CS152
Computer Architecture and Engineering
Lecture 5

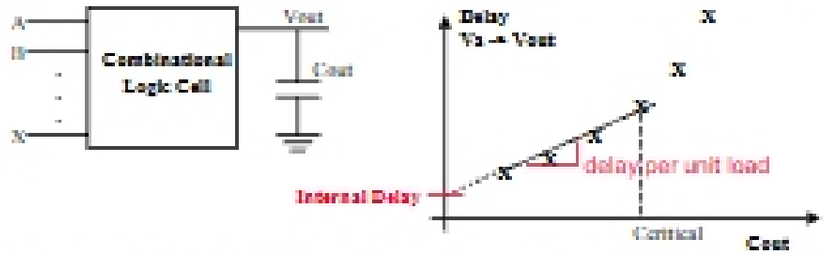
The Design Process

February 10, 2003

John Kubiatowicz (www.cs.berkeley.edu/~kubitron)

lecture slides: http://www-inst.eecs.berkeley.edu/~cs152/

Review: recall General C/L Cell Delay Model



° Combinational Cell (symbol) is fully specified by:
- functional (input -> output) behavior
  - truth-table, logic equation, VHDL
- load factor of each input
- critical propagation delay from each input to each output for each transition
  - $T_{IL}(A, o)$ = Fixed Internal Delay + Load-dependent-delay x *load*

° Linear model composes

Review: Storage Element's Timing Model

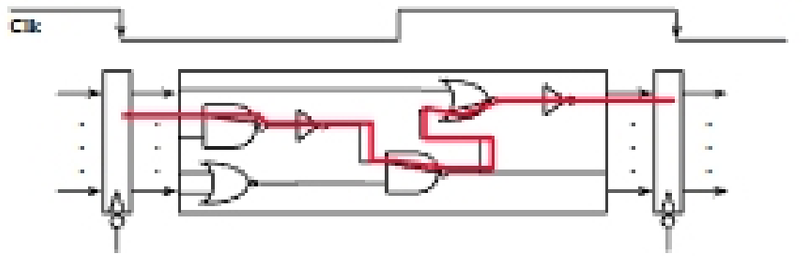
° Setup Time: Input must be stable BEFORE trigger clock edge

° Hold Time: Input must REMAIN stable after trigger clock edge

° Clock-to-Q time:
- Output cannot change instantaneously at the trigger clock edge
- Similar to delay in logic gates, two components:
  - Internal Clock-to-Q
  - Load dependent Clock-to-Q

° Typical for class: 1ns Setup, 0.5ns Hold

Review: Critical Path & Cycle Time



° Critical path: the slowest path between any two storage devices

° Cycle time is a function of the critical path

° must be greater than:

Clock-to-Q + Longest Path through Combination Logic + Setup

Review: Tricks to Reduce Cycle Time

° Reduce the number of gate levels

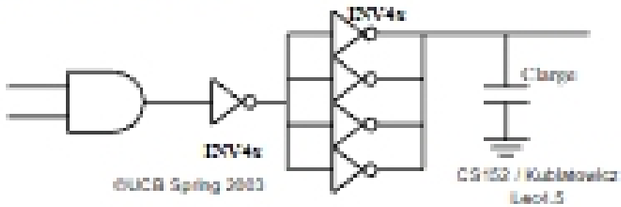
° Use ecoteric/dynamic timing methods

° Pay attention to loading

  ° One gate driving many gates is a bad idea

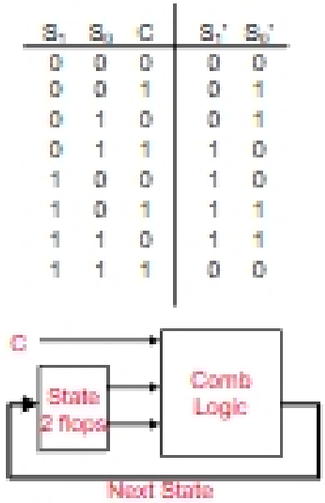
  ° Avoid using a small gate to drive a long wire

° Use multiple stages to drive large load

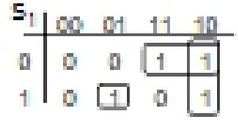


Review: Karnaugh Map for easier simplification

S_1	S_0	C	S_1'	S_0'
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	1	0
1	0	1	1	1
1	1	0	1	1
1	1	1	0	0



$$S_0' = (\overline{S_0} \cdot C) + (S_0 \cdot \overline{C})$$



$$S_1' = (\overline{S_1} \cdot S_0 \cdot C) + (S_1 \cdot \overline{C}) + (S_1 \cdot \overline{S_0})$$

Review: DeMorgan's theorem: Push Bubbles and Morph

NAND Gate

A	B	Out
0	0	1
0	1	1
1	0	1
1	1	0

NOR Gate

A	B	Out
0	0	1
0	1	0
1	0	0
1	1	0

$\text{Out} = \overline{A \cdot B} = \overline{A} + \overline{B}$

$\text{Out} = \overline{A + B} = \overline{A} \cdot \overline{B}$

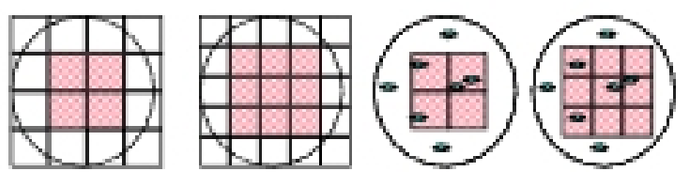
A	B	$\overline{A}$	$\overline{B}$	Out
0	0	1	1	1
0	1	1	0	1
1	0	0	1	1
1	1	0	0	0

A	B	$\overline{A}$	$\overline{B}$	Out
0	0	1	1	1
0	1	1	0	0
1	0	0	1	0
1	1	0	0	0

Review: Integrated Circuit Costs

$$\text{Die cost} = \frac{\text{Wafer cost}}{\text{Dies per Wafer} * \text{Die yield}}$$

$$\text{Dies per wafer} = \frac{\pi * (\text{Wafer diam}/2)^2}{\text{Die Area}} - \frac{\pi * \text{Wafer diam}}{\sqrt{2} * \text{Die Area}} - \text{Test dies} \approx \frac{\text{Wafer Area}}{\text{Die Area}}$$



$$\text{Die Yield} = \frac{\text{Wafer yield}}{\left\{1 + \frac{\text{Defects per unit area} * \text{Die Area}}{\alpha}\right\}^{\alpha}}$$

Die Cost is goes roughly with (die area)³ or (die area)⁴

The Design Process

"To Design is To Represent"

Design activity yields description/representation of an object

- Traditional craftsman does not distinguish between the conceptualization and the artifact
- Separation comes about because of *complexity*
- The concept is captured in one or more *representation languages*
- This process is design

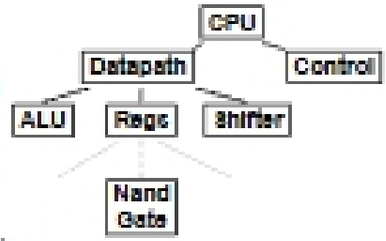
Design Begins With Requirements

- Functional Capabilities: what it will do
- Performance Characteristics: Speed, Power, Area, Cost, . . .

Design Process (cont.)

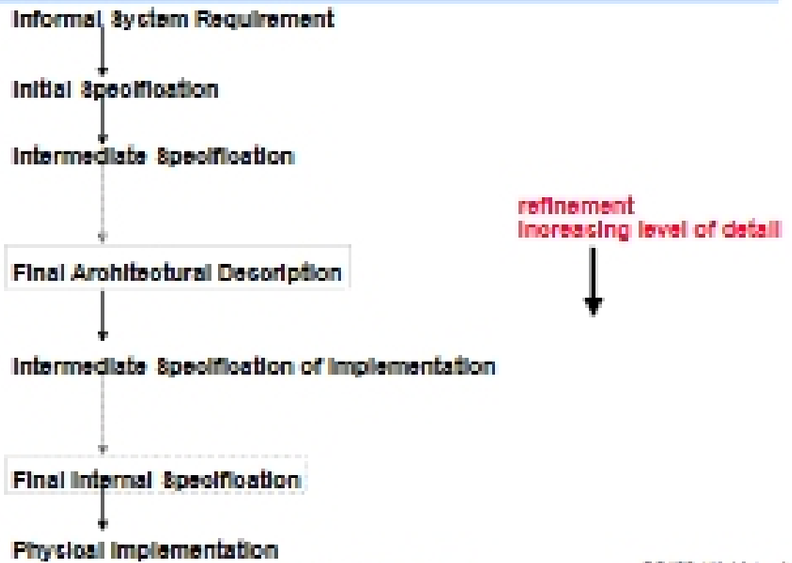
Design Finishes As Assembly

- Design understood in terms of components and how they have been assembled
- Top Down *decomposition* of complex functions (behaviors) into more primitive functions
- bottom-up *composition* of primitive building blocks into more complex assemblies

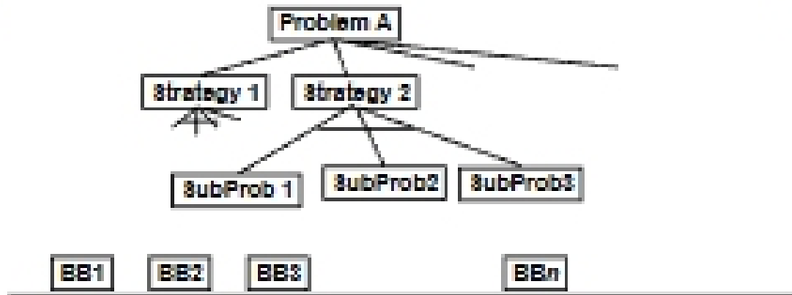


Design is a "creative process," not a simple method

Design Refinement



Design as Search



Design involves educated guesses and verification

- Given the goals, how should these be prioritized?
- Given alternative design pieces, which should be selected?
- Given design space of components & assemblies, which part will yield the best solution?

Feasible (good) choices vs. Optimal choices