VLSI Design EE 523
Spring 2025

Shahid Masud

Lecture 16

Topics for lecture 16

- Power Gating and Clock Gating Techniques
- Sources of Static Power Dissipation
- Some Discussion on Dynamic Power Dissipation
- Some solved examples of Power Dissipation in CMOS circuits
- Some solved examples from these topics
- **MIDTERM EXAM NEXT LECTURE**

Power Dissipation Questions – Sedra Smith book

EXERCISES

14.19 Find the dynamic power dissipation of the inverter analyzed in Example 14.7 when operated at a 1-GHz frequency. Recall that $C = 6.25$ fF and $V_{DD} = 2.5$ V.
Ans. 39 μW.

14.20 Find the dynamic power dissipation of a CMOS inverter operated from a 1.8-V supply and having a load capacitance of 100 fF. Let the inverter be switched at 100 MHz.
Ans. 32.4 μW

14.21 A particular inverter circuit initially designed in a 0.5-μm process is fabricated in a 0.13-μm process. Assuming that the capacitance C will scale down in proportion to the minimum feature size (more on this in the next chapter) and that the power supply will be reduced from 5 V to 1.2 V, by what factor do you expect the dynamic power dissipation to decrease? Assume that the switching frequency f remains unchanged.
Ans. 66.8

Estimate Switching Power – E.g. from textbook

Example 5.1

A digital system-on-chip in a 1 V 65 nm process (with 50 nm drawn channel lengths and λ = 25 nm) has 1 billion transistors, of which 50 million are in logic gates and the remainder in memory arrays. The average logic transistor width is 12 λ and the average memory transistor width is 4 λ. The memory arrays are divided into banks and only the necessary bank is activated so the memory activity factor is 0.02. The static CMOS logic gates have an average activity factor of 0.1. Assume each transistor contributes 1 fF/μm of gate capacitance and 0.8 fF/μm of diffusion capacitance. Neglect wire capacitance for now (though it could account for a large fraction of total power). Estimate the switching power when operating at 1 GHz.

SOLUTION: There are $(50 \times 10^6$ logic transistors$)(12\ \lambda)(0.025\ \mu m/\lambda)((1 + 0.8)$ fF/μm) = 27 nF of logic transistors and $(950 \times 10^6$ memory transistors$)(4\ \lambda)(0.025\ \mu m/\lambda)((1 + 0.8)$ fF/μm) = 171 nF of memory transistors. The switching power consumption is $[(0.1)(27 \times 10^{-9}) + (0.02)(171 \times 10^{-9})](1.0\ V)^2(10^9\ Hz) = 6.1$ W.

Dynamic Power Example

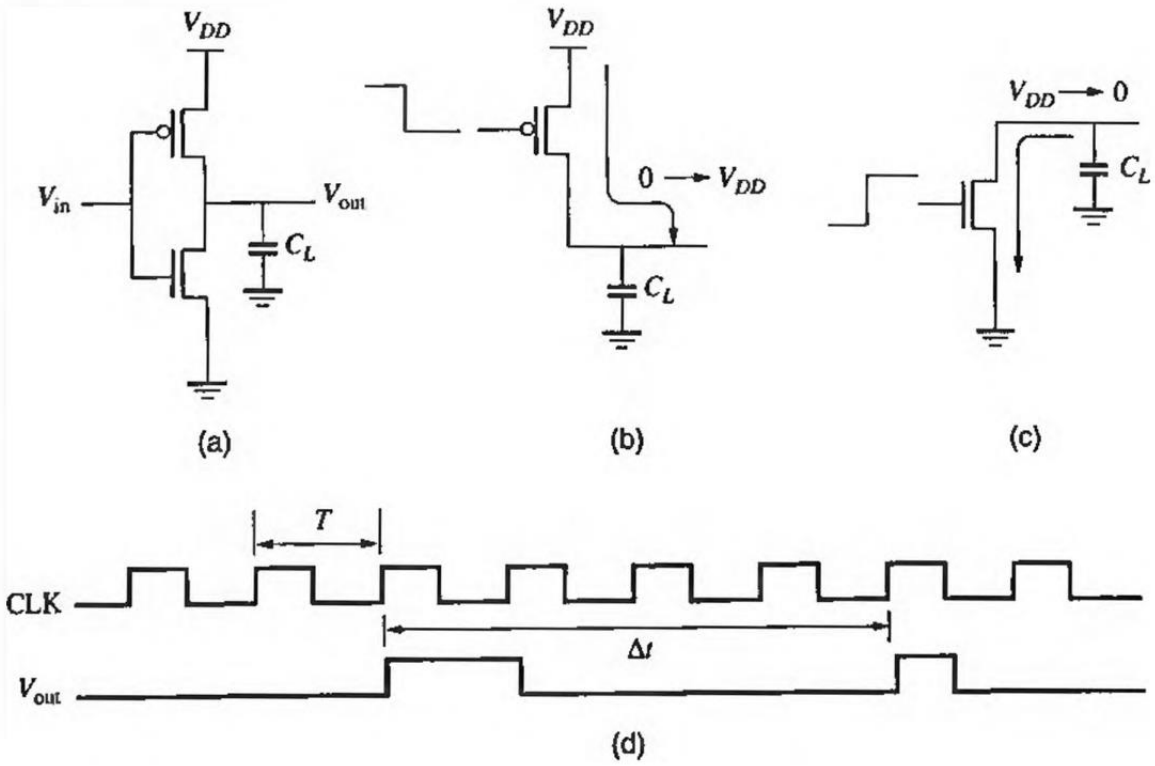

Figure 5.29
Dynamic power dissipation considerations.

Activity Factor Calculation

Problem:

In Figure 5.29d, we have a total of four toggles of the output over the duration of eight clock cycles. What is the activity factor for this node?

Solution:

Number of clock cycles = 8
Number of toggles at output = 4. Two toggles are required for power dissipation.

$$\alpha_{0\to1} = \frac{\#\text{ toggles}/2}{\#\text{ clock cycles}} = \text{activity factor (switching factor)} = \frac{4/2}{8} = 25\%$$

Short-Circuit Component in Dynamic Power

Dynamic power also includes a short-circuit power component caused by power rushing from V_{DD} to GND when both the pullup and pulldown networks are partially ON while a transistor switches. This is normally less than 10% of the whole, so it can be conservatively estimated by adding 10% to the switching power.

Switching power is consumed by delivering energy to charge a load capacitance, then dumping this energy to GND. Intuitively, one might expect that power could be saved by shuffling the energy around to where it is needed rather than just dumping it. Resonant circuits, and adiabatic charge-recovering circuits [Maksimovic00, Sathe07] seek to achieve such a goal. Unfortunately, all of these techniques add complexity that detracts from the potential energy savings, and none have found more than niche applications.

Clock Gating

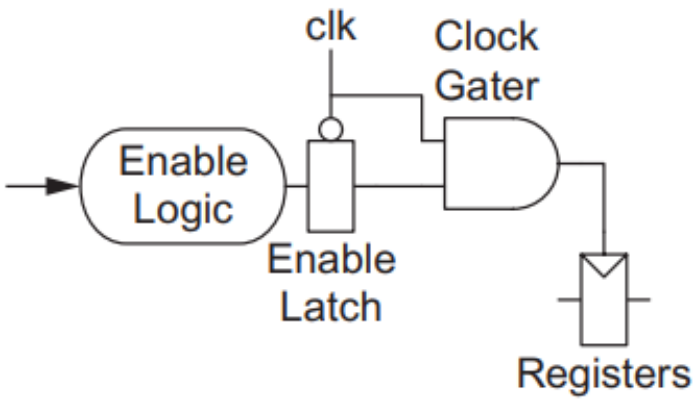

FIGURE 5.7 Clock gating

5.2.1.1 Clock Gating Clock gating ANDs a clock signal with an enable to turn off the clock to idle blocks. It is highly effective because the clock has such a high activity factor, and because gating the clock to the input registers of a block prevents the registers from switching and thus stops all the activity in the downstream combinational logic.

Clock gating can be employed on any enabled register. Section 10.3.5 discusses enabled register design. Sometimes the logic to compute the enable signal is easy; for example, a floating-point unit can be turned off when no floating-point instructions are being issued. Often, however, clock gating signals are some of the most critical paths of the chip.

The clock enable must be stable while the clock is active (i.e., 1 for systems using positive edge-triggered flip-flops). Figure 5.7 shows how an enable latch can be used to ensure the enable does not change before the clock falls.

When a large block of logic is turned off, the clock can be gated early in the clock distribution network, turning off not only the registers but also a portion of the global network. The clock network has an activity factor of 1 and a high capacitance, so this saves significant power.

Switching Probability

5.2.1.2 Switching Probability Recall that the activity factor of a node is the probability that it switches from 0 to 1. This probability depends on the logic function. By analyzing the probability that each node is 1, we can estimate the activity factors. Although designers don't manually estimate activity factors very often, the exercise is worth doing here to gain some intuition about switching activity.

Define P_i to be the probability that node i is 1. $\overline{P}_i = 1 - P_i$ is the probability that node i is 0. α_i, the activity factor of node i, is the probability that the node is 0 on one cycle and 1 on the next. If the probability is uncorrelated from cycle to cycle,

$$\alpha_i = \overline{P}_i P_i \tag{5.14}$$

Completely random data has $P = 0.5$ and thus $\alpha = 0.25$. Structured data may have different probabilities. For example, the upper bits of a 64-bit unsigned integer representing a physical quantity such as the intensity of a sound or the amount of money in your bank account are 0 most of the time. The activity factor is lower than 0.25 for such data.

Switching probabilities

Gate	P_Y
AND2	$P_A P_B$
AND3	$P_A P_B P_C$
OR2	$1 - \overline{P}_A \overline{P}_B$
NAND2	$1 - P_A P_B$
NOR2	$\overline{P}_A \overline{P}_B$
XOR2	$P_A \overline{P}_B + \overline{P}_A P_B$

Example 5.2

Figure 5.8 shows a 4-input AND gate built using a tree (a) and a chain (b) of gates. Determine the activity factors at each node in the circuit assuming the input probabilities $P_A = P_B = P_C = P_D = 0.5$.

SOLUTION: Figure 5.9 labels the signal probabilities and the activity factors at each node based on Table 5.1 and EQ (5.14). The chain has a lower activity factor at the intermediate nodes.

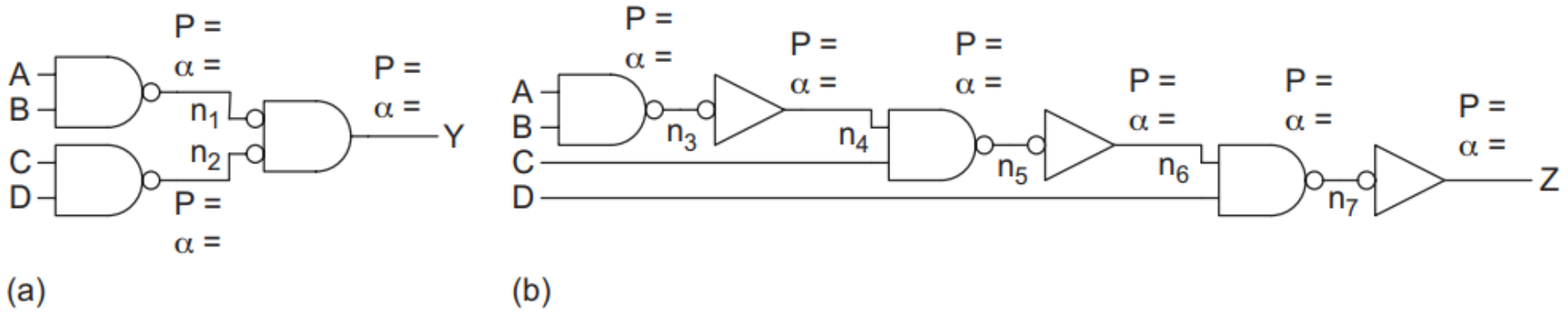

FIGURE 5.8 4-input AND circuits

Solution to Activity Factor Example

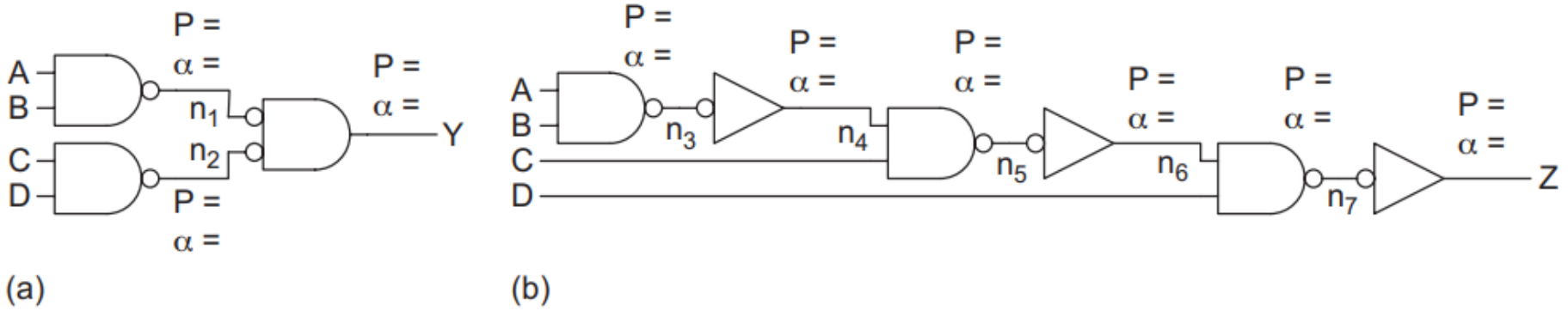

FIGURE 5.8 4-input AND circuits

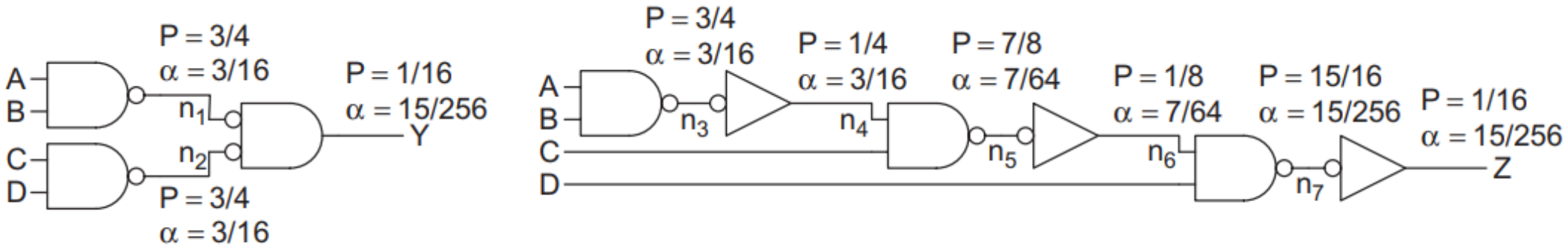

FIGURE 5.9 Signal probabilities and activity factors

Capacitance Effect of Power

5.2.2 Capacitance

Switching capacitance comes from the wires and transistors in a circuit. Wire capacitance is minimized through good floorplanning and placement (the locality aspect of structured design). Units that exchange a large amount of data should be placed close to each other to reduce wire lengths.

Device-switching capacitance is reduced by choosing fewer stages of logic and smaller transistors. Minimum-sized gates can be used on non-critical paths. Although Logical Effort finds that the best stage effort is about 4, using a larger stage effort increases delay only slightly and greatly reduces transistor sizes. Therefore, gates that are large or have a high activity factor and thus dominate the power can be downsized with only a small performance impact. For example, buffers driving I/O pads or long wires may use a stage effort of 8–12 to reduce the buffer size. Similarly, registers should use small clocked transistors because their activity factor is an order of magnitude greater than transistors in combinational logic. In Chapter 6, we will see that wire capacitance dominates many circuits. The most energy-efficient way to drive long wires is with inverters or buffers rather than with more complex gates that have higher logical efforts [Stan99].

Voltage Effect on Power

5.2.3 Voltage

Voltage has a quadratic effect on dynamic power. Therefore, choosing a lower power supply significantly reduces power consumption. As many transistors are operating in a velocity-saturated regime, the lower power supply may not reduce performance as much as long-channel models predict. The chip may be divided into multiple *voltage domains*, where each domain is optimized for the needs of certain circuits. For example, a system-on-chip might use a high supply voltage for memories to ensure cell stability, a medium voltage for a processor, and a low voltage for I/O peripherals running at lower speeds. In Section 5.3.2, we will examine how voltage domains can be turned off entirely to save leakage power during sleep mode.

Voltage also can be adjusted based on operating mode; for example, a laptop processor may operate at high voltage and high speed when plugged into an AC adapter, but at lower voltage and speed when on battery power. If the frequency and voltage scale down in proportion, a cubic reduction in power is achieved. For example, the laptop processor may scale back to 2/3 frequency and voltage to save 70% in power when unplugged.

Voltage Domains and Power

5.2.3.1 Voltage Domains Some of the challenges in using voltage domains include converting voltage levels for signals that cross domains, selecting which circuits belong in which domain, and routing power supplies to multiple domains.

Figure 5.14 shows direct connection of inverters in two domains using high and low supplies, V_{DDH} and V_{DDL}, respectively. A gate in the V_{DDH} domain can directly drive a gate in the V_{DDL} domain. However, the gate in the V_{DDL} domain will switch faster than it would if driven by another V_{DDL} gate. The timing analyzer must consider this when computing the contamination delay, lest a hold time be violated. Unfortunately, the gate in the V_{DDL} domain cannot directly drive a gate in the V_{DDH} domain. When n_2 is at V_{DDL}, the pMOS transistor in the V_{DDH} domain has $V_{gs} = V_{DDH} - V_{DDL}$. If this exceeds V_t, the pMOS will turn ON and burn contention current. Even if the difference is less than V_t, the pMOS will suffer substantially increased leakage. This problem may be alleviated by using a high-V_t pMOS device in the receiver if the voltage difference between domains is small enough [Tawfik09].

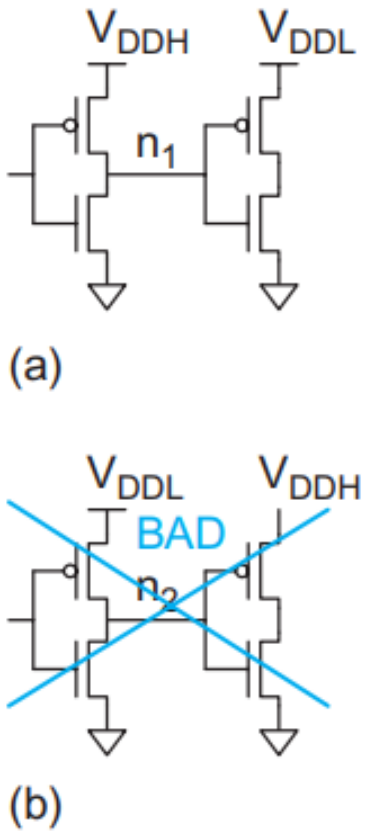

FIGURE 5.14
Voltage domain crossing

Voltage Level Converter Circuit

The standard method to handle voltage domain crossings is a *level converter*, shown in Figure 5.15. When $A = 0$, $N1$ is OFF and $N2$ is ON. $N2$ pulls Y down to 0, which turns on $P1$, pulling X up to V_{DDH} and ensuring that $P2$ turns OFF. When $A = 1$, $N1$ is ON and $N2$ is OFF. $N1$ pulls X down to 0, which turns on $P2$, pulling Y up to V_{DDH}. In either case, the level converter behaves as a buffer and properly drives Y between 0 and V_{DDH} without risk of transistors remaining partially ON. Unfortunately, the level converter costs delay (about 2 FO4) and power at each domain crossing. [Kulkarni04] and [Ishihara04] survey a variety of other level converters. The cost can be partially alleviated by building the converter into a register and only crossing voltage domains on clock cycle boundaries.

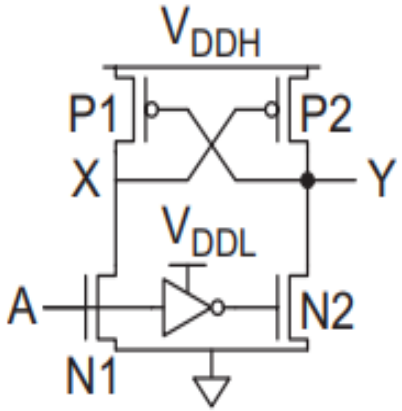

FIGURE 5.15 Level converter

Dynamic Voltage Scaling

5.2.3.2 Dynamic Voltage Scaling (DVS) Many systems have time-varying performance requirements. For example, a video decoder requires more computation for rapidly moving scenes than for static scenes. A workstation requires more performance when running SPICE than when running Solitaire. Such systems can save large amounts of energy by reducing the clock frequency to the minimum sufficient to complete the task on schedule, then reducing the supply voltage to the minimum necessary to operate at that frequency. This is called *dynamic voltage scaling* (DVS) or *dynamic voltage/frequency scaling* (DVFS) [Burd00]. Figure 5.17 shows a block diagram for a basic DVS system. The DVS controller takes information from the system about the workload and/or the die temperature. It determines the supply voltage and clock frequency sufficient to complete the workload on schedule or to maximize performance without overheating. A switching voltage regulator efficiently steps down V_{in} from a high value to the necessary V_{DD}. The core logic contains a phase-locked loop or other clock synthesizer to generate the specified clock frequency.

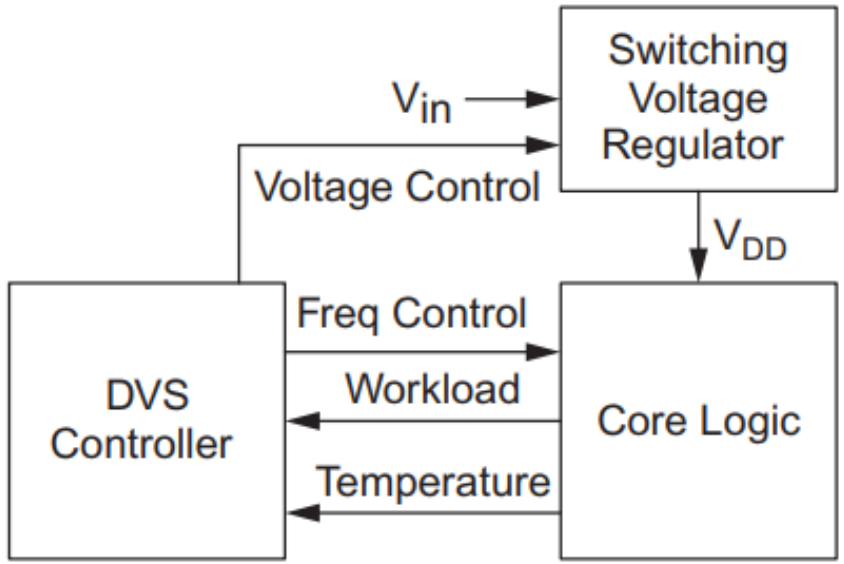

FIGURE 5.17 DVS system

Sources of Static Power Dissipation

- Subthreshold Leakage Current
- Gate Leakage (Tunneling)
- Junction Leakage (via Substrate)
- Contention Issues (Due to Circuit Design – i.e. Pseudo NMOS – later)

E.g. related to Leakage current and Static Power

Example 5.4

Consider the system-on-chip from Example 5.1. Subthreshold leakage for OFF devices is 100 nA/μm for low-threshold devices and 10 nA/μm for high-threshold devices. Gate leakage is 5 nA/μm. Junction leakage is negligible. Memories use low-leakage devices everywhere. Logic uses low-leakage devices in all but 5% of the paths that are most critical for performance. Estimate the static power consumption.

SOLUTION: There are $(50 \times 10^6$ logic transistors$)(0.05)(12\ \lambda)(0.025\ \mu m/\lambda) = 0.75 \times 10^6$ μm of low-threshold devices and $[(50 \times 10^6$ logic transistors$)(0.95)(12\ \lambda) + (950 \times 10^6$ memory transistors$)(4\ \lambda)](0.025\ \mu m/\lambda) = 109.25 \times 10^6\ \mu$m of high-threshold devices. Neglecting the benefits of series stacks, half the transistors are OFF and contribute subthreshold leakage. Half the transistors are ON and contribute gate leakage. $I_{sub} = [(0.75 \times 10^6\ \mu m)(100\ nA/\mu m) + (109.25 \times 10^6\ \mu m)(10\ nA/\mu m)]/2 = 584$ mA. $I_{gate} = ((0.75 + 109.25) \times 10^6\ \mu m)(5\ nA/\mu m)/2 = 275$ mA. $P_{static} = (584\ mA + 275\ mA)(1\ V) = 859$ mW. This is 15% of the switching power and is enough to deplete the battery of a hand-held device rapidly.

Power Header Switch Sizing

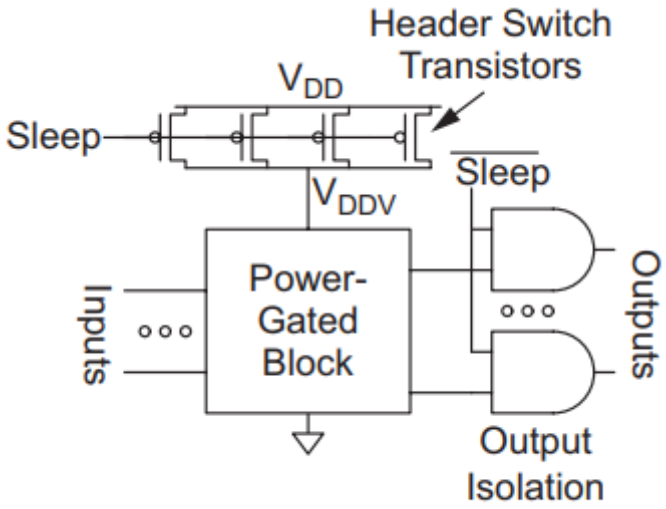

FIGURE 5.24 Power gating

Example 5.5

A cache in a 65 nm process consumes an average power of 2 W. Estimate how wide should the pMOS header switch be if delay should not increase by more than 5%?

SOLUTION: The 65 nm process operates at 1 V, so the average current is 2 W / 1 V = 2 A. The pMOS transistor has an ON resistance of $R = 2\ \text{k}\Omega \cdot \mu\text{m}$. A 5% delay increase corresponds to a droop on V_{DDV} of about 5% (check this using EQ (4.29). Thus, $R_{\text{switch}} = 0.05 \times 1\ \text{V} / 2\ \text{A} = 25\ \text{m}\Omega$. So the transistor width must be $\text{k}\Omega \cdot \mu\text{m}/25\ \text{m}\Omega = 8 \times 10^4\ \mu\text{m}$. The ON resistance at low V_{ds} is lower than R. Circuit simulation shows that a width of $3.7 \times 10^4\ \mu\text{m}$ suffices to keep droop to 5%.

4.4.6.4 Voltage Dependence Designers often need to predict how delay will vary if the supply or threshold voltage is changed. Recalling that delay is proportional to CV_{DD} / I and using the α-power law model of EQ (2.30) for I_{dsat}, we can estimate the scaling of the RC time constant and of gate delay as

$$\tau = k \frac{CV_{DD}}{\left(V_{DD} - V_t\right)^{\alpha}} \qquad (4.29)$$

where k reflects process parameters.

Readings

- Chapter 5 of textbook ‘CMOS VLSI Design’ by Weste and Harris related to Power Dissipation in CMOS circuits