CMOS Power Consumption

Lecture 13
18-322 Fall 2003

Textbook: [Sections 5.5 5.6 6.2 (p. 257-263) 11.7.1 ]
Overview

- Low-power design
  - Motivation
  - Sources of power dissipation in CMOS
  - Power modeling
  - Optimization Techniques (a survey)
Why worry about power?
-- Heat Dissipation

Handhelds

Portables

Desktops

Servers
Power Density Trends

![Graph showing power density trends with various processors and comparison to nuclear reactor and rocket nozzle.]

Surpassed hot-plate power density in 0.5\(\mu\)m
Not too long to reach nuclear reactor

Courtesy of Fred Pollack, Intel
CoolChips tutorial, MICRO-32
High End Power Consumption

- While you can probably afford to pay for 100-200W of power for your desktop...

- Getting that heat off the chip and out of the box is expensive
A Booming Market: Portable Devices

What we'd like…
- Video decompression
- Speech recognition
- Protocols, ECC, ...
- Handwriting recognition
- Text/Graphics processing
- Java interpreter

- Up to 1 month of uninterrupted operation!

What we would need…

![Graph showing expected battery lifetime increase over next 5 years: 30-40%](image-url)
Where Does Power Go in CMOS?

- **Switching power**: due to charging and discharging of output capacitances:
  \[
  \text{Energy/transition} = C_L \times V_{dd}^2 \\
  \text{Power} = \text{Energy/transition} \times f = C_L \times V_{dd}^2 \times f
  \]

- Short-circuit power: due to non-zero rise/fall times
- Leakage power (important with decreasing device sizes)
  - Typically between 0.1nA - 0.5nA at room temperature
Short-Circuit Power

- Inputs have finite rise and fall times
  - Depends on device sizes

- Direct current path from \( V_{DD} \) to GND while PMOS and NMOS are ON simultaneously for a short period

\[
P_{SC} = I_{mean} \cdot V_{DD}
\]
Leakage Current

\[ I_O = i_s (e^{\frac{V_q}{kT}} - 1) \]

Sub-threshold current

\[ I_D = K \cdot e^{(V_{gs} - V_t)q / nkT} \left( 1 - e^{V_{ds}q / kT} \right) \]
New Problem: Gate Leakage

- Now about 20-30% of all leakage, and growing
- Gate oxide is so thin, electrons tunnel thru it…
- NMOS is much worse than PMOS
Gate/Circuit-Level Power Estimation

- It is a very difficult problem

▲Challenges

▲ $V_{DD}$, $f_{clk}$, $C_L$ are known
  • Actually, the layout will determine the interconnect capacitances

▲ Need *node-by-node* accuracy
  • Power dissipation is highly data-dependent

▲ Need to estimate switching activity accurately
  • Simulation may take days to complete
Dynamic Power Consumption - Revisited

Power = Energy/transition * transition rate

\[ \text{Power} = C_L * V_{dd}^2 * f_{0\rightarrow1} \]

\[ = C_L * V_{dd}^2 * P_{0\rightarrow1} * f \]

\[ = C_{EFF} * V_{dd}^2 * f \]

\[ P = C_L (V_{dd}^2/2) f_{clk \text{SW}} \]

\[ C_{EFF} = \text{Effective Capacitance} = C_L * P_{0\rightarrow1} \]

Power Dissipation is Data Dependent
Function of Switching Activity
Example: Static 2 Input NOR

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>Out</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

Truth Table of 2 input NOR gate

Assume:

\[ P(A=1) = \frac{1}{2} \]
\[ P(B=1) = \frac{1}{2} \]

Then:

\[ P(\text{Out}=1) = \frac{1}{4} \] (this is the *signal probability*)
\[ P(0 \rightarrow 1) = P(\text{Out} = 0) \cdot P(\text{Out} = 1) \]
\[ = \frac{3}{4} \times \frac{1}{4} = \frac{3}{16} \] (this is the *transition probability*)

\[ C_{\text{EFF}} = \frac{3}{16} C_L \]
Power Consumption is Data Dependent

Suppose now that only patterns 00 and 11 can be applied (with equal probabilities). Then:

\[
\begin{align*}
0\rightarrow 0 &\quad 0\rightarrow 0 &\quad 1\rightarrow 1 \\
0\rightarrow 0 &\quad 0\rightarrow 1 &\quad 1\rightarrow 0 \\
0\rightarrow 0 &\quad 1\rightarrow 0 &\quad 0\rightarrow 1 \\
0\rightarrow 0 &\quad 1\rightarrow 1 &\quad 0\rightarrow 0 \\
0\rightarrow 1 &\quad 0\rightarrow 0 &\quad 1\rightarrow 0 \\
0\rightarrow 1 &\quad 0\rightarrow 1 &\quad 1\rightarrow 0 \\
0\rightarrow 1 &\quad 1\rightarrow 0 &\quad 0\rightarrow 0 \\
0\rightarrow 1 &\quad 1\rightarrow 1 &\quad 0\rightarrow 0 \\
1\rightarrow 0 &\quad 0\rightarrow 0 &\quad 0\rightarrow 1 \\
1\rightarrow 0 &\quad 0\rightarrow 1 &\quad 0\rightarrow 0 \\
1\rightarrow 0 &\quad 1\rightarrow 0 &\quad 0\rightarrow 1 \\
1\rightarrow 0 &\quad 1\rightarrow 1 &\quad 0\rightarrow 0 \\
1\rightarrow 1 &\quad 0\rightarrow 0 &\quad 0\rightarrow 0 \\
1\rightarrow 1 &\quad 0\rightarrow 1 &\quad 0\rightarrow 0 \\
1\rightarrow 1 &\quad 1\rightarrow 0 &\quad 0\rightarrow 0 \\
1\rightarrow 1 &\quad 1\rightarrow 1 &\quad 0\rightarrow 0
\end{align*}
\]

Similarly, suppose that every 0 applied to the input A is immediately followed by a 1 while every 1 applied to B is immediately followed by a 0. \(P(0\rightarrow 1) = ?\)
Transition Probabilities for Basic Gates

<table>
<thead>
<tr>
<th></th>
<th>$P_{0\rightarrow1}$</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AND</strong></td>
<td>$(1-P_A P_B)P_A P_B$</td>
</tr>
<tr>
<td><strong>OR</strong></td>
<td>$(1-P_A)(1-P_B)(1-(1-P_A)(1-P_B))$</td>
</tr>
<tr>
<td><strong>EXOR</strong></td>
<td>$(1 - (P_A + P_B - 2P_A P_B))(P_A + P_B - 2P_A P_B)$</td>
</tr>
</tbody>
</table>

Switching Activity for Static CMOS

$$P_{0\rightarrow1} = P_0 P_1$$
(Big) Problem: Re-convergent Fanout

In this case, \( Z = B \) as it can be easily seen. The previous analysis simply fails because the signals are not independent!

\[
P(Z=1) = P(B=1) \cdot P(X=1 \mid B=1) = P(B=1)
\]

Main issue: Becomes complex and intractable real fast!
Another (Big) Problem: Glitching in Static CMOS

also called: dynamic hazards

<table>
<thead>
<tr>
<th>ABC</th>
<th>101</th>
<th>000</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Z</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Unit Delay

wasted power
Example: A Chain of NAND Gates
Glitch Reduction Using Balanced Paths

Equalize Lengths of Timing Paths Through Design
Delay is important: Delay vs. $V_{DD}$ and $V_T$

Think about (Power $\times$ Delay) product!

■ Delay for a 0->1 transition to propagate to the output:

$$t_{pLH} = \frac{C_L V_{DD}}{k_n (V_{DD} - V_{Tn})^2}$$

Similar for a 1->0 transition
Delay vs. $V_{DD}$
Power-Performance Trade-offs

■ Prime choice: $V_{DD}$ reduction
  ✓ In recent years we have witnessed an increasing interest in supply voltage reduction (e.g. Dynamic Voltage Scaling)
    • High $V_{DD}$ on critical path or for high performance
    • Low $V_{DD}$ where there is some available slack
  ✓ Design at very low voltages is still an open problem (0.6 – 0.9V by 2010!)
    • Ensures lower power
    • … but higher latency – loss in performance

■ Reduce switching activity
  ✓ Logic synthesis
  ✓ Clock gating

■ Reduce physical capacitance
  ✓ Proper device sizing
  ✓ Good layout
How about POWER?
Ways to reducing power consumption

- Load capacitance ($C_L$)
  - Roughly proportional to the chip area

- Switching activity (avg. number of transitions/cycle)
  - Very data dependent
  - A big portion due to glitches (real-delay)

- Clock frequency ($f$)
  - Lowering only $f$ decreases average power, but total energy is the same and throughput is worse

- Voltage supply ($V_{DD}$)
  - Biggest impact

![Graph showing normalized delay vs. voltage supply for different components]
Using parallelism (1)

\[ P_{\text{ref}} = C_{\text{ref}} V_{\text{DD}}^2 f_{\text{ref}} \]

Assume: \( t_p = 25\text{ns} \) (worst-case, all modules) at \( V_{\text{DD}} = 5\text{V} \)
Using parallelism (2)

- \( C_{par} = 2.15C \) (extra-routing needed)
- \( f_{par} = f/2 \) \( (t_{p,new} = (50)\text{ns} \Rightarrow V_{DD} \sim 2.9V; \quad V_{DD,par} = 0.58 \ V_{DD}) \)
- \( P_{par} = C_{par} V_{DD}^2 f_{par} = 0.36 P_{ref} \)

Area increases about 3.4 times!
Using pipelining

- $C_{pipe} = 1.15C$
- Delay decreases 2 times ($V_{DD,pipe} = 0.58 V_{DD}$)
- $P_{pipe} = 0.39 P$
Question for you:

Which of the two designs is more energy efficient?

Assume:

- Zero-delay model
- All inputs have a signal probability of 0.5

Hint: Calculate $p_{0 \rightarrow 1}$ for W, X and F
Chain vs. balanced design

- For the zero-delay model
  - Chain design is better
  - But ignores **glitching**
    - Depending on the gate delays, the chain design may be worse
Low energy gates – transistor sizing

- Use the *smallest transistors* that satisfy the delay constraints
  - Increasing transistor size improves the speed but it also increases power dissipation (since the load capacitances increases)
    - Slack time - difference between required time and arrival time of a signal at a gate output
      - Positive slack - size down
      - Negative slack - size up

- Make gates that toggle more frequently smaller
Better to postpone the introduction of signals with a high transition rate (signals with signal probability close to 0.5)
Control circuits

- State encoding has a **big impact** on the power efficiency
- **Energy driven** -> try to minimize number of bit transitions in the state register
  - Fewer transitions in state register
  - Fewer transitions propagated to combinational logic
Bus encoding

- Reduces number of bit toggles on the bus
- Different flavors
  - Bus-invert coding
    - Uses an extra bus line $invert$:
      - if the number of transitions is $< K/2$, $invert = 0$ and the symbol is transmitted as is
      - if the number of transitions is $> K/2$, $invert = 1$ and the symbol is transmitted in a complemented form
  - Low-weight coding
    - Uses $transition$ signaling instead of $level$ signaling
Bus invert coding

Under uniform random signal conditions (non-correlated data), 25% upper bound on toggle reduction

Source: M.Stan et al., 1994
Summary

- Power Dissipation is already a prime design constraint.

- Low-power design requires operation at lowest possible voltage and clock speed.

- Low-power design requires optimization at all levels of abstraction.
Announcements

- Project M1:
  - Check off in lab session
  - Report by Friday

- Exam Review Session:
  - Monday Oct 13, 4:30-6:30pm
  - PH 125C