

CMOS Gate Delays, Power, and Scaling

GATE DELAYS

Earlier in the term (Lec. 16) we calculated the gate delay for a symmetrical CMOS inverter with

$$V_{Tn} = |V_{Tp}| \equiv V_T, C_{oxn}^* = C_{oxp}^* \equiv C_{ox}^*, \text{ and } K_n = K_p,$$

in which both the n- and p-channel devices were minimum gate length devices, i.e., $L_n = L_p = L_{min}$. The p-channel device was made twice as wide as the n-channel device to get the desired K equality, because we assumed $\mu_e = 2 \mu_h$.

We found that the gate delay was given by:

$$t_{GD} = \frac{4 C_L V_{DD}}{K_n (V_{DD} - V_T)^2}$$

Replacing C_L and K_n , to write this in terms of the device dimensions, we found after a bit of simple algebra:

$$t_{GD} = \frac{12 n}{\mu_e} L_{min}^2 \frac{V_{DD}}{(V_{DD} - V_T)^2}$$

POWER

There is zero static power in CMOS so the only contribution is the dynamic power

$$P_{ave} = C_L V_{DD}^2 f$$

where f is the operating frequency and C_L is the loading capacitance. This load will be the average fan-out, n , times the input capacitance of a similar CMOS gate, plus any parasitic interconnect capacitance:

$$\begin{aligned} C_L &= n C_{ox}^* (L_{min} W_n + L_{min} W_p) + C_{parasitic} \\ &= 3 n C_{ox}^* L_{min} W_n + C_{parasitic} \end{aligned}$$

Neglecting $C_{\text{parasitic}}$, we can write

$$P_{\text{ave}} = 3 n C_o^* \times L_{\text{min}} W_n V_D^2 f$$

MAXIMUM POWER

The maximum power dissipation will occur when the gate is operated at its maximum frequency (bit rate), which is in turn proportional to $1/\tau_{GD}$. Thus we can say

$$P_{\text{ave max}} \approx 3 n C_o^* \times L_{\text{min}} W_n V_D^2 \frac{1}{\tau_{GD}}$$

$$= \frac{1}{4} \frac{W_n}{L_{\text{min}}} \mu_e C_o^* \times V_{DD} (V_{DD} - V_T)^2 = \frac{1}{4} K_n V_{DD} (V_{DD} - V_T)^2$$

The importance of keeping V_{DD} small is quite evident from this expression, but the situation is not black and white because making V_{DD} small makes τ_{GD} large; the same is true of making K_n small. The whole problem of what to reduce how while maintaining high performance and not frying the IC chips is a complex one and has led to the development of rules for scaling dimensions and voltages; we will discuss scaling rules after first looking at one more important parameter, the maximum average power dissipation per unit area.

POWER DISSIPATION PER UNIT AREA

In many situations the power dissipation per unit area is more important than the total power dissipation. To estimate how this factor varies with the device dimensions we make the assumption that the density of devices in an integrated circuit increases inversely with the gate area, $W_n L_{\text{min}}$. We have:

$$P_{\text{density max}} \approx \mu \frac{P_{\text{ave max}}}{W_n L_{\text{min}}} \approx \mu \frac{\mu_e C_o^*}{4 L_{\text{min}}^2} V_{DD} (V_{DD} - V_T)^2$$

SCALING RULES

We in general want to simultaneously reduce gate delays, decrease power dissipation, and increase packing density, while not exceeding a certain power density. The place we start is with a reduction of the gate length, but we quickly find we must do more than that or we get into trouble.

For example, as the gate length is reduced, the oxide thicknesses and the junction depths (of the sources and drains) must be reduced proportionally to obtain good transistor characteristics. One is essentially maintaining a long, thin geometry consistent with the gradual channel approximation, and this turns out to be just what is needed to get good saturation (flat curves; small g_o) of the device output (i_D vs v_{DS}) characteristics. Thus, if we reduce the minimum gate length, L_{min} , by a factor of s , we will also want to reduce the gate oxide, t_{ox} , by the same factor. To increase the packing density further, we also reduce the gate width, W , by the same factor:

$$\begin{array}{ll} L_{min} & L_{min}/s \\ W & W/s \\ t_{ox} & t_{ox}/s \end{array}$$

With these changes we find that our gate delay, average power, device density, and power density change as follows:

$$\begin{array}{ll} GD & GD/s^2 \\ P_{ave} & s P_{ave} \\ \text{Device Density} & s^2 \text{ Device Density} \\ P_{density\ max} & s^3 P_{density\ max} \end{array}$$

Clearly this is a formula for disaster because the power density will increase dramatically if we only scale dimensions. We either have to develop much better ways to get the heat out of an IC chip and package, so we can tolerate a higher power density, or we have to change more than the dimensions. Packaging and heat sinking have been improved, to be sure, but the big gain comes from scaling the voltages as well as the dimensions. If we scale the supply and threshold voltages as follows:

$$V_{DD} \quad V_{DD}/s$$

$$V_T \quad V_T/s$$

then we find:

$$GD \quad GD/s$$

$$P_{ave} \quad P_{ave}/s^2$$

$$\text{Device Density} \quad s^2 \text{ Device Density}$$

$$P_{\text{density max}} \quad P_{\text{density max}}$$

This is clearly a much better situation. At the same time it must be noted that it is not as easy to scale the voltages as it might at first seem and it has taken longer to do so than it has to reduce dimensions because of a number of factors. The control over the threshold voltage must be improved which places more demands on the process line, and the noise margins decrease by a factor $1/s$ so noise sources on the chip must be reduced. Also, supply voltages are not totally arbitrary since they must be tied to standard battery cells, which come in increments of roughly 1 Volt (they range from 1.1 to 0.9 V over their useful lifetime). Early bipolar and MOSFET logic used V_{DD} 's of 5 V, but this has recently been reduced to 3, 2, and, even, 1 V.

Scaling examples:

<u>Parameter</u>	Intel Families		
	<u>386</u>	<u>486</u>	<u>Pentium</u>
Scaling factor, s	1	2	3
L_{\min} (μm)	1.5	0.75	0.5
w_n (μm)	10	5	3
t_{ox} (nm)	30	15	9
V_{DD} (V)	5	3.3	2.2
V_{T} (V)	1	-	-
Fan out	3	3	3
K ($\mu\text{A}/\text{V}^2$)	230	450	600
t (ps)	840	400	250
f_{max} (MHz)	29	50	100
$P_{\text{ave/gate}}$ (μW)	92	23	10
Density (kgates/cm ² @ 20 W/cm ² max)	220	880	2,000

Sources: Professor Jesus del Alamo and Intel

Intel Pentium Families

<u>Parameter</u>	<u>486</u>	<u>Pentium generations</u>		
L_{\min} (μm)	1.0	0.8	0.5	0.35
Scaling factor, s	-	1	1.6	2.3
SRAM Cell Area (μm^2)	-	111	44	21
Die size (mm^2)	170	295	163	91
f_{\max} (MHz)	38	66	100	200
t_{ox} (nm)	20	10	8	6
Metal layers	2	3	4	4
Planarization	SOG	CMP	CMP	CMP
Poly type	n	n, p	n, p	n, p
Transistors	CMOS	BiCMOS	BiCMOS	BiCMOS

Source: Dr. Leon D. Yau, Intel, MIT VLSI Seminar, Cambridge, MA, Oct. 8, 1996. (This table is meant to illustrate the trend; see the second12/6/01 handout for data from 2000.)

Yet another view:

CMOS Scaling Trends

From: "Design Challenges in Multi-GHz Microprocessors," by Bill Herrick, Alpha/Compaq, MIT VLSI Symposium, 2/15/00

Moore's Law: the trend that the demand for IC functions and the capacity of the semiconductor industry to meet that demand, will double every 1.5 to 2 years.

Historical Trends: Then and Now

Circa 1970

Circa 2000

12 μm PMOS

0.18 μm CMOS

1000 transistors

10-100 million transistors

5-10 mm² die size

300-400 mm² die size

10V supply

2.5 V supply

50-100 kHz frequency

500-1000 Mhz frequency

100-200 mW

50-100 W

16 pin DIPs

500-1000 pin BGAs

Intel Trends

The 4004 (1971)

2300 transistors in a 10 μm process

108 kHz operation, executing 0.06 MIPs

The Pentium III (1999)

28 million transistors in a 0.18 μm process

733 Mhz operation, executes 2000 MIPs