Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.

Exercises

on the materials introduced in this chapter.
2.1 Introduction

MOS Transistor

Theory

In practice, the MOS transistor is used in various applications, including amplifiers, power supplies, and digital circuits. The MOS transistor operates by controlling the flow of current through a channel, which is controlled by a gate voltage. There are two types of MOS transistors: enhancement mode and depletion mode.

Enhancement Mode MOS Transistor

In enhancement mode, the drain current is controlled by the gate voltage, and no current flows through the transistor until a threshold voltage is applied. The threshold voltage is the minimum gate voltage required to turn on the transistor. The drain-source current (ID) is given by the equation:

ID = k(VGS - VTH)

where k is the transconductance, VGS is the gate-source voltage, and VTH is the threshold voltage.

Depletion Mode MOS Transistor

In depletion mode, the drain current flows through the transistor even when no voltage is applied to the gate. The drain current is controlled by the gate voltage, which is inversely proportional to the drain current. The drain-source current (ID) is given by the equation:

ID = k(VTH - VGS)

where k is the transconductance, VGS is the gate-source voltage, and VTH is the threshold voltage.
Exercise 2.7

The expression for the current in the transistor is

\[ i_C = \frac{v_{CE}}{r_C} \]

where \( v_{CE} \) is the collector-emitter voltage and \( r_C \) is the collector resistance.

Show the current in the transistor in terms of \( V \) and \( I \).

\[ i_C = \frac{v_{CE}}{r_C} \]

Exercise 2.9

Consider an NPN transistor in a low-frequency circuit with \( V_T \) of 0.7 V.

2.1

\[ I_C = \frac{V_T}{r_C} \]

2.2

\[ I_C = \frac{V_T}{r_C} \]

2.3

\[ I_C = \frac{V_T}{r_C} \]

2.4

\[ I_C = \frac{V_T}{r_C} \]

Exercises
6.4 Design a static CMOS circuit to compute \( F = (A \cdot B)(C + D) \) with least delay. Each input can present a maximum of 30 \( \lambda \) of transistor width. The output must drive a load equivalent to 500 \( \lambda \) of transistor width. Choose transistor sizes to achieve least delay and estimate this delay in \( \tau \).

6.5 Figure 6.76 shows two series transistors modeling the pull-down network of 2-input NAND gate.

\[ a) \quad \text{Plot } I \text{ vs. } \tau \text{ using ideal transistor models for } 0 \leq A \leq 1, B = Y = 1, T = 0, \]
\[ \beta = 1. \text{ On the same axes, plot } I \text{ vs. } B \text{ for } 0 \leq B \leq 1, A = 1. \text{ How will you need to solve for } x? \text{ This can be done best numerically.} \]

\[ b) \quad \text{Using your results from (a), explain why the inner input of a 2-input NAND gate has a slightly greater logical effort than the outer input.} \]

6.6 What is the logical effort of an OR-AND-INVERT gate at either of the OR terminals? At the AND terminal? What is the parasitic delay if only diffusion capacitance on the output is counted?

6.7 Simulate a 3-input NOR gate in your process. Determine the logical effort and parasitic delay from each input.

6.8 Using the data sheet from Figure 4.25, find the rising and falling logical effort and parasitic delay of the XT 2-input NAND gate from the \( A \) input.

6.9 Repeat Exercise 6.8 for the \( B \) input. Explain why the results are different for the different inputs.

6.10 Sketch HI-skew and LO-skew 3-input NAND and NOR gates. What are the logical efforts of each gate on its critical transition?

6.11 Derive a formula for \( g_{ds}, g_{ds}, \) and \( g_{ds} \) for HI-skew and LO-skew 3-input NAND gates with a skew factor of \( s < 1 \) (i.e., the noncritical transistor is \( s \) times normal size) as a function of \( s \) and \( k \).

6.12 Design an asymmetric 3-input NOR gate that favors a critical input over the other two. Choose transistor sizes so the logical effort on the critical input is 1.5. What is the logical effort of the noncritical inputs?

6.13 Prove that the \( P/N \) ratio that gives lowest average delay in a logic gate is the square root of the ratio that gives equal rise and fall delays.

6.14 Let \( p(g, \rho) \) be the best stage effort of a path if one is free to add extra buffers with a parasitic delay \( \rho \) and logical effort \( g \). For example, Section 4.3.3 showed that \( \rho(1, 1) = 3.59 \). It is easy to make a plot of \( p(1, \rho) \) by solving Eq. (4.19) numerically; this gives the best stage effort of static CMOS circuits where the inverter has a parasitic delay of \( \rho \). Prove the following result, which is useful for determining the best stage effort of domino circuits where buffers have lower logical efforts:

\[ p(g, \rho) = g \rho(1, g) \]
6.15 Simulate a fan-out-of-4 inverter. Use a unit-sized nMOS transistor. How wide must the pMOS transistor be to achieve equal rising and falling delays? What is the delay? How wide must the pMOS transistor be to achieve minimum average delay? What is the delay? How much faster is the average delay?

6.16 Many standard cell libraries choose a PIV ratio for an inverter in between that which would give equal rising and falling delays and that which would give minimum average delay. Why is this done?

6.17 A static CMOS NOR gate uses 4 transistors, while a pseudo-nMOS NOR gate uses only 3. Unfortunately, the pseudo-nMOS output does not swing rail to rail. If both the inputs and their complements are available, it is possible to build a 3-transistor NOR that swings rail to rail without using any dynamic nodes. Show how to do it. Explain any drawbacks of your circuit.

6.18 Sketch pseudo-nMOS 3-input NAND and NOR gates. Label the transistor widths. What are the rising, falling, and average logical efforts of each gate?

6.19 Sketch a pseudo-nMOS gate that implements the function

\[ F = M(\beta + C + D) + E \cdot F \cdot G. \]

6.20 Design an 8-input AND gate with an electrical effort of 6 using pseudo-nMOS logic. If the parasitic delay of an n-input pseudo-nMOS NOR gate is \( (4n + 2)/9 \), what is the path delay?

6.21 Simulate a pseudo-nMOS inverter in which the pMOS transistor is half the width of the nMOS transistor. What are the rising, falling, and average logical efforts? What is \( V_{OH} \)?

6.22 Repeat Exercise 6.21 in the FS and SF process corners.

6.23 Sketch a 3-input symmetric NOR gate. Size the inverters so that the pull-down is four times as strong as the net worst-case pull-up. Label the transistor widths. Estimate the rising, falling, and average logical efforts. How do they compare to a static CMOS 3-input NOR gate?

6.24 Sketch a 2-input symmetric NAND gate. Size the inverters so that the pull-down is four times as strong as the net worst-case pull-up. Label the transistor widths. Estimate the rising, falling, and average logical efforts. How do they compare to a static CMOS 3-input NAND gate?

6.25 Compare the average delays of a 2, 4, 8, and 16-input pseudo-NMOS and SFPL NOR gate driving a fanout of 4 identical gates.

6.26 Sketch a 3-input CVSL OR/NOR gate.

6.27 Sketch dynamic footed and unfooted 3-input NAND and NOR gates. Label the transistor widths. What is the logical effort of each gate?

6.28 Sketch a 3-input dual-rail domino OR/NOR gate.

6.29 Sketch a 3-input dual-rail domino majority/minority gate. This is often used in domino full adder cells. Recall that the majority function is true if more than half of the inputs are true.

6.30 Compare a standard keeper with the noise tolerant precharge device. Large pMOS transistors result in a higher \( V_{PP} \) (and thus better noise margins) but more delay. Simulate a 2-input footed NAND gate and plot \( V_{IL} \) vs. delay for various sizes of keepers and noise tolerant precharge transistors.

6.31 Design a 4-input footed dynamic NAND gate driving an electrical effort of 1. Estimate the worst charge sharing noise as a fraction of \( V_{DD} \) assuming that diffusion capacitance on uncontacted nodes is about half of gate capacitance and on contacted nodes it equals gate capacitance.

6.32 Repeat Exercise 6.31, generating a graph of charge sharing noise vs. electrical effort for \( b = 0, 1, 2, 4, \) and 8.

6.33 Repeat Exercise 6.31 if a small secondary precharge transistor is added on one of the internal nodes.

6.34 Perform a simulation of your circuits from Exercise 6.31. Explain any discrepancies.

6.35 Design a domino circuit to compute \( F = (A \cdot B)(C + D) \) as fast as possible. Each input may present a maximum of 30 \( \lambda \) of transistor width. The output must drive a load equivalent to 50 \( \lambda \) of transistor width. Choose transistor sizes to achieve least delay and estimate this delay in \( \tau \).

6.36 Redesign the memory decoder from Section 4.3.4 using footed domino logic. You can assume you have both true and complementary monotonic inputs available, each capable of driving 5 unit transistors. Label gate sizes and estimate the delay.

6.37 Sketch an NP Domino 8-input AND circuit.

6.38 Sketch a 4:1 multiplexer. You are given four data signals \( D0, D1, D2, \) and \( D3 \), and two select signals, \( S0 \) and \( S1 \). How many transistors does each design require?

a) Use only static CMOS logic gates.

b) Use a combination of logic gates and transmission gates.

6.39 Sketch 3-input XOR functions using each of the following circuit techniques:

a) static CMOS

b) pseudo-nMOS

c) dual-rail domino

d) CPL

e) EEPL

f) DCVSG
uates the timing constraints across many stages of a pipeline rather than isolating them at each stage. Not all timing analyzers handle latches gracefully, especially when there are different amounts of clock skew between different clocks [Harris94]. Two-phase latches have been used in the Alpha 21064 and 21164 [Gronowski95], PowerPC 603 [Genosa94], and many other IBM designs.

Pulsed latches have low sequencing overhead. They present a tradeoff when choosing pulse width. A wide pulse permits more time borrowing and skew tolerance, but makes

min-delay constraints harder to meet. Pulsed latches are also popular because they can be modeled as fast flip-flops with a lousy hold time from the point of view of a timing analyzer (or novice designer) if intentional time borrowing is not permitted. The min-delay problems can be largely overcome by mixing pulsed latches for long paths and flip-flops for short paths. Unfortunately, many real designs have paths in which the propagation delay is very long but the contamination delay is very short, making robust design more challenging. Pulsed latches have been used on Itanium 2 [Naflziger02], Pentium 4 [Koide01], Athlon [Drape97], and CRAY 1 [Unger86]. However, they can wreak havoc with conventional commercially available design flows and are best avoided unless the performance requirements are extreme.

Domino circuits are widely used in high-performance systems because they are 1.5–2x faster than static CMOS. Traditional domino circuits with latches have high sequencing overhead that wastes much of the potential speedup, so most designers have moved to skew-tolerant techniques. Static-to-domino interfaces impose hard edges and the associated sequencing overhead, motivating the use of domino throughout critical loops. Single-rail domino only computes noninverting functions, so most loops require dual-rail domino that consumes more area, wiring, and power and is ill-suited to wide NORs. An alternative is to push the inverting functions to the end of the pipeline, using single-rail domino through most of the pipeline and nonmonomorphic static logic at the end. The area savings comes at the cost of one hard edge in the cycle.

Four-phase or delayed reset skew-tolerant domino circuits work well in datapaths because the clock generation is relatively simple. Self-resetting domino is ideally suited to memories where the decoder power consumption is greatly reduced by only precharging the output that switched and where the number of unique circuits to design is relatively small. It was also used on the Pentium 4, but was costly in terms of designer effort because so many pulse constraints must be satisfied.

Clock-delayed domino is used in wide dynamic NOR functions where the power consumption of pseudo-nMOS is unacceptable. For example, it is an important technique for CAMs and PLAs. The delay matching raises an unpleasant tradeoff between speed and correct operation, requiring significant margin for safe operation. The risk of race conditions deters many designers from using it more widely. Annihilation gates and complementary signal generators are interesting special cases in which no clock gate delay at all is required. Output prediction logic is also interesting, but has yet to be proven in a large application.

When inputs to a system arrive asynchronously, they cannot be guaranteed to meet setup or hold times at clocked elements. Even if we do not care whether an input arrived in one cycle or the next, we must ensure that the clocked element produces a valid logic level. Unfortunately, if the element samples a changing input at just the wrong time, it may produce a metastable output that remains invalid for an unbounded amount of time. The probability of metastability drops off exponentially with time. Systems use synchronizers to sample the asynchronous input and hold it long enough to resolve to a valid logic level with very high probability before passing it onward.

Most synchronous VLSI systems use opaque sequencing elements to separate one token from the next. In contrast, many optical systems transmit data as pulses separated in time. As long as the propagation medium does not disperse the pulses too badly, they can be recovered at a receiver. Similarly, if a VLSI system has low dispersion, i.e., nearly equal contamination and propagation delays, it can send more than one wave of data without explicit latching. Such wave pipelining offers the potential of high throughput and low sequencing overhead. However, it is difficult to perform in practice because of the variability of data delays.

**Exercises**

Use the following timing parameters for the questions in this section.

| Table 7.5 | Sequencing element parameters |
| --- | --- | --- | --- | --- | --- | --- |
| Setup Time | eR-to-Q Delay | D-to-Q Delay | Contamination Delay | Hold Time |
| Flip-flops | 65 ps | 50 ps | 70 ps | 35 ps | 30 ps |
| Latches | 25 ps | 50 ps | 40 ps | 35 ps | 30 ps |

7.1 For each of the following sequencing styles, determine the maximum logic propagation delay available within a 500 ps clock cycle. Assume there is zero clock skew and no time borrowing takes place.

a) Flip-flops
b) Two-phase transparent latches
c) Pulsed latches with 80 ps pulse width

7.2 Repeat Exercise 7.1 if the clock skew between any two elements can be up to 50 ps.

7.3 For each of the following sequencing styles, determine the minimum logic contamination delay in each clock cycle (or half-cycle, for two-phase latches). Assume there is zero clock skew.

a) Flip-flops
b) Two-phase transparent latches with 50% duty cycle clocks
c) Two-phase transparent latches with 60 ps of nonoverlap between phases
d) Pulled latches with 80 ps pulse width

7.4 Repeat Exercise 7.3 if the clock skew between any two elements can be up to 50 ps.

7.5 Suppose one cycle of logic is particularly critical and the next cycle is nearly empty. Determine the maximum amount of time the first cycle can borrow into the second for each of the following sequencing styles. Assume there is zero clock skew.

a) Flip-flops
b) Two-phase transparent latches with 50% duty cycle clocks
c) Two-phase transparent latches with 60 ps of nonoverlap between phases
d) Pulled latches with 80 ps pulse width

7.6 Repeat Exercise 7.5 if the clock skew between any two elements can be up to 50 ps.

7.7 Prove EQ(7.17).

7.8 Consider a flip-flop built from a pair of transparent latches using nonoverlapping clocks. Express the setup time, hold time, and clock-to-Q delay of the flip-flop in terms of the latch timing parameters and $t_{nonoverlap}$.

7.9 For the path in Figure 7.97, determine which latches borrow time and if any setup time violations occur. Repeat for cycle times of 1200, 1000, and 800 ps. Assume there is zero clock skew and that the latch delays are accounted for in the propagation delay $\Delta t$.

a) $\Delta t_1 = 550$ ps; $\Delta t_2 = 580$ ps; $\Delta t_3 = 450$ ps; $\Delta t_4 = 200$ ps
b) $\Delta t_1 = 300$ ps; $\Delta t_2 = 600$ ps; $\Delta t_3 = 400$ ps; $\Delta t_4 = 550$ ps

7.10 Determine the minimum clock period at which the circuit in Figure 7.98 will operate correctly for each of the following logic delays. Assume there is zero clock skew and that the latch delays are accounted for in the propagation delay $\Delta t$.

a) $\Delta t_1 = 300$ ps; $\Delta t_2 = 400$ ps; $\Delta t_3 = 200$ ps; $\Delta t_4 = 350$ ps

7.11 Repeat Exercise 7.10 if the clock skew is 100 ps.

7.12 Label the timing types of each signal in the circuit from Figure 7.97. The flip-flop is constructed with back-to-back transparent latches—the first controlled by $d\bar{c}\bar{k}$ and the second by $d\bar{c}$.

7.13 Using a simulator, compare the $D$-to-$Q$ propagation delays of a conventional dynamic latch from Figure 7.17(d) and a TSPC latch from Figure 7.10(a). Assume each latch is loaded with a fanout of 4. Use 4 $\lambda$-wide clocked transistors and tune the other transistor sizes for least propagation delay.

7.14 Using a simulator, find the setup and hold times of a TSPC latch under the assumptions of Exercise 7.13.

7.15 Determine the maximum logic propagation delay available in a cycle for a traditional domino pipeline using a 500 ps clock cycle. Assume there is zero clock skew.

7.16 Repeat Exercise 7.15 if the clock skew between any two elements can be up to 50 ps.

7.17 Determine the maximum logic propagation delay available in a cycle for a four-phase skew-tolerant domino pipeline using a 500 ps clock cycle. Assume there is zero clock skew.

7.18 Repeat Exercise 7.17 if the clock skew between any two elements can be up to 50 ps.

7.19 How much time can one phase borrow into the next in Exercise 7.18 if the clocks each have a 50% duty cycle?

7.20 Repeat Exercise 7.18 if the clocks have a 65% duty cycle.

7.21 Design a fast-pulsed latch. Make the gate capacitance on the clock and data inputs equal. Let the latch drive an output load of four identical latches. Simulate your latch and find the setup and hold times and clock-to-Q propagation and contamination delays. Express your results in FO4 inverter delays.
7.22 Simulate the worst-case propagation delay of an 8-input dynamic NOR gate driving a fanout of 4. Report the delay in all 16 design corners (voltage, temperature, nMOS, pMOS). Also determine the delay of a fanout-of-4 inverter in each of these corners. By what percentage does the absolute propagation delay of the NOR gate vary across corners? By what percentage does its normalized delay vary (in terms of FO4 inverters)? Comment on the implications for circuits using matched delays.

7.23 A synchronizer uses a flip-flop with $t_f = 54$ ps and $T_o = 21$ ps. Assuming the input toggles at 10 MHz and the setup time is negligible, what is the minimum clock period for which the mean time between failures exceeds 100 years?

7.24 Simulate the synchronizer flip-flop of Figure 7.82 and make a plot analogous to Figure 7.80. From your plot, find $\Delta t_{\text{up}}$, $\Delta t_{\text{down}}$, and $T_o$.

7.25 Inferior Circuits, Inc., wants to sell you a perfect synchronizer that they claim never produces a metastable output. The synchronizer consists of a regular flip-flop followed by a high-gain comparator that produces a high output for inputs above $0.25 \cdot V_{\text{DD}}$ and a low output for inputs below that point. The VP of marketing argues that even if the flip-flop enters metastability, its output will hover near $V_{\text{DD}}/2$ so the synchronizer will produce a good high output after the comparator. Why wouldn’t you buy this synchronizer?

8.1 Introduction

The manner in which you go about designing a particular system, chip, or circuit can have a profound impact on both the effort expended and the outcome of the design. IC designers have developed and adapted strategies from allied disciplines such as software engineering to form a cohesive set of principles to increase the likelihood of timely, successful designs. We will explore these principles in this chapter. While the broad principles of design have not changed in decades, the details of design styles and tools have evolved along with advances in technology and increasing levels of productivity. This chapter represents current CMOS design methods and provides an overview of a complex subject that could fill many books on its own. We encourage you to actively monitor the companies discussed and literature cited in the chapter to track the latest developments in this rapidly changing field.

As introduced in Section 1.6, an integrated circuit can be described in terms of three domains: (1) the behavioral domain, (2) the structural domain, and (3) the physical domain. The behavioral domain specifies what we wish to accomplish with a system. For instance, at the highest level, we might want to build an ultra-low-power radio for a distributed sensor network. The structural domain specifies the interconnection of components required to achieve the behavior we desire. Again, by way of example, our sensor radio might require a sensor, a radio transceiver, a processor, and memory (with software), and a power source connected in a particular manner. Finally, the physical domain specifies how to arrange the components in order to connect them, which in turn allows the required behavior. Our example might start with the specification for an enclosure to hold the device, followed by a succession of physical drawings or specifications that may culminate in a description of geometry to be used to define a chip. Design flows from behavior to structure and ultimately to a physical implementation via a set of manual or automated transformations. At each transformation, the correctness of the transformation is tested by comparing the pre- and post-transformation design. For instance, if a power level is specified in the original behavioral description of the sensor radio, a test is run on the design in the structural domain with feedback from the physical domain to ensure this design goal is met.

In each of these domains there are a number of design options that can be selected to solve a particular problem. For instance, at the behavioral level, we can choose the wireless standard and the format in which data is transmitted by the sensor radio. In the structural