EE141-Spring 2010 Digital Integrated Circuits

Lecture 22
Energy

Administrativia

- Midterm 2 over – grades by next Wednesday
- DO NOT FORGET THE PROJECT! Due on Wednesday April 14
  - Extra office hours of TAs on Mo and Tu (during lab hours) in 353 Cory
- Project Phase 1 has been graded

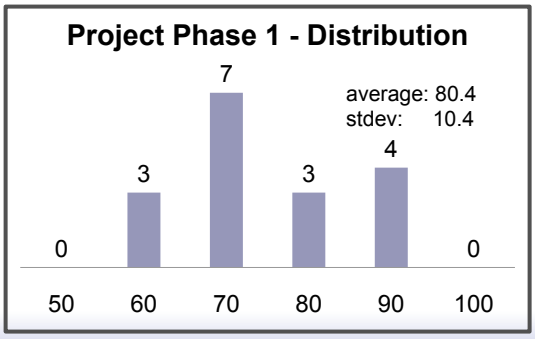
Class Material

- Last lecture
  - Technology Scaling
  - Midterm
- Today's lecture
  - Adders
- Reading (Ch 5, Ch 11)

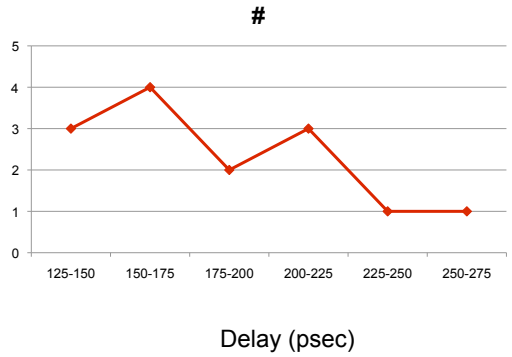
Project Phase-1 Grades

- Grading Policy
  - Approach and Correctness (40%)
  - Results (30%)
  - Creativity (10%)
  - Report (20%)

You will receive individual feedback.

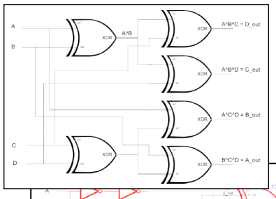


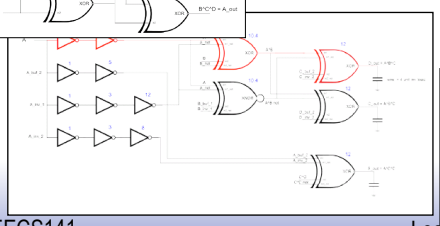
Delay

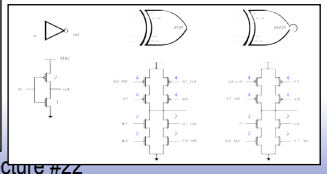


Example of Good Phase-1 Report

- Starting from Optimal # of Stages
- Correct Hand Calculations
- Optimized Logic Network







C_{load} = bottom plate + fringing + 4 * min sized invertor
= 0.25 mm * 0.14 um * 19.9 aF/ um^2 + 2 * 0.25 mm * 25.6 aF/um + 4 * 0.582 fF
= 15.82 fF

The total path fanout with our design is (C_{load}/C_{in}), which would be (15.82fF/0.582fF) = 27.2, with a single unit-sized invertor starting the path. The path branch effort is 4 -- 2 for each path that branches to two xor gates. The path logical effort is 4 because each XOR has a logical effort of 2. Therefore, our total path effort is fanout*(branch effort)*(path logical effort) = 434.538.

The optimal number of stages to have a fanout of 3.6 is with log(total path effort)/log(3.6), which is approximately 4.7. Since we can only add an even number of inverters, we choose to have 4 stages. With four stages, the stage effort becomes (434.538)^(1/4) = 4.56569. This implies that a path with 4 stages will have a larger, less optimal fanout, than a path with 5 stages.

We will call the sizes of the path 1, x, y, and z for each of the gates from beginning to end on input A_buf_1.

$$f = \frac{15.82\,fF}{z \times C_{g,inv}} = \frac{EF}{LE}$$

$$z = \frac{15.82\,fF \times LE}{EF \times C_{g,inv}} = \frac{15.82\,fF \times 2}{4.56 \times 0.582\,fF} = 11.8968 \approx 12$$

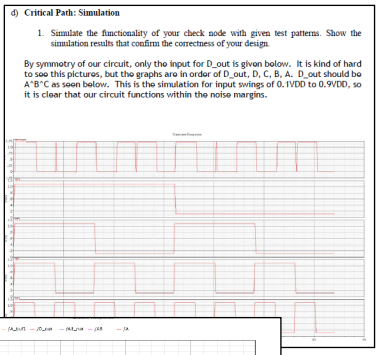
$$y = \frac{z \times LE_z \times b_z}{EF} = \frac{12 \times 2 \times 2}{4.56} = 10.42 \approx 10.4$$

$$x = 1 \times 4.56 = 4.56 \approx 4.5$$

Thus, we got sizes 1, 4.5, 10.4, and 12 for the critical path. The sizes gates on the other paths are sized with these sizes

Example of Good Phase-1 Report

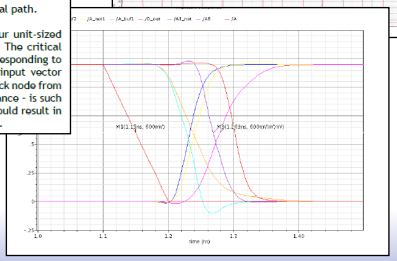
- Worst Case Input Pattern Analysis
- Noise Margin Simulation
- Well-Organized Report



d) **Critical Path: Simulation**

1. Simulate the functionality of your check node with given test patterns. Show the simulation results that confirm the correctness of your design.

By symmetry of our circuit, only the input for D_out is given below. It is kind of hard to see this pictures, but the graphs are in order of D_out, D, C, B, A. D_out should be A*B*C as seen below. This is the simulation for input swings of 0.1VDD to 0.9VDD, so it is clear that our circuit functions within the noise margins.



c) **Critical Path: Analytical**

1. Identify the critical path(s) and explain the reason.

Show schematics of the check node with highlighted critical path. Show the hand calculation of how you size the gates for minimum delay. Annotate the sizes of all gates along the critical path.

Refer to schematic above with proper sizes and highlighted critical path.

Maximum input capacitance = $4C_{min\_inv}$, which we meet with four unit-sized inverters for each branch of the inputs as in the picture above. The critical path is the one with the two inverters followed by two XORs, corresponding to input chain A_buf_1 above. This is for the output D_out, and input vector (A,B,C) = (1->0, 1, 1) and (0 -> 1, 1, 1). The total fanout of the check node from input to output -- as specified by the max input and load capacitance - is such that an even number of buffer stages preceding the XOR gates would result in the fanout-per-stage being suboptimal. This calculation is below.

Energy/Power Revisited

Transition Activity and Power

- Energy consumed in N cycles, E_N:

$$E_N = C_L \bullet V_{DD}^2 \bullet n_{0\to1}$$

$n_{0\to1}$ – number of 0→1 transitions in N cycles

$$P_{avg} = \lim_{N\to\infty} \frac{E_N}{N} \cdot f = \left(\lim_{N\to\infty} \frac{n_{0\to1}}{N} \right) \cdot C_L \cdot V_{DD}^2 \cdot f$$

$$\alpha_{0\to1} = \lim_{N\to\infty} \frac{n_{0\to1}}{N} \cdot f$$

$$\boxed{P_{avg} = \alpha_{0\to1} \cdot C_L \cdot V_{DD}^2 \cdot f}$$

Factors Affecting Transition Activity

- "Static" component (does not account for timing)
  - ⇨ Type of Logic Function (NOR vs. XOR)
  - ⇨ Type of Logic Style (Static vs. Dynamic)
  - ⇨ Signal Statistics
  - ⇨ Inter-signal Correlations
- "Dynamic" or timing dependent component
  - ⇨ Circuit Topology
  - ⇨ Signal Statistics and Correlations

Type of Logic Function: NOR

Example: Static 2-input NOR Gate

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

Assume signal probabilities

$p_{A=1} = 1/2$

$p_{B=1} = 1/2$

Then transition probability

$p_{0\to1} = p_{Out=0} \times p_{Out=1}$

$= 3/4 \times 1/4 = 3/16$

If inputs switch every cycle

$\alpha_{0\to1} = 3/16$

Type of Logic Function: NAND

Example: Static 2-input NAND Gate

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

Assume signal probabilities

$p_{A=1} = 1/2$

$p_{B=1} = 1/2$

Then transition probability

$p_{0\to1} = p_{Out=0} \times p_{Out=1}$

$= 3/4 \times 1/4 = 3/16$

If inputs switch every cycle

$\alpha_{0\to1} = 3/16$

Type of Logic Function: XOR

Example: Static 2-input XOR Gate

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

Assume signal probabilities

$p_{A=1} = 1/2$

$p_{B=1} = 1/2$

Then transition probability

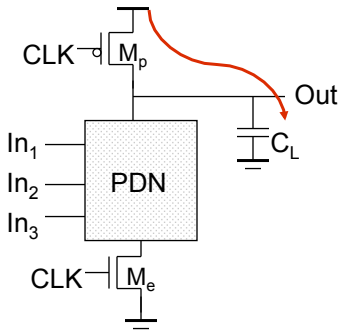
$p_{0\to1} = p_{Out=0} \times p_{Out=1}$

=

If inputs switch in every cycle

$\alpha_{0\to1}$ =

Power Consumption of Dynamic Gates



Power only dissipated when previous Out = 0

Dynamic Power Consumption is Data Dependent

Dynamic 2-input NOR Gate

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

Assume signal probabilities

$P_{A=1} = 1/2$

$P_{B=1} = 1/2$

Then transition probability

$P_{0\rightarrow1} = P_{out=0} \times P_{out=1}$

$= 3/4 \times 1 = 3/4$

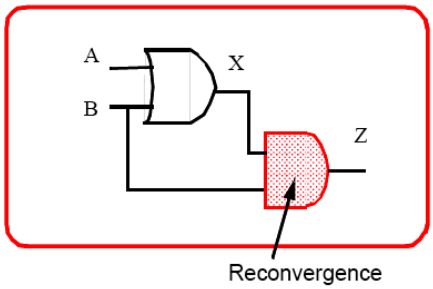
Switching activity always higher in dynamic gates!

$P_{0\rightarrow1} = P_{out=0}$

Clock

- Always switches
- Often consumes 25-50% of total power
- Clock gating commonly employed

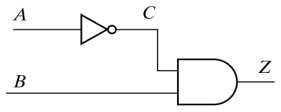
Problem: Reconvergent Fanout



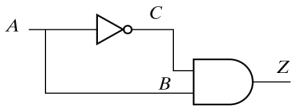
$P(Z = 1) = P(B = 1) \cdot P(X = 1 \mid B=1)$

Becomes complex and intractable fast

Inter-Signal Correlations



(a) Logic circuit without reconvergent fan-out



(b) Logic circuit with reconvergent fan-out

Logic without reconvergent fanout

$p_{0\to1} = (1 - p_{\bar{A}}p_B)\, p_{\bar{A}}p_B$

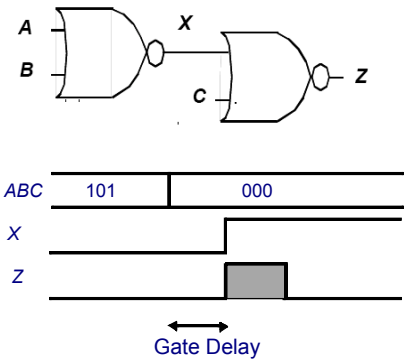
Logic with reconvergent fanout

$P(Z = 1) = p(C=1 \mid B=1)\ p(B=1)$

$p_{0\to1} = 0$

- Need to use conditional probabilities to model inter-signal correlations
- CAD tools best for performing such analysis

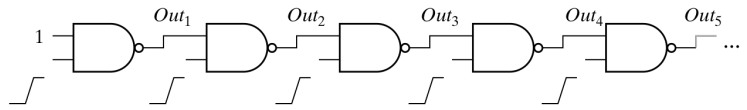
Glitching in Static CMOS

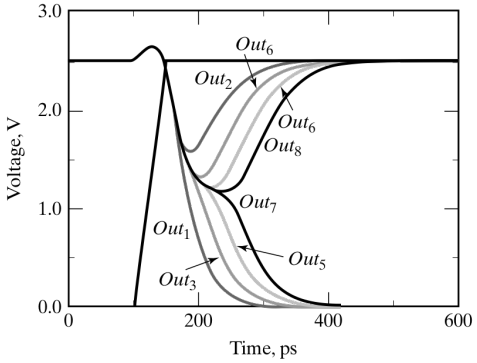


Also known as dynamic hazards

The result is correct,
but there is extra power dissipated

Example: Chain of NAND Gates

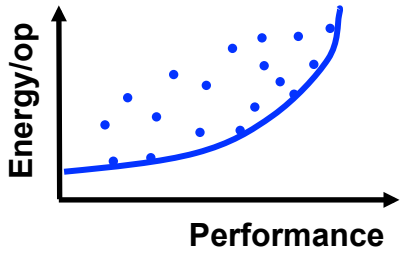




Principles for Power Reduction

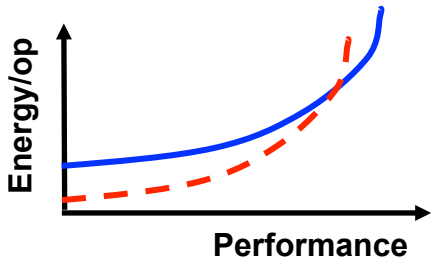
- Most important idea: reduce waste
- Examples:
  - Don't switch capacitors you don't need to
    - Clock gating, glitch elimination, logic re-structuring
  - Don't run circuits faster than needed
    - Power α V_{DD}^2 – can save a lot by reducing supply for circuits that don't need to be as fast
    - Parallelism falls into this category
- Let's say we do a good job of that – then what?

Energy – Performance Space



- Plot all possible designs on a 2-D plane
  - No matter what you do, can never get below/to the right of the solid line
- This line is called "Pareto Optimal Curve"
  - Usually (always) follows law of diminishing returns

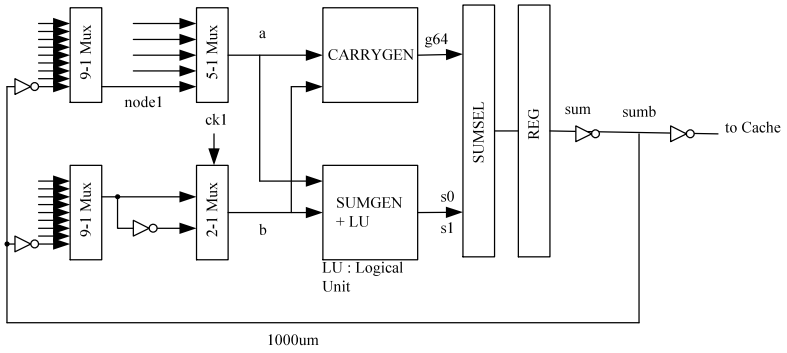
Optimization Perspective



- Instead of metrics like EDP, this curve often provides information more directly
  - Ex1: What is minimum energy for XX performance?
  - Ex2: Over what range of performance is a new technique (dotted line) actually beneficial?

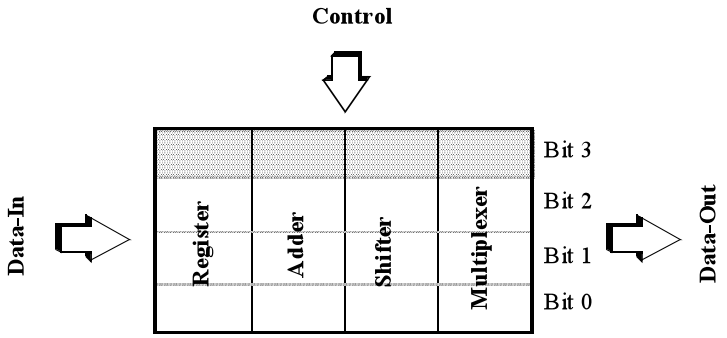
Adders

An Intel Microprocessor



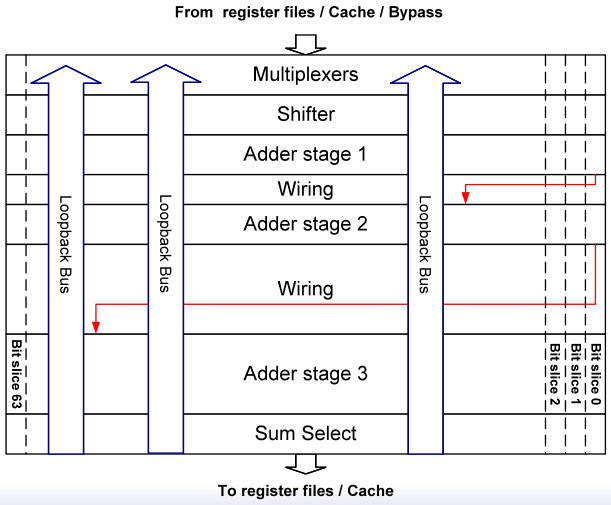
Itanium has 6 64-bit integer execution units like this

Bit-Sliced Design

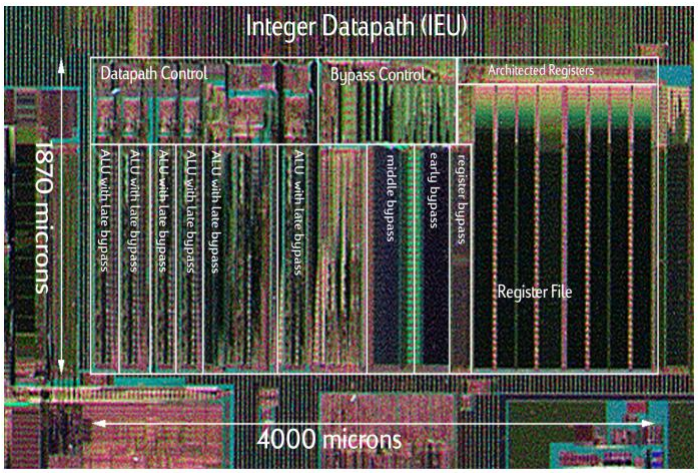


Tile identical processing elements

Bit-Sliced Datapath

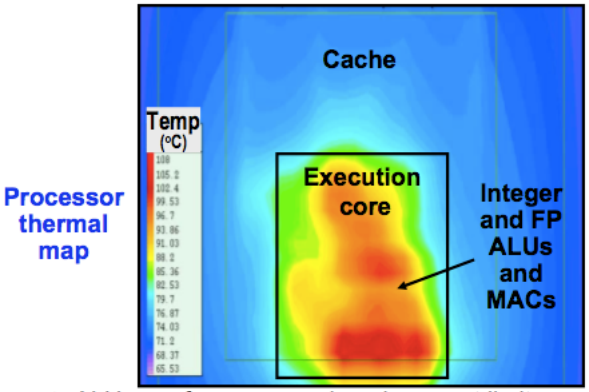


Itanium Integer Datapath



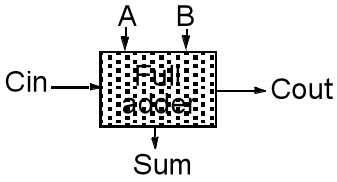
 Fetzer, Orton, ISSCC'02

Data Paths Are Thermal Hogs



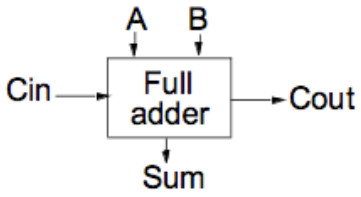
- ALUs: performance and peak-current limiters
- Goal: high-performance **energy-efficient** design

Full-Adder



A	B	C_i	S	C_o	*Carry status*	
0	0	0	0	0	delete	(kill)
0	0	1	1	0	delete	(kill)
0	1	0	1	0	propagate	
0	1	1	0	1	propagate	
1	0	0	1	0	propagate	
1	0	1	0	1	propagate	
1	1	0	0	1	generate	
1	1	1	1	1	generate	

The Binary Adder



$$S = A \oplus B \oplus C_i$$
$$= A\bar{B}\bar{C}_i + \bar{A}B\bar{C}_i + \bar{A}\bar{B}C_i + ABC_i$$
$$C_o = AB + BC_i + AC_i$$

Express Sum and Carry as a function of P, G, K

Define 3 new variable which ONLY depend on A, B

Generate (G) = AB

Propagate (P) = A ⊕ B

Kill = A B

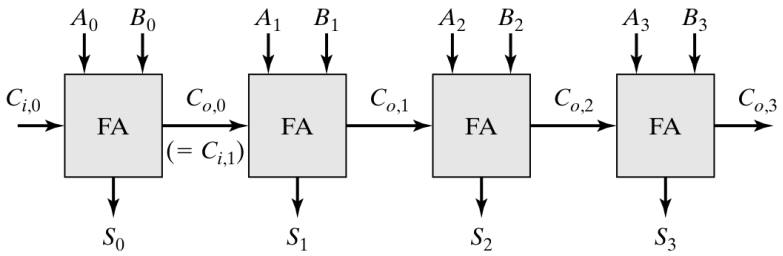
$$C_o(G,P) = G + PC_i$$
$$S(G,P) = P \oplus C_i$$

Can also derive expressions for *S* and C_o based on *K* and *P*

Note that we will sometimes use an alternate definition for

Propagate (P) = A + B

Simplest Adder: Ripple-Carry



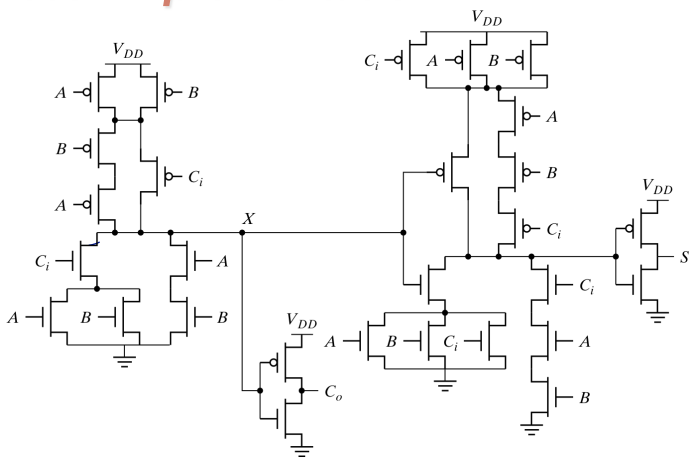
Worst case delay linear with the number of bits

$t_d = O(N)$

$t_{adder} = (N\text{-}1)t_{carry} + t_{sum}$

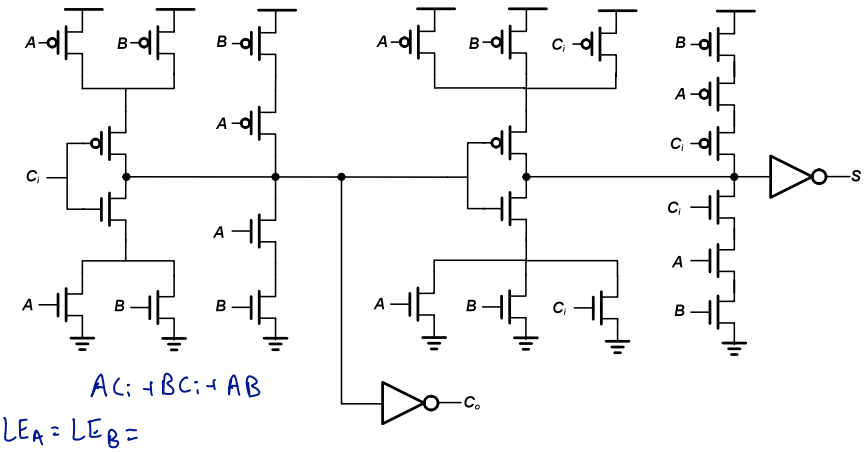
Goal: Make the fastest possible carry path circuit

Complementary Static CMOS Full Adder: "Direct" Implementation



28 Transistors

Complementary Static CMOS Full Adder



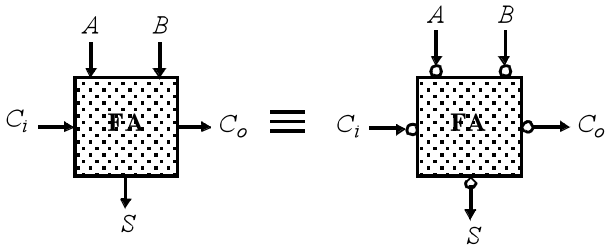
$AC_i + BC_i + AB$

$LE_A = LE_B =$

$LE_{C_i} =$

28 Transmistors

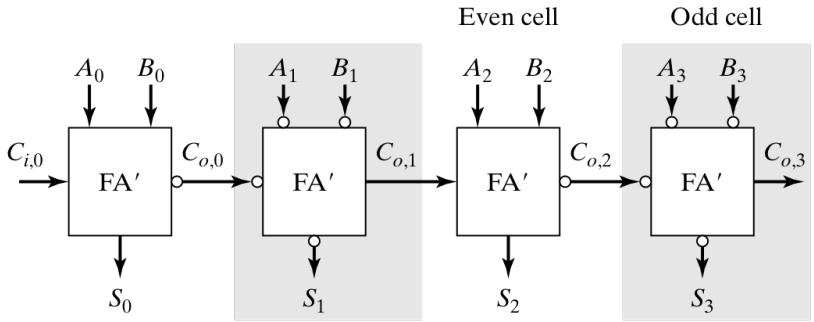
Inversion Property



$$\bar{S}(A,B,C_i) = S(\bar{A},\bar{B},\bar{C}_i)$$

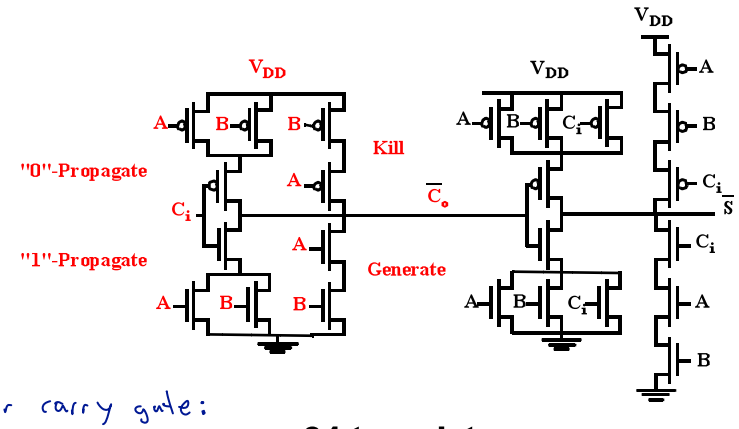
$$\overline{C_o}(A,B,C_i) = C_o(\bar{A},\bar{B},\bar{C}_i)$$

Minimize Critical Path by Reducing Inverting Stages



Exploit Inversion Property

A Better Structure: The Mirror Adder



For carry gate:

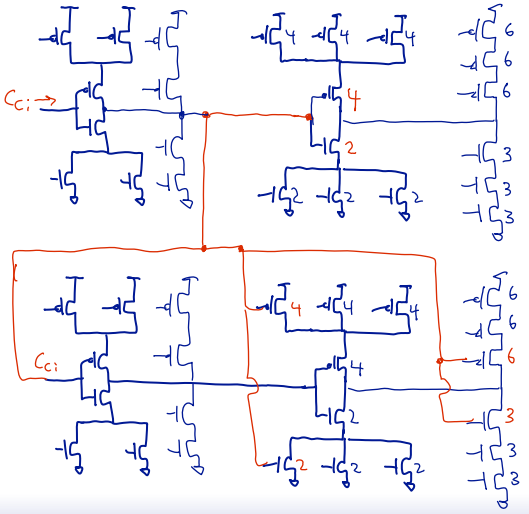
$LE_{C_i} = 2$

24 transistors

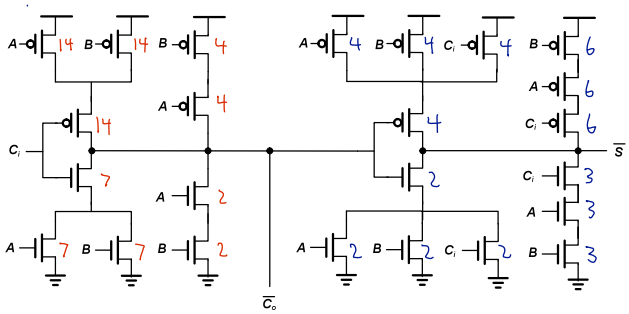
Sizing the Mirror Adder: Fanout

- Since LE of carry gate is 2, want *f* of 2 to get EF of 4
- Use min. size sum gates to reduce load on carry.
- Total load on carry gate is:

$C_{load} = C_{Ci} + (6+6+9)$
$C_{load} = 2C_{Ci}$



Sizing the Mirror Adder



- $C_{load} = C_{Ci} + (6+6+9) = 2C_{Ci}$

  → $C_{Ci} = 21$

- Minimum size G and K stacks to reduce diffusion loading

Mirror Adder Summary

- The NMOS and PMOS chains are completely symmetrical. Maximum of two series transistors in the carry-generation gate.
- When laying out the cell, the most critical issue is the minimization of the capacitance at node C_o. Reduction of the diffusion capacitances is particularly important.
- Carry signals are critical - transistors connected to C_i are placed closest to the output.
- Only the transistors in the (propagate) carry chain have to be optimized for speed. All transistors in the sum stage can be minimal size.