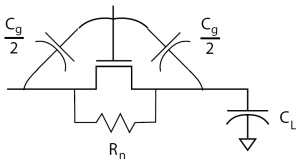
Topics

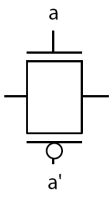
- Switch Logic
- Pseudo-nMOS gates.
- DCVS logic.
- Domino gates.

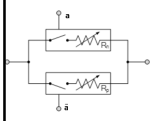
n-type Switch



- It requires only one transistor and one gate signal. It transmits a logic 0 well, but when V_{DD} is applied to the drain, the voltage at the source is ($V_{DD} - V_{in}$).
- When switch logic drives gate logic, n-type switches can cause electrical problems.

Transmission gate

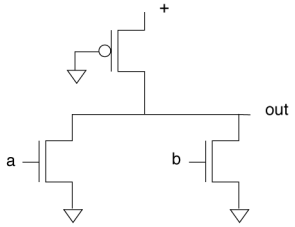




- when V_{DD} or V_{SS} is applied to the drain, V_{DD} or V_{SS} appears at the source.
- It requires
  - two transistors and their associated tubs;
  - both true and complement forms of the gate signal

Pseudo-nMOS

- Uses a p-type as a resistive pullup, n-type network for pulldowns.



Characteristics

- Consumes static power.
- Has much smaller pullup network than static gate.
- Pulldown time is longer because pullup is fighting.

Output voltages

- Logic 1 output is always at V_{DD}.
- Logic 0 output is above Vss.
- $V_{OL} = 0.25 (V_{DD} - V_{SS})$ is one plausible choice.

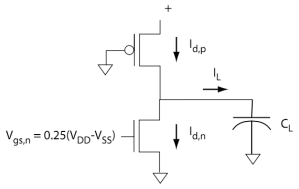
Producing output voltages

- For logic 0 output, pullup and pulldown form a voltage divider.
- Must choose n, p transistor sizes to create effective resistances of the required ratio.
- Effective resistance of pulldown network must be comptued in worst case—series n-types means larger transistors.

Transistor ratio calculation

- In steady state logic 0 output:
  - pullup is in linear region, $V_{ds} = V_{out} - (V_{DD} - V_{SS})$;
  - pulldown is in saturation.
- Pullup and pulldown have same current flowing through them.

Transistor ratio, cont'd.

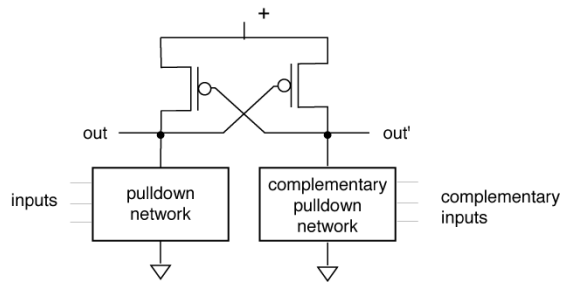


- Equate two currents:
  - $I_{dp} = I_{dd}$.
- Using 0.5 mm parameters, 3.3V power supply:
  - $W_p/L_p / W_n/L_n = 3.9$.

DCVS logic

- DCVSL = differential cascode voltage logic.
- Static logic—consumes no dynamic power.
- Uses latch to compute output quickly.
- Requires true/complement inputs, produces true/complement outputs.

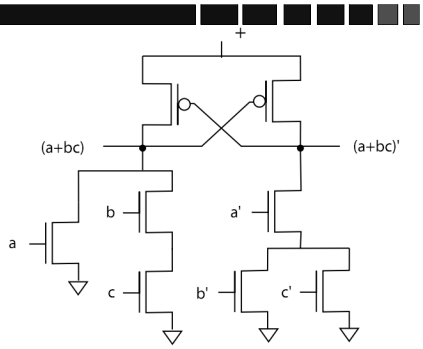
DCVS structure



DCVS operation

- Exactly one of true/complement pulldown networks will complete a path to the power supply.
- Pulldown network will lower output voltage, turning on other p-type, which also turns off p-type for node which is going down.

DCVS example



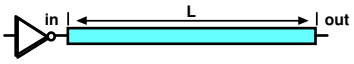
Other low-power gates

- The simplest way to reduce the operating voltage of a gate is to connect it to a lower power supply. We saw the relationship between power supply voltage and power consumption in Section 3.3.5:
  - For large V_t, Equation 3-7 tells us that delay changes linearly with power supply voltage.

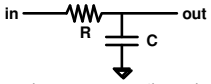
$$t_f = \frac{C_L(V_{DD}-V_{SS})}{I_d} = \frac{C_L(V_{DD}-V_{SS})}{0.5k'(W/L)(V_{DD}-V_{SS}-V_t)^2}. \quad \text{(EQ 3-7)}$$

  - Equation 3-15 tells us that power consumption varies quadratically with power supply voltage.

Delay in Wires: Lumped RC Model



Lumped RC Model:



R = R_s * L / W = r*L
(r = R_s / W - resistance per unit length)
C = L * W * C_{plate} = c*L
(c = W * C_{plate} - capacitance per unit length)

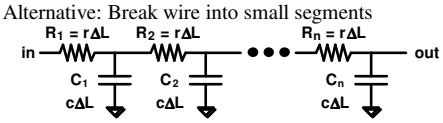
- Delay time constant (ignoring driving gate)

$\tau = R * C = (R_s * L / W) * (L * W * C_{plate})$
$= r * c * L^2$

- Problem: Overly Pessimistic

Delay in Long Wires - Distributed RC Model

- Alternative: Break wire into small segments



- Approx. Solution - 1st moment of impulse response

$$\tau(V_{out}) = rc(\Delta L)^2\left(\frac{N(N+1)}{2}\right)$$

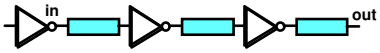
$$\tau(V_{out}) = \frac{rcL^2}{2} \quad \text{for } N \to \infty$$

- Important: delay still grows as square of length

Delay in Long Wires - Consequences in design

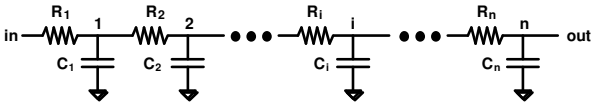
- Distributed RC model:

- Delay grows as square of L!
  - Choose wire material that minimizes r, c
  - Break wire into buffered segments to optimize delay



Elmore Delay

- Consider R-C ladder network with **unequal** values



- First-order time constant at node N is

$$\tau_N = \sum_{i=1}^{N} R_i \sum_{j=i}^{N} C_j = \sum_{i=1}^{N} C_i \sum_{j=1}^{i} R_j$$

- First-order time constant and node I is

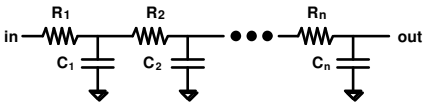
$$\tau_i = C_1 R_1 + C_2 (R_1 + R_2) + ... + C_i (R_1 + R_2 + ... + R_i)$$

Elmore Delay Applications

- Wire sizing to minimize delay
- Delay prediction of complex networks (as long as they take the form of a ladder)

Wire Sizing

- Recall distributed model of wire: multiple segments



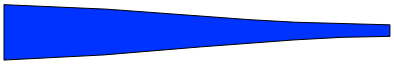
$$\tau_i = C_1 R_1 + C_2 (R_1 + R_2) + ... + C_i (R_1 + R_2 + ... + R_i)$$

note strong impact of R_1, lesser impact of R_2, etc

- Idea: Reduce overall delay by tapering segments
  - Make Segment 1 widest to reduce R1 (increases C1)
  - Make Segment 2 less wide to reduce R2 (increses C2)
  - etc.

Wire Sizing

- Ideal Result wire should taper exponentially - see Eq. 3-20, p. 163 [Fis95]:

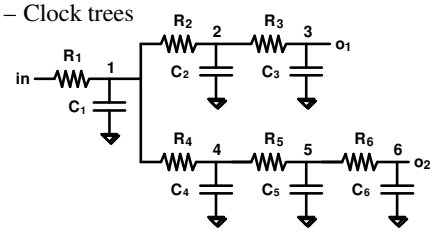


More pragmatic approach: step-tapered wire

[Fis95] J. Fishburn and C. Schevon, "Shaping a distributed-RC line to minimize Elmore delay", *IEEE Trans. on Circuits and Systems-I*, December 1995, pp. 1020-1022

Delay in RC-Trees

- Many interconnection networks are trees
  - Extracted RC circuit modeling a gate output
  - Clock trees

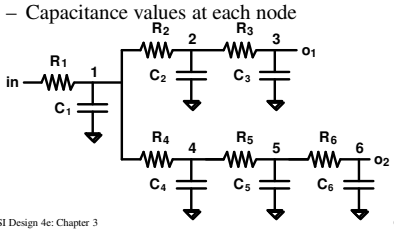


Delay in RC-Trees: Penfield-Rubenstein Bounds

- Key idea: characterize time constants in terms of
  - Path resistances between nodes

$$R_{ko} = \sum R_j \Rightarrow (R_j \in [path(in \rightarrow o) \cap path(k \rightarrow o)])$$

  - Capacitance values at each node



Elmore Delay Problem

- What are the Elmore time constants τ_1, τ_2, τ_3?