7. Combinational Circuits

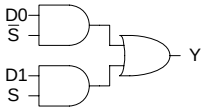
- Last module:
  - Delay in logic networks
  - Logical Effort
  - Choosing the best number of stages
- This module:
  - Designing CMOS gate networks
  - Speeding up combinational gates

Example 1

```
module mux(input  s, d0, d1,
           output y);

   assign y = s ? d1 : d0;
endmodule
```

1) Sketch a design using AND, OR, and NOT gates.

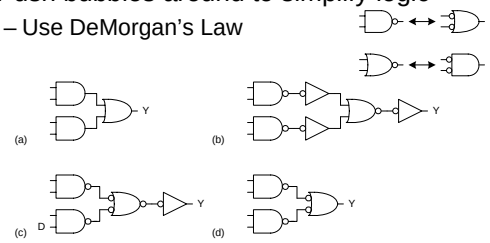


Example 2

2) Sketch a design using NAND, NOR, and NOT gates. Assume ~S is available.

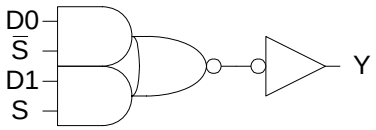
Bubble Pushing

- Start with network of AND / OR gates
- Convert to NAND / NOR + inverters
- Push bubbles around to simplify logic
  - Use DeMorgan's Law



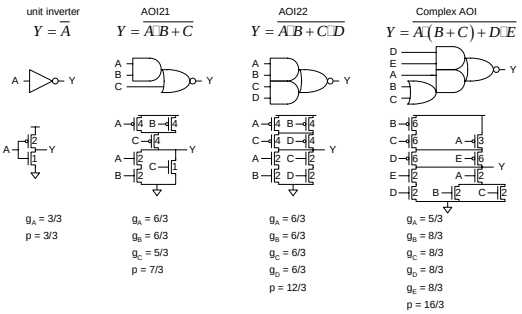
Example 3

3) Sketch a design using one compound gate and one NOT gate. Assume ~S is available.



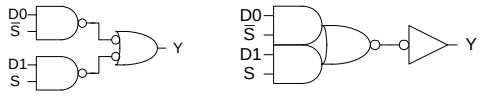
Compound Gates

- Logical Effort of compound gates



Example 4

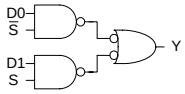
- The multiplexer has a maximum input capacitance of 16 units on each input. It must drive a load of 160 units. Estimate the delay of the NAND and compound gate designs.



H = 160 / 16 = 10
B = 1
N = 2

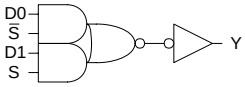
NAND Solution

$P = 2 + 2 = 4$
$G = (4/3)(4/3) = 16/9$
$F = GBH = 160/9$
$\hat{f} = \sqrt[N]{F} = 4.2$
$D = N\hat{f} + P = 12.4\tau$



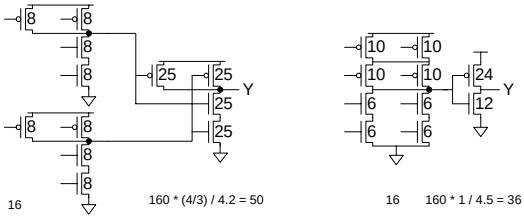
Compound Solution

$P = 4 + 1 = 5$
$G = (6/3)(1) = 2$
$F = GBH = 20$
$\hat{f} = \sqrt[N]{F} = 4.5$
$D = N\hat{f} + P = 14\tau$



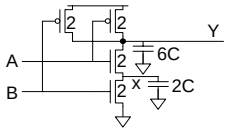
Example 5

- Annotate your designs with transistor sizes that achieve this delay.



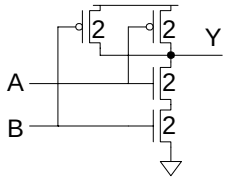
Input Order

- Our parasitic delay model was too simple
  - Calculate parasitic delay for Y falling
    - If A arrives latest? 2τ
    - If B arrives latest? 2.33τ



Inner & Outer Inputs

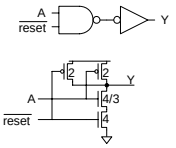
- *Inner* input is closest to output (A)
- *Outer* input is closest to rail (B)



- If input arrival time is known
  - Connect latest input to inner terminal

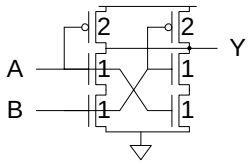
Asymmetric Gates

- Asymmetric gates favor one input over another
- Example: Suppose input A of a NAND gate is most critical
  - Use smaller transistor on A (less capacitance)
  - Boost size of noncritical input
  - So total resistance is same
- $g_A = 10/9$
- $g_B = 2$
- $g_{total} = g_A + g_B = 28/9$
- Asymmetric gate approaches g = 1 on critical input
- But total logical effort goes up



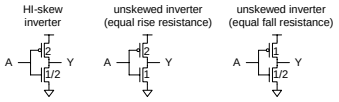
Symmetric Gates

- Inputs can be made perfectly symmetric



Skewed Gates

- Skewed gates favor one edge over another
- Ex: suppose rising output of inverter is most critical
  - Downsize noncritical nMOS transistor

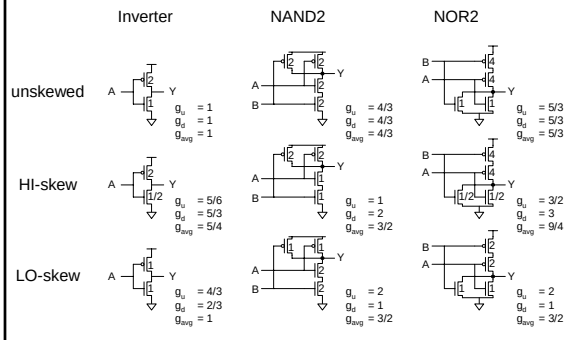


- Calculate logical effort by comparing to unskewed inverter with same effective resistance on that edge.
  - $g_u = 2.5 / 3 = 5/6$
  - $g_d = 2.5 / 1.5 = 5/3$

HI- and LO-Skew

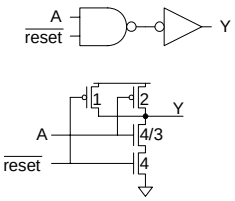
- Logical effort of a skewed gate for a particular transition is the ratio of the input capacitance of that gate to the input capacitance of an unskewed inverter delivering the same output current for the same transition
- Skewed gates reduce size of noncritical transistors
  - HI-skew gates favor rising output (small nMOS)
  - LO-skew gates favor falling output (small pMOS)
- Logical effort is smaller for favored direction
- But larger for the other direction

Catalog of Skewed Gates



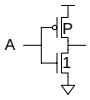
Asymmetric Skew

- Combine asymmetric and skewed gates
  - Downsize noncritical transistor on unimportant input
  - Reduces parasitic delay for critical input



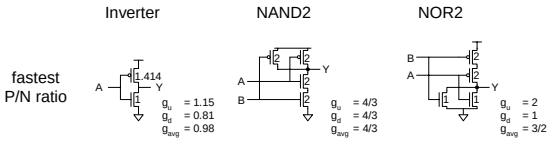
Best P/N Ratio for Least Avg Delay

- We have selected P/N ratio for unit rise and fall resistance (μ = 2-3 for an inverter).
- Alternative: ratio for least average delay
- Ex: inverter
  - Delay driving identical inverter
  - $t_{pdf} = (P+1)$
  - $t_{pdr} = (P+1)(\mu/P)$
  - $t_{pd} = (P+1)(1+\mu/P)/2 = (P + 1 + \mu + \mu/P)/2$
  - Differentiate t_{pd} w.r.t. P
  - Least delay for P = $\sqrt{\mu}$



P/N Ratios

- In general, best P/N ratio for lowest avg delay is the square root of that giving equal delay
  - Only improves average delay slightly for inverters
  - But significantly decreases area and power for NOR



- In practice, P/N ratio consideration not really average delay