CHAPTER
11

Power Grid and Clock Design

CHAPTER OUTLINE

11.1 Introduction

In this chapter, we address two chip-level design challenges that are dominated by interconnect issues: power and clock distribution. Here, the word *power* is intended to mean the power and ground distribution systems, which include V_{DD} and Gnd (or V_{SS}). The power supply, up to this point in the book, has been assumed to be a constant value that is available everywhere on the chip. However, it must actually be routed from the power pads to the gates, memory circuits, and all other functions on the chip. Furthermore, its value fluctuates over time, depending on the switching activity on the chip. These variations in the voltage must be tolerated by the gates on the chip. The same applies to the clock signal. It has been assumed to be available at all required points on the chip with a well-defined clock period. However, it too must be routed from the clock source or root node to the appropriate destinations inside the chip. It also has some variability in its period and arrival times across the chip that must be tolerated by the logic circuits.

Power distribution and clock design are now complex tasks that must be done with great care in deep submicron technology. Gone are the days of simply laying out metal tracks for the power supply or running a clock signal to all the flip-flops in the design. Today, many issues come into play and affect the integrity of the power grid or the clock network. This chapter addresses many of these issues, although it is difficult to completely describe all of the issues and their interactions

in a single chapter. The chapter will highlight the interaction of the power system with the clock design. Many design groups work on power system and clock design together so that the complete problem is properly addressed. We will also address clock generation and phase-lock loop (PLL) circuits as part of this chapter.

11.2 Power Distribution Design

Power distribution, when considered in its entirety, actually involves on-chip and off-chip issues starting from off-chip dc-dc converters, printed circuit boards (PCBs), power planes, packages, sockets, power pins or solder bumps, and finally the connection to the gates. Proper power grid design requires interaction of system designers, thermal designers, system architects, board designers, and chip designers. The problem demands global optimization rather than localized chip-level optimization. This section will focus on the chip-level issues, but the reader should keep in mind that this is only one part of the complete design problem.

Much of the complexity of power distribution systems arises due to the large number of transistors on a chip, the *RLC* nature of interconnect, the frequency of operation, and the current demand of high-speed circuits today. The highest frequency of operation is well over 1 GHz, while power has exceeded 100 W and current levels have reached 100 A in state-of-the-art designs. Meanwhile, supply voltage scaling to reduce power has led to V_{DD} values of 1.2 V and below. Fluctuations in the power system at these levels lead to timing variations and, eventually, design failure. Allowable voltage noise in the supply is budgeted at $\pm 10\% \times (V_{DD} - \text{Gnd})$ as a rule of thumb.

In order to manage fluctuations, the total impedance seen from the V_{DD} pads to the gate connection must be controlled based on the needed current. The target impedance of the power grid as seen from the pads is given by

$$Z = \frac{V_{DD} \times (\text{fractional noise budget})}{I_{DD}} \tag{11.1}$$

For a supply voltage of $V_{DD} = 1.2$ V and a supply current of $I_{DD} = 100$ A, $Z = 1.2\,\text{V} \times (10\%)/100\,\text{A} = 1.2\,\text{m}\Omega$. This is an incredibly small number to achieve over the gigahertz frequency range and makes power distribution design very challenging.

The noise on the supply is primarily due to *IR* drop and *Ldi/dt* as will be explained later. At this stage, it is sufficient to realize that voltage drops occur on the power grid due to the resistance and inductance along the current path. These voltage drops affect clock skew, gate performance, and clock and PLL timing jitter.

There are also issues of power grid electromigration that affect long-term reliability. The metal migrates due to the eroding effects of large currents and eventually breaks. Ultimately, the power system design will limit the system performance and may lead to failures in the field. Therefore, it must be designed carefully to avoid such potential problems that are collectively called *power integrity* issues.

One approach that has been used in the past is to overdesign the system to avoid any potential failures. *Overdesign* is a term used often by designers when design risks

are removed by going well beyond the needed safeguards and recommended guidelines. One example of this is the use of decoupling capacitance on the chip to control *IR* and *Ldi/dt*. In the case of overdesign, most of the available area would be committed to decoupling capacitance rather than to devices. However, there is a large area penalty associated with this overdesign (and an even larger one for underdesign—nonworking chips!) so it is important to understand the key issues and make the proper tradeoffs wherever possible. With this brief overview, we now describe the main issues in power system design and then examine a few basic structures of the power distribution system.

11.2.1 *IR* Drop and *Ldi/dt*

The conditions contributing to the complexity of power distribution systems are primarily due to interconnect and its impact on chip performance. *IR* drop due to resistance on V_{DD} lines impacts overall timing and functionality. These effects are made worse by the presence of *Ldi/dt* voltage variations at package pins due to the increased rate of change of current in high-speed designs. Together, the total voltage drop at any point in the power grid is given by

$$V = IR + L\frac{di}{dt} \tag{11.2}$$

We consider each of these effects separately starting with *IR* drop. The basic concept of *IR* drop is illustrated in Figure 11.1, which depicts two large buffers connected to a resistive power supply. Initially, all voltage levels in the power grid are at V_{DD}. As the large driver, inv2, begins to switch, the demand for current reduces the voltage in the power grid. Specifically, the wire resistance creates voltage drops that increase as the current moves from the external supply toward inv2. The voltage remains relatively high near the V_{DD} connection at the periphery of the chip, and drops by ΔV at the connection to inv2. In practice, *IR* drop is caused by simultaneous switching of clock buffers, bus drivers, memory decoder drivers, etc. These simultaneous

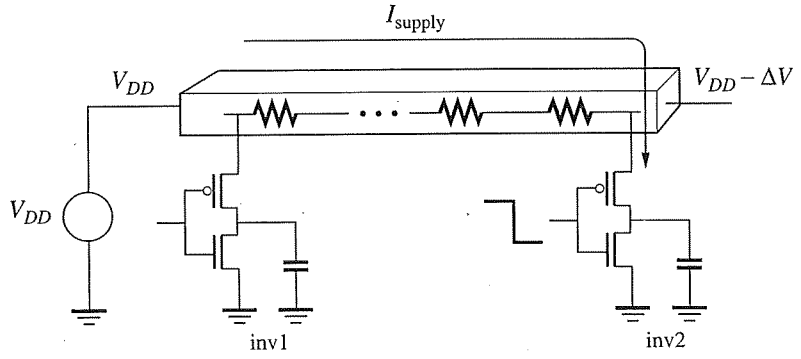


Figure 11.1
IR drop in power distribution system.

switching events can occur anywhere on the chip and, therefore, all regions are susceptible to *IR* drop violations. The ground grid is subject to the same type of problem when the outputs switch low, except that the value will increase in voltage. This is sometimes referred to as *ground bounce.*

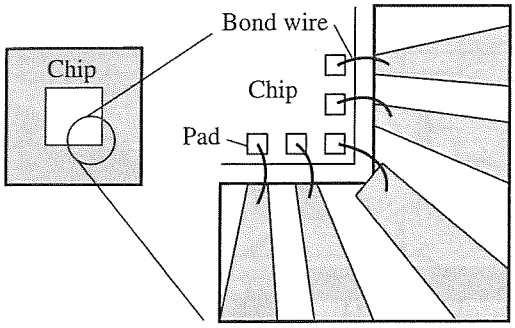
Voltage drops on the power grid primarily affect timing. The effect of *IR* drop on performance can be significant. *IR* drop reduces the drive capability of the gates and increases the overall delay. Typically, a 5% drop in supply voltage can affect delay by 10–15%, or more. Such an increase becomes serious when managing clock skews in the range of 100 picoseconds. In the previous chapter, we found that path delays are no longer predictable due to interconnect issues. This problem is exacerbated by *IR* drop. Ideally, timing calculations should account for worst-case *IR* drop to ensure a working design. The analysis can be carried out using process corner simulation, as described in Chapter 3.

IR drop also compromises the noise margins of logic gates, due not only to voltage drop in the power grid, but also to the increase in voltage in the ground grid. Once the noise margins drop below the budgeted amount, typically 10%, the design is not guaranteed to operate properly. Over the years, supply voltage has been shrinking to avoid transistor punch-through conditions, hot-electron effects, and device breakdown. More recently, V_{DD} scaling has been driven by the need to reduce power dissipation rather than the other issues. Either way, this has resulted in smaller and smaller noise margins. With *IR* drop, the margins are reduced even further which makes it more difficult to manage the effects of interconnect coupling noise in a multimillion-transistor design.

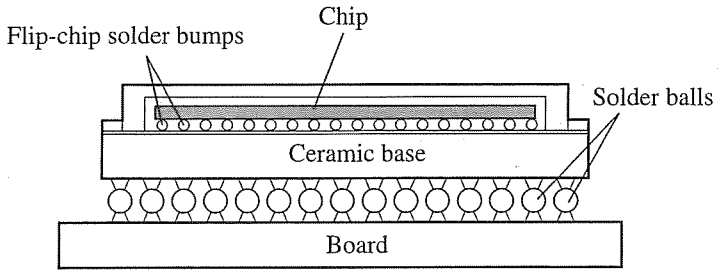
Now consider the Ldi/dt term of Equation (11.2). This is another source of voltage drop in the power supply due to package pin inductance—typically around 1–2 nH. The inductance arises from the bonding wire used to connect the chip I/O pads to the lead frame in a traditional dual-inline package (DIP), as in Figure 11.2a. While the inductance L has not changed significantly over the years, the value of di/dt has continued to increase and the supply voltage has been decreasing from 5 V to 3.3 V, 2.5 V, 1.8 V, and recently 1.2 V and below. These trends have reached a point where the Ldi/dt drop can contribute significantly to an overall voltage drop in the power grid, especially in peak demand situations.

Today, many companies have moved to a ceramic *ball-grid array (BGA)* packaging due to the high number of chip I/O and power and ground connections needed. In this packaging technology, solder balls or *bumps* replace the pins of the DIP as shown in Figure 11.2b. The bumps can be placed anywhere in the chip area and allow 300 to 500 I/O connections. This is a more expensive solution but must be used whenever the I/O count exceeds the capacity of the previous packaging technologies. The inductance of each solder bump is of the order of 0.1 nH.

Consider the example of Ldi/dt on the BGA package option shown in Figure 11.3. Here, the power supply is driving a few large inverters and we assume that the resistance of the line is zero. The inductances of interest are the ones due to the solder bump and via connecting from the bump and the package plane (providing clean power). A current level of 25 mA is supplied by V_{DD} and flows into the circuit over a 100 ps time interval. At this same time, some of the buffers are discharging



(a) Dual-inline packaging



(b) Ball grid array packaging

Figure 11.2
Chip packaging options.

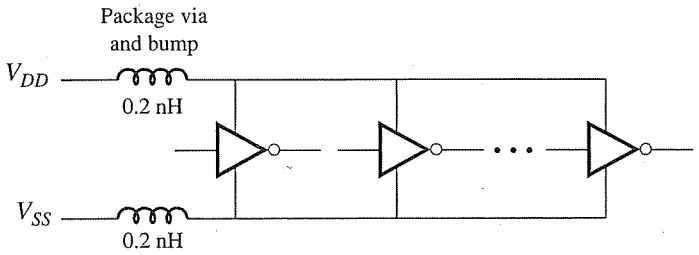


Figure 11.3
Effect of off-chip inductance.

through V_{SS} with roughly the same level of current in the same time period. If the bump and via generate 0.2 nH of inductance, then the total voltage drop due to the inductance on both rails is

$$V_L = 2 \times L\frac{di}{dt} = 2 \times 0.2 \text{ nH} \times \frac{25 \text{ mA}}{100 \text{ ps}} = 100 \text{ mV}$$

This is a significant drop considering that the supply voltage may only be 1.2 V. Furthermore, this is only one of the voltage drops to consider. If the inductance of the

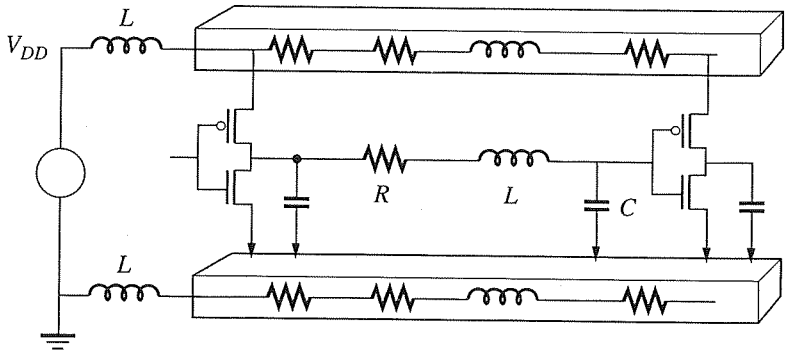


Figure 11.4
Combined resistive and inductive effects of power/ground system.

power system is included, we will begin to observe *Ldi/dt* effects due to the power grid itself, in addition to the pin inductance. We also neglected the resistance of the power grid, and the fact that many other buffers may be switching at the same time.

The overall voltage drop due to both *Ldi/dt* and *IR*, which are both dynamic phenomena, requires the modeling approach shown in Figure 11.4. The packaging inductances, power grid inductances, and power grid resistances must all be considered together with an *RLC* model of interconnect. Of particular concern are situations that give rise to simultaneous switching noise (SSN) where many buffers switch together producing large *IR* drops and ground bounce.

11.2.2 Electromigration

The high currents experienced in the power grid also induce electromigration (EM) effects where metal lines begin to migrate and eventually break during the operation of the chip. The process involves the migration of metal molecules due to the high current densities and narrow line widths leading to a short or open in the metal line. Once this occurs, the chip may not function as expected or may no longer meet the timing specifications. Existing cracks or other imperfections in the metal lines are particularly prone to this problem since they are already weak links in the power system. Failures due to EM can be catastrophic because they occur in the field when the chip is in a system and in a customer's hands.

An example of an electromigration failure is shown in Figure 11.5. In the past, low current densities and wide metal lines, combined with special processing, reduced the likelihood of EM. As we exceeded speeds of 100 MHz, and reached geometries below 0.8 μm, the potential for EM problems increased in aluminum. One of the reasons for switching to copper was its superior performance with respect to electromigration. Cu is approximately 10 times better in terms of EM than Al, but Cu diffuses quickly through oxide and it was not used in ICs because of this problem. However, advances in processing technology to solve this problem led to the eventual switch from Al to Cu. Initial EM results of Cu indicate that it is slightly better than Al, but large improvements are expected as the technology matures. Unfortunately, today the copper vias experience a greater degree of EM failures compared to the tungsten vias used with aluminum.

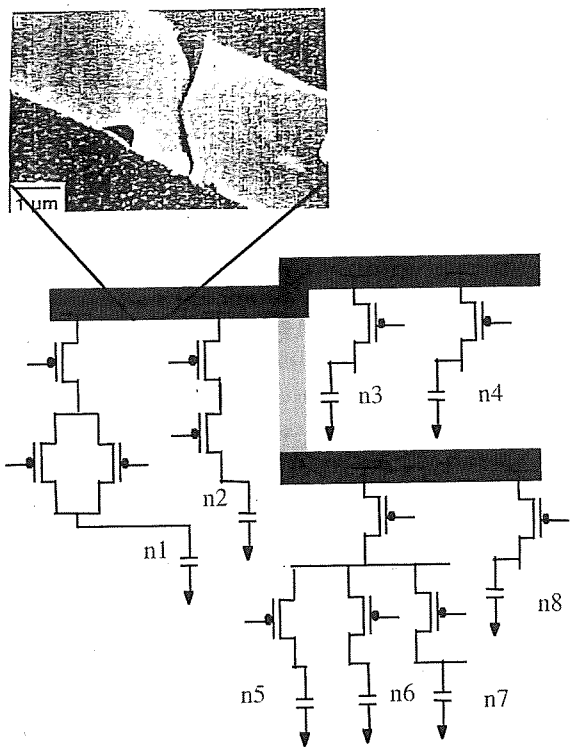


Figure 11.5
Electromigration failure in power bus.

Electromigration in the power grid is a function of the average current flow in metal lines and vias. Current flows primarily in one direction in each power bus, since it is delivering current from the power pad to the gate in the case of V_{DD} (see Figure 11.1). Therefore, it is called a dc phenomenon. To assess the EM tolerance of a piece of metal, we can examine its average dc current density, J, and compare it to some allowable level, J_{max}, when EM failure occurs. Since metal lines vary in height and material properties at different levels in the design, each metal layer may have a different failure criteria.

It is useful to characterize the lifetime of a chip due to long-term reliability problems through a metric called the *mean time to failure (MTTF)*. If the value of MTTF is 5–10 times greater than the expected lifetime of the product in which the chip is embedded, then we can safely ship the design to customers. However, if the chip lifetime is less than the product lifetime, improvements to the design must be made. Black's law is used to estimate MTTF of a metal line using the average current density, J, and the activation energy[1] of the failure, ΔH:

$$\text{MTTF} = \frac{A}{J^2}\exp(\Delta H/kT) \tag{11.3}$$

[1] Electromigration is modeled as a diffusion process that is thermally activated at a given energy level. The activation energy depends on the diffusion mechanism, either through the lattice ($\Delta H = 1.4$ eV) or across the grain boundaries ($\Delta H = 0.6$ eV).

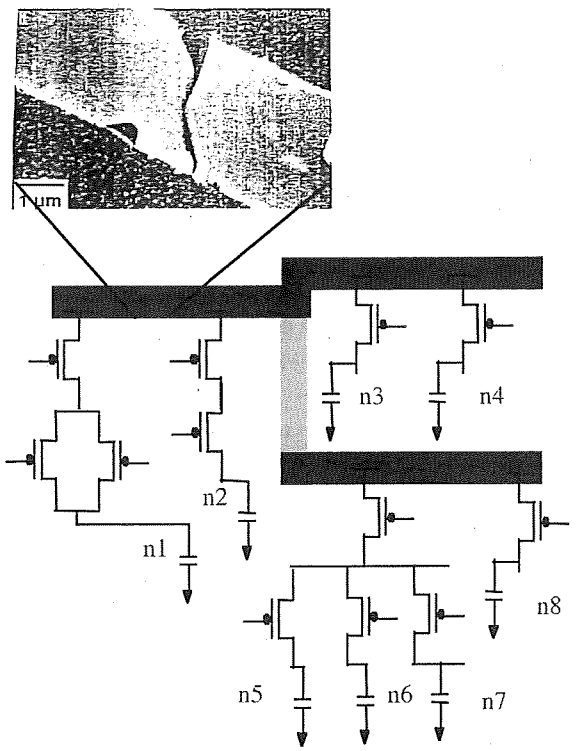


Figure 11.5
Electromigration failure in power bus.

Electromigration in the power grid is a function of the average current flow in metal lines and vias. Current flows primarily in one direction in each power bus, since it is delivering current from the power pad to the gate in the case of V_{DD} (see Figure 11.1). Therefore, it is called a dc phenomenon. To assess the EM tolerance of a piece of metal, we can examine its average dc current density, J, and compare it to some allowable level, J_{max}, when EM failure occurs. Since metal lines vary in height and material properties at different levels in the design, each metal layer may have a different failure criteria.

It is useful to characterize the lifetime of a chip due to long-term reliability problems through a metric called the *mean time to failure (MTTF)*. If the value of MTTF is 5–10 times greater than the expected lifetime of the product in which the chip is embedded, then we can safely ship the design to customers. However, if the chip lifetime is less than the product lifetime, improvements to the design must be made. Black's law is used to estimate MTTF of a metal line using the average current density, J, and the activation energy[1] of the failure, ΔH:

$$\text{MTTF} = \frac{A}{J^2}\exp(\Delta H/kT) \tag{11.3}$$

[1] Electromigration is modeled as a diffusion process that is thermally activated at a given energy level. The activation energy depends on the diffusion mechanism, either through the lattice ($\Delta H = 1.4$ eV) or across the grain boundaries ($\Delta H = 0.6$ eV).

Here, MTTF is the median time to failure in an ensemble of samples and A is an empirical fitting coefficient. The MTTF is inversely proportional to J^2 and is very sensitive to temperature. For a target MTTF, the value of J_{max} can be determined from the equation in units of amperes/cm^2. A typical value of J_{max} is approximately 10^6 A/cm^2, or 10 mA/μm^2. This value is compared with the actual current density in each segment to evaluate the potential for electromigration.

Example 11.1 **Use of MTTF to Establish Maximum Allowable Current Density**

Problem:

A design has a required lifetime of 10 years. Measurements for Al electromigration using an accelerated testing method indicates that A = 2×10^7 hr-cm^2/ampere and $\Delta H = 0.85$ *eV*. What is the maximum allowable current density in Al interconnect at 125 °C?

Solution:

The MTTF is the time at which 50% of the parts will fail, that is, t_{50}. For a desired lifetime of 10 years, the MTTF should be 5-10 times larger than this value. We will use a factor of 10 to be on the safe side:

$$t_{50} = \text{MTTF} = 10 \times 10 \text{ years} \times 365 \text{ days} \times 24 \text{ hours} = 8.8 \times 10^5 \text{ hours}$$

Using Equation (11.3), we can solve for J_{max}:

$$J_{max} = \sqrt{\frac{A}{t_{50}} \exp\left(\frac{\Delta H}{kT}\right)} = \sqrt{\frac{2(10^7)}{8.8(10^5)} \exp\left(\frac{0.85eV}{8.62 \times 10^{-5} eV/°K(398°K)}\right)}$$

$$= 8.95(10^5) A/cm^2$$

This value can be used as the maximum tolerable current density for electromigration.

The current density in each wire segment is computed using the wire dimensions

$$J_{avg} = \frac{I_{avg}}{W \times T} \tag{11.4}$$

where W is the wire width and T is the wire thickness. The criterion for electromigration failure is

$$J_{avg} > J_{max} \tag{11.5}$$

This comparison must be performed for every segment and via in the power grid to identify those nets that are prone to EM failure in the prescribed lifetime. Failing metal segments and vias must be modified, typically by adjusting the wire width according to Equation (11.4), until condition (11.5) is no longer violated. Not all

EM failures must be fixed since a break in the metal line does not necessarily mean that the part will fail. However, it may increase the likelihood of failure in other branches since there are usually many different current paths to each gate. The failures should be ordered in some manner, and then fixed in that order, until the probability of failure is below some predetermined threshold.

11.2.3 Power Routing Considerations

Integrated circuit power distribution systems are designed to provide the needed voltages and currents to the transistors that perform the logic functions of a chip. IC designers must design power systems with Equations (11.2) and (11.3) in mind for reliable operation. A complete picture of power grid integrity can only be obtained when effects such as *IR* drop, ground bounce, Ldi/dt, and electromigration are considered together. These are full-chip issues that must be addressed by interconnect verification tools that have the capacity and performance required to analyze detailed representations of the chip in a reasonable time. We will leave these higher-level issues to CAD tools and vendors. Instead we will focus on the methods needed to avoid these types of issues.

Initially, one can make estimates of the *IR* drop and Ldi/dt from *R*, *C*, and *L* calculations. Designers typically compute resistance along a conductor by counting the number of squares along the line and multiplying by the sheet resistivity. Similarly, the capacitance along the line is computed by multiplying the length of the line times the capacitance per unit length. Inductance can be estimated for the bonding wire or solder bumps, and crude estimates can be obtained for the power lines. Using this information and the expected current levels and rates of change, the anticipated voltage drops can be computed, along with electromigration lifetimes. Carrying out this type of analysis for a whole chip is difficult so rules of thumb have emerged to help guide the power grid design. These rules are associated with the metal width needed to keep the resistance low enough for the expected current levels and current densities.

A simplified method of power distribution is illustrated in Figure 11.6. The outer ring is V_{DD} and the inner ring is V_{SS}. The power and ground straps are routed

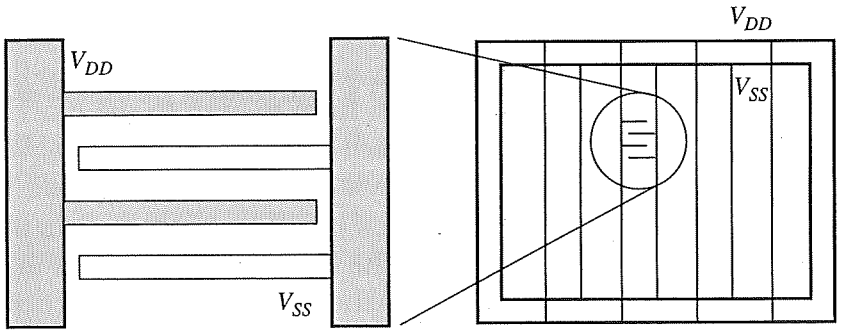


Figure 11.6
Large block level power distribution.

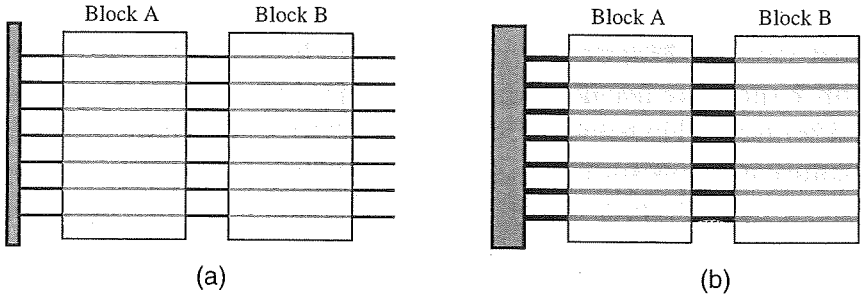


Figure 11.7
Power routing options.

vertically and interleaved so that at the lowest level they can be connected to the gates at the appropriate contact points. Of course, this type of routing does not fully account for the different types of blocks at the lower level, such as memory, logic, clock, and bus drivers, that may be connecting to the supply. However, the general idea of distributing power routes around the chip is conveyed in this diagram.

The power distribution system should be designed to route current to the gates, taking into account *IR* and EM. This requires a more careful consideration of the routing methodology. For example, consider the two blocks in Figure 11.7a. If power is routed through Block A to Block B, a larger *IR* drop will occur in Block B since power is also being consumed by Block A before it reaches Block B. As more and more blocks are added, the complex interactions between blocks determine the actual voltage drops. The placement of these blocks is typically based on the timing requirements of the system or the size and shape of blocks at the floor-planning stage. Therefore, sizing the power buses properly to minimize *IR* drop, while satisfying the required timing and area constraints, is a key design challenge.

The power busses could be widened to lower the resistance of the lines as in Figure 11.7b. The large metal trunks of power have to be sized to handle all the current for each block. Although routing power in this way makes it easier to manage *IR* drop, it also requires more area to implement. This forces designers to set aside large areas for power busing that takes away from the available global signal routing area. Since the same routing tracks are used for clocks, busses, and other global signals, the tradeoff between power and signal routing must be carefully managed.

Other alternatives are shown in Figure 11.8. In Figure 11.8a, current is delivered from both ends of the block thereby minimizing *IR* drop in the middle. Since the total *IR* drop is based on the resistance seen from the pin to the block, one could route around the block and feed power to each block separately as shown in Figure 11.8b. Ideally, the main trunks should be wide enough to handle all the current flowing through separate branches. However, the T-junctions have a high current density as current crowding occurs around the bends. It is important in this type of grid to examine the current density at all junctions, especially the corners, to ensure that EM problems do not exist.

Another approach to minimizing *IR* drop, depicted in Figure 11.9, is to have a grid of two metal layers. In the figure, assuming a five-layer metal process, Metal 4

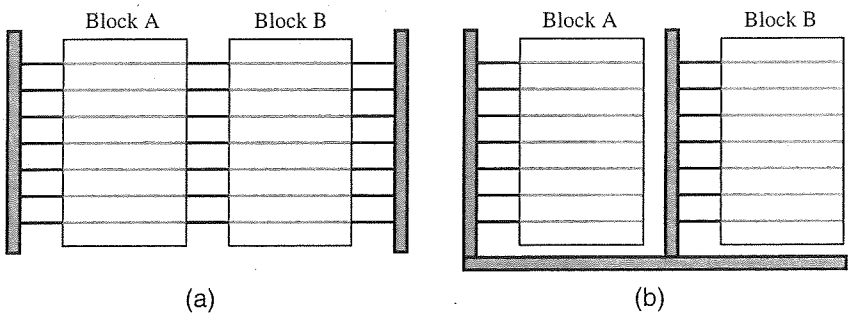


Figure 11.8
Simple techniques to reduce *IR* drop.

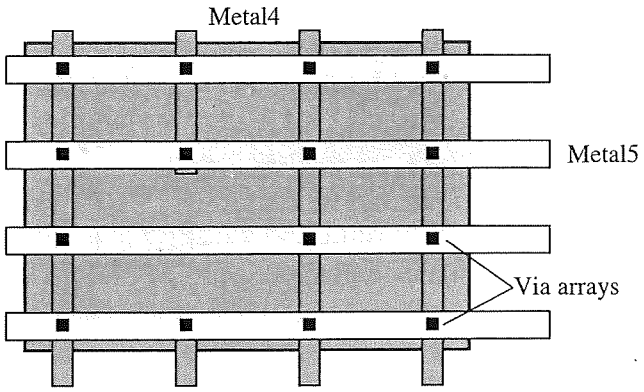


Figure 11.9
Vias in a mesh array methodology.

and Metal 5 use a via array to connect the two layers, effectively tying the whole grid to V_{DD}. However, part of the grid may have to be removed to route some signals. If you arbitrarily remove a strap that is conducting a large amount of current, the excess current must flow in adjacent straps that may push the current density in them beyond acceptable levels for electromigration. These examples all illustrate that design tradeoffs must be made with both EM and *IR* in mind.

11.2.4 Decoupling Capacitance Design

On-chip decoupling capacitances (decaps) are commonly used to keep the power supply within the noise budget, especially during peak demand periods in high-frequency switching applications. The decoupling capacitors are large-valued capacitances that hold a reservoir of charge located near the power pins and any large drivers, as shown in Figure 11.10. When large buffers switch from low to high, these decaps are the first line of defense for *IR* drop and Ldi/dt effects. The needed current for the switching process is obtained from the local decap. Later, current flows from the V_{DD} pad to refill the reservoir of charge for the next switching operation.

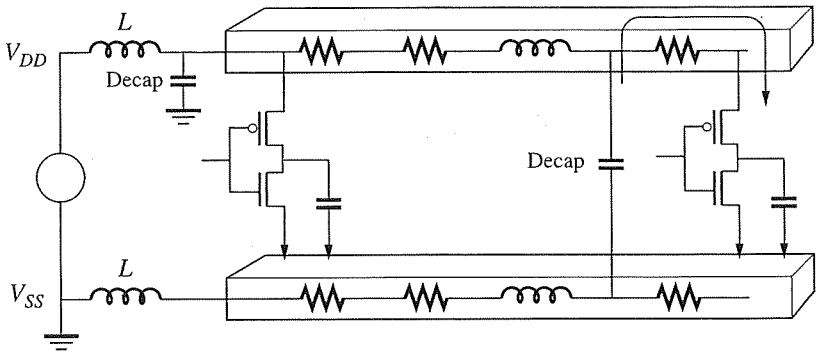


Figure 11.10
On-chip decoupling capacitance.

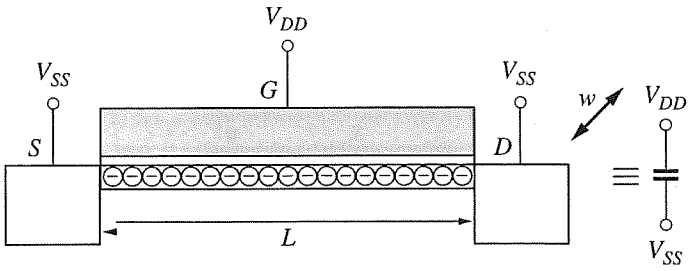


Figure 11.11
On-chip decap using MOS capacitance.

The on-chip decoupling capacitance is usually implemented using an NMOS transistor with the gate connected to V_{DD} and the source/drain connected to V_{SS}. The device is therefore in the linear region of operation. This is illustrated in Figure 11.11 where the parallel-plate capacitor is formed by the poly on one side and the channel inversion layer on the other. A first-order calculation of the capacitance is given by

$$C_{\text{decap}} = C_{ox}WL$$

and therefore the required decoupling value can be obtained with the proper choice of W and L.

A more accurate model would include the fringing and overlap capacitances of the poly edge to the source/drain regions:

$$C_{\text{decap}} = C_{ox}WL + C_{\text{ol}}W$$

The two main design issues are to decide how much decoupling capacitance to include and where to place them. Assuming that we somehow know the amount of decap needed, we must select the proper values of W and L to optimize the decap while achieving a target capacitance value. The selection of the proper value of W

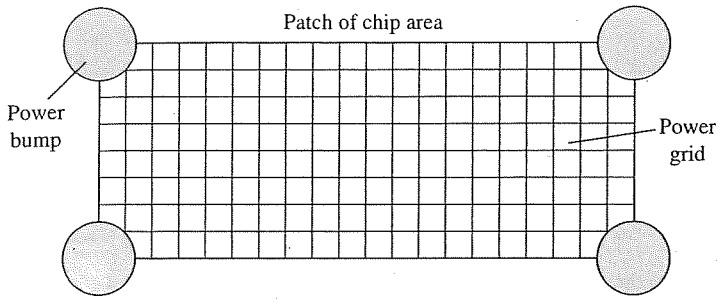


Figure 11.12
Simulation of patch area of chip.

and L can be based on a large number of simulations of inverter circuits with decaps as in Figure 11.10. The location of the decaps should be based on the location of the large buffers that are switching during the peak demand periods. Simulations can be carried out on a representative patch of the power grid between power bumps, shown in Figure 11.12. The power bumps are essentially the solder bumps connected to V_{DD}. Any circuitry connected to the grid should represent the types of buffers and drivers that will typically be found in the design.

There are many factors to consider when deciding how much capacitance to employ and where to place them. First, there is a certain amount of decoupling that is already present in the circuit due to the devices that do not switch. This includes gate and source/drain capacitances, as well as wire capacitances, for all nodes that are charged up to V_{DD}. This value may be subtracted off the target decoupling capacitance needed in the circuit. We also need to know the noise budget, the switching activity of the circuit, the current provided by the power grid and the rate of change of current with respect to time. Since the actual location and amount of decoupling capacitance are difficult to determine in an optimal way, some rules of thumb are used to ensure proper performance. Typically, the decap amount should be 10 times the switching capacitance, and decaps should be placed in as many open areas of the chip as possible. In addition, decaps should be located near the power pins to offset any effects of inductance due to solder bumps or bonding wires.

11.2.5 Power Distribution Design Example

To reinforce many of the concepts described in the previous sections, we sequence through a power distribution design example. Figure 11.13 shows an actual power distribution system for a large digital chip. Only the metal lines associated with the power grid are shown here. There are large blocks that are clearly distinguishable in this design. Each block was designed by a separate group. The horizontal and vertical patterns of the power grid are quite evident in each block. The memory block in the lower middle portion has a high-density grid as would be expected by the density of memory circuits. This chip example will be used several times in this chapter.

To illustrate a possible tradeoff between IR drop and electromigration, we first plot a contour map of the IR drop for the chip, as shown in Figure 11.14. The two

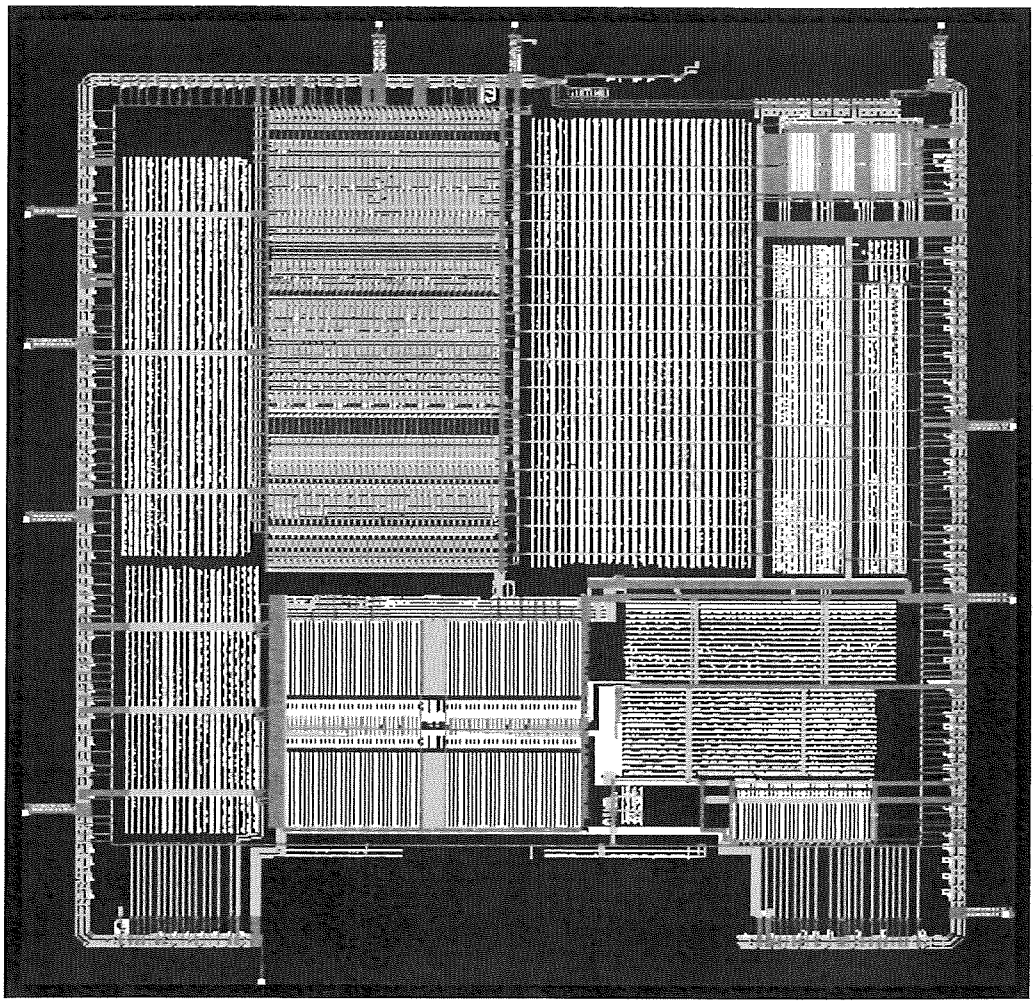


Figure 11.13
Example of a chip power distribution system.

plots show the voltage drop in the power grid with different shading levels to indicate the severity of the drop. This type of plot is generated by extracting all the resistors in the physical power grid and assembling a large resistive mesh; the logic circuits and memory blocks are then represented as ideal current sources and attached to the resistive grid. Each current source is assigned the average current level of the corresponding gate. A simulation is performed to compute the voltage drop at every point in the grid, and this voltage is assigned a grayscale value and plotted in the figure.

Consider the *IR* drop plot on the left side of Figure 11.14. The large oval region near the center of the chip is a region of excessive *IR* drop. In fact, it violates the 10% noise budget assigned to the power grid. The reason for the violation is that the lower part of the grid is not connected to the upper part of the grid. This was inadvertent since the chip was assembled simply by connecting the separately designed blocks together. Unfortunately, the upper portion of the grid must derive all needed current from the six or seven V_{DD} pads around the periphery of the top portion of the chip. To remedy this problem, the designer simply connected the lower portion

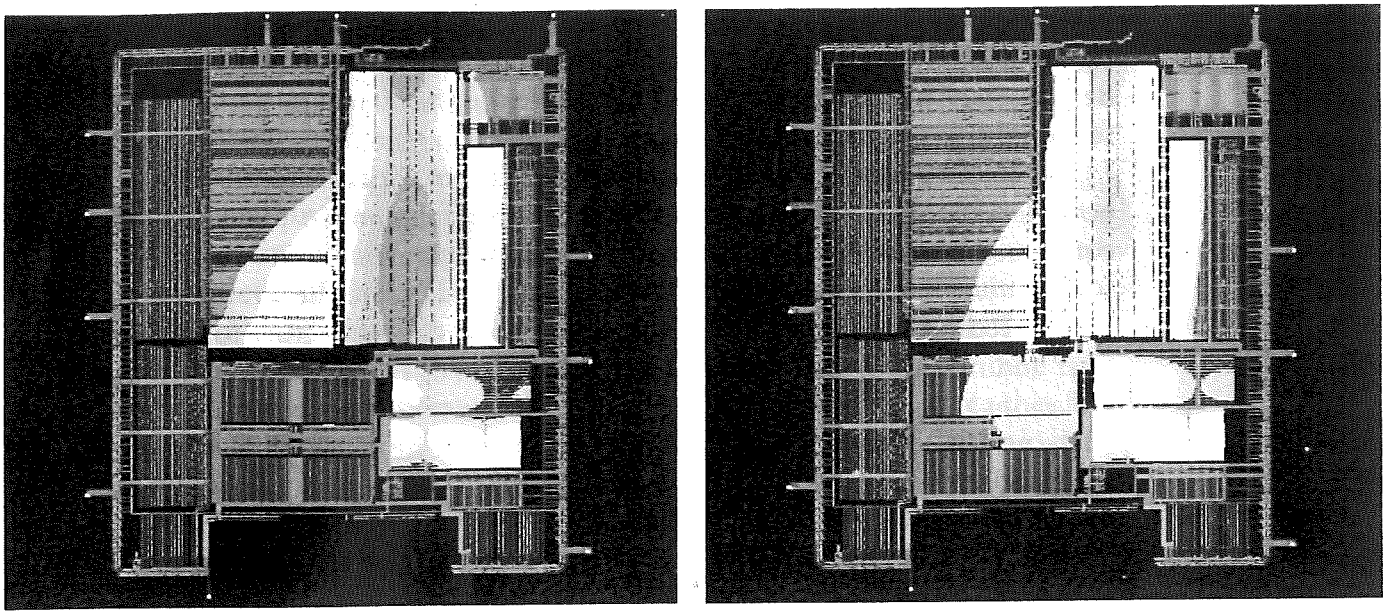


Figure 11.14
IR drop plots before and after repairs.

of the chip to the upper portion with a number of power straps (i.e., metal connections). The resulting *IR* drop plot is shown in the right-hand side figure. The power grid now satisfies the noise budget.

The side effect of the changes made to the grid is illustrated in Figure 11.15. Here, the electromigration failures have been plotted as dots on the power grid.

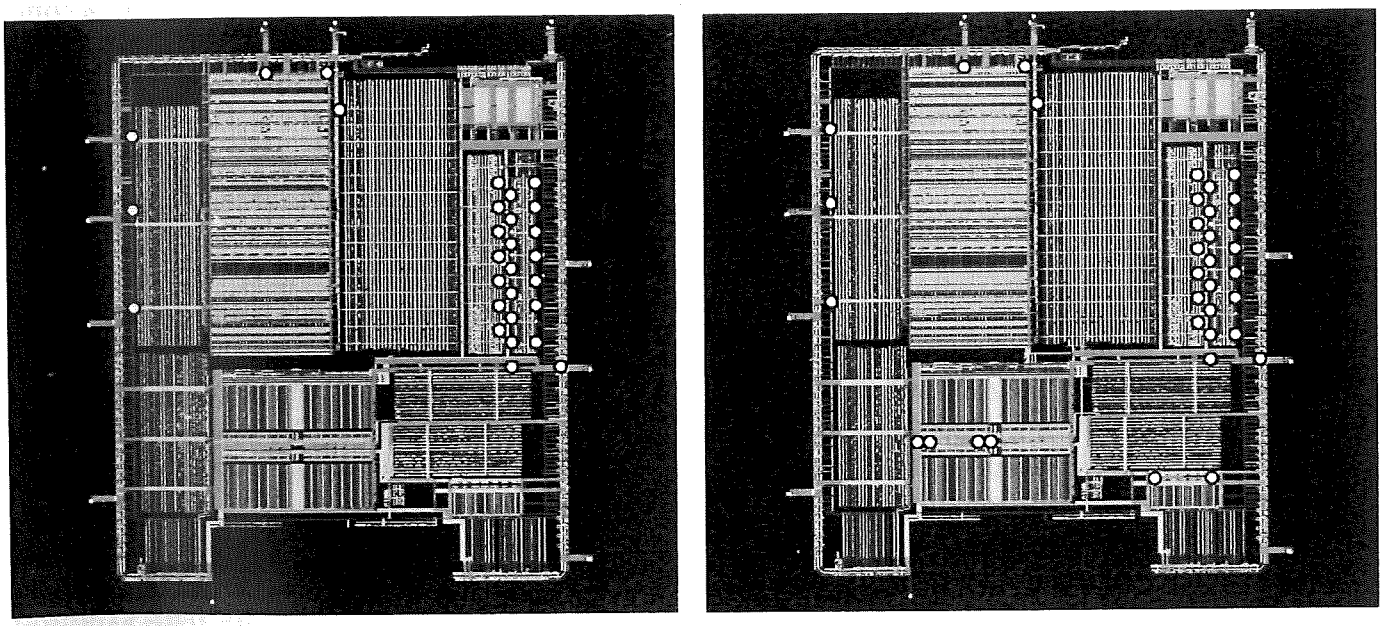


Figure 11.15
Electromigration plots before and after repairs.

Each failure is based on whether or not a given section of metal satisfies the EM criterion based on Equation (11.5). The equation requires a calculation of the current density, J_{avg}. This value is simply the dc current in the metal line divided by its cross-sectional area. Once computed, if the J_{avg} of the segment is less than a specified level, J_{max}, then the metal segment is flagged as a violation. Since large currents flow at the boundaries of the chip, we would expect more violations in these regions. This is confirmed by the locations of the dots in the two figures.

The left-hand side figure shows the violations before the straps were added to fix the *IR* drop problem. The right-hand side figure is the picture for EM violations after adding straps. Notice that the lower portion of the grid experiences more EM violations than in the other diagram. This is due to the fact that the lower portion of the grid is now supplying more of the current to the upper portion and the metal segments have a higher current density. Some of the densities are high enough to violate the EM criteria. Clearly, the power system must be redesigned by considering both *IR* and EM together.

Electromigration failures can be reduced in several ways. The basic idea in all approaches is to reduce the average current density experienced by any metal segment. The simplest approach is to widen the metal lines. However, increasing the width beyond a certain point leads to overdesign, which costs area and can reduce yields. Another approach is to change the current flow in the power grid itself by adding jumpers and straps between different points in the grid. This would reroute current from the affected areas, but such changes would require another verification pass to confirm that the problem has not simply been moved to another part of the design. Switching to copper is perhaps the most significant step toward reduction or elimination of these types of failures in metal segments. However, electromigration in vias and the local heating of neighboring lines may be limiting factors in the ultimate improvement that can be achieved using Cu.

Reducing the impact of *IR* drop in a power distribution system can be accomplished in several ways. The simplest approach is to widen the lines that experience the largest voltage drops since increasing the width decreases the resistance (and the *IR* drop). However, this may not always be possible due to constraints in the routing area. Another approach is to add/remove metal straps as mentioned above. Since *IR* drop is due primarily to simultaneous switching events, another approach is to stagger the gates that are switching together such that they switch at slightly different times—at least enough to keep the problem within the noise budget. Alternatively, buffer sizes can be reduced, but this may not be possible if the design fails to meet performance requirements with smaller devices.

The most effective approach is to use *decoupling capacitors* between power and ground, which can deliver the additional current needed by the power distribution system. As described earlier, a decoupling capacitor is a large capacitance that is connected from V_{DD} to Gnd typically using a large MOS transistor. The charge stored in the decoupling capacitors is used to initially provide current to the gates, thereby reducing the *IR* drop and *Ldi/dt* effects. These decoupling caps are usually scattered throughout the design in any available space, using transistors with their gates tied to V_{DD} and their source-drain regions tied to Gnd. All empty regions of the chip are

filled with decoupling caps using the philosophy that you can never have enough (i.e., overdesign). However, the large area overhead may act to reduce yield. Typically designers add a total decoupling capacitance that is 10 times the amount of capacitance switched on every cycle, as a rule of thumb. Package and pin *Ldi/dt* effects can be mitigated by placing large capacitances near the pins.

The number of pins assigned to V_{DD} and Gnd can also be increased to reduce *IR* drop. By providing more supply pins, the current requirements for a given section can be satisfied from a number of sources. Of course, this limits the number of pins available for I/O. A more aggressive solution is to use a ball-grid array where the power supply connections can be placed at various points within the chip. The key design issue is proper placement of the bumps around the chip. Note that solder bumps cannot be used in sensitive areas such as memories and dynamic logic because the bumps generate α-particles that may cause logic value upsets on the sensitive nodes. Nevertheless, when used appropriately, solder bump technology can reduce *IR* drop significantly.

11.3 Clocking and Timing Issues

11.3.1 Clock Definitions and Metrics

In modern VLSI systems, the clock is perhaps the most important signal because it controls the rate of data processing and communication. It provides a structured framework for dealing with high-complexity digital systems. A clock network distributes the clock signal from the clock generator, or source, to the clock inputs of the synchronizing components. We typically build the FSMs (finite state machines) from combinational logic (for next state logic) and flip-flops (for storage elements). The logic elements have differing delays and, as a result, the path delays through the logic blocks will differ. We control the storage elements with a clock to keep everything synchronized. Unfortunately, we can have fast or slow signals in the combinational logic, and fast or slow clocks that make timing synchronization more difficult.

There are a number of important definitions and figures of merit associated with the clock as shown in Figure 11.16. Ideally, the clock should arrive at all flip-flops at the same time, have a fixed period, T_{cycle}, and near zero rise/fall times. Actual clocks fall short of this ideal model. First, the arrival times of the clock to the flip-flop (FF)

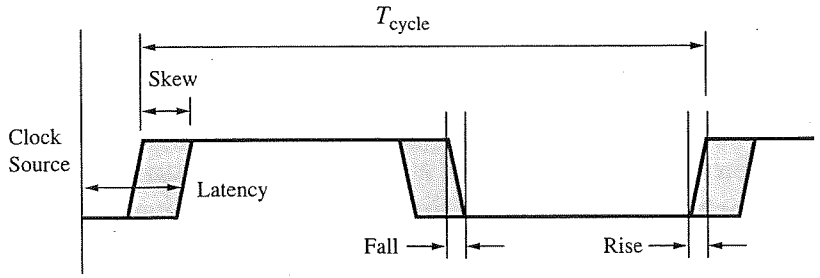


Figure 11.16
Clock latency and skew definitions.

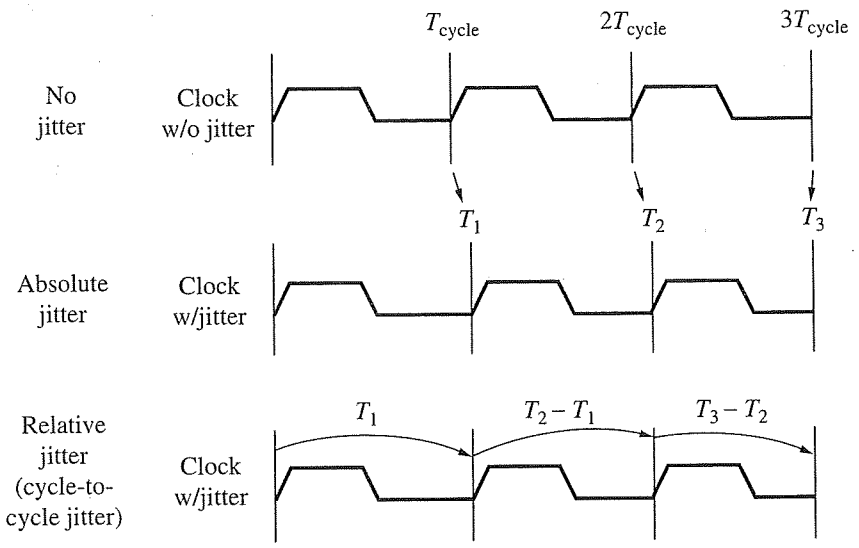


Figure 11.17
Definitions for clock jitter.

inputs around the chip are different. The *clock skew* is defined as the maximum difference of the clock arrival times to the inputs of FFs. Another important parameter is clock *latency*, as shown in Figure 11.16. The latency is defined as the maximum delay from the clock source to any FF clock input. This is important since it determines whether the external system clock is properly synchronized with the internal clock. Ideally, we would like the internal and external clock edges to be positioned at the same point in time. Otherwise, the chip I/O will require large setup and hold times to account for latency. The clock *rise/fall times* should be small and kept about equal, but this requires the use of large buffers. Unfortunately, clock networks with large buffers can consume a large portion of the total power of synchronous VLSI systems (up to 40%). The clock design objectives must be attained while minimizing the use of system resources such as power and area.

Another important metric for the quality of a clock is the clock *jitter*. This is the variation of the clock period from one cycle to another, as seen in Figure 11.17. Normally, we expect the clock edge, either rising or falling, to be at the same point in time, spaced by a period T_{cycle}. However, a variety of factors move the clock edge around from cycle to cycle. This may be viewed as an error or uncertainty in the clock edge or an uncertainty in the clock period. If we superimposed the clock waveform over each period, we would notice it drifting back and forth over time. This is the effect of jitter. The difference between skew and jitter is that skew is the difference between the same clock edge at two different locations on the chip; jitter is the difference between the expected time of the clock edge and the actual time at the same point in the chip.

There are a couple of ways to measure jitter as shown in Figure 11.17. One approach is to use a clock signal without jitter as a reference, such as the first clock that is evenly spaced by a given period T_{cycle}. The second clock has jitter and its clock

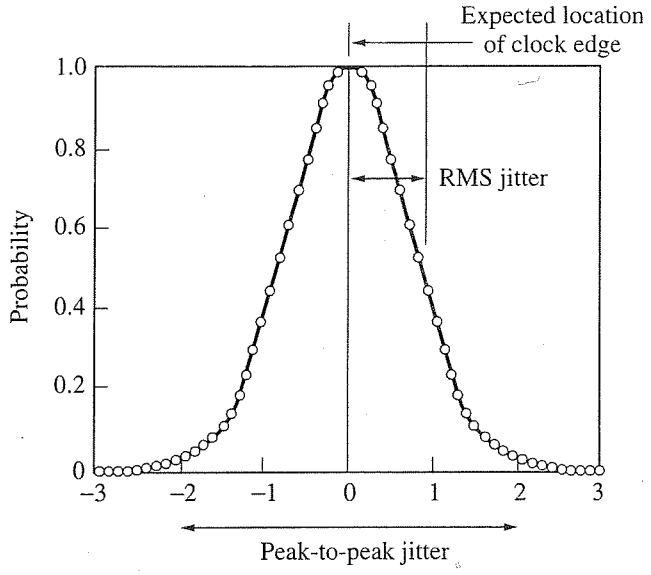


Figure 11.18
Jitter measurements.

edges are measured against the absolute points in time where the jitter-free clock edges occur. That is, T_1 is compared to T_{cycle}, and T_2 is compared to $2T_{cycle}$, and so on. This is an *absolute jitter* measurement: $T_n - nT_{cycle}$.

Another approach is to ignore errors in the edge locations up to the last cycle and measure jitter relative to the last clock edge. This is indicated using the third clock waveform which has the same error as the second clock. In this case, we simply compare the current period of the clock with the expected period. This is *cycle-to-cycle jitter*: $(T_n - T_{n-1}) - T_{cycle}$. For on-chip applications, we are mainly interested in cycle-to-cycle jitter since we care only if the logic functions can be completed within the current clock cycle and not on any absolute time reference. Absolute jitter is more important for off-chip timing synchronization between two chips.

Cycle-to-cycle jitter is typically smaller than absolute jitter since it is always measured relative to the last clock edge. If we measured the jitter over thousands of cycles, we would produce a histogram with a distribution of the form shown in Figure 11.18. Here, the normalized probability of the clock edge is plotted against its arrival time. The expected arrival time is at the point marked "0" on the x-axis. Some edges will arrive earlier than this time, while others will arrive later. The x-axis is labeled in terms of the standard deviation, or σ, of the distribution. Jitter measurements are usually provided as peak-to-peak values (4σ) or RMS values (one σ), as shown. The goal in clock design is to minimize jitter, especially within the clock generation circuits.

11.3.2 Clock Skew

Much of the effort in clock design centers on minimizing the clock skew. An example of the clock skew problem is shown in Figure 11.19. The clock driver is located

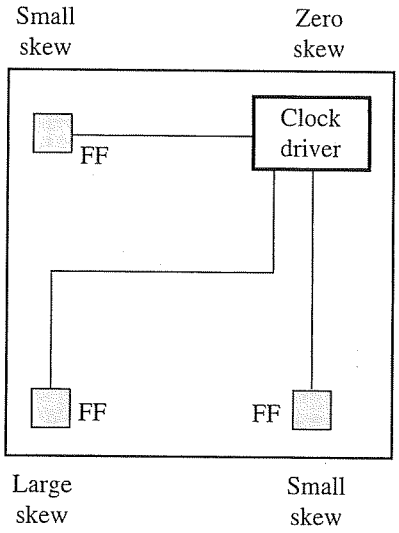


Figure 11.19
Clock skew example.

in one corner of the chip while the target flops are scattered around the chip. The arrival times of the clock to each of the FFs will be slightly different; some will receive the clock early while others receive the clock late, relative to each other. The delay differences are due to differences in the length of the global wire from clock to FF, gate delays along the different paths, and the fanout driven by the signal. If we examine the FFs in each corner, the two near-end FFs will have a smaller skew relative to each other (early clock), whereas the one that is diagonally opposite to the clock will have a large skew (late clock).

At first, this skew appears to be more an annoyance than a problem. In reality, it can greatly affect the performance and proper functionality of the overall system. To illustrate this, consider the situation shown in Figure 11.20. In the upper part of the figure, a sequential system is shown with positive-edge triggered flip-flops at each end and a combinational "cloud" representing the logic gates between the flops. There are multiple paths through the logic that can be characterized by two parameters: the worst-case path with delay T_{dMAX}, and best-case path with delay, T_{dMIN}.

Now, consider the effect of skew on cycle time. In Figure 11.20, assume that the clock arrives early at the far-end flop and late at the near-end flop. When the clock arrives at the near-end flop, it launches new data into the combinational cloud (i.e., into the logic block) after a delay of $T_{clk\text{-}q}$ which is a property of the flop, as discussed in Chapter 5. This is followed by a maximum delay of T_{dMAX} through the logic. Upon arriving at the far-end flop, a setup time for the flop, T_{setup}, must be satisfied. The three components, $T_{clk\text{-}q}+T_{dMAX}+T_{setup}$, would normally comprise the cycle time, T_{cycle}.

However, if the clock arrives early at the far-end flop due to skew, we may have a problem as shown in the waveforms of Figure 11.20. The early clock of the next cycle would arrive and latch data that are not yet stable from the current cycle, according to the setup time. This would cause a functional error in the circuit.

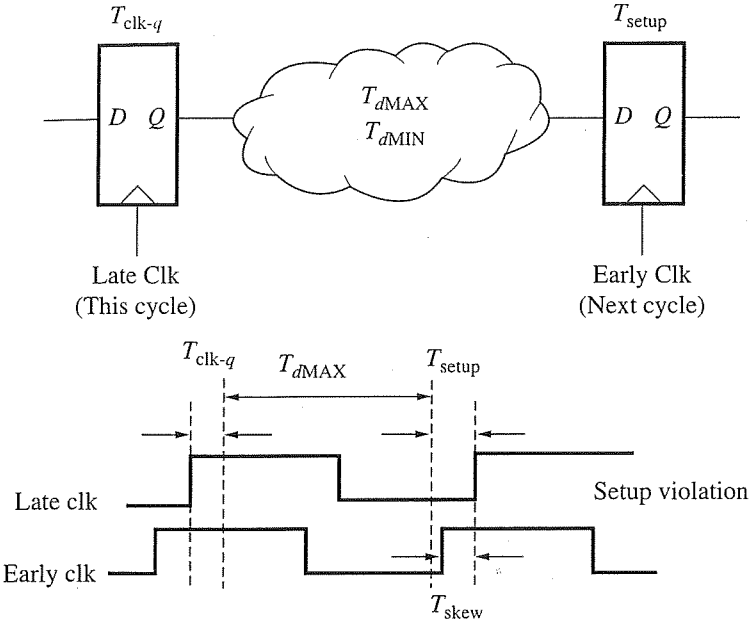


Figure 11.20
Effect of skew on cycle time.

When circuits do not meet the requirements of the cycle time, it is usually referred to as a *setup time violation*, even though it is due to skew. There is a simple solution to this problem. If we increase the clock period by T_{skew} we can guarantee that the circuit always works. However, the penalty is that the cycle time is longer by the skew value. The new cycle time is given by

$$T_{cycle} = T_{clk-q} + T_{dMAX} + T_{setup} + T_{skew} \tag{11.6}$$

Next, consider Figure 11.21. Here the arrival times of the clock at the flip-flops are reversed. This time we are concerned about the shortest path through the logic block. When the early clock arrives at the first flop, the logic propagates through the combinational cloud and the fastest signal path reaches the second flop in a time $T_{clk\text{-}q} + T_{dMIN}$. If this is too fast, there is a problem. When the late clock arrives at the second flop in the same cycle, it latches the incoming data regardless of whether they are the correct data or not. It may inadvertently capture the new (incorrect) data. The skew is working against the goal of latching the right data. Why? Because if the new data arrives before the hold time of the previous data is satisfied, the second FF incorrectly samples the new data instead of the correct data. This is referred to as a *hold time violation* as shown in Figure 11.21. Note that this problem cannot be solved by slowing down the clock. The criterion for proper operation is that

$$T_{skew} + T_{hold} < T_{clk-q} + T_{dMIN} \tag{11.7}$$

To overcome hold time problems, the designer must increase the delay of the shortest paths that violate Equation (11.7). This can be accomplished by adding buffers

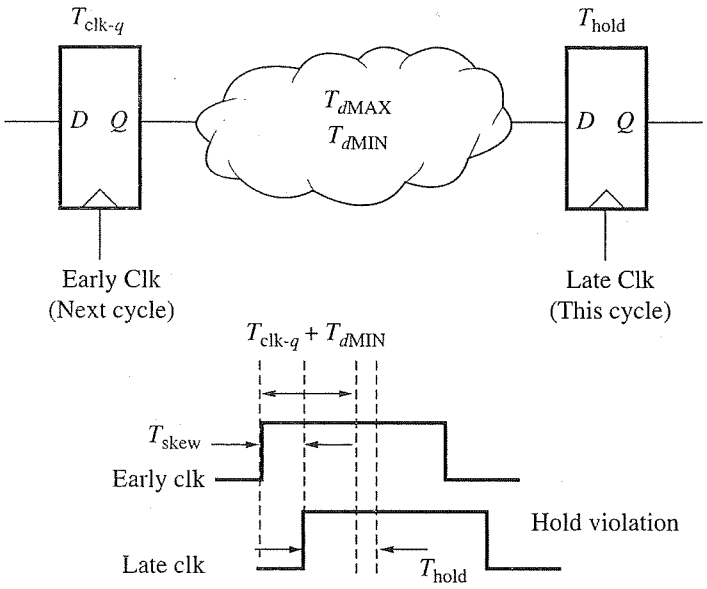


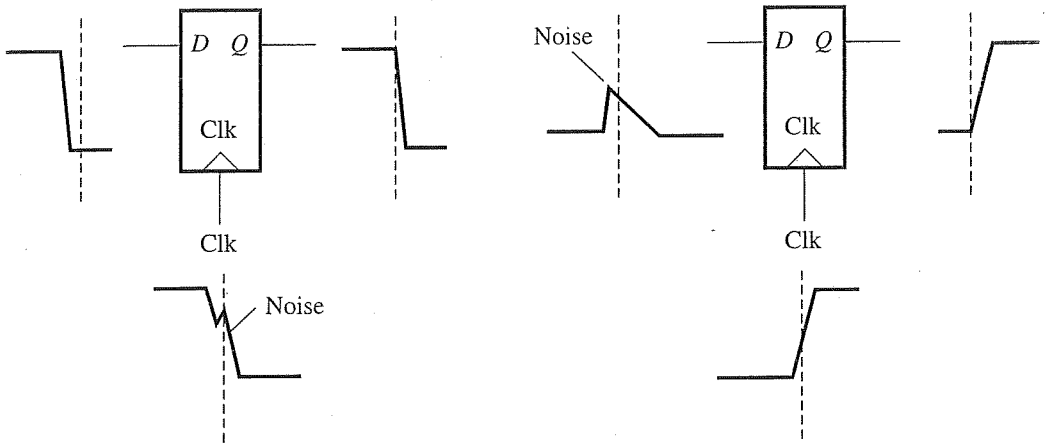
Figure 11.21
Effect of clock skew on functionality.

or reducing the size of some of the existing gates. Care must be taken so as not to inadvertently increase the delay of the longest paths. Ultimately, the clock circuit must be designed with minimum skew to reduce the number of potential setup and hold problems.

11.3.3 Effect of Noise on Clocks and FFs

At this stage, it is useful to point out some of the serious consequences of noise due to crosstalk, as described in Chapter 10, especially relating to clocks and flip-flops. Consider the positive edge-triggered D-flop of Figure 11.22a. Normally, when the clock goes from high to low, the D-flop continues to hold the output signal at its previous value. However, if a capacitive or inductive noise event couples to the clock node and is fed to the FF, it may inadvertently capture a random value at the D input. Here, the clock has a short positive-going edge during a fall transition. It is positioned at the switching threshold of the flop and triggers a capture event at the input. It is critical that noise events due to interconnect be eliminated when designing the clock distribution network.

A second case is shown in Figure 11.22b where the clock is free of any glitches, but the data input experiences a coupling noise event. Again, it occurs exactly when it cannot be tolerated (according to *Murphy's Law*)—on the positive-edge of the clock—and therefore the wrong value is captured. The noise does not satisfy the setup and hold requirements of the FF, but it still produces an unwanted change at the output. Therefore, a significant amount of noise cannot be tolerated at the data input to the FF.



(a) Glitch on Clk Input (b) Glitch on Data Input

Figure 11.22
Effect of coupling noise on FF performance.

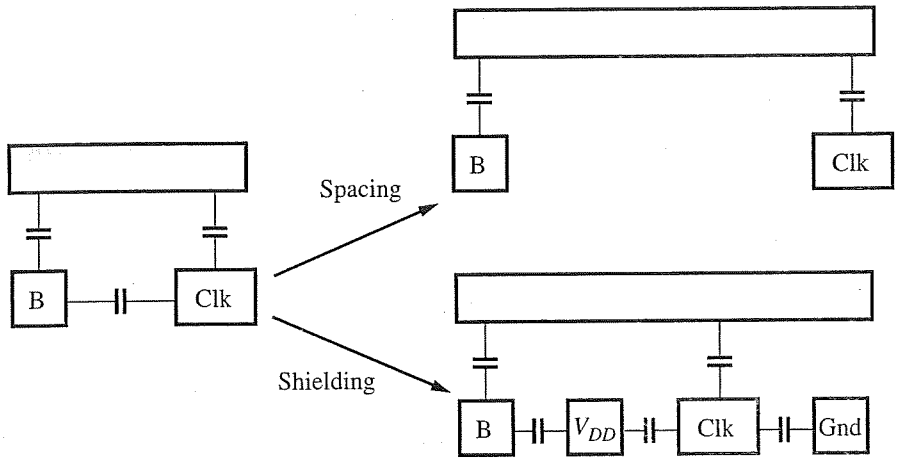


Figure 11.23
Layout solutions to coupling problems.

To limit noise events around FFs, one can employ the spacing and shielding techniques described earlier. For example, in Figure 11.23, signal B and the Clk signal are situated very close to one another. There is a potential for noise events between these signals due to coupling capacitance. The possible layout solutions are to separate B and Clk by a sufficient spacing or to shield the Clk with V_{DD} and Gnd. The spacing approach has a lower capacitance and therefore reduces power to some degree. However, the shielding is beneficial for both capacitive coupling and inductive coupling. Typically, clocks are shielded to protect them from unwanted noise, but also to protect other sensitive signals from unexpected mutual coupling to the clock. The price of shielding is increased capacitance and power dissipation.

11.3.4 Power Dissipation in Clocks

The total power dissipation in a clock circuit can be significant. As mentioned earlier, it may constitute 30–40% of the total power in high-performance designs. The most dominant component of the power is due to dynamic switching:

$$P = CV_{DD}^2 f_{\text{clk}}$$

Note that the α term is not present in this expression since the clock switches on every cycle. Effectively, $\alpha = 1$. Since the frequency f_{clk} and the supply voltage V_{DD} are known, the only term left to compute is the capacitance value, C. The capacitance due to the clock is very high in large digital circuits, possibly in the nano-Farad range.

As an example, consider a 0.18 μm technology with $V_{DD} = 1.8$ V. If a circuit operates at 500 MHz and has a clock capacitance of 1 nF, the total power of the clock circuit alone is

$$P = (1 \text{ nF})(1.8)^2(500)(10^6) = 1.6 \text{ W}$$

The capacitance of the clock arises from many sources. First, the interconnect capacitance of the metal lines is a major source of capacitance. Second, there are large buffers used in the clock distribution network that give rise to large fanout and self-capacitance terms. Third, there are capacitances associated with the inputs of the flip-flops driven by the clock. Finally, there is another source of power dissipation whenever the FFs are triggered. That is, the flops themselves consume power each time the clock signal switches. When designing the clock distribution network, it is important to minimize all of these capacitances since they contribute directly to the power.

One popular technique is *clock gating*. It is a relatively simple concept, as shown in Figure 11.24. If a functional unit is not required for an extended period of time, the clock feeding the module is turned off by means of an Enable signal that is ANDed with the clock. Most functional units are inactive a majority of the time. For example, a floating-point unit may be inactive while a number of integer operations are being performed. If the clock continues to toggle the D-flops associated with

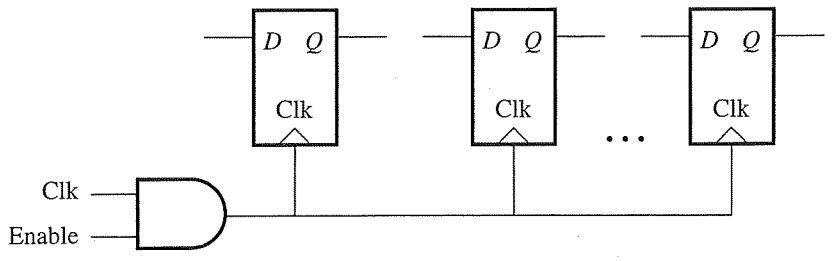


Figure 11.24
Gated clock to reduce power.

this unit, power is being dissipated unnecessarily. Clock gating can disable the unit to save the power that would be consumed by the flops.

There are a number of issues associated with clock gating. First, it is important to decide whether gating is appropriate for a given block. Clearly, a gated clock at the input of all flops does not save much power. Therefore, the location of such gates around the chip should be determined carefully. Second, the gate may add skew to the clock, so it must be inserted in a balanced way. That is, even if certain blocks do not require it, a gate should be added to maintain balance. But this runs counter to the first issue. Third, a certain amount of logic circuitry is needed for the Enable signal that may affect performance or consume power. Depending on the complexity of the clock gating, this tradeoff should be monitored to ensure that a gain here is not offset by a loss elsewhere. And finally, timing verification complexity and reusability of a block are additional considerations for this approach. Nevertheless, this technique is very popular as it saves unnecessary clock activities inside the gated module.

11.3.5 Clock Generation

With the basic definitions and design considerations completed, we now turn our attention to the design of the clock circuit itself. A simple technique for on-chip generation of a primary clock signal is to use a ring oscillator as shown in Figure 11.25. A ring oscillator has an odd number of inverter stages and produces an oscillating signal at each node. The period of oscillation is determined by the delay of each stage and the number of stages. This type of clock circuit has been used in low-end microprocessor chips. However, the generated clock signal can be quite process-dependent and unstable. Consequently, separate clock chips that use crystal oscillators have been used for high-performance VLSI circuits.

Usually, a VLSI circuit receives one or more primary clock signals from the external clock source and then generates necessary derivates for its internal use. Typically, we must generate both true and complementary clock signals for internal use. Figure 11.26 shows a CMOS generator/driver that uses a transmission gate (TG) delay element. The first driver of the chain is an inverter with transistors Mp1

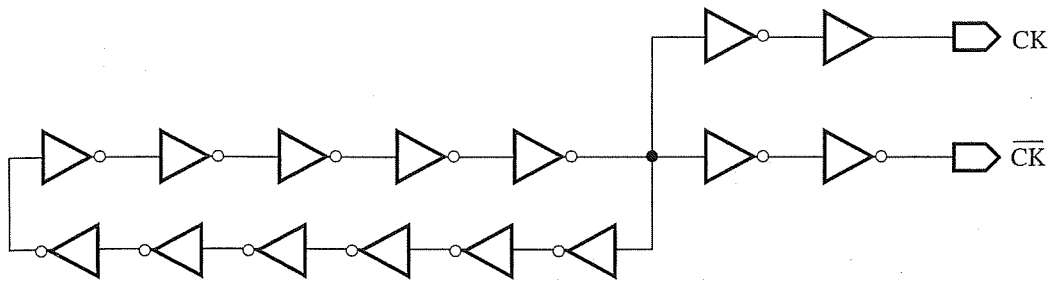


Figure 11.25
A simple on-chip clock generator using a ring oscillator.

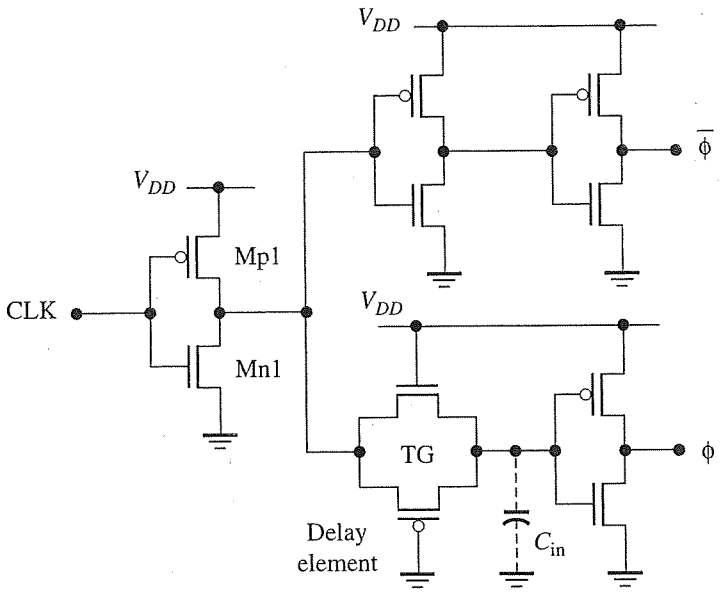


Figure 11.26
A clock generator with a transmission gate delay.

and Mn1. The top branch of the chain consists of two cascaded inverters and generates a signal $\overline{\phi} \equiv \overline{\text{CLK}}$, whereas the bottom branch consists only of a single inverter and a transmission gate and provides $\phi \equiv$ CLK. The TG is used as a delay element to minimize the clock skew between the two generated signals.

Another simple approach uses an SR latch to generate two-phase clock signals (see Figure 11.27). This circuit has two weaknesses that should be checked and properly designed. First, the input CLK signal propagates through an additional inverter to generate $\overline{\phi}$. The second problem is related to the first one. The NOR gates should not be designed to be equal in size, but rather their output signals should have an identical response. The design should also take into account the total capacitance of the interconnect that is dependent on the layout topology.

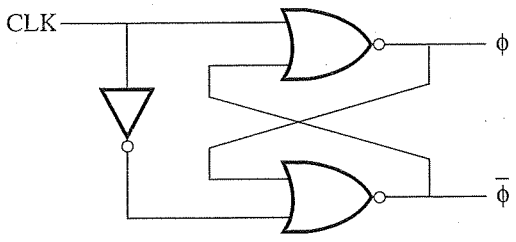


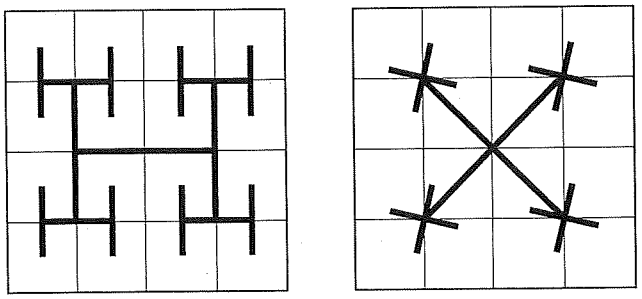
Figure 11.27
Latch-based clock generator.

11.3.6 Clock Distribution for High-Performance Designs

Once the clock signal is generated, it is necessary to distribute it around the chip. This is a complicated task due to the issues of skew, noise, and power. Since clock signals are required almost uniformly over the chip area, it is desirable that all clock signals be distributed with a uniform delay. Ideal distribution networks are the H-tree and X-tree structures shown in Figure 11.28. In an H-tree structure, the distances from the center to all branch points are the same and, hence, the signal delays are expected to be the same. The limitation of this structure is a difficult implementation due to routing constraints and nonuniform fanout requirements. Another configuration that yields equal-length interconnections is the X-clock tree. Compared to an H-tree, it suffers from the following:

- The clock line produces sharper corners than 90°; inductive discontinuities are significant and, as a result, reflections are larger.
- The fanout at the branching points is always 4 as opposed to a fanout of 2 in the H-tree. An increased fanout degrades matching the line impedance and increases the reflections.
- Two clock lines are in a close proximity (farther apart in an H-tree) which increases the crosstalk.

To avoid reflections at the branching points, the line separates into two branches with characteristic impedances twice the impedance of the incoming line. In parallel, they act like a single line and have the same impedance as the incoming line. In order to obtain such a perfect impedance matching, wiring-oriented approaches have to be applied. The designer has to narrow the line width at the branching points because the line impedance is inversely proportional to its capacitance. This type of layout construction is shown in Figure 11.29a. Other alternative approaches are a wide-bus technique and a fat clock bus. The wide-bus technique increases the capacitance and decreases the resistance on source-to-sink paths (Figure 11.29b).



H-tree (a) X-tree (b)

Figure 11.28
Symmetric clock trees.

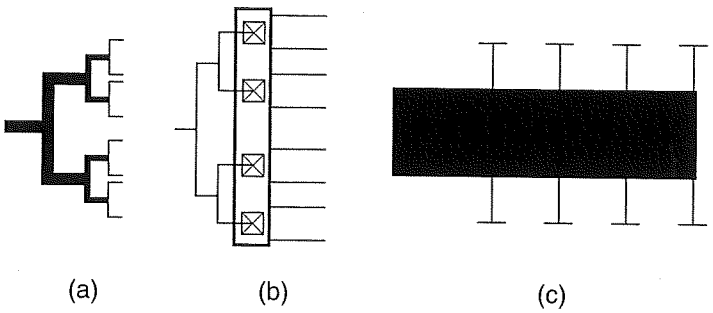


Figure 11.29
(a) Clock tree with tapered wire widths. (b) Clock tree using a wide-bus. (c) Fat bus clock tree implementation.

The construction of a fat clock bus (Figure 11.29c) requires a very large clock driver, thus the driver and the interconnect will consume large amounts of power. However, since delays of the clock tree are dominated by the capacitance of the wide-bus, the bus-to-clock interconnection can be done simply and efficiently.

Because of the significant delays associated with long wires in clocks, the designer must insert buffers at various points along the clock wire. In essence, buffer-oriented solutions partition the large clock network into small segments. The main challenge of any buffered clock architecture is to maintain balance in the delays. Several examples of buffered clock distributions are shown in Figure 11.30 and Figure 11.31. Figure 11.30 shows three conventional clock-tree networks. The

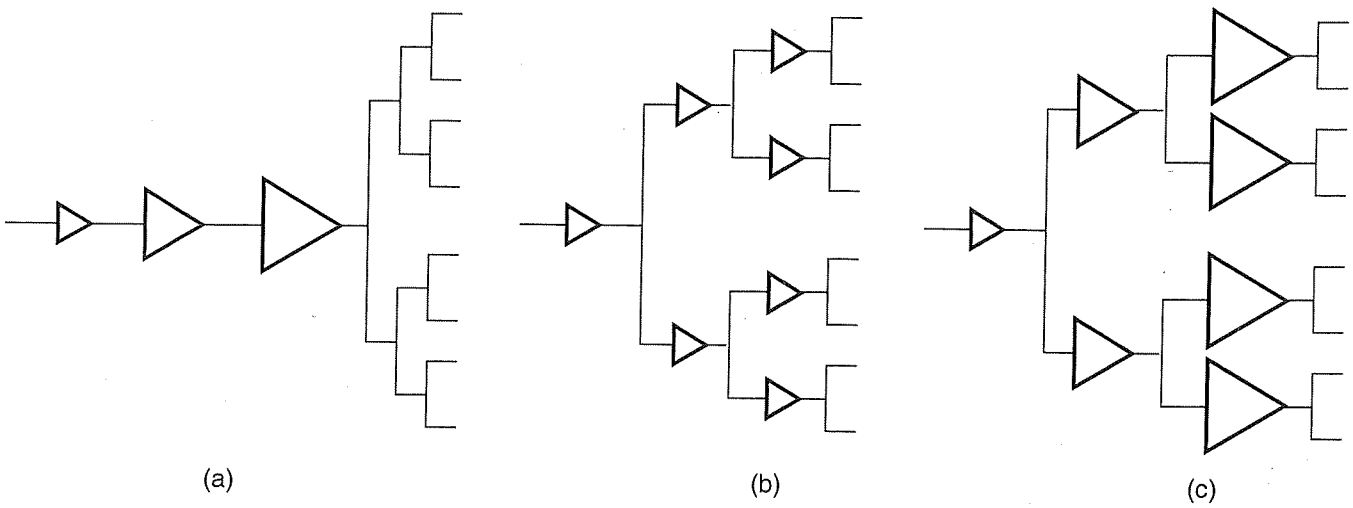


Figure 11.30
(a) Buffer chain with tapered sizes driving clock tree. (b) Clock power-up tree with uniform buffer size. (c) Clock power-up tree with tapered buffer sizes.

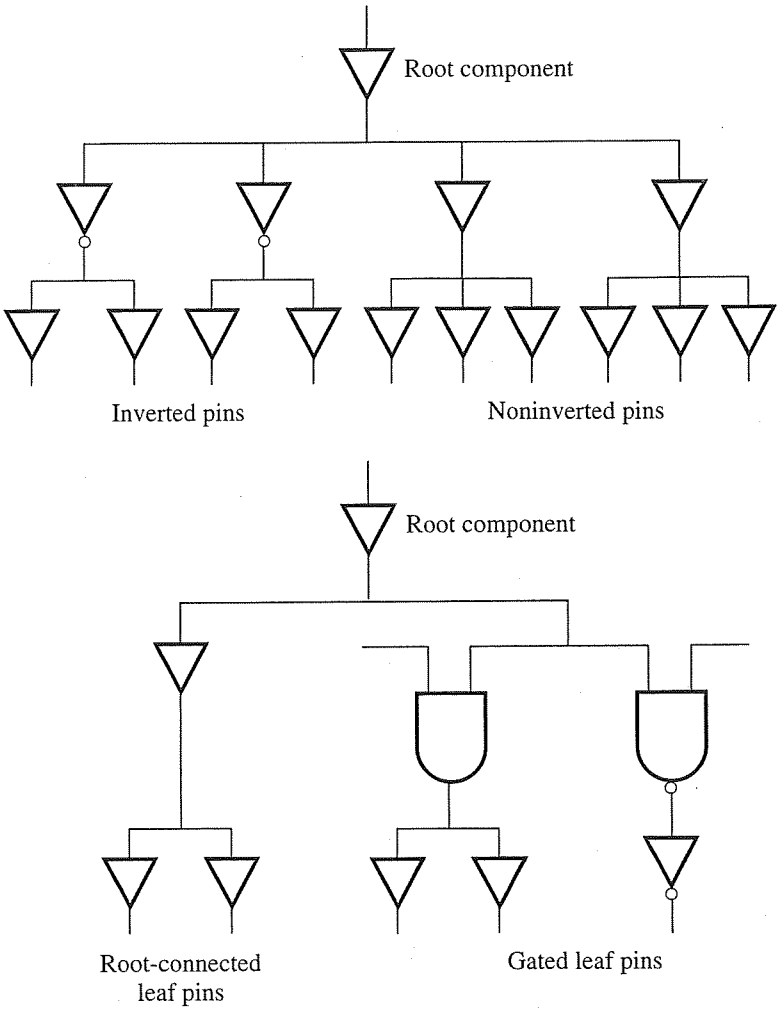


Figure 11.31
Clock tree examples.

first one uses a super buffer technique to drive the clock tree. The second approach inserts equal-sized buffers in the clock tree. The third approach increases the buffer sizes as they get closer to the loads.

Figure 11.31 shows two clock networks with additional characteristics. In the top picture, the tree generates an inverted clock in addition to a standard clock signal, and in the bottom picture, the tree generates a gated clock signal and a standard clock signal. Gated clocks are useful when we want to prevent the clock from triggering a flip-flop to reduce power.

11.3.7 Example of a Clock Distribution Network

Figure 11.32 illustrates a clock network extracted from an industrial chip. It is the same example used in Section 11.2.5. The interconnect associated with the entire clock is shown in the figure. The clock tree starts at a root node and splits off into a

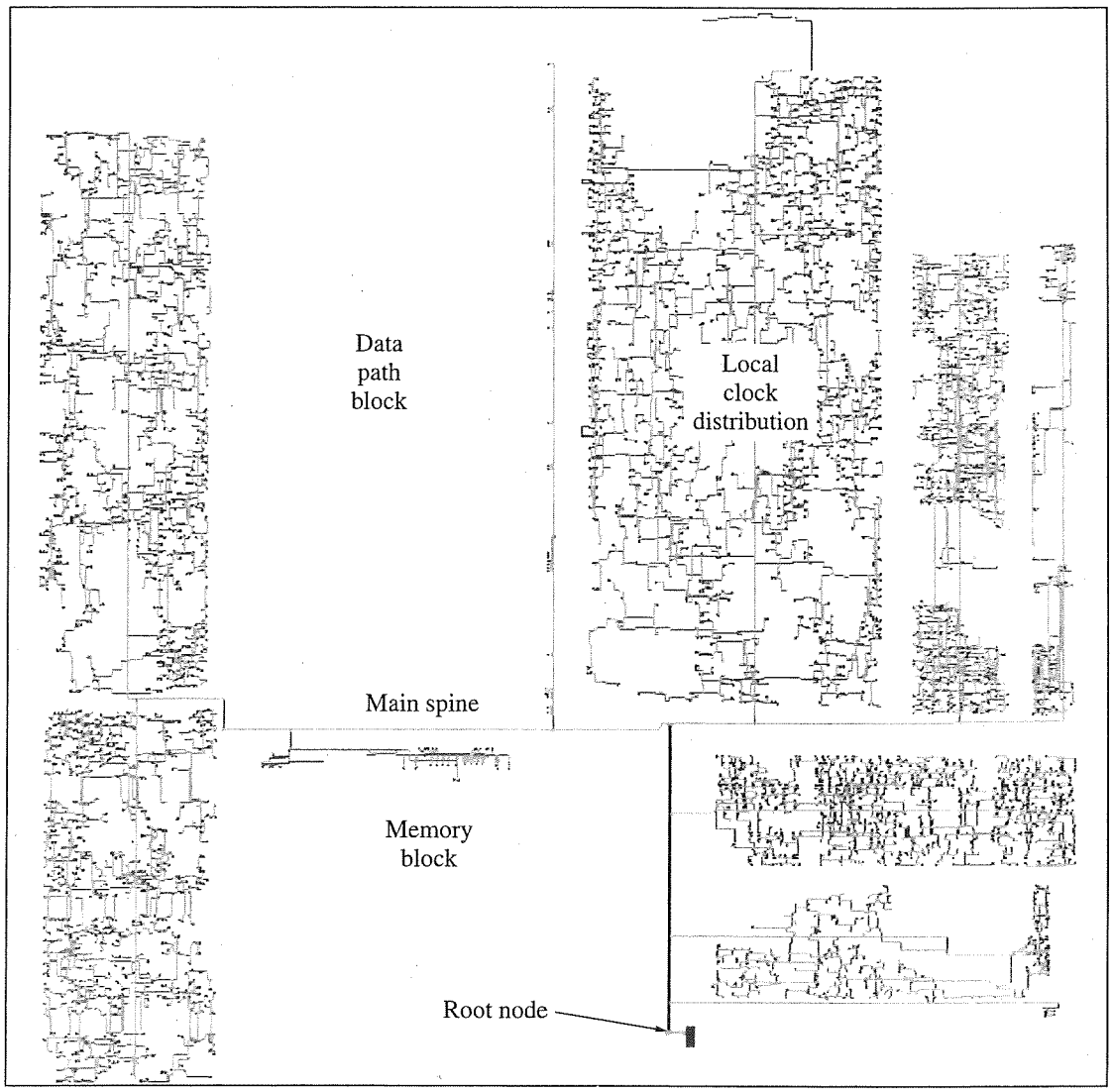


Figure 11.32
Clock tree routing for industrial chip.

number of major spines, from which local clocks emanate. The two empty regions are the memory and data path areas. The clock drives the decoders in the memory circuit (just below the *main spine* label) and the data path is driven by the clock spine on the right side. The rest of the blocks obtain their local clocks from internal spines that run through them.

The buffers in this clock tree are of the form shown in Figure 11.30b with uniform sizes. These buffers draw a significant amount of current when switching. In fact, the section labeled *local clock distribution* has such a large number of buffers that, when they switch simultaneously, it creates significant *IR* drop in the power grid. Recall from Figure 11.14 that the *IR* drop was most significant in this region

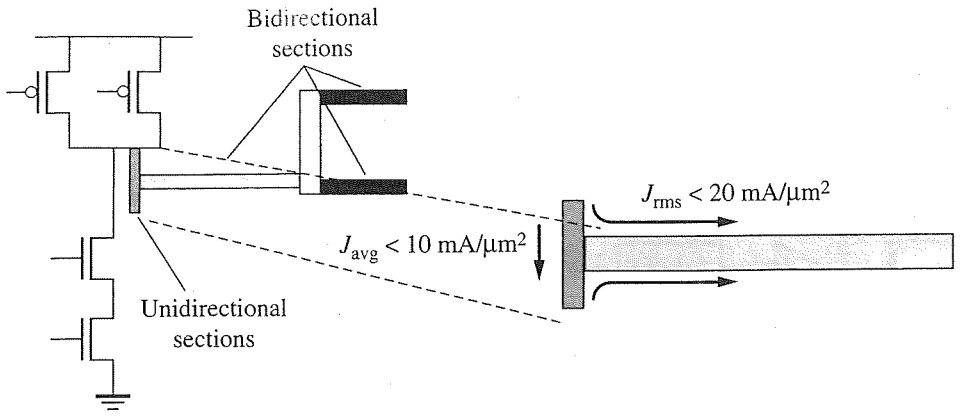


Figure 11.33
Electromigration effects in signal lines due to ac and dc current flow.

and violated the 10% noise budget. This *IR* drop translates into additional delay, resulting in additional clock skew. Ironically, the clock itself is causing the problem of *IR* drop. This is why the clock and power distribution systems are often designed together.

Recently, electromigration (EM) problems have been encountered on clock signal lines as a result of large currents that flow back and forth on these lines. EM on signal lines is therefore an ac phenomenon. As a result, we must compute the *root-mean-square (RMS)* value of the current rather than the average current (which tends to be close to zero for the ac case). The RMS current density, J_{rms}, is compared to another EM limit, $J_{AC,max}$, that is based on ac *Joule heating* as current moves back and forth in the wire. A typical value of $J_{AC,max}$ is 20 mA/μm^2. The failure criterion is

$$J_{rms} > J_{AC,max} \tag{11.8}$$

If the actual RMS current density is below the tolerable amount, the signal line is expected to have a lifetime well beyond the life of the corresponding product.

Consider Figure 11.33 which illustrates the EM problem areas for the clock. The metal segments at the output of the gate experience both unidirectional (dc) and bidirectional (ac) current flow mechanisms. Most of the wire encounters bidirectional current due to charging and discharging of capacitances. Therefore, the current density criteria would be based on Equation (11.8) for EM failure. However, a short unidirectional segment of metal connects the PMOS devices to the NMOS devices. It would be evaluated using the criteria in Equation (11.5) for EM failure. In the figure, the criteria to *pass* the EM checks are shown for each of the two segments.

Figure 11.34 shows the results of an EM analysis on a portion of the clock tree of Figure 11.32. The areas that are highlighted with squares have violated the EM criteria, either (11.5) or (11.8). All signal EM failures must be fixed since any break in the wire is catastrophic. Again, wire widening is the typical solution to this problem.

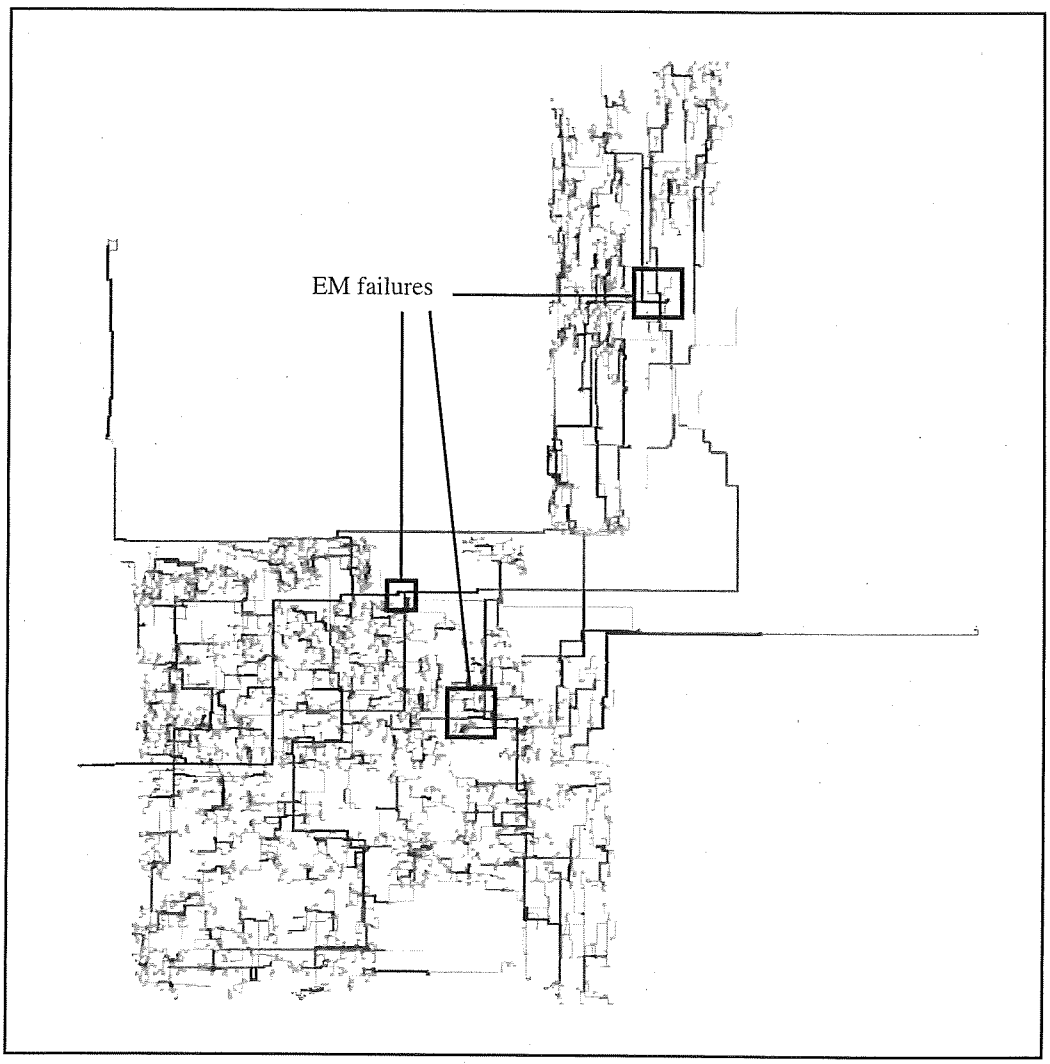


Figure 11.34
Signal EM failures in clock tree.

*11.4 Phase-Locked Loops/Delay-Locked Loops

In this section, we examine the problem of synchronizing the externally supplied master clock with the internal clock of the chip. In digital VLSI systems with multiple chips on a printed circuit board (PCB) or in system-on-chip (SOC) applications, severe problems may arise due to latency between the master reference clock and locally generated slave clocks. The value of the on-chip clock latency may differ from chip to chip or from module to module. Since the data transfers are synchronized with the slave clocks, timing hazards may occur and/or data errors may be generated (data may be sampled at a wrong time). These problems can be rectified using *phase-locked loop* (PLL) and *delay-locked loop* (DLL) circuits.

On-chip PLLs can be used to synchronize the internal clock of a chip with the external clock and data, and to generate the internal clock signal running at a higher rate of operation than the external clock input. Consider the timing diagram of

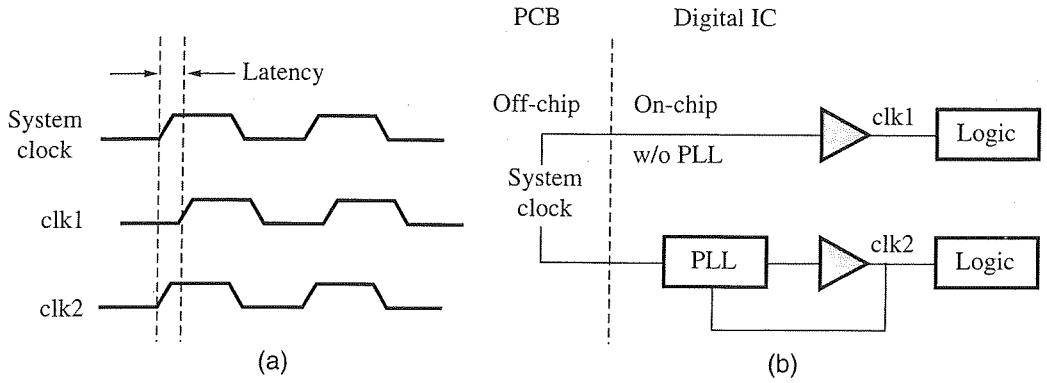


Figure 11.35
On-chip PLL for clock synchronization.

Figure 11.35a. The external system clock of the PCB is the off-chip clock. This clock is fed into a chip to be used as the internal clock as shown in the top portion of Figure 11.35b. The on-chip clock is buffered to drive the flops or latches of the logic blocks. However, the delay through the clock buffers and wires (i.e., clock latency) creates a difference between the internal and external clocks as seen in clk1 of Figure 11.35a. This leads to additional setup and hold requirements at the I/O pins of the chip. Ideally, we would like the two edges to line up so that the internal clock is synchronized with the external clock, as in the clk2 waveform of Figure 11.35a. To do this, we could use a delay-locked loop or phase-locked loop, as shown for clk2 in Figure 11.35b.

The PLL and DLL circuits are closely related.[2] Both use feedback control to lock the output clock to the incoming clock. PLLs must lock on to the frequency and phase of the reference clock, whereas DLLs simply lock to a constant phase of the reference clock. Therefore, the locking process in a PLL requires more time (since it must first lock on to the frequency and then the phase) than a DLL (since it must only lock on to the phase). If we simply want synchronization, we could use a DLL. However, if we needed the internal clock to run at a multiple of the external clock frequency, we must use a PLL.

The PLL and DLL feedback control systems have very similar architectures, as will be seen shortly. The main difference is the use of a voltage-controlled oscillator (VCO) in the PLL and a voltage-controlled delay line (VCDL) in the DLL. There are also differences in the stability properties of the two types of loops, and their ability or inability to perform frequency synthesis. Many books on analog design cover these types of circuits and their properties in great detail due to their widespread use in other applications.

As digital designers, it seems that PLLs/DLLs are beyond the scope of this book due to their analog nature. This is essentially a true statement. However, it is important to understand the basic operation of a PLL circuit, as well as DLL, due to their

[2] Even their names imply that they are related. That is, the term *delay* in the time domain has the same meaning as *phase* in the frequency domain.

increased use for skew management, clock synthesis, and data and clock recovery. We therefore present the operation and design considerations from a digital designer's point of view using a basic PLL block diagram as a guideline. The reader is encouraged to consult the references at the end of the chapter for further information.

11.4.1 PLL Design Considerations

Figure 11.36 shows a block diagram of a charge pump PLL for clock synchronization and clock synthesis. It consists of a phase/frequency detector (PFD), a charge pump (CP), a loop filter (LPF), and a voltage-controlled oscillator (VCO). The phase detector detects the phase difference between the reference clock and the VCO output and applies charge-up or charge-down pulses to the charge pump. These pulses are used to switch voltage or current sources, which charge or discharge its output. The pulses are filtered by the loop filter and applied as the control voltage of the VCO. The VCO changes its oscillation frequency according to the control voltage. If the VCO oscillates at a multiple of the input frequency, it must be divided down before a comparison is performed in the PFD.

The phase/frequency detector (PFD) is a circuit that detects the difference between the edges of the reference clock (Ref) and the feedback clock (FB). A simple version is shown in Figure 11.37a consisting of two D-flops and an AND gate. The D-inputs are tied to 1. The role of this circuit is to control the VCO by moving its frequency up or down depending on the edges of the incoming clocks.

The truth table for the PFD is given in Figure 11.37b. If the *up* and *down* signals are both 0, then the desired frequency and phase have been obtained and the PLL is in the locked condition. However, if the reference clock switches ahead of the feedback clock, the *up* signal is raised to indicate that the frequency of the VCO should be increased. Similarly, if the feedback clock switches ahead of the reference clock, a *down* signal is generated to slow down the VCO. If both the *up* and *down* signals are high, it is viewed as a reset condition, so the AND gate acts to reset the flops to 0.

To understand its operation further, consider the timing diagram of Figure 11.37c. Initially Ref Clk switches high ahead of FB Clk. This is a phase error in the two signals. The initial rise of Ref Clk sets its flop output to 1 while the rising edge

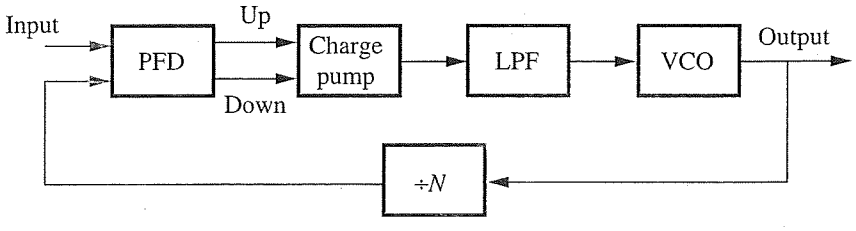


Figure 11.36
A basic PLL block diagram.

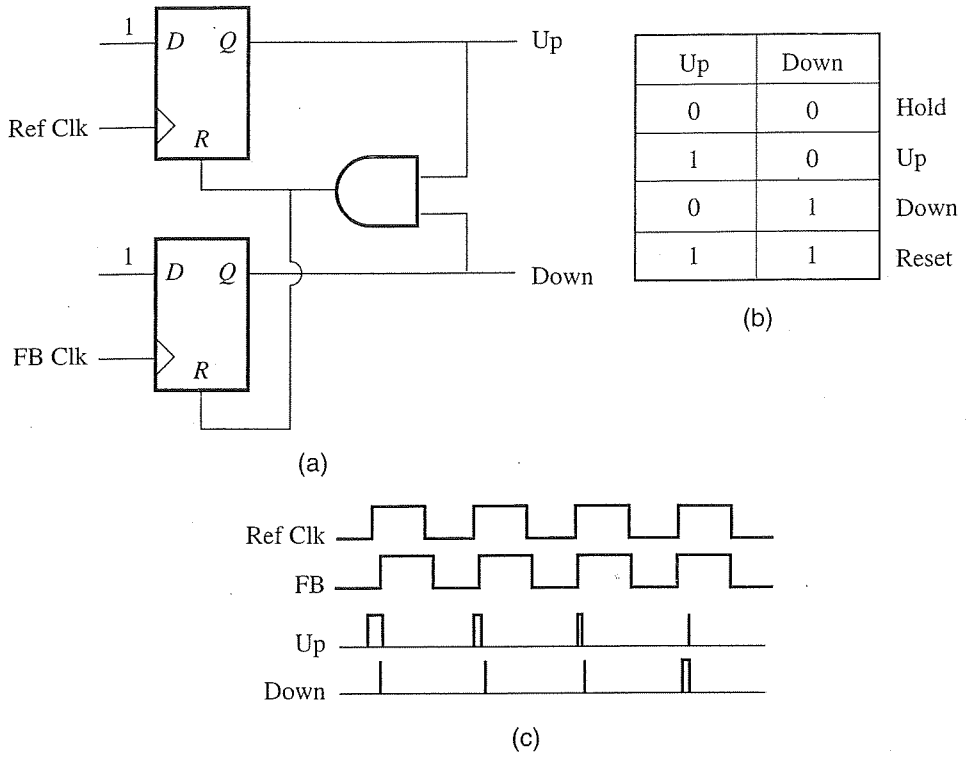


Figure 11.37
Phase-frequency detector.

of FB Clk sets its flop output to 1. The combination of the two high outputs forces a reset. The pulse width on the *up* signal is proportional to the phase delay of the two signals. The *down* signal produces a short pulse that can easily be removed by the next stage, as will be seen. Since the VCO receives a "speedup" signal, the next set of edges of the two clocks will have a smaller phase difference. This shortens the *up* pulse. This continues until the phase difference switches signs, at which point the *down* signal is enabled. Eventually, the system stabilizes and there is a negligible difference between the two clocks.

Once the *up*/*down* signals have been generated, they are applied to a charge-pump which is illustrated in Figure 11.38 along with the output filter. Assume that the filter is a capacitor for the time being. The *up* signal will switch on the upper current source, I_P, and raise the output voltage at V_C. Similarly, the *down* signal will discharge the output capacitance and lower the control voltage, V_C. There are situations when both switches turn on simultaneously. However, as long as $I_P = I_N$, the overlapping pulses for the *up* and *down* signals do not create a problem since both switches are on and the current flows harmlessly from V_{DD} to Gnd.[3]

[3] In practice, it is difficult to get these two currents to match over supply and process variations, and temperature changes. This leads to a dc offset in the output that must be removed in some way (typically with a unity-gain amplifier and some additional switches).

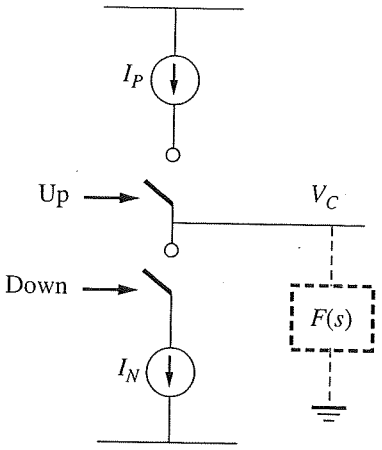


Figure 11.38
Charge-pump and loop filter $F(s)$.

The filter $F(s)$ determines the order and stability of the overall PLL system. Its role is to filter out the high-frequency switching components in the *up/down* signals and deliver a slowly changing control voltage to the VCO. Therefore, it must be designed in the context of the overall feedback control system and not in isolation. Even though the PLL is a highly nonlinear system, it can be analyzed as a linear system when it is in the locked or nearly locked condition. The control system representation is given in Figure 11.39, which we now analyze using linear system theory. The phase is the loop variable in this case, and the phase difference is determined and amplified by K_{PD} in the PFD:

$$v_d = K_{PD}(\theta_{\text{in}} - \theta_{\text{out}})$$

This is passed through the low-pass filter with a transfer function of $F(s)$. Then the signal passes through the VCO to create the output phase. The relationship between frequency and phase is that phase is the integral of frequency, and the frequency is determined by the control voltage:

$$\theta = \int f dt = \int K_o v_c dt$$

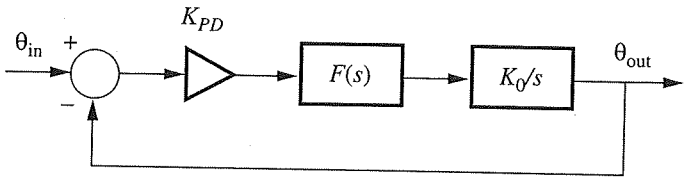


Figure 11.39
Linear feedback control system for PLL.

Therefore, the VCO can be viewed as an integrator that converts the control voltage into a phase. Its transfer function is given by K_0/s. The meaning of this transform is that the VCO introduces a pole into the system transfer function at $s = 0$. Any additional poles in the transfer function of $F(s)$ may lead to instability of the overall system, depending on their exact placement. This is why the loop filter plays such a strong role in the overall stability of the PLL.

We can now determine the two important transfer functions of the system: the open loop gain and the closed loop gain. The open loop gain is obtained by examining the direct path from the input to the output:

$$\frac{\theta_{\text{out}}}{\theta_{\text{in}}}(s) = K_{PD}F(s)\frac{K_0}{s} = G(s)$$

The closed loop gain is obtained by going around the loop once and can be written in terms of the open loop gain:

$$K_{PD}(\theta_{\text{in}}(s) - \theta_{\text{out}}(s))F(s)\frac{K_0}{s} = \theta_{\text{out}}(s) \tag{11.9}$$

$$\therefore \frac{\theta_{\text{out}}}{\theta_{\text{in}}}(s) = \frac{K_{PD}F(s)\dfrac{K_0}{s}}{1 + K_{PD}F(s)\dfrac{K_0}{s}} = \frac{G(s)}{1 + G(s)}$$

The classical study of the stability properties of such a system is carried out by analyzing the frequency response of the open loop system. This can be understood by examining Equation (11.9) above. If $G(s)$ in the denominator is equal to -1, the closed loop gain goes to infinity. The condition $G(s) = -1$ occurs when the magnitude of $G(s)$ is 1 and its phase is $-180°$. In practical terms, the system is said to be unstable in this condition. By examining Equation (11.9), the term that determines the stability of the system is the loop filter $F(s)$.

In Figure 11.40, three options are presented for the loop filter. We had assumed that the loop filter could be implemented as a grounded capacitance. This is a first-order filter (with one pole) and it produces a second-order PLL (since the PLL already has one pole due to the integrator). However, the use of a simple capacitor as the loop filter renders the PLL unstable since it introduces two poles at $s = 0$, which immediately sets the phase to $-180°$.

If a resistor is placed in series with the capacitor as in the second diagram, it inserts a zero into the transfer function and this allows the overall frequency response to be stabilized for the PLL. This is also a second-order PLL since the filter is of first-order. When a time-domain simulation is performed with this filter, we would notice that the charge-pump introduces small step discontinuities in the value of V_C as it switches, due to capacitive feedthrough. These discontinuities can be damped by adding a shunt capacitance as in the third diagram. While this is a more complicated filter, it can be shown that the PLL can be stabilized by the proper

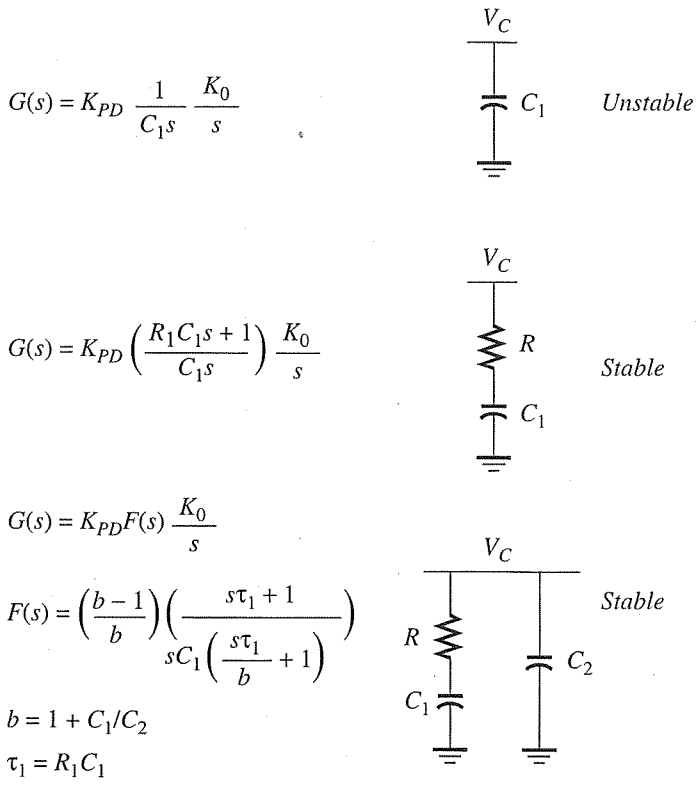


Figure 11.40
Filter options for PLL low-pass filter.

selection of R, C_1, and C_2. Typically $C_2 = (1/10)C_1$. The second-order filter makes the PLL a third-order system. Most on-chip PLLs use this type of filter.

The next block to consider is the VCO which acts as an integrator and generates a periodic output based on a control voltage input. A typical VCO for digital applications is presented in Figure 11.41a. This is a ring oscillator as we have seen before except that we have a controllable delay, τ. The frequency of oscillation is $f = 1/2N\tau$, where N is an odd number. The τ of each stage can be adjusted by the control voltage V_C. In Figure 11.41b, we show a "current-starved" inverter where the control voltage adjusts the amount of current delivered to the inverter to charge and discharge the next stage. As the current varies, the delay through the inverter varies as well. By connecting an odd number of stages in series, the oscillator can be designed to generate the required clock signal. As the control voltage is adjusted, the oscillation frequency changes over a prescribed range. Obviously, this figure is oversimplified compared to the actual VCOs but the basic principle is the same.

Since the output of the VCO generates the internal clock signal, it will have to be buffered so that it can drive the large loads associated with the clock tree. The appropriate buffer sizing methods are required so that the output of the VCO does not directly drive a large load. The last step of the PLL loop is the divider circuit, shown in Figure 11.36. The purpose of this digital block is to reduce the frequency of the VCO output to that of the incoming reference clock so that a proper phase comparison can be carried out.

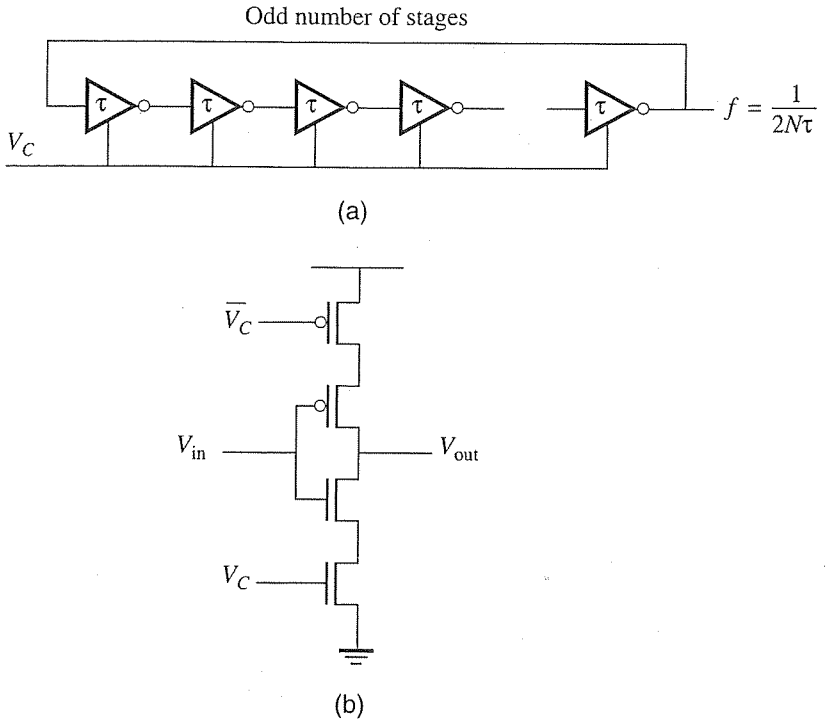


Figure 11.41
(a) Typical VCO. (b) Current-starved inverter.

This completes our brief look at the PLL. Clearly this is a challenging analog circuit design, especially in the presence of temperature and process variations. A critical design block is the VCO since it generates the clock used by the chip and is prone to *jitter* problems due to supply noise and other factors. As described earlier, jitter is the drift of the clock edge relative to the desired position. Note that the jitter at the PLL input is not propagated to the output since the VCO generates a new clock signal. Therefore, the input jitter is effectively filtered out by PLL. One layout problem is that the PLL is sensitive to the switching noise of other circuits. The interaction with other blocks must be minimized for proper operation. In addition, simulations of such circuits have been known to take a long time to complete. The use of this type of circuit will increase as chips get larger and there are more issues of clock synchronization on chips.

The other useful circuit in digital VLSI designs is a DLL block that synchronizes the data with the clock signal. The PLL can be modified to implement a DLL by a substitution of the VCO with a voltage-controlled delay line (VCDL) (shown in Figure 11.42). The same type of analysis described for the PLL can be applied to the DLL. The circuit utilizes the control signal, V_C, to vary the delay of each inverter. The VCDL circuit can be designed with multiple cascaded stages to construct a delay line. This circuit is popular since it has better stability properties than the PLL and is much easier to design. It can be stabilized using a simple capacitor as the loop filter. However, it transfers input jitter directly to the output. This circuit has been widely used in clock and data recovery applications.

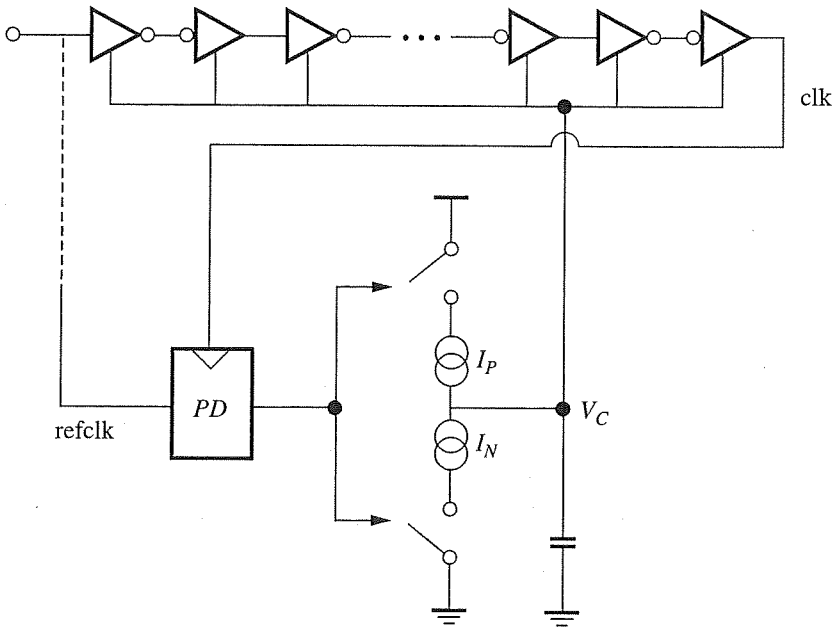


Figure 11.42
DLL with VCDL (Voltage-Controlled-Delay Line).

11.4.2 Clock Distribution Summary

To summarize, the key to proper clock design is to balance as much of the design as possible. First, the clock skew should be minimized by using architectures such as a buffered H-tree. It should be controlled by limiting the maximum and minimum interconnect length, using symmetric tree architectures with equal-sized buffers, and using an *all-or-none* strategy for gated clocks. The last stage of buffering to drive the local flip-flops may not be identical since the fanouts are not necessarily the same. However, the delay through the final buffers should be made the same. The clock edges should be kept as sharp as possible. The proper tradeoffs should be made to shield clock lines with V_{DD} and Gnd to limit crosstalk. The power should be minimized through the careful control of capacitance, and the use of gated clocks. The supply voltage should be regulated through the use of decoupling capacitors, especially in the area of the large clock buffers that switch simultaneously. PLL or DLL circuits should be used to synchronize external and internal clock signals.

Alpha 21264 is a good example of integrating clock hierarchy in its clocking network design, as illustrated in Figure 11.43. The global clock (GCLK) is generated by an on-chip, low-jitter phase-locked loop (PLL). The GCLK signal is routed along a trunk to the center of the die and is distributed by X-trees and H-trees to 16 distributed GCLK drivers located in a windowpane pattern across the chip. Figure 11.44 shows the clock hierarchy of the microprocessor. Local clocks and local conditional clocks (gated clocks) are driven several stages past GCLK. The novelty of this approach is the use of a clock grid structure, much like the power grid structure

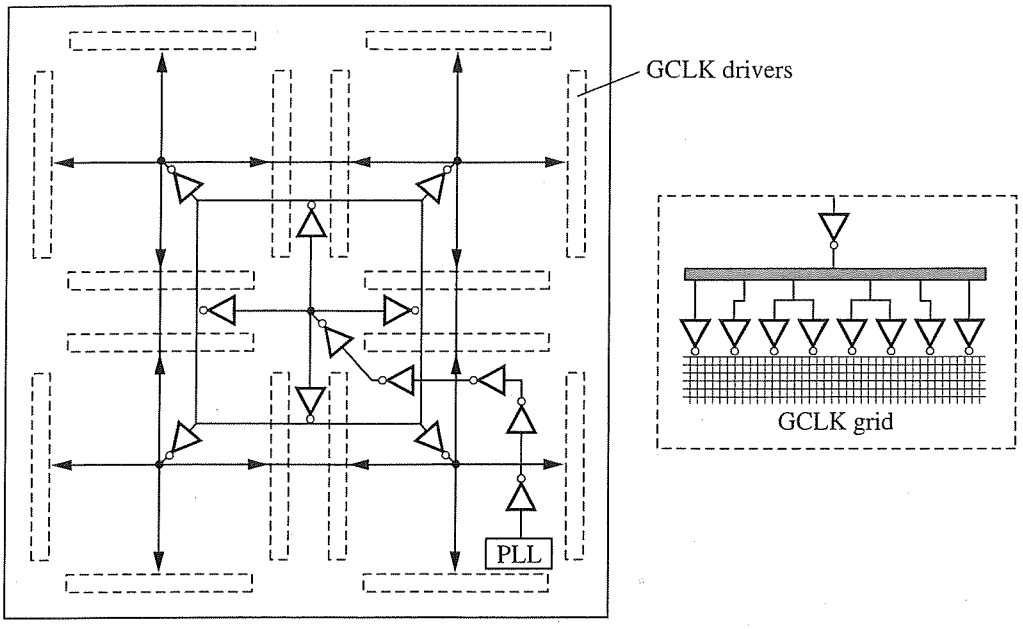


Figure 11.43
Global clock (GCLK) distribution network.

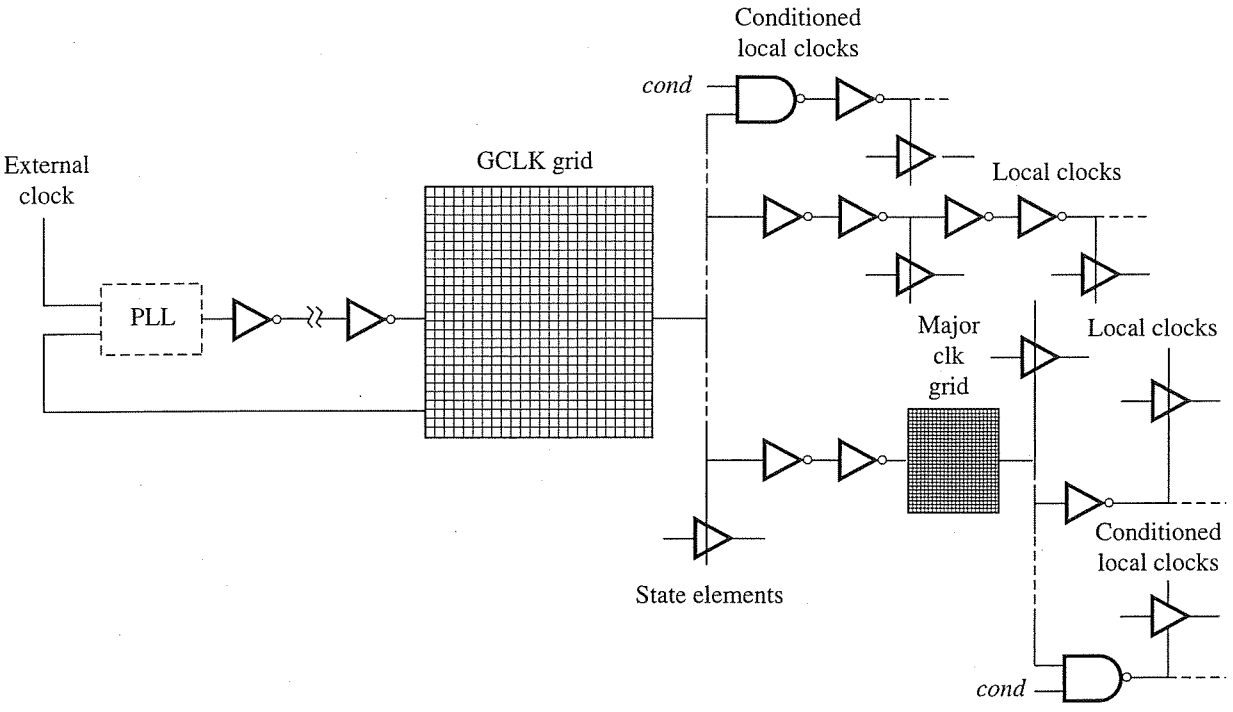


Figure 11.44
Clock hierarchy of 600 MHz alpha microprocessor.

described earlier in this chapter to reduce clock skew. State elements and clocking points exist from zero to eight gates past GCLK. There are six other global clocks, referred to as box clocks, that drive large grids over their respective execution units: a floating point, a bus interface, a load/store, integer pads, and an instruction issue. Smaller local clocks are generated as needed from any clocks without strict limits on the number, size, or logic function of the local buffers or on the requirements on the duty cycle of the generated clocks.

REFERENCES

1. M. Pedram and J. Rabaey, *Power Aware Design Methodologies.* Kluwer Academic Publishers, Boston, MA, 2002.
2. F. Gardiner, *Phaselock Techniques.* John Wiley–Interscience, 2nd ed., New York, 1979.
3. I. Young, J. Greason, and K. Wong, "A PLL Clock Generator with 5 to 110 MHz of Lock Range for Microprocessors," *JSSC*, Vol. 27, No. 11, November 1992, pp. 1599–1607.
4. Behrad Razavi, *Design of Analog CMOS Integrated Circuits.* McGraw-Hill, New York, 2001.

PROBLEMS

P11.1 List several known methods to address *IR* drop problems. Which of these methods are also effective for Ldi/dt and EM issues?

P11.2 (a) A 0.4 μm wide wire running 400 μm is used for a power bus in a 0.18 μm technology. It is connected to a pin with an inductance of $L = 2$ nH. There are 32 inverters with $W = 1$ μm connected to the power line, half of which are switching high simultaneously in a 100 ps time period. What is the worst-case voltage drop at the far end of the power bus (near inv32)? Parameters: Metal 5 resistivity = 54m$\Omega/\square$, thickness = 0.8 μm.

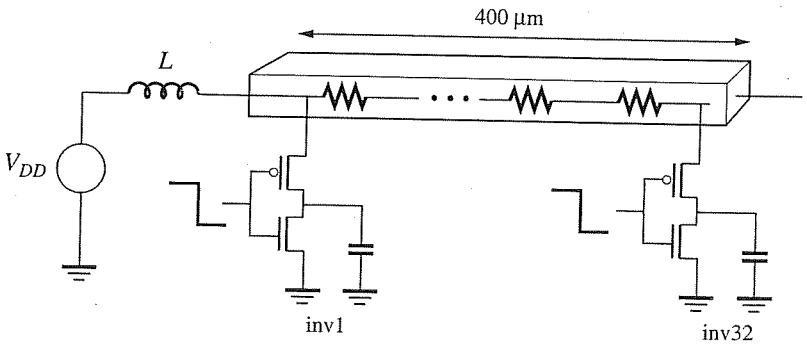


Figure P11.2

(b) Assuming that $J_{max} = 10^6$ A/cm, does the wire pass the electromigration check?

(c) If not, what width of wire is needed to pass the electromigration check?

P11.3 In the simultaneous switching case shown in Figure P11.3, 31 of the 32 output drivers switch from high to low while the one on the far right remains "stable." Assuming that the bonding wire inductance is 10 nH, how much ground bounce is experienced when each inverter discharges 5 pF over 1.8 ns through the chip ground? $V_{DD} = 1.8$ V.

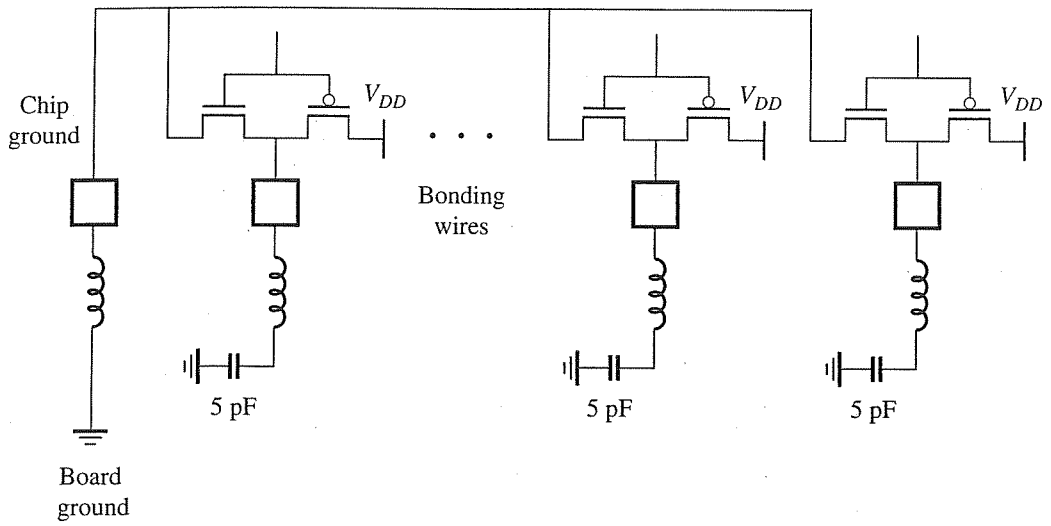


Figure P11.3

P11.4 In the 2-phase non-overlapping clock system shown in Figure P11.4, what is the minimum delay needed between Phi2 falling and Phi1 rising for the system to work properly? Assume that the maximum skew between any of the clocks is 1 ns, and that the delay through the latches is 0. Explain your answer. (Hint: Note that the answer could be positive or negative.) Also, what is the minimum clock cycle?

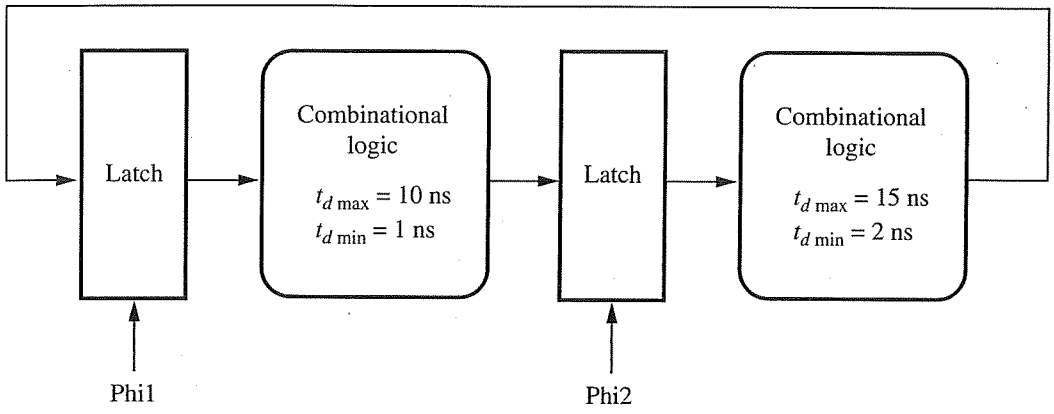


Figure P11.4

P11.5 In the above problem, redo the analysis assuming flip-flops in place of latches and a single clock.

P11.6 In the system shown in Figure P11.6 which uses two edge-triggered flip-flops, find the minimum clock cycle time. Assume that the setup time of the flop is 0.4 ns, the clk-to-q delay is 0.3 ns, and the hold time is 0.1 ns. The worst-case clock skew is 0.6 ns. The dotted lines in the combinational logic (CL) block indicate the longest delay along the path from input to output. Highlight the critical path, and label which clock is skewed *early* and which is skewed *late* for your calculation of the clock cycle. (Hint: think about the time to go from the output of one flop to the input of the next and take into account all the time components to make this trip.)

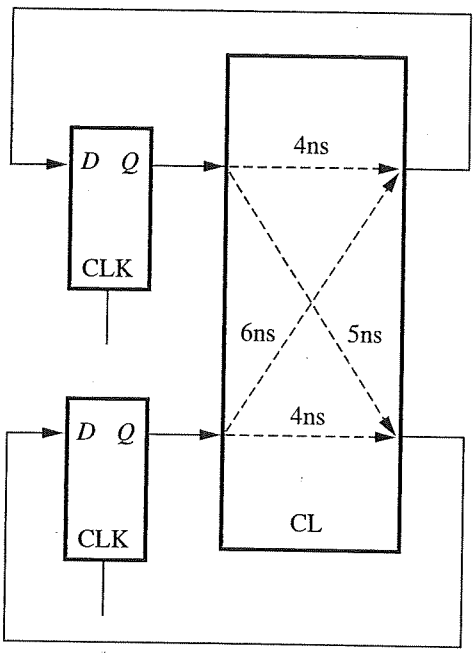


Figure P11.6

P11.7 The circuit shown in Figure P11.7 is to be used to design a decoupling capacitor and, in particular, to select the proper channel length. Simulate this circuit in SPICE by applying an initial condition to the supply node of 1.8 V (do not use a voltage source here). Plot the voltage at the supply node when the inverter output is switched from low to high. Generate a family of curves by changing the W and L of the decoupling capacitor, but maintain a constant value of gate area for the decap.

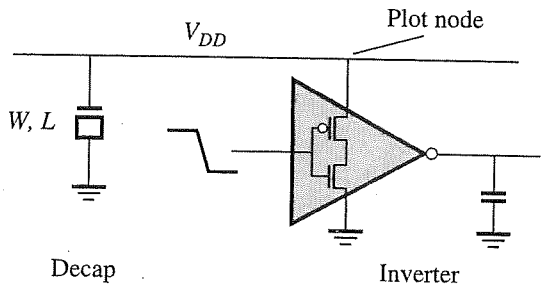


Figure P11.7

What is the optimal value of L according to your results? Assume 0.18 μm technology parameters. You will have to use a nonquasistatic model in SPICE.

P11.8 For the circuit in Figure 11.27, plot the incoming clock signal and the output clock signals. Design the circuit so that it has equal delays from the Clk to both outputs.

P11.9 For the circuit in Figure 11.41b, use SPICE to plot τ versus V_C if $W =$ 0.5 μm for all devices. Assume that the inverter drives another identical inverter. Given the results from SPICE, what is the frequency range of the VCO in Figure 11.41a if there are 25 stages?

P11.10 You are required to drive a 10 mm wire with minimum delay. However, your input capacitance must be 12λ (i.e., due to the input capacitance of a 2X inverter). Answer the following questions using rapid "back-of-the-envelope" calculations, and then redo the problem using detailed analysis. Finally, compare the results with SPICE.

(a) First draw a schematic of the circuit involving only inverters and wires that will produce the minimum delay. Your circuit should have inverters that start at 2X and increase in size until they reach the buffers that are inserted to drive the wire segments. Show the entire delay path from end to end.

(b) Size the inverters to produce the minimum delay.

(c) Compute the delay value from the 2X inverter to the end of the wire.