \tag{7.71}$$

and

$$\alpha_n = 2.2R_n = \frac{2.2}{\beta_n(V_{DD} - V_{Tn})} \tag{7.72}$$

Note that these are inversely proportional to the aspect ratios since

$$\beta_p = k_p'\left(\frac{W}{L}\right)_p, \qquad \beta_n = k_n'\left(\frac{W}{L}\right)_n \tag{7.73}$$

For a given load capacitance C_L, t_r and t_f can be reduced by using large FETs. However, increasing the aspect ratio of a transistor implies that it will consume more area on the chip, which in turn decreases the number of devices that can be placed on the die area allocated for the circuit. Designing for speed thus decreases the integration density of the circuit. This is called the **speed versus area trade-off** which says that

Fast circuits consume more area than slow circuits

Chip designers regularly face the problem of minimizing the switching delays without requiring excessive amounts of silicon "real estate," which is slang for chip area.

Example 7.4

Let us use the results of Example 7.3 to find the general delay equations for the case where the internal FET capacitance is $C_{FET} = 80$ fF.

The rise time t_r is controlled by the pFET that has a resistance of $R_p = 822.9\ \Omega$. The slope is given by

$$\alpha_p = 2.2R_p = 1,810.4\Omega \tag{7.74}$$

while

$$t_{r0} = 2.2R_pC_{FET} = 2.2(822.9)(80\times10^{-15}) = 144.9 \text{ ps} \tag{7.75}$$

The rise time can thus be written in the form

$$t_r = t_{r0} + \alpha_p C_L = 144.9 + 1.810C_L \text{ ps} \tag{7.76}$$

which requires that C_L be in units of fF.

For the fall time equation, we calculate

$$\alpha_n = 2.2(427.35) = 940.2\Omega$$

and

$$t_{f0} = 2.2(940.2)(80\times10^{-15}) = 165.5 \text{ ps}$$

yielding

$$t_f = 165.5 + 0.940C_L \text{ ps}$$

as the general expression.

As an example of using these equations, suppose that the load is specified as $C_L = 150$ fF. We compute

$$t_r = 144.9 + 1.810(150) = 416.4 \text{ ps}$$
$$t_f = 165.5 + 0.940(150) = 306.5 \text{ ps} \quad (7.5)$$

for the rise and fall times at the output. This corresponds to a maximum switching frequency for the gate of $f_{max} = 1.38$ GHz.

The relative values of $(W/L)_n$ and $(W/L)_p$ determine the shape of the output waveform. For example, if we design the circuit such that

$$R_p = R_n \quad (7.81)$$

then the output waveform is symmetrical with

$$t_r = t_f \quad (7.82)$$

To equalize the resistances we must design the circuit such that

$$\beta_p(V_{DD} - |V_{Tp}|) = \beta_n(V_{DD} - V_{Tn}) \quad (7.83)$$

is satisfied. If $V_{Tn} = |V_{Tp}|$, then the requirement reduces to

$$\beta_p = \beta_n \quad (7.84)$$

which gives the DC midpoint voltage at $V_M = (V_{DD}/2)$. This illustrates the fact that the nFET/pFET ratio (β_n/β_p) determines the DC midpoint voltage, while the individual values of β_n and β_p establish the switching times t_f and t_r, respectively.

7.2.5 Summary of the Inverter Circuit

It is worth taking the time to summarize the results of our study to this point. The electrical characteristics of an isolated CMOS inverter are established by two sets of parameters:

- The processing variables, such as k' and V_T values, and parasitic capacitances,

and,

- The transistor aspect ratios $(W/L)_n$ and $(W/L)_p$.

VLSI designers do not have any control over the processing parameters, as they are set by the details of the manufacturing sequence. Device sizing thus becomes the critical issue in high-speed circuit design.

System design is accomplished by using cascades of logic gates to perform the necessary binary operations. In electrical terms, the logic flow path establishes the load capacitance C_L seen by each gate. The choice of aspect ratios is the key to achieving the desired transient response of a chain of gates.

7.3 Power Dissipation

An important characteristic of CMOS integrated circuits is the power dissipated by a particular design technique. The general problem is shown in Figure 7.16. The current I_{DD} flowing from the power supply to ground gives a dissipated power of

$$P = V_{DD}I_{DD} \quad (7.85)$$

Since the value of the voltage supply V_{DD} is assumed to be a constant, we can find the value of P by studying the nature of the current flow. We usually divide the currents into DC and dynamic (or switching) contributions, so let us write

$$P = P_{DC} + P_{dyn} \quad (7.86)$$

where P_{DC} is the DC term and P_{dyn} is due to dynamic switching events.

The DC contribution can be calculated by examining the voltage transfer curve reproduced in Figure 7.17(a). When the input voltage V_{in} is stable at a low logic 0 value, the nFET Mn is off; as seen earlier in Figure 7.2, there is no direct current flow path between V_{DD} and ground. Ideally, the

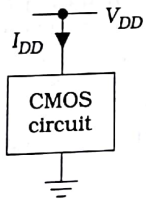


Figure 7.16 Origin of power dissipation calculation

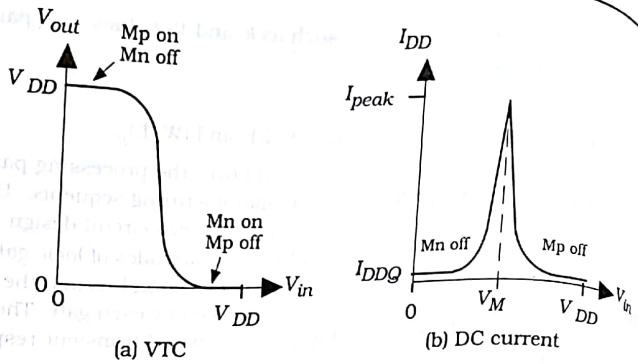


Figure 7.17 DC current flow

DC current flow for this case would be $I_{DD} = 0$, but in a realistic circuit small **leakage currents** exist.[4] The value is denoted as I_{DDQ} and is called the **quiescent** leakage current. When V_{in} is switched, the current flow reaches a peak value I_{peak} at V_M as shown in Figure 7.17(b). However, when the input reaches a logic 1 voltage, then the pFET Mp turns off, once again preventing a direct current flow path. If we assume that the inputs are in stable 0 or 1 states as in an idle system, the DC power dissipation is given by

$$P_{DC} = V_{DD} I_{DDQ} \tag{7.87}$$

The leakage current I_{DDQ} is usually quite small, with a typical value on the order of a picoampere per gate. The value of P_{DC} is thus quite small. This consideration was a major factor in the move to CMOS in the mid-1990's.

To find the dynamic power dissipation P_{dyn}, we use a square-wave input voltage $V_{in}(t)$ as shown in Figure 7.18(a). The waveform has a period T corresponding to a switching frequency of

$$f = \frac{1}{T} \tag{7.88}$$

with units of Hertz; the frequency is the number of cycles completed in one second. During the first half-cycle, the input voltage is at a value $V_{in} = 0$. This turns on the pFET Mp as shown in Figure 7.18(b). Since the nFET is off, the current i_{DD} flows through Mp and charges C_{out} to a voltage of $V_{out} = V_{DD}$. During the second half-cycle, the input voltage is high, turning on the nFET Mn. This causes the discharge event illustrated in Figure

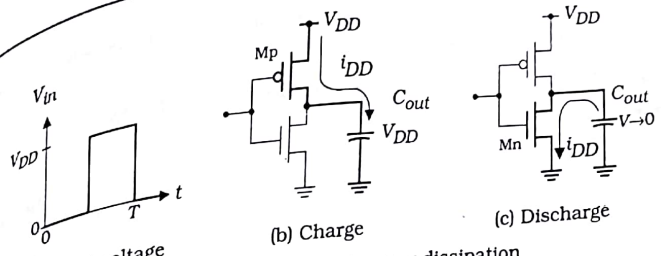


Figure 7.18 Circuit for finding the transient power dissipation

7.18(c) where V_{out} decays to 0 V. The dynamic power P_{dyn} arises from the observation that a complete cycle effectively creates a path for current to flow from the power supply to ground: during the charge event, current flows to the capacitor C_{out} while the discharge path to ground completes the circuit.

To calculate P_{dyn}, we note that the charging event leaves C_{out} with a voltage of $V_{out} = V_{DD}$. This corresponds to a stored electric charge on the capacitor of

$$Q_e = C_{out} V_{DD} \tag{7.89}$$

which has units of coulombs. When the capacitor is discharged through the nFET, the same amount of charge is lost. The average power dissipated over a single cycle with a period T is

$$P_{av} = V_{DD} I_{DD} = V_{DD}\left(\frac{Q_e}{T}\right) \tag{7.90}$$

Substituting for Q_e gives

$$P_{sw} = C_{out} V_{DD}^2 f \tag{7.91}$$

as the switching power. Combining the DC and dynamic power terms gives the total power as

$$P = V_{DD} I_{DDQ} + C_{out} V_{DD}^2 f \tag{7.92}$$

which will usually be dominated by the dynamic term. This illustrates an extremely important point:

- The dynamic power dissipation is proportional to the signal frequency

In other words, a fast circuit dissipates more power than a slow circuit. If we double the switching speed, then the dynamic power dissipation doubles. These are simply statements of the physical law that we must pro-

vide energy to induce a change in the circuit. It is not possible to switch a circuit without expending energy.

7.4 DC Characteristics: NAND and NOR Gates

The basic calculations introduced for the inverter circuit can be used to analyze NAND and NOR gates. Both the DC and transient characteristics can be obtained with relatively simple techniques. In this section we will examine the relationship between device sizes and the transitions described by the VTC.

7.4.1 NAND Analysis

Let us start with the NAND2 gate illustrated in Figure 7.19. We will analyze the case where like-polarity FETs have the same aspect ratio. This means that both pFETs are described by β_p and both nFETs have the same β_n. Since the pFETs are in parallel while the nFETs are in series, the circuit behaves quite differently from the simple inverter.

The presence of two independent inputs implies that more than one VTC curve is needed to describe the circuit. Suppose that we look for transitions where V_{out} is initially high at V_{DD} and then falls to 0 V when inputs are changed. Figure 7.20(a) summarizes the possible starting points that can lead to this situation. In case (i), both V_A and V_B are at 0 V and then switched to the bottom line condition where $V_A = V_B = V_{DD}$ such that $V_{out} = 0$ V. Since both inputs are increased at the same time, this describes the case for simultaneous input switching. The other two possibilities (ii) and (iii) describe cases where only a single input is changed. For example, in (ii) V_A is changed from 0 V to V_{DD} while V_B is held constant at V_{DD}. These three possibilities lead to the three distinct transitions shown in the plot of Figure 7.20(b). This shows that the simultaneous switching case is "pushed to the right" compared to the single-switched input cases.

It is instructive to calculate the value of the midpoint voltage V_M for the

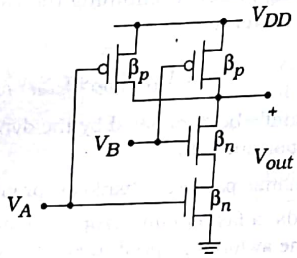
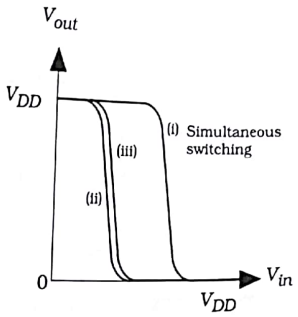


Figure 7.19 NAND2 logic circuit

	V_A	V_B	V_{out}
(i)	0	0	V_{DD}
(ii)	0	V_{DD}	V_{DD}
(iii)	V_{DD}	0	V_{DD}
	V_{DD}	V_{DD}	0

(a) Transition table



(b) VTC family

Figure 7.20 NAND2 VTC analysis

case of simultaneous switching using layout drawings. The circuit problem is illustrated in Figure 7.21, where W_n and W_p are the nFET and pFET channel widths, respectively. All transistors are assumed to have the same channel length L. Now then, for this case both input voltages V_A and V_B are equal to V_M. On the layout plot, both gates are thus at the same potential and can be connected to simplify the calculations.

Consider the nFETs first. In Figure 7.22(a), the layout is shown in its original form with two separate series-connected transistors. Let us "merge" the two gates together into one to obtain the patterning shown in Figure 7.22(b). If we ignore the n+ region that separates the two gates, then the structure can be approximated as a single nFET with an aspect ratio of $(W_n/2L)$ as shown. Since the original nFETs each had a device transconductance of β_n, the single equivalent transistor is described by the value $(\beta_n/2)$.

The pFETs can be combined in a similar manner. The original parallel-connected transistors are illustrated in Figure 7.23(a). Owing to the paral-

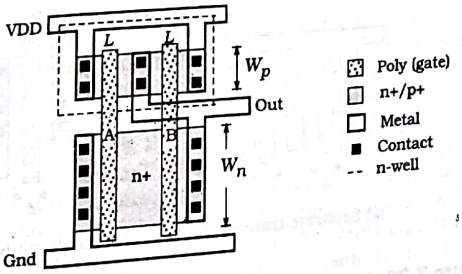
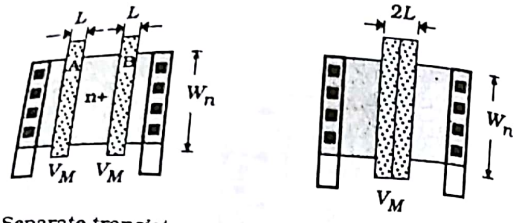


Figure 7.21 Layout of NAND2 for V_M calculation



(a) Separate transistors (b) Single equivalent FET

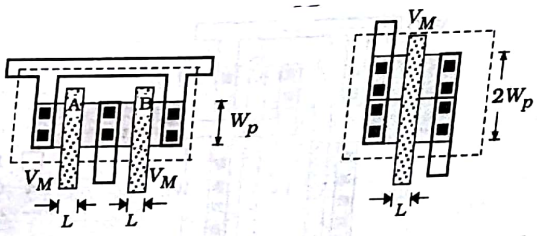
Figure 7.22 Simplification of the series-connected nFETs

lel wiring, the left and right sides are electrically the same point, so that the two may be simplified into the single gate structure shown in Figure 7.23(b). In this case, the two combine to act as a single pFET with an aspect ratio of $(2W_p/L)$. If the original devices each have β_p, then the equivalent structure acts as a pFET with $2\beta_p$.

Let us now use these results to find V_M for the case of simultaneous switching. Replacing the transistor pairs by their single-FET equivalents gives the inverter circuit in Figure 7.24, where the nFET and pFET transconductances are $(\beta_n/2)$ and $2\beta_p$, respectively. The calculation then proceeds in the same manner as for the "normal" NOT gate. Both transistors are saturated, so equating currents gives

$$\frac{(\beta_n/2)}{2}(V_M - V_{Tn})^2 = \frac{(2\beta_p)}{2}(V_{DD} - V_M - |V_{Tp}|)^2 \qquad (7.93)$$

Taking square roots of both sides and solving for the midpoint voltage results in the expression



(a) Separate transistors (b) Single equivalent FET

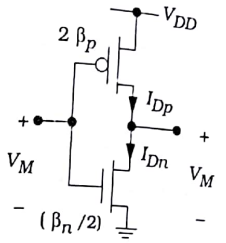


Figure 7.24 Simplified V_M circuit for the NAND2 gate

$$V_M = \frac{V_{DD} - |V_{Tp}| + \frac{1}{2}\sqrt{\frac{\beta_n}{\beta_p}}V_{Tn}}{1 + \frac{1}{2}\sqrt{\frac{\beta_n}{\beta_p}}} \qquad (7.94)$$

This has the same form as the NOT gate in equation (7.14), except that the square root term is multiplied by a factor of (1/2). This reduces the denominator, which is why the VTC curve is shifted toward the right. If we apply the same reasoning to an N-input NAND gate, the simultaneous switching point is found to be

$$V_M = \frac{V_{DD} - |V_{Tp}| + \frac{1}{N}\sqrt{\frac{\beta_n}{\beta_p}}V_{Tn}}{1 + \frac{1}{N}\sqrt{\frac{\beta_n}{\beta_p}}} \qquad (7.95)$$

The right shift is due to the series-connected nFETs, since their resistances add.

4.2 NOR Gate

The NOR2 gate can be analyzed using the same techniques. We assume that the nFETs have the same β_n and that both pFETs are described by β_p as shown in the basic circuit of Figure 7.25. To construct VTC, note that $V_{out} = V_{DD}$ requires that $V_A = V_B = 0$ V. If either input (or both) are switched to logic 1 values, then the output will fall to $V_{out} = 0$ V. The three combinations are listed in the function table of Figure 7.26(a). As with the NAND2 gate, there are three distinct transitions shown in the VTC family of Figure 7.26(b). Case (i) describes the simultaneous switching event where both V_A and V_B are increased from 0 V toward V_{DD}. This case is the leftmost plot in the VTC family, exactly opposite to that found for the NAND2. Single-input switching cases (ii) and (iii) are distinct, but are

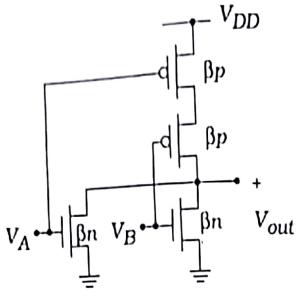


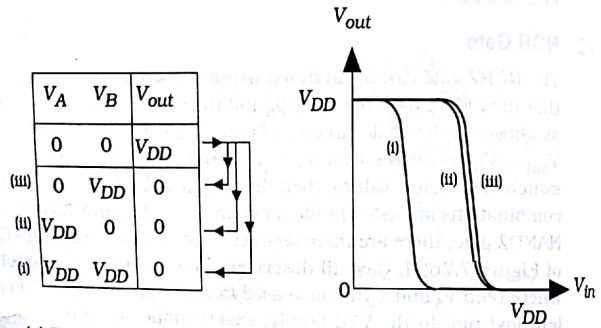
Figure 7.25 NOR2 circuit

The techniques of combining series and parallel transistors may be used to compute V_M for the simultaneous switching case. Since the nFETs are in parallel, they may be combined to a single equivalent nFET with a transconductance of $2\beta_n$. The series-connected pFETs act as a single pFET with $(\beta_p/2)$ which gives rise to the simplified equivalent circuit in Figure 7.27. Equating the saturation currents using the effective transconductance values gives us

$$\frac{(2\beta_n)}{2}(V_M - V_{Tn})^2 = \frac{(\beta_p/2)}{2}(V_{DD} - V_M - |V_{Tp}|)^2 \tag{7.96}$$

This may be solved to give

$$V_M = \frac{V_{DD} - |V_{Tp}| + 2\sqrt{\frac{\beta_n}{\beta_p}}V_{Tn}}{1 + 2\sqrt{\frac{\beta_n}{\beta_p}}} \tag{7.97}$$



	V_A	V_B	V_{out}
	0	0	V_{DD}
(iii)	0	V_{DD}	0
(ii)	V_{DD}	0	0
(i)	V_{DD}	V_{DD}	0

(a) Transition table

(b) VTC family

Figure 7.26 NOR2 VTC construction

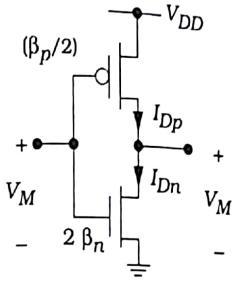


Figure 7.27 NOR2 V_M calculation for simultaneous switching

Comparing this with the NOT and NAND expressions shows that the only difference is the factor of 2 multiplying the square root term. This increases the denominator, which decreases the value of V_M from that of an inverter with a device ratio of (β_n/β_p). The midpoint voltage for an N-input NOR gate is

$$V_M = \frac{V_{DD} - |V_{Tp}| + N\sqrt{\frac{\beta_n}{\beta_p}}V_{Tn}}{1 + N\sqrt{\frac{\beta_n}{\beta_p}}} \tag{7.98}$$

It is worthwhile noting that the NAND and NOR gates tend have opposite behaviors with respect to the reference NOT gate VTC.

As a final comment, we note that both the NAND and NOR gates exhibit low DC power dissipation values of

$$P_{DC} = V_{DD}I_{DDQ} \tag{7.99}$$

since there is no direct current flow path from the power supply to ground when the inputs are stable logic 0 or logic 1 values. The low power characteristic of the gates is due to the use of complementary pairs and series-parallel structuring of the transistor arrays. Dynamic power is still present in the general form

$$P_{sw} = C_{out}V_{DD}^2 f_{gate} \tag{7.100}$$

which shows the dependence on gate switching frequency f_{gate}. Since it takes more than a single input to switch the gate, f_{gate} is different from the basic switching frequency used for the inverter. This is discussed in more detail later.

7.5 NAND and NOR Transient Response

Transient switching times often represent the limiting factor in designing a digital logic chain. In this section we will examine how the FET topology and device sizing affect the operational speed of the gate.

7.5.1 NAND2 Switching Times

Consider the NAND2 gate shown in Figure 7.28. The total output capacitance is denoted as

$$C_{out} = C_{FET} + C_L \tag{7.101}$$

where C_L is the external load and

$$C_{FET} = C_{Dn} + 2C_{Dp} \tag{7.102}$$

represents the parasitic internal FET capacitances. Note that there are two contributions of C_{Dp} since two pFETs are connected to the output node. The drawing identifies the transistors by their resistance values

$$R_p = \frac{1}{\beta_p(V_{DD} - |V_{Tp}|)}, \qquad R_n = \frac{1}{\beta_n(V_{DD} - V_{Tn})} \tag{7.103}$$

The transient calculations are based on finding RC time constants for the charge time (t_r or t_{LH}) and fall time (t_f or t_{HL}) for the transitions. The procedure is complicated by the presence of two inputs. We will concentrate on estimating the worst-case values of the switching times.

Let us consider the rise time t_r first. The output voltage is initially at a value $V_{out}(0) = 0$ V and is then charged to V_{DD}. If only one pFET is conducting, we obtain the simplified charging circuit shown in Figure 7.29(a) where C_{out} charges through a pFET resistance R_p. Since this looks like the charging circuit for a simple inverter, we can write

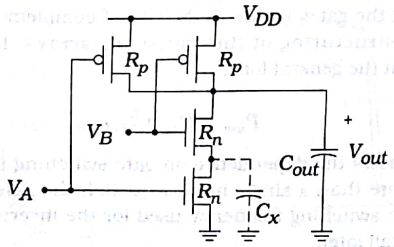
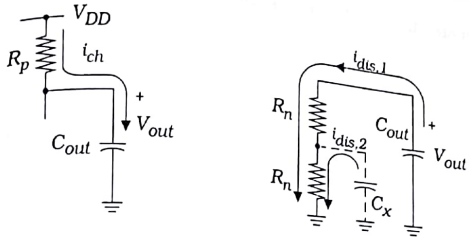


Figure 7.28 NAND2 circuit for transient calculations



(a) Charging circuit (b) Discharging circuit

Figure 7.29 NAND2 subcircuits for estimating rise and fall times

$$V_{out}(t) = V_{DD}[1 - e^{-t/\tau_p}] \tag{7.104}$$

where

$$\tau_p = R_p C_{out} \tag{7.105}$$

is the time constant. The rise time is thus given by

$$t_r \approx 2.2\tau_p \tag{7.106}$$

This is considered to be a "worst-case" situation since only one pFET is charging C_{out}. Note that this can be cast into the linear form

$$t_r = t_0 + \alpha_0 C_L \tag{7.107}$$

where

$$t_0 = 2.2R_p C_{FET} \tag{7.108}$$

is the zero-load value, and

$$\alpha_0 = 2.2R_p \tag{7.109}$$

is the slope of t_r as a function of the load capacitance C_L. If both pFETs are conducting, then the equivalent resistance is lowered to $(R_p/2)$ since the two are in parallel; this would be the "best-case" event, i.e., the one with the shortest charging time. Design is usually based on worst-case analysis since we want to insure that the circuit operates under all conditions.

The situation is more complicated when we analyze the fall time t_f where C_{out} discharges through the series-connected nFET chain. RC modeling of each device leads to the "ladder" network shown in Figure 7.29(b). While the main item of interest is discharging C_{out}, the situation is com-

plicated by the presence of the inter-FET capacitance C_X between the two n-channel transistors. In the worst-case analysis, C_X will have charge that will flow through nFET MnA to ground. Since the current through a FET is limited by its aspect ratio (W/L), the discharge rate is limited by the current that MnA can maintain.

The discharge can be described by modeling the output voltage in the exponential form

$$V_{out}(t) = V_{DD}e^{-t/\tau_n} \tag{7.110}$$

such that the time constant is given by the **Elmore formula** as

$$\tau_n = C_{out}(R_n + R_n) + C_X R_n \tag{7.111}$$

This estimates the time constant as the superposition of time constants

$$\tau_n = \tau_{n1} + \tau_{n2} \tag{7.112}$$

where

$$\tau_{n1} = C_{out}(R_n + R_n) \tag{7.113}$$

is the time constant for C_{out} discharging through two nFETs, each with a resistance R_n; this is shown by the current $i_{dis,1}$ in the drawing. The other term

$$\tau_{n2} = C_X R_n \tag{7.114}$$

is the time constant for C_X discharging through one nFET with a resistance R_n. This is corresponds to the discharge current $i_{dis,2}$. The fall time t_f is then given by

$$t_f \approx 2.2\tau_n \tag{7.115}$$

Substituting the time constant expression transforms this into

$$t_f \approx 2.2[(C_{FET} + C_L)(2R_n) + C_X R_n] \tag{7.116}$$

Grouping terms results in the linear expression

$$t_f = t_1 + \alpha_1 C_L \tag{7.117}$$

with a zero-load delay of

$$t_1 = 2.2R_n(2C_{FET} + C_X) \tag{7.118}$$

and a slope of

$$\alpha_1 = 4.4R_n \tag{7.119}$$

where the multiplier is from (2×2.2). Although we are able to write t_f as a

linear function of C_L, both the zero-load delay and the slope are affected by the series-connected nFETs in the discharge circuitry.

The Elmore formulation of time constants for RC ladder-type networks illustrates that series-connected FETs lead to longer delays in CMOS circuits. To understand this comment, let us rewrite equation (7.111) as

$$\tau_n = R_n(2C_{out} + C_X) \tag{7.120}$$

In this form, we can interpret the time constant as R_n multiplying an effective capacitance with a value

$$C_{eff} = 2C_{out} + C_X \tag{7.121}$$

which is larger than twice the output capacitance. Alternately, we may write

$$\tau_n = C_{out}(2R_n) + C_X R_n \tag{7.122}$$

which clearly shows the effect of the series-connected FETs in the term $2R_n$ and the increase due to the parasitic capacitance C_X. Regardless of the interpretation one chooses, it is important to remember that series-connected FET chains can lead to excessive logic delays.

7.5.2 NOR2 Switching Times

The analysis of the NOR2 transients proceeds in the same manner. Figure 7.30 shows the circuit with FET resistances and the capacitances. The output capacitance for any gate is given by the general form

$$C_{out} = C_{FET} + C_L \tag{7.123}$$

For the NOR2 circuit, the internal capacitance can be broken down into components as

$$C_{FET} = 2C_{Dn} + C_{Dp} \tag{7.124}$$

since there are two nFETs connected to the output node but only one

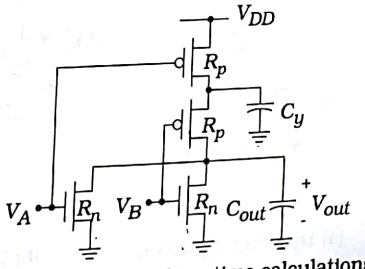


Figure 7.30 NOR2 circuit for switching time calculations

pFET. The inter-FET capacitance C_y represents the parasitic contribu
tions between the two pFETs.

Figure 7.31 shows the subcircuits for the output transients. The fa
time t_f may be computed using the worst-case circuit in Figure 7.31a,
where only one nFET acts to discharge the output capacitance. We thus
write the output voltage as

$$V_{out}(t) = V_{DD}e^{-t/\tau_n} \tag{7.125}$$

with

$$\tau_n = R_n C_{out} \tag{7.126}$$

as the time constant. The fall time is then given by

$$t_f \approx 2.2\tau_n \tag{7.127}$$

which is identical to that for a simple inverter. Expanding C_{out} gives the linear dependence

$$t_f = t_1 + \alpha_1 C_L \tag{7.128}$$

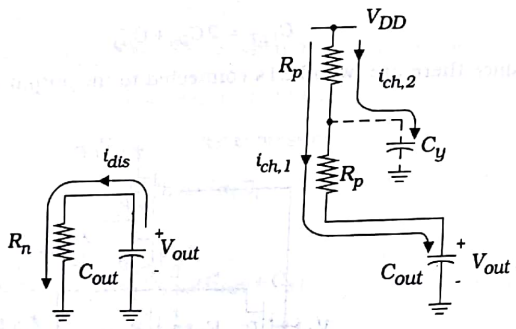
where the zero-load delay is

$$t_1 = 2.2R_n C_{FET} \tag{7.129}$$

and the slope is

$$\alpha_1 = 2.2R_n \tag{7.130}$$

These results are similar to the NOT gate, but it is important to remember that C_{FET} is larger for the NOR2 gate.



(a) Discharging circuit (b) Charging circuit

Figure 7.31 Subcircuits for the NOR2 transient calculations

The charging circuit for finding the rise time t_r is shown in Figure 7.31(b). We will write the output voltage in the exponential form

$$V_{out}(t) = V_{DD}[1 - e^{-t/\tau_p}] \tag{7.131}$$

However, since C_y will be charged during this event, we must use the Elmore formula to find the time constant. The two paths are shown as $i_{ch,1}$ and $i_{ch,2}$ in the drawing. The primary charge path due to $i_{ch,1}$ is described by a time constant

$$\tau_1 = C_{out}(R_p + R_p) \tag{7.132}$$

while that associated with $i_{ch,2}$ is

$$\tau_2 = C_y R_p \tag{7.133}$$

Superposing gives the total effective time constant in the form

$$\begin{aligned} \tau_p &= \tau_1 + \tau_2 \\ &= C_{out}(2R_p) + C_y R_p \end{aligned} \tag{7.134}$$

such that the rise time is

$$t_r = 2.2\tau_p \tag{7.135}$$

Since the series-connected pFETs introduce a large time constant, the rise time may be quite large compared to the fall time. Substituting for C_{out} gives the linear equation

$$t_r = t_0 + \alpha_0 C_L \tag{7.136}$$

where

$$t_0 = 2.2R_p(2C_{FET} + C_y) \tag{7.137}$$

and

$$\alpha_0 = 4.4R_p \tag{7.138}$$

characterize the dependence of t_r on C_L. As with the NAND2 gate, the presence of series-connected FETs slows down the associated switching time.

7.5.3 Summary

The analyses above illustrate that the NAND and NOR gates exhibit complementary characteristics at both the DC and transient levels. This arises because they are constructed using complementary series-parallel transistor arrangements.

While the DC characteristics are important, most design effort is

directed toward minimizing delays through logic chains. The study above allows us to make some general statements about NAND and NOR gates as compared to the simpler NOT circuit. First, we have seen that the rise time can be written in the form

$$t_r = t_0 + \alpha_0 C_L \tag{7.139}$$

while the fall time has the same structure with

$$t_f = t_1 + \alpha_1 C_L \tag{7.140}$$

The constants (t_0 and α_0 for the rise time, and t_1 and α_1 for the fall time) depend upon the parasitic transistor resistances and capacitances. The constants are the smallest for a NOT gate, so we often use it as a reference. This, of course, is because the inverter consists of only two FETs. In general, adding complementary transistor pairs increases the delay times because C_{FET} is increased. The number of inputs to a logic gate is called the **fan-in** (FI). Since every input is connected to a complementary pair, we can state that

- Switching delays increase with the fan-in.

This says, for example, a NAND3 gate will be slower than a NAND2 gate if the two use the same size transistors. Of course, the actual delay depends upon the value of the load capacitance C_L such that

- Switching delays increase with the external load.

Since logic functions are implemented using cascades of gates, the effect of this dependence varies with the circuit.

Let us summarize the results of the NAND and NOR analysis. As with the inverter, the electrical characteristics of these gates are set by

- The processing variables and
- The aspect ratios $(W/L)_n$ and $(W/L)_p$ of every FET

Furthermore, series transistors introduced us to the problem of parasitic capacitance between the two devices. This factor leads us to make one additional statement

- The details of the layout geometries affect the transient response of the logic gate.

We thus conclude that the physical layout and structure of the circuitry is a critical factor in designing high-speed logic networks.

7.6 Analysis of Complex Logic Gates

The analysis techniques developed for the NAND and NOT circuits may be extended to analyze complex CMOS logic gates with AOI and OAI structuring. The most important problem is the transient delay associated with

series-connected FETs.

Consider the complex logic gate shown in Figure 7.32. This implements the logic function

$$f = \overline{x \cdot (y + z)} \tag{7.141}$$

with series-parallel FET arrays. The aspect ratio values shown in the drawing are the critical parameters that affect the rise and fall times. The fall time is governed by the nFETs. If we assume that they are all the same size with

$$\left(\frac{W}{L}\right)_{nx} = \left(\frac{W}{L}\right)_{ny} = \left(\frac{W}{L}\right)_{nz} \tag{7.142}$$

then the nFET resistance R_n can be used to describe each one. The worst-case fall time will occur when $x = 1$, but only one of the ORed inputs y or z is 1. This results in a 2-FET series pair that must handle the discharge of the output capacitor

$$C_{out} = C_{FET} + C_L \tag{7.143}$$

With the capacitance C_n in the chain, the time constant is

$$\tau_n = R_n C_n + 2R_n C_{out} \tag{7.144}$$

which gives a fall time of

$$\begin{aligned} t_f &= 2.2\tau_n \\ &= 2.2R_n[C_n + 2(C_{FET} + C_L)] \\ &= t_1 + \alpha_1 C_L \end{aligned} \tag{7.145}$$

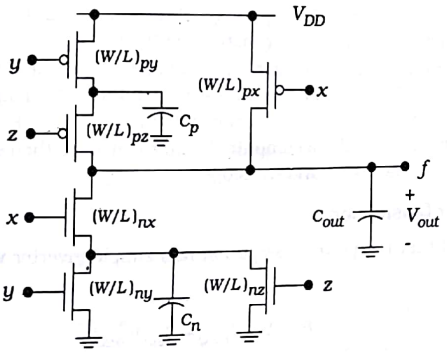


Figure 7.32 Complex logic gate circuit

$$a_{NOR2} = \left(\frac{3}{4}\right)\left(\frac{1}{4}\right) = \frac{3}{16} \tag{7.158}$$

The NAND2 gate can be analyzed in the same manner. For this gate, the truth table shows that $p_0 = (1/4)$ and $p_1 = (3/4)$ so

$$a_{NAND2} = \left(\frac{3}{4}\right)\left(\frac{1}{4}\right) = \frac{3}{16} \tag{7.159}$$

has the same value as the NOR2 gate. If we look at 3-input gates, the truth tables give

$$a_{NOR3} = \frac{7}{64} = a_{NAND3} \tag{7.160}$$

Similarly, we can calculate

$$a_{XNOR2} = \frac{1}{4} = a_{XOR2} \tag{7.161}$$

since $p_0 = (1/4) = p_1$. The technique can be applied to an arbitrary gate.

The limit on this simple treatment is that, in practice, we rarely have input combinations that occur with equal probability. More advanced techniques have been developed to handle these situations. The interested reader is directed to Reference [2] for an excellent discussion of the details. Reference [8] is a very thorough analysis of power dissipation and low-power design.

7.7 Gate Design for Transient Performance

High-speed circuits are limited by the switching time of individual gates. Logic formation determines the series and parallel connections of the transistors. The aspect ratios are the critical design parameters for both the DC and transient switching times. Once these are specified and the transistors are created in the layout, all of the parasitics are set.

The DC switching characteristics are often considered less important than the switching speed. It is common to design a gate to have the desired transient times, and then check the DC VTC to insure that it is acceptable. This approach is based on the fact that the individual nFET and pFET aspect ratios determine the switching response, while the DC transition point is a result of the ratio of the nFET to pFET values. For example, the value of β_n/β_p gives V_M for an inverter, while t_r depends primarily on β_p and t_f is established by β_n.

The design philosophy used to select aspect ratios varies with the situation. A straightforward approach is to use the inverter as a reference and then attempt to design other gates that have approximately the same switching times. Since the NOT gate is the simplest, it can be built using

relatively small transistors. We will use the device transconductance

$$\beta = k'\left(\frac{W}{L}\right) \tag{7.161}$$

as being equivalent to the aspect ratio.

Figure 7.34(a) shows an inverter with device sizes specified by β_p and β_n, which we will assume are known. These set the rise and fall times t_r and t_f for the circuit, which serve as the reference switching times. Since both transistors drive the same capacitance, the difference is in the resistance values

$$R_p = \frac{1}{\beta_p(V_{DD} - |V_{Tp}|)}, \qquad R_n = \frac{1}{\beta_n(V_{DD} - V_{Tn})} \tag{7.162}$$

Recall that a symmetrical inverter has

$$\beta_n = \beta_p \tag{7.163}$$

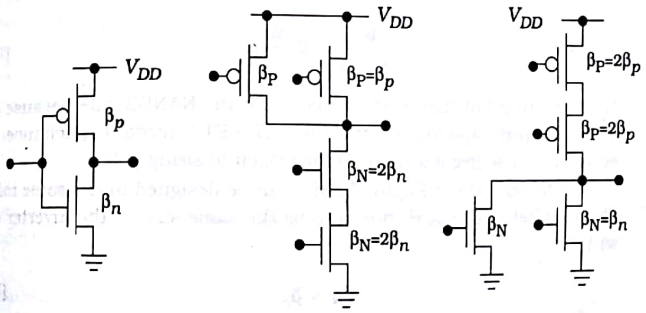
and requires the device sizes to be related by

$$\left(\frac{W}{L}\right)_p = r\left(\frac{W}{L}\right)_n \tag{7.164}$$

where

$$r = \frac{k'_n}{k'_p} \tag{7.165}$$

is the process transconductance ratio. A nonsymmetrical design that uses equal size transistors such that $\beta_n > \beta_p$ is also commonly used as a reference.



(a) Inverter (b) NAND2 (c) NOR2

Figure 7.34 Relative FET sizing

Let us use these values to find the device sizes β_P and β_N for the NAND gate in Figure 7.34(b) with the philosophy that we want to achieve similar rise and fall times. Consider first the parallel pFETs. Since the worst-case situation is where only one transistor contributes to the rise time, we select the same size as the inverter:

$$\beta_P = \beta_p \tag{7.165}$$

The actual rise time t_r will be longer than that of the inverter because C_L is larger. The series-connected nFET chain has to be modeled as two series-connected resistors between the output and ground, with a total value of

$$R = R_N + R_N \tag{7.166}$$

where

$$R_N = \frac{1}{\beta_N(V_{DD} - V_{Tn})} \tag{7.167}$$

Using the inverter as a reference, we set

$$R = R_n = 2R_N \tag{7.168}$$

Substituting,

$$\frac{1}{\beta_n(V_{DD} - V_{Tn})} = \frac{2}{\beta_N(V_{DD} - V_{Tn})} \tag{7.170}$$

which has the solution

$$\beta_N = 2\beta_n \tag{7.171}$$

i.e., the series-connected nFETs are twice as large as the inverter transistor:

$$\left(\frac{W}{L}\right)_N = 2\left(\frac{W}{L}\right)_n \tag{7.172}$$

The resulting fall time t_f will be larger in the NAND2 gate because of the larger output capacitance and the FET-FET internal capacitance. However, this does give a structured approach to sizing gates.

The NOR2 gate in Figure 7.34(c) can be designed in the same manner. The parallel nFETs are chosen to be the same size as the inverter device with

$$\beta_N = \beta_n \tag{7.173}$$

since this gives the worst-case discharge. The series-connected pFET resistances add to a total of $2R_p$. Equating this to the inverter resistance

R_p gives

$$\frac{1}{\beta_p(V_{DD} - |V_{Tp}|)} = \frac{2}{\beta_P(V_{DD} - |V_{Tp}|)} \tag{7.174}$$

so that

$$\beta_P = 2\beta_p \tag{7.175}$$

indicating that the pFETs are twice as large as the inverter transistors:

$$\left(\frac{W}{L}\right)_P = 2\left(\frac{W}{L}\right)_p \tag{7.176}$$

The main problem is that pFETs are intrinsically slow, so that the value of $(W/L)_p$ may be large to begin with.

This technique can be extended to larger chains. For n series-connected FETs, the size must be n times larger than the inverter value. The NAND3 gate in Figure 7.35(a) would thus be designed with

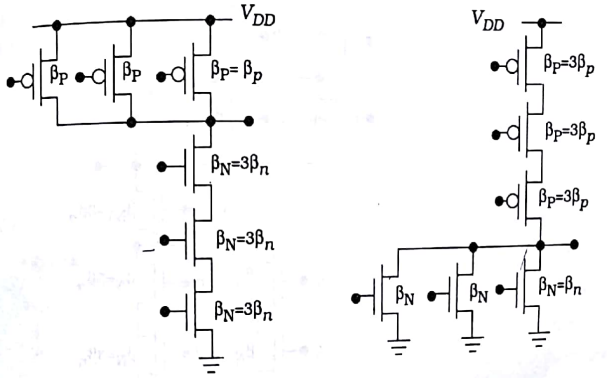
$$\beta_N = 3\beta_n, \qquad \beta_P = \beta_p \tag{7.177}$$

such that

$$\left(\frac{W}{L}\right)_N = 3\left(\frac{W}{L}\right)_n, \qquad \left(\frac{W}{L}\right)_P = \left(\frac{W}{L}\right)_p \tag{7.178}$$

while the NOR3 gate in Figure 7.35(b) would have

$$\beta_N = \beta_n, \qquad \beta_P = 3\beta_p \tag{7.179}$$



(a) NAND3 (b) NOR3

Figure 7.35 Sizing for 3-input gates

with

$$\left(\frac{W}{L}\right)_N = \left(\frac{W}{L}\right)_n \qquad \left(\frac{W}{L}\right)_P = \beta\left(\frac{W}{L}\right)_p$$

Since the reference values β_n and β_p are arbitrary, the sizes can be adjusted as needed to accommodate reasonable values. Also note that if we select a symmetric inverter design with $\beta_n = \beta_p$, then the resulting gates will also be approximately symmetric.

Complex logic gates can be designed in the same manner. Consider the gate in Figure 7.36 that has an output of

$$f = \overline{(a \cdot b + c \cdot d) \cdot x}$$

using series-parallel structuring. Consider the nFET array first. Any discharge event will have current flow through a minimum of three series-connected nFETs. The device sizes would all be the same with the value

$$\beta_N = 3\beta_n = \beta_{N1}$$

The pFET array is a little different. The worst-case charge path is through two series-connected transistors on the left side of the circuit. The sizes would be

$$\beta_P = 2\beta_p$$

for the pFETs in the inputs *a*, *b*, *c*, and *d*. The *x*-input pFET is alone, so

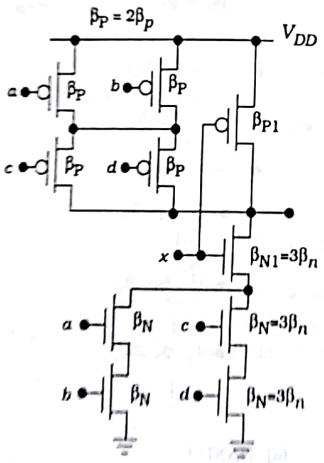


Figure 7.36 Sizing of a complex logic gate

that we can select the size as being the same as for an inverter

$$\beta_{P1} = \beta_p$$

Alternately, the choice

$$\beta_{P1} = \beta_P = 2\beta_p$$

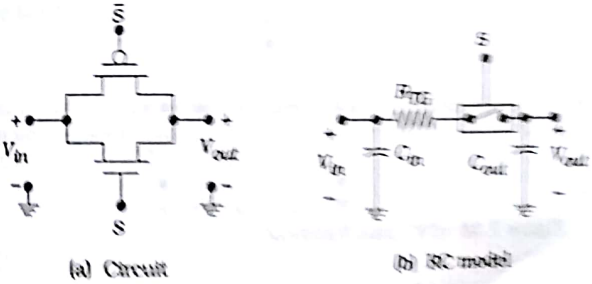
may lead to simpler layout since only a single size pFET would be used. Note that the two options for β_{P1} result in different input capacitances for the *x*-input.

Although this approach provides a nice structured methodology, it leads to large transistors. The designer must decide whether the real estate consumption is worth the added speed. The problem becomes more complicated as the number of FETs increases since the FET-to-FET parasitic capacitance terms in the Elmore time constant formula will also increase. In practice, we may just select a standard cell that meets the area allocation and then find the overall speed of the logic cascade. If the design is not fast enough, we can apply some of the techniques in the next chapter to find a better design.

7.8 Transmission Gates and Pass Transistors

Transmission gates consist of an nFET/pFET pair wired in parallel as shown in Figure 7.37(a). The RC switching model shown in Figure 7.37(b) consists of a TG resistance R_{TG} and capacitances that account for the parasitic contributions of both FETs. Even though the FETs are in parallel, one usually dominates the conduction process at any given time. For example, a logic 0 transmission is controlled by the nFET. Owing to this, a reasonable approximation for the linear resistance is

$$R_{TG} = \max(R_n, R_p)$$



(a) Circuit (b) RC model

Figure 7.37 Transmission gate modeling