Admin

- Project proposals due by Fr 5pm (by e-mail to Huifang and myself)
  - Title
  - Short abstract of 10-15 lines describing the problem you are trying to address
- Special office hours today right after class (3:30-4:30pm)
- Some feedback on ISSCC? What did catch your eye?
Today’s lecture

- Using the models we have created so far to do create an environment for optimization

Reading:
- ICCAD paper by Stojanovic et al.
- Chapters 2 and 3 in the text by K. Bernstein (High Speed CMOS Design Styles)
- Background material from Rabaey, 2nd ed, Chapters 5, 6.

Static Timing Analysis

- Computing critical (longest) path delay
  - Longest path algorithm on DAG [Kirkpatrick, IBM Jo. R&D, 1966]

- Used in most ASIC designs today

- Limitations
  - False paths
  - Simultaneous arrival times
Signal Arrival Times

- NAND gate:

\[ V_{DC} \]

\[ \text{Out} \]

1

Signal Arrival Times

- NAND gate:

\[ V_{DC} \]

\[ \text{Out} \]

1
Simultaneous Arrival Times

NAND gate:

Impact of Arrival Times

Up to 25%
Optimization for Performance

- Performance critical blocks
- Start with a synthesized design
  - Easier to explore architectures
  - Easy to verify
  - Provides some level of performance optimization
- Understand the limits of synthesized designs

Performance Optimization

Increasing the performance increases power!
How to Increase Performance?

- Scale technology
- Circuit level:
  - Transistor sizing, buffering
  - Wire optimization, repeaters
  - Supply and Threshold voltage
  - Logic styles
  - Timing, latches
- Microarchitecture
  - Block topologies (adders, multipliers)
  - Pipelining
  - Parallelism

Sizing Logic Paths for Speed

- Frequently, input capacitance of a logic path is constrained
- Logic has to drive some capacitance
- Example: ALU load in an Intel’s microprocessor is > 0.5pF
- How do we size the ALU datapath to achieve maximum speed?
- Review the method of logical effort
Inverter Chain

If $C_L$ and $C_{in}$ are given:
- How many stages are needed to minimize the delay?
- How to size the inverters?

May need some additional constraints.

Delay Formula

Delay $\sim R_W (C_{int} + C_L)$

$$t_p = kR_W C_{int} \left(1 + \frac{C_L}{C_{int}}\right) = t_{p0} \left(1 + f / \gamma\right)$$

$C_{int} = \gamma C_{gin}$ with $\gamma = 1$

$f = C_L/C_{gin}$ - effective fanout

$R = \frac{R_{unit}}{W} ; C_{int} = WC_{unit}$

$t_{p0} = 0.7 R_{unit} C_{unit}$
Apply to Inverter Chain

\[ t_p = t_{p1} + t_{p2} + \ldots + t_{pN} \]

\[ t_{pj} \sim R_{\text{unit}} C_{\text{unit}} \left( 1 + \frac{C_{\text{gin},j+1}}{\gamma C_{\text{gin},j}} \right) \]

\[ t_p = \sum_{j=1}^{N} t_{p,j} = t_{p0} \sum_{j=1}^{N} \left( 1 + \frac{C_{\text{gin},j+1}}{\gamma C_{\text{gin},j}} \right), \quad C_{\text{gin},N+1} = C_L \]
Optimal Tapering for Given $N$

Delay equation has $N - 1$ unknowns, $C_{gin,2} - C_{gin,N}$

Minimize the delay, find $N - 1$ partial derivatives

Result: $C_{gin,j+1}/C_{gin,j} = C_{gin,j}/C_{gin,j-1}$

Size of each stage is the geometric mean of two neighbors

$$C_{gin,j} = \sqrt{C_{gin,j-1}C_{gin,j+1}}$$

- each stage has the same effective fanout ($C_{out}/C_{in}$)
- each stage has the same delay

Optimum Delay and Number of Stages

When each stage is sized by $f$ and has same effective fanout $f$:

$$f^N = F = C_L / C_{gin,1}$$

Effective fanout of each stage:

$$f = \sqrt[N]{F}$$

Minimum path delay

$$t_p = N t_{p0} \left(1 + \sqrt[N]{F} / \gamma \right)$$
Example

![Diagram of a circuit with CL = 8 C1 and C1/C1 evenly distributed across N = 3 stages.]

\[ f = \sqrt[3]{8} = 2 \]

Optimum Number of Stages

For a given load, \( C_L \) and given input capacitance \( C_{in} \), find optimal number of stages, \( N \), and optimal sizing, \( f \).

\[ C_L = F \cdot C_{in} = f^N C_{in} \quad \text{with} \quad N = \frac{\ln F}{\ln f} \]

\[ t_p = N t_{p0} \left( F^{1/N} \frac{1}{\gamma} + 1 \right) = \frac{t_{p0} \ln F}{\gamma} \left( \frac{f}{\ln f} + \frac{\gamma}{\ln f} \right) \]

\[ \frac{\partial t_p}{\partial f} = \frac{t_{p0} \ln F}{\gamma} \left( \frac{\ln f - 1 - \gamma/f}{\ln^2 f} \right) = 0 \]

For \( \gamma = 0 \), \( f = e \), \( N = \ln F \)

\[ f = e^{1+\gamma/f} \]
Optimum Effective Fanout $f$

Optimum $f$ for given process defined by $\gamma$

$$f = e^{(1+\gamma/f)}$$

$f_{opt} = 3.6$

for $\gamma=1$

---

Impact of Loading on $t_p$

With self-loading $\gamma=1$

---
Extending the Model

For given $N$: $C_{i+1}/C_i = C_i/C_{i-1}$
To find $N$: $C_{i+1}/C_i \sim 4$

Method of logical effort generalizes this to any logic path

$$\text{Delay} = \sum_{i=1}^{N} (p_i + g_i \cdot f_i) \quad \text{(in units of } \tau_{\text{inv}})$$

Logical Effort

$$\text{Delay} = k \cdot R_{\text{unit}} \cdot C_{\text{unit}} \left(1 + \frac{C_L}{\gamma C_{\text{in}}} \right)$$
$$= \tau (p + g \cdot f)$$

$p$ – intrinsic delay - gate parameter $\neq f(W)$
$g$ – logical effort – gate parameter $\neq f(W)$
$f$ – electrical effort (fanout)

Normalize everything to an inverter:
$g_{\text{inv}} = 1, R_{\text{inv}} = \text{unit}$

Divide everything by $\tau_{\text{inv}}$
(everything is measured in unit delays $\tau_{\text{inv}}$)
Assume $\gamma = 1$. 
Delay in a Logic Gate

Gate delay:
\[ d = h + p \]
- effort delay
- intrinsic delay

Effort delay:
- logical effort
- effective fanout

\[ \text{effective fanout} = \frac{C_{out}}{C_{in}} \]

Logical effort is a function of topology, independent of sizing
Effective fanout (electrical effort) is a function of load/gate size

Logical Effort

- Inverter has the smallest logical effort and intrinsic delay of all static CMOS gates
- Logical effort of a gate presents the ratio of its input capacitance to the inverter capacitance when sized to deliver the same current
- Logical effort increases with the gate complexity
Logical Effort

Logical effort is the ratio of input capacitance of a gate to the input capacitance of an inverter with the same output current.

\[ g = 1 \]

Size factor: 1.5

Size factor: 1.8

Logical Effort of Gates

Fan-out \( f \)

Normalized delay \( d \)

\( F(\text{Fan-in}) \)

1 2 3 4 5 6 7

\[ t_{\text{pNAND}} \]

\[ t_{\text{pINV}} \]

\[ g = \]

\[ p = \]

\[ d = \]

\[ g = \]

\[ p = \]

\[ d = \]

30
**Logical Effort of Gates**

\[
g = \frac{3.5}{3} \\
p = \frac{5.5}{3} \\
d = (\frac{3.5}{3})f + 1.8
\]

**Add Branching Effort**

Branching effort:

\[
b = \frac{C_{\text{on-path}} - C_{\text{off-path}}}{C_{\text{on-path}}}
\]
Multistage Networks

\[ Delay = \sum_{i=1}^{N} (p_i + g_i \cdot f_i) \]

Stage effort: \( h_i = g_i f_i \)
Path electrical effort: \( F = \frac{C_{out}}{C_{in}} \)
Path logical effort: \( G = g_1 g_2 \cdots g_N \)
Branching effort: \( B = b_1 b_2 \cdots b_N \)
Path effort: \( H = GFB \)
Path delay \( D = \Sigma d_i = \Sigma p_i + \Sigma h_i \)

Optimum Effort per Stage

When each stage bears the same effort:

\[ h^N = H \]
\[ h = \frac{N}{\sqrt[N]{H}} \]

Stage efforts: \( g_1 f_1 = g_2 f_2 = \cdots = g_N f_N \)
Effective fanout of each stage: \( f_i = h / g_i \)
Minimum path delay

\[ \hat{D} = \sum (g_i f_i + p_i) = NH^{1/N} + P \]
Optimal Number of Stages

For a given load, and given input capacitance of the first gate, find optimal number of stages and optimal sizing.

\[ D = NH^{1/N} + P \]

\[ \frac{\partial D}{\partial N} = -H^{1/N} \ln(H^{1/N}) + H^{1/N} + P = 0 \]

Substitute 'best stage effort' \( h = H^{1/N} \)

Logical Effort Optimization Methodology

› For smaller problems, easy to translate into set of analytical expressions
› Feed them into Matlab optimizer
› With some careful manipulations, can be turned into a convex optimization problem (Stojanovic)
› Easily extended to add power/energy
Optimization for Performance

Options

- Technology choice
  CMOS, bipolar, BiCMOS, GaAs, Superconducting
- Logic level optimizations
  logic depth, network topology, fan-out, gate complexity
- Circuit optimizations
  logic style, transistor sizing
- Physical optimization
  implementation choice, layout strategy
- Wires are the key

Logic Level Optimizations

Logic Depth

Techniques: Restructuring, pipelining, retiming, technology mapping

Well covered by today’s logic and sequential synthesis
Logic Optimizations (2)

Fanout

$T_p = O(FO)$
also effects wiring capacitance

Technique: Removal of common sub-expression
Start from tree structure/output

Logic Optimizations (3)

Fanin

$T_p = O(FI^2)$

Observation: only true if FI translates in series devices - otherwise linear
e.g. NAND pull-down
NOR pull-up

AVOID LARGE FAN-IN GATES! (Typically not more than FI < 4)
Logic Optimizations (4)

Slope is a function of “driving strength”

All the gates have the same drive current

Technology Mapping for Performance

Use low FI modules on critical path(s)
Library composition?
CMOS Logic Styles

CMOS tradeoffs:
- Speed
- Power (energy)
- Area

Design tradeoffs
- Robustness, scalability
- Design time

Many styles: don’t try to remember the names – remember the principles

Changing the logic style – can it be done without breaking the synthesis flow?

---

CMOS Logic Styles

- Complementary
  - robust
  - scales
  - large and slow

- Pass Transistor Logic
  - simple and fast
  - not always very efficient
  - versatile
CMOS Logic Styles

Ratioed Logic

Dynamic Logic

small & fast static power

Small & fastest!
Noise issues Scales?

Pulsed Static CMOS

RH – Reset high
RL – Reset low

Fast pull-up
Fast pull-down

PS-CMOS

Evaluation and reset waves: reset is 1.5x slower

Advantages:
- No dynamic nodes – good noise immunity
- Reset delay slower than evaluation
- No data dependent delay (worst case gets better)
- No false transitions

Disadvantages
- Width of reset wave limits logic depth
- Margin in design
Skewing Gates

- Different rising and falling delays

\[ LE = \]

\[ W \]

\[ W \]

Skewing Gates

\[ LE = \]

\[ 4W \]

\[ W \]
Skewing Gates

Circuit 1

Input A

Input B

NAND

Input C

GND

Circuit 2

Input A

Input B

NAND

Input C

GND

NOR Q

VDD

N1

N2

N3

VDD

GND

GND

GND