

## 2. Defects, Failures, and Faults

### 2.1 Introduction

Failures in integrated circuits can be characterized according to their duration, permanent or temporary, and by their mode, parameter degradation or incorrect design. *Permanent failures*, also called *hard failures*, are usually caused by breaks due to mechanical rupture, some wear-out phenomenon, or an incorrect manufacturing procedure. They occur less frequently than temporary failures, which are failures that cannot be replicated. Temporary failures, also called *soft failures*, are categorized as transient or intermittent. Transient *failures* are induced by external perturbation such as power supply fluctuations or radiation. *Intermittent failures* are usually due to some degradation of the component parameters. Figure 2.1 shows classifications of IC failures.

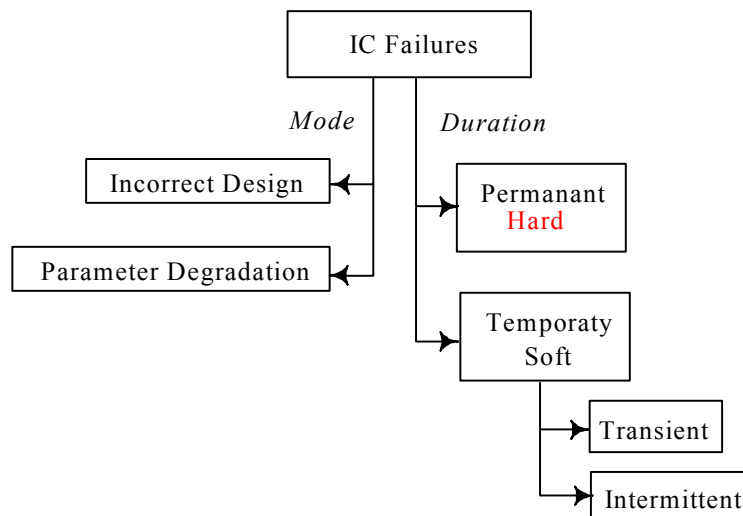


Figure 2.1 Classifications of IC Failures

In this chapter we examine the physical defects and their manifestation on the electrical level as *failure modes*. Failure modes are then modeled as *faults* at the logic and behavioral level. The mapping from the physical domain to the logic and behavioral domains facilitates the detection process of failures. Figure 2.2a shows a representation of the scanning electron microscope (SEM) image of an embedded particle in an IC that produced a short [Henderson 1997]. It is clear that the short possibly spans several metal lines. We assume here, for simplicity, that it affects only two lines, L1 and L2, as shown in Fig. 2.2b. On the logic level, the failure mode may be

interpreted in different ways depending on the technology used and the relationship of the two metal lines. Depending on the other line, four different possibilities are illustrated in Fig. 2.2c and d. The second metal line may be power or ground, and thus the output of the inverter is held high or low. This results in *stuck-at-1* and *stuck-at-0* faults, respectively. The second line may, instead, be the output of the same gate, and the short results in a *feedback bridging fault*. Finally, if the second line is the output of another gate, a *simple bridging fault* results.

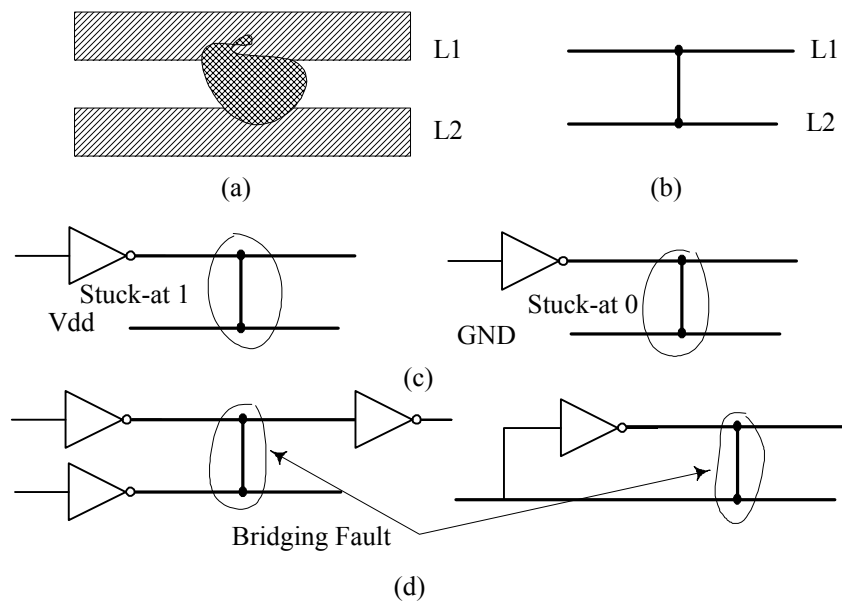


Figure 2.2 Mapping Physical Defects onto Faults  
 a) Metal Mask with Dust Causing Extra Metal, b) Failure Mode: a Short  
 c) Faults on the Logic Level, Stuck-at Faults  
 d) Bridging Faults

We first describe the main causes of physical failures in MOS technology. The manifestation of these defects as two main failure modes, short and open, is then explained. The remainder of the chapter is devoted to fault modeling. Notice that the importance of the fault is to facilitate detection of the defect even if it is not a faithful representation of this defect. Thus a fault model may represent different failures [Sachdev 1998].

## 2.2 Physical Defects

Physical defects are due to different *failure mechanisms*, which are largely dependent on the technology and layout of the circuit. A list of the principal failure mechanisms is given below.

- Surface and bulk effect
  - Passivation pits and cracks
  - Gate oxide breakdown
  - Pinholes or thin spots in oxide

Electrical over-stress  
Surface potential instability

- Metalization and metal semiconductor
  - Open metal at oxide steps
  - Wire bonding failure
  - Inter-metallic compound formation
- Electromigration
- Package related failures
  - Mass transport of metal atoms
  - Momentum exchange with electrons

Reference is made to the various steps used to fabricate an IC, which are several repetitions of depositing, conducting, and insulating material, oxidation, photolithography, and etching [Sze 1983]. Transistor formation is assumed to occur where the gate (poly) and active layers (diffusion) intersect. Other reasons for physical defects are the instabilities in the process conditions. These include: (1) random fluctuation in the actual environment; for example, turbulent flow of gases used for diffusion and oxidation; (2) inaccuracies in the control or furnace; and (3) variation in the physical and chemical parameters of the material, such as fluctuation in the density and viscosity of the photoresist, and water and gas contaminants. Physical defects may often occur because of human errors such as mishandling wafers or processing equipment. We next examine three more frequently occurring types of defects: (1) extra and missing material, (2) oxide breakdown, and (3) electromigration.

### 2.2.1 Extra and Missing Material

Extra and missing material defects may be caused by dust particles on the mask or wafer surface, or in the processing chemicals. During the photolithography steps, these particles lead to unexposed photoresist areas or resist pinholes, thus causing unwanted material or unwanted etching of the material on a layer [Strapper 1980]. For this reason, these defects are sometimes referred to as *photo* or *lithography* defects [Strapper 1976]. These defects cause extra and missing polysilicon, active and metal. An extra metal may cause shorts such as those described in the preceding section. An extra poly over an active area will form an additional unwanted transistor.

### 2.2.2 Oxide Breakdown

The primary cause of poor yield in gate oxide is the formation of pinhole defects due to insufficient oxygen at the interface of silicon (Si) and silicon dioxide (SiO<sub>2</sub>), chemical contamination, nitride cracking during field oxidation, and crystal defects [Schugraf 1994]. Oxide breakdown may also be due to the operational conditions. For example,

as the oxide thickness decreases without reducing the power supply voltage, the electric field across the oxide is increased and causes local breakdown. A large discharge through the oxide may cause the same damage. The charge can be transferred from other objects or from people, causing what is commonly known as Electrostatic Discharge (ESD). The damage may be permanent if the energy was enough to cause material to flow, for example, between the gate and the channel of a transistor [Bakoglu 1990]. Also, hot electrons cause the trapping of a charge in the gate oxide and eventually break it down [Schugraf 1994].

### 2.2.3 Electromigration

Electromigration is one of the major failure mechanisms in IC interconnects [Black 1969], [Ho 1982] [Merchant 1982]. It is caused by the transport of metal atoms when a current flows through the wire. Because of a low melting point, aluminum has large self-diffusion properties, which increase its electromigration liability. As atoms are displaced, three main phenomena may happen (Fig. 2.3). The metal line breaks and causes an open in the interconnect. The displaced atoms may accumulate and spread in very thin lines (whiskers) to another metal line in the vicinity, or they may migrate through the silicon dioxide insulation to another metal line on an adjacent layer (hillcock) [Bakoglu 1990].

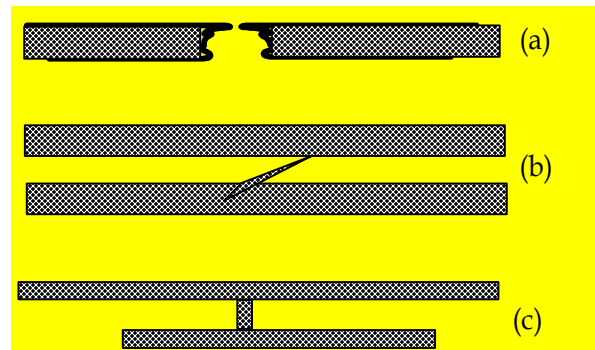


Figure 2.3 Electromigration's Effects  
 (a) Open In a Line, (b) Short between Two Lines on the Same Layer (whisker);  
 (c) Short Between Lines on Different Layers (hillcock) [Bakoglu 1990]

Electromigration is related to stresses applied to the IC, such as increased current densities or temperature. The mean time to failure (MTTF) is proportional to the width and thickness of the metal lines [Woods 1984] and inversely proportional to the current density. This cause of IC failure is becoming more prevalent now that the trend is toward very deep submicron technology (VDSM):  $-2\lambda \leq 0.5\mu\text{m}$ . Complete characterization of physical defects is difficult. However, simulation studies using statistical approach based on data from experimental studies

have proved to be very effective in developing a systematic classification of physical defects in MOS [Maly 1984, Shen 1985, Walker 1987, Ferguson 1988].

In this section we highlighted the main failure mechanisms that are current in present technologies. Could we have skipped this information without affecting the knowledge to be gained from this book? The answer is, probably, yes. However, to increase the IC yield, it is becoming more important to understand the failure mechanisms. Interest in this area is evident by the large number of fault analysis articles recently published [ITC 1997, D&T 1998].

## 2.3 Failure Modes

Most manufacturing failure mechanisms are manifested on the circuit level as *failure modes*. The most common failure modes are open and short interconnections or parameter degradation.

### 2.3.1 Opens

*Opens* may be formed when missing material spans a line or via on an interacting layer, splitting a net into unconnected branches. If the failure occurs at a fan-out, several gates will have an open gate. For example, a missing metal defect can break a metal line, or a missing first -level via defect can cause an open via. Opens may also be caused by extra material if, for example, this material completely covers a via so that the lower layer is no longer connected to the upper layer.

### 2.3.2 Shorts

Some extra material defects may result in connecting a metal line to another on the same layer, or to one on a layer below it through incomplete vias, or to a layer above it by blocking a via. Gate oxide breakdown causes *resistive* shorts between the gate and the source, drain, or channel. A resistive short is the type of failure that connects the gate to another terminal through a finite, but not necessarily zero, resistance [Hawkins 1986].

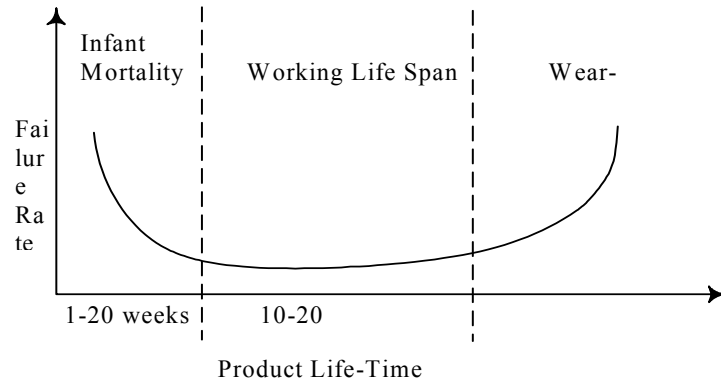


Figure 2.4 Failure Rate versus Product Lifetime

An IC can fail at different stages of its lifetime. The failure rate over the lifetime is given by the well-known bathtub curve shown in Fig. 2.4. Defects that escape visual and optical scanning may cause many chips to fail within 20 weeks of their operation (infant mortality period). At the end of this period, the failure rate tends to stabilize for 10 to 20 years (normal lifetime period). Eventually, due to excessive use of the components, there is an exponential increase in the failure rate (wear-out period). In present ICs, a major factor causing wear-out failure is excessive heat dissipation. This is due to high device density and increased circuit activities. Other failure modes that do not have roots in physical defects result from noise induced within the circuit, such as crosstalk and ground bounce. These faults are discussed in Section 2.11

Table 2.1 Most Commonly Used Fault Models

Fault Model	Description
Single Stuck-at Faults (SSA)	One line takes the value 0 or 1.
Multiple Stuck-at Faults	Two or more lines have fixed values, not necessarily the same.
Bridging Faults	Two or more lines that are normally independent become electrically connected
Stuck-Open Faults (SOP)	A failure in a pull-up or pull-down transistor in a CMOS logic device causes it to behave like a memory element.
Stuck-On Faults (SON)	A transistor is always conducting.
Delay Faults	A fault is caused by delays in one or more paths in the circuit.
Intermittent Faults	Caused by internal parameter degradation. Incorrect signal values occur for some but not all states of the circuit. Degradation is progressive until permanent failure occurs
Transient Faults	Incorrect signal values caused by coupled disturbance. Coupling may be via power bus capacitive or inductive coupling. Includes both internal and external sources as well as particle irradiation.

## 2.4 Faults

Failure modes are manifested on the logical level as incorrect signal values. A *fault* is a model that represents the effect of a failure by means of the change that is produced in the system signal. Several defects are usually mapped to one fault model. It is a many-to-one mapping. But some defects may also be represented by more than one fault model. Table 2.1 lists the common fault models [Mourad 1987]. Fault models have the advantage of being a more tractable representation than physical failure modes. As a model, the fault does not have to be an exact representation of the defects, but rather, to be useful in detecting the defects. For example, the most common fault model assumes *single stuck-at* (SSA) lines even though it is clear that this model does not accurately represent all actual physical failures. The rationale for continuing to use the stuck-at-fault model is the fact that it has been satisfactory in the past. Also, test sets that have been generated for this fault type have been effective in detecting other types of faults. However, as with any model, a fault cannot represent all failures, and a stuck-at fault is no longer sufficient for present circuits and technologies. With the advent of MOS technology, it has become evident that other fault models are needed to represent more accurately the failure modes in this technology.

## 2.5 Stuck-at Faults

### 2.5.1 Single Stuck-at Faults

A single stuck-at (SSA) fault represents a line in the circuit that is fixed to logic value 0 or 1. One may think of it as representing a short between the faulty line and the ground or  $V_{dd}$  rail. Examples of failure modes that manifest themselves as stuck-at faults are shown in Fig. 1.5. The first example shows how a defective resistance caused the input of the bipolar XOR gate to short to ground. This is represented by a stuck-at-0 fault. Similarly, in the CMOS gate, the breakdown in gate oxide causing a resistive short between gate and source results in a stuck-at-0 fault on the input. Another example is shown in Fig. 2.2c, which resulted from extra metal shorting the output of a gate to the power or ground rail.

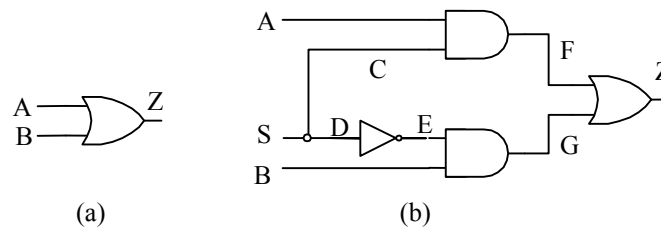


Figure 2.5 (a) A 2-input OR Gate, (b) A 2-to-1 Multiplexer

Independent of how accurately the stuck-at fault represents the physical defect, we next continue investigating how to generate patterns that detect these faults. In Chapter 1 we developed a truth table listing the good and faulty responses of a simple AND gate to stuck-at faults on the input - output leads. Here we first consider the dual function, OR gate, and use the concepts of controllability and observability to determine the test patterns. Consider a stuck-at 0 on input  $A$  of the OR gate in Fig. 2.5a. To determine whether or not  $A$  is stuck at 0, we apply logic 1 to it and sensitize the result to the output of the gate,  $Z$ . To observe the outcome on the output, we need to hold the other input,  $B$ , low. If the output is low, this confirms that  $A$  actually is SA0 and it is detected; otherwise, it is not stuck at 0. Therefore, the pattern  $AB = (10)$  will detect the  $A/0$ . Due to the symmetry of the circuit, the pattern  $AB = (01)$  will detect  $B/0$ . Using the same reasoning, we should be able to generate the test pattern for  $A/1$ , which is  $AB = (00)$ . Again, because of symmetry, the same pattern detects  $B/1$ . To detect SA faults on the output, we need only control this node to the complement logic value to which it is stuck. Thus controlling  $Z$  to 1 detects  $Z/0$ . This can then be accomplished by either of the patterns that detect  $A/0$  or  $B/0$ . Similarly, to detect  $Z/1$ , we force it to 0, which can be done with the pattern  $AB = (00)$ . Summing up the SSA faults for the two-input OR gate, we showed that: (1), Three patterns are sufficient to detect all faults (10, 01, 00); (2), The three faults,  $A/1$ ,  $B/1$ , and  $Z/1$ , are detected by the same pattern (00); (3), Detecting stuck-at 0 on the inputs guarantees detection of the same fault on the output.

Let us next consider a more complex logic gate, a 2-to-1 multiplexer (MUX), as shown in Fig. 2.5b. This circuit helps illustrate the following important features: fan-out and reconverging fan-out. The input  $S$  has two fan-out lines,  $C$  and  $D$ . Usually,  $S$  is called the *stem* and the other lines the *branches*. It is obvious that lines  $C$  and  $D$  are electrically identical to the primary input,  $S$ . They might be considered indistinguishable from  $S$ , as the three lines are all stuck at 1 or stuck at 0 simultaneously. However, there are cases when a failure on one of the branches does not affect the other branch or the stem. For example, due to electromigration, a branch may be detached from the

stem. If the input to the inverter is open, branch  $D$  is no longer electrically connected to the stem  $S$  or to the other branch,  $C$ . With this explanation, we find that the MUX has nine lines in total. However, here we concentrate on the stuck-at faults on three lines,  $S$  and its branches,  $C$  and  $D$ . A table similar to that constructed for the AND gate in Chapter 1 is developed for the MUX. In Table 2.2 the first column lists all possible test patterns, a combination of the three primary inputs. The second column lists the fault-free (FF) responses to these patterns. The other columns give the faulty response for each of the stuck-at faults listed in the first row. Whenever the response diverges from the good circuit response, column FF, the corresponding input combination is a test set that detects the fault. There is only one test pattern,  $SAB = (010)$ , that detects  $C/1$ . This pattern also detects  $S/1$ , a fault that is also detected by  $SAB = (001)$ . Detecting  $C/1$  implies the detection of  $S/1$ . We observe also that there is a common pattern,  $SAB = (110)$ , that detects both  $C/0$  and  $S/0$ , although each of these faults is also detected by other patterns,  $SAB = (101)$  and  $(111)$ , respectively. A similar relationship exists between SA faults on  $D$  and  $S$ . The pattern  $SAB = (101)$  is the only pattern that detects  $D/0$ , and it also detects  $S/0$ . In addition, the test pattern,  $SAB = (001)$  detects  $D/1$  and  $S/1$ . We then conclude that detecting the SA faults on the branches  $C$  and  $D$  leads to the detection of the SA faults on the stem,  $S$ .

Table 2.2: Stuck-at Faults for a 2-to-1 MUX

Input SAB	Response						
	FF	S/0	S/1	C/0	C/1	D/0	D/1
000	0	0	0	0	0	0	0
001	1	1	<b>0</b>	1	1	1	<b>0</b>
010	0	0	<b>1</b>	0	<b>1</b>	0	0
011	1	1	1	1	1	1	<b>0</b>
100	0	0	0	0	0	0	0
101	0	<b>1</b>	0	0	0	<b>1</b>	0
110	1	<b>0</b>	1	<b>0</b>	1	1	1
111	1	1	1	<b>0</b>	1	1	1

Constructing similar tables for circuits other than elementary gates is not a feasible solution to determine the test patterns. The higher the number of lines in the circuit, the more complex the problem. As we have discovered in Chapter 6, test pattern generation is a complex proposition, and whenever possible, it is advantageous to minimize the fault list. For this we generalize the observations made on the simple gates by defining more formally in subsequent sections of this chapter the concepts of *equivalence* and *dominance*.

### 2.5.2 Multiple Stuck-at Faults

A defect may cause multiple stuck-at faults. That is, more than one line may be stuck-at high or low simultaneously. With decreased device geometry and increased gate density on the chip, the likelihood is greater that more than one SSA fault can occur simultaneously. It has been recommended to check  $m$ -way stuck-at faults up to  $m = 6$  [Goldstein 1970]. This is particularly true with present technology circuits because of the high device density. The number of faults increases exponentially with  $m$  as indicated in Table 2.3. A set of  $m$  lines has  $2^m$  combinations of SA faults. Since the total number of  $m$ -sets of lines in an  $N$ -line circuit is

$C(N, m) = \frac{N!}{m!(N-m)!}$ , the total number of  $m$ -way faults is  $2^m C(N, m)$ . Thus the total number of multiple

faults is  $\sum_{m=1}^N 2^m C(N, m) = 2^N - 1$ .

Table 2.3: Number of Multiple Stuck-at Faults in an  $N$ -line Circuit

Number of Nodes	Number of Faults		
	Single	Double	Triple
N	2N	4C(N,2)	8C(N,3)
10	20	180	960
100	200	19,800	1.3x10 <sup>6</sup>
1000	2,000	1,998,000	>10 <sup>9</sup>
10000	20,000	199,980,000	>10 <sup>12</sup>

In detecting multiple stuck-at faults, it is always possible to use exhaustive and pseudoexhaustive testing. However, this is not practical for large circuits. It has been shown that using a SSA fault test yields high fault coverage in detecting multiple stuck-at faults [Hughes 1984]. The most important factors that affect the detectability of multiple stuck-at faults are the number of primary outputs and the reconverging fanouts [Schertz 1971, Hughes 1984]. Comparing a multiple-output circuit such as an ALU to a parity tree, the multiple fault coverage of SAF test decreased from 99.9% to 83.33% [Mourad 1986]. The high reconverging-path nature of a parity tree causes fault masking, but the fault coverage increased to 96% when the test was increased.

## 2.6 Fault Lists

To generate tests for digital circuits, the test tools are provided with a circuit description, a netlist. The tool then creates a list of all faults to be detected. For large circuits, the list can be quite long. It is thus beneficial to minimize the list whenever possible. As we have seen, previously, some SA faults may be detected by the same test patterns. Therefore, only one of these faults needs to be included in the fault list. In the following sections we define and use the concepts of *equivalence* and *dominance* to collapse the faults and reduce the fault list.

### 2.6.1 Equivalence Relation

**Definition 1.** Two faults are called *equivalent* if every pattern that detects one of the faults also detects the other. That is, their test sets,  $T_1$  and  $T_2$ , are identical:  $T_2 \subseteq T_1$  and  $T_1 \subseteq T_2$ .

As was pointed out for the two-input AND gate, SA0 faults on the inputs and output are equivalent. All three are detected by the same test pattern. Any one of them can represent the three faults. Accordingly, the fault list is reduced from six to four faults. Similarly, SA1 for faults on the I/O of an OR gate are equivalent. For an inverter, the SA0 (SA1) on the input is equivalent to the SA1 (SA0) on the output. We can then use this knowledge about the three elementary gates, AND, OR and NOT, to develop equivalence for a circuit built of such gates.

### 2.6.2 Dominance Relation

**Definition 2.** A fault,  $f_1$ , dominates another fault,  $f_2$ , if the test set of the latter,  $T_2$ , is a subset of the test set of the former,  $T_1$ ; that is,  $T_2 \subseteq T_1$ . Any test pattern that detects  $f_2$  will also detect  $f_1$ . Therefore,  $f_2$  implies  $f_1$  and it is sufficient to include  $f_2$  in the fault list.

Going back to the case of the two-input AND gate, we have found that a test pattern for the SA1 fault on any of the inputs detects SA1 on the output.  $Z/1$  can be dropped from the fault list, which is then reduced to three faults:  $\{A/0, A/1, B/1\}$ . In general, an  $N$ -input AND gate has only  $N + 1$  unique faults. In other words, the gate has  $N + 1$  equivalence classes. This is also true for other simple logic functions, such as OR, NAND, NOR, and XOR.

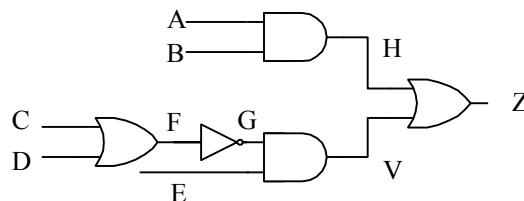


Figure 2.6 An Example of Fanout Free Circuit

### 2.6.3 Fault Collapsing

Using equivalence and dominance, we have shown how to reduce the fault list of any simple  $N$ -input gate. Next we generalize the concept to other circuits. Consider the fan-out-free circuit shown in Fig. 2.6. It has 10 lines and the fault list must have 20 faults. Applying the equivalence relation to the AND and OR gates, we get the following equivalence classes:

1.  $\{A/0, B/0, H/0\}$
2.  $\{C/1, D/1, F/1, G/0\}$
3.  $\{E/0, G/0, V/0\}$
4.  $\{H/1, V/1, Z/1\}$
5.  $\{F/0, G/1\}$

Based on the transitive property of equivalence relations, and since classes 2 and 3 include the same fault,  $G/0$ , they can be merged. All faults in the two classes are equivalent. Next, we list the dominance relations:

6.  $A/1 \rightarrow H/1$ , thus  $A/1$  can represent  $H/1$  and all its equivalent faults in class 4.
7.  $C/0 \rightarrow F/0$ , thus  $C/0$  can represent  $F/0$  and all its equivalent faults in class 5.
8.  $V/0 \rightarrow Z/0$ , but  $V/0$  belongs to equivalence class 3, which has been merged into class 2. Any fault from this class is dominated by  $Z/0$ .
9.  $B/1 \rightarrow H/1$
10.  $D/0 \rightarrow F/0$
11.  $E/1 \rightarrow V/1$

From all these equivalence and dominance relations, we can reduce the fault classes from 11 to 7. An example of the shortened list will include a fault from class 1 ( $A/0$ ); one from the merged classes 2 and 3 ( $C/1$ ), a fault from classes 4 and 5; or faults that imply all the faults in each of these classes as shown by dominance relations 6 and 7:  $A/1$  and  $C/0$ ; and finally, faults from classes 9, 10, and 11:  $B/1$ ,  $D/0$ , and  $E/1$ . In summary, a fault list is given by  $\{A/0, A/1, B/1, C/0, C/1, D/0, E/1\}$ . This list is not unique since faults  $A/0$  and  $C/1$  may be represented by other faults from the same equivalence class as representative of the class. Notice that the fault list contains only faults on the primary inputs. This is a convenient property of reconverging fan-out-free circuit [Breuer 1976]. However, most circuits are more complex. They include fan-out and also reconverging fan-out.

We will form the fault list for the 2-to-1 MUX of Fig 2.5*b*. We have learned in Section 2.5 that the faults on  $S$  dominate those of  $C$  and  $D$ . Hence, in developing the fault list, we exclude SA faults on  $S$  since they will be covered with test patterns developed for the branches  $C$  and  $D$ . Excluding  $S$  reduces the circuit to a fan-out-free circuit. It is thus sufficient to generate patterns for the primary inputs and the fanout branches. The primary inputs and the branches of all fan-outs of the circuit are called *checkpoints*. Using checkpoints is valid only for *irredundant* circuits. An irredundant circuit is an implementation of a logic function that does not contain

redundant literal or terms. Not all faults on redundant gates are detectable, as we will show next. In addition, the faults on the checkpoints as defined above do not cover all SSF in a redundant circuit. In such a case, additional patterns have to be used.

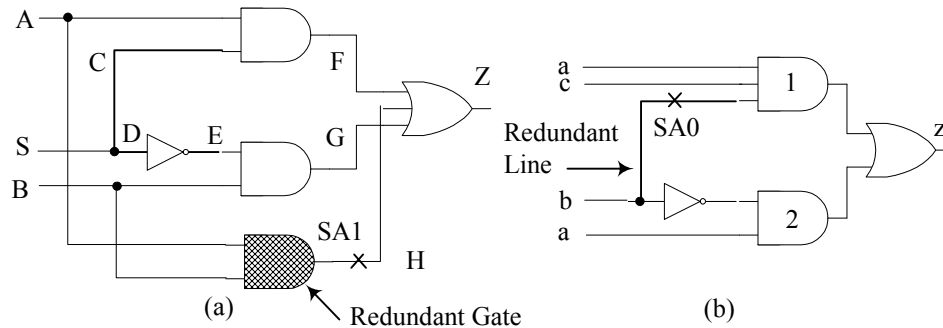


Figure 2.7 Redundant Faults: (a) A Redundant Product to Eliminate Hazard, (b) A Redundant Literal

Redundancy is sometimes included to eliminate static hazard. Other times, redundancy is introduced inadvertently as a result of revising the circuit. The 2-to-1 MUX of Fig. 2.5b has potential for static hazard as  $S$  changes from 1 to 0 while both  $A$  and  $B$  are high. This is due to the reconverging structure of the circuit:  $F(A, B, S) = AS + BS'$ . To remove the hazard, we add the term  $AB$ .  $F(A, B, S) = AS + BS' + AB$  and we obtain the circuit shown in Fig. 2.7. There are no test patterns to detect the H/0 since this requires that  $A = B = 1$ ; hence, either  $F$  or  $G$  will be equal to 1 and the fault on  $H$  cannot be sensitized to the output  $Z$ . Such a fault is called a *redundant fault*. It is simply undetectable because of redundancy in the circuit. Implementation of the function  $F(a, b, c) = ab' + abc$ , which is shown in Fig. 2.7b, also includes a redundant fault,  $b/1$ , as indicated in the figure. This function includes a *redundant literal*,  $b$  in the second product, since it may be expressed as  $F(a, b, c) = ab' + ac$ .

## 2.7 Bridging Faults

Bridging faults occur when two or more lines are shorted together and create wired logic [Mei 1974]. When the fault involves  $r$  lines with  $r \geq 2$ , it is said to be of *multiplicity*  $r$ ; otherwise, it is a *simple bridging fault*. Multiple bridging faults are more likely to occur at the primary inputs of a chip. Bridging faults are becoming more predominant because the devices are becoming smaller and the gate density higher. The total number of all possible simple

bridging faults in an  $m$ -line circuit is  $C(m, 2)$ . However, in reality most pairs of lines are not likely to be shorted. Thus the actual number is much smaller than theoretically calculated and is layout dependent.

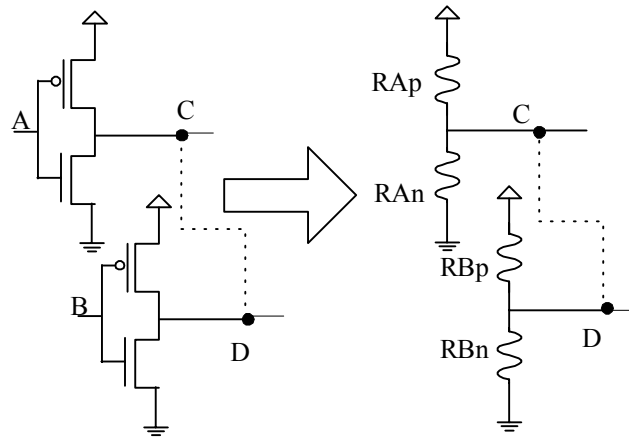


Figure 2.8 Bridging Faults Voting Model

The behavior of the circuit in the presence of bridging faults is dependent on the technology. The short between the outputs of two inverters as shown in Fig. 2.2d can be modeled as a wired logic. If the implementation is in TTL technology, it is a wired AND; in the case of ECL technology, it is a wired-OR. In the case of CMOS technology, the wired logic depends on the type of gate driving the shorted lines and the input on these gates.

We use the example in Fig. 2.8 to illustrate these effects. The two CMOS inverters are represented by their pull-up and pull-down resistances. Of course, the value of these resistances will depend on the inverters' input signals. For example, if  $A = 0$ , the NMOS transistor is off, and its corresponding resistance,  $RA_n$ , is infinite while the PMOS is on and  $RA_p$  assumes the value of the on resistance. The circuit may then be represented as a voltage divider and the value of the output will depend on the on resistance of the various transistors, as listed in Table 2.4.

If the input signals are the same for both gates, the output will be the same for both gates and the fault will not be detected. Now if  $A = 0$  and  $B = 1$ , the output will depend on  $RA_p$  and  $RB_n$ . If  $RA_p > RB_n$ , the output is 0, as if the two outputs,  $C$  and  $D$ , are wired ANDed. Following the same reasoning, one can understand the remainder of the entries in Table 2.4.

Table 2.4 Bridging Faults Models for the Circuit in Fig. 2.8

Input Condition	Relative Drive	Output Values	Wired Logic
$A=B$	any ratio	$C = D = A' = B'$	AND, OR

$A=0, B=1$	$RA_p > RB_n$	$C = D = 0$	AND
	$RA_p < RB_n$	$C = D = 1$	OR
$A=1, B=0$	$RA_n > RB_p$	$C = D = 1$	OR
	$RA_n < RB_p$	$C = D = 0$	AND

Bridging faults may cause a change in the functionality of the circuit that cannot be represented by a known fault model. An example of this type of fault is shown in Fig. 2.9, where the function of the good NMOS circuit is  $AB + CD$ , while the bridging fault changes the functionality of the gate to  $(A + C)(B + D)$ . Another important consequence of bridging faults is observed where the bridged wires are the input and the output of the same gate. Called a *feedback bridging fault*, this is illustrated on the right in Fig.2.2d. The fault transforms a combinational circuit into a sequential circuit and increases the number of states in a sequential circuit.

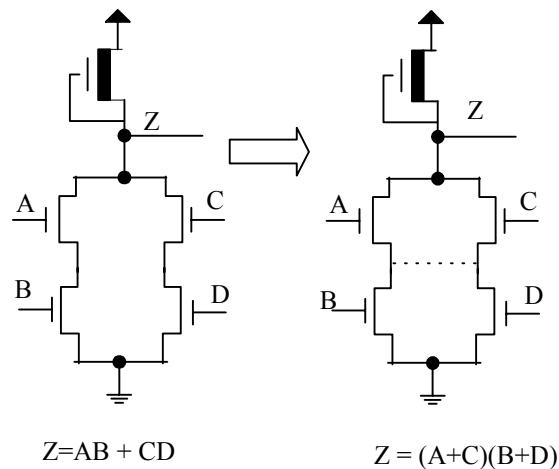


Figure 2.9 Change in Functionality Due to Bridging Faults

SSA fault test sets have been used to detect bridging faults. They yield 100% fault detection for some special circuits. The approach is to alter the order of the patterns. It is also possible to use exhaustive test sets. Detection of bridging faults by current testing are detailed in Chapter 7 [Hawkins 1994].

## 2.8 Shorts and Opens Faults

Some defects in CMOS technology defy representation by stuck-at faults [Wadsack 1978], [El-Ziq 1981, Soden 1993]. The main reason for this is that MOS combinational circuits do not necessarily remain combinational under

all faulty conditions (other than feedback bridging faults). There are several failure modes in MOS technology: (1), Transistors shorts and opens ; (2), Open-on gate, drain, or source contacts; (3), Shorts between gate and drain, or source, or channel.

Defects of the third category are primarily due to gate oxide breakdown, and they have been studied thoroughly [Syrzycki 1989], [Segura 1995]. Models for these defects are shown in Fig. 2.10 for MOS transistors with a gate of  $n$ -doped polysilicon [Segura 1995]. These *resistive shorts* are mentioned in Section 3.2. It has been proven that they can be modeled by a resistance for  $n$ -channel transistors and a resistance and a diode for the  $p$ -channel transistor. Unless the short resistance is negligible, most of these defects tend to pass undetectable; that is, the circuit behaves correctly on the functional level. However, the dynamic behavior of the circuit is affected due to the delays. These defects, however, are observable using current testing, which is the topic of Chapter 7.

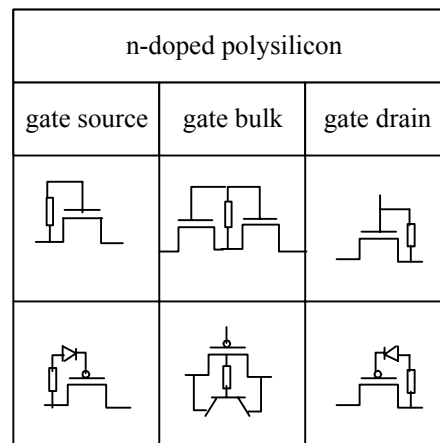


Figure 2.10 Gate Oxide Shorts [Segura 1996]

In the remainder of this section we examine the effect of shorts and opens assuming zero resistance, but we show the effect of the resistance value in one case. We consider both NMOS and CMOS technology using the inverters shown in Figs. 2.11 to 2.13. In these figures, the numbers on some of the transistor terminals refer to short or open on these terminals. Then we consider the case of a short between gate and bulk.

### 2.8.1 NMOS Circuits

Site 1 in Fig. 2.11 is an open of the pull-down to GND, site 2 is a short between the gate and source of the NMOS driver, site 3 is a short between the source and drain of the driver transistor, and so on. The logic table next to the circuit includes one column for the input,  $A$ , and another for the good circuit response on the output,  $Z$ . The remaining columns give the outputs of the faulty circuit outputs in the presence of faults at the corresponding sites.

A close observation of this table shows that for sites 1 through 4, the failure modes can easily be modeled by a stuck-at fault --  $A/0$  in the first three cases and  $A/1$  for the fourth case. The other sites need more careful examination to clarify the failure in MOS technology. The open at site 5 results in a floating gate. The trapped charge at the gate will keep the transistor on or off, depending on its charge value. However, it is probable that the charge will eventually leak and the output will become high.

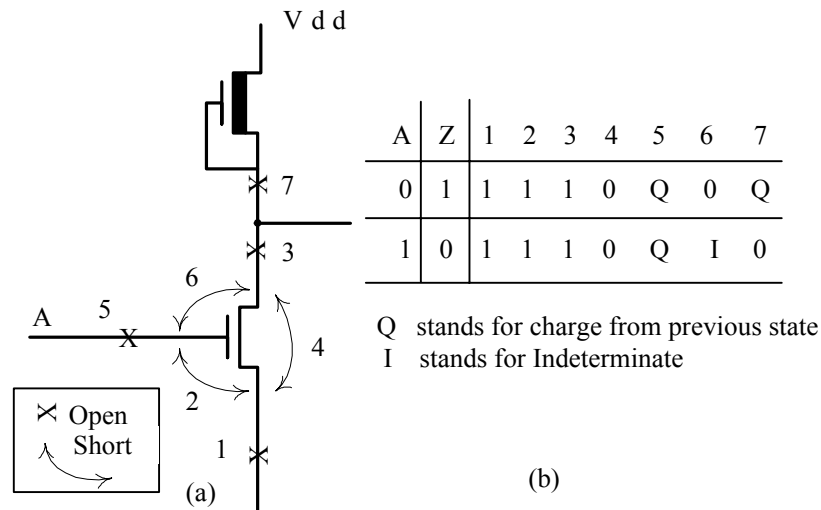


Figure 2.11 Open and Shorts in NMOS

The short between gate and drain at site 6 will tie the output to the input. When the input is low, so will be the output. However, when the input is high, the transistor will be on and will pull the output down, but the output will also be pulled up through the load transistor as well as through the gate driving the inverter under test. The output will thus be undetermined. Finally, the open at site 7 will cause the output to depend on the input signal. If the input is high,  $A = 1$ , the output will be discharged to ground ( $Z = 0$ ). For  $A = 0$ , the transistor is not conducting and will retain the previous value, which is indicated by  $Q$ . This failure mode has caused the circuit to memorize the previous state, and as such, the circuit needs first to be put in a known state ( $A = 1, Z = 0$ ), then forced to change value by making  $A = 0$  and observing the output. If the output is still 0, this indicates that the fault is detected; otherwise, the circuit is not faulty. In this case, as in the case of the simple  $SR$  latch in Chapter 1, a sequence of test patterns is needed to detect the fault.

## 2.8.2 CMOS Circuits

**2.8.2.1 Stuck-Open Faults (SOP)** In the case of CMOS circuits, open failures in pull-up and pull-down transistors cause the inverter to behave as a memory element. Detection of an open in such a circuit requires a specific test sequence, as in the case of the fault at site 7 of the NMOS inverter. For an open at sites 1 and 2 in Fig. 2.12, the sequence  $A = 0$  ( $Z = 1$ ), followed by  $A = 1$ , will still detect the fault if  $Z = 1$ ; otherwise, the circuit is not faulty. Similarly, for sites 4, 5, and 6, the needed sequence to detect these faults is  $A = 1$ ,  $A = 0$ . This type of fault, called *stuck-open* has been investigated extensively by [Banerjee 1984]. It has also been demonstrated that such faults may sometimes be avoided by proper design practice [Koeppel 1987] that makes SOP faults appear as SA faults. Test generation algorithms for MOS circuits have been recommended and are addressed in a later chapter.

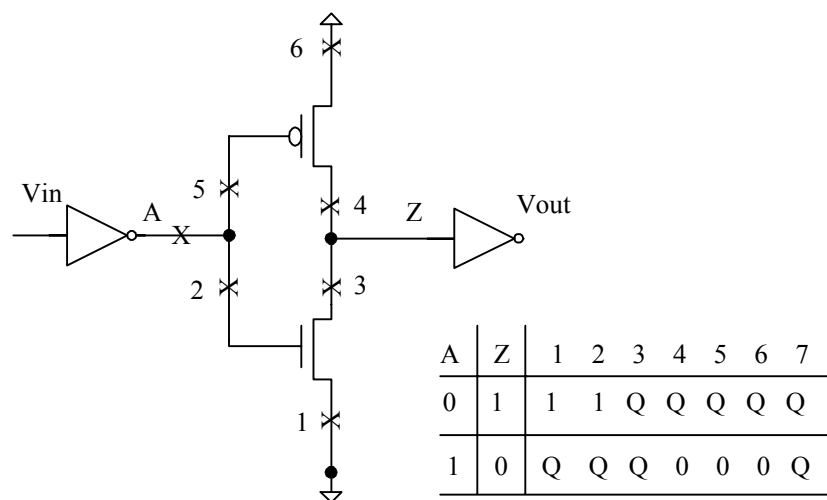


Figure 2.12 Stuck Open Faults in CMOS Circuits

### 2.8.2.2 Stuck-on Faults (SON) and Shorts

Next, we examine possible shorts as indicated in Fig. 2.13. Shorts between the source and drain of a transistor are equivalent because this transistor is on all the time. This failure mode is represented by a stuck-on fault (SON). The stuck-on fault at site 1 will keep the output pulled down to 0 and 1 can represent this failure mode as a stuck at 0 on  $Z$ . It is interesting to notice, though, that when  $A = 0$ , the PMOS transistor is on and there is a direct flow of current from  $V_{dd}$  to ground that will continue to flow as long as  $A$  is kept low. In such conditions, it is possible to observe the fault by monitoring the current, since under normal operating conditions, current flows only during switching and not in a static situation. This is known as current testing or  $I_{DDQ}$  testing and is discussed in detail in Chapter 7.

Because of the symmetry of the CMOS transistor, the SON fault on the PMOS transistor can be observed by keeping the input  $A = 1$ .

If the NMOS is stuck-on (site 1) or the gate is shorted to the source in the PMOS (site 4), the circuit behaves as if the input of the inverter is stuck at 0 and stuck at 1. Similarly, for sites 2 and 3, the inverter behaves as if its input is stuck at 0 ( $A/0$ ). Shorts at sites 5 and 6 are equivalent. They cause the circuit to behave in a fashion similar to the NMOS inverter in the presence of the short at site 7 (Fig. 2.11). The output follows the input.

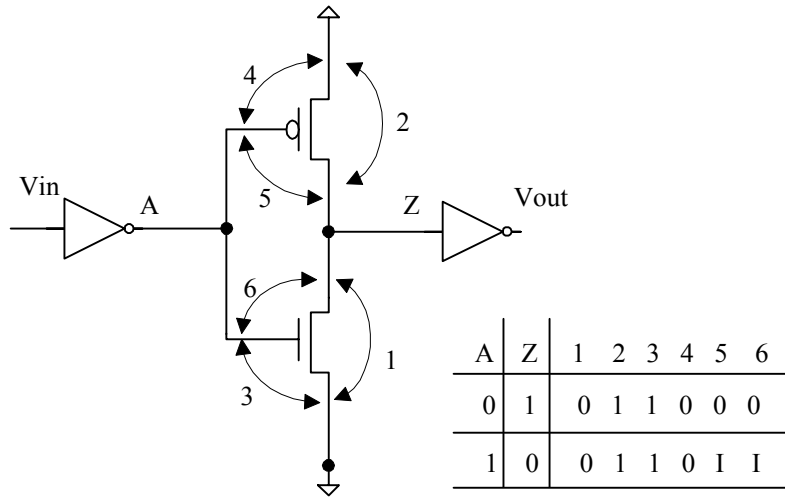
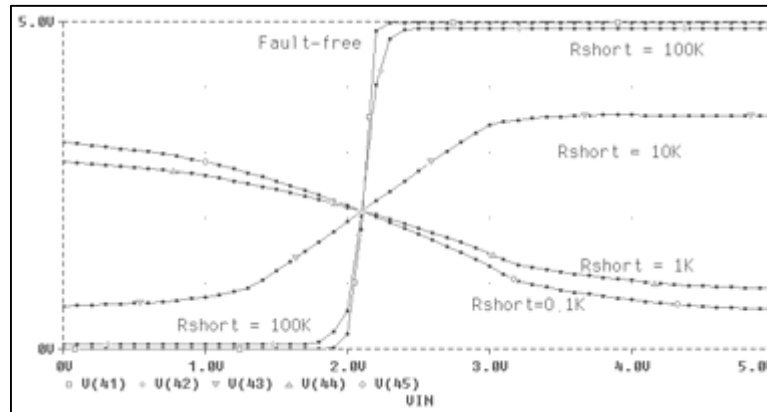
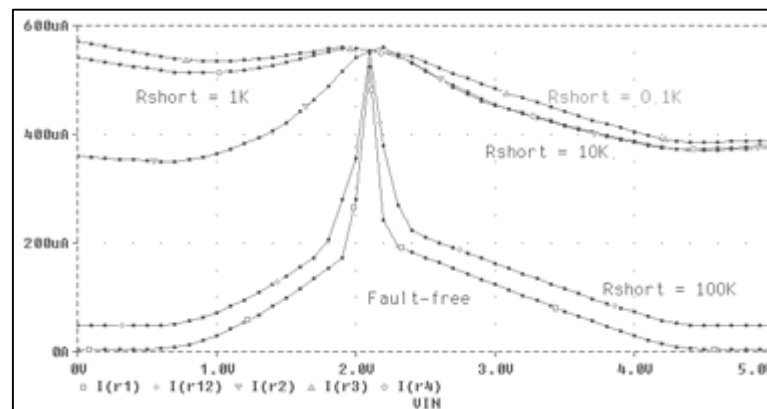


Figure 2.13 Shorts in CMOS Circuits



(a)



(b)

Figure 2.14 Spice Simulation: Effect of the Short Resistance on (a) Output Voltage, (b) Switching Current

So far in this section we have ignored the resistance of the shorts. However, we know from Section 2.2 that the short is resistive. The value of the resistance depends on the severity of the failure mechanism. The resistance varies from a few ohms to several thousand ohms. We re-consider the short between the gate and drains of the CMOS inverter in Fig. 2.13. The simulation results are shown in Fig. 2.14. The characteristic curves for different values of the short resistance,  $R_{shorts}$ , varying from 1000 K $\Omega$  to 100  $\Omega$  are shown in Fig. 2.14a. and the corresponding currents are given in Fig. 2.14b. Although for high values of the shorting resistance the output appears to be acceptable, the current is elevated and this fact will be used for current testing, which is the topic of Chapter 7.

## 2.9 Delay Faults

It is possible for a circuit to be structurally correct but to have signal paths with delays that exceed the bounds required for correct operation. In such a case, a *delay fault* is said to have occurred. Delay faults may not be

provoked if the operating frequency is low. The ultimate goal for detecting delay faults is to determine that the circuit works without malfunctions at the designed clock frequency. Thus it is appropriate to assume that the bound for correct operations should be the slack of the signal at the lead at which the fault is detected. The slack is the difference between the clock period and the longest delay path. Delay testing of a circuit determines if it contains signal paths that are too slow or too fast in propagating input transitions.

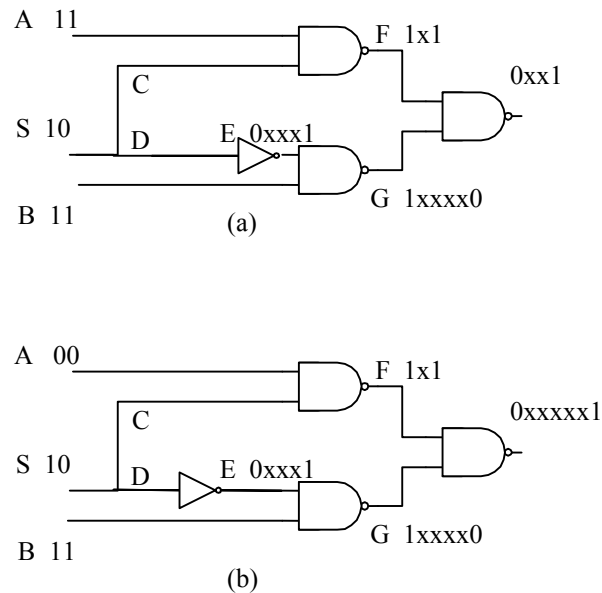


Figure 2.15 Two-Vector Test for Delay Faults

Two main models are used for delay testing. The first *gate delay fault (GDF)* is gate-oriented. The second model *path delay fault (PDF)* is path-oriented [Smith 1985]. The GDF assumes that the delay faults are lumped at the faulty gate. The delay at the output of the gate will depend on whether this signal is switching from 0 to 1 (rise) or vice versa (fall). There is a disadvantage in adopting this type of fault because it does not capture the cumulative effects from other gates, and it also ignores the delays in the interconnect wires. Also, the gate delay may cause a local delay at its output without affecting the delay of the circuit. For example, no delay is noticed at the output of the multiplexer in Fig. 2.15a when  $A = B = 1$  and  $S$  switches from 0 to 1. This is not true, however, when the switching is from 1 to 0.

The PDF model takes into account the cumulative delay from the primary input to the output. Although this model requires the consideration of too many paths, it is more realistic particularly for present technology circuits, where delays are due primarily to the interconnect wires. As the technology features decreased, gate delays

have been reduced. Meanwhile, the resistance of the interconnect wires has increased due to the reduction of their cross-sectional area.

Except for domino logic gates, delay testing of CMOS circuits consists of applying a pair of input patterns at the desired operational speed and observing the outputs for early or late transitions. Application of the pattern pairs to sensitize and propagate the fault is illustrated in Fig. 2.15b. The first test initializes the circuit, and the second pattern sensitizes the delay to the output. The circuit shows a 2-to-1 MUX that is implemented in NAND gates. The delay in the inverter is propagated to the output by the pair of patterns  $(p_1, p_2) = (011, 001)$ . The inputs signals at  $A$  and  $B$  remain the same in order to sensitize a path for any transition on  $S$  to appear on the output.

A two-pattern test is said to be a *robust delay test* for transition on a path,  $L$ , if and only if, when  $L$  is faulty and a test pair  $(p_1, p_2)$  is applied, the circuit output is different from the expected state at sampling time independently of the delays along gate [Lin 1987], [Lam 1997]. Deterministic test generation techniques for delay faults are proposed in [Hsieh 1977], [Shedletskey 1978], and [Malaiya 1984]. In addition to algorithmic delay test generation, pseudorandom testing [Wagner 1985] is also used. The latter is more suitable for a BIST environment. Simulation issues in delay testing are discussed in [Koeppel 1986]. Delay testing is becoming a very important aspect of digital testing.

## 2.10 Temporary Failures

Hard failures result in a need to change a component or repair it, causing a long mean time to repair; soft or temporary failures are, however, more frequent [Ball 1967, Siewiorek 1978]. Temporary failures are much harder to track because it is usually not possible to reproduce the fault when a component, chip, or board is tested. Hence they are not studied as thoroughly as are hard failures. They are encountered in different digital components, but are particularly important in memory chips and microprocessors. Memory defects and testing are investigated in more detail in Chapter 12.

There are two main types of temporary failures: *transient* and *intermittent* (recurring). The first type is usually due to some temporary external conditions, while the second is due to varying hardware states: parameter degradation or improper timing. Table 2.5 lists the various types of temporary failures and some of their main causes.

Table 2.5 Temporary Faults

Types	Causes
Transient	Power supply disturbances Electromigration interference Charged particles Atmospheric discharges Electrostatic discharges
Intermittent	Parameter degradation

### 2.10.1 Transient Faults

A transient fault occurs when a logic signal has its value temporarily altered by noise signals, and the resulting signal may be interpreted incorrectly by the rest of the circuit [McCluskey 1986]. Such a fault is difficult to diagnose and correct. It is thus important to minimize the noise in the circuit and increase the circuit's noise immunity. Transient faults may be caused by fluctuations in the power supply, metastability, or cosmic radiation.

#### 2.10.1.1 Power Supply Disturbances

Power supply disturbances are known to cause errors in the operation of digital systems. Experiments were run to measure circuit susceptibility to power supply disturbances by measuring the change in the outputs of the circuit whose inputs are kept at constant signals. These experiments related the disturbances to the noise immunity of the circuits. Later experimentation was carried out on circuits in different technologies: CMOS gate array, CMOS breadboard and bipolar breadboard. Here the experiment was carried out under more realistic assumptions: that the logic signal changes with time. It was found that the susceptibility of the circuits to power supply voltage disturbance is related to the operating frequencies. Errors are more likely to occur as the operation frequency increases. Also, propagation delay variation is the dominant effect, and noise immunity plays a smaller role in error occurrences. Failures due to power supply disturbance can be modeled by delay faults.

#### 2.10.1.2 Metastability.

Another form of a transient fault is caused by metastability in latches and flip-flops. Metastability occurs when a latch is given only enough energy to switch its state halfway to another stable state, and when the latch enters a critical stable, or metastable state, which exists somewhere between the two stable states. The latch will remain in this metastable state for some indeterminate amount of time but will eventually leave it for one of the stable states. No method of eliminating all metastability is known. It is important, however, to determine to a certain degree the

probability of metastability occurrences. For this, metastability sensors have been developed [Stucki 1979]. The mean time between metastability (MTBM) failures can be predicted in terms of latch parameters [Chaney 1979].

### 2.10.1.3 Radiation Induced Failures

Depending on the sensitive area in an electronic device, a high-energy particle such as alpha particles or cosmic rays may have different effects [Sharma 1997]:

- Single-event upset (SEU) is the ionizing radiation-induced logic change in the cell of a storage device such as a RAM cell. This may cause hard or soft failure [May 1979, Berger 1985]. That is, the logic reversal may be permanent or temporary.
- Transient errors are dependent on the ionizing dose rate rather than on the total amount of radiation delivered.
- Single-event latch-up is caused by a latched low-impedance state by turning on the parasitic *pnp* and *npn* in the bulk of the CMOS circuit.

There is a general consensus that radiation hardening and proper packaging of integrated circuits are sufficient to decrease the occurrence of radiation-induced transient faults [Dieh 1982, Messenger 1992]. However, this approach is not effective in presenting the effect of alpha particles emanating from the impurities in the metal used in the fabrication.

### 2.10.2 Intermittent Faults

Intermittent failures are recognized to be an important cause of field failures in computer systems. Very little is known about the failure mechanisms because spontaneous intermittent failures are difficult to observe and control. However, artificially induced intermittent failures can easily be produced, controlled, and observed. Several papers [Koren 1977, Varshney 1979], [Savir 1980] have addressed the problem of testing for intermittent failures. The intermittent fault models presented in these papers assume signal-independent faults. This assumption has turned out to be inappropriate after experimental evidence of pattern-sensitive intermittent failures.

Pattern sensitivity was first encountered in memory testing [Hayes 1980a,b] and microprocessors [Hackmeister 1975]. It is possible to explain pattern sensitivity in a microprocessor as being due to the embedded large RAMs. These failures are better described by delay faults than by pattern sensitivity. It is reasonable to

conjecture that different instruction streams exercise different portions of the chip, and failures are caused by delay faults due to supply voltage reductions as described in [Cortes 1986].

An attempt to collect data on intermittent failures on Sperry--Univac computers was reported by [O'Neill 1980]. Hard failures that were believed to have appeared earlier as intermittent failures were analyzed. Instead, two classes of failure mechanisms were responsible for this type of failure: (1) metal-related open and short circuits; and (2) violations of operating margins. The latter class of failures has been investigated further by [Cortes 1986]. In his study, catalog parts are forced into intermittent faulty behavior by stressing supply voltage, temperature, and loading. The temperature stress changes the voltage transfer characteristics. The voltage stress affects the noise immunity. The loading stress reduces the driving capability. All stresses used in the experiments have some impact on the logic interfacing between two gates, thereby causing the circuit under stress to exhibit behavior similar to that of a marginal circuit under normal operating conditions [McCluskey 1987].

## 2.11 Noise Failures

IC failures so far have been attributed mainly to process and manufacturing defects. Except for single-event transient errors, it has been implied that the circuits are noise immune. Also, attention was given only to the gates forming the circuit independently of their interconnects. Changes in technology have resulted in digital ICs with the following characteristics:

- The smaller devices are capable of switching faster, thus allowing an increase to the operating frequency. The slope of the ramp function mentioned above is not longer but is much smaller than the device dimensions.
- The higher device density enables the designer is able to make more complex and larger circuits.
- The power supply voltage was decreased from 5 V to 3.3 V and, very soon, to less than 2 V. Thus the circuit is becoming less immune to noise.
- The use of thinner and longer interconnect wires, resulting in a smaller cross section, has caused an increase in the resistance. Because of the fine pitch, fringing capacitances have been added to capacitance to ground. Also, mutual capacitance between interconnects on the same layer and on adjacent layers cannot be neglected.

As a result of these changes, the interconnect can no longer be modeled as a lumped model but as a distributed network or, not in the far future, as a transmission line. Also, as the number of switching devices has

increased, the switching current has exaggerated the resistance of the substrate and the inductance of the ground and power pins. These new physical characteristics of very deep submicron technology together with the increased operating frequencies have resulted in compromising signal integrity. The most important types of failures that are due to noise are *crosstalk* and *simultaneous switching noise (SSN)*, which is more commonly known as *ground bounce*. The first phenomenon is due to coupling capacitance between long wires and can be minimized by careful routing which will be discussed in Chapter 4. The other phenomenon is due to the package pin's inductance. It has usually been tamed by shorting the power and ground rails with appropriate capacitances. Efforts in characterizing the effects of these phenomena are ongoing [Haydt 1994, Breuer 1996, Rubio 1996].

## References

- Bakoglu, H.B., (1990), *Circuits, Interconnections and Packaging*, Addison-Wesley, Reading, MA.
- Ball, M.O. and F. Hardie, (1967), "Effects and detection of intermittent failures in digital systems," *IBM. 67-825-2137*.
- Banerjee, P. and J.A. Abraham, (1984), "Characterization and testing of physical failures in MOS logic circuits," *IEEE Design & Test of Computers*, Vol. 1 No. 8, pp. 76-86.
- Berger, E.R. et al., (1985) "Single Event upset in Microelectronics: Third cosmic ray Upset Experiment," *IBM Tech.Directions, Federal Systems Division*, Vol. 11 No. 1, pp. 33-40.
- Black, J.R., (1969), "Electromigration – A brief survey on some recent results," *IEEE Trans. on Electron Devices*, Vol. ED-16 No. 2, p. 388.
- Breuer, M. A. and Friedman, (1976), *Diagnosis and Reliable Design of Digital Systems*, Computer Science Press.
- Breuer, M.A. and S.K. Gupta, (1996) "Process aggravated Noise: New validation and test problems," *Proc. Int'l Test Conf.*, pp. 914-923.
- Chakravarty, S. and Y. Gong., (1995), "Voting Model based diagnosis of bridging faults in combinational circuit.," *IEEE Int'l Conf. on VLSI Design*, pp. 338-342.
- Chaney, T.J. and F.U. Rosenberger, (1979), "Characterization and scaling of MOS Flip-Flop Performance in Synchronizer Applications," *Proc. CALTECH Conf. on VLSI*, pp. 357-374.
- Cortes, M.L. et al., (1986), "Properties of Transient Errors due to Power Supply Disturbances," *IEEE Int'l Symposium of Circuits and Systems*, pp. 1046-1049.
- D&T, (1998), *IEEE Design and Test of Computers*, Vol. 15 No. 4.
- Dieh, S.E. et al., (1982), "Error Analysis and Prevention of Cosmic Ion-Induced soft Errors in Static CMOS-RAMs," *IEEE Trans. on Nuclear Science*, Vol. NS-30 No. 6, pp. 2032-2039.
- El-Ziq, Y.M., (1981), "Functional-level test generation for stuck-open faults in CMOS VLSI," *Proc. Int'l Test Conf.*, pp. 536-546

- Ferguson, F.J. and J.P. Shen, (1988), "Extraction and simulation of realistic CMOS faults using inductive fault analysis," *Proc. Int'l Test Conf.*, pp. 475-484.
- Goldstein, H., (1970) "A probabilistic analysis of multiple faults in LSI circuits," *IEEE Computer Society Repository R-77*, Long Beach, CA.
- Hackmeister, D. and A.C.L. Chiang, (1975), "Microprocessor Test Technique Reveals Instruction Pattern - Sensitivity," *Computer Design*, No. 12, pp. 81-85.
- Hawkins, C.F. et al., (1994), "Defect classes -- an overdue paradigm for CMOS IC Testing," *Proc. Int'l Test Conf.*, pp. 413-425.
- Hawkins, C.F. and J.M. Soden, (1986), "Reliability of electrical properties of gate oxide shorts in CMOS ICs," *Proc. Int'l Test Conf.*, pp. 443-451.
- Haydt, M.S., R. Owens, and S. Mourad, (1994) "Modeling and Characterization of Ground Bounce," *Proc. Int'l Test Conf.*, pp. xx-xx.
- Hayes, J.P., (1980a) "Detection of Pattern-Sensitive Faults in Random -Access Memories," *IEEE Trans. on Computers*, Vol. C-29 No.3, pp. 713-719.
- Hayes, J.P., (1980b), "Testing memory for Single-Cell Pattern-Sensitive Fault," *IEEE Trans. on Computers*, Vol. C-29 No. 3, pp. 713-719.
- Henderson, C. et al., (1997), "Yield and failure analysis challenges in IC manufacturing," *Future Fab Int'l*, Vol. 1 No. 2, pp. 335 - 343
- Ho, P.S., (1982), "Basic problems for electromigration in VLSI," *Proc. Of 20<sup>th</sup> Annual Int'l Reliability Physics Symp*, p. 288.
- Hsieh, E.P. et al., (1977), "Delay test generation," *Proc. 14<sup>th</sup> ACM-IEEE Design Automation Conf.*, pp. 486-491.
- Hughes, J.L.A. and E.J. McCluskey, (1984), "Multiple stuck-at fault coverage of single stuck-at fault test sets," *Proc. Int'l Test Conf.*, pp. 52-58.
- ITC, (1997), *Proc. Int'l Test Conf.*
- Koeppe, S., (1986), "Modeling and simulation of delay faults in CMOS stuck-open logic circuits," *Proc. Int'l Test Conf.*, pp. 530-536.
- Koeppe, S., (1987), "Optimum layout to avoid CMOS stuck-open faults," *Proc. 24<sup>th</sup> ACM-IEEE Design Automation Conf.*, pp. 829-835.
- Koren, I. and Z. Kohavi, (1977), "Diagnosis of Intermittent Faults in Combinational Networks," *IEEE Trans. on Computers*, Vol. C-26 No. 11, pp. 1154-1157.
- Lam, W.K. et al., (1997) "Delay fault covering," *IEEE Trans. on CAD*, Vol. 16 No. 1, pp. 32-44.
- Lin, C.J., and S.M. Reddy, (1987), "On delay fault testing in logic circuits," *IEEE Trans. on CAD*, Vol. CAD-6 No. 9, pp. 694-703.
- Malaiya, Y.K. and R. Narayanaswamy, (1984), "Modeling and testing for timing faults in synchronous sequential circuits," *IEEE Design & Test of Computers*, Vol. 1, No. 4, pp. 62-74.
- Maly, W., F.J. Ferguson, and J.P. Shen, (1984), "Systematic characterization of physical defects for fault analysis of MOS IC cells," *Proc. Int'l Test Conf.*, pp. 237-245.

- May, T.C. and M.H. Wood, (1979), "Alpha-Particles-Induced Soft Errors in Dynamic Memories," *IEEE Trans. on Electron Devices*, Vol. ED-26 No.1, pp. 2-7.
- McCluskey, E.J., (1986), *Principles of Logic Design*, Prentice Hall, Upper Saddle River, NJ.
- McCluskey, E.J. and S. Mourad, (1987), "Comparing Causes of IC Failures," *Development in Integrated Circuit Testing*, D.M. Miller Ed., pp.13-46.
- Mei, K., (1974), "Bridging and stuck-at faults," *IEEE Trans on Computers*, Vol. C-23 No. 7, pp. 720-727.
- Merchant, P.P., (1982), "Electromigration: an overview (VLSI metalization)," *Hewlett-Packard Journal*, Vol. 33 No. 8, p. 28.
- Messenger, G.C. and M. Ash, (1992), *The Effects of Radiation on Electronic Systems*, Van Nostrand Reinhold.
- Mourad S., and E.J. McCluskey, (1987), "Fault Models" in *Testing and Diagnosis of VLSI and ULSI*, Eds F. Lombardi and M. Sami Kluwer, pp. 49 -58.
- Mourad, S., et al., (1986), "Effectiveness of single stuck-at fault tests to detect multiple faults in parity trees," *Comput. Math. Applications*, Vol 13, No. 5/6, pp. 455-459.
- O'Neill, E.J. and J.R. Halverson, (1980), "Study of intermittent field hardware failure data in digital electronics," *NASA Contractor Report #159269*.
- Rubio, A. et al., (1996), "An approach to the analysis and detection of crosstalk faults in digital circuits," *IEEE Trans. on Computers*, Vol. 36, No. 3, pp. 387-394.
- Sachdev, M. (1998), *Defects Oriented Testing for CMOS analog and digital circuits*, Kluwer Publishers.
- Savir, J., (1980), "Testing for Single Intermittent Failures in Combinational Circuits by Maximizing the Probability of Fault Detection," *IEEE Trans. Comput.*, Vol. C- 29, No. 5, May, pp. 410-416.
- Schertz, D. R., and G. Metze, (1971), "On the design of multiple fault diagnosable networks," *IEEE Trans. on Comput.*, Vol. C-20, No.11, pp.1361-1364.
- Schugraf, K.F. and C. Hu, (1994), "Reliability of thin SiO<sub>2</sub>," *Proceedings of IEEE*, vol. 9, pp. 989-1004, September.
- Segura, J. et al., (1995), "A Detailed Analysis of GOS Defects in MOS Transistors: Testing Implications at Circuit Level," *Proc. of Int'l Test Conference*, pp. 544-551.
- Segura, J. et al., (1996), "A detailed analysis of electrical modeling of gate oxide shorts in MOS transistors," *Journal of Electronic Testing, Theory and applications*, Vol. 8, No. 3, pp. 229-240.
- Sharma, A.K., (1997), *Semiconductor Memories: Technology, Testing, and Reliability*. IEEE Press, Piscataway, NJ.
- Shedletskey, J., (1978), "Delay testing LSI logic," *Proc., 8<sup>th</sup> Int'l Fault Tolerance Computing Symposium*, June 1978, pp. 410-416.
- Shen, J.P., W. Maly and F. J. Ferguson, (1985), "Inductive fault analysis of MOS integrated circuits," *Design and Test of Computers*, Vol. 2, No. 6, pp. 13-26.
- Siewiorek, D.P., (1978), "A case study of C.mmp, Cm\* and Cvmp," *Proc. of IEEE*, Vol. 2, No. 6, pp. 1178-1220.
- Smith, G.L., (1985), "Models for delay faults based on path," *Proc. Int'l Test Conference*, Philadelphia, PA, Oct., pp. 342-349

Soden, J.M. and C.F. Hawkins, (1993), "Correct CMOS defect models for quality testing," NASA, (1993), *NASA Symposium on VLSI Design*, Albuquerque, NM.

Strapper, C., A. McLaren, and M. Dreckmann, (1980), "Yield model for productivity optimization of VLSI memory chips with redundancy and partially good product", *IBM Journal of Research and Development*, Vol. 24, No. 3, pp. 398-409, May.

Strapper, C., (1976), "LSI yield modeling and process monitoring," *IBM Journal of Research and Development*, Vol. 20, No. 3, pp. 228-234, May.

Stucki, M.J., and J.R., Cox Jr., (1979), "Synchronization Strategies," *Proc. CALTECH Conf. On VLSI*, pp. 375-393.

Szyzki, M., (1989), "Modeling of Gate Oxide Shorts in MOS Transistors," *IEEE Trans. on CAD*, Vol. 8, No. 3, pp. 193-202, March.

Sze, S., (1983), *VLSI Technology*, McGraw-Hill, New York.

Varshney, P.K., (1979), "On analytical modeling of Intermittent faults in digital systems," *IEEE Trans. on Comput.*, Vol. C-28, No. 10, Oct., pp. 786-791.

Wagner, K.D., (1985), "The error latency of delay faults in combinational and sequential circuits." *Proc. Int'l Test Conference*, Philadelphia, PA, Oct., pp. 334-341.

Wadsack, R.L., (1978) "Fault modeling and logic simulation of CMOS and MOS integrated circuits," *Bell System Technical Journal*, Vol. 57, No. 5, May-June 1978, pp. 1449-1474.

Walker, D. M. H., (1987), *Yield simulation for integrated circuits*, Kluwer Academic Publishers, Boston.

Woods, M.H., (1984), "The implications of scaling on VLSI reliability," *Proc. Of 22<sup>nd</sup> Annual Int'l Reliability Physics Symp.*, p. 288, 1984.

## Problems

1. The two patterns (000) and (111) toggle all nodes of a 3-input NAND gate. Are they sufficient to detect all stuck-at faults in the gate? Explain why or why not.
2. Prove that the patterns that detect the SSF on the primary inputs will detect all SSF in a fanout free combinational circuit with single output.
3. Show that the implementation of the function,  $F(a,b,c) = ab' + abc$ , has a redundant fault.
4. Determine the checkpoints of each of the two circuits in Fig. P2.4.
5. Use the equivalence and dominance properties to reduce the faults on the checkpoints determined in Problem 4.
6. For the circuit in Fig. 2Pa, generate test patterns to detect the bridging faults among the primary inputs. Model this fault by a wired AND.

7. Determine test sequences that detect stuck-open on the gates of all transistors of the CMOS circuits implementing: a) a 2-input NAND gate, b) a 2-input NOR gate, c) the gate representing  $((ab)' + c)'$ .
8. Consider the circuit in Fig. 2.13. Assume that there is a resistive short between the gate and drain, Rshort. Use a SPICE like simulator to examine the effect of the load on the driving inverter (the first inverter). That is, add a capacitive load,  $C_L$ , and plot the output voltage versus  $C_L$ .

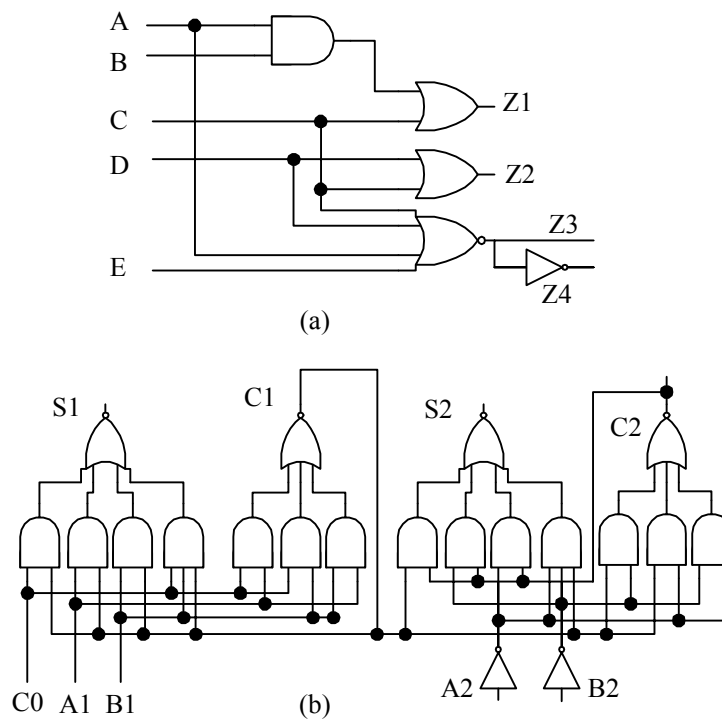


Figure P2.4