15-8
44

# PROBABILITY OF LATCHING SINGLE EVENT UPSET ERRORS IN VLSI CIRCUITS

by

Kenneth Chris Holland

Thesis submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of
Master of Science
in
Electrical Engineering

**APPROVED:**

Dr. Joseph G. Tront

Dr. Richard W. Conners

Dr. Charles E. Nunnally

February, 1991
Blacksburg, Virginia

**PROBABILITY OF LATCHING SINGLE EVENT UPSET ERRORS IN VLSI CIRCUITS**

by

Kenneth Chris Holland

Dr. Joseph G. Tront, Chairman

Electrical Engineering

**(ABSTRACT)**

The ability of radiation to cause transient faults in space borne as well as ground based computers is well known. With the density of VLSI circuits increasing every year, the probability of an upset by radiation is becoming more likely. However, research in this area has matured over the last decade, and the mechanisms which cause such faults are better understood. This understanding enables us to propose ideas to eliminate or lessen the effects of radiation on VLSI circuits.

Most of the research to date has concentrated on the effect of transient faults on flip-flops rather than combinational logic. This is due to several reasons. First, transient faults, also known as Single Event Upsets (SEU), were first observed in memory circuits located on board satellites. Second, an SEU can leave a lasting effect on a circuit if it occurs in a flip-flop, and third, SEUs can cause the output of a flip-flop to change state more easily if it occurs directly

in the flip-flop rather than in the combinational logic.

In combinational logic, the node struck by the radiation is completely disjoint from the flip-flops output node. This in effect causes the SEU to satisfy more criteria in order to change the flip-flops output state. The criteria that the SEU must satisfy tend to be complex, and this complexity has caused many researchers to believe that SEUs that occur in combinational logic cause negligible errors in the state of flip-flops.

Thus, in this thesis, the criteria for latching a SEU are discussed, and original methods are presented that can be used to determine the probability of an SEU occurring at any node in a circuit will cause a change in the output state of a flip-flop. The methods are then incorporated into a program, named SUPER II, that is able to evaluate the circuit to determine the nodes with the highest probability of having a SEU error latched. The results from the program show that SEUs that occur in combinational logic can have a significant probability of becoming latched.

## Acknowledgements

I would like to first thank my loving wife for going back to college with me and making a long time dream a reality. I would like to also thank her for her unending devotion and patience with me over the last two years, and for giving birth to our seven month old daughter, Rachael. My love also goes out to my mother and father for encouraging me and believing in me through out my life.

I would like to also thank Dr. Tront for introducing me to the field of VLSI circuit design and for all of his help, advice, and encouragement when I needed it. I sincerely hope that we may work together again in the future. Also, my gratitude goes out to Dr. Nunnally and Dr. Conners for serving on my committee and for their friendship over the last two years.

# TABLE OF CONTENTS

# List Of Tables

# List Of Illustrations

# 1.0 INTRODUCTION

## 1.1 INTRODUCTION

In the mid-1970s scientists and researchers first observed strange errors occurring in satellite electronics. These errors occurred as changes in the output state of memory devices, such as flip-flops and registers, and were only temporary in nature. After the device changed state, the memory device did not have any permanent damage and functioned normally. Transient faults, such as this, became known as **single event upsets**(SEU).

Since the first observance of SEUs, a lot of research has been performed to determine the mechanisms that cause them. Models have been developed by several researchers including Pickel and Blandford[1] and Sivo[2] to predict error rates due to cosmic rays, and each model has been shown to predict error rates in accord with the observed rates.

Experiments have also been performed by such agencies as NASA, JPL, and NRL, and other researchers such as Pickel and Blandford [1], Guenzer et al.[3], and Blake and Mandel [4].

Those experiments usually included irradiating different types of memory circuits with different ions to determine the error rate for the device.

Other researchers have found techniques that can reduce the effects of cosmic rays. These techniques include shielding the circuit, putting the circuit in a low equatorial orbit, using proven radiation hardened technologies, or even applying fault-tolerant design methods.

However, most of the above research has been performed on flip-flops in which a charged particle strikes a small capacitance stored on the off transistor drain in the flip-flop. A charge build up then occurs that can be seen as a voltage spike which might then travel through the flip-flop causing the output state to change permanently.

SEUs that occur in combinational logic, however, have not been studied as much. If a transient occurs on a node in a combinational logic circuit, it must travel to the input of a flip-flop in order to become latched. This implies that a critical path must be set up for the transient fault from the struck node to the flip-flop. The transient must also arrive at the input of a flip-flop with the active edge of the clock in order to cause the flip-flop to change state. Once the

state of the flip-flop is changed, the error might propagate throughout the system causing improper system operation.

In this thesis, an attempt is made at determining a comprehensive solution to finding the probability that an SEU that occurs in combinational logic will cause an error at the output of a flip-flop. Two methods are presented (probability building and probability propagation) which are capable of finding the critical path probability. Another approach is presented for finding the probability that the transient fault will arrive at the same time as the active edge of the clock. Also, variations in the amplitude and pulse width of the transient pulse as it propagates down the critical path are accounted for.

The above methods are incorporated into a program, called SUPER II, that can evaluate each node in a logic circuit and find the probability that an SEU occurring at each node will cause an error at the output of a flip-flop. A designer might then be able to make design changes to the circuit to decrease the error probability. Those changes might include increasing the number of inputs to a gate along the critical path, or possibly decreasing the amount of fan-out encountered by the transient pulse as it propagates down the critical path.

## 1.2 SUMMARY OF CHAPTERS

Chapter 2 provides an explanation of the basic upset mechanism for MOS transistors. Several mechanisms exist, but only upsets due to a single ionized particle striking a sensitive volume are considered in this research.

An explanation of how SEUs occur in CMOS combinational logic, and the requirements necessary for an SEU to cause an error at the output of a flip-flop are introduced in Chapter 3.

In Chapter 4, the methods necessary to find the probability that the SEU will cause an error at the output of a flip-flop are presented. Those methods include two original methods named probability building and probability propagation. The two methods are examined to determine their advantages and disadvantages, and a rough timing analysis is also provided for each method.

An overview of SUPER II and how it might be used is given in Chapter 5. The EDIF netlist format is discussed, and test results are presented for several ISCAS type circuits.

Chapter 6 contains a summary of the research and suggestions are given for future work in this area.

# 2.0 THE BASIC SEU MECHANISM

## 2.1 ORIGIN OF COSMIC RAYS

All of the elements that make up cosmic radiation have been created in stars. Giant clouds of gas, called nebulae, are the beginnings of a star, and they consist mainly of hydrogen and helium.

The process begins when gravity causes the gas clouds to contract. High temperatures begin to exert an outward pressure that slows down the effect of gravity. Eventually the nebula is compact enough and has a high enough temperature to radiate energy in the visible light spectrum. Nuclear processes also begin in the core of the star. These processes convert the hydrogen and helium into the other heavier elements observed in cosmic rays [5].

These heavier elements explode into space at high speeds and energies when the star supernovas. The particles then encounter other gas clouds that tend to decrease the particle's energy by approximately 300-400 MeVs/u where u is an atomic mass unit. These particles travel for 10 million

years before they reach our solar system [6].

When these particles enter our solar system they encounter the sun's solar wind which is a collection of particles consisting mainly of alpha particles (Helium nuclei). Collisions with the solar wind causes the energy of the incoming atoms to be decreased once again, and also causes many of the ions to scatter away from the earth. Therefore, the flux and energy of particles at or near earth is highly dependent on the solar wind, which goes through maximum and minimum phases every 11 years.

Solar maximum is the period of time when the sun has the greatest sun spot and solar flare activity. Because of the strong winds during solar maximum few cosmic rays make the final voyage to earth, and the rest are swept away from the solar system. However, during times of solar minimum cosmic rays are more freely able to enter the solar system [7].

**2.2 THE GEOMAGNETIC CUTOFF**

When ionized atoms reach the earth, they must pass through the earth's magnetic field that prohibits some particles from entering and allows others to pass through effortlessly. This effect is known as the Geomagnetic Cutoff.

The *Geomagnetic Cutoff* refers to the magnetic rigidity level that a particle must overcome in order to pass through the earth's magnetic field effortlessly. The *penetrating ability* or *magnetic rigidity* is determined by the particles momentum divided by its charge. The further this ratio becomes below the geomagnetic cutoff, then the shallower the particle will penetrate the earths magnetic field.

In order for a particle to pass through the earth's magnetic field it must cross magnetic field lines, and the deeper that the particle penetrates the more magnetic field lines that will be crossed. Thus, a particle requires more energy the deeper that it travels in order to overcome the magnetic field. The polar regions and the outer magnetosphere tend to be easier for the particles to penetrate because of the lower rigidities needed [7].

Therefore, the placement of an electronic device within the magnetosphere determines the amount of radiation that the device might receive. For example, if a device is put in a low equatorial orbit, it has a lower incidence of cosmic rays because of the larger number of magnetic field lines that must be crossed by the particle. On the other hand, if a device is in a polar region it has a higher incidence of cosmic rays because of the fewer magnetic field lines that the particle

must cross.

## 2.3 THE COMPOSITION OF COSMIC RAYS

Cosmic radiation consists of particles of matter covering the full spectrum of elements found in the universe, and they posses a full range of energies given in units of MeVs. The make-up of cosmic rays is similar to the universal composition of matter shown in Table 1, but many of these elements are very rare and tend not to cause problems in space electronics. Given this, only the first 28 elements are considered in studying SEUs.

In section 2.1 the particle flux was mentioned for elements entering the solar system. All elements that exist have a flux verses energy spectrum characteristic curve like the one shown in Figure 1 for iron. As the energy of a particle given in MeV/u, increases, the flux decreases at a nonlinear rate. Here u denotes the atomic mass unit of an element, and the flux is given in units of the number of particles/($m^2$-steradian-sec-MeV/u).

For most applications, however, the **Linear Energy Transfer** (LET) spectrum shown in Figure 2 is more useful. The LET is also known as the **stopping power** of a particle [7]. This

**Table 1. The universal composition of matter.**

| Universal Composition of matter | |
|---|---|
| Hydrogen | 78% |
| Helium | 20% |
| Oxygen | 0.8% |
| Iron | 0.04% |
| Other | 1.16% |

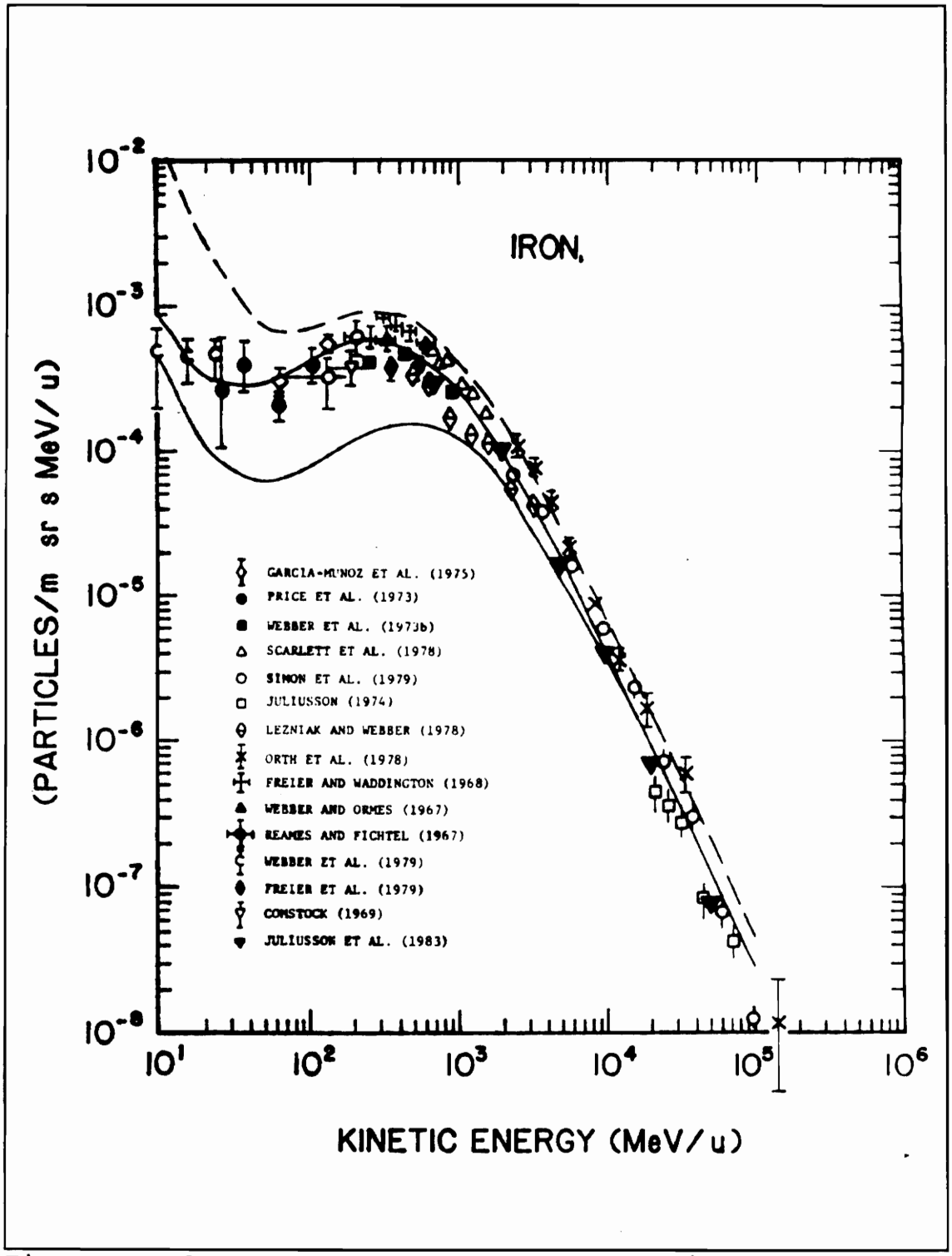



Figure 1. Flux versus energy spectrum for iron.

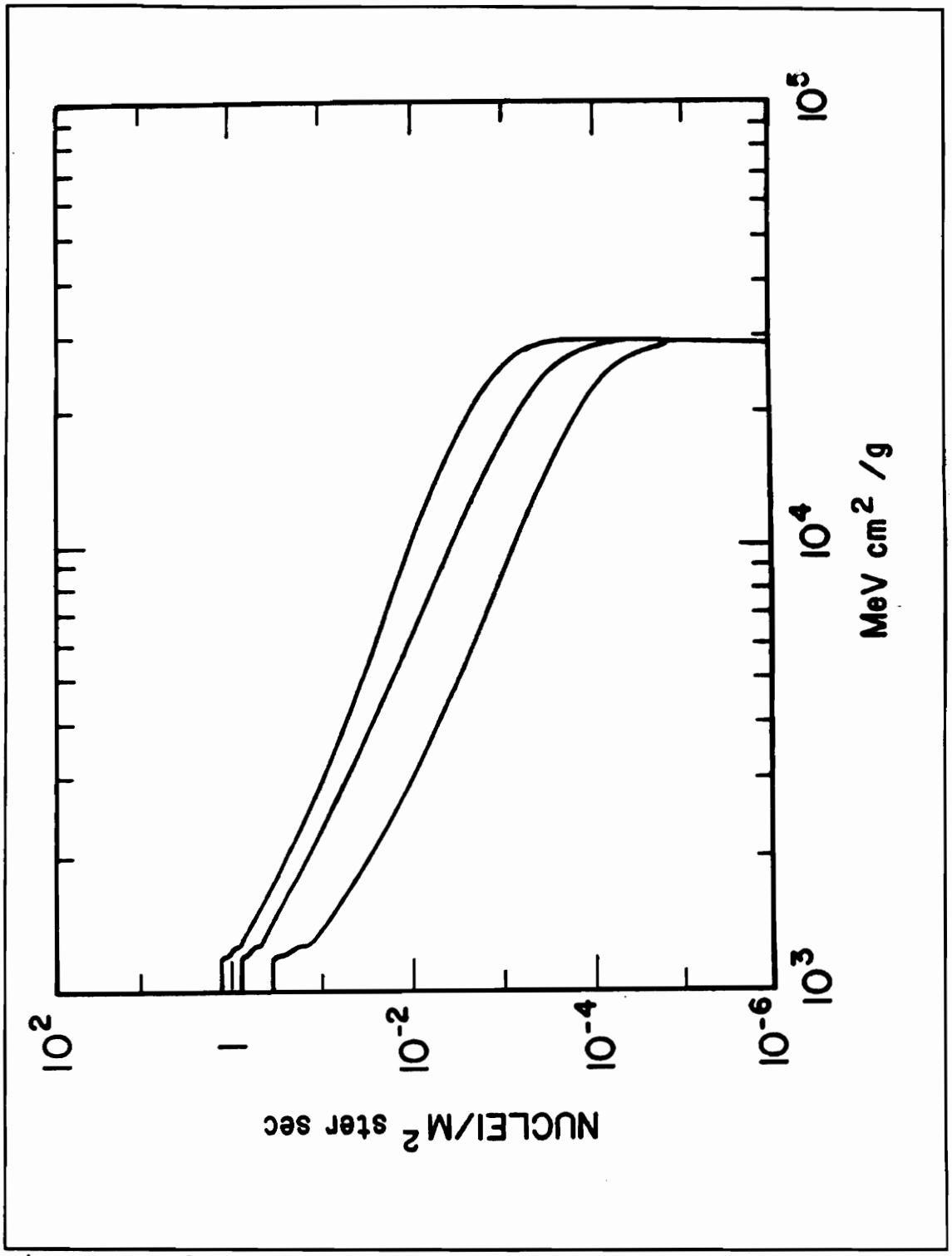



Figure 2. Flux versus LET spectrum.

spectrum is derived from the energy spectrum for each particle and is given as the amount of energy deposited per unit path length, dE/dx (MeV/micrometer) [7]. The LET is dependent on the type of material being penetrated, the energy of the particle, and also the path length that the particle traverses in the material. Because the LET is dependent on the material being penetrated, some researchers divide the LET by the density of the material to give units of MeV-micron$^2$/gm.

### 2.4 BASIC MECHANISM

SEUs caused by cosmic radiation are attributable to an ionized particle striking the small capacitance on the drain of an off MOS transistor. The off transistor drain has a reverse biased p-n junction which in turn has a depletion region. When the particle penetrates the depletion region or *sensitive volume*, it travels in approximately a straight line path and releases energy as it does so. The area of the sensitive volume is determined to be the width and length of the drain, and the height is taken to be the width of the depletion region. Some distance is also added to the width and length of the drain to account for the depletion region on the sides of the drain's p-n junction.

Generally, large amounts of energy are released when the

particle traverses the sensitive volume. When the amount of energy released is compared to the path length that the particle travels through the silicon, most researchers consider that the energy released per unit distance is constant [1,2,8].

As the particle penetrates the sensitive volume the energy released by the particle causes electron-hole pairs to be created as shown in Figure 3. The minority carriers are swept toward the bulk voltage, and the majority carriers are swept into the drain. A quick charge build up results on the drains junction capacitance, usually on the order of few picoseconds, causing a net voltage change on the drain node. This change can be seen as a transient pulse as shown in Figure 4.

From the previous discussion, questions might be asked on how much energy is needed to cause an upset, and what is its dependency on the capacitance of the sensitive volume?

### 2.5 THE ENERGY RELEASED

In order to find the energy released by a particle when it strikes a sensitive volume, the path length traveled by the particle in the sensitive volume needs to be found. Since the particle might enter the sensitive volume from any angle, as shown in Figure 5, the average path length can be used for a

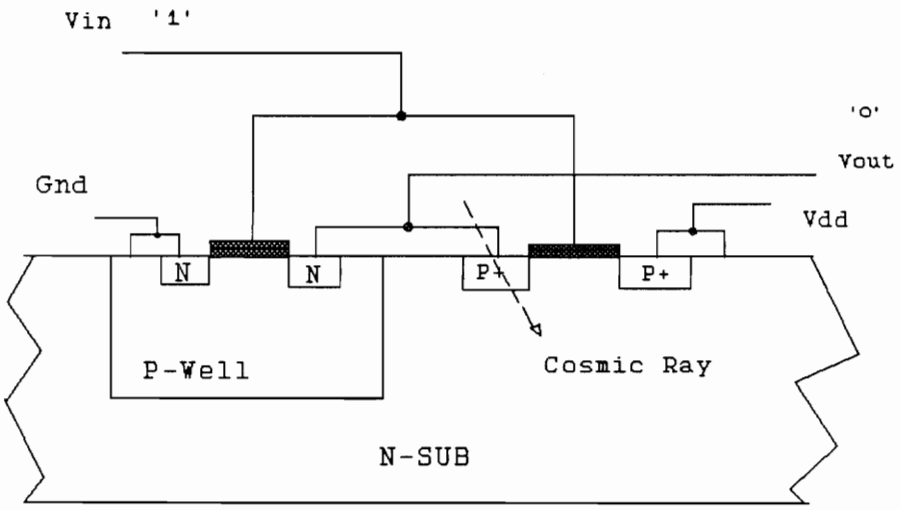



Figure 3. The particle stikes the off transistor drain.

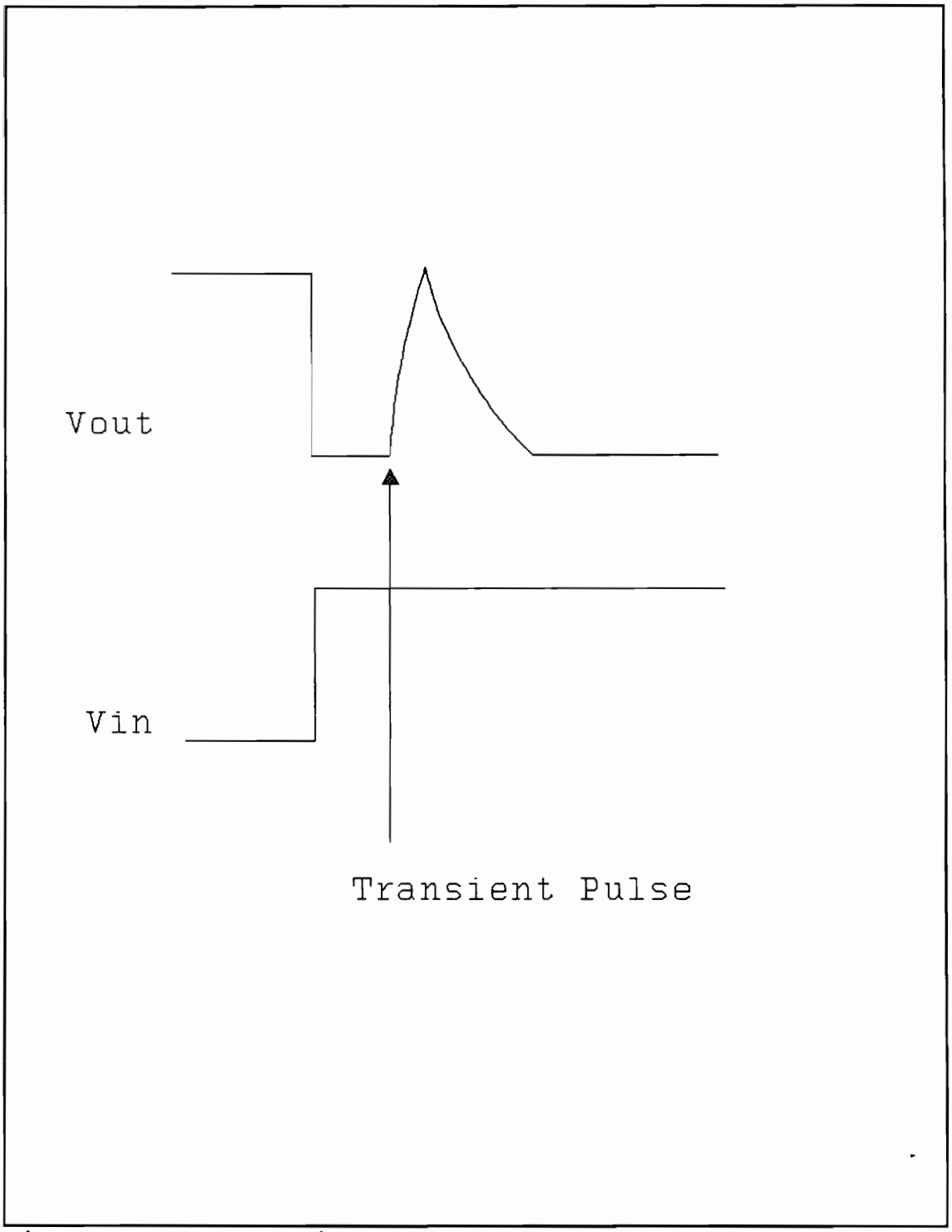



**Figure 4. The transient pulse as seen on the output of an inverter.**

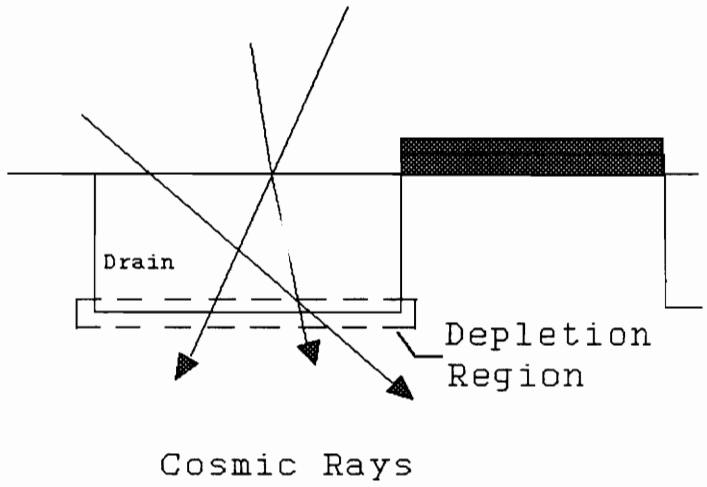



Figure 5. Particle can enter from any angle.

typical analysis or for a worst case analysis the longest path length through the sensitive volume can be used [1,2,8].

The average path length (S) through the sensitive volume can be found by using the formula:

$$S = V/A$$

where V = wlh and A = (lw+lh+wh)/2, and the longest path length (s) can be found by:

$$s = (l^2+w^2+h^2)^{1/2}.$$

Thus the total energy deposited in the sensitive volume is equal to:

$$E = S*LET \text{ or } s*LET$$

depending on the type of analysis chosen.

**2.6 THE CHARGE AND VOLTAGE**

Once the energy released by the particle is known, the resulting charge produced by the particle can be calculated by:

$$Q(\text{picocoulombs}) = E(\text{MeV}) \; / \; 22.5$$

where 22.5 is a given conversion factor in units of MeV/pC. If the capacitance at the node is known, then the resulting voltage amplitude can be found by using:

$$V = Q \; / \; C.$$

## 2.7 FINDING THE MINIMUM LET

As noted previously, SEUs in memory devices have been researched extensively, so the following introductory discussion will begin with a static Random Access Memory (RAM) cell. Shown in Figure 6 is a static RAM cell that might be struck by a particle at the off transistor drain nodes P2 or N1.

If, for example, P2 is struck, the energy released by the particle creates electron-hole pairs which are separated by the electric field around the drains depletion region. The electrons flow to the bulk potential, and the holes flow into the drain which charges up the drain's junction capacitance. This charge is seen as a positive-going voltage pulse on the input to the next inverter in the flip-flop. If the charge is large enough the next inverter in series will change it's output state which in turn will propagate back to the originally struck inverter. Therefore, if the transient can propagate through the cross-coupled inverters fast enough, the output of the struck inverter will remain in the wrong output state.

Since the transistors, P1 and N1, in Figure 6 have a threshold voltage, there exists some minimum voltage ($V_{min}$) necessary for

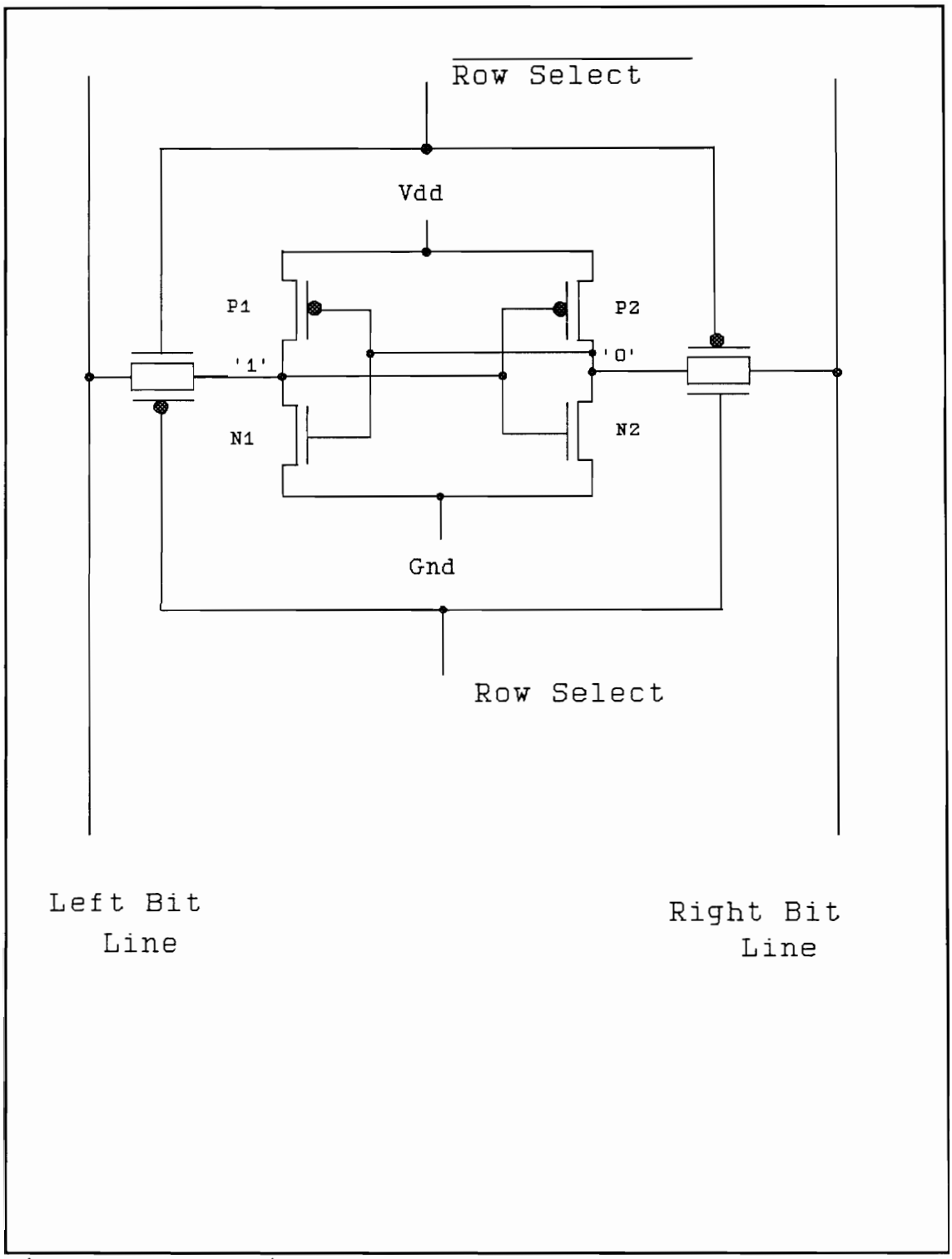



**Figure 6. A static CMOS RAM cell.**

the flip-flop to change states. If the output of P2 is at 0 volts, then the output voltage might have to rise to at least VDD, the supply voltage, minus $V_{tp}$, the threshold voltage of P1, in order to turn on N1 and to turn off P1. Thus, the minimum charge or critical charge needed to cause an upset is equal to:

$$Q_{min} = (VDD-V_{tp})*C = V_{min}*C$$

where C is the drain's junction capacitance. Thus, the minimum energy needed is:

$$E_{min} = 22.5*Q_{min} = 22.5*V_{min}*C.$$

Therefore, there exists a minimum LET that is needed by a particle before it can cause an upset which is given by:

$$LET_{min} = E_{min}/s = (22.5*V_{min}*C)/s.$$

Note that $LET_{min}$ is dependent on $V_{min}$ and the junction capacitance of the sensitive volume, and also on the path length of the particle through the sensitive volume. Thus, if the junction capacitance and/or $V_{min}$ are kept large, then the circuit will be less vulnerable to upsets. Also, if the path length through the sensitive volume is kept small, the circuit will be less vulnerable.

**2.7 EXAMPLE**

Suppose a static CMOS flip-flop is designed so that VDD = 5

volts, and the threshold voltages of the transistors are $V_{tn}$ = 1 volt and $V_{tp}$ = -1 volt. For todays technology, a drain's junction capacitance for a 4x4x1 (wlh) micron sensitive volume will be approximately 11 Ff. If $LET_{min}$ is to be found, then a worst case analysis can be performed using the maximum path length traveled by the particle through the sensitive volume, as shown in Figure 7.

$LET_{min}$ is thus:

$$LET_{min} = (22.5*V_{min}*C)/s$$

$$= ((22.5 \text{ MeV/pC})*(4 \text{ volts})*(.011 \text{ pF}))/(5.74 \text{ micrometers})$$

$$= .172 \text{ MeV/micron.}$$

Therefore, the particle incident on the drain must have a LET larger than .172 MeV/micron in order to cause an upset. That value is on the very low end of the typical LET spectrum, thus indicating that many particles might be capable of causing an upset in this circuit.

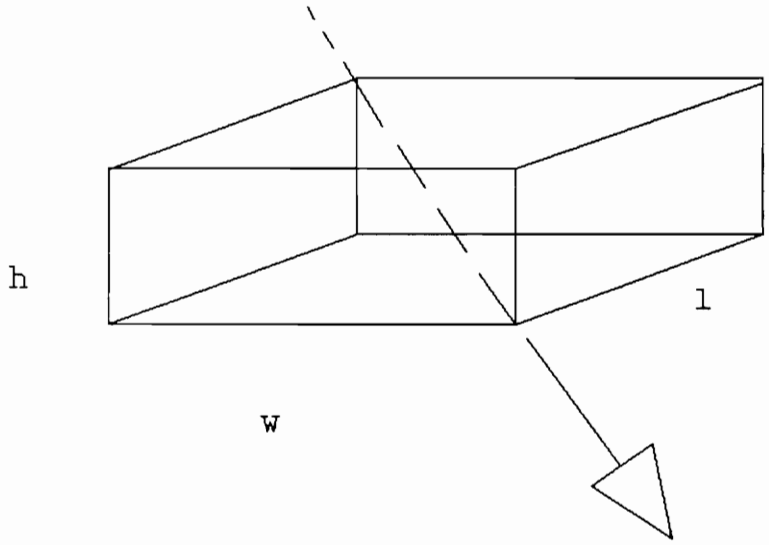



Figure 7. The longest path through the sensitive volume.

# 3.0 SEUs IN COMBINATIONAL LOGIC

## 3.1 INTRODUCTION

Research over the last decade has centered on SEUs that occur in circuits containing memory devices such as RAMs and microprocessors. This is because SEUs leave their lasting effect in flip-flops, and because the resulting state changes at a flip-flops output can occur much more easily if the SEU occurs in the flip-flop as apposed to the SEU occurring in the combinational logic.

SEUs that occur in combinational logic, however, have not been studied very much. This is because an SEU that occurs in combinational logic must become latched before it can have a lasting effect on the overall circuit, and there are four basic criteria that determine whether an SEU will become latched:

1. The incident particle must strike a sensitive volume and have enough energy to cause a voltage spike of sufficient amplitude and pulse width to propagate through the logic to the input of a flip-

flop.

2. The circuit must have a critical path set up from the struck node to the input of a flip-flop.

3. The transient pulse must have sufficient amplitude and pulse width at the input to the flip-flop to cause the flip-flop to change to an erroneous state.

4. The transient pulse must also arrive at the proper time with respect to the clock in order to cause an error in the output state of the flip-flop.

From the previous criteria, it seems unlikely that SEUs occurring in combinational logic could cause errors in the state of a flip-flop, but they do. Once the upset is latched, it might propagate through out the system causing even more problems with the overall systems operation.

To understand the above criteria, each one will be examined in detail. In Chapter 2 the mechanism leading to a transient pulse on the output of a static RAM cell was discussed, and the same mechanism occurs in logic gates. Therefore, the

first criterion will only be discussed lightly in this chapter. The last three criteria, however, will be introduced in this chapter and discussed in more detail in Chapter 4.

### 3.2 SEUs IN LOGIC GATES

The first criterion concerns the SEU mechanism as it occurs in combination logic. Since the NAND, NOR, and inverter gates are the basic logic gates used in popular gate-array circuit designs, each gate will be discussed next.

#### 3.2.1 NAND GATE

The transistor level diagram of a typical CMOS two-input NAND gate is shown in Figure 8. When a logic one appears on both inputs of the NAND gate, the two n-type transistors are turned on, and the two p-type transistors are turned off. This causes the output to be at the ground potential.

If a particle strikes node 2 of the p-type transistors, electron-hole pairs are created in the junction of the drain's depletion region or sensitive volume. The electrons and holes are then separated by the electric field around the depletion region. The holes are swept into the drain giving a net positive charge that can be seen as a positive-going voltage spike on the output of the gate. Thus, the area that is sensitive to a strike is equal to the sensitive volume of one

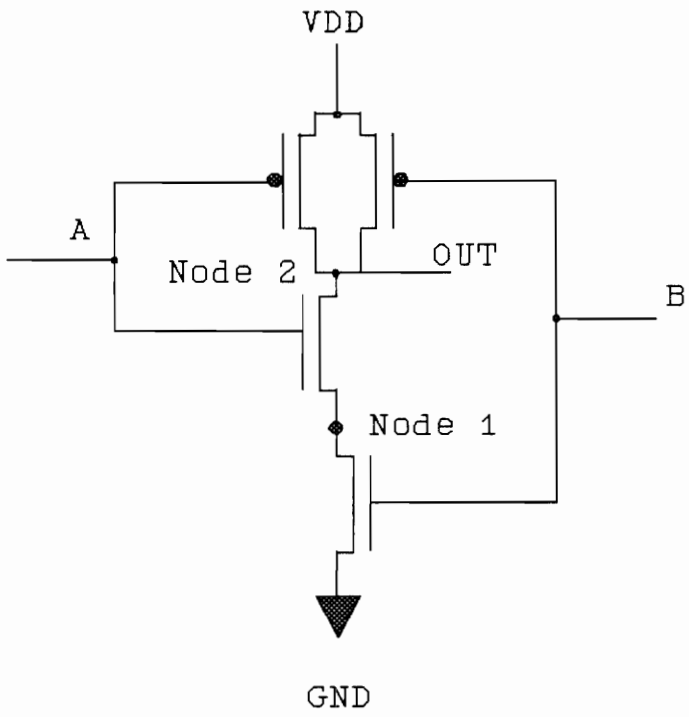



Figure 8. A typical transistor level diagram of a two-input CMOS NAND gate.

the p-type transistors.

Next, the B input is set to a logic one and the A input is set to a logic zero. This causes the n-type and p-type transistors connected to input B to be turned on which causes the output to be at the VDD potential.

If a particle strikes node 2, of the n-type transistors, the holes created in the sensitive volume are swept to the bulk voltage, and the electrons are swept into the drain. This gives a net negative charge on the node that can be seen as a negative-going voltage spike on the output of the gate. The total drain area that is sensitive to a strike is thus equal to the sensitive volume of one of the n-type transistors.

Similar discussions are possible for the other two logic combinations, AB = 10 and AB = 00. The main point is that the sensitive volume is different depending on the inputs applied to the gate. From the last chapter, the capacitance of the sensitive volume and the path length traversed by the particle through that volume are important to the amplitude calculation of the transient pulse.

### 3.2.2 NOR GATE

The transistor level diagram for a typical two-input CMOS NOR

gate is shown in Figure 9 and will be used for this discussion. When a logic zero appears on both inputs of the NOR gate, the two p-type transistors are turned on, and the two n-type transistors are turned off. This causes the output to be at the source or VDD potential.

If a particle strikes node 2 of the n-type transistors, the holes created are quickly swept to the bulk ground potential, and the electrons are swept into the n-type drain giving a net negative charge that can be seen as a negative-going voltage spike on the output of the gate. Thus, the area that is sensitive to a strike is equal to the sensitive volume of one of the n-type transistor drains.

Similar discussions are possible for the other three input logic combinations, AB = 10, AB = 01, and AB = 11. Again, the main point is that the sensitive volume varies according to the inputs applied to the gate.

### 3.2.3 INVERTER

Examining Figure 10, the CMOS inverter might have either the p-type or the n-type transistor turned off depending on which input value is applied. This suggests that either one of the sensitive volumes in the inverter might be sensitive to a particle strike depending on which input is applied.

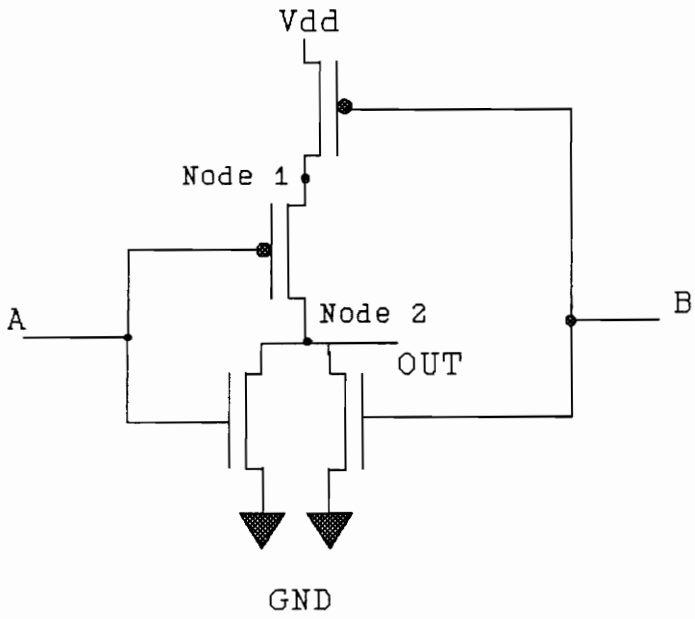



Figure 9. A typical transistor level diagram of a two-input CMOS NOR gate.

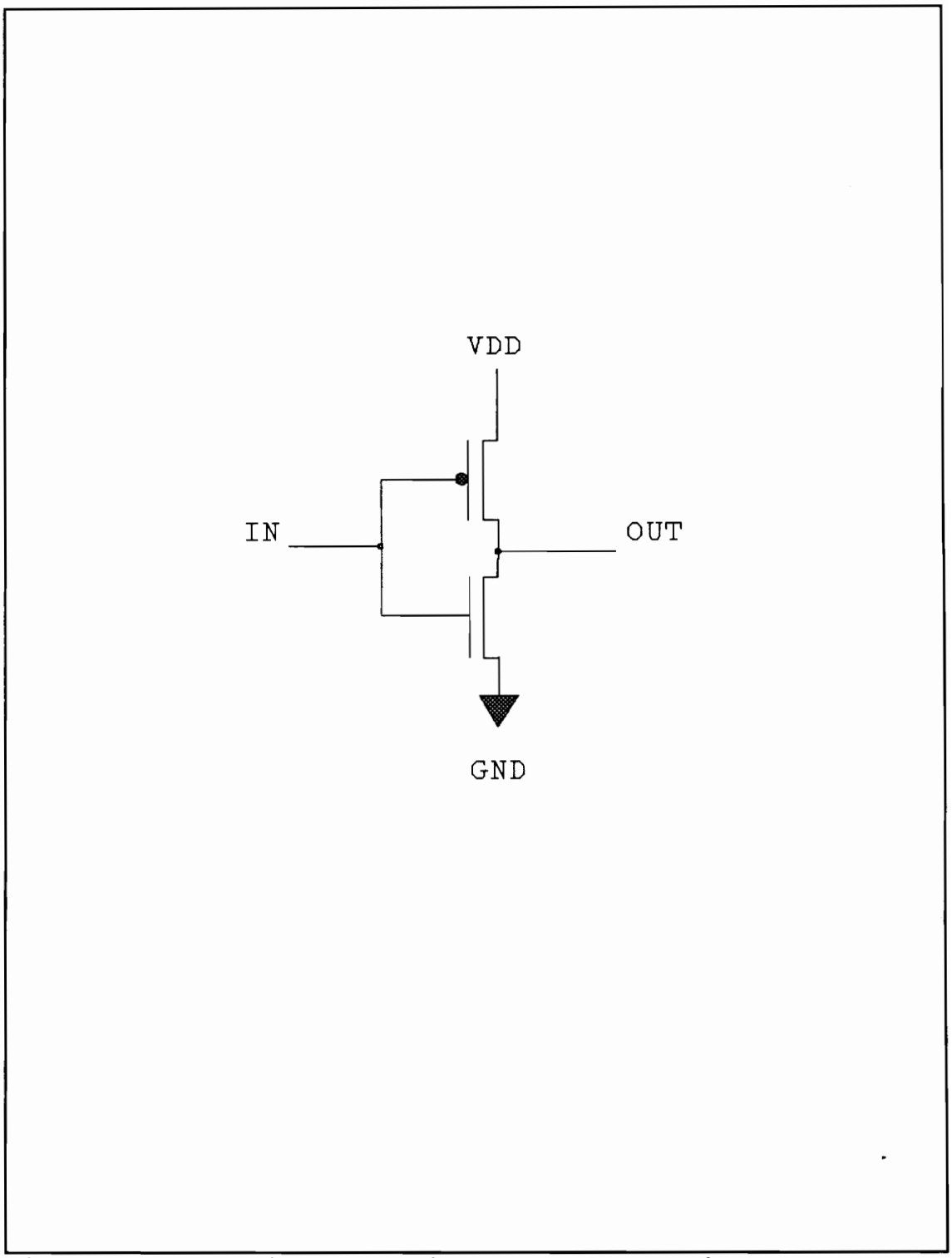



Figure 10. A typical transistor level CMOS inverter.

## 3.3 THE CRITICAL PATH

### 3.3.1 INTRODUCTION

Once an SEU occurs in a combinational logic circuit the upset must propagate to the input of a flip-flop. This means that the upset might have to pass through several gates before reaching a flip-flop. Each gate along the path must, therefore, have the proper logic values set up on their inputs in order to allow the faulty signal to pass through. In the case of NAND gates, all of the inputs other than the faulty input must be at a logic one value, and for NOR gates all of the inputs except the faulty input must be set to a logic zero value. When a path is setup in this manner, it is known as a *critical path*. A better understanding of the critical path can be realized if an example is used as shown in Figure 11.

First, a transient pulse occurs at the output of gate 1 as shown in Figure 11. The NAND gates along the path to the flip-flop must have a logic one value on their inputs, and the NOR gates must have a logic zero value on their inputs. If those non-faulty input logic values are then traced towards the primary inputs, several primary input patterns can be constructed that will allow the critical path to be set up. In Figure 11, the patterns xx 1 00 11, xx 1 01 11, and xx 1 10 11 set up the critical path where the x denotes that a don't care condition exists for that primary input.

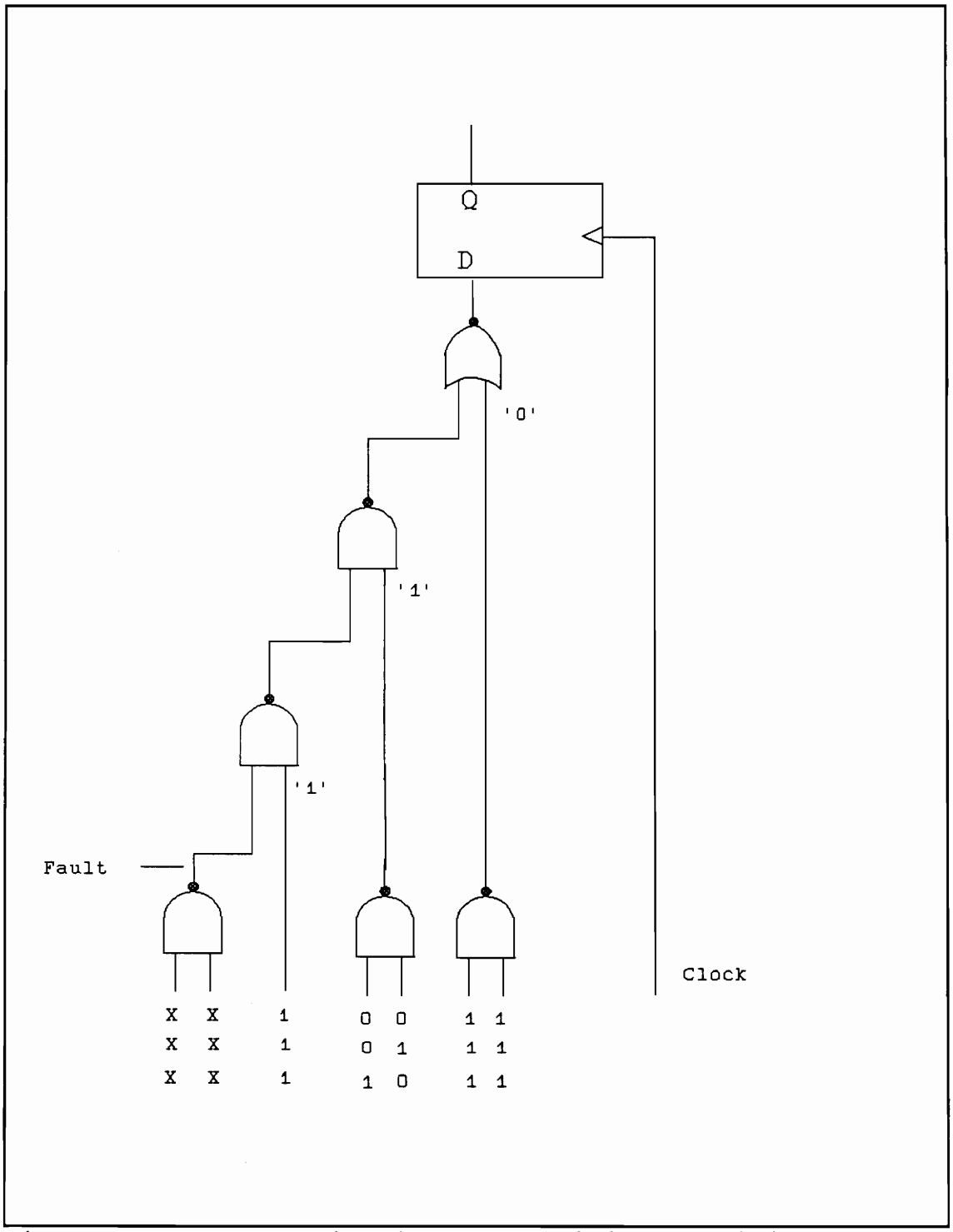



Figure 11. Example circuit for examining a critical path.

Given those input combinations, 12 out of the 128 possible primary input patterns can set up the critical path. If an assumption is made that any input combination is equally likely to be applied to a circuit when a transient upset occurs, the probability of setting up the critical path for the circuit discussed above is 12/128 or 3/32. However, if the critical path probability is needed for several paths from the struck node or if several nodes in the circuit need to have the critical path probability calculated, this method would consume a lot of time. Another method, however, can be used to find the critical path probability with less calculations but is only appropriate for circuits with cascaded gates. This method will be discussed next.

### 3.3.2 CRITICAL PATH PROBABILITY

The truth tables of the individual gates can be used to find the probability that a transient pulse will propagate down a critical path to the input of a flip-flop. In this section the truth tables will be examined for the NOR and the NAND gate, and an example will be worked that illustrates the use of the method.

The following is an explanation of the truth table method as it applies to the two-input NAND gate, and the truth table for the gate is shown in Table 2. The A' input in the table is

Table 2. Truth table for a two-input NAND gate.

| A' | A | B | Y | Y' |
|---|---|---|---|---|
| 1 | 0 | 0 | 1 | 1 |
| 1 | 0 | 1 | 1 | 0 |
| 0 | 1 | 0 | 1 | 1 |
| 0 | 1 | 1 | 0 | 1 |

Table 3. Truth table for a three-input NAND gate.

| A' | A | B | C | Y | Y' |
|---|---|---|---|---|---|
| 1 | 0 | 0 | 0 | 1 | 1 |
| 1 | 0 | 0 | 1 | 1 | 1 |
| 1 | 0 | 1 | 0 | 1 | 1 |
| 1 | 0 | 1 | 1 | 1 | 0 |
| 0 | 1 | 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 1 | 1 | 1 |
| 0 | 1 | 1 | 0 | 1 | 1 |
| 0 | 1 | 1 | 1 | 0 | 1 |

the logic state of the A input when a transient occurs on it. The table shows that whenever the B input is set to a logic one value any change in the A input causes a change in the output of the gate. So for a two-input NAND gate, the output might change for two out of the four possible combinations, or a 1/2 probability exists that the transient pulse will propagate through the gate.

For the three-input NAND gate, shown in Table 3, the output will change for two out of the eight possible combinations when a fault occurs on input A. This corresponds to a 1/4 probability that the transient pulse will propagate through the gate. Therefore, the probability that the output will change state given that a transient pulse occurs on the input of general NAND gate is $1/2^{m-1}$, where m is the number of inputs to the gate.

For a two-input NOR gate, as shown in Table 4, the output logic value changes whenever the B input is set to a logic zero value. This corresponds to two out of the four possible combinations allowing the fault to propagate through the gate, or a 1/2 probability exists that the transient pulse will propagate through the gate. Also, four out of the eight possible combinations can propagate the transient pulse

Table 4. Truth table for a two-input NOR gate.

| A' | A | B | Y | Y' |
|---|---|---|---|---|
| 1 | 0 | 0 | 1 | 0 |
| 1 | 0 | 1 | 0 | 0 |
| 0 | 1 | 0 | 0 | 1 |
| 0 | 1 | 1 | 0 | 0 |

Table 5. Truth table for a three-input NOR gate.

| A' | A | B | C | Y | Y' |
|---|---|---|---|---|---|
| 1 | 0 | 0 | 0 | 1 | 0 |
| 1 | 0 | 0 | 1 | 0 | 0 |
| 1 | 0 | 1 | 0 | 0 | 0 |
| 1 | 0 | 1 | 1 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 |
| 0 | 1 | 0 | 1 | 0 | 0 |
| 0 | 1 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 | 0 |

through the three-input NOR gate in Table 5, or a 1/4 probability exists that the transient pulse will propagate through the gate. Again the probability that a transient pulse occurring on one of the inputs to a general NOR gate will propagate through the gate is $1/2^{m-1}$.

### 3.3.3 EXAMPLE

Shown in Figure 12 are four gates cascaded ending with a flip-flop. If a SEU occurs at the output of gate 1, then the probability that the output of gate 2 will change is 1/2. The upset must then pass through gate 3 which has two inputs. From the equation above, this corresponds to a 1/2 probability that the output will change. So together, there is a (1/2)*(1/2) = 1/4 chance that the upset will propagate through both gates. Next, the fault must pass through a two-input NOR gate which has a 1/2 chance of changing its output, and that corresponds to a (1/2)*(1/2)*(1/2) or 1/8 probability that the upset will propagate through all three gates to the flip-flops input.

However, as noted before, this analysis is good for cascaded gates, but is not very good for general circuits. The internal logic values of a circuit might have higher or lower probabilities of occurring than the 1/2 probability inferred

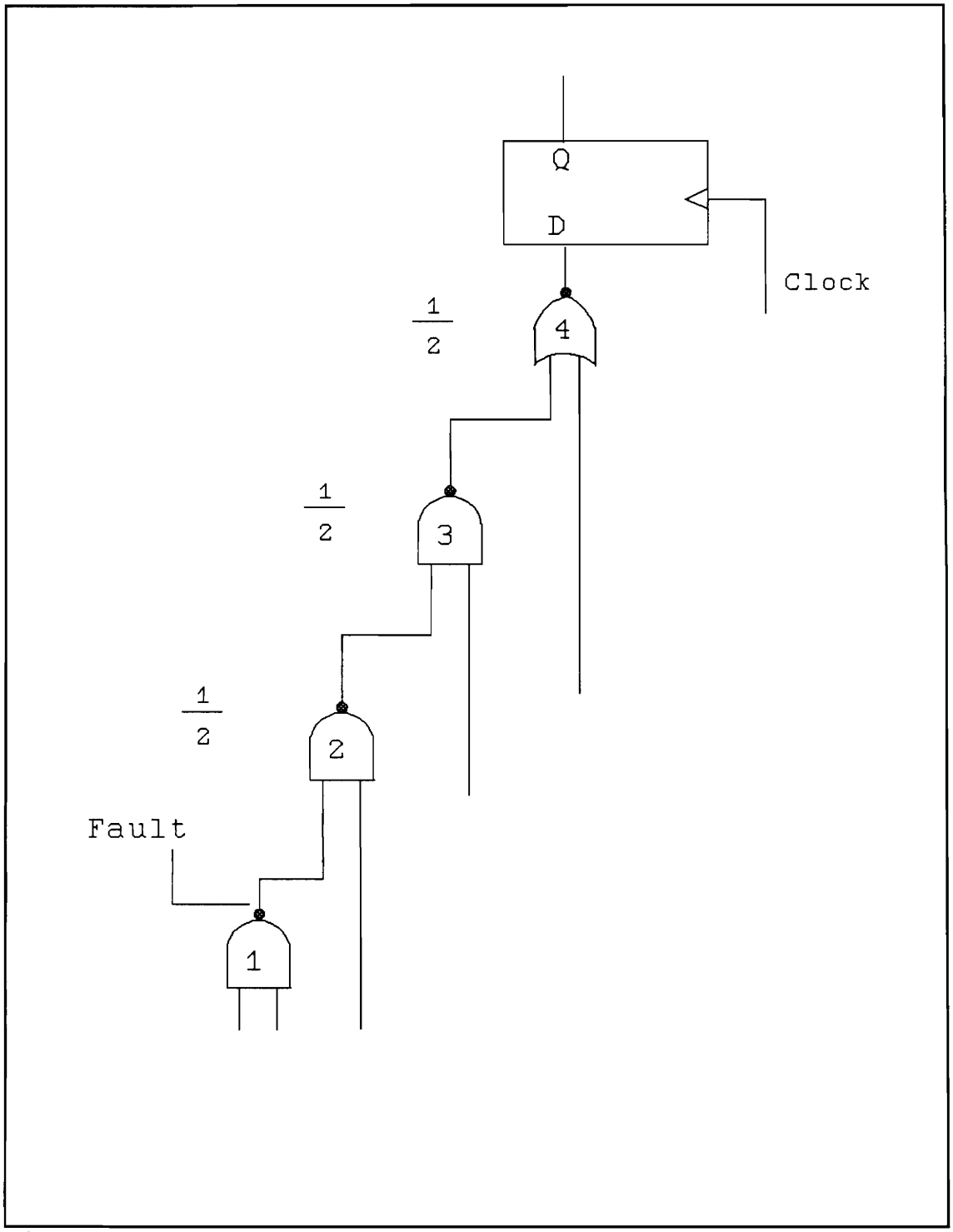



Figure 12. Example used for finding the critical path probability.

from the previous method. The internal logic value probabilities depend on the circuits structure and the inputs applied to the circuit when a transient fault occurs. Therefore, another method will be given in Chapter 4 that can be used to find the internal logic value probabilities of the circuit, so that accurate critical path probabilities can be found.

Now, that the critical path probability has been explained, the pulse must arrive at the flip-flops input at a time when it might be latched by the clocks active edge. This is discussed next using Figure 13.

### 3.4 PULSE ARRIVAL TIME

Notice that the D flip-flop, shown in Figure 13, has a set-up and hold time. The *setup* time is defined as the time needed for the input state to propagate through the flip-flop logic and to stabilize before the clock input changes, and the *hold* time specifies the time needed for the clock signal to propagate through the flip-flop and also have the data input stable. Those times must be met in order for a manufacturer to guarantee that the data present on the input will be latched. However, a transient pulse cannot be told when it will be allowed to arrive. So several events might occur when a transient pulse arrives at the input to a flip-flop.

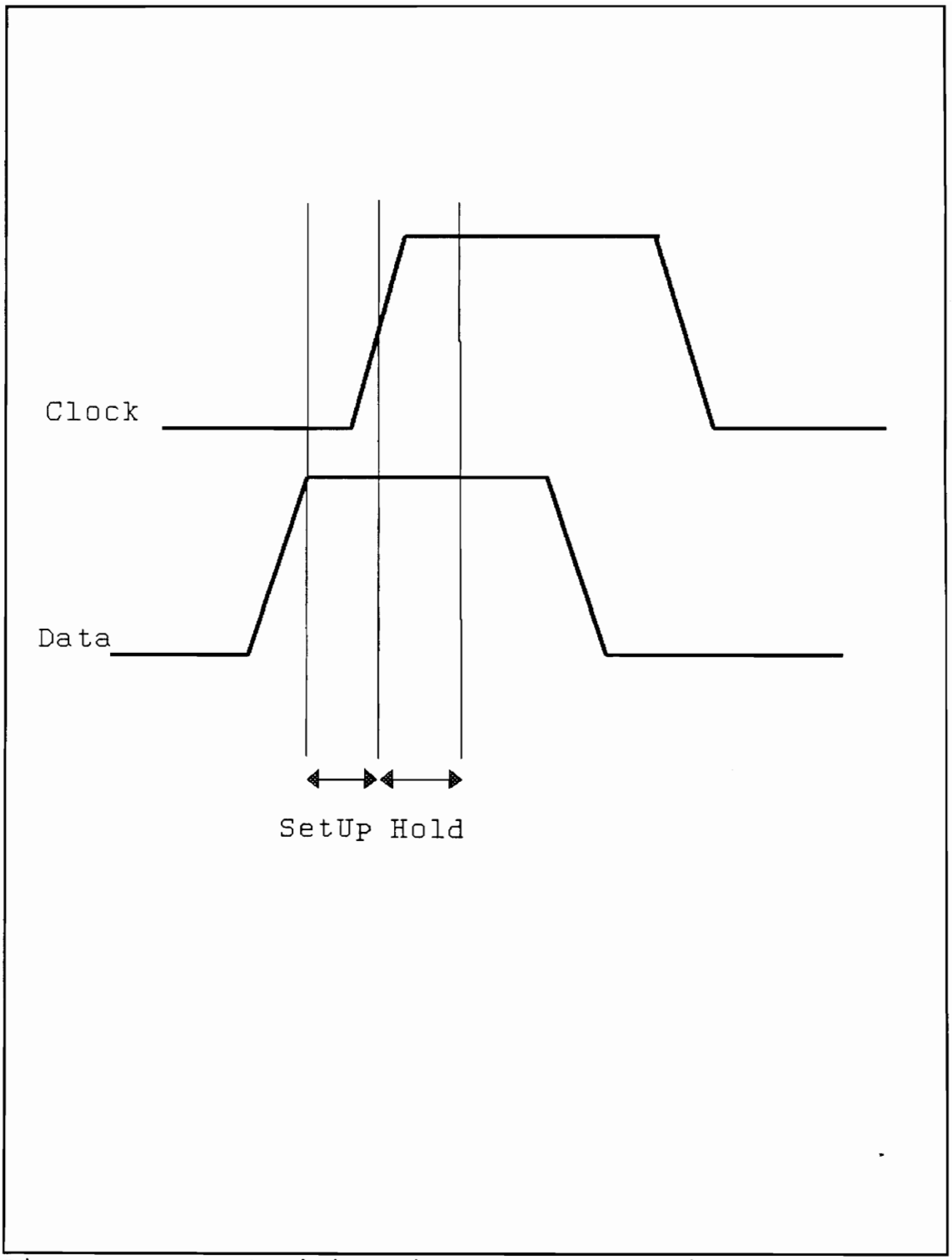



**Figure 13. Input timing diagram for a D flip-flop.**

1. If the transient fault has sufficient pulse width to meet the setup and hold time, then the fault will be latched on the output of the flip-flop.

2. If the transient fault changes the state of the input signal during a setup or hold time, the flip-flop's output might go into a metastable or indeterminate state.

3. If the transient fault transitions during a setup or hold time, the flip-flop might become unstable and go into oscillations.

The flip-flop will latch the erroneous logic value if event one occurs. However, if events two or three occur, the flip-flop's output might go into some unstable state which might still cause errors to propagate through out the system. So for all purposes in this research, if one of the above events occurs, the flip-flop is considered to have latched a erroneous signal or caused an error that will ultimately propagate through out the system and hinder correct system operation.

In Chapter 4, a formula will be derived that will find the

probability that the transient pulse will arrive at the proper time to cause an error at the output of the flip-flop. When this probability is found the total probability for the transient pulse to propagate down the critical path and to cause an error at the output of the flip-flop can be found. This total probability will be known as the *error probability* of the transient upset.

# 4.0 CALCULATING LATCH PROBABILITY FOR SEUs

## 4.1 INTRODUCTION

In section 4.2, existing programs and methods will be analyzed to find out if they might be suitable for finding the error probability of an SEU that occurs in combinational logic. Also, in section 4.3 the total solution for this research is given which includes two original methods for calculating the probability of setting up the critical path.

## 4.2 EXISTING PROGRAMS AND METHODS

### 4.2.1 EXISTING PROGRAMS

In searching for a method to find the error probability of an SEU, two existing programs were considered that might help to find the critical path probability. PODEM [9] and FAN [10] were chosen as candidates because of their performance with handling stuck-at faults.

First, PODEM applies logic levels to each primary input, one at a time, until the stuck-at fault is propagated to an output. The primary input and logic level are chosen using

some heuristic. Thus, one input pattern is found for each stuck-at fault, and the fault is propagated to only one output even if more are possible. PODEM therefore only gives one possible solution when many might be possible.

FAN is an improved PODEM algorithm where logic values are first assigned to nodes that will propagate the fault to an output. The program propagates the logic values forward and backwards from the point of the fault. While propagating backwards fan-out free regions might be reached. FAN stops proceeding backwards at this point, and propagates the fault forward until an output is reached. The program then assigns values to the fan-out free regions. FAN gives one pattern for each stuck-at fault, and also propagates the fault to just one output.

From the previous discussion, each program would be run for each fault in the circuit, therefore taking a lot of time. Also, the programs only produce a single test pattern for each fault, and the chosen test pattern might only propagate the fault to one output.

What is actually needed is a way to find all of the test patterns that can set up the critical path for each node in the circuit as described in Chapter 3, and then calculate the

probability of setting up that path. These programs showed no promise of giving such information easily.

### 4.2.2 DIEHL METHOD

Next, a method was considered that could be used to calculate the probability of latching transient upsets that occur in combinational logic. Diehl et al. [11] gave a method to find the error rate of SEUs that occur in combinational logic, and that method could be modified to find the error probability of an SEU occurring at any node in a circuit. However, the Diehl method lacked in several aspects to be discussed later.

As was discussed in Chapter 2 a sensitive volume exists for each logic gate, and a path must exist from that sensitive node to a flip-flops input before a transient pulse can be latched. Diehl et al. described a method that is shown in Figure 14. First, there exist (z) total sensitive nodes and each sensitive node has (L) total paths to a flip-flop. Each path, as shown in Figure 14, then has Rs nodes or (Rs - 1) gates associated with it. Table 6 gives the notation used along with the corresponding indices.

The truth tables are then examined for each gate to determine that the probability of propagating an upset through the gate

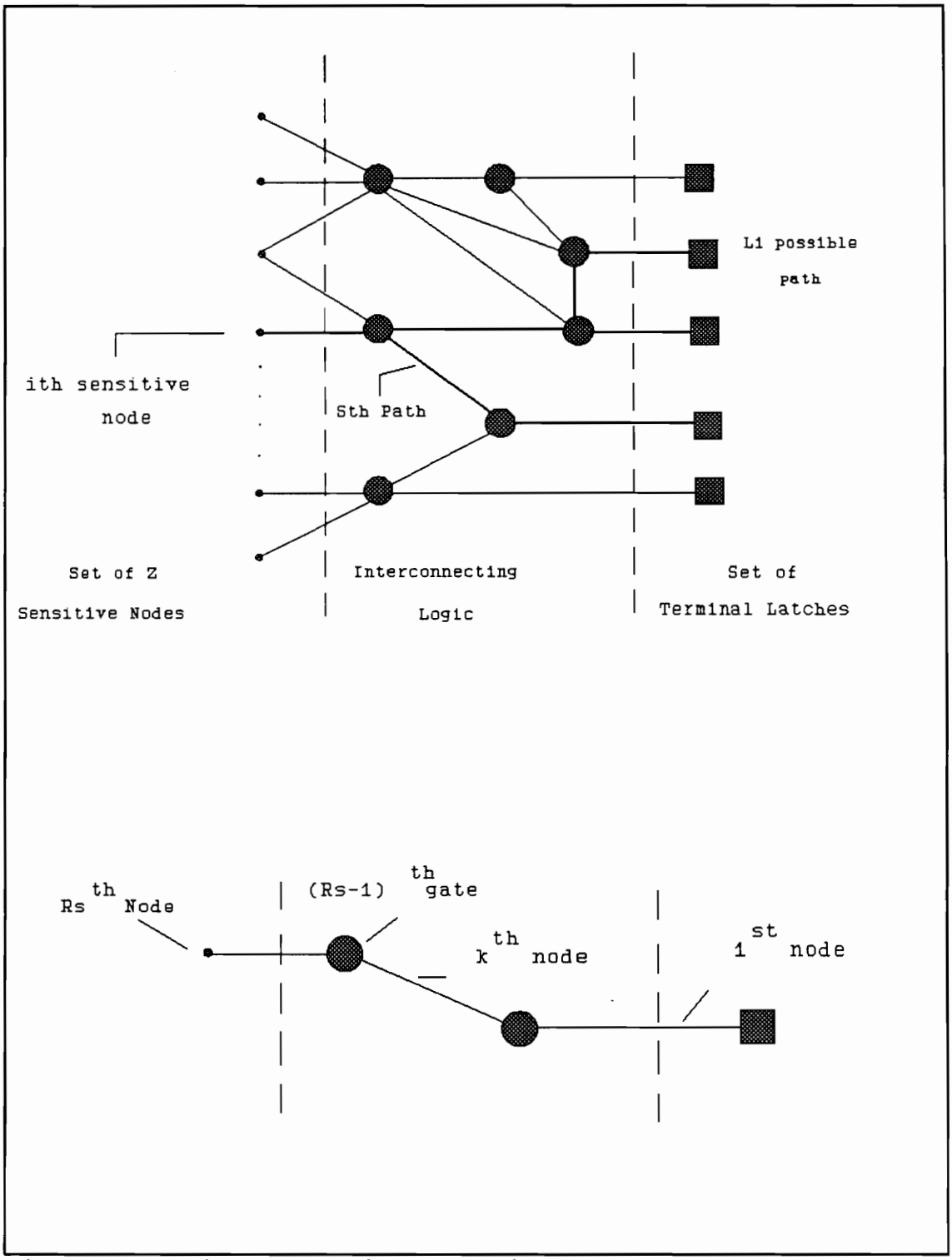



Figure 14. Figure showing the Diehl method.

Table 6. Notation used in the Diehl method.

| Item | Index | Total |
|---|---|---|
| Sensitive component types | n | N |
| Sensitive nodes | i | Z |
| Pipes from node i | s | Li |
| Gates in path s | k | Rs-1 |

is equal to $1/2^{m-1}$, where m equals the number of inputs to the gate. Note that this is the same formula derived in the previous chapter for upset propagation, and in effect each node in the circuit has a 1/2 probability of being at a logic one or a logic zero value. That formula is then incorporate into a more general formula to find a sensitive nodes critical path probability.

$$\sum_{1}^{L} \prod_{1}^{Rs-1} 2^{-(m-1)}$$

where m is the number of inputs to each gate in the path.

From the above formula, the gate probabilities ($1/2^{m-1}$) from the first gate to the (Rs-1) gate along a path from a sensitive node to a flip-flop are multiplied to find the total critical path probability. Each critical path probability for the first path to the $L^{th}$ path is then summed to give the total probability of propagating the upset to a flip-flop in the circuit.

Recall from Chapter 2 that one of the criteria for latching an upset is that it must arrive at the flip-flop at the proper time to become latched. This probability is given by Diehl et al. as Ws. Where Ws equals the SEU's pulse width divided by

the clock period. The total probability of latching a transient upset is given by Diehl et al. as:

$$\sum_{1}^{L} (Ws * \prod_{1}^{Rs-1} 2^{-(m-1)})$$

Diehl also assumes that once a particle strikes a sensitive node with an energy greater than $E_{min}$, the pulse will have the same magnitude and pulse width at the flip-flop as it does at the struck node. However, this assumption is not true in actual circuits.

The above method has four errors. First, Diehl's method assumes that each node has a 1/2 probability of being a logic zero or a logic one. However, nodes internal to a circuit might have higher or lower logic value probabilities depending on the circuits structure and the inputs applied to the circuit.

Diehl's method also sums the critical path probabilities for each possible path from a sensitive node. In other words, if a struck node has more that one path to different flip-flops, then the probability of each critical path is added together. If one path has an 80 percent probability and another path has

a 30 percent probability, then the result is a 110 percent probability of having the upset latched. In probability theory, however, the maximum probability cannot be more 100 percent.

Third, Diehl's method does not consider the flip-flops input setup and hold time, and last, the method does not consider that the transient pulse might be modified as it propagates down the critical path.

Given all of the above inabilities for Diehls method, a better solution is needed. In the next section, a solution is given that is based on Diehl's method but corrects the above inabilities.

### 4.3 RESEARCH SOLUTION

In order to find a solution to calculating the error probability of an SEU, all four criteria discussed in Chapter 3 must be met. Diehl's method discussed in the last section is close to meeting all of those criteria, but lacks in several aspects. For this research, a better solution is sought by introducing two original methods for finding the critical path probability. Also, solutions are presented for accounting for a flip-flop's setup and hold time and for modifications made to the transient pulse as it propagates

down the critical path. Finally, a summary is given of the total solution to be used in this research.

### 4.3.1 CALCULATING THE CRITICAL PATH

In Diehls method, each node has a 1/2 probability of being at either a logic one or a logic zero value. This might not be true depending on the structure of the circuit and the input applied to the circuit when the SEU occurs. Since the occurrence of SEUs are random, exactly which input pattern will be applied to the primary inputs when it occurs cannot be known. So if the assumption is made that any input combination is equally likely to be applied to the circuit when the upset occurs, then the logic value probabilities at each node in the circuit can be found.

Given the above assumption, the probability building and probability propagation methods are presented which will give the logic value probabilities at each node in the circuit. This in turn will allow the critical path probability to be calculated in a similar manner as discussed in section 4.1.6.

#### 4.3.1.1 PROBABILITY BUILDING

Given the above assumption, a straight forward method for finding the inner circuit logic value probabilities can be found if all of the possible input patterns are applied to and

propagated through the circuit. A count is then kept of the number of logic ones and zeros that occur on each node in the circuit. Each count could then be divided by the total number of patterns applied to the circuit to get the logic zero and the logic one probabilities for each node in the circuit when the SEU occurs.

Shown in Figure 15 is a simple circuit to illustrate the probability building method. Also, the truth table shown Table 7 will be used to verify the results.

After applying all of the input patterns, the node probabilities are calculated as show in Figure 15. The output node is shown to have a 3/32 probability of having a logic one value and a 29/32 probability of having a logic zero value when an SEU occurs. From the logic values in Table 7, all of the other node probabilities agree with the truth table as expected.

Note, that the node probabilities for the circuit in Figure 15 are quit different from the 1/2 proposed by Diehl. In fact the output node is 13/32 different from the 1/2 probability proposed by the Diehl method. So, obviously, the internal nodes might have higher or lower inner circuit logic value probabilities.

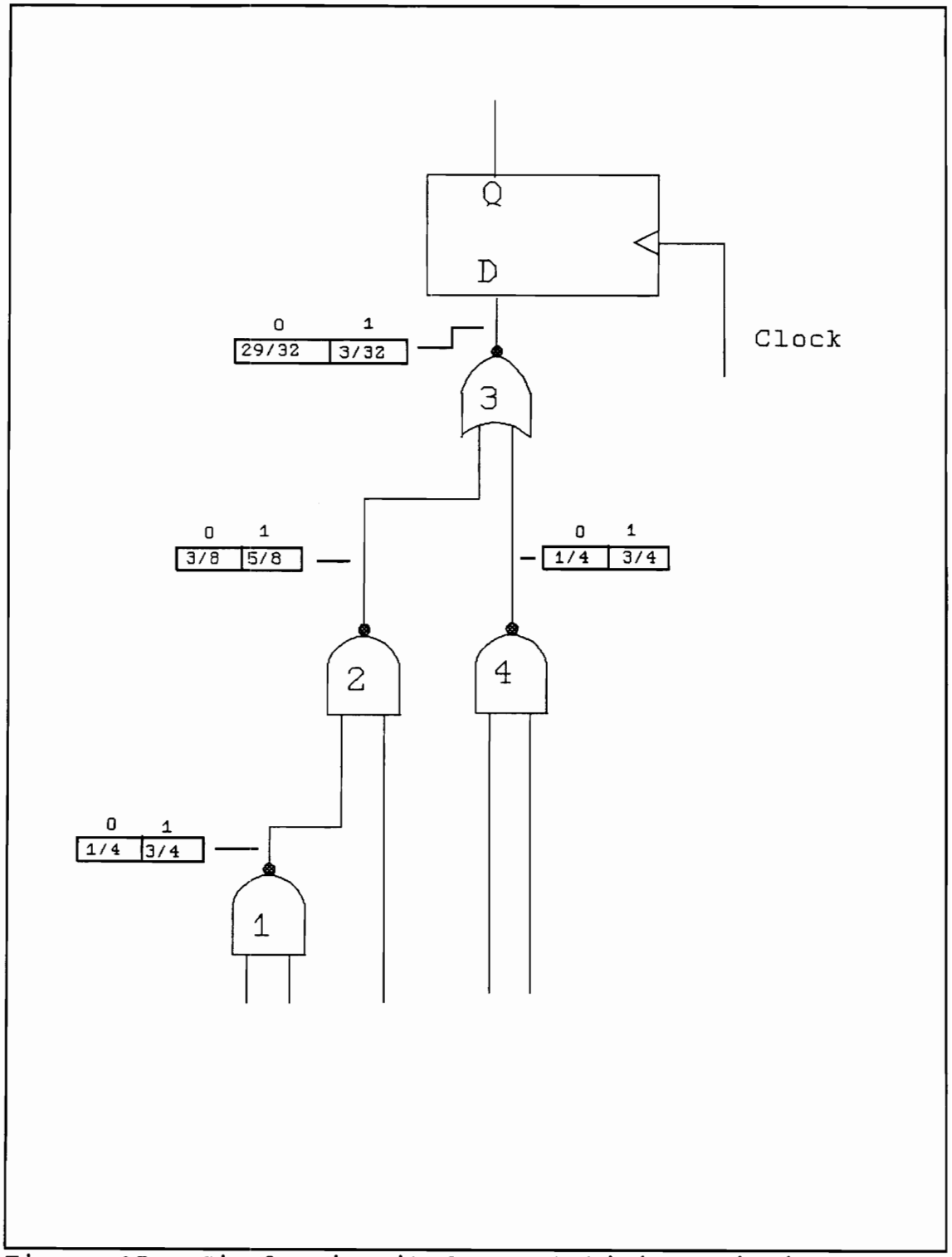



Figure 15. Simple circuit for probability building.

**Table 7. Truth table for the circuit in Figure 15.**

| INPUTS | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| 0 0 0 0 0 | 1 | 1 | 1 | 0 |
| 0 0 0 0 1 | 1 | 1 | 1 | 0 |
| 0 0 0 1 0 | 1 | 1 | 1 | 0 |
| 0 0 0 1 1 | 1 | 1 | 0 | 0 |
| 0 0 1 0 0 | 1 | 0 | 1 | 0 |
| 0 0 1 0 1 | 1 | 0 | 1 | 0 |
| 0 0 1 1 0 | 1 | 0 | 1 | 0 |
| 0 0 1 1 1 | 1 | 0 | 0 | 1 |
| 0 1 0 0 0 | 1 | 1 | 1 | 0 |
| 0 1 0 0 1 | 1 | 1 | 1 | 0 |
| 0 1 0 1 0 | 1 | 1 | 1 | 0 |
| 0 1 0 1 1 | 1 | 1 | 1 | 0 |
| 0 1 1 0 0 | 1 | 0 | 1 | 0 |
| 0 1 1 0 1 | 1 | 0 | 1 | 0 |
| 0 1 1 1 0 | 1 | 0 | 1 | 0 |
| 0 1 1 1 1 | 1 | 0 | 0 | 1 |
| 1 0 0 0 0 | 1 | 1 | 1 | 0 |
| 1 0 0 0 1 | 1 | 1 | 1 | 0 |
| 1 0 0 1 0 | 1 | 1 | 1 | 0 |
| 1 0 0 1 1 | 1 | 1 | 0 | 0 |
| 1 0 1 0 0 | 1 | 0 | 1 | 0 |
| 1 0 1 0 1 | 1 | 0 | 1 | 0 |
| 1 0 1 1 0 | 1 | 0 | 1 | 0 |
| 1 0 1 1 1 | 1 | 0 | 0 | 1 |
| 1 1 0 0 0 | 0 | 1 | 1 | 0 |
| 1 1 0 0 1 | 0 | 1 | 1 | 0 |
| 1 1 0 1 0 | 0 | 1 | 1 | 0 |
| 1 1 0 1 1 | 0 | 1 | 0 | 0 |
| 1 1 1 0 0 | 0 | 1 | 1 | 0 |
| 1 1 1 0 1 | 0 | 1 | 1 | 0 |
| 1 1 1 1 0 | 0 | 1 | 1 | 0 |
| 1 1 1 1 1 | 0 | 1 | 0 | 0 |

This method is, therefore, better at calculating the internal logic value probabilities because it takes into account the circuits structure and the possible inputs that might be applied to the circuit at the time that an upset occurs. Note that *internal logic value probabilities* refers to the probability of a node in the circuit receiving a logic zero or a logic one value when a transient occurs. However, the probability building method might be time consuming when put to actual use for a circuit with a large number of inputs, therefore another method is considered in an attempt to speed up the process of finding the inner logic value probabilities.

#### 4.3.1.2 PROBABILITY PROPAGATION

The above method of probability building is a very straight forward and easy method to find the needed internal logic value probabilities, but as noted earlier the method might be very time consuming for a large number of primary inputs. So the following method of probability propagation is proposed which is based on the same assumption as the probability building method. That is, each possible input pattern has an equal probability of being applied to the circuit when an SEU occurs. Therefore, each input to the circuit has a 50 percent or (1/2) probability of having a logic zero or a logic one value applied to it.

Using that assumption, the input probabilities can be propagated through the circuit to find the internal logic value probabilities. However, in order to understand the method of probability propagation, the truth tables for each type of gate will first be examined. Shown in Table 8 are the output probabilities for each possible input combination to a two-input NAND gate.

The logic one output probability for the AB = 00 input combination in Table 8 is found by multiplying 1/2, the probability that the A input will be a logic zero, by 1/2, the probability that the B input will be a logic zero. This gives a 1/4 probability that the output will receive a logic one value, and all of the other output probability values are calculated the same way. By summing the output probabilities for an output of logic one, the result is a 3/4 probability that the output will have a logic one value given that the input probabilities are 1/2. Thus, from Table 8, the probability that the output will be a logic zero or a logic one is 1/4 and 3/4, respectively.

Now, suppose that the inputs to the two-input NAND gate have different probabilities other than 1/2. Suppose that the inputs have a 1/4 chance of being a logic zero and a 3/4 chance of being a logic one which is shown in Table 9. Then,

Table 8. Truth table for a two-input NAND gate showing the output probabilities.

| A | B | Y | P(Y) |
|---|---|---|---|
| 0 | 0 | 1 | 1/2*1/2 = 1/4 |
| 0 | 1 | 1 | 1/2*1/2 = 1/4 |
| 1 | 0 | 1 | 1/2*1/2 = 1/4 |
| 1 | 1 | 0 | 1/2*1/2 = 1/4 |

total = 4/4

Table 9. Truth table for a two-input NAND gate showing the output probabilities.

| A | B | Y | P(Y) |
|---|---|---|---|
| 0 | 0 | 1 | 1/4*1/4 = 1/16 |
| 0 | 1 | 1 | 1/4*3/4 = 1/16 |
| 1 | 0 | 1 | 3/4*1/4 = 3/16 |
| 1 | 1 | 0 | 3/4*3/4 = 9/16 |

total = 16/16

from the truth table, the probability of producing a logic zero value on the output for AB = 11 becomes the product of 3/4, for the A input being set to logic one, and 3/4, for the B input being set to logic one, which equals 9/16. The output probability for AB = 01 is equal to (1/4 * 3/4) 3/16, and the other outputs are found the same way. Therefore, the probability that the output of the gate will receive a logic one is 7/16, and the corresponding logic zero output probability is 9/16.

The truth table in Table 10 is for a two-input NOR gate with a 1/2 probability of receiving a logic zero or logic one value on it's inputs. Note, that the output probabilities in the table are found the same way as for the NAND gate. However, the resulting output probabilities for a logic one and a logic zero are 1/4 and 3/4, respectively. Also, in Table 11 the two-input NOR gate has a 1/4 probability of receiving a logic zero and a 3/4 probability of receiving a logic one on it's input. From that table, the logic one output probability is 3/16, and corresponding logic zero output probability is 13/16.

#### 4.3.1.2.1 EXAMPLE OF PROBABILITY PROPAGATION

Given this insight into propagating the input probabilities to the output of a gate, the probability propagation method can

Table 10. Truth table for a two-input NOR gate showing the output probabilities.

| A | B | Y | P(Y) |
|---|---|---|---|
| 0 | 0 | 1 | 1/2*1/2 = 1/4 |
| 0 | 1 | 0 | 1/2*1/2 = 1/4 |
| 1 | 0 | 0 | 1/2*1/2 = 1/4 |
| 1 | 1 | 0 | 1/2*1/2 = 1/4 |

total = 4/4

Table 11. Truth table for a two-input NOR gate showing the output probabilities.

| A | B | Y | P(Y) |
|---|---|---|---|
| 0 | 0 | 1 | 1/4*1/4 = 1/16 |
| 0 | 1 | 0 | 1/4*3/4 = 3/16 |
| 1 | 0 | 0 | 3/4*1/4 = 3/16 |
| 1 | 1 | 0 | 3/4*3/4 = 9/16 |

total = 16/16

now be extended to the circuit level. As shown in Figure 15 the logic level probabilities at each node have been calculated. Here again, the assumption was made that a 1/2 probability exists that the primary inputs will be at either a logic one or a logic zero value when a transient occurs in the circuit. The output of NAND gates 1 and 4 in Figure 15 have a 1/4 probability of being at a logic zero value because of the one-one pattern (1/2 * 1/2) on their inputs. The corresponding logic one output probability for gates 1 and 4 can be found by either subtracting the logic zero probability from one (1 - 1/4) to get 3/4 or by finding the output probabilities of the zero-zero (1/4), zero-one (1/4), and the one-zero (1/4) input logic combinations and then adding them to get 3/4.

The probability that gate 2 will produce a logic zero on its output is equal to the product of 3/4, the probability of receiving a logic one on the output of gate 1, and 1/2, the probability of receiving a logic one on the input to gate 2. This gives the 3/8 probability shown in Figure 15. The corresponding logic one output value is calculated as before for gates 1 and 4 which results in the 5/8 probability shown. Last, the logic one output probability for gate 3 can be found by multiplying 3/8, the gate 2 logic zero output value, by 1/4, the gate 4 logic zero output value, to get 3/32. The

corresponding logic zero output probability is 1 - 3/32 which equals 29/32.

Comparing the resulting logic value probabilities using the probability propagation method with those using the probability building method, they are the same. Also, note that the probability propagation method can be done in one sweep of the circuit while the probability building method requires $2^n$ sweeps to get the required logic value node probabilities. Here, n is the total number of primary inputs to the circuit.

#### 4.3.1.2.2 RECONVERGENT FAN-OUT AND PROBABILITY PROPAGATION

For the probability building method reconvergent fan-out in a combinational logic circuit is not a problem. However, for probability propagation, reconvergence becomes a very complex issue. To illustrate some of this complexity, the circuit in Figure 16 will be used.

First, probability propagation yields the values shown in Figure 16. Note that this method indicates that the output of the circuit at gate 4 has a 25/64 and a 39/64 chance of producing a logic zero or a logic one, respectively. Table 12 is the truth table for the circuit in Figure 16, and it

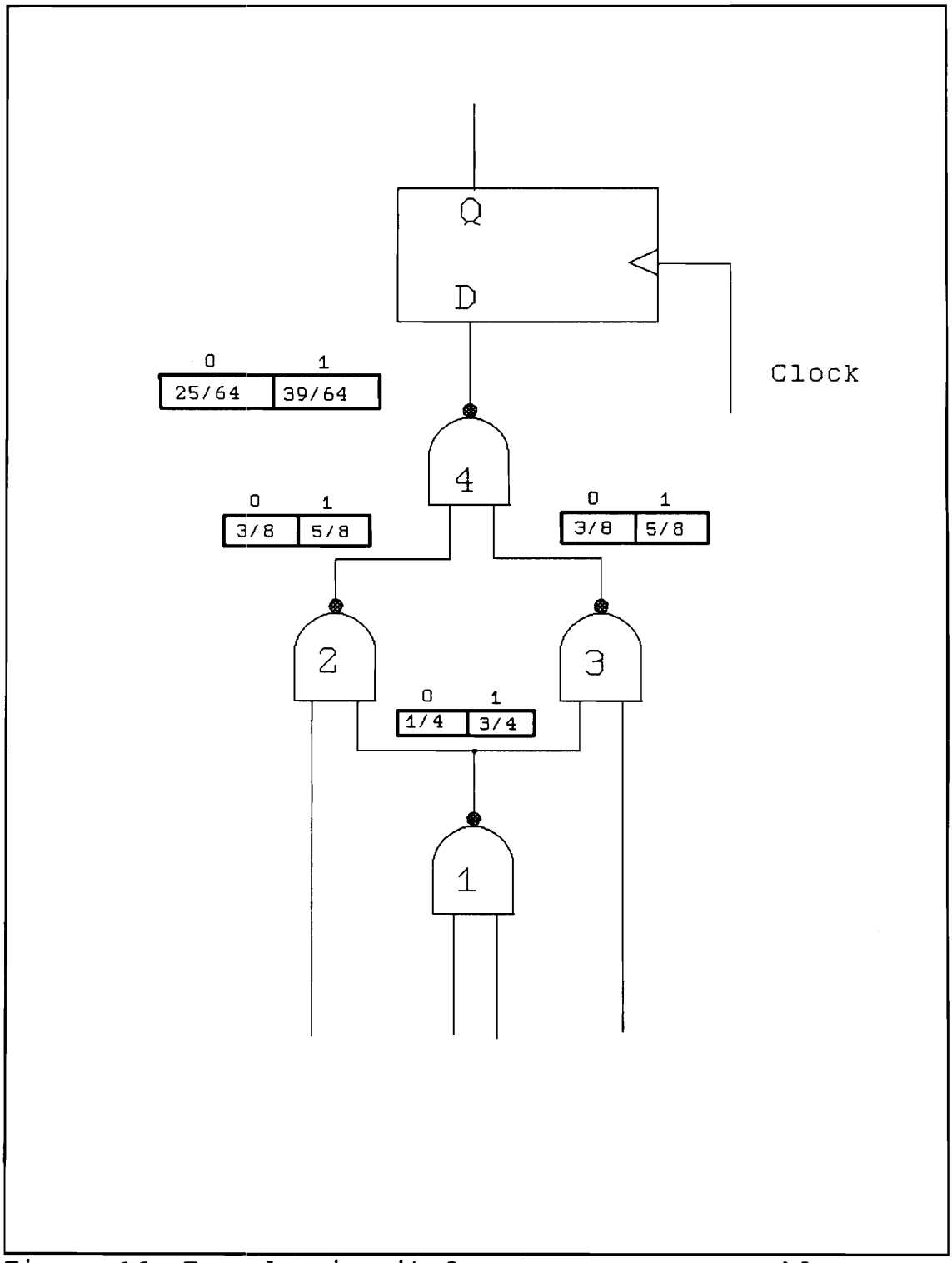



Figure 16. Example circuit for reconvergence problem.

Table 12. Truth table for Figure 16.

| INPUT | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| 0000 | 1 | 1 | 1 | 0 |
| 0001 | 1 | 1 | 0 | 1 |
| 0010 | 1 | 1 | 1 | 0 |
| 0011 | 1 | 1 | 0 | 1 |
| 0100 | 1 | 1 | 1 | 0 |
| 0101 | 1 | 1 | 0 | 1 |
| 0110 | 0 | 1 | 1 | 0 |
| 0111 | 0 | 1 | 1 | 0 |
| 1000 | 1 | 0 | 1 | 1 |
| 1001 | 1 | 0 | 0 | 1 |
| 1010 | 1 | 0 | 1 | 1 |
| 1011 | 1 | 0 | 0 | 1 |
| 1100 | 1 | 0 | 1 | 1 |
| 1101 | 1 | 0 | 0 | 1 |
| 1110 | 0 | 1 | 1 | 0 |
| 1111 | 0 | 1 | 1 | 0 |

clearly shows that the logic zero probability at the output of the circuit should be 7/16, and the corresponding logic one output probability is 9/16. If the probabilities at node 2 and 3 are then compared with Table 12, they are correct. So what could cause this discrepancy?

Insight into this problem can be obtained, if the evolution of the logic value probabilities at each node in the circuit are traced through the circuit to the reconvergence point at gate 4. If the primary input probabilities are propagated through gate 1 then the new values become 1/4 and 3/4 for the fan-out stem. Next, the probabilities are propagated through gates 2 and 3 yielding the logic zero and logic one probability equations shown in Figure 17. Under each equation is a line showing the various combinations of inputs that will provide the output value under consideration. The term FO, in Figure 17, denotes the logic value on the fan-out stem, and the other logic values in the equation are for the other inputs to the gate under consideration. Those probabilities are then propagated through gate 4 which then form the final logic value probability equations in Figure 17.

In the probability equation for gate 4, the fan-out stem values, which were propagated down two paths, are multiplied by each other at the convergence gate. This causes a

```
          output
          values

Gate 2:       0 :   1/2 * 3/4    = 3/8
                     1   1(FO)

              1 :   1/2 * 1/4 + 1/2 * 3/4 + 1/2 * 1/4   = 5/8
                     0   0(FO)    0   1(FO)    1   0(FO)

Gate 3:       0 :   1/2 * 3/4   = 3/8
                     1    1(FO)

              1 :   1/2 * 1/4 + 1/2 * 3/4 + 1/2 * 1/4   = 5/8
                     0   0(FO)    0   1(FO)    1   0(FO)

Gate 4:       0 :   1/2 * 1/4 * 1/2 * 1/4 +
                     0   0(FO)    0   0(FO)
                ^    1/2 * 3/4 * 1/2 * 1/4 +
                     0   1(FO)    0   0(FO)
                     1/2 * 1/4 * 1/2 * 1/4 +
                     0   0(FO)    0   0(FO)
                ^    1/2 * 1/4 * 1/2 * 3/4 +
                     0   0(FO)    0   1(FO)
                     1/2 * 3/4 * 1/2 * 3/4 +
                     0   1(FO)    0   1(FO)
                ^    1/2 * 1/4 * 1/2 * 3/4 +
                     1   0(FO)    0   1(FO)
                     1/2 * 1/4 * 1/2 * 1/4 +
                     0   0(FO)    1   0(FO)
                ^    1/2 * 3/4 * 1/2 * 1/4 +
                     0   1(FO)    1   0(FO)
                     1/2 * 1/4 * 1/2 * 1/4     =  25/64
                     1   0(FO)    1   0(FO)

              1 :   1/2 * 3/4 * 1/2 * 3/4 +
                     1   1(FO)    1   1(FO)
                ^    1/2 * 3/4 * 1/2 * 1/4 +
                     1   1(FO)    0   0(FO)
                     1/2 * 3/4 * 1/2 * 3/4 +
                     1   1(FO)    0   1(FO)
                ^    1/2 * 3/4 * 1/2 * 1/4 +
                     1   1(FO)    1   0(FO)
                ^    1/2 * 3/4 * 1/2 * 1/4 +
                     1   1(FO)    0   0(FO)
                     1/2 * 3/4 * 1/2 * 3/4 +
                     1   1(FO)    0   1(FO)
                ^    1/2 * 3/4 * 1/2 * 1/4    =  39/64
                     1   1(FO)    1   0(FO)
```

Figure 17. Probability equations for Figure 16.

duplication of information, and illegal combinations are also formed. The product terms that multiply a logic zero fan-out stem value by a logic one fan-out stem value are considered illegal because the fan-out stem can only have a single logic value at any one time, and those illegal combinations are denoted by a ^ in Figure 17. For example, the fan-out stem cannot produce a logic zero that will propagate through gate 2 to the input of gate 4 while at the same time producing a logic one that will propagate through gate 3 to the other input of gate 4.

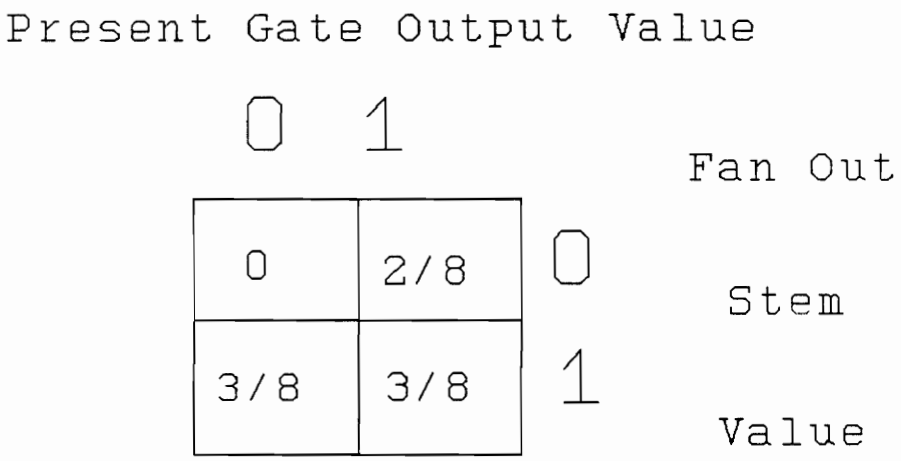
In Figure 18, the duplicate information has been replaced by a value of 1, and the illegal combinations have been eliminated. After simplifying the equation, the result yields the same probabilities expected from the truth table in Table 12. So in order to get the correct logic value probabilities at a reconvergence point, the fan-out stem information must only be propagated one way from the fan-out point, thus eliminating the duplication problem, and a method must be developed to eliminate the illegal combinations.

The solution to the illegal combination problem uses a table such as the one shown in Table 13. This table is known as the *reconvergence table*. The logic values along the top of the table refer to the output logic values of the gate presently

```
          output
          values

Gate 2:     0 :   1/2 * 3/4
                   1    1(FO)

            1 :   1/2 * 1/4 + 1/2 * 3/4 + 1/2 * 1/4
                   0   0(FO)   0   1(FO)   1   0(FO)

Gate 3:     0 :   1/2 * 1
                   1    1(FO)

            1 :   1/2 * 1 +  1/2 * 1   + 1/2 * 1
                   0   0(FO)  0   1(FO)   1   0(FO)


Gate 4:     0:    1/2 * 1/4 * 1/2   1   +
                   0   0(FO)   0   0(FO)
                  1/2 * 1/4 * 1/2 * 1   +
                   0   0(FO)   0   0(FO)
                  1/2 * 3/4 * 1/2 * 1   +
                   0   1(FO)   0   1(FO)
                  1/2 * 1/4 * 1/2 * 1   +
                   0   0(FO)   1   0(FO)
                  1/2 * 1/4 * 1/2 * 1     =  7/16
                   1   0(FO)   1   0(FO)

            1:    1/2 * 3/4 * 1/2 *  1  +
                   1   1(FO)   1   1(FO)
                  1/2 * 3/4 * 1/2 *  1  +
                   1   1(FO)   0   1(FO)
                  1/2 * 3/4 * 1/2 *  1    =  9/16
                   1   1(FO)   0   1(FO)
```

Figure 18. Reduced probability equations.

**Table 13. Table used for reconvergent circuits.**



having the table propagated through it. From here on, that gate will be known as the *present gate*. The values on the side of the table refer to the logic values present on the fan-out stem. Found in the top left block of the table in Table 13 is the probability of a logic zero appearing on the output of the present gate when the fan-out stem has a logic zero value. In the case of Table 13 that probability is 0. Also, if the fan-out stem has a logic one value, then the logic zero output probability for the present gate is given in the bottom left block of the table which shows a 3/8 probability. The other two blocks have similar explanations. This table can now be propagated down two paths from the fan-out stem to the input of the reconvergence gate. One table will resemble Table 13, and the other will have ones in the place of all probabilities that are not zero. To illustrate the use of this table, the circuit in Figure 16 will be used again as shown in Figure 19.

The reconvergence table is set up on each branch of the output of gate 1 in Figure 19. For the initial set up the present gate is considered to be the gate driving the fan-out stem which in this case is gate 1. The table for branch 1, on the left of the fan-out stem, shows that the probabilities at this fan-out stem for a logic zero and for a logic one are 1/4 and 3/4, respectively. Notice that 1/4 is in the top left block

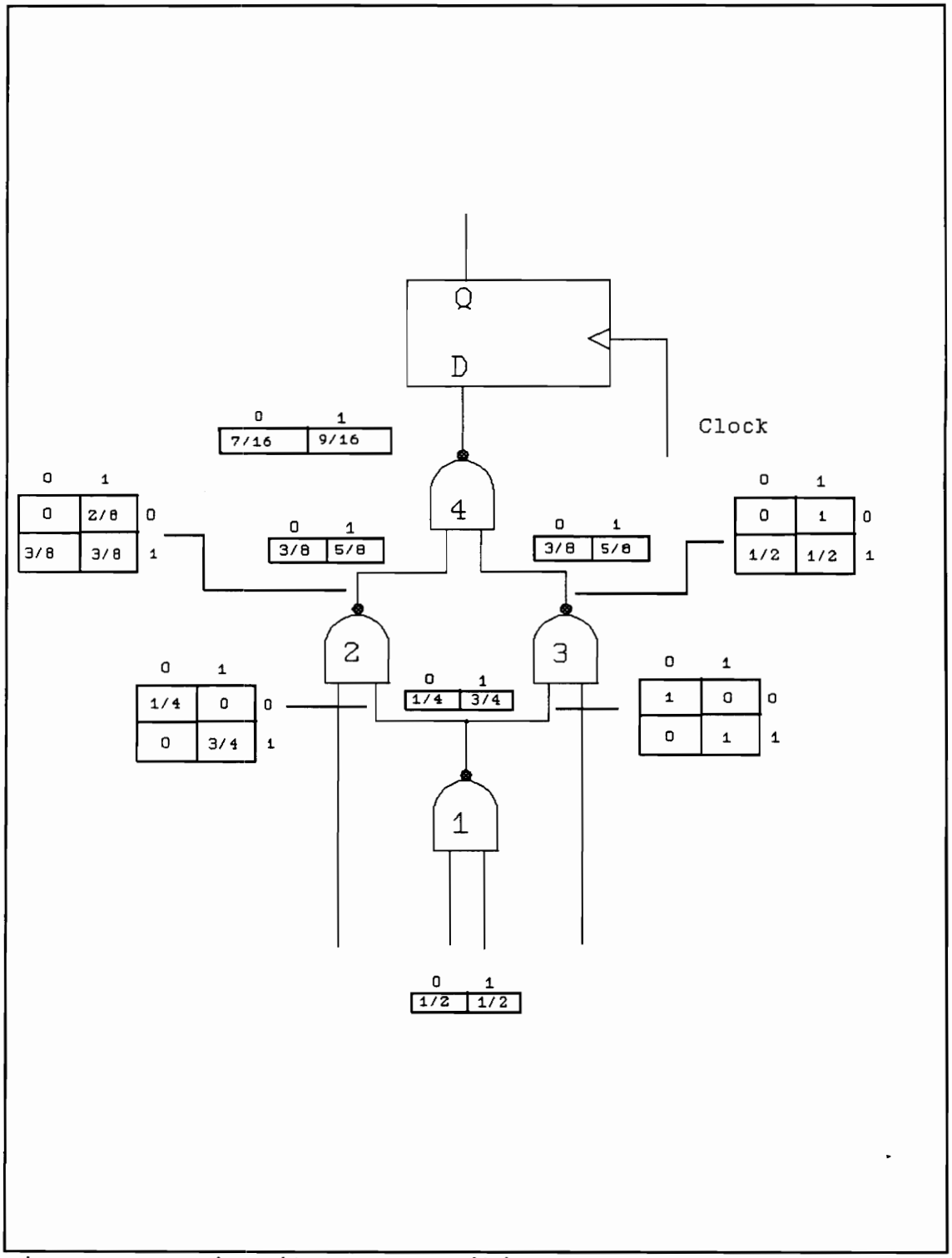



Figure 19. Circuit for describing the reconvergence table.

of the branch 1 table which signifies a logic zero stem value and a logic zero present gate output value, and the 3/4 probability in the bottom right block signifies a logic one stem value and a logic one present gate output value. All other blocks contain a zero value since at the output of this gate a logic zero stem value and a logic one present value would not make sense and vice versa. The table for branch 2, shows ones along the diagonal, and zeros elsewhere. The ones are used so that duplicate information is not propagated to the reconvergence point, and the zeros are used as before. Note that the branch assignment of the two tables is arbitrary. The tables could be switched and the same result would be obtained at the reconvergence point gate 4.

The reconvergence tables and other probabilities can now be propagated through gates 2 and 3. In the branch 1 table, on the left fan-out branch, the values along the top now represent the logic values on the output node of gate 2, the present gate. The values down the side represent the logic fan-out stem values at the output of gate 1.

In order to get the probability of a logic zero on the output of gate 2, the inputs to gate 2 must have all logic one values applied to them. Therefore, if the logic one probability from the primary input to gate 2 (1/2) is multiplied by each value

in column one of the branch 1 table (1/4,0), the probability of getting a zero on the output of gate 2 is found to be (1/8,0). From here on, the values in parentheses represent a column of the reconvergence table. The left value in parentheses is from the top row of the table, and the right value is from the bottom row of the table.

The probability of getting a logic one on the output of gate 2, however, is dependent on the probability of receiving the zero-zero, zero-one, and one-zero input logic combinations on gate 2. The logic one output probability is thus found by finding the probability of each of those combinations and summing to get the result. First, the logic zero primary input probability (1/2) is multiplied by the probabilities in the logic zero column of the branch 1 table (1/4,0) to get (1/8,0). Then, the input probability (1/2) is multiplied by the probabilities in the logic one column (0,3/4) of the table to get (0,3/8). Last, the primary input probability for a logic one value (1/2) is multiplied by the probabilities in the logic zero column (1/4,0) of the table to get (1/8,0). When these values are added together, the output probabilities for the branch 1 table at the output of gate 2 are found to be (2/8,3/8), and the probabilities shown in the branch 2 table for the output of gate 3 are found the same way.

Next, the branch 1 and the branch 2 tables are combined at gate 4, the point of reconvergence, in such a way as to eliminate any illegal combinations. This is easy since the fan-out stem values are found on the side of the table.

So the logic zero output probability for gate 4 is found by multiplying the probabilities in column one of the branch 1 table by the probabilities in column one of the branch 2 table. That is, (2/8,3/8)*(1,1/2) = (2/8*1)+(3/8*1/2) = 7/16 the correct zero output probability. The logic one probability can now be found by subtracting the zero probability, 7/16, from one, or by summing the other three combinations formed by the table columns to get the desired result. Note that the logic value probabilities at each node in the circuit are found as before in Figure 16, and the reconvergence tables are created only to gather the probability information at each gate along the path to the reconvergence point. When the two tables meet, they are combined and are not carried further.

To extend this technique to higher numbers of reconvergent fan-out branches and to include complex circuits with many reconvergent fan-out points that interact with each other would require a very complex heuristic. However, this method of probability propagation drastically saves computation time

when compared to the probability building method, so future work to extend the technique might be fruitful.

In this research, however, the probability propagation method is used for simple forms of reconvergence which were described above. That is, the restriction applied here is that there exists only two paths from the fan-out point to the point of reconvergence, and those paths do not interact with other reconvergent paths from other fan-out points. From here on, that type of reconvergent structure will be known as *simple reconvergence*.

#### 4.3.1.3 LOOPS

In this section, a curious situation is discussed that occurs when the probability building or probability propagation methods are used on circuits that contain loops. Circuits that contain loops are typically used to remember previous states which allows the circuit to make future decisions based on the present inputs to the circuit and the present state. Classically circuits such as the Mealy and the Moore machines, shown in Figure 20, have been used for such purposes.

If the internal logic level probabilities of circuits containing loops are found by the probability building method, then different probabilities are found depending on the order

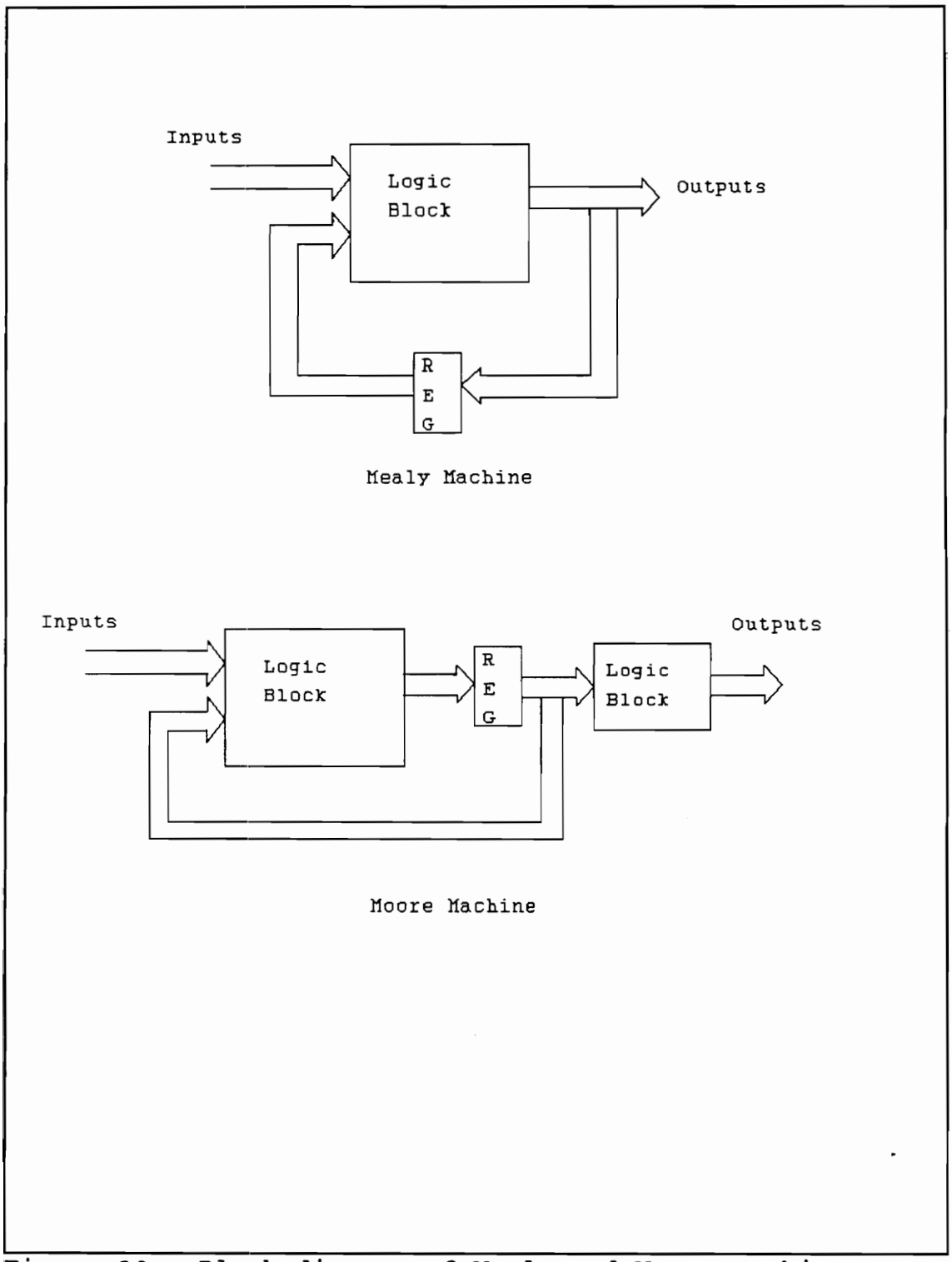



Figure 20. Block diagram of Mealy and Moore machines.

in which the inputs are applied and the assumed initial state of the flip-flops. This is understandable because the machine will proceed to a different next state depending on the input applied and the present state of the machine. In other words, if the inputs are applied in a different order, then the machine will go to different next states. Also, when the input combinations are applied in this way, the internal node probabilities represent the possibility that the logic value will occur on that node over a certain period of time. However, in this research, the logic values in the circuit at an instance of time is the concern. The concern is not the logic values over a period of time.

When a transient pulse occurs in a circuit with loops, the exact present state of the flip-flops and the primary input pattern applied to the circuit are not know. Therefore, in order to find the logic level probabilities, all of the input combinations and all of the state combinations need to be applied to the circuit.

This can only be done if the loops are broken as shown in Figure 21. Now, the loops can be used as pseudo-primary inputs when the probability building or probability propagation methods are used. Figure 22 shows an example of a Mealy machine having a cut in the feedback loop. The

probabilities are found using either one of the previously discussed methods. Notice that the loops are broken at the output of the flip-flop which does not hinder the process of finding the critical path probability for any node. Since the transient pulses of concern in this research are those that originally occur in combinational logic rather than at the output of flip-flops. However, it should be noted that the number of input patterns applied to the circuit for the probability building method increases by $2^m$ where m is the number of broken loops. This gives a total of $2^{(m+n)}$ input combinations, where n is the number of primary inputs to the circuit. The added computational overhead due to loops in the circuit makes the probability propagation method much more attractive.

#### 4.3.1.4 SUMMARY OF ADVANTAGES AND DISADVANTAGES

The probability building and probability propagation methods are good estimators of the internal logic value probabilities because they taken into account the circuits structure and the inputs applied to the circuit at the time that an upset might occur. If an upset is totally random, then an equal probability exists that any input pattern out of the total number possible, might be applied to the primary inputs. Using one of the two methods should then allow for better

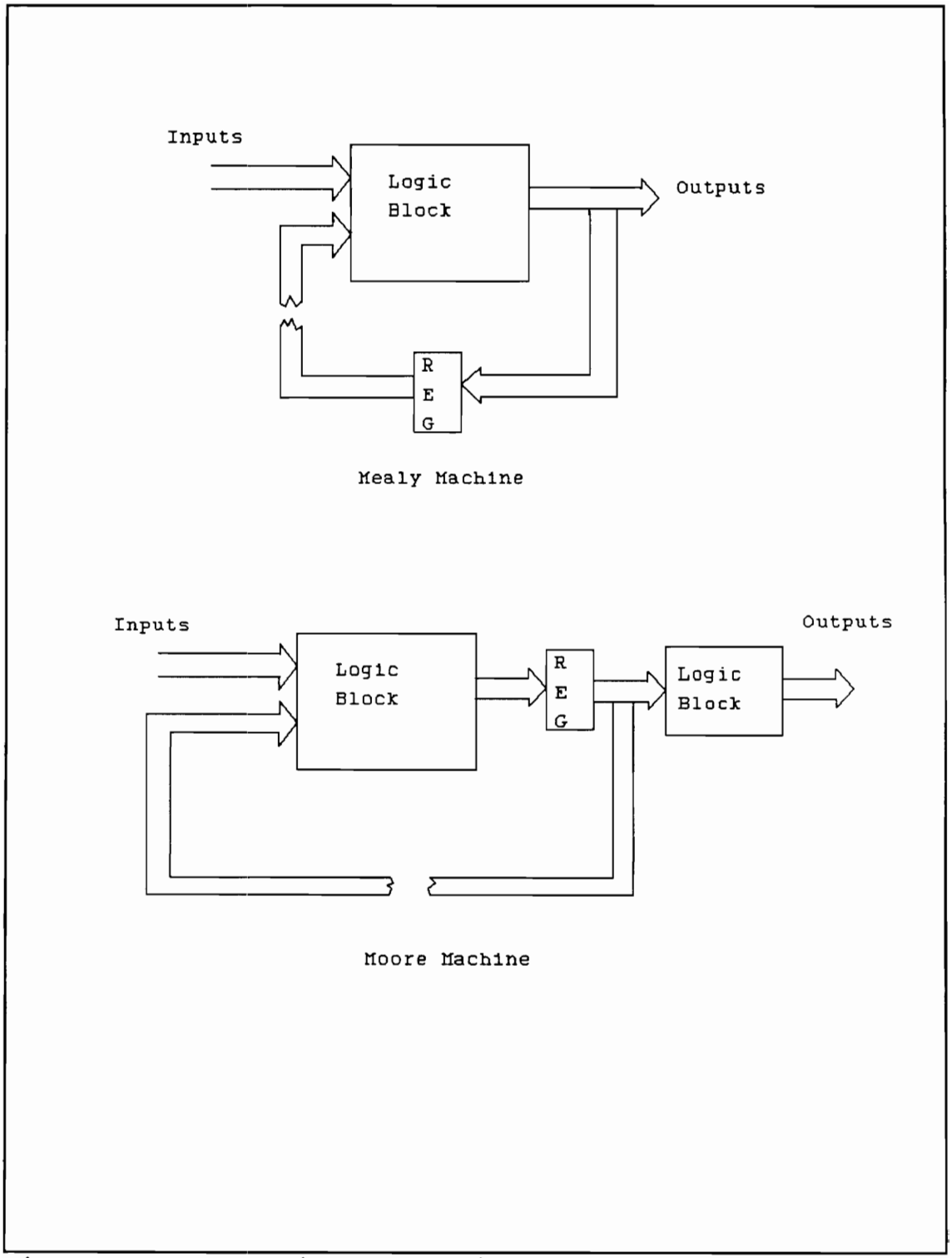



Figure 21. Block diagram showing broken loops.

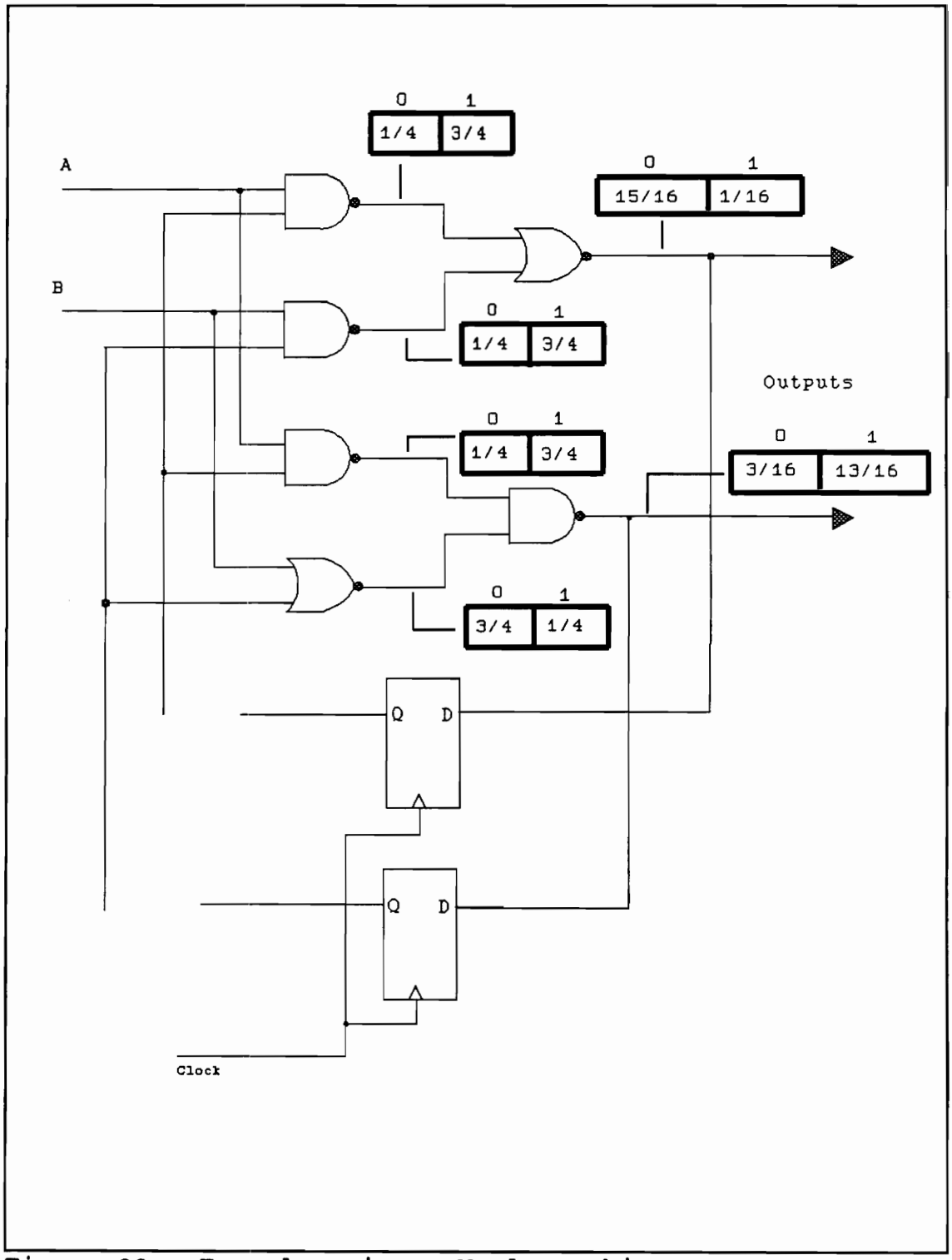



Figure 22. Example using a Mealy machine.

calculations of the critical path probability.

For the probability building method, a large number of inputs and gates might tend to make the processing time very large, but on the other hand, any circuit can be evaluated using the method. Probability propagation, on the other hand, is much faster, but only works for circuits with no reconvergence or simple reconvergence as described in the section on reconvergent fan-out and probability propagation. A rough processing time ratio can be found to give some idea of the difference in the two methods.

In general, the probability building method must propagate logic values through (N) gates for $2^n$ total input patterns where (n) is the total number of inputs applied to the circuit. This gives the following formula:

$$T_{pb} = 2^n * N * t_{inc} * m.$$

$T_{pb}$ is the total time for probability building to propagate all of the possible input patterns through the circuit, and $t_{inc}$ is the time needed to increment the logic value count at m inputs to each gate.

The probability building method must then divide each input count by the total number of input combinations to get the logic value probabilities on the input to each gate. When this processing time is added to the above formula, it yields:

$$T_{pb} = 2^n * N * t_{inc} * m + N * t_{div} * 2 * m.$$

The term 2*m in the formula denotes that the logic zero and the logic one count must both be divided by the total number of combinations for all inputs to a gate. If the processing time to increment a number is 20 percent of the time to divide two numbers ($t_{div}$), then the above formula can be simplified to:

$$T_{pb} = N * t_{div} * m * (2^n * (.2) + 2).$$

Next, the total time for the probability propagation method to propagate the probabilities through the circuit can be found if the circuit is assumed to not contain simple reconvergence. Since the method calculates the easiest output probability for a gate first, and then subtracts that number from one, the following formula is derived:

$$T_{pp} = N * (t_{mult}(m-1) + t_{sub})$$

where (m-1) multiplications are performed to propagate the easiest output probability. $T_{pp}$ is the time to propagate the probabilities, and $t_{mult}$ and $t_{sub}$ are the times needed for the computer to multiply and subtract two numbers, respectively. The easiest output probability to calculate for a NAND gate is the logic zero output probability, and for a NOR gate the logic one output probability is the easiest. This formula can be reduced if the processing time to subtract two numbers is 20 percent of the time to multiply two numbers.

$$T_{pp} = N*t_{mult}*(m-(0.8)).$$

Now, a ratio of $t_{pb}$ to $t_{pp}$ can be taken, and if $t_{mult}$ equals $t_{div}$ then the ratio R is:

$$R = \frac{m*(2^n*(0.2)+2)}{m-(0.8)}.$$

If the largest number of inputs to a gate is 3, the ratio can be reduced to:

$$R = \frac{(2^n*(0.6)+6)}{2.2}.$$

For a 10 input circuit, the ratio is equal to 282 to 1. This ratio tends to slightly underestimate circuits with a small number of inputs ( < 6 ) and overestimates circuits with a large number of inputs ( > 9 ), as will be shown in Chapter 5.

The ratio, however, is very good at showing that a large processing time difference exists between the two methods.

Since both methods have good advantages the probability propagation and the probability building methods will be used for this research, and which one used to evaluate a circuit will depend on the circuits structure.

#### 4.3.1.5 EXAMPLE OF CRITICAL PATH PROBABILITY

If an SEU occurs at the output of gate 1 in a fan-out free circuit, as indicated in Figure 23, then it must propagate through gates 3 and 5 to reach the flip-flop. Since the output of gate 2 must have a logic one value in order to propagate the upset through gate 3, the probability of setting up the critical path to this point is 3/4.

Next, the upset must pass through gate 5. Since this is a NOR gate, the input from gate 4 must be set to a logic zero value. From the figure, gate 4 has a 1/4 probability of having a logic zero on it's output. Thus, the total probability of setting up the critical path is 3/4 * 1/4 or 3/16.

If all of the input combinations to the circuit are examined, a total of 12 out of the 64 total combinations possible will set up the critical path. Reducing that number yields 3/16

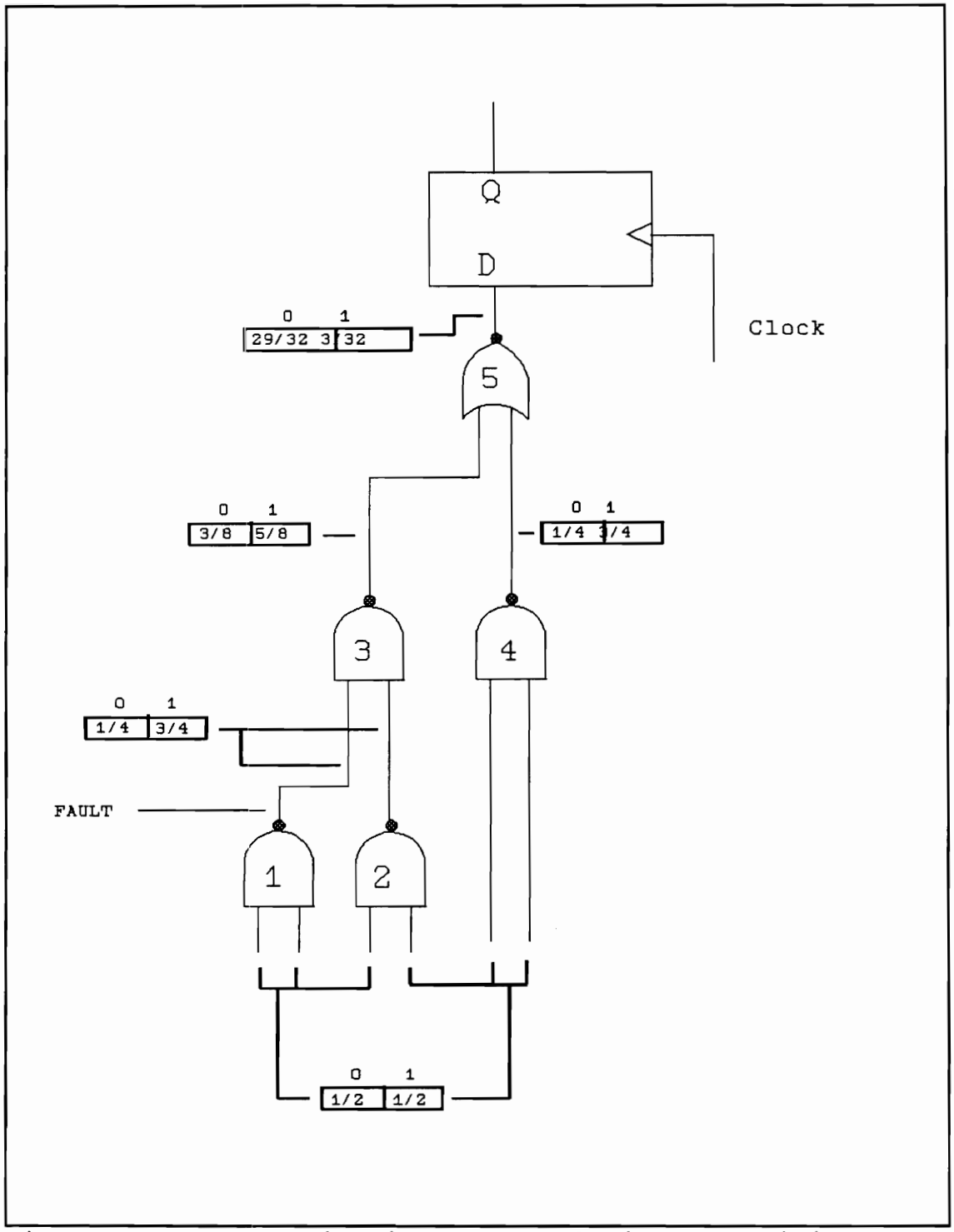



Figure 23. Example circuit for calculating the critical path probability.

the same result as calculated above.

#### 4.3.1.6 MULTIPLE PATHS

Recall that Diehls method found the error probability for each critical path from a sensitive node to a flip-flop and then added those individual error probabilities to get the total error probability. However, this is not a proper way to find the total error probability. Therefore, in this section a method is discussed that uses basic probability theory to find the total error probability.

The circuit in Figure 24 shows an upset that has several paths to different flip-flops. When the upset occurs, it can travel down all or none of the paths depending on the actual values on each node. But since the logic values on each node are not known when radiation strikes the circuit, either the probability building or the probability propagation method discussed earlier can be used. Figure 24 shows the calculated probabilities at each node.

Since the event of latching an upset at each flip-flop is an independent event, the resulting path probabilities can be multiplied together to find the probability that the upset causes an error at the output of all of the terminating flip-flops. The system, however, is in error if only one flip-flop

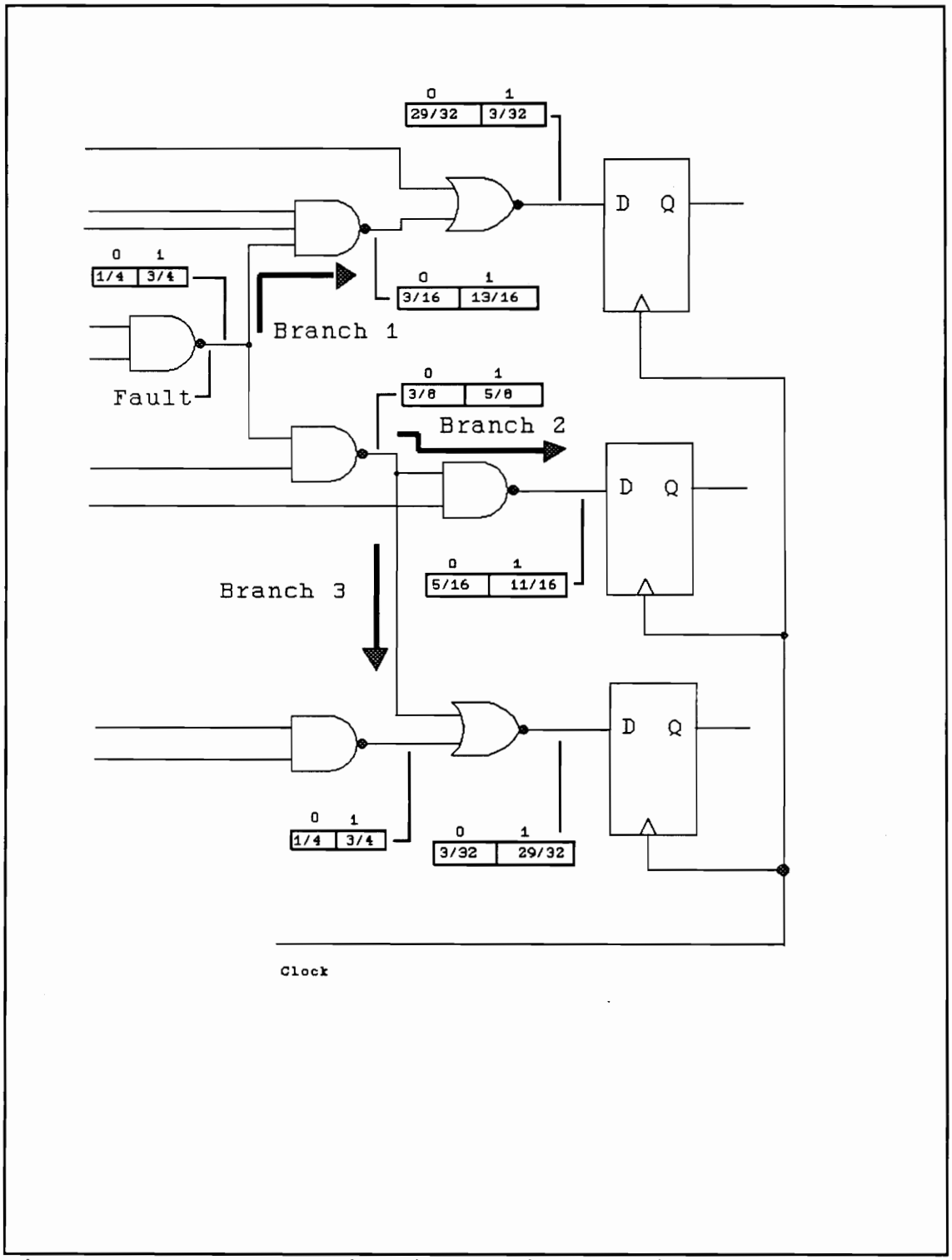



Figure 24. Example circuit showing multiple paths.

is in error. Therefore, a better indication of the ability of an SEU to cause an error at the output of a flip-flop would be the probability that the upset is latched in at least one flip-flop.

In order to calculate the probability that the upset will be latched in at least one flip-flop, the probability that the upset will not be latched by a flip-flop is first needed. If the error probability for each critical path is subtracted from one, then the probability that the upset will not be latched by that flip-flop is found. In other words, if the SEU error probability is P(path*i*) for the *i*th path from a struck node, then the probability of not causing an error at the output of a flip-flop is 1-P(path*i*).

Those probabilities for each path are then multiplied together to find the probability that the upset will not be latched by any flip-flop. Subtracting that probability from one gives the probability that the upset will be latched in at least one flip-flop.

$$1 - (1 - P(path1))*(1 - P(path2))...*(1 - P(path_i)) .$$

From Figure 24, the critical path probabilities for path 1, 2, and 3 can be found to be equal to 1/8, 1/4, and 1/8,

respectively. If the probability that the upset will arrive at the appropriate time to become latched in a flip-flop is 1/2, then the probability that the upset will become latched in at least one flip-flop is:

$$1- ((1- (1/8)*(1/2))*(1- (1/4)*(1/2))*(1 -(1/8)*(1/2))) = 473/2048.$$

This a 23 percent chance that the upset will be latched in at least one flip-flop. This is an improved method for calculating the error probability when the SEU can travel down several paths to a flip-flop.

### 4.3.2 PULSE WIDTH AND AMPLITUDE

Now that the critical path probability can be calculated using one of the above methods, the modifications made to the transient pulse as it propagates down the critical path are examined. A discussion of the Diehl method in section 4.1.2 showed that an assumption was made that if a particle generated a charge equal to or greater than the critical charge, then the SEU would propagate down the critical path and arrive at a flip-flop with the same amplitude and pulse width as the transient pulse generated at the original site. In this section, a method will be discussed that will show changes in the transient pulse's amplitude and pulse width as it propagates through the logic to the input of a flip-flop.

When a transient pulse propagates down a critical path, it might encounter different capacitive loads on the output of each gate in the critical path which might then change the shape of the transient pulse. Another possibility for changing the shape of the transient is that a gate in the critical path might have an insufficient gain to further propagate a rising or falling input pulse. Those differences could account for a lengthening of the pulse width or possible a decrease in the output amplitude of the pulse.

In order to account for changes in a transient pulse, a method is used that involves building a database of the electronic effects of each type of gate in the circuit. This is done by performing SPICE simulations on the different gates used in a circuit design. The amplitude and pulse width of a transient pulse on the input to a gate are used to look up the resulting output transient pulse's amplitude and pulse width in the database.

The simulations used to build the database were made by holding all of the inputs to a multiple input gate except one at a constant voltage. The value of that voltage is dependant on what type of gate is being simulated. VDD was used for NAND gates, and ground was used for NOR gates. The other input had a pulse applied to it with a certain amplitude and

pulse width to represent the transient pulse. The resulting output amplitude and pulse width were recorded and put in the database. For this research all circuit designs were done using MOSIS standard cells from the CMOSN library [12] and a source voltage of 5 volts. The gates used in the circuit designs included the two and three-input NAND gates, the two and three-input NOR gates, and the INVERTER. The transient pulse amplitude was varied at 0.5 volt increments from 0 volts to 2 volts and also from 4 volts to 5 volts. The interval from 2 volts to 4 volts was then varied at 0.1 volt increments. The pulse width was also varied, from 1 nanosecond(ns) to 9 ns at 1 ns increments, and the output results were recorded in the database. Also, a capacitive load of 1.11 picofarads was added to the output of the gate being simulated to represent the loading by other gates connected to the output. This load is taken to be three worst case loads or approximately nine typical loads from the MOSIS standard cells. This data should provide a good indication of how the output will react given the input transient pulse information.

The database is built in two different ways for each gate. First, the database is built using a rising pulse, and then the database is built using a falling pulse. This is done because the gate might perform differently for the two pulse types.

The database described above will be used in a program called SUPER II that will be discussed in Chapter 5. While the critical path probability is being found by SUPER II, the changes in the transient pulse are found for each gate in the critical path until the terminating flip-flop is found. Using the database, the transients amplitude and pulse width will be known when it arrives at the input to a flip-flop.

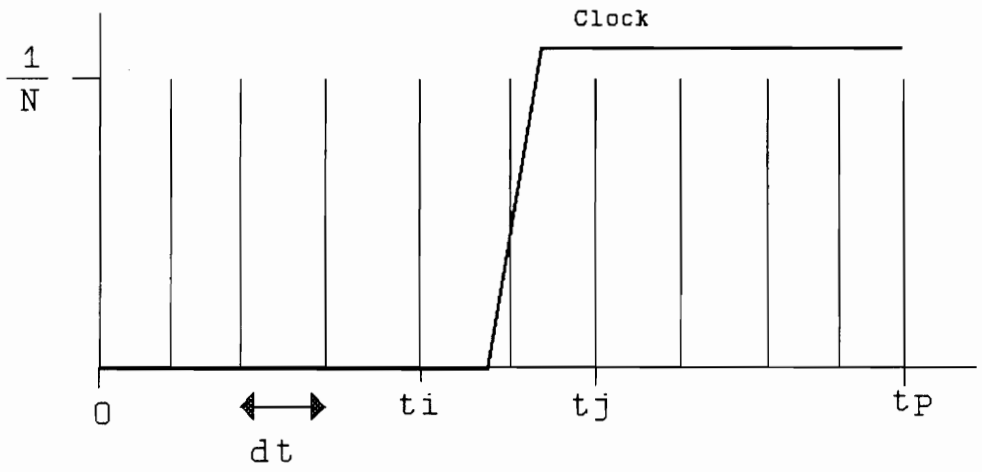
### 4.3.3 PULSE ARRIVAL TIME

As mentioned in Chapter 3, the flip-flop's setup and hold times must be met in order for the data on the input to the flip-flop to be latched properly. Otherwise, the flip-flop might not behave normally. In this section, the probability that the transient pulse will arrive at the flip-flop's input at the proper time with the active edge of the clock to cause an error at the flip-flop's output will be discussed.

A transient pulse might arrive at the data input to a flip-flop at any time during a clock period, but certain arrival times will not coincide with the active edge of the clock. Thus, the transient pulse will not cause an error at the output of the flip-flop. Other arrival times, however, will coincide with the clock and the flip-flop's output to be in error.

Assume, for the moment, that there are only certain times that an upset can arrive and cause an error at the output of an edge-triggered D-type flip-flop. These times are evenly spaced out according to the following interval, dt = tp/n, where tp signifies the clock period and n signifies the number of possible arrival times. If the occurrence of the pulse is completely random, then the probability that the pulse will arrive at any given time is equally possible. Since there are n different arrival times, the probability that the upset will arrive at any one of those times is 1/n. This is illustrated in Figure 25.

During a clock period there are only certain times that an upset can arrive and cause an error at the output of a flip-flop. This is illustrated in the figure by the interval $t_i$ to $t_j$. If the probability of each arrival time within this interval is added together, then the total probability of the upset causing an error at the flip-flop's output can be found. The number of arrival times in the interval from $t_i$ to $t_j$ can be found by dividing $t_j$-$t_i$ by the spacing between arrival times dt = tp/n. This number is then multiplied by the probability of each individual arrival time 1/n to get the probability that the upset will arrive at a time so as to cause an error at the output of the flip-flop.




N = total # of times

dt = tp/N

**Figure 25. Graph of possible transient arrival times.**

This probability is:

$$\frac{(t_j - t_i) * n}{t_p} * \frac{1}{n} = \frac{t_j - t_i}{t_p}$$

Therefore, if the above discussion is extended to include an infinite number of arrival times, then the result is the same as the above equation given as:

$$\lim_{n \to \infty} \frac{t_j - t_i}{t_p} = \frac{t_j - t_i}{t_p}$$

Thus, the probability of latching an SEU is independent of the number of possible arrival times.

From the discussion in Chapter 3 on setup and hold times, remember that if the upset transitions during a setup and hold time then an error occurs on the output of the flip-flop. Also, if the upset meets the setup and hold times, then the upset is latched by the flip-flop. Taking those situations into account the following formula can be used to find the probability that the upset arrives at a time which will cause an error at the output of the flip-flop.

$$P_{arr} = \frac{t_{set} + t_{pw} + t_{hold}}{t_p}$$ for transient pulse widths < tp

or

$P_{arr}$ = 1.0 for transient pulse widths = > tp

Here, $t_{pw}$ is the pulse width of the transient pulse, and $t_{sct}$ and $t_{hold}$ are the setup and hold times of the flip-flop. Shown in Figure 26(a) is the earliest arrival time of a transient pulse that can possibly cause an error at the output of the flip-flop, and shown in Figure 26(b) is the last arrival time of a transient that can possible cause an error at the output of the flip-flop. This interval is the total amount of time during which the flip-flop is susceptible to a transient pulse. This value is given as the numerator in the above equation. By dividing this sum by the clock period, the probability that the transient pulse will arrive at a time during which the flip-flop is susceptible is found.

### 4.3.4 THE TOTAL SOLUTION FOR THIS RESEARCH

From the previous discussions on the probability of latching an SEU, a total solution can be found by combining the methods discussed up to this point. The critical path probability can be found for any node in any circuit by using the probability building method, and if the circuit has simple reconvergence, the critical path probabilities can be found much faster by using the probability propagation

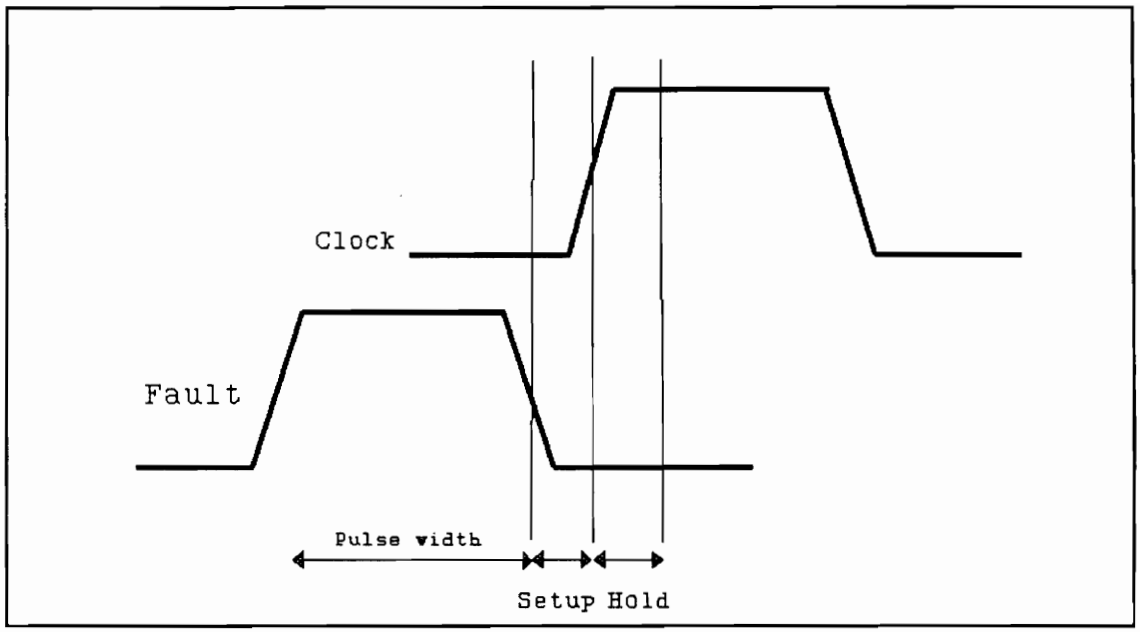



Figure 26(a). Initial arrival time to cause an error.

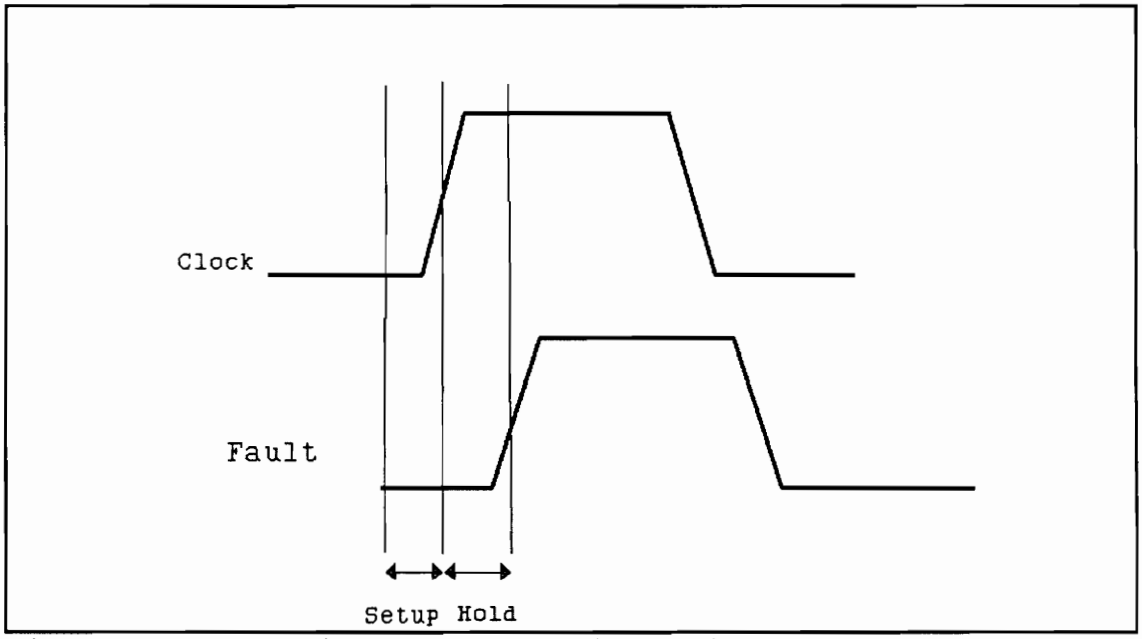



Figure 26(b). Final pulse arrival time to cause an error.

method. The SPICE database discussed in section 4.3.2 can also be used in conjunction with finding the critical path probability to find the resulting pulse width and amplitude at the terminating flip-flop. Finally, the probability that the upset will arrive at a time to causes an error at the output of the flip-flop can then be calculated using the formula in section 4.3.3. Thus, the error probability for a transient pulse is equal to the critical path probability times the arrival time probability discussed in the previous section. If multiple paths exist to different flip-flops then the error probability might need to be calculated differently as was discussed in section 4.3.1.6.

# 5.0 SUPER II - THE SUSCEPTIBILITY UPSET PROGRAM

## 5.1 INTRODUCTION

SUPER II (the **S**usceptibility **UP**s**E**t p**R**ogram) is a digital circuit evaluation program that can determine the node with the highest SEU error probability. To start the evaluation, the user provides a netlist written in the Electronic Design Interchange Format (EDIF) Version 2.0.0 along with information about the radiation incident on the circuit. The program then allows the user to perform a search, in several different ways, for the highest SEU error probability in the circuit. While searching for the highest probability, results may be saved to ASCII type files for review later.

## 5.2 SYSTEM REQUIREMENTS

The program was written in C and runs on an IBM PC. At least 512k of RAM is needed to execute the program, but more might be needed depending on the number of gates in the circuit being evaluated. At present the program is set up to handle 500 gates, 20 fan-out branches per fan-out stem, 20 primary inputs including pseudo-primary inputs from loops found in the

circuit, and 20 primary outputs. For the purposes of this research, the gate selection has been limited to two and three-input NAND and NOR gates, as well as INVERTERS and DFFs.

The program uses a database compiled from SPICE simulations run on MOSIS CMOSN standard gate cells. Workviews Viewlogic schematic capture program was used to draw the circuits used for testing SUPER II, and the CMOSN standard gate cells from MOSIS were used in the drawings. Workview also provides a utility program for exporting the circuit into the EDIF netlist.

### 5.3 OVERVIEW OF THE PROGRAM

In this section each aspect of the program is discussed. The flow of the discussion is the same as that of the program beginning with reading in the EDIF netlist. Next, certain parts of the circuit are eliminated because they are not necessary for the evaluation of the circuit. Next a search is performed for loops in the circuit, and when one is found it is broken and added to the list of primary inputs. Afterwards, the logic level probabilities for each node in the circuit are found using either the probability building or the probability propagation method. Finally, the different ways to search for the node with the highest SEU error probability are used.

### 5.3.1 THE ELECTRONIC DESIGN INTERCHANGE FORMAT

SUPER II begins by reading in the EDIF version 2.0.0 netlist. This version of EDIF was approved March 14, 1988 and is widely used in industry. EDIF began in November 1983 when different manufacturers needed to exchange design information that were stored in different formats. Therefore, a standard was needed so that design information could be easily transferred between many types of design software.

Today, most software packages are capable of producing a wealth of information about a design including, but not limited to, graphical, simulation, netlist, and documentation information. So EDIF was designed to transfer ten different views of a circuit. These views include Behavior, Document, Graphic, Logic model, Mask layout, Netlist, PCB layout, Schematic, Stranger, and Symbolic views. An EDIF file might contain one or more of the different views depending on the intended use of the information. For the purposes of this research the Netlist view is used.

The syntax of EDIF looks somewhat like LISP in that it uses parenthesis to delimit constructs and also uses keywords to identify those constructs. The keywords are used to refer to different aspects of a design. For example, the design might include libraries, cells, nets, etc.

The first block in EDIF is the header which contains information on the version and level of EDIF used. It might also include status information on when the file was written and by what program. Next, a library is specified. The library contains all the information used in the design such as the cells and the technology used. The cells in turn contain all the information about a component. This might include interface port information or particular attributes about a cell such as delay times, rise and fall times, or even voltage characteristics. Instances of other cells might also be made within a cell construct. One feature of EDIF is that a cell must be declared before the cell is first instanced which provides a well organized way to read and parse the EDIF file.

Figure 28 is an EDIF netlist without a header for the circuit in Figure 27 [13]. The example begins with the library construct that contains technology information about the library and one or more cell descriptions. The library begins with a technology section that may include scaling information for a layout or the base number for an element such as capacitance. In the example shown, no technology information is included.

Next, one cell has been included in the library which is named

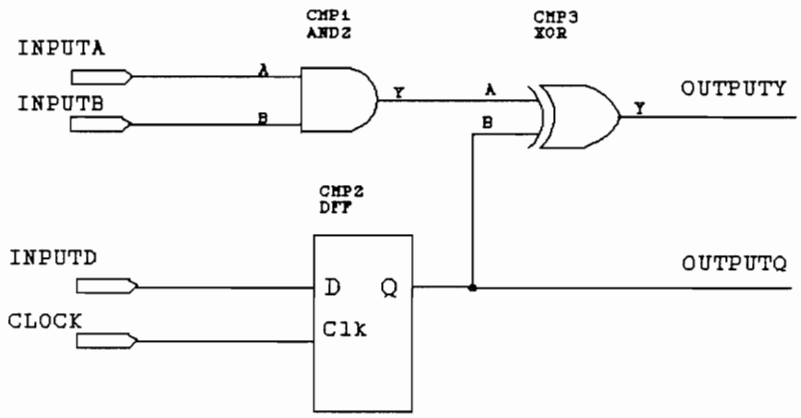



Figure 27. Example circuit for the EDIF netlist in Figure 28.

```
(library mylib (EDIFLevel 0)
  (technology (numberDefinition))
  (cell sample(cellType GENERIC)
    (view netview (viewType NETLIST)
      (interface
        (port INPUTA (direction INPUT))
        (port INPUTB (direction INPUT))
        (port INPUTD (direction INPUT))
        (port CLOCK (direction INPUT))
        (port OUTPUTY (direction INPUT))
        (port OUTPUTQ (direction INPUT))
      )
      (contents
        (instance CMP1
          (viewRef netView
            (cellRef AND2 (libraryRef partsLibrary))
          )
        )
        (instance CMP2
          (viewRef netView
            (cellRef DFF (libraryRef partsLibrary))
          )
        )
        (instance CMP3
          (viewRef netView
            (cellRef XOR (libraryRef partsLibrary))
          )
        )
        (net INPUTA
          (joined
            (portRef A (instanceRef CMP1))
            (portRef INPUTA)
          )
        )
        (net INPUTB
          (joined
            (portRef B (instanceRef CMP1))
            (portRef INPUTB)
          )
        )
        (net INPUTD
          (joined
            (portRef D (instanceRef CMP2))
            (portRef INPUTD)
          )
        )
```

Figure 28. EDIF netlist for Figure 27 (continued).

```
        (net CLOCK
          (joined
            (portRef CLK (instanceRef CMP2))
            (portRef CLOCK)
          )
        )
        (net net001
          (joined
            (portRef Y (instanceRef CMP1))
            (portRef A (instanceRef CMP3))
          )
        )
        (net OUTPUTQ
          (joined
            (portRef Q (instanceRef CMP2))
            (portRef B (instanceRef CMP3))
            (portRef OUTPUTQ)
          )
        )
        (net OUTPUTY
          (joined
            (portRef Y (instanceRef CMP3))
            (portRef OUTPUTY)
          )
        )
      )
    )
  )
```

Figure 28. EDIF netlist for Figure 27.

"sample." This cell, as with all cells, begins with the keyword cellType that describes the use of the cell, and in most cases that type will be GENERIC. The cell view is then given, and the view might be any one of the ten specified earlier.

The cell always includes an interface section and might contain a contents section. The interface section will contain port names and the appropriate direction of data flow for the port. The contents section is further divided into two parts, the instance and net sections. The instance section contains the instance name as well as the cell name of the component, and where to find information on the component. Next, the component instances are connected by the keyword net. Different node names are listed here that are connected to the net, and the components instance name is also given. With this information a complete netlist is formed for the circuit in Figure 28.

### 5.3.2 THE ELIMINATION OF GATES

A typical circuit might appear as the block diagram shown in Figure 29. In this research, the main concern is finding the probability that an SEU causes an error at the output of a flip-flop. Therefore, if a transient occurs in the combinational logic attached to a primary output then the

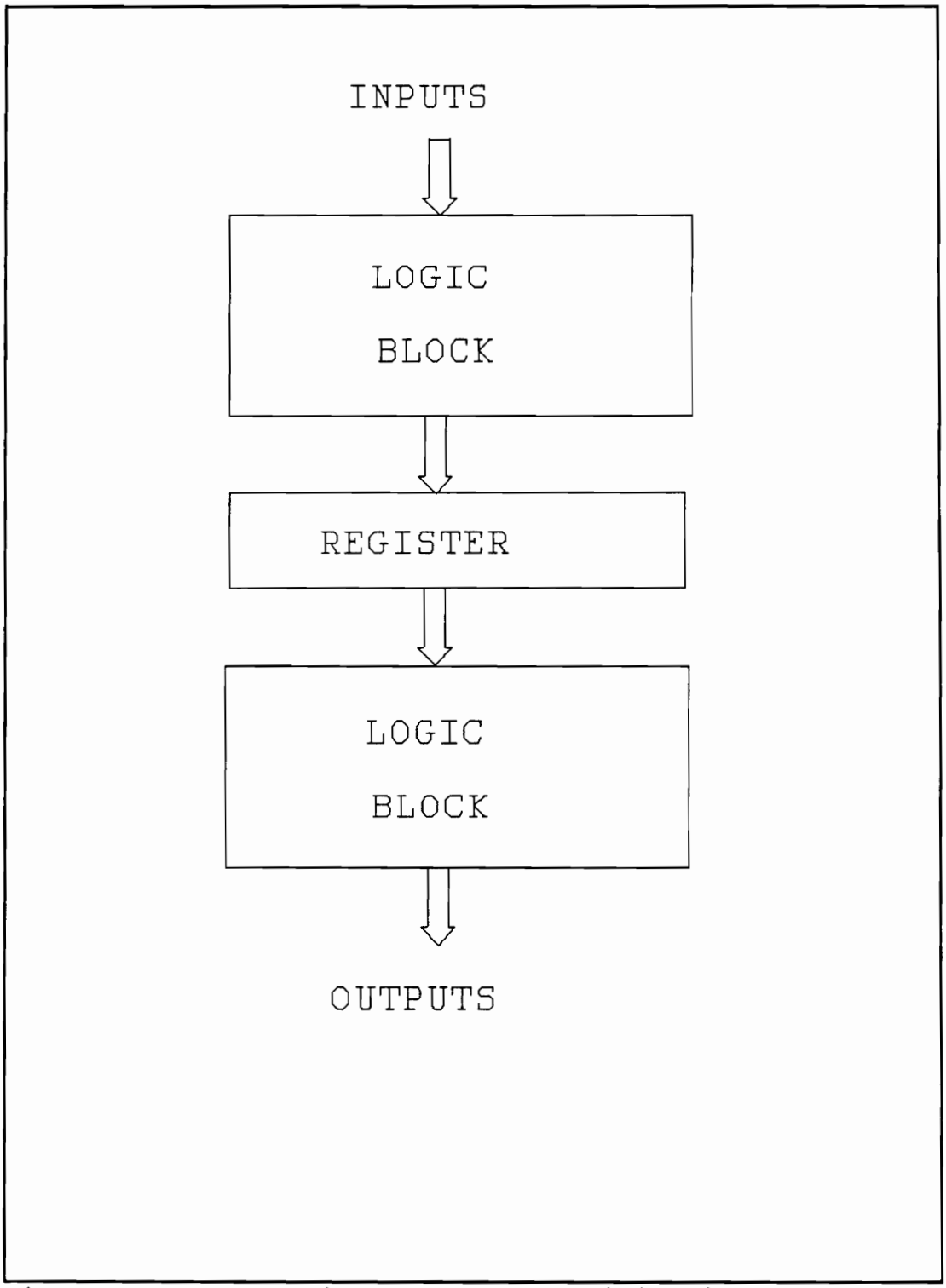



**Figure 29. Block diagram for the elimination of gates.**

upset will not be latched in the circuit. Therefore any combinational logic attached to a primary input will be eliminated from being evaluated.

### 5.3.3 LOOP SEARCH

SUPER II then performs a search for loops in the circuit. The routine assumes that at least one flip-flop exists in all loops. Therefore, the routine begins the loop search at the output of each flip-flop in the circuit. The routine searches out into the circuit until it finds the original flip-flop. The program then breaks the loop at the output of the flip-flop, and transfers the flip-flop's output into a primary input array. The transferred output is now a pseudo-primary input.

### 5.3.4 GENERATING PROBABILITIES

Two methods are used in this program when finding the critical path probability. They are the probability building and the probability propagation methods. The probability building method is straight forward and works for any type of circuit, but it can consume a lot of time for a large number of inputs. Therefore, the probability propagation method is included in SUPER II to speed up the process of finding the internal logic level probabilities.

The probability propagation method provides a fast means for generating node probabilities, but is lacking in one aspect. The method works for all circuits except the ones containing higher forms of reconvergence. This means that if a fan-out point reconverges it must adhere to the following criteria. At the fan-out stem only two fan-out branches are allowed, and only one path may exist from each fan-out branch to the reconvergence point. If other forms of reconvergence are used in the logic circuit, the program will run normally, but will not give exact logic level probabilities for each node in the circuit. Determining if the circuit meets the fan-out restriction is left to the user.

### 5.3.5 PERFORMING THE SEARCH

SUPER II allows the user to perform a search for the node with the highest SEU error probability, and the search may be performed in several ways.

1. According to the nodes level in the logic circuit. The output of the first gate at a flip-flop's input is considered to be the zeroth level, and for each gate closer to the primary inputs, the level count increases by one. The SEU error probabilities for each gate at the specified level can then be found by SUPER II.

2. Given a specific node. A search is performed from the given node to find the probability that the SEU occurring at that node causes an error at the output of a flip-flop.

3. Given a specific node, a backward search can be performed from that node. This allows the user to find the error probability for an SEU that might occur at nodes preceding the specified node.

4. According to the fan-out points found in the circuit. This type of search is included because fan-out in a circuit will, in general, produce higher SEU error probabilities than if no fan-out existed. This higher probability is due to the potentially larger number of flip-flops that a transient might propagate to.

The information generated by the search can then be saved to several ASCII type files depending on the type of search performed.

### 5.4 BASIC PROGRAM OPERATION

When the user loads the SUPER II program, the name of the file containing the EDIF description will be asked for. If the

file exists in the current directory then the full path name does not need to be given. The program then reads in the file and displays the file header information in the EDIF window.

Next, the program jumps to the Search window. Here, the program asks the user if the circuit is restricted. Restricted means that the circuit has simple reconvergence as given by the criteria in section 5.2.4 for probability propagation. The answer should be 'y'es if the circuit does conform to the criteria. Otherwise, the answer should be 'n'o.

If the circuit is not restricted, then a counter is displayed. The left number for the counter is the maximum number of input patterns that need to be applied to the circuit, and the right number tells how many combinations have currently been applied. However, if the circuit is a restricted type circuit, a counter will not be displayed, and the primary input logic level probabilities will be propagated through out the circuit until all of the logic level probabilities are found.

Once finished, the Search window changes. A menu is presented showing the available options for performing the search. There are four basic search options that have already been

discussed in section 5.4, and three of the four options will cause the program to prompt the user for additional information. The fan-out option does not need any additional information to perform the search. However, if the search by level option is chosen the level number is needed. If the search by node or backwards search options are selected then a specific node or gate name is needed. The exit and parameters options are also included in this menu. If information on the incident radiation needs to be changed from the default values, then the parameters option should be selected. This brings up another menu that lists the parameters used for the circuit evaluation, they include the LET of the particle, the pulse width and voltage amplitude of the generated transient, the period of the clock, and whether the transient is a rising or falling pulse. Last, the exit option is available. If this option is chosen, the user is asked to verify that he would like to quit. Verification is required in order to prevent an accidental exit.

Once a search is finished, a list will appear in the Results window. This list contains the type of search performed, the node names searched, and the SEU error probability for each node. At the end of the list is the maximum error probability found during the search.

The results list can then be saved to an ASCII file for later examination, and the filename of the saved information will be the same as the EDIF file. However, the extension for the filename will signify which search option was chosen. For example, if the search was by level for filename EXAM, the results would be stored in a file named EXAM.lvl.

## 5.5 RESULTS OF CIRCUIT EVALUATIONS

In order to test the program, ten circuits were designed that contained loops, fan-out, reconvergent fan-out, and multiple paths. They were designed to be similar to the International Symposium on Circuits and Systems (ISCAS) sequential benchmark circuits used for studying stuck-at faults [14]. The ISCAS circuits are made up of real circuit designs and some synthesized circuits. All of the circuits in the ISCAS set have the flip-flop clock inputs tied to a common clock. So in keeping with the ISCAS benchmark circuits, five of the circuits designed for this research are real circuit designs, and the other five are synthesized. The synthesized circuits were done randomly by hand using Viewlogics Workview program. A common clock is also assumed for the D-type flip-flops.

The circuits designed for evaluation by SUPER II are summarized in Table 14. The first five are real circuit designs:

Table 14. SUPER II test circuits.

| Circuits | Gates | P.I. | P.O. | Loops | P. P. | P. B. | Ratio |
|---|---|---|---|---|---|---|---|
| 7449 | 53 | 5 | 7 | 0 | 0.06 | 1.09 | 18 |
| 7483 | 85 | 9 | 5 | 0 | 0.17 | 19.94 | 117 |
| 74181 | 139 | 14 | 8 | 0 | 0.22 | 1027.50 | 4670 |
| B.S.G. | 78 | 2 | 8 | 4 | 0.16 | 2.42 | 15 |
| Hist | 60 | 4 | 4 | 5 | 0.11 | 13.18 | 120 |
| s100 | 111 | 7 | 5 | 0 | 0.22 | 7.03 | 32 |
| s101 | 170 | 14 | 10 | 0 | 0.27 | 1266.86 | 4692 |
| s200 | 76 | 15 | 8 | 0 | 0.17 | 1107.07 | 6512 |
| s201 | 199 | 7 | 13 | 6 | 0.55 | 754.57 | 1372 |
| s400 | 341 | 7 | 8 | 6 | 1.10 | 1345.89 | 1224 |

1) a 7449 BCD to seven segment display that contains 53 gates, 5 primary inputs and 7 primary outputs.

2) a 7483 4-bit adder with a fast carry that contains 85 gates, 9 primary inputs and 5 primary outputs.

3) a 74181 4-bit ALU that contains 139 gates, 14 primary inputs, and 8 primary outputs.

4) a Bus Signal Generator that contains 78 gates, 2 primary inputs, and 8 primary outputs, and 4 loops.

5) a Histogram generator control unit that contains 60 gates, 4 inputs, 4 primary outputs, and 5 loops.

The next five circuits in Table 14 are synthesized and have the following characteristics:

1) an s100 which contains 111 gates, 7 primary inputs, and 5 primary outputs.

2) an s101 which contains 170 gates, 14 primary inputs, and 10 primary outputs.

3) an s200 which contains 76 gates, 15 primary inputs, and 8 primary outputs, and it also has the simple reconvergence structure discussed in section 5.2.4.

4) an s201 which contains 199 gates, 7 primary inputs, 13 primary outputs, and 6 loops.

5) an s400 which contains 341 gates, 7 primary inputs, 8 primary outputs, and 6 loops.

Each of the circuits in Table 14 have been analyzed successfully using SUPER II, and the test data has been tabulated. In this section, however, the 7449 will be examined in detail, and the test results for the other circuits are found in Appendix A.

A processing time analysis was performed on the probability propagation and probability building methods, but in order to compare the two methods, SUPER II was developed so that the probability propagation method could be used for any circuit. However, the resulting probabilities on each node in the circuit might not be accurate if the circuit does not have a simple reconvergent structure. Also, the tests were performed on an IBM AT compatible running at 10 Mhz. In Table 14 the

processing time for each method, in seconds, is given for each of the test circuits. The ratio of the processing time for each method is also given in Table 14. Notice that the probability building method might take from 15 to 6512 times longer to find the needed logic level probabilities for each node in the circuit. Therefore, the probability propagation method could save a significant amount of processing time when a large number of primary inputs exist for a circuit. Also, notice that as the number of primary inputs to the circuit and the number of loops in the circuit increases, more time is required for the probability building method to find the needed logic value probabilities in the circuit.

For the 7449 circuit, several tests were run to show that some SEU error probability features that seem intuitive are indeed true. Also, some features were found by the tests that were not intuitive. All of the results from the test were taken with the clock speed set to 50 Mhz which gives a 20 nanosecond(ns) pulse width. This clock speed was chosen because current trends in logic circuit design are at that speed.

Shown in Table 15 are the SEU error probabilities and level for each gate in the circuit. Using this table a comparison was performed for variations in a transient's pulse width. In

Table 15. Table showing the latching probabilities when the pulse width of the transient is varied.

| Gate | Level | Pulse Width(ns) | | |
|---|---|---|---|---|
| | | 1 | 5 | 9 |
| INV5: | | 20 | 40 | 60 |
| INV6: | | 20 | 40 | 60 |
| INV7: | | 20 | 40 | 60 |
| INV8: | 0 | 20 | 40 | 60 |
| INV9: | | 20 | 40 | 60 |
| INV10: | | 20 | 40 | 60 |
| INV11: | | 20 | 40 | 60 |
| | | | | |
| ND24: | | 19 | 38.5 | 58 |
| ND26: | | 19 | 38.5 | 58 |
| ND28: | | 19 | 38.5 | 58 |
| ND22: | 1 | 19 | 38.5 | 58 |
| ND23: | | 19 | 38.5 | 58 |
| ND25: | | 19 | 38.5 | 58 |
| ND27: | | 19 | 38.5 | 58 |
| | | | | |
| ND6: | | 0 | 12.49 | 16.3 |
| ND8: | | 0 | 19.38 | 25.29 |
| ND17: | | 0 | 20.82 | 27.17 |
| ND18: | | 0 | 12.49 | 16.3 |
| ND7: | | 0 | 9.61 | 12.54 |
| ND9: | 2 | 0 | 10.57 | 13.79 |
| ND10: | | 0 | 10.57 | 13.79 |
| ND13: | | 0 | 14.57 | 19.02 |
| ND14: | | 0 | 14.57 | 19.02 |
| ND15: | | 0 | 27.07 | 35.32 |
| ND19: | | 0 | 9.61 | 12.54 |
| ND20: | | 0 | 36.75 | 46.81 |

Table 15 (continued). Table showing the latching probabilities when the pulse width of the transient is varied.

| Gate | Level | Pulse Width(ns) | | |
|---|---|---|---|---|
| | | 1 | 5 | 9 |
| ND11: | | 0 | 34.12 | 43.47 |
| ND12: | | 0 | 15.75 | 20.06 |
| ND16: | 2 | 0 | 10.5 | 13.37 |
| ND21: | | 0 | 18.37 | 23.41 |
| INV13: | | 0 | 26.25 | 33.44 |
| ND5: | | 0 | 20.82 | 27.17 |
| | | | | |
| INV1: | | 0 | 38.3 | 55.89 |
| INV2: | | 0 | 41.68 | 53.01 |
| INV3: | | 0 | 33.36 | 47.16 |
| INV4: | | 0 | 32.97 | 39.21 |
| ND1: | 3 | 0 | 61.22 | 73.01 |
| ND2: | | 0 | 54.22 | 71.66 |
| ND3: | | 0 | 55.59 | 69.87 |
| ND4: | | 0 | 41.35 | 57.9 |

order to do the comparison the transient's amplitude was held at 5 volts, and the pulse width was varied from 1ns to 9ns at 4ns increments.

Notice that as the pulse width of the initial transient increases the pulse width of the arriving transient at a flip-flop also increases. Recall that the equation from section 4.3.3 for calculating a transient's arrival time probability includes the pulse width for the transient in the numerator. Using this equation, it can be seen that as the arrival pulse width increases, the arrival time probability also increases. This in turn increases the error probability of the SEU.

Also, notice in Table 15 that gates at the same level do not necessarily have similar error probabilities. Gate ND17, at level 2 and a 5ns pulse width, has a 20.82 percent error probability, and gate ND7, for the same level and pulse width, has a 9.61 percent error probability. This is due to several reasons. The critical path probabilities could be different for each gate, or the transient pulse for one node might be modified differently when propagating through the logic than a pulse originating from another node. The pulse originating at one node might also propagate to more flip-flops than a pulse originating at another node which increases the SEU error probability.

In general, as the level of the node in the circuit increases the lower the error probabilities. This is primarily due to the lower probabilities found for setting up the critical path when the node is further from a flip-flop. The exception to this is generally due to fan-out. Recall from section 4.3.1.6 that when multiple paths exist for a transient pulse, the number of possible flip-flops that a pulse can propagate to might increase. Multiple paths, therefore, tend to increase the SEU error probability in general. A designer can never be sure that a transient occurring at a certain level will not have a higher error probability at a higher level. In circuit 7449 all of the nodes at level 3 are fan-out nodes, and notice that their SEU error probabilities are generally higher than the ones at level 2.

Finally, the error probability for a transient might actually decrease when the initial transient's pulse width increases. Circuit 7449 does not show this phenomena, but several circuits in Appendix A do show it, such as circuit BSG. The above stated phenomena is due to changes in the transient's pulse width as it propagates through the logic, and the database of SPICE simulations accessed by SUPER II reflects those changes. For certain gates when the input pulse width increases the resulting output transient pulse width will decrease slightly. This change is then propagated down the

critical path where other gates might again decrease or increase the pulse width of the transient.

Next, Table 16 is used to compare the effect of variations of the amplitude of the transient pulse on the SEU error probability. The circuit is tested with the pulse width held at 5ns, and the voltage amplitude of the transient is taken to be 2 volts, 4 volts, and 5 volts. From this table, the SEU error probabilities generally increase as the voltage amplitude of the transient increases. This is primarily due to the pulse width of the transient pulse increasing as the amplitude increases. However, variations can exist that will give lower probabilities for higher transient amplitudes. For circuit 7449, that variation is not apparent. Again, this variation is due to the modifications made to the transient pulse as it propagates through the logic.

Next, a comparison is done for a rising and a falling transient pulse using Table 17. The rising transient pulse was held at a voltage amplitude of 5 volts and a pulse width of 5ns, and the amplitude of the falling pulse was held at 0 volts and a pulse width of 5ns. From Table 17, the error probabilities for either type of pulse does not change very much. Therefore, the MOSIS gates used in the design give similar changes in the transient pulse as it propagates

Table 16. Table showing the latching probabilities when the amplitude of the transient is varied.

| Gate | Level | Voltage Amplitude(volts) | | |
|---|---|---|---|---|
| | | 3 | 4 | 5 |
| INV5: | | 40 | 40 | 40 |
| INV6: | | 40 | 40 | 40 |
| INV7: | | 40 | 40 | 40 |
| INV8: | 0 | 40 | 40 | 40 |
| INV9: | | 40 | 40 | 40 |
| INV10: | | 40 | 40 | 40 |
| INV11: | | 40 | 40 | 40 |
| | | | | |
| ND24: | | 27 | 36 | 38.5 |
| ND26: | | 27 | 36 | 38.5 |
| ND28: | | 27 | 36 | 38.5 |
| ND22: | 1 | 27 | 36 | 38.5 |
| ND23: | | 27 | 36 | 38.5 |
| ND25: | | 27 | 36 | 38.5 |
| ND27: | | 27 | 36 | 38.5 |
| | | | | |
| ND6: | | 9.9 | 10.21 | 12.49 |
| ND8: | | 15.36 | 15.83 | 19.38 |
| ND17: | | 16.5 | 17.01 | 20.82 |
| ND18: | | 9.9 | 10.21 | 12.49 |
| ND7: | | 7.62 | 7.85 | 9.61 |
| ND9: | 2 | 8.38 | 8.64 | 10.57 |
| ND10: | | 8.38 | 8.64 | 10.57 |
| ND13: | | 11.55 | 11.91 | 14.57 |
| ND14: | | 11.55 | 11.91 | 14.57 |
| ND15: | | 21.46 | 22.12 | 27.07 |
| ND19: | | 7.62 | 7.85 | 9.61 |
| ND20: | | 0 | 23.62 | 36.75 |

Table 16 (continued). Table showing the latching probabilities when the amplitude of the transient is varied.

| | | | | |
|---|---|---|---|---|
| ND11: | | 0 | 21.94 | 34.12 |
| ND12: | | 0 | 10.13 | 15.75 |
| ND16: | 2 | 0 | 6.75 | 10.5 |
| ND21: | | 0 | 11.81 | 18.37 |
| INV13: | | 0 | 16.88 | 26.25 |
| ND5: | | 16.5 | 17.01 | 20.82 |
| | | | | |
| INV1: | | 0 | 13.45 | 38.3 |
| INV2: | | 0 | 9.97 | 41.68 |
| INV3: | | 0 | 14.69 | 33.36 |
| INV4: | | 0 | 6.87 | 32.97 |
| ND1: | 3 | 0 | 53.25 | 61.22 |
| ND2: | | 0 | 39.98 | 54.22 |
| ND3: | | 0 | 26.5 | 55.59 |
| ND4: | | 0 | 13.14 | 41.35 |

Table 17. Table showing the latching probabilities when the transient is either rising or falling.

| Gate | Level | R | F |
|---|---|---|---|
| | | | |
| INV5: | | 40 | 40 |
| INV6: | | 40 | 40 |
| INV7: | | 40 | 40 |
| INV8: | 0 | 40 | 40 |
| INV9: | | 40 | 40 |
| INV10: | | 40 | 40 |
| INV11: | | 40 | 40 |
| | | | |
| ND24: | | 38.5 | 42 |
| ND26: | | 38.5 | 42 |
| ND28: | | 38.5 | 42 |
| ND22: | 1 | 38.5 | 42 |
| ND23: | | 38.5 | 42 |
| ND25: | | 38.5 | 42 |
| ND27: | | 38.5 | 42 |
| | | | |
| ND6: | | 12.49 | 12.19 |
| ND8: | | 19.38 | 18.91 |
| ND17: | | 20.82 | 20.31 |
| ND18: | | 12.49 | 12.19 |
| ND7: | | 9.61 | 9.38 |
| ND9: | 2 | 10.57 | 10.31 |
| ND10: | | 10.57 | 10.31 |
| ND13: | | 14.57 | 14.22 |
| ND14: | | 14.57 | 14.22 |
| ND15: | | 27.07 | 26.41 |
| ND19: | | 9.61 | 9.38 |
| ND20: | | 36.75 | 45.94 |

Table 17 (continued). Table showing the latching probabilities when the transient is either rising or falling.

| | | | |
|---|---|---|---|
| ND11: | | 34.12 | 42.66 |
| ND12: | | 15.75 | 19.69 |
| ND16: | | 10.5 | 13.12 |
| ND21: | 2 | 18.37 | 22.97 |
| INV13: | | 26.25 | 32.81 |
| ND5: | | 20.82 | 20.31 |
| | | | |
| INV1: | | 38.3 | 56.17 |
| INV2: | | 41.68 | 51.32 |
| INV3: | | 33.36 | 48.88 |
| INV4: | | 32.97 | 34.64 |
| ND1: | 3 | 61.22 | 58.54 |
| ND2: | | 54.22 | 63.96 |
| ND3: | | 55.59 | 57.62 |
| ND4: | | 41.35 | 54.35 |

through the logic.

The s200 circuit has a simple reconvergent structure which recall from Chapter 4 is needed for the probability propagation method to give accurate critical path probabilities. A brief test was done on circuit s200 to find out if the probability propagation method gives the same logic value probabilities as the probability building method. The test was run with a rising 5ns 5 volt transient pulse, and as shown in Table 18, the error probabilities are the same as expected.

From the previous discussion for circuit 7449, there are several generalities that can be formed. First, the error probabilities decrease as the level of the gate in the circuit increases, and also, as the pulse width and amplitude of the transient pulse increases, the error probability increases. Variations on those generalities occur when fan-out in a circuit gives several paths to different flip-flops which increases the SEU error probability. Also, variations might occur due to the changes in the transients amplitude and pulse width as it propagates through the gates to the input of a flip-flop.

In conclusion, exactly where in the circuit the nodes with the

Table 18. Table for the s200 comparing the probability building and the probability propagation methods.

| Gates | Level | P. B. | P. P. |
|---|---|---|---|
| ND6: | | 60 | 60 |
| NR8: | | 60 | 60 |
| ND26: | | 60 | 60 |
| NR12: | 0 | 60 | 60 |
| ND51: | | 60 | 60 |
| NR16: | | 60 | 60 |
| ND43: | | 60 | 60 |
| NR4: | | 60 | 60 |
| | | | |
| NR3: | | 40.23 | 40.23 |
| ND5: | | 5.98 | 5.98 |
| ND21: | | 30.53 | 30.53 |
| ND20: | | 13.67 | 13.67 |
| NR7: | | 40.23 | 40.23 |
| ND25: | | 5.98 | 5.98 |
| ND37: | | 30.53 | 30.53 |
| ND36: | 1 | 13.67 | 13.67 |
| NR11: | | 40.23 | 40.23 |
| ND52: | | 5.98 | 5.98 |
| ND49: | | 30.53 | 30.53 |
| ND50: | | 13.67 | 13.67 |
| NR14: | | 40.23 | 40.23 |
| ND42: | | 5.98 | 5.98 |
| ND12: | 1 | 30.53 | 30.53 |

Table 18 (continued). Table for s200 comparing the probability building and the probability propagation methods.

| | | | |
|---|---|---|---|
| ND13: | | 13.67 | 13.67 |
| | | | |
| NR2: | | 3.74 | 3.74 |
| ND24: | | 13.75 | 13.75 |
| ND19: | | 18.61 | 18.61 |
| ND18: | | 13.22 | 13.22 |
| NR6: | | 3.74 | 3.74 |
| ND30: | | 13.75 | 13.75 |
| ND35: | | 18.61 | 18.61 |
| ND31: | 2 | 13.22 | 13.22 |
| NR10: | | 3.74 | 3.74 |
| ND41: | | 13.75 | 13.75 |
| ND48: | | 18.61 | 18.61 |
| ND45: | | 13.22 | 13.22 |
| NR15: | | 3.74 | 3.74 |
| ND10: | | 18.61 | 18.61 |
| ND11: | | 13.22 | 13.22 |
| ND4: | | 13.75 | 13.75 |
| | | | |
| NR5: | | 1.5 | 1.5 |
| ND15: | 3 | 7.09 | 7.09 |
| ND16: | | 14.91 | 14.91 |
| ND17: | | 14.91 | 14.91 |
| ND23: | | 6.88 | 6.88 |
| ND22: | | 6.88 | 6.88 |

highest SEU error probabilities exist can not be known intuitively. When nodes are found that have probabilities higher than a designer would like to have, several modifications might possibly be done on the circuit to decrease the error probability. If the number of inputs to each gate in the circuit is increased, the critical path probability would decrease, and if fan-out is kept to a minimum, the number of possible paths to different flip-flops might be decreased. The circuit might also be run at a slower clock speed in order to decrease the arrival probability of the transient pulse. Finally, the transistors in the gates might be modified by increasing the drain capacitance or changing the technology to lower the susceptibility of the gate to a transient upset by a cosmic particle.

# 6.0 CONCLUSIONS AND FUTURE WORK

The purpose of this research was to attempt to find a solution to finding the probability that an SEU will cause an error at the output of a flip-flop in any CMOS combinational logic circuit. The solution for this research was based on a method developed by Diehl et al with several modifications. The solution for this research was then implemented in a program called SUPER II that can determine the nodes with the highest SEU error probability. To date, SUPER II is believed to be the only program of its kind to evaluate combinational logic circuits for SEU latching probabilities.

Again, the basic approach used in this research was derived from a method given by Diehl et al which made many assumptions that were not good or did not consider the effects that can occur to transients as they propagate through the circuit. Each of the criteria for latching an SEU were explored, and some original solutions were found for each criterion. The probability building and the probability propagation methods were presented, and an arrival time probability equation was

derived which accounts for the setup and hold times of a flip-flop. 2700 SPICE simulations were performed on the 2 and 3-input NAND, NOR gates, and the INVERTER CMOSN standard cells from MOSIS to account for transient pulse variations while propagating down a critical path. Also, such circuit structures as multiple paths and loops are accounted for in the total solution.

The program SUPER II was developed which reads in a circuit netlist in the EDIF format. The program then allows the user to perform several different searches for the node with the highest SEU error probability. Ten testbench circuits were developed to test SUPER II, and to explore the behavior of the transient pulse as it propagates through the combinational logic. The results were as expected with one exception. The variations made to the transient pulse as it propagates through the logic were shown to not be negligible as was implied in the Diehl et al method.

Future work on the development of SUPER II could include increasing the number of different type gates used from the MOSIS CMOSN standard cell library. Also, the number of gates, primary inputs, and primary outputs could be increased in order for the program to evaluate larger circuits. A graphical and a mouse interface might also be helpful, so that

a user could click the mouse on a particular node and get the resulting latching probabilities. Since WorkView includes an export facility for the EDIF schematic view, SUPER II might be adapted to the graphical interface. Also, if the designer knows which input patterns are likely to be applied to the circuit when the transient fault occurs, then those patterns could be entered into SUPER II to give more accurate internal logic level probabilities.

As shown in Chapter 5, the probability propagation method showed a lot of promise for speeding up the process of finding the needed logic value probabilities, so future research in this area might be fruitful. The solution found in this research does not consider the effect of a transient pulse reconverging on itself at some gate in the circuit. This effect would cause the transient to either not propagate through the gate or when it does propagate through the gate, the resulting pulse width might be either shorter or longer than the original pulses. Also, an equation is needed to find the pulse width of the transient pulse when just the LET of a particle is given.

# APPENDIX A

# TEST RESULTS

Table 19. Circuit 7483 error probabilities.

| Gates | Level | Pulse Width(ns) | | | Voltage Amplitude | | |
|---|---|---|---|---|---|---|---|
| | | 1 | 5 | 9 | 3 | 4 | 5 |
| INV7: | | 20 | 40 | 60 | 40 | 40 | 40 |
| ND19: | | 20 | 40 | 60 | 40 | 40 | 40 |
| ND22: | 0 | 20 | 40 | 60 | 40 | 40 | 40 |
| ND25: | | 20 | 40 | 60 | 40 | 40 | 40 |
| ND28: | | 20 | 40 | 60 | 40 | 40 | 40 |
| | | | | | | | |
| ND15: | | 19 | 38.5 | 58 | 27 | 36 | 38.5 |
| ND17: | | 0 | 23.25 | 37.5 | 15.38 | 21 | 23.25 |
| ND18: | | 0 | 23.25 | 37.5 | 15.38 | 21 | 23.25 |
| ND20: | 1 | 0 | 23.25 | 37.5 | 15.38 | 21 | 23.25 |
| ND21: | | 0 | 23.25 | 37.5 | 15.38 | 21 | 23.25 |
| ND23: | | 0 | 31 | 50 | 20.5 | 28 | 31 |
| ND24: | | 0 | 15.5 | 25 | 10.25 | 14 | 15.5 |
| ND27: | | 0 | 23.25 | 37.5 | 15.38 | 21 | 23.25 |
| ND26: | | 0 | 23.25 | 37.5 | 15.38 | 21 | 23.25 |
| | | | | | | | |
| NR9: | | 0 | 32.09 | 41.87 | 25.43 | 26.22 | 32.09 |
| INV21: | | 0 | 16.5 | 23.44 | 0 | 16.31 | 16.5 |
| INV5: | | 0 | 19.41 | 25.33 | 15.39 | 15.86 | 19.41 |
| INV6: | | 0 | 21.44 | 27.98 | 17 | 17.52 | 21.44 |
| NR15: | 2 | 0 | 16.5 | 23.44 | 0 | 16.31 | 16.5 |
| NR10: | | 0 | 16.5 | 23.44 | 0 | 16.31 | 16.5 |
| NR11: | | 0 | 21.75 | 25 | 0 | 21.75 | 21.75 |
| NR14: | | 0 | 16.5 | 23.44 | 0 | 16.31 | 16.5 |
| ND16: | | 0 | 16.5 | 23.44 | 0 | 16.31 | 16.5 |

Table 19 (continued). Circuit 7483 error probabilities.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| NR12: | 2 | 0 | 16.5 | 23.44 | 0 | 16.31 | 16.5 |
| NR13: | | 0 | 44 | 62.5 | 0 | 30 | 44 |
| INV15: | | 0 | 16.5 | 23.44 | 0 | 11.25 | 16.5 |
| INV14: | | 0 | 16.5 | 23.44 | 0 | 11.25 | 16.5 |
| INV28: | | 0 | 16.5 | 23.44 | 0 | 11.25 | 16.5 |
| INV27: | | 0 | 16.5 | 23.44 | 0 | 11.25 | 16.5 |
| INV26: | | 0 | 22 | 31.25 | 0 | 15 | 22 |
| INV25: | | 0 | 0 | 0 | 0 | 0 | 0 |
| INV23: | | 0 | 16.5 | 23.44 | 0 | 11.25 | 16.5 |
| INV24: | | 0 | 16.5 | 23.44 | 0 | 11.25 | 16.5 |
| | | | | | | | |
| NR5: | | 0 | 18.94 | 27.46 | 8.76 | 18.94 | 18.94 |
| NR6: | | 0 | 20.92 | 30.33 | 9.67 | 20.92 | 20.92 |
| INV3: | | 0 | 14.31 | 25.04 | 0 | 8.35 | 14.31 |
| INV2: | | 0 | 15.13 | 26.48 | 0 | 8.83 | 15.13 |
| INV10: | | 0 | 12.8 | 15.47 | 0 | 0 | 12.8 |
| ND3: | | 0 | 21.48 | 25.72 | 0 | 10.08 | 21.48 |
| INV17: | | 0 | 12.8 | 15.47 | 0 | 0 | 12.8 |
| INV18: | | 0 | 9.2 | 11.12 | 0 | 5.66 | 9.2 |
| INV19: | 3 | 0 | 10.4 | 12.57 | 0 | 6.4 | 10.4 |
| INV29: | | 0 | 25.5 | 37.5 | 0 | 0 | 25.5 |
| INV20: | | 0 | 5.69 | 6.88 | 0 | 0 | 5.69 |
| INV22: | | 0 | 12.8 | 15.47 | 0 | 0 | 12.8 |
| INV13: | | 0 | 9.6 | 11.6 | 0 | 0 | 9.6 |
| NR1: | | 0 | 27.09 | 39.72 | 0 | 9.56 | 27.09 |
| NR7: | | 0 | 13.46 | 16.27 | 0 | 0 | 13.46 |
| NR2: | | 0 | 35.2 | 42.54 | 0 | 8.99 | 35.2 |

Table 19 (continued). Circuit 7483 error probabilities.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| NR3: | | 0 | 31.41 | 36.6 | 0 | 12 | 31.41 |
| NR4: | 5 | 0 | 44.72 | 53.96 | 0 | 5.33 | 44.72 |
| INV1: | | 0 | 46.37 | 54.12 | 0 | 28.69 | 46.37 |
| | | | | | | | |
| ND1: | | 0 | 22.45 | 28.49 | 0 | 16.97 | 22.45 |
| ND5: | | 0 | 20.43 | 26.99 | 0 | 20.43 | 20.43 |
| INV4: | | 0 | 2.31 | 2.37 | 0 | 0 | 2.31 |
| ND6: | | 0 | 19.31 | 25.51 | 0 | 19.31 | 19.31 |
| ND7: | | 0 | 18.47 | 19 | 0 | 0 | 18.47 |
| ND8: | | 0 | 10.2 | 10.49 | 0 | 5.39 | 10.2 |
| INV8: | | 0 | 5.74 | 5.9 | 0 | 3.03 | 5.74 |
| INV9: | 4 | 0 | 8.6 | 8.85 | 0 | 4.55 | 8.6 |
| ND2: | | 0 | 23.26 | 27.93 | 0 | 18.23 | 23.26 |
| INV11: | | 0 | 10.01 | 11.92 | 0 | 8.3 | 10.01 |
| INV12: | | 0 | 9.47 | 11.27 | 0 | 7.84 | 9.47 |
| NR8: | | 0 | 11.6 | 13.18 | 0 | 11.6 | 11.6 |
| ND12: | | 0 | 8.79 | 12.63 | 0 | 8.79 | 8.79 |
| ND4: | | 0 | 36.29 | 40.76 | 0 | 23.14 | 36.29 |
| ND13: | | 0 | 9.94 | 14.28 | 0 | 9.94 | 9.94 |
| ND14: | | 0 | 41.25 | 46.88 | 0 | 41.25 | 41.25 |
| | | | | | | | |
| ND9: | | 0 | 3.81 | 10.49 | 0 | 3.81 | 3.81 |
| ND10: | 5 | 0 | 3.61 | 9.92 | 0 | 3.61 | 3.61 |
| ND11: | | 0 | 4.64 | 5.54 | 0 | 0 | 4.64 |
| INV16: | | 0 | 3.92 | 4.67 | 0 | 0 | 3.92 |

Table 20. Circuit 74181 error probabilities.

| Gates | Level | Pulse Width(ns) | | | Voltage Amplitude | | |
|---|---|---|---|---|---|---|---|
| | | 1 | 5 | 9 | 3 | 4 | 5 |
| | | | | | | | |
| ND69: | | 20 | 44.45 | 64.04 | 42.54 | 44.45 | 44.45 |
| ND66: | | 20 | 43.13 | 62.71 | 40 | 42.31 | 43.13 |
| ND62: | | 20 | 43.02 | 62.62 | 40 | 42.23 | 43.02 |
| ND59: | 0 | 20 | 41.77 | 61.54 | 40 | 41.31 | 41.77 |
| NR2: | | 20 | 57.46 | 78.77 | 51.54 | 55.77 | 57.46 |
| INV29: | | 20 | 40 | 60 | 40 | 40 | 40 |
| ND56: | | 20 | 40 | 60 | 40 | 40 | 40 |
| NR8: | | 20 | 40 | 60 | 40 | 40 | 40 |
| | | | | | | | |
| ND68: | | 0 | 28.11 | 43.59 | 16.66 | 22.75 | 28.11 |
| ND67: | | 0 | 28.11 | 43.59 | 16.66 | 22.75 | 28.11 |
| ND64: | | 0 | 25.89 | 41.51 | 16.1 | 21.98 | 25.89 |
| ND65: | | 0 | 26.9 | 43.1 | 16.74 | 22.86 | 26.9 |
| ND60: | | 0 | 25.33 | 40.64 | 15.78 | 21.55 | 25.33 |
| ND61: | 1 | 0 | 26.91 | 43.12 | 16.78 | 22.91 | 26.91 |
| ND57: | | 0 | 10.03 | 16.22 | 6.36 | 8.69 | 10.03 |
| ND58: | | 0 | 31.63 | 50.72 | 20.24 | 27.65 | 31.63 |
| ND54: | | 0 | 17.27 | 25.33 | 5.12 | 13.55 | 17.27 |
| INV22: | | 0 | 27.98 | 40.29 | 8.57 | 22.09 | 27.98 |
| NR4: | | 19 | 38.5 | 58 | 27 | 36 | 38.5 |
| INV27: | | 0 | 4.36 | 7.03 | 2.88 | 3.94 | 4.36 |
| ND63: | | 0 | 11.62 | 18.75 | 7.69 | 10.5 | 11.62 |
| INV43: | | 0 | 4.38 | 7.07 | 2.9 | 3.96 | 4.38 |
| | | | | | | | |

Table 20 (continued). Circuit 74181 error probabilities.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| ND41: | | 0 | 33.16 | 39.66 | 0 | 30.59 | 33.16 |
| ND34: | | 0 | 33.73 | 40.16 | 0 | 31.36 | 33.73 |
| ND27: | | 0 | 30 | 34.45 | 0 | 28.88 | 30 |
| ND28: | | 0 | 30 | 34.45 | 0 | 28.88 | 30 |
| INV41: | | 0 | 30.28 | 41.9 | 0 | 18.28 | 30.28 |
| INV42: | | 0 | 30.28 | 41.9 | 0 | 18.28 | 30.28 |
| INV40: | | 0 | 9.7 | 13.44 | 0 | 5.89 | 9.7 |
| INV39: | | 0 | 27.34 | 37.75 | 0 | 16.93 | 27.34 |
| INV37: | | 0 | 9.47 | 13.13 | 0 | 5.77 | 9.47 |
| INV35: | | 0 | 26.09 | 36.07 | 0 | 16.19 | 26.09 |
| INV44: | 2 | 0 | 34.26 | 47.82 | 0 | 22.22 | 34.26 |
| INV45: | | 0 | 1.07 | 1.5 | 0 | 0.68 | 1.07 |
| ND31: | | 0 | 29.59 | 40.83 | 0 | 25.94 | 29.59 |
| ND37: | | 0 | 29.57 | 40.84 | 0 | 25.81 | 29.57 |
| ND53: | | 0 | 14.28 | 18.56 | 0 | 10.71 | 14.28 |
| ND52: | | 0 | 9.05 | 11.82 | 0 | 6.76 | 9.05 |
| INV24: | | 0 | 8.4 | 10.97 | 0 | 6.27 | 8.4 |
| NR9: | | 0 | 35.65 | 41.27 | 0 | 34.93 | 35.65 |
| INV28: | | 0 | 6.23 | 7.94 | 0 | 4.01 | 6.23 |
| ND50: | | 0 | 31.5 | 40.13 | 0 | 20.25 | 31.5 |
| NR3: | | 0 | 7.38 | 10.05 | 4.22 | 7.38 | 7.38 |
| INV3: | | 0 | 54.69 | 63.29 | 12.54 | 52.85 | 54.69 |
| ND49: | | 0 | 42.13 | 58.74 | 0 | 28.22 | 42.13 |
| | | | | | | | |
| ND39: | | 0 | 29.36 | 36.02 | 0 | 0 | 29.36 |
| ND33: | 3 | 0 | 13.48 | 16.71 | 0 | 0 | 13.48 |
| ND32: | | 0 | 30.09 | 36.76 | 0 | 0 | 30.09 |

Table 20 (continued). Circuit 74181 error probabilities.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| ND25: | | 0 | 12.5 | 14.81 | 0 | 0 | 12.5 |
| ND26: | | 0 | 28.39 | 33.52 | 0 | 0 | 28.39 |
| ND29: | | 0 | 18.04 | 26.64 | 0 | 0 | 18.04 |
| ND30: | | 0 | 19.4 | 28.64 | 0 | 0 | 19.4 |
| ND35: | | 0 | 19.01 | 25.33 | 0 | 13.26 | 19.01 |
| ND36: | | 0 | 18.36 | 24.48 | 0 | 12.81 | 18.36 |
| INV38: | | 0 | 19.52 | 26.02 | 0 | 13.63 | 19.52 |
| NR1: | | 0 | 6.29 | 10.9 | 0 | 6.29 | 6.29 |
| ND44: | | 0 | 4.78 | 5.81 | 0 | 0 | 4.78 |
| INV32: | | 0 | 29.42 | 35.46 | 0 | 0 | 29.42 |
| ND51: | | 0 | 1.79 | 2.29 | 0 | 1.27 | 1.79 |
| INV25: | | 0 | 0.53 | 0.68 | 0 | 0.38 | 0.53 |
| INV26: | | 0 | 0.35 | 0.45 | 0 | 0.25 | 0.35 |
| INV13: | 3 | 0 | 39.7 | 47.99 | 0 | 10.59 | 39.7 |
| ND6: | | 0 | 54.68 | 65.14 | 0 | 54.68 | 54.68 |
| INV2: | | 0 | 39.61 | 51.37 | 0 | 10.3 | 39.61 |
| INV4: | | 0 | 25.84 | 39.14 | 0 | 8.21 | 25.84 |
| INV5: | | 0 | 22.3 | 28.07 | 0 | 26.1 | 22.3 |
| INV8: | | 0 | 41.52 | 43.55 | 0 | 20.67 | 41.52 |
| INV9: | | 0 | 38.91 | 51.51 | 0 | 37.82 | 38.91 |
| INV10: | | 0 | 38.34 | 43.03 | 0 | 16.33 | 38.34 |
| ND48: | | 0 | 12.14 | 17.93 | 0 | 0 | 12.14 |
| ND47: | | 0 | 28.14 | 41.35 | 0 | 0 | 28.14 |
| ND40: | | 0 | 13.16 | 16.37 | 0 | 0 | 13.16 |
| | | | | | | | |
| ND4: | 4 | 0 | 24.55 | 27.91 | 0 | 21.27 | 24.55 |
| INV19: | | 0 | 3.41 | 4.15 | 0 | 0 | 3.41 |

Table 20 (continued). Circuit 74181 error probabilities.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| INV16: | | 0 | 23.04 | 27.48 | 0 | 0 | 23.04 |
| INV17: | | 0 | 3.47 | 4.19 | 0 | 0 | 3.47 |
| INV15: | | 0 | 20.1 | 23.01 | 0 | 0 | 20.1 |
| INV14: | | 0 | 2.94 | 3.38 | 0 | 0 | 2.94 |
| ND5: | | 0 | 24.55 | 27.91 | 0 | 21.27 | 24.55 |
| ND12: | | 0 | 29.41 | 29.31 | 0 | 29.41 | 29.41 |
| NR7: | | 0 | 22.33 | 32.64 | 0 | 22.33 | 22.33 |
| ND8: | | 0 | 37.06 | 40.81 | 0 | 37.06 | 37.06 |
| INV23: | | 0 | 1.45 | 1.71 | 0 | 0 | 1.45 |
| ND55: | | 0 | 7.24 | 8.48 | 0 | 0 | 7.24 |
| INV33: | 4 | 0 | 4.18 | 4.26 | 0 | 3.52 | 4.18 |
| ND42: | | 0 | 4.52 | 4.59 | 0 | 3.81 | 4.52 |
| ND43: | | 0 | 4.36 | 4.44 | 0 | 3.68 | 4.36 |
| NR5: | | 0 | 32.4 | 32.72 | 0 | 32.4 | 32.4 |
| ND18: | | 0 | 40.35 | 47.21 | 0 | 40.35 | 40.35 |
| ND15: | | 0 | 43.08 | 44.35 | 0 | 43.08 | 43.08 |
| ND21: | | 0 | 34.13 | 42.04 | 0 | 34.13 | 34.13 |
| INV12: | | 0 | 43.1 | 49.73 | 0 | 30.23 | 43.1 |
| ND2: | | 0 | 38.74 | 46.19 | 0 | 38.74 | 38.74 |
| INV21: | | 0 | 4.21 | 5.68 | 0 | 0 | 4.21 |
| INV20: | | 0 | 28.96 | 39 | 0 | 0 | 28.96 |
| INV18: | | 0 | 22.67 | 27.16 | 0 | 0 | 22.67 |
| | | | | | | | |
| ND10: | | 0 | 8.61 | 10.87 | 0 | 9.27 | 8.61 |
| ND9: | 5 | 0 | 25.67 | 35.11 | 0 | 0 | 25.67 |
| ND7: | | 0 | 25.67 | 35.11 | 0 | 0 | 25.67 |
| ND11: | | 0 | 8.61 | 10.87 | 0 | 9.27 | 8.61 |

Table 20 (continued). Circuit 74181 error probabilities.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| ND38: | | 0 | 17.09 | 19.61 | 0 | 0 | 17.09 |
| INV36: | | 0 | 4.31 | 4.96 | 0 | 0 | 4.31 |
| NR6: | | 0 | 0.15 | 4.61 | 0 | 0.15 | 0.15 |
| ND46: | | 0 | 16.36 | 19.41 | 0 | 0 | 16.36 |
| INV30: | | 0 | 4.16 | 4.94 | 0 | 0 | 4.16 |
| INV31: | | 0 | 4.16 | 4.94 | 0 | 0 | 4.16 |
| ND16: | 5 | 0 | 20.82 | 22.26 | 0 | 12.89 | 20.82 |
| ND14: | | 0 | 36.34 | 40.16 | 0 | 0 | 36.34 |
| ND13: | | 0 | 36.34 | 40.16 | 0 | 0 | 36.34 |
| ND17: | | 0 | 20.82 | 22.26 | 0 | 12.89 | 20.82 |
| INV1: | | 0 | 11.62 | 14.69 | 0 | 8.52 | 11.62 |
| ND1: | | 0 | 39.52 | 43.59 | 0 | 0 | 39.52 |
| ND20: | | 0 | 37.48 | 38.63 | 0 | 0 | 37.48 |
| ND19: | | 0 | 37.48 | 38.63 | 0 | 0 | 37.48 |
| ND24: | | 0 | 40.04 | 46.4 | 0 | 40.04 | 40.04 |
| ND3: | | 0 | 39.52 | 43.59 | 0 | 0 | 39.52 |
| | | | | | | | |
| ND45: | | 0 | 3.15 | 3.2 | 0 | 0 | 3.15 |
| INV34: | | 0 | 0.4 | 0.4 | 0 | 0 | 0.4 |
| INV6: | | 0 | 10.45 | 9.65 | 0 | 9.23 | 10.45 |
| INV7: | 6 | 0 | 11.39 | 19.64 | 0 | 11.14 | 11.39 |
| INV11: | | 0 | 10.33 | 16.99 | 0 | 7.83 | 10.33 |
| ND22: | | 0 | 18.98 | 21.6 | 0 | 10.06 | 18.98 |
| ND23: | | 0 | 18.98 | 21.6 | 0 | 10.06 | 18.98 |

Table 21. Circuit HIST error probabilities.

| Gates | Level | Pulse Widths(ns) | | | Voltage Amplitude | | |
|---|---|---|---|---|---|---|---|
| | | 1 | 5 | 9 | 3 | 4 | 5 |
| INV6: | | 20 | 40 | 60 | 40 | 40 | 40 |
| ND14: | | 20 | 40 | 60 | 40 | 40 | 40 |
| ND16: | 0 | 20 | 40 | 60 | 40 | 40 | 40 |
| ND18: | | 20 | 40 | 60 | 40 | 40 | 40 |
| ND12: | | 20 | 58.9 | 76.05 | 40 | 52.15 | 58.9 |
| | | | | | | | |
| NR3: | | 19 | 38.5 | 58 | 27 | 36 | 38.5 |
| ND13: | | 0 | 23.25 | 37.5 | 15.38 | 21 | 23.25 |
| INV10: | | 0 | 27.12 | 43.75 | 17.94 | 24.5 | 27.12 |
| INV7: | | 0 | 35.36 | 48.19 | 10.25 | 19.72 | 35.36 |
| INV14: | | 0 | 27.12 | 43.75 | 17.94 | 24.5 | 27.12 |
| NR1: | 1 | 20 | 40 | 60 | 40 | 40 | 40 |
| ND15: | | 0 | 15.5 | 25 | 10.25 | 14 | 15.5 |
| INV18: | | 0 | 25.19 | 40.62 | 16.66 | 22.75 | 25.19 |
| ND17: | | 0 | 15.5 | 25 | 10.25 | 14 | 15.5 |
| ND6: | | 20 | 40 | 60 | 40 | 40 | 40 |
| ND10: | | 0 | 47.45 | 65.16 | 17.94 | 24.5 | 47.45 |
| INV8: | | 0 | 47.45 | 65.16 | 17.94 | 24.5 | 47.45 |
| | | | | | | | |
| NR2 | | 0 | 31.5 | 40.13 | 0 | 20.25 | 31.5 |
| INV9: | | 0 | 9 | 11.72 | 0 | 6.66 | 9 |
| INV12: | 2 | 0 | 6 | 7.81 | 0 | 4.44 | 6 |
| INV13: | | 0 | 6 | 7.81 | 0 | 4.44 | 6 |
| INV16: | | 0 | 6 | 7.81 | 0 | 4.44 | 6 |
| INV17: | | 0 | 9 | 11.72 | 0 | 6.66 | 9 |

Table 21 (continued). Circuit Hist error probabilities.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| INV15: | 2 | 0 | 15.5 | 25 | 10.25 | 14 | 15.5 |
| ND4: | | 0 | 27.12 | 43.75 | 17.94 | 24.5 | 27.12 |
| INV1: | | 0 | 10.5 | 13.67 | 0 | 7.77 | 10.5 |
| INV2: | | 0 | 10.5 | 13.67 | 0 | 7.77 | 10.5 |
| INV3: | | 0 | 27.12 | 43.75 | 17.94 | 24.5 | 27.12 |
| INV5: | | 0 | 11.93 | 16.47 | 0 | 6.01 | 11.93 |
| INV4: | | 0 | 11.93 | 16.47 | 0 | 6.01 | 11.93 |
| ND11: | | 0 | 62.97 | 75.71 | 26.25 | 62.97 | 62.97 |
| | | | | | | | |
| ND5: | | 0 | 45.94 | 62.56 | 26.25 | 45.94 | 45.94 |
| ND7: | 3 | 0 | 51.42 | 59.4 | 0 | 21 | 51.42 |
| ND8: | | 0 | 45.99 | 53.41 | 0 | 18.38 | 45.99 |
| ND9: | | 0 | 45.99 | 53.41 | 0 | 18.38 | 45.99 |
| | | | | | | | |
| ND1: | | 0 | 29.75 | 38.06 | 0 | 21 | 29.75 |
| ND2: | 4 | 0 | 26.03 | 33.3 | 0 | 18.38 | 26.03 |
| ND3: | | 0 | 26.03 | 33.3 | 0 | 18.38 | 26.03 |

Table 22. Circuit BSG error probabilities.

| Gates | Level | Pulse Width(ns) | | | Voltage Amplitude | | |
|---|---|---|---|---|---|---|---|
| | | 1 | 5 | 9 | 3 | 4 | 5 |
| ND7: | | 20 | 40 | 60 | 40 | 40 | 40 |
| ND18: | | 20 | 40 | 60 | 40 | 40 | 40 |
| ND21: | | 20 | 40 | 60 | 40 | 40 | 40 |
| ND25: | | 20 | 40 | 60 | 40 | 40 | 40 |
| INV20: | | 20 | 40 | 60 | 40 | 40 | 40 |
| ND28: | 0 | 20 | 40 | 60 | 40 | 40 | 40 |
| ND31: | | 20 | 40 | 60 | 40 | 40 | 40 |
| INV24: | | 20 | 40 | 60 | 40 | 40 | 40 |
| ND33: | | 20 | 40 | 60 | 40 | 40 | 40 |
| ND9: | | 20 | 40 | 60 | 40 | 40 | 40 |
| ND17: | | 20 | 40 | 60 | 40 | 40 | 40 |
| ND4: | | 20 | 40 | 60 | 40 | 40 | 40 |
| | | | | | | | |
| ND3: | | 0.55 | 0.98 | 1.31 | 0.96 | 0.92 | 0.98 |
| INV6: | | 7.5 | 13.5 | 18 | 13.12 | 12.56 | 13.5 |
| INV7: | | 7.5 | 13.5 | 18 | 13.12 | 12.56 | 13.5 |
| ND8: | | 0.47 | 0.84 | 1.13 | 0.82 | 0.79 | 0.84 |
| INV18: | | 0 | 15.5 | 25 | 10.25 | 14 | 15.5 |
| INV19: | 1 | 0 | 15.5 | 25 | 10.25 | 14 | 15.5 |
| ND19: | | 0 | 23.25 | 37.5 | 15.38 | 21 | 23.25 |
| ND20: | | 0 | 23.25 | 37.5 | 15.38 | 21 | 23.25 |
| INV8: | | 11.25 | 20.25 | 27 | 19.69 | 18.84 | 20.25 |
| ND22: | | 19 | 38.5 | 58 | 27 | 36 | 38.5 |
| INV21: | | 11.25 | 20.25 | 27 | 19.69 | 18.84 | 20.25 |
| ND23: | | 7.5 | 13.5 | 18 | 13.12 | 12.56 | 13.5 |

Table 22 (continued). Circuit BSG error probabilities.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| ND24: | | 7.5 | 13.5 | 18 | 13.12 | 12.56 | 13.5 |
| INV22: | | 11.25 | 20.25 | 27 | 19.69 | 18.84 | 20.25 |
| ND26: | | 7.5 | 13.5 | 18 | 13.12 | 12.56 | 13.5 |
| ND27: | 1 | 7.5 | 13.5 | 18 | 13.12 | 12.56 | 13.5 |
| INV23: | | 11.25 | 20.25 | 27 | 19.69 | 18.84 | 20.25 |
| ND29: | | 7.5 | 13.5 | 18 | 13.12 | 12.56 | 13.5 |
| ND30: | | 7.5 | 13.5 | 18 | 13.12 | 12.56 | 13.5 |
| ND32: | | 19 | 38.5 | 58 | 27 | 36 | 38.5 |
| INV25: | | 5 | 9 | 12 | 8.75 | 8.37 | 9 |
| INV26: | | 5 | 9 | 12 | 8.75 | 8.37 | 9 |
| INV27: | | 5 | 9 | 12 | 8.75 | 8.37 | 9 |
| ND10: | | 0.47 | 0.84 | 1.13 | 0.82 | 0.79 | 0.84 |
| ND5: | | 0.47 | 0.84 | 1.13 | 0.82 | 0.79 | 0.84 |
| INV9: | | 11.25 | 20.25 | 27 | 19.69 | 18.84 | 20.25 |
| ND15: | | 0.94 | 1.69 | 2.25 | 1.64 | 1.57 | 1.69 |
| ND16: | | 0.94 | 1.69 | 2.25 | 1.64 | 1.57 | 1.69 |
| INV5: | | 13.12 | 23.62 | 31.5 | 22.97 | 21.98 | 23.62 |
| ND6: | | 5 | 9 | 12 | 8.75 | 8.37 | 9 |
| | | | | | | | |
| INV12: | | 0 | 0.67 | 0.96 | 0 | 0.52 | 0.67 |
| INV13: | | 0 | 5.59 | 7.17 | 0 | 2.27 | 5.59 |
| INV14: | | 0 | 42.51 | 55.32 | 14.33 | 35.21 | 42.51 |
| INV15: | | 0 | 40.45 | 53.74 | 6.75 | 33 | 40.45 |
| INV16: | 2 | 0 | 62.61 | 77.89 | 43.95 | 61.83 | 62.61 |
| INV17: | | 0 | 44.77 | 56.76 | 28.25 | 41.34 | 44.77 |
| ND11: | | 0 | 25.31 | 36.56 | 15.19 | 25.31 | 25.31 |
| ND14: | | 0 | 25.31 | 36.56 | 15.19 | 25.31 | 25.31 |

Table 22 (continued). Circuit BSG error probabilities.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| ND2: | 2 | 0 | 29.53 | 42.66 | 17.72 | 29.53 | 29.53 |
| INV10: | | 0 | 7.51 | 10.53 | 0 | 4.45 | 7.51 |
| | | | | | 0 | | |
| INV11: | | 0 | 2.92 | 3.73 | 0 | 0 | 2.92 |
| INV3: | 3 | 0 | 5.84 | 7.45 | 0 | 4.22 | 5.84 |
| INV4: | | 0 | 11.67 | 14.91 | 0 | 0 | 11.67 |
| INV2: | | 0 | 6.81 | 8.7 | 0 | 4.92 | 6.81 |
| | | | | | | | |
| ND12: | | 0 | 6.33 | 9.14 | 0 | 6.33 | 6.33 |
| ND13: | 4 | 0 | 16.88 | 18.28 | 0 | 16.88 | 16.88 |
| ND1: | | 0 | 7.38 | 10.66 | 0 | 7.38 | 7.38 |
| | | | | | | | |
| INV1: | 5 | 0 | 2.17 | 2.17 | 0 | 0.88 | 2.17 |

Table 23. Circuit S100 error probabilities.

| Gates | Level | Pulse Width(ns) | | | Voltage Amplitude | | |
|---|---|---|---|---|---|---|---|
| | | 1 | 5 | 9 | 3 | 4 | 5 |
| ND2: | | 20 | 40 | 60 | 40 | 40 | 40 |
| INV1: | | 20 | 40 | 60 | 40 | 40 | 40 |
| NR31: | | 36 | 64 | 84 | 64 | 64 | 64 |
| ND16: | | 20 | 40 | 60 | 40 | 40 | 40 |
| ND9: | | 20 | 40 | 60 | 40 | 40 | 40 |
| ND28: | 0 | 20 | 40 | 60 | 40 | 40 | 40 |
| ND27: | | 20 | 40 | 60 | 40 | 40 | 40 |
| INV11: | | 24.5 | 67.43 | 85.8 | 60.67 | 65.33 | 67.43 |
| INV19: | | 20 | 86.84 | 96.95 | 68.03 | 85.9 | 86.84 |
| ND18: | | 20 | 40 | 60 | 40 | 40 | 40 |
| ND20: | | 20 | 40 | 60 | 40 | 40 | 40 |
| ND21: | | 20 | 40 | 60 | 40 | 40 | 40 |
| ND22: | | 20 | 40 | 60 | 40 | 40 | 40 |
| | | | | | | | |
| ND1: | | 0.63 | 1.13 | | 1.09 | 1.05 | 1.13 |
| INV2: | | 13.79 | 24.83 | 33.11 | 24.14 | 23.1 | 24.83 |
| ND10: | | 13.67 | 24.61 | 32.81 | 23.93 | 22.9 | 24.61 |
| ND7: | | 0 | 40.74 | 58.26 | 23.62 | 40.74 | 40.74 |
| NR1: | | 2.19 | 3.94 | 5.25 | 3.83 | 3.66 | 3.94 |
| NR36: | 1 | 1.25 | 2.25 | 3 | 2.19 | 2.09 | 2.25 |
| NR35: | | 0 | 0 | 0 | 0 | 0 | 0 |
| NR2: | | 4.37 | 7.88 | 10.5 | 7.66 | 7.33 | 7.88 |
| ND17: | | 3.75 | 6.75 | 9 | 6.56 | 6.28 | 6.75 |
| ND26: | | 0 | 0 | 0 | 0 | 0 | 0 |
| ND23: | | 18.75 | 33.75 | 45 | 32.81 | 31.41 | 33.75 |

Table 23 (continued). Circuit S100 error probabilities.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| INV10: | | 19 | 79.18 | 91.9 | 58.5 | 78.33 | 79.18 |
| INV3: | | 15.45 | 29.38 | 38.66 | 27.04 | 25.88 | 29.38 |
| INV6: | | 17.5 | 33.63 | 43.63 | 30.62 | 31.51 | 33.63 |
| NR25: | | 19 | 84.52 | 92.39 | 37.28 | 83.89 | 84.52 |
| NR21: | | 20 | 36 | 48 | 35 | 33.5 | 36 |
| NR16: | 1 | 14.06 | 25.31 | 33.75 | 24.61 | 23.55 | 25.31 |
| ND3: | | 19 | 38.5 | 58 | 27 | 36 | 38.5 |
| INV13: | | 4.69 | 8.44 | 11.25 | 8.2 | 7.85 | 8.44 |
| INV20: | | 36 | 59.04 | 72.96 | 57.75 | 55.78 | 59.04 |
| ND19: | | 0 | 0 | 0 | 0 | 0 | 0 |
| ND15: | | 0 | 0 | 0 | 0 | 0 | 0 |
| | | | | | | | |
| ND24: | | 0 | 35.16 | 48.34 | 0 | 28.56 | 35.16 |
| ND6: | | 0 | 0 | 0 | 0 | 0 | 0 |
| NR8: | | 0 | 8.57 | 10.7 | 0 | 6.31 | 8.57 |
| ND5: | | 0 | 88.36 | 99.03 | 27 | 80.09 | 88.36 |
| NR13: | | 0 | 7.42 | 10.95 | 0 | 3.66 | 7.42 |
| NR14: | | 0 | 10.79 | 16.9 | 6.33 | 10.55 | 10.79 |
| NR15: | | 0 | 0.47 | 0.64 | 0 | 0.38 | 0.47 |
| NR24: | | 0 | 29.3 | 40.28 | 0 | 23.8 | 29.3 |
| NR22: | 2 | 0 | 0 | 0 | 0 | 0 | 0 |
| NR23: | | 0 | 24.03 | 36.39 | 0 | 0 | 24.03 |
| ND4: | | 0 | 36.16 | 51.42 | 20.86 | 36.16 | 36.16 |
| ND14: | | 0 | 90.23 | 95.63 | 0 | 27 | 90.23 |
| ND25: | | 0 | 0 | 0 | 0 | 0 | 0 |
| ND8: | | 0 | 0 | 0 | 0 | 0 | 0 |
| INV7: | | 0 | 2.2 | 3.22 | 0 | 1.64 | 2.2 |

Table 23 (continued). Circuit S100 error probabilities.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| NR3: | | 0 | 5.12 | 6.69 | 4.06 | 4.19 | 5.12 |
| NR4: | | 0 | 2.56 | 3.34 | 2.03 | 2.09 | 2.56 |
| NR28: | | 0 | 0 | 0 | 0 | 0 | 0 |
| NR20: | 2 | 0 | 0 | 0 | 0 | 0 | 0 |
| INV8: | | 0 | 1.28 | 1.67 | 1.02 | 1.05 | 1.28 |
| INV18: | | 0 | 0 | 0 | 0 | 0 | 0 |
| INV17: | | 0 | 0 | 0 | 0 | 0 | 0 |
| | | | | | | | |
| ND12: | | 0 | 88.46 | 88.98 | 0 | 78.64 | 88.46 |
| NR37: | | 0 | 0 | 0 | 0 | 0 | 0 |
| NR9: | | 0 | 0 | 0 | 0 | 0 | 0 |
| NR11: | | 0 | 91.44 | 94.28 | 0 | 89.14 | 91.44 |
| NR12: | | 0 | 6.75 | 10.65 | 0 | 3.3 | 6.75 |
| NR5: | | 0 | 27.68 | 35.08 | 0 | 19.01 | 27.68 |
| NR38 | | 0 | 21.67 | 21.42 | 0 | 16.99 | 21.67 |
| NR7: | 3 | 0 | 8.03 | 13.1 | 0 | 0 | 8.03 |
| NR6: | | 0 | 25.63 | 38.82 | 0 | 0 | 25.63 |
| ND13: | | 0 | 7.53 | 12.28 | 0 | 0 | 7.53 |
| INV9: | | 0 | 52.38 | 89.85 | 0 | 28.73 | 52.38 |
| INV16: | | 0 | 0 | 0 | 0 | 0 | 0 |
| INV5: | | 0 | 0 | 0 | 0 | 0 | 0 |
| NR19: | | 0 | 0 | 0 | 0 | 0 | 0 |
| INV15: | | 0 | 0 | 0 | 0 | 0 | 0 |
| | | | | | | | |
| NR10: | | 0 | 16.63 | 34.43 | 0 | 0 | 16.63 |
| NR18: | 4 | 0 | 0.06 | 19.5 | 0 | 0 | 0.06 |
| ND11: | | 0 | 17.89 | 20.01 | 0 | 11.12 | 17.89 |

Table 23 (continued). Circuit S100 error probabilities.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| INV12: | | 0 | 0 | 0 | 0 | 0 | 0 |
| INV4: | | 0 | 0 | 13.9 | 0 | 0 | 0 |
| NR27: | 4 | 0 | 0 | 0 | 0 | 0 | 0 |
| NR26: | | 0 | 0 | 0 | 0 | 0 | 0 |
| INV14: | | 0 | 0 | 0 | 0 | 0 | 0 |
| | | | | | | | |
| NR17: | 5 | 0 | 2.59 | 2.54 | 0 | 0 | 2.59 |

Table 24. Circuit S101 error probabilities.

| Gates | Level | Pulse Width(ns) | | | Voltage Amplitude | | |
|---|---|---|---|---|---|---|---|
| | | 1 | 5 | 9 | 3 | 4 | 5 |
| ND5: | | 20 | 40 | 60 | 40 | 40 | 40 |
| INV3: | | 20 | 40 | 60 | 40 | 40 | 40 |
| INV4: | | 20 | 40 | 60 | 40 | 40 | 40 |
| NR3: | | 20 | 40 | 60 | 40 | 40 | 40 |
| NR12: | | 20 | 40 | 60 | 40 | 40 | 40 |
| INV27: | | 20 | 40 | 60 | 40 | 40 | 40 |
| INV28: | | 20 | 40 | 60 | 40 | 40 | 40 |
| INV29: | | 20 | 40 | 60 | 40 | 40 | 40 |
| INV22: | | 20 | 40 | 60 | 40 | 40 | 40 |
| INV21: | | 20 | 40 | 60 | 40 | 40 | 40 |
| INV5: | | 20 | 40 | 60 | 40 | 40 | 40 |
| NR39: | 0 | 20 | 40 | 60 | 40 | 40 | 40 |
| ND13: | | 20 | 40 | 60 | 40 | 40 | 40 |
| NR46: | | 20 | 40 | 60 | 40 | 40 | 40 |
| ND37: | | 20 | 40 | 60 | 40 | 40 | 40 |
| ND49: | | 20 | 40 | 60 | 40 | 40 | 40 |
| INV16: | | 20 | 40 | 60 | 40 | 40 | 40 |
| ND45: | | 20 | 40 | 60 | 40 | 40 | 40 |
| INV10: | | 20 | 40 | 60 | 40 | 40 | 40 |
| ND51: | | 20 | 40 | 60 | 40 | 40 | 40 |
| NR26: | | 20 | 40 | 60 | 40 | 40 | 40 |
| INV8: | | 20 | 40 | 60 | 40 | 40 | 40 |
| NR44: | | 20 | 40 | 60 | 40 | 40 | 40 |
| NR45: | | 20 | 40 | 60 | 40 | 40 | 40 |
| INV1: | | 20 | 40 | 60 | 40 | 40 | 40 |

Table 24 (continued). Circuit S101 error probabilities.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| ND26: | 0 | 20 | 40 | 60 | 40 | 40 | 40 |
| NR28: | | 20 | 40 | 60 | 40 | 40 | 40 |
| | | | | | | | |
| ND4: | | 0 | 3.88 | 6.25 | 2.56 | 3.5 | 3.88 |
| ND7: | | 19 | 49.49 | 69.65 | 34.48 | 46.1 | 49.49 |
| ND3: | | 0 | 3.88 | 6.25 | 2.56 | 3.5 | 3.88 |
| NR7: | | 19 | 38.5 | 58 | 27 | 36 | 38.5 |
| NR10: | | 0 | 29.06 | 46.88 | 19.22 | 26.25 | 29.06 |
| NR11: | | 0 | 31 | 50 | 20.5 | 28 | 31 |
| NR47: | | 19 | 63.59 | 80.35 | 33.55 | 55.21 | 63.59 |
| ND38: | | 0 | 7.75 | 12.5 | 5.12 | 7 | 7.75 |
| NR49: | | 19 | 38.5 | 58 | 27 | 36 | 38.5 |
| ND11: | | 2.5 | 40.47 | 50.31 | 4.37 | 28.68 | 40.47 |
| ND39: | | 19 | 74.1 | 88.92 | 52.22 | 72.86 | 74.1 |
| NR35: | 1 | 19 | 38.5 | 58 | 27 | 36 | 38.5 |
| ND14: | | 19 | 38.5 | 58 | 27 | 36 | 38.5 |
| NR37: | | 0 | 27.12 | 43.75 | 17.94 | 24.5 | 27.12 |
| NR38: | | 0 | 29.06 | 48.39 | 19.22 | 26.25 | 29.06 |
| NR51: | | 0 | 28.09 | 45.31 | 18.58 | 25.37 | 28.09 |
| NR13: | | 7.5 | 13.5 | 18 | 13.12 | 12.56 | 13.5 |
| INV6: | | 3.75 | 6.75 | 9 | 6.56 | 6.28 | 6.75 |
| INV26: | | 0 | 7.75 | 12.5 | 5.12 | 7 | 7.75 |
| ND48: | | 0 | 26.16 | 42.19 | 17.3 | 23.62 | 26.16 |
| INV25: | | 0 | 31 | 50 | 20.5 | 28 | 31 |
| ND17: | | 0 | 56.32 | 70.19 | 3.84 | 39.74 | 56.32 |
| NR20: | | 19 | 38.5 | 58 | 27 | 36 | 38.5 |
| NR15: | | 19 | 38.5 | 58 | 27 | 36 | 38.5 |

Table 24 (continued). Circuit S101 error probabilities.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| INV2: | | 0 | 25.19 | 40.62 | 16.66 | 22.75 | 25.19 |
| NR17: | | 19 | 71.26 | 87.39 | 41.03 | 63.15 | 71.26 |
| ND20: | | 0 | 5.81 | 9.38 | 3.84 | 5.25 | 5.81 |
| ND18: | | 0 | 23.25 | 37.5 | 15.38 | 21 | 23.25 |
| ND34: | | 0.47 | 0.92 | 1.23 | 0.82 | 0.84 | 0.92 |
| ND35: | | 0.23 | 0.42 | 0.56 | 0.41 | 0.39 | 0.42 |
| INV18: | | 0 | 35.77 | 54.48 | 24.25 | 32.57 | 35.77 |
| ND36: | 1 | 0.16 | 0.28 | 0.38 | 0.27 | 0.26 | 0.28 |
| INV19: | | 0 | 23.25 | 37.5 | 15.38 | 21 | 23.25 |
| NR2: | | 19 | 38.5 | 58 | 27 | 36 | 38.5 |
| ND25: | | 0 | 31 | 50 | 20.5 | 28 | 31 |
| INV15: | | 0 | 31 | 50 | 20.5 | 28 | 31 |
| NR27: | | 1.52 | 2.74 | 3.66 | 2.67 | 2.55 | 2.74 |
| ND28: | | 0.47 | 0.84 | 1.13 | 0.82 | 0.79 | 0.84 |
| ND29: | | 4.06 | 7.31 | 9.75 | 7.11 | 6.8 | 7.31 |
| | | | | | | | |
| ND1: | | 0 | 5.25 | 6.69 | 0 | 3.38 | 5.25 |
| ND6: | | 0 | 15.72 | 20.65 | 0 | 10.25 | 15.72 |
| NR5: | | 0 | 43.41 | 55.03 | 0 | 29.22 | 43.41 |
| ND2: | | 0 | 11.12 | 14.22 | 0 | 7.19 | 11.12 |
| NR6: | | 0 | 31.5 | 40.13 | 0 | 20.25 | 31.5 |
| NR8: | 2 | 0 | 36.75 | 46.81 | 0 | 23.62 | 36.75 |
| ND8: | | 0 | 2.11 | 2.75 | 0 | 1.56 | 2.11 |
| ND9: | | 0 | 2.11 | 2.75 | 0 | 1.56 | 2.11 |
| ND27: | | 0 | 1.43 | 2.09 | 0 | 1.07 | 1.43 |
| NR9: | | 0 | 3 | 3.91 | 0 | 2.22 | 3 |
| ND10: | | 0 | 9 | 11.72 | 0 | 6.66 | 9 |

Table 24 (continued). Circuit S101 error probabilities.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| NR48: | | 0 | 31.5 | 40.13 | 0 | 20.25 | 31.5 |
| NR50: | | 0 | 9.97 | 14.16 | 0 | 6.8 | 9.97 |
| NR36: | | 0 | 22.13 | 15.23 | 0 | 16.49 | 22.13 |
| ND12: | | 0 | 31.5 | 40.13 | 0 | 20.25 | 31.5 |
| NR33: | | 0 | 31.5 | 40.13 | 0 | 20.25 | 31.5 |
| ND40: | | 0 | 20.12 | 25.33 | 0 | 6.75 | 20.12 |
| NR32: | | 0 | 52.45 | 63.22 | 0 | 20.25 | 52.45 |
| INV20: | | 0 | 4.93 | 7 | 0 | 3.37 | 4.93 |
| INV17: | | 0 | 31.27 | 44.27 | 0 | 21.13 | 31.27 |
| NR34: | | 0 | 36.75 | 46.81 | 0 | 23.62 | 36.75 |
| ND41: | | 0 | 2.63 | 3.42 | 0 | 1.94 | 2.63 |
| ND42: | | 0 | 11.92 | 15.29 | 0 | 8.74 | 11.92 |
| INV30: | | 0 | 35.7 | 45.4 | 0 | 25.77 | 35.7 |
| INV24: | 2 | 0 | 36.14 | 51.38 | 0 | 24.64 | 36.14 |
| NR53: | | 0 | 31.87 | 45.06 | 0 | 21.36 | 31.87 |
| NR4: | | 0 | 4.81 | 6.84 | 0 | 3.28 | 4.81 |
| NR52: | | 0 | 13.12 | 17.88 | 7.5 | 13.12 | 13.12 |
| NR54: | | 0 | 52.5 | 71.5 | 30 | 52.5 | 52.5 |
| NR42: | | 0 | 24.44 | 31.84 | 0 | 17.38 | 24.44 |
| NR18: | | 0 | 34.12 | 43.47 | 0 | 21.94 | 34.12 |
| NR14: | | 0 | 0.52 | 0.73 | 0 | 0.35 | 0.52 |
| INV9: | | 0 | 0.52 | 0.73 | 0 | 0.35 | 0.52 |
| ND16: | | 0 | 31.5 | 40.13 | 0 | 20.25 | 31.5 |
| ND19: | | 0 | 47.51 | 57.92 | 0 | 37.33 | 47.51 |
| INV11: | | 0 | 66.56 | 77.32 | 0 | 53.85 | 66.56 |
| NR43: | | 0 | 0.41 | 0.6 | 0 | 0.31 | 0.41 |
| INV7: | | 0 | 48.88 | 59.1 | 0 | 38.67 | 48.88 |

Table 24 (continued). Circuit S101 error probabilities.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| ND24: | | 0 | 50.87 | 62.25 | 0 | 40.33 | 50.87 |
| NR23: | | 0 | 23.4 | 29.93 | 0 | 9.38 | 23.4 |
| NR24: | 2 | 0 | 52.5 | 71.5 | 30 | 52.5 | 52.5 |
| NR1: | | 0 | 4.81 | 6.84 | 0 | 3.28 | 4.81 |
| NR16: | | 0 | 20.71 | 26.99 | 0 | 14.69 | 20.71 |
| | | | | | | | |
| ND44: | | 0 | 0.07 | 0.1 | 0 | 0.06 | 0.07 |
| ND50: | | 0 | 9.23 | 11.82 | 0 | 0.59 | 9.23 |
| NR29: | | 0 | 31 | 35.57 | 0 | 31 | 31 |
| ND33: | | 0 | 0.51 | 0.72 | 0 | 0 | 0.51 |
| ND32: | | 0 | 0.51 | 0.72 | 0 | 0 | 0.51 |
| NR41: | | 0 | 13.77 | 18.25 | 0 | 2.95 | 13.77 |
| INV14: | | 0 | 1.48 | 1.71 | 0 | 0 | 1.48 |
| ND15: | | 0 | 27.89 | 38.06 | 0 | 0 | 27.89 |
| NR22: | | 0 | 20.32 | 23.25 | 0 | 17.07 | 20.32 |
| NR19: | 3 | 0 | 2.57 | 3.18 | 0 | 1.89 | 2.57 |
| ND43: | | 0 | 0 | 0 | 0 | 0 | 0 |
| ND47: | | 0 | 4.69 | 5.98 | 0 | 3.64 | 4.69 |
| NR40: | | 0 | 0 | 0 | 0 | 0 | 0 |
| ND23: | | 0 | 29.56 | 33.81 | 0 | 24.83 | 29.56 |
| INV12: | | 0 | 0.4 | 0.46 | 0 | 0 | 0.4 |
| | | | | | | | |
| ND31: | | 0 | 8.18 | 9.89 | 0 | 0 | 8.18 |
| NR30: | | 0 | 0.56 | 0.77 | 0 | 0 | 0.56 |
| NR31: | 4 | 0 | 0.56 | 0.77 | 0 | 0 | 0.56 |
| ND30: | | 0 | 4.06 | 4.91 | 0 | 0 | 4.06 |
| NR21: | | 0 | 0.34 | 0.49 | 0 | 0 | 0.34 |

Table 24 (continued). Circuit S101 error probabilities

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| INV23: | | 0 | 0 | 0 | 0 | 0 | 0 |
| ND46: | 4 | 0 | 0 | 0 | 0 | 0 | 0 |
| INV13: | | 0 | 1.29 | 1.84 | 0 | 0 | 1.29 |
| NR25: | | 0 | 0.48 | 0.52 | 0 | 0.48 | 0.48 |
| | | | | | | | |
| ND21: | 5 | 0 | 0 | 0 | 0 | 0 | 0 |
| ND22: | | 0 | 0.17 | 0.18 | 0 | 0 | 0.17 |

Table 25. Circuit S200 error probabilities.

| Gates | Level | Pulse Width(ns) | | | Voltage Amplitude(volts) | | |
|---|---|---|---|---|---|---|---|
| | | 1 | 5 | 9 | 3 | 4 | 5 |
| ND6: | | 20 | 40 | 60 | 40 | 40 | 40 |
| NR8: | | 20 | 40 | 60 | 40 | 40 | 40 |
| ND26: | | 20 | 40 | 60 | 40 | 40 | 40 |
| NR12: | 0 | 20 | 40 | 60 | 40 | 40 | 40 |
| NR17: | | 20 | 40 | 60 | 40 | 40 | 40 |
| NR16: | | 20 | 40 | 60 | 40 | 40 | 40 |
| ND43: | | 20 | 40 | 60 | 40 | 40 | 40 |
| NR4: | | 20 | 40 | 60 | 40 | 40 | 40 |
| | | | | | | | |
| NR3: | | 0 | 24.95 | 40.23 | 16.5 | 22.53 | 24.95 |
| ND5: | | 0 | 3.71 | 5.98 | 2.45 | 3.35 | 3.71 |
| ND21: | | 0 | 21.22 | 30.53 | 4.48 | 16.04 | 21.22 |
| ND20: | | 0 | 8.48 | 13.67 | 5.61 | 7.66 | 8.48 |
| NR7: | | 0 | 24.95 | 40.23 | 16.5 | 22.53 | 24.95 |
| ND25: | | 0 | 3.71 | 5.98 | 2.45 | 3.35 | 3.71 |
| ND37: | | 0 | 21.22 | 30.53 | 4.48 | 16.04 | 21.22 |
| ND36: | 1 | 0 | 8.48 | 13.67 | 5.61 | 7.66 | 8.48 |
| NR11: | | 0 | 24.95 | 40.23 | 16.5 | 22.53 | 24.95 |
| NR18: | | 0 | 3.71 | 5.98 | 2.45 | 3.35 | 3.71 |
| ND49: | | 0 | 21.22 | 30.53 | 4.48 | 16.04 | 21.22 |
| ND50: | | 0 | 8.48 | 13.67 | 5.61 | 7.66 | 8.48 |
| NR14: | | 0 | 24.95 | 40.23 | 16.5 | 22.53 | 24.95 |
| ND42: | | 0 | 3.71 | 5.98 | 2.45 | 3.35 | 3.71 |
| ND12: | | 0 | 21.22 | 30.53 | 4.48 | 16.04 | 21.22 |
| ND13: | | 0 | 8.48 | 13.67 | 5.61 | 7.66 | 8.48 |

Table 25 (continued). Circuit S200 error probabilities.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| NR2: | | 0 | 2.63 | 3.74 | 0 | 1.79 | 2.63 |
| ND24: | | 0 | 9.68 | 13.75 | 0 | 6.6 | 9.68 |
| ND19: | | 0 | 13.05 | 18.61 | 0 | 4.1 | 13.05 |
| ND18: | | 0 | 9.23 | 13.22 | 0 | 2.87 | 9.23 |
| NR6: | | 0 | 2.63 | 3.74 | 0 | 1.79 | 2.63 |
| ND30: | | 0 | 9.68 | 13.75 | 0 | 6.6 | 9.68 |
| ND35: | | 0 | 13.05 | 18.61 | 0 | 4.1 | 13.05 |
| ND31: | | 0 | 9.23 | 13.22 | 0 | 2.87 | 9.23 |
| NR10: | 2 | 0 | 2.63 | 3.74 | 0 | 1.79 | 2.63 |
| ND41: | | 0 | 9.68 | 13.75 | 0 | 6.6 | 9.68 |
| ND48: | | 0 | 13.05 | 18.61 | 0 | 4.1 | 13.05 |
| ND45: | | 0 | 9.23 | 13.22 | 0 | 2.87 | 9.23 |
| NR15: | | 0 | 2.63 | 3.74 | 0 | 1.79 | 2.63 |
| ND10: | | 0 | 13.05 | 18.61 | 0 | 4.1 | 13.05 |
| ND11: | | 0 | 9.23 | 13.22 | 0 | 2.87 | 9.23 |
| ND4: | | 0 | 9.68 | 13.75 | 0 | 6.6 | 9.68 |
| | | | | | | | |
| NR5: | | 0 | 1.02 | 1.5 | 0 | 0 | 1.02 |
| ND15: | | 0 | 5.07 | 7.09 | 0 | 0 | 5.07 |
| ND16: | | 0 | 10.73 | 14.91 | 0 | 0 | 10.73 |
| ND17: | | 0 | 10.73 | 14.91 | 0 | 0 | 10.73 |
| ND23: | 3 | 0 | 4.68 | 6.88 | 0 | 0 | 4.68 |
| ND22: | | 0 | 4.68 | 6.88 | 0 | 0 | 4.68 |
| NR9: | | 0 | 1.02 | 1.5 | 0 | 0 | 1.02 |
| ND32: | | 0 | 5.07 | 7.09 | 0 | 0 | 5.07 |
| ND33: | | 0 | 10.73 | 14.91 | 0 | 0 | 10.73 |

Table 25 (continued). Circuit S200 error probabilities.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| ND34: | | 0 | 10.73 | 14.91 | 0 | 0 | 10.73 |
| ND29: | | 0 | 4.68 | 6.88 | 0 | 0 | 4.68 |
| ND28: | | 0 | 4.68 | 6.88 | 0 | 0 | 4.68 |
| NR13: | | 0 | 1.02 | 1.5 | 0 | 0 | 1.02 |
| ND44: | | 0 | 5.07 | 7.09 | 0 | 0 | 5.07 |
| ND46: | | 0 | 10.73 | 14.91 | 0 | 0 | 10.73 |
| ND47: | | 0 | 10.73 | 14.91 | 0 | 0 | 10.73 |
| ND40: | 3 | 0 | 4.68 | 6.88 | 0 | 0 | 4.68 |
| ND39: | | 0 | 4.68 | 6.88 | 0 | 0 | 4.68 |
| ND7: | | 0 | 5.07 | 7.09 | 0 | 0 | 5.07 |
| ND8: | | 0 | 10.73 | 14.91 | 0 | 0 | 10.73 |
| ND9: | | 0 | 10.73 | 14.91 | 0 | 0 | 10.73 |
| ND3: | | 0 | 4.68 | 6.88 | 0 | 0 | 4.68 |
| ND2: | | 0 | 4.68 | 6.88 | 0 | 0 | 4.68 |
| NR1: | | 0 | 1.02 | 1.5 | 0 | 0 | 1.02 |
| | | | | | | | |
| ND14: | | 0 | 3.37 | 4.58 | 0 | 0 | 3.37 |
| ND27: | 4 | 0 | 3.37 | 4.58 | 0 | 0 | 3.37 |
| ND38: | | 0 | 3.37 | 4.58 | 0 | 0 | 3.37 |
| ND1: | | 0 | 3.37 | 4.58 | 0 | 0 | 3.37 |

Table 26. Circuit S201 error probabilities.

| Gates | Level | Pulse Width(ns) | | | Voltage Amplitude | | |
|---|---|---|---|---|---|---|---|
| | | 1 | 5 | 9 | 3 | 4 | 5 |
| NR40: | | 20 | 40 | 60 | 40 | 40 | 40 |
| NR28: | | 20 | 40 | 60 | 40 | 40 | 40 |
| ND24: | | 20 | 40 | 60 | 40 | 40 | 40 |
| ND21: | | 20 | 40 | 60 | 40 | 40 | 40 |
| NR29: | | 20 | 40 | 60 | 40 | 40 | 40 |
| ND25: | | 20 | 40 | 60 | 40 | 40 | 40 |
| NR16: | | 20 | 40 | 60 | 40 | 40 | 40 |
| ND15: | | 20 | 40 | 60 | 40 | 40 | 40 |
| NR17: | | 20 | 40 | 60 | 40 | 40 | 40 |
| NR18: | | 20 | 40 | 60 | 40 | 40 | 40 |
| ND7: | | 20 | 40 | 60 | 40 | 40 | 40 |
| ND5: | | 20 | 40 | 60 | 40 | 40 | 40 |
| INV4: | 0 | 20 | 40 | 60 | 40 | 40 | 40 |
| NR5: | | 20 | 40 | 60 | 40 | 40 | 40 |
| ND8: | | 20 | 40 | 60 | 40 | 40 | 40 |
| ND20: | | 20 | 40 | 60 | 40 | 40 | 40 |
| ND72: | | 20 | 40 | 60 | 40 | 40 | 40 |
| ND64: | | 20 | 40 | 60 | 40 | 40 | 40 |
| INV31: | | 20 | 40 | 60 | 40 | 40 | 40 |
| ND65: | | 20 | 40 | 60 | 40 | 40 | 40 |
| ND63: | | 20 | 40 | 60 | 40 | 40 | 40 |
| INV14: | | 20 | 40 | 60 | 40 | 40 | 40 |
| ND38: | | 20 | 40 | 60 | 40 | 40 | 40 |
| ND37: | | 20 | 40 | 60 | 40 | 40 | 40 |
| ND26: | | 20 | 40 | 60 | 40 | 40 | 40 |

Table 26 (continued). Circuit S201 error probabilities.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| NR34: | | 20 | 40 | 60 | 40 | 40 | 40 |
| NR39: | 0 | 20 | 40 | 60 | 40 | 40 | 40 |
| ND57: | | 20 | 40 | 60 | 40 | 40 | 40 |
| ND75: | | 20 | 40 | 60 | 40 | 40 | 40 |
| | | | | | | | |
| ND74: | | 0 | 7.75 | 12.5 | 5.12 | 7 | 7.75 |
| ND6: | | 13.12 | 23.62 | 31.5 | 22.97 | 21.98 | 23.62 |
| INV3: | | 0 | 7.75 | 12.5 | 5.12 | 7 | 7.75 |
| INV18: | | 0 | 3.88 | 6.25 | 2.56 | 3.5 | 3.88 |
| NR4: | | 4.35 | 14.94 | 21.57 | 12.32 | 13.75 | 14.94 |
| ND54: | | 13.12 | 23.62 | 31.5 | 22.97 | 21.98 | 23.62 |
| INV17: | | 20 | 51.75 | 66.77 | 35 | 45.72 | 51.75 |
| NR25: | | 20 | 52.77 | 67.7 | 35 | 46.59 | 52.77 |
| NR55: | | 20 | 40.2 | 52.65 | 37.44 | 37.86 | 40.2 |
| ND55: | | 1.25 | 6.04 | 9.06 | 4.69 | 5.52 | 6.04 |
| INV6: | | 22.79 | 61.58 | 81.42 | 47.23 | 58.01 | 61.58 |
| INV20: | 1 | 2.5 | 39.21 | 58.88 | 27.87 | 35.84 | 39.21 |
| INV22: | | 0.63 | 1.12 | 1.5 | 1.09 | 1.05 | 1.12 |
| ND53: | | 2.48 | 4.45 | 5.91 | 4.33 | 4.14 | 4.45 |
| ND29: | | 1.25 | 12.62 | 17.49 | 2.19 | 9.38 | 12.62 |
| INV11: | | 2.5 | 8.2 | 11.87 | 6.83 | 7.54 | 8.2 |
| ND27: | | 5 | 49.1 | 69.06 | 36.63 | 45.35 | 49.1 |
| INV7: | | 0.63 | 1.12 | 1.5 | 1.09 | 1.05 | 1.12 |
| NR12: | | 2.5 | 4.5 | 6 | 4.37 | 4.19 | 4.5 |
| NR15: | | 0 | 18.77 | 29.69 | 12.55 | 17.01 | 18.77 |
| ND19: | | 0 | 0 | 0 | 0 | 0 | 0 |
| NR11: | | 9.84 | 17.72 | 23.63 | 17.23 | 16.49 | 17.72 |

Table 26 (continued). Circuit S201 error probabilities.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| NR10: | | 0 | 0 | 0 | 0 | 0 | 0 |
| NR46: | | 13.12 | 23.62 | 31.5 | 22.97 | 21.98 | 23.62 |
| INV30: | | 0 | 7.75 | 12.5 | 5.12 | 7 | 7.75 |
| NR52: | | 4.35 | 14.94 | 21.57 | 12.32 | 13.75 | 14.94 |
| INV33: | | 22.79 | 61.58 | 81.42 | 47.23 | 58.01 | 61.58 |
| INV34: | | 0.63 | 1.12 | 1.5 | 1.09 | 1.05 | 1.12 |
| ND35: | | 0 | 4.77 | 8.63 | 2.56 | 3.5 | 4.77 |
| NR20: | | 19 | 40.76 | 62.14 | 27 | 36 | 40.76 |
| INV23: | | 0 | 19.25 | 27.34 | 10.25 | 14 | 19.25 |
| ND34: | | 0 | 15.5 | 25 | 10.25 | 14 | 15.5 |
| ND33: | 1 | 0 | 27.12 | 43.75 | 17.94 | 24.5 | 27.12 |
| NR31: | | 0 | 15.5 | 25 | 10.25 | 14 | 15.5 |
| ND39: | | 3.75 | 6.75 | 9 | 6.56 | 6.28 | 6.75 |
| INV25: | | 3.75 | 6.75 | 9 | 6.56 | 6.28 | 6.75 |
| ND40: | | 11.25 | 20.25 | 27 | 19.69 | 18.84 | 20.25 |
| INV40: | | 0 | 7.75 | 12.5 | 5.12 | 7 | 7.75 |
| NR36: | | 2.5 | 42.67 | 56.27 | 4.37 | 31.72 | 42.67 |
| ND52: | | 5 | 54.24 | 75.25 | 40.47 | 50.19 | 54.24 |
| ND51: | | 1.25 | 2.25 | 3 | 2.19 | 2.09 | 2.25 |
| NR38: | | 2.5 | 4.5 | 6 | 4.37 | 4.19 | 4.5 |
| | | | | | | | |
| INV2: | | 0 | 21.53 | 31.58 | 0 | 16.08 | 21.53 |
| NR37: | | 0 | 2.34 | 3.44 | 0 | 1.75 | 2.34 |
| INV37: | | 0 | 35.62 | 47.89 | 0 | 3.5 | 35.62 |
| ND48: | 2 | 0 | 8.25 | 11.72 | 0 | 5.62 | 8.25 |
| ND4: | | 0 | 19.47 | 28.25 | 0 | 14.58 | 19.47 |
| INV5: | | 0 | 13.13 | 17.27 | 0 | 10.19 | 13.13 |

Table 26 (continued). Circuit S201 error probabilities.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| ND56: | | 0 | 1.7 | 2.45 | 0.84 | 1.62 | 1.7 |
| ND23: | | 0 | 44.26 | 56.76 | 0 | 21.89 | 44.26 |
| NR26: | | 0 | 2.12 | 2.8 | 0 | 1.61 | 2.12 |
| ND22: | | 0 | 64.4 | 76.32 | 27 | 64.13 | 64.4 |
| INV16: | | 0 | 0 | 0 | 0 | 0 | 0 |
| NR21: | | 0 | 0 | 0 | 0 | 0 | 0 |
| INV21: | | 0 | 8.35 | 10.98 | 0 | 6.38 | 8.35 |
| NR27: | | 0 | 61.06 | 78.5 | 37.44 | 61.06 | 61.06 |
| ND14: | | 0 | 11.82 | 16.34 | 7 | 11.82 | 11.82 |
| ND11: | | 0 | 1.41 | 2.03 | 0.84 | 1.41 | 1.41 |
| INV13: | | 0 | 44.97 | 59.38 | 0 | 13.87 | 44.97 |
| INV12: | | 0 | 36.49 | 49.09 | 0 | 26.25 | 36.49 |
| NR13: | | 0 | 58.65 | 64.44 | 15 | 34.9 | 58.65 |
| ND13: | | 0 | 2.34 | 3.44 | 0 | 1.75 | 2.34 |
| NR14: | 2 | 0 | 3.42 | 4.84 | 0 | 2.33 | 3.42 |
| ND30: | | 0 | 24.03 | 33.13 | 0 | 16.42 | 24.03 |
| ND9: | | 0 | 75.36 | 90.7 | 52.68 | 75.36 | 75.36 |
| NR8: | | 0 | 3.26 | 2.95 | 0 | 3.26 | 3.26 |
| INV8: | | 0 | 0 | 0 | 0 | 0 | 0 |
| ND17: | | 0 | 0 | 0 | 0 | 0 | 0 |
| ND18: | | 0 | 1.15 | 1.69 | 0 | 0.86 | 1.15 |
| NR9: | | 0 | 0 | 0 | 0 | 0 | 0 |
| INV9: | | 0 | 2.31 | 3.38 | 0 | 1.72 | 2.31 |
| ND71: | | 0 | 1.41 | 2.03 | 0.84 | 1.41 | 1.41 |
| NR53: | | 0 | 2.26 | 3 | 0 | 1.74 | 2.26 |
| INV32: | | 0 | 13.13 | 17.27 | 0 | 10.19 | 13.13 |
| INV29: | | 0 | 21.53 | 31.58 | 0 | 16.08 | 21.53 |

Table 26 (continued). Circuit S201 error probabilities.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| NR45: | | 0 | 19.47 | 28.25 | 0 | 14.58 | 19.47 |
| NR44: | | 0 | 13.12 | 17.88 | 7.5 | 13.12 | 13.12 |
| NR51: | | 0 | 75.36 | 90.7 | 52.68 | 75.36 | 75.36 |
| ND31: | | 0 | 34.79 | 42.54 | 0 | 22.03 | 34.79 |
| NR19: | | 0 | 35.06 | 45.11 | 15 | 31.57 | 35.06 |
| INV15: | | 0 | 10.5 | 13.67 | 0 | 7.77 | 10.5 |
| NR30: | | 0 | 18.38 | 23.93 | 0 | 13.59 | 18.38 |
| ND32: | 2 | 0 | 33.96 | 48.07 | 0 | 23.2 | 33.96 |
| NR33: | | 0 | 8.44 | 12.19 | 5.06 | 8.44 | 8.44 |
| ND43: | | 0 | 1.76 | 2.58 | 0 | 1.31 | 1.76 |
| ND58: | | 0 | 5.27 | 7.73 | 0 | 3.94 | 5.27 |
| INV26: | | 0 | 5.27 | 7.73 | 0 | 3.94 | 5.27 |
| NR1: | | 0 | 13.13 | 17.88 | 7.5 | 13.13 | 13.13 |
| NR3: | | 0 | 2.26 | 3 | 0 | 1.74 | 2.26 |
| ND49: | | 0 | 13.13 | 17.88 | 7.5 | 13.13 | 13.13 |
| NR35: | | 0 | 21.69 | 26.44 | 0 | 1.75 | 21.69 |
| ND50: | | 0 | 1.17 | 1.72 | 0 | 0.88 | 1.17 |
| INV39: | | 0 | 39.97 | 53.48 | 0 | 28.74 | 39.97 |
| | | | | | | | |
| INV1: | | 0 | 0.53 | 0.68 | 0 | 0.38 | 0.53 |
| INV38: | | 0 | 0.16 | 0.19 | 0 | 0 | 0.16 |
| NR22: | | 0 | 6.13 | 9.06 | 1.28 | 6.13 | 6.13 |
| NR23: | | 0 | 37.93 | 47.21 | 0 | 0 | 37.93 |
| NR24: | 3 | 0 | 37.93 | 47.21 | 0 | 0 | 37.93 |
| NR7: | | 0 | 53.46 | 63.37 | 0 | 0 | 53.46 |
| ND10: | | 0 | 60.08 | 70.46 | 0 | 0 | 60.08 |
| ND1: | | 0 | 17.53 | 20.66 | 0 | 1.5 | 17.53 |

Table 26 (continued). Circuit S201 error probabilities.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| ND28: | | 0 | 39.21 | 42.99 | 0 | 38.25 | 39.21 |
| ND12: | | 0 | 8.09 | 10.28 | 0 | 0 | 8.09 |
| INV10: | | 0 | 0.65 | 0.75 | 0 | 0 | 0.65 |
| ND16: | | 0 | 0 | 0 | 0 | 0 | 0 |
| ND2: | | 0 | 30.82 | 28.21 | 0 | 30.43 | 30.82 |
| ND60: | | 0 | 17.53 | 20.66 | 0 | 1.5 | 17.53 |
| NR48: | | 0 | 10.33 | 14.65 | 2.93 | 10.22 | 10.33 |
| ND61: | | 0 | 30.82 | 28.21 | 0 | 30.43 | 30.82 |
| INV28: | | 0 | 0.53 | 0.68 | 0 | 0.38 | 0.53 |
| NR47: | | 0 | 2.8 | 3.37 | 0 | 0 | 2.8 |
| NR49: | 3 | 0 | 53.46 | 63.37 | 0 | 0 | 53.46 |
| NR50: | | 0 | 60.08 | 70.46 | 0 | 0 | 60.08 |
| ND3: | | 0 | 2.8 | 3.37 | 0 | 0 | 2.8 |
| ND36: | | 0 | 17.51 | 20.51 | 0 | 9.44 | 17.51 |
| NR6: | | 0 | 10.34 | 14.67 | 2.93 | 10.22 | 10.34 |
| ND41: | | 0 | 2.48 | 2.53 | 0 | 1.77 | 2.48 |
| INV24: | | 0 | 2.48 | 2.53 | 0 | 1.77 | 2.48 |
| ND42: | | 0 | 7.39 | 9.01 | 0 | 1.37 | 7.39 |
| NR32: | | 0 | 9.25 | 9.07 | 0 | 8 | 9.25 |
| ND46: | | 0 | 4.87 | 6.15 | 0 | 0 | 4.87 |
| ND47: | | 0 | 40.84 | 44.92 | 0 | 40.84 | 40.84 |
| | | | | | | | |
| NR2: | 4 | 0 | 0 | 0 | 0 | 0 | 0 |

Table 27. Circuit S400 error probabilities.

| Gates | Level | Pulse Width(ns) | | | Voltage Amplitude | | |
|---|---|---|---|---|---|---|---|
| | | 1 | 5 | 9 | 3 | 4 | 5 |
| &_1I1_NR40_32: | | 20 | 40 | 60 | 40 | 40 | 40 |
| &_1I1_NR28_54: | | 20 | 40 | 60 | 40 | 40 | 40 |
| &_1I1_ND24_62: | | 20 | 40 | 60 | 40 | 40 | 40 |
| &_1I1_ND21_63: | | 20 | 40 | 60 | 40 | 40 | 40 |
| &_1I1_NR29_65: | | 20 | 40 | 60 | 40 | 40 | 40 |
| &_1I1_ND25_66: | | 20 | 40 | 60 | 40 | 40 | 40 |
| &_1I1_NR16_83: | | 20 | 40 | 60 | 40 | 40 | 40 |
| &_1I1_ND15_85: | | 20 | 40 | 60 | 40 | 40 | 40 |
| &_1I1_NR17_86: | | 20 | 40 | 60 | 40 | 40 | 40 |
| &_1I1_NR18_89: | | 20 | 40 | 60 | 40 | 40 | 40 |
| &_1I1_ND7_93: | | 20 | 40 | 60 | 40 | 40 | 40 |
| &_1I1_ND5_99: | | 20 | 40 | 60 | 40 | 40 | 40 |
| &_1I1_INV4_109: | | 20 | 40 | 60 | 40 | 40 | 40 |
| &_1I1_NR5_110: | | 20 | 40 | 60 | 40 | 40 | 40 |
| &_1I1_ND8_116: | | 20 | 40 | 60 | 40 | 40 | 40 |
| &_1I1_ND20_117: | | 20 | 40 | 60 | 40 | 40 | 40 |
| &_1I1_ND72_120: | | 20 | 40 | 60 | 40 | 40 | 40 |
| &_1I1_ND64_125: | | 20 | 40 | 60 | 40 | 40 | 40 |
| &_1I1_INV31_127: | | 20 | 40 | 60 | 40 | 40 | 40 |
| &_1I1_ND65_128: | | 20 | 40 | 60 | 40 | 40 | 40 |
| &_1I1_ND63_129: | | 20 | 40 | 60 | 40 | 40 | 40 |
| &_1I1_INV14_166: | | 20 | 40 | 60 | 40 | 40 | 40 |
| &_1I1_ND38_171: | | 20 | 40 | 60 | 40 | 40 | 40 |
| &_1I1_ND37_173: | | 20 | 40 | 60 | 40 | 40 | 40 |
| &_1I1_ND26_176: | | 20 | 40 | 60 | 40 | 40 | 40 |

Table 27 (continued). Circuit S400 error probabilities.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| &_1I1_NR34_203: | | 20 | 40 | 60 | 40 | 40 | 40 |
| &_1I1_NR39_221: | | 20 | 40 | 60 | 40 | 40 | 40 |
| &_1I1_ND57_223: | | 20 | 40 | 60 | 40 | 40 | 40 |
| &_1I1_ND75_224: | | 20 | 40 | 60 | 40 | 40 | 40 |
| &_1I2_ND69_241: | | 20 | 40 | 60 | 40 | 40 | 40 |
| &_1I2_ND66_250: | | 20 | 41.8 | 61.56 | 40 | 41.33 | 41.8 |
| &_1I2_ND62_255: | | 20 | 40.45 | 60.39 | 40 | 40.33 | 40.45 |
| &_1I2_ND59_260: | | 20 | 43.6 | 63.13 | 40 | 42.66 | 43.6 |
| &_1I2_NR2_298: | | 20 | 58.6 | 80 | 52.3 | 56.8 | 58.6 |
| &_1I2_INV29_302: | | 20 | 40 | 60 | 40 | 40 | 40 |
| &_1I2_ND56_313: | | 20 | 40 | 60 | 40 | 40 | 40 |
| &_1I2_NR8_314: | | 20 | 40 | 60 | 40 | 40 | 40 |
| | | | | | | | |
| &_1I1_ND74_31: | | 0 | 7.75 | 12.5 | 5.12 | 7 | 7.75 |
| &_1I1_ND6_38: | | 13.12 | 23.62 | 31.5 | 22.97 | 21.98 | 23.62 |
| &_1I1_INV3_39: | | 0 | 7.75 | 12.5 | 5.12 | 7 | 7.75 |
| &_1I1_INV18_44: | | 0 | 3.88 | 6.25 | 2.56 | 3.5 | 3.88 |
| &_1I1_NR4_45: | | 4.35 | 14.94 | 21.57 | 12.32 | 13.75 | 14.94 |
| &_1I1_ND54_47: | | 13.12 | 23.62 | 31.5 | 22.97 | 21.98 | 23.62 |
| &_1I1_INV17_51: | | 20 | 51.75 | 66.77 | 35 | 45.72 | 51.75 |
| &_1I1_NR25_52: | | 20 | 52.77 | 67.7 | 35 | 46.59 | 52.77 |
| &_1I1_NR55_53: | | 20 | 40.2 | 52.65 | 37.44 | 37.86 | 40.2 |
| &_1I1_ND55_55: | | 1.25 | 6.04 | 9.06 | 4.69 | 5.52 | 6.04 |
| &_1I1_INV6_56: | | 22.79 | 61.58 | 81.42 | 47.23 | 58.01 | 61.58 |
| &_1I1_INV20_57: | | 2.5 | 84.3 | 92.81 | 27.87 | 83.43 | 84.3 |
| &_1I1_INV22_60: | | 0.63 | 1.12 | 1.5 | 1.09 | 1.05 | 1.12 |
| &_1I1_ND53_73: | | 2.48 | 4.45 | 5.91 | 4.33 | 4.14 | 4.45 |

Table 27 (continued). Circuit S400 error probabilities.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| &_1I1_ND29_74: | | 1.25 | 12.62 | 17.49 | 2.19 | 9.38 | 12.62 |
| &_1I1_INV11_79: | | 2.5 | 8.2 | 11.87 | 6.83 | 7.54 | 8.2 |
| &_1I1_ND27_80: | | 5 | 49.1 | 69.06 | 36.63 | 45.35 | 49.1 |
| &_1I1_INV7_82: | | 0.63 | 1.12 | 1.5 | 1.09 | 1.05 | 1.12 |
| &_1I1_NR12_84: | | 2.5 | 4.5 | 6 | 4.37 | 4.19 | 4.5 |
| &_1I1_NR15_88: | | 0 | 18.77 | 29.69 | 12.55 | 17.01 | 18.77 |
| &_1I1_ND19_112: | | 0 | 0 | 0 | 0 | 0 | 0 |
| &_1I1_NR11_114: | | 9.84 | 17.72 | 23.63 | 17.23 | 16.49 | 17.72 |
| &_1I1_NR10_115: | | 0 | 0 | 0 | 0 | 0 | 0 |
| &_1I1_NR46_126: | | 13.12 | 23.62 | 31.5 | 22.97 | 21.98 | 23.62 |
| &_1I1_INV30_137 | | 0 | 7.75 | 12.5 | 5.12 | 7 | 7.75 |
| &_1I1_NR52_146: | | 4.35 | 14.94 | 21.57 | 12.32 | 13.75 | 14.94 |
| &_1I1_INV33_147: | | 22.79 | 61.58 | 81.42 | 47.23 | 58.01 | 61.58 |
| &_1I1_INV34_150: | | 0.63 | 1.12 | 1.5 | 1.09 | 1.05 | 1.12 |
| &_1I1_ND35_164: | | 0 | 4.77 | 8.63 | 2.56 | 3.5 | 4.77 |
| &_1I1_NR20_165: | | 19 | 40.76 | 62.14 | 27 | 36 | 40.76 |
| &_1I1_INV23_169 | | 0 | 19.25 | 27.34 | 10.25 | 14 | 19.25 |
| &_1I1_ND34_170: | | 0 | 15.5 | 25 | 10.25 | 14 | 15.5 |
| &_1I1_ND33_172: | | 0 | 27.12 | 43.75 | 17.94 | 24.5 | 27.12 |
| &_1I1_NR31_174: | | 0 | 15.5 | 25 | 10.25 | 14 | 15.5 |
| &_1I1_ND39_188: | | 3.75 | 6.75 | 9 | 6.56 | 6.28 | 6.75 |
| &_1I1_INV25_195: | | 3.75 | 6.75 | 9 | 6.56 | 6.28 | 6.75 |
| &_1I1_ND40_200: | | 11.25 | 20.25 | 27 | 19.69 | 18.84 | 20.25 |
| &_1I1_INV40_215: | | 0 | 7.75 | 12.5 | 5.12 | 7 | 7.75 |
| &_1I1_NR36_217: | | 2.5 | 42.67 | 56.27 | 4.37 | 31.72 | 42.67 |
| &_1I1_ND52_218: | | 5 | 54.24 | 75.25 | 40.47 | 50.19 | 54.24 |
| &_1I1_ND51_222: | | 1.25 | 2.25 | 3 | 2.19 | 2.09 | 2.25 |

Table 27 (continued). Circuit S400 error probabilities.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| &_1I1_NR38_225: | | 2.5 | 4.5 | 6 | 4.37 | 4.19 | 4.5 |
| &_1I2_ND68_243: | | 0 | 15.5 | 25 | 10.25 | 14 | 15.5 |
| &_1I2_ND67_244: | | 0 | 31 | 50 | 20.5 | 28 | 31 |
| &_1I2_ND64_247: | | 0 | 32.03 | 51.36 | 20.5 | 28 | 32.03 |
| &_1I2_ND65_248: | | 0 | 24.11 | 38.77 | 15.38 | 21 | 24.11 |
| &_1I2_ND60_252: | | 0 | 31.26 | 50.34 | 20.5 | 28 | 31.26 |
| &_1I2_ND61_253: | | 0 | 0 | 0 | 0 | 0 | 0 |
| &_1I2_ND57_257: | | 0 | 29.04 | 46.43 | 17.94 | 24.5 | 29.04 |
| &_1I2_ND58_258: | | 0 | 33.07 | 52.72 | 20.5 | 28 | 33.07 |
| &_1I2_ND54_282: | | 0 | 0 | 0 | 0 | 0 | 0 |
| &_1I2_INV22_299 | | 0 | 61.36 | 81.25 | 20.5 | 49.6 | 61.36 |
| &_1I2_NR4_304: | | 19 | 38.5 | 58 | 27 | 36 | 38.5 |
| &_1I2_INV27_307: | | 0 | 0 | 0 | 0 | 0 | 0 |
| &_1I2_ND63_315: | | 0 | 15.5 | 25 | 10.25 | 14 | 15.5 |
| &_1I2_INV43_316: | | 0 | 0 | 0 | 0 | 0 | 0 |
| | | | | | | | |
| &_1I1_INV2_28: | | 0 | 21.53 | 31.58 | 0 | 16.08 | 21.53 |
| &_1I1_NR37_29: | | 0 | 2.34 | 3.44 | 0 | 1.75 | 2.34 |
| &_1I1_INV37_30: | | 0 | 35.62 | 47.89 | 0 | 3.5 | 35.62 |
| &_1I1_ND48_34: | | 0 | 8.25 | 11.72 | 0 | 5.62 | 8.25 |
| &_1I1_ND4_35: | | 0 | 19.47 | 28.25 | 0 | 14.58 | 19.47 |
| &_1I1_INV5_36: | | 0 | 13.13 | 17.27 | 0 | 10.19 | 13.13 |
| &_1I1_ND56_40: | | 0 | 1.7 | 2.45 | 0.84 | 1.62 | 1.7 |
| &_1I1_ND23_41: | | 0 | 44.26 | 56.76 | 0 | 21.89 | 44.26 |
| &_1I1_NR26_43: | | 0 | 2.12 | 2.8 | 0 | 1.61 | 2.12 |
| &_1I1_ND22_46: | | 0 | 64.4 | 76.32 | 27 | 64.13 | 64.4 |
| &_1I1_INV16_48: | | 0 | 0 | 0 | 0 | 0 | 0 |

Table 27 (continued). Circuit S400 error probabilities.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| &_1I1_NR21_49: | | 0 | 0 | 0 | 0 | 0 | 0 |
| &_1I1_INV21_50: | | 0 | 8.35 | 10.98 | 0 | 6.38 | 8.35 |
| &_1I1_NR27_61: | | 0 | 88.02 | 94.67 | 37.44 | 88.02 | 88.02 |
| &_1I1_ND14_69: | | 0 | 11.82 | 16.34 | 7 | 11.82 | 11.82 |
| &_1I1_ND11_71: | | 0 | 1.41 | 2.03 | 0.84 | 1.41 | 1.41 |
| &_1I1_INV13_72: | | 0 | 44.97 | 59.38 | 0 | 13.87 | 44.97 |
| &_1I1_INV12_75: | | 0 | 36.49 | 49.09 | 0 | 26.25 | 36.49 |
| &_1I1_NR13_76: | | 0 | 58.65 | 64.44 | 15 | 34.9 | 58.65 |
| &_1I1_ND13_81: | | 0 | 2.34 | 3.44 | 0 | 1.75 | 2.34 |
| &_1I1_NR14_87: | | 0 | 3.42 | 4.84 | 0 | 2.33 | 3.42 |
| &_1I1_ND30_91: | | 0 | 24.03 | 33.13 | 0 | 16.42 | 24.03 |
| &_1I1_ND9_92: | | 0 | 75.36 | 90.7 | 52.68 | 75.36 | 75.36 |
| &_1I1_NR8_105: | | 0 | 3.26 | 2.95 | 0 | 3.26 | 3.26 |
| &_1I1_INV8_106: | | 0 | 0 | 0 | 0 | 0 | 0 |
| &_1I1_ND17_107: | | 0 | 0 | 0 | 0 | 0 | 0 |
| &_1I1_ND18_108: | | 0 | 1.15 | 1.69 | 0 | 0.86 | 1.15 |
| &_1I1_NR9_111: | | 0 | 0 | 0 | 0 | 0 | 0 |
| &_1I1_INV9_113: | | 0 | 2.31 | 3.38 | 0 | 1.72 | 2.31 |
| &_1I1_ND71_134: | | 0 | 1.41 | 2.03 | 0.84 | 1.41 | 1.41 |
| &_1I1_NR53_136: | | 0 | 2.26 | 3 | 0 | 1.74 | 2.26 |
| &_1I1_INV32_141: | | 0 | 13.13 | 17.27 | 0 | 10.19 | 13.13 |
| &_1I1_INV29_142: | | 0 | 21.53 | 31.58 | 0 | 16.08 | 21.53 |
| &_1I1_NR45_143: | | 0 | 19.47 | 28.25 | 0 | 14.58 | 19.47 |
| &_1I1_NR44_144: | | 0 | 13.13 | 17.88 | 7.5 | 13.13 | 13.13 |
| &_1I1_NR51_151: | | 0 | 75.36 | 90.7 | 52.68 | 75.36 | 75.36 |
| &_1I1_ND31_161: | | 0 | 34.79 | 42.54 | 0 | 22.03 | 34.79 |
| &_1I1_NR19_162: | | 0 | 35.06 | 45.11 | 15 | 31.57 | 35.06 |

Table 27 (continued). Circuit S400 error probabilities.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| &_1I1_INV15_163: | | 0 | 10.5 | 13.67 | 0 | 7.77 | 10.5 |
| &_1I1_NR30_168: | | 0 | 18.38 | 23.93 | 0 | 13.59 | 18.38 |
| &_1I1_ND32_175: | | 0 | 33.96 | 48.07 | 0 | 23.2 | 33.96 |
| &_1I1_NR33_191: | | 0 | 8.44 | 12.19 | 5.06 | 8.44 | 8.44 |
| &_1I1_ND43_193: | | 0 | 1.76 | 2.58 | 0 | 1.31 | 1.76 |
| &_1I1_ND58_197: | | 0 | 5.27 | 7.73 | 0 | 3.94 | 5.27 |
| &_1I1_INV26_201: | | 0 | 5.27 | 7.73 | 0 | 3.94 | 5.27 |
| &_1I1_NR1_205: | | 0 | 13.13 | 17.88 | 7.5 | 13.13 | 13.13 |
| &_1I1_NR3_211: | | 0 | 2.26 | 3 | 0 | 1.74 | 2.26 |
| &_1I1_ND49_214: | | 0 | 13.13 | 17.88 | 7.5 | 13.13 | 13.13 |
| &_1I1_NR35_216: | | 0 | 21.69 | 26.44 | 0 | 1.75 | 21.69 |
| &_1I1_ND50_219: | | 0 | 1.17 | 1.72 | 0 | 0.88 | 1.17 |
| &_1I1_INV39_220: | | 0 | 39.97 | 53.48 | 0 | 28.74 | 39.97 |
| &_1I2_ND41_227: | | 0 | 43.83 | 50.34 | 0 | 43.83 | 43.83 |
| &_1I2_ND34_232: | | 0 | 44.82 | 51.36 | 0 | 44.82 | 44.82 |
| &_1I2_ND27_237: | | 0 | 43.5 | 50 | 0 | 43.5 | 43.5 |
| &_1I2_ND28_240: | | 0 | 22 | 31.25 | 0 | 15 | 22 |
| &_1I2_INV41_242: | | 0 | 44 | 62.5 | 0 | 30 | 44 |
| &_1I2_INV42_245: | | 0 | 11 | 15.62 | 0 | 7.5 | 11 |
| &_1I2_INV40_246: | | 0 | 11.76 | 16.45 | 0 | 7.5 | 11.76 |
| &_1I2_INV39_249: | | 0 | 45.91 | 63.96 | 0 | 30 | 45.91 |
| &_1I2_INV37_251: | | 0 | 44.48 | 62.87 | 0 | 30 | 44.48 |
| &_1I2_INV35_254: | | 0 | 44.48 | 62.87 | 0 | 30 | 44.48 |
| &_1I2_INV44_256: | | 0 | 6.3 | 8.71 | 0 | 3.75 | 6.3 |
| &_1I2_INV45_259: | | 0 | 0 | 0 | 0 | 0 | 0 |
| &_1I2_ND31_262: | | 0 | 26.19 | 36.58 | 0 | 26.35 | 26.19 |
| &_1I2_ND37_274: | | 0 | 43.83 | 50.34 | 0 | 43.83 | 43.83 |

Table 27 (continued). Circuit S400 error probabilities.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| &_1I2_ND53_279: | | 0 | 0 | 0 | 0 | 0 | 0 |
| &_1I2_ND52_280: | | 0 | 0 | 0 | 0 | 0 | 0 |
| &_1I2_INV24_284: | | 0 | 0 | 0 | 0 | 0 | 0 |
| &_1I2_NR9_297: | | 0 | 37.15 | 50.97 | 0 | 27.58 | 37.15 |
| &_1I2_INV28_301: | | 0 | 0 | 0 | 0 | 0 | 0 |
| &_1I2_ND50_305: | | 0 | 31.5 | 40.13 | 0 | 20.25 | 31.5 |
| &_1I2_NR3_311: | | 0 | 0 | 0 | 0 | 0 | 0 |
| &_1I2_INV3_342: | | 0 | 83.52 | 93.01 | 30 | 74.77 | 83.52 |
| &_1I2_ND49_352: | | 0 | 47.82 | 65.43 | 0 | 30 | 47.82 |
| | | | | | | | |
| &_1I1_INV1_27: | | 0 | 0.53 | 0.68 | 0 | 0.38 | 0.53 |
| &_1I1_INV38_33: | | 0 | 0.16 | 0.19 | 0 | 0 | 0.16 |
| &_1I1_NR22_42: | | 0 | 6.13 | 9.06 | 1.28 | 6.13 | 6.13 |
| &_1I1_NR23_58: | | 0 | 84.98 | 87.87 | 0 | 0 | 84.98 |
| &_1I1_NR24_59: | | 0 | 84.98 | 87.87 | 0 | 0 | 84.98 |
| &_1I1_NR7_67: | | 0 | 53.46 | 63.37 | 0 | 0 | 53.46 |
| &_1I1_ND10_68: | | 0 | 60.08 | 70.46 | 0 | 0 | 60.08 |
| &_1I1_ND1_70: | | 0 | 17.53 | 20.66 | 0 | 1.5 | 17.53 |
| &_1I1_ND28_77: | | 0 | 39.21 | 42.99 | 0 | 38.25 | 39.21 |
| &_1I1_ND12_78: | | 0 | 8.09 | 10.28 | 0 | 0 | 8.09 |
| &_1I1_INV10_90: | | 0 | 0.65 | 0.75 | 0 | 0 | 0.65 |
| &_1I1_ND16_104: | | 0 | 0 | 0 | 0 | 0 | 0 |
| &_1I1_ND2_118: | | 0 | 30.82 | 28.21 | 0 | 30.43 | 30.82 |
| &_1I1_ND60_119: | | 0 | 17.53 | 20.66 | 0 | 1.5 | 17.53 |
| &_1I1_NR48_135: | | 0 | 10.33 | 14.65 | 2.93 | 10.22 | 10.33 |
| &_1I1_ND61_138: | | 0 | 30.82 | 28.21 | 0 | 30.43 | 30.82 |
| &_1I1_INV28_139: | | 0 | 0.53 | 0.68 | 0 | 0.38 | 0.53. |

Table 27 (continued). Circuit S400 error probabilities.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| &_1I1_NR47_140: | | 0 | 2.8 | 3.37 | 0 | 0 | 2.8 |
| &_1I1_NR49_148: | | 0 | 53.46 | 63.37 | 0 | 0 | 53.46 |
| &_1I1_NR50_149: | | 0 | 60.08 | 70.46 | 0 | 0 | 60.08 |
| &_1I1_ND3_159: | | 0 | 2.8 | 3.37 | 0 | 0 | 2.8 |
| &_1I1_ND36_167: | | 0 | 17.51 | 20.51 | 0 | 9.44 | 17.51 |
| &_1I1_NR6_177: | | 0 | 10.34 | 14.67 | 2.93 | 10.22 | 10.34 |
| &_1I1_ND41_189: | | 0 | 2.48 | 2.53 | 0 | 1.77 | 2.48 |
| &_1I1_INV24_190: | | 0 | 2.48 | 2.53 | 0 | 1.77 | 2.48 |
| &_1I1_ND42_192: | | 0 | 7.39 | 9.01 | 0 | 1.37 | 7.39 |
| &_1I1_NR32_194: | | 0 | 9.25 | 9.07 | 0 | 8 | 9.25 |
| &_1I1_ND45_206: | | 0 | 35.99 | 42.29 | 0 | 30.97 | 35.99 |
| &_1I1_ND46_212: | | 0 | 4.87 | 6.15 | 0 | 0 | 4.87 |
| &_1I1_ND47_213: | | 0 | 40.84 | 44.92 | 0 | 40.84 | 40.84 |
| &_1I2_ND39_228: | | 0 | 46.01 | 55.46 | 0 | 0 | 46.01 |
| &_1I2_ND33_231: | | 0 | 12.21 | 14.63 | 0 | 0 | 12.21 |
| &_1I2_ND32_233: | | 0 | 47.54 | 56.83 | 0 | 0 | 47.54 |
| &_1I2_ND25_236: | | 0 | 22.75 | 27.5 | 0 | 0 | 22.75 |
| &_1I2_ND26_238: | | 0 | 45.5 | 55 | 0 | 0 | 45.5 |
| &_1I2_ND29_261: | | 0 | 29 | 29.22 | 0 | 0 | 29 |
| &_1I2_ND30_263: | | 0 | 29 | 29.22 | 0 | 0 | 29 |
| &_1I2_ND35_265: | | 0 | 46.01 | 55.46 | 0 | 24 | 46.01 |
| &_1I2_ND36_266: | | 0 | 46.01 | 55.46 | 0 | 24 | 46.01 |
| &_1I2_INV38_276: | | 0 | 46.01 | 55.46 | 0 | 24 | 46.01 |
| &_1I2_NR1_283: | | 0 | 0 | 0 | 0 | 0 | 0 |
| &_1I2_ND44_287: | | 0 | 0 | 0 | 0 | 0 | 0 |
| &_1I2_INV32_296: | | 0 | 27.8 | 40.85 | 0 | 0 | 27.8 |
| &_1I2_ND51_306: | | 0 | 0 | 0 | 0 | 0 | 0 |

Table 27 (continued). Circuit S400 error probabilities.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| &_1I2_INV25_308: | | 0 | 0 | 0 | 0 | 0 | 0 |
| &_1I2_INV26_310: | | 0 | 0 | 0 | 0 | 0 | 0 |
| &_1I2_INV13_322: | | 0 | 23.74 | 28.16 | 0 | 12.26 | 23.74 |
| &_1I2_ND6_340: | | 0 | 76.87 | 88.91 | 0 | 76.87 | 76.87 |
| &_1I2_INV2_341: | | 0 | 49.59 | 65.43 | 0 | 0 | 49.59 |
| &_1I2_INV4_343: | | 0 | 43.83 | 50.34 | 0 | 0 | 43.83 |
| &_1I2_INV5_344: | | 0 | 43.83 | 50.34 | 0 | 0 | 43.83 |
| &_1I2_INV8_345: | | 0 | 44.82 | 51.36 | 0 | 0 | 44.82 |
| &_1I2_INV9_346: | | 0 | 49.81 | 58.44 | 0 | 42.21 | 49.81 |
| &_1I2_INV10_347: | | 0 | 52.85 | 58.9 | 0 | 12.26 | 52.85 |
| &_1I2_ND48_350: | | 0 | 31.67 | 46.43 | 0 | 0 | 31.67 |
| &_1I2_ND47_351: | | 0 | 36.06 | 52.72 | 0 | 0 | 36.06 |
| &_1I2_ND40_356: | | 0 | 46.01 | 55.46 | 0 | 0 | 46.01 |
| | | | | | | | |
| &_1I1_NR2_37: | | 0 | 0 | 0 | 0 | 0 | 0 |
| &_1I1_ND44_198: | | 0 | 25.92 | 35.46 | 0 | 0 | 25.92 |
| &_1I2_ND4_226: | | 0 | 15 | 16.8 | 0 | 9.47 | 15 |
| &_1I2_INV19_229: | | 0 | 0 | 0 | 0 | 0 | 0 |
| &_1I2_INV16_230: | | 0 | 33.81 | 38.77 | 0 | 0 | 33.81 |
| &_1I2_INV17_234: | | 0 | 0 | 0 | 0 | 0 | 0 |
| &_1I2_INV15_235: | | 0 | 21.75 | 25 | 0 | 0 | 21.75 |
| &_1I2_INV14_239: | | 0 | 0 | 0 | 0 | 0 | 0 |
| &_1I2_ND5_269: | | 0 | 74.54 | 79.99 | 0 | 55.68 | 74.54 |
| &_1I2_ND12_272: | | 0 | 52.98 | 55.46 | 0 | 52.98 | 52.98 |
| &_1I2_NR7_275: | | 0 | 43.83 | 50.34 | 0 | 43.83 | 43.83 |
| &_1I2_ND8_277: | | 0 | 52.98 | 55.46 | 0 | 52.98 | 52.98 |
| &_1I2_INV23_278: | | 0 | 0 | 0 | 0 | 0 | 0 |

Table 27 (continued). Circuit S400 error probabilities.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| &_1I2_ND55_281: | | 0 | 0 | 0 | 0 | 0 | 0 |
| &_1I2_INV33_285: | | 0 | 0 | 0 | 0 | 0 | 0 |
| &_1I2_ND42_289: | | 0 | 0 | 0 | 0 | 0 | 0 |
| &_1I2_ND43_290: | | 0 | 0 | 0 | 0 | 0 | 0 |
| &_1I2_NR5_295: | | 0 | 45.71 | 50.97 | 0 | 45.71 | 45.71 |
| &_1I2_ND18_321: | | 0 | 56.25 | 56.72 | 0 | 56.25 | 56.25 |
| &_1I2_ND15_323: | | 0 | 54.43 | 56.83 | 0 | 54.43 | 54.43 |
| &_1I2_ND21_330: | | 0 | 53.84 | 62.62 | 0 | 53.84 | 53.84 |
| &_1I2_INV12_348: | | 0 | 43.76 | 51.98 | 0 | 26.25 | 43.76 |
| &_1I2_ND2_349: | | 0 | 46.15 | 52.72 | 0 | 46.15 | 46.15 |
| &_1I2_INV21_353: | | 0 | 0 | 0 | 0 | 0 | 0 |
| &_1I2_INV20_354: | | 0 | 6.57 | 8.71 | 0 | 0 | 6.57 |
| &_1I2_INV18_355: | | 0 | 0 | 0 | 0 | 0 | 0 |
| | | | | | | | |
| &_1I1_INV19_64: | | 0 | 56.27 | 75.21 | 0 | 48.2 | 56.27 |
| &_1I1_ND73_160: | | 0 | 0.27 | 67.07 | 0 | 0 | 0.27 |
| &_1I1_INV27_196: | | 0 | 21.33 | 26.74 | 0 | 0 | 21.33 |
| &_1I1_ND59_199: | | 0 | 16.63 | 18.02 | 0 | 0 | 16.63 |
| &_1I2_ND10_264: | | 0 | 50.34 | 50.34 | 0 | 0 | 50.34 |
| &_1I2_ND9_267: | | 0 | 50.34 | 50.34 | 0 | 0 | 50.34 |
| &_1I2_ND7_268: | | 0 | 0 | 0 | 0 | 0 | 0 |
| &_1I2_ND11_270: | | 0 | 0 | 0 | 0 | 0 | 0 |
| &_1I2_ND38_271: | | 0 | 41.7 | 41.7 | 0 | 0 | 41.7 |
| &_1I2_INV36_273: | | 0 | 0 | 0 | 0 | 0 | 0 |
| &_1I2_NR6_286: | | 0 | 0 | 0 | 0 | 0 | 0 |
| &_1I2_ND46_292: | | 0 | 0 | 0 | 0 | 0 | 0 |
| &_1I2_INV30_293: | | 0 | 0 | 0 | 0 | 0 | 0 |

Table 27 (continued). Circuit S400 error probabilities.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| &_1I2_INV31_294: | | 0 | 0 | 0 | 0 | 0 | 0 |
| &_1I2_ND16_300: | | 0 | 1.88 | 2.24 | 0 | 1.19 | 1.88 |
| &_1I2_ND14_303: | | 0 | 38.77 | 38.77 | 0 | 0 | 38.77 |
| &_1I2_ND13_309: | | 0 | 51.36 | 51.36 | 0 | 0 | 51.36 |
| &_1I2_ND17_312: | | 0 | 14.53 | 17.13 | 0 | 9.43 | 14.53 |
| &_1I2_INV1_317: | | 0 | 8.04 | 15.39 | 0 | 5.25 | 8.04 |
| &_1I2_ND1_325: | | 0 | 56.36 | 58.66 | 0 | 0 | 56.36 |
| &_1I2_ND20_326: | | 0 | 58.43 | 58.9 | 0 | 0 | 58.43 |
| &_1I2_ND19_327: | | 0 | 31.48 | 31.84 | 0 | 0 | 31.48 |
| &_1I2_ND24_329: | | 0 | 45.77 | 49.23 | 0 | 45.77 | 45.77 |
| &_1I2_ND3_339: | | 0 | 7.51 | 7.82 | 0 | 0 | 7.51 |
| | | | | | | | |
| &_1I1_ND68_155: | | 0 | 0 | 0 | 0 | 0 | 0 |
| &_1I1_NR43_157: | | 0 | 35.6 | 35.6 | 0 | 6.16 | 35.6 |
| &_1I1_ND67_158: | | 0 | 0 | 0 | 0 | 0 | 0 |
| &_1I1_NR54_202: | | 0 | 7.13 | 10.05 | 0 | 0 | 7.13 |
| &_1I2_ND45_288: | | 0 | 0 | 0 | 0 | 0 | 0 |
| &_1I2_INV34_291: | | 0 | 0 | 0 | 0 | 0 | 0 |
| &_1I2_INV6_318: | | 0 | 55.46 | 79.96 | 0 | 52.98 | 55.46 |
| &_1I2_INV7_319: | | 0 | 11 | 12.88 | 0 | 10.53 | 11 |
| &_1I2_INV11_320: | | 0 | 33.45 | 43.87 | 0 | 27.29 | 33.45 |
| &_1I2_ND22_324: | | 0 | 26.52 | 28.29 | 0 | 11.37 | 26.52 |
| &_1I2_ND23_328: | | 0 | 0 | 0 | 0 | 0 | 0 |
| | | | | | | | |
| &_1I1_NR42_121: | | 0 | 4.93 | 4.93 | 0 | 4.93 | 4.93 |
| &_1I1_ND69_122: | | 0 | 0 | 0 | 0 | 0 | 0 |
| &_1I1_INV35_123: | | 0 | 0 | 0 | 0 | 0 | 0 |

Table 27 (continued). Circuit S400 error probabilities.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| &_1I1_ND70_124: | | 0 | 0 | 2.59 | 0 | 0 | 0 |
| &_1I1_NR41_154: | | 0 | 0 | 0 | 0 | 0 | 0 |
| &_1I1_INV36_156: | | 0 | 0 | 5.15 | 0 | 0 | 0 |
| | | | | | | | |
| &_1I1_ND66_153: | | 0 | 0 | 0 | 0 | 0 | 0 |
| | | | | | | | |
| &_1I1_ND62_145: | | 0 | 0 | 0 | 0 | 0 | 0 |

# BIBLIOGRAPHY


1) James C. Pickel and James T. Blandford, **"Cosmic Ray Induced Errors In MOS Devices"**, IEEE Transactions On Nuclear Science, NS-27,pp. 1006-1015, April 1980.

2) L. L. Sivo et al, **"Cosmic Ray-Induced Soft Errors In Static MOS Memory Cells"**, IEEE Transactions on Nuclear Science, NS-26, pp. 5042-5047, December 1979.

3) C. Guenzer et al. **"Single Event Upsets in 4K and 16K Dynamic RAM's induced by Protons and Neutrons below 100 MeV"**, IEEE Transactions on Nuclear Science NS-26, pp. 1485-1489, December 1980.

4) J. Blake and R. Mandel, **"On-orbit Observation of Harris 1K RAM's"**, IEEE Transactions on Nuclear Science NS-33, pp. 1616-1619, December 1986.

5) William K. Hartmann, **"Astronomy: The Cosmic Journey"**, Wadsworth Publishing, 1978.

6) R. Silberberg, C. H. Tsao, and J. R. Letaw, **"Origin, Propagation And Acceleration Of Cosmic Rays"**, NRL Report.

7) J. H. Adams, R. Silberberg, and C. H. Tsao, **"Cosmic Ray Effects On Microelectronics, Part I: The Near-Earth Particle Environment"**, NRL Memorandum Report 4506, August 1981.

8) James C. Pickel and James T. Blandford, **"Cosmic Ray Induced Errors In MOS Memory Cells"**, IEEE Transactions On Nuclear Science, NS-25,pp. 1166-11170, December 1978.

9) P. Goel, **"An Implicit Enumeration Algorithm to Generate Tests for Combinational Logic Circuits"**, IEEE Transactions on Computers, C-30, pp. 215-222, March 1981.

10) H. Fujiwara, and T. Shimono, **"On The Acceleration of Test Generation Algorithms"**, Dig., 13th Annu. Int. Symp. Fault-Tolerant Comput., pp. 98-105, June 1983.

11) S. E. Diehl-Nagle, J. E. Vinson, and E. L. Peterson, **"Single Event Upset Rate Predictions For Complex Logic Systems"**, IEEE Transactions on Nuclear Science, NS-31, pp. 1132-1138,December 1984.

12) MOSIS, **"CMOSN Standard Cell Library"**.

13) Wayne Angevine, **"An Introduction TO EDIF"**, Semicustom Design Guide, pp. 26-35, 1987.

14) Franc Brglez, David Bryan, and Krzysztof Kozminski, **"Combinational Profiles of Sequential Benchmark Circuits"**, 1989 Int. Symp. on Circuits and Systems, May 1989.

15) R. G. Bennetts, **"Design Of Testable Logic Circuits"**, Addison-Wesley Publishing, 1984.

16) J. R. Srour et al, **"Radiation Effects On And Dose Enhancement Of Electronic Materials"**, Noyes Publications, 1984.

17) Norman G. Einspruch, **"VLSI Electronics Microstructure Science Volume 7"**, Academic Press, 1983.

18) Glen G. Langdon, Jr., **"Computer Design"**, Computeach Press, 1982.

19) Neil Weste, Kamran Eshraghian, **"Principles Of CMOS VLSI Design. A Systems Perspective"**, Addison Wesley Publishing, 1988.

20) Paul Stanford, Paul Mancuso, **"EDIF Electronic Design Interchange Format Version 2 0 0"**, Electronic Industries Association, 1990.

21) Paul Stanford, Paul Mancuso, **"EDIF Introduction To EDIF Volume 1"**, Electronic Industries Association, 1988.

22) Paul Stanford, Paul Mancuso, **"EDIF EDIF Connectivity Volume 2"**, Electronic Industries Association, 1989.

23) Esther Marx, Hart Switzer, Mike Waters, **"EDIF Readers/Writers Handle Data Transfer Between CAE Systems"**, EDN, March 4, 1987.

24) Gorden P. Ansell, Joe S. Tirado, **"CMOS In Radiation Environments"**, VLSI Systems Design, Sept. 1986.

25) J. Zhoutynedek, **"Technical Support Package On Estimating Rates Of Single Event Upsets"**, NASA JPL Tech Brief, Volume 12, No. 10, Item #152, November 1988.

26) S. E. Diehl, J. E. Vinson, B. D. Shafer, and T. M. Mnich, **"Considerations For Single Event Immune VLSI Logic"**, IEEE Transactions on Nuclear Science, NS-30, pp. 4501-4507, December 1983.

27) Y. Savaria, N. C. Rumin, J. F. Hayes, and V. K. Agarwal, **"Characterization Of Soft Error Sources"**, Report No. 84-11R Department of Electrical Engineering McGill University, 1984.

28) Viewlogic Systems, Inc., **"Workview 4.0"**, 1990.

# VITA

Kenneth Chris Holland was born in Franklin, Virginia on December 18, 1961. In 1983, he enrolled at Paul D. Camp Community college and graduated summa cum laude with a A.A.S in Electrical Engineering Technology in 1984. He then continued his education at Old Dominion University, and received a B.S. cum laude in Electrical Engineering Technology in 1986. After spending two years working for Virginia Power, he joined the graduate school at Virginia Tech to pursue a M.S. degree in Electrical Engineering.

His current interests include VLSI design and microprocessor applications. His personal interests include his 7 month old baby daughter, Rachael, his wife of 5 years, Kathy, backpacking, fishing, astronomy, and photography.