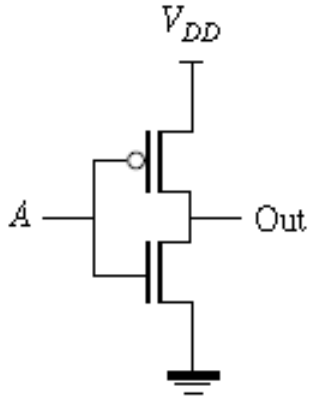
ELEC 402

Dynamic Logic Design
(Transfer Gates)
Lecture 11

Reza Molavi
Dept. of ECE
University of British Columbia
reza@ece.ubc.ca

Slides Courtesy : Dr. Res Saleh (UBC), Dr. D. Sengupta (AMD), Dr. B. Razavi (UCLA)

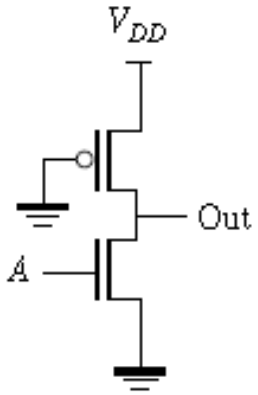
Static vs. Dynamic Logic Design



PMOS typically

2-times NMOS

For symmetric T_{PHL}/T_{PLH}



NMOS typically

4-times NMOS

For symmetric low V_{OL}

CMOS Logic

- No static power consumption
- Ideal V_{OL} and V_{OH}
- Symmetric rise and fall achievable
- Complementary pull-up required (area)

Pseudo-NMOS Logic

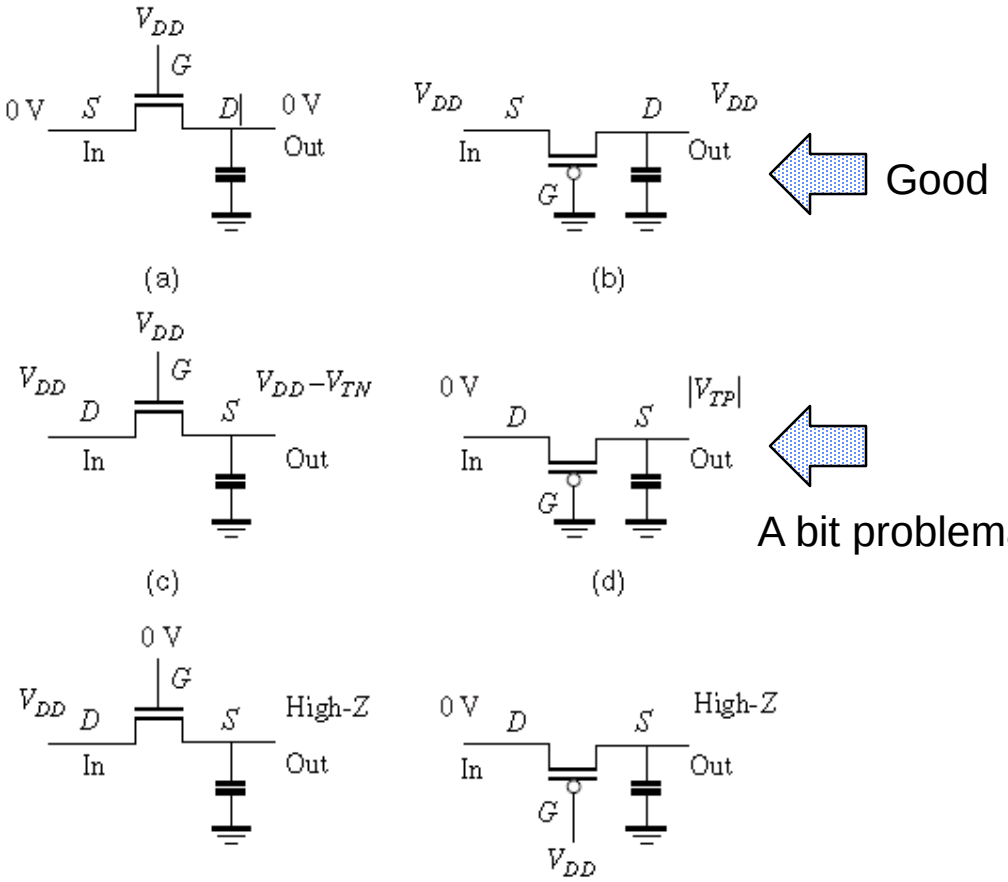
- Single pull-up device only
- Degraded V_{OL}(ratioed inverter)
- Static power when output is low
- Asymmetric rise/fall times

A solution which is both Fast and Area-efficient

Pass Transistor – Dynamic Storage Concept

NMOS

PMOS



Good

A bit problematic

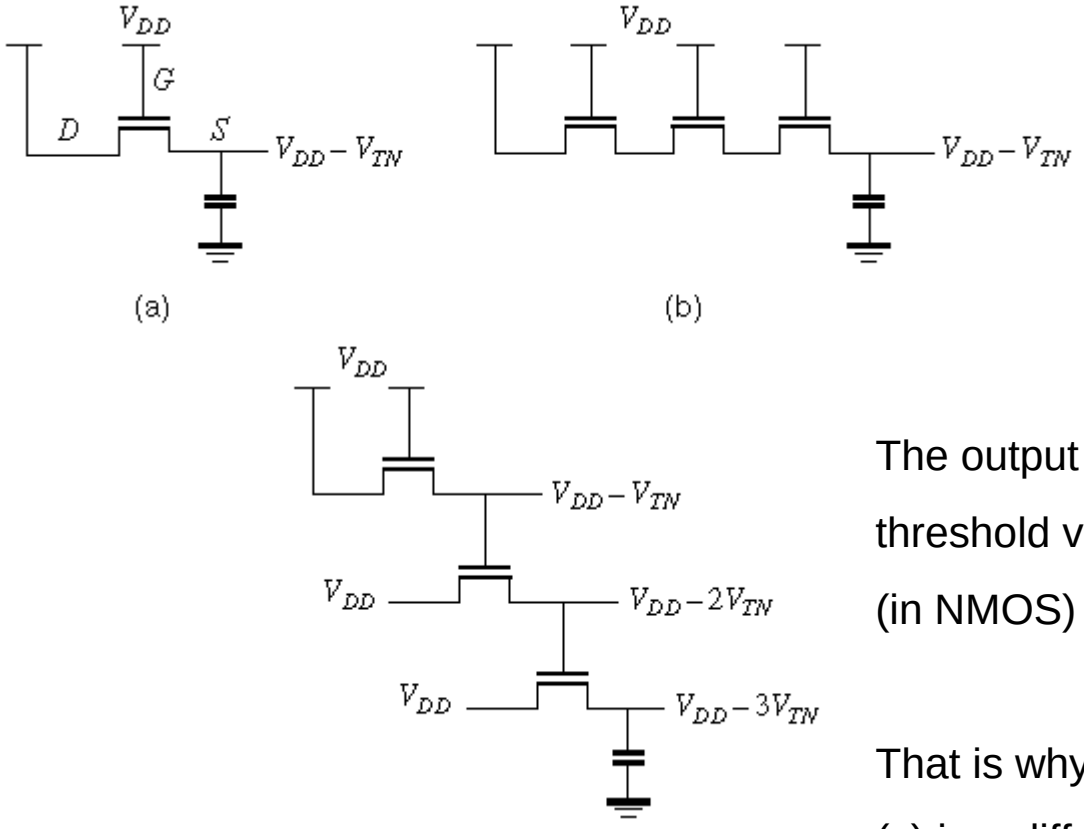
Transfers an input signal unaltered to the output when gate is ON

Hold the output value on output Capacitance when the gate is OFF (High – Z)

-Not always a DC path to Gnd or supply

- Prone to error from environment

Single Pass Transistor Shortcoming



The output can only climb up within one threshold voltage (V_t) of voltage applied to gate (in NMOS) – opposite applies for PMOS

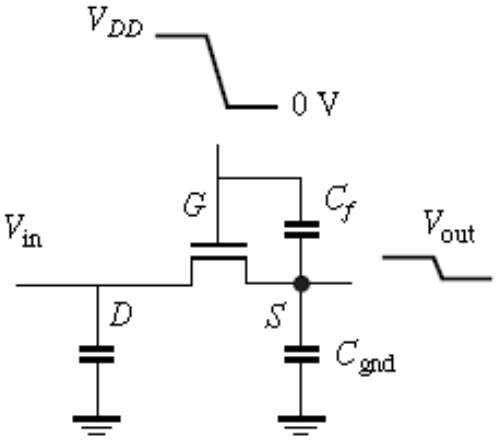
That is why (a) and (b) produce the same output bu

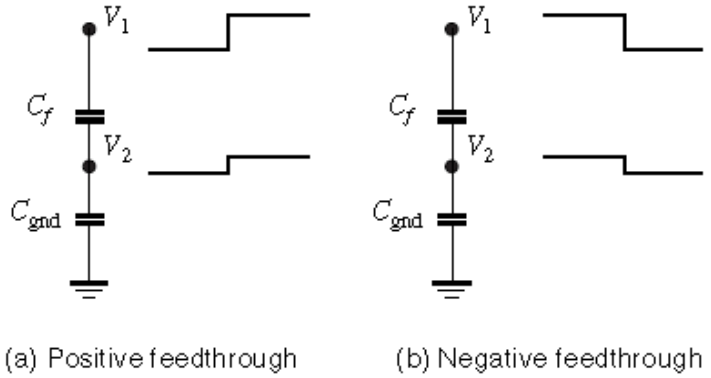
(c) is a different output voltage. .

Issues with Pass Transistors

Capacitive feed through

- Just as the switch (pass gate) is turning off there might be sudden change in the output value due to capacitive voltage division between the C_f and output capacitance (C_{gnd} in the figure)
- Not desirable, the stored value would be different/degraded



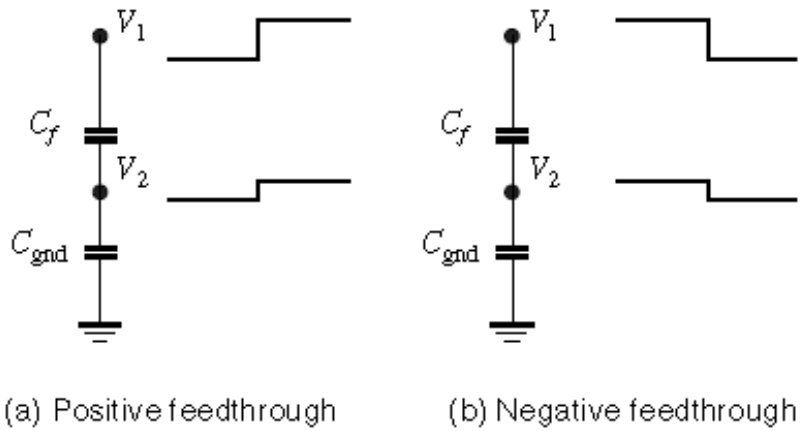


(a) Positive feedthrough (b) Negative feedthrough

Case (a) is an issue for PMOS

Case (b) is an issue for NMOS

Capacitive Feedthrough



(a) Positive feedthrough (b) Negative feedthrough

Charge equilibrium between two

Series-connected capacitors

$$C_f(V_1 - V_2) = C_{gnd}V_2$$

$$V_2 = \frac{C_f V_1}{C_f + C_{gnd}}$$

If abrupt

change in V_1

$$\Delta V_2 = \frac{C_f \Delta V_1}{C_f + C_{gnd}}$$

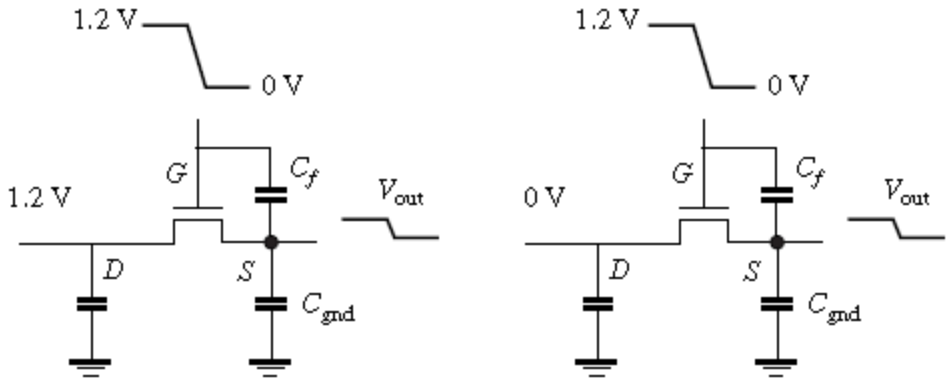
We need to know the value of C_{GS} and C_{out} to

Properly calculate this change (need to know what

Region of operation the switch is throughout the change)

Capacitive Feedthrough - Example

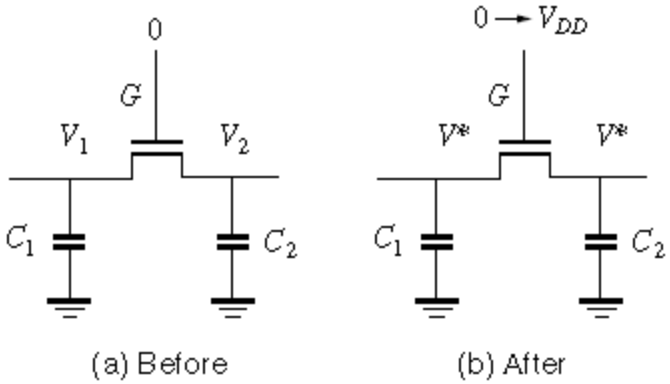
In the circuit below with the input at 1.2 V, what is the initial value of the output when the clock is at 1.2 V. Estimate the final value after the clock goes low. Repeat the problem for the case when the input is 0 V. Assume that the device is 4λ/2λ in 0.13 μm technology.



Charge Sharing

If two isolated nodes with different voltages are suddenly Connected they would share the charge,

may degrade the stored value



$$Q_{total} = C_1 V_1 + C_2 V_2$$ before

$$Q_{total} = (C_1 + C_2) V^*$$ after

$$V^* = \frac{(C_1 V_1 + C_2 V_2)}{C_1 + C_2}$$ New voltage

- Note that after charge sharing (switch turning on) there should still be V_T drop across V_{GS}
- Otherwise the source would not exceed V_{DD} - V_T (see next example)

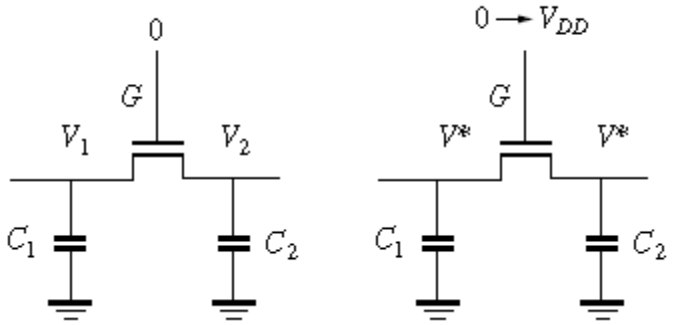
Charge Sharing – Example

In Figure 7.6a, compute the charge-sharing effects for the following cases assuming 0.13 μm technology parameters:

(a) $C_1 = 100$ fF, $C_2 = 20$ fF, $V_1 = 0$, $V_2 = 1.2$ V

(b) $C_1 = 20$ fF, $C_2 = 20$ fF, $V_1 = 0$, $V_2 = 1.2$ V

(c) $C_1 = 20$ fF, $C_2 = 100$ fF, $V_1 = 0$, $V_2 = 1.2$ V



(a) $V^* = \dfrac{(C_1V_1 + C_2V_2)}{C_1 + C_2} = \dfrac{(100\text{ fF} \times 0 + 20\text{ fF} \times 1.2)}{(20\text{ fF} + 100\text{ fF})} = 0.2$ V

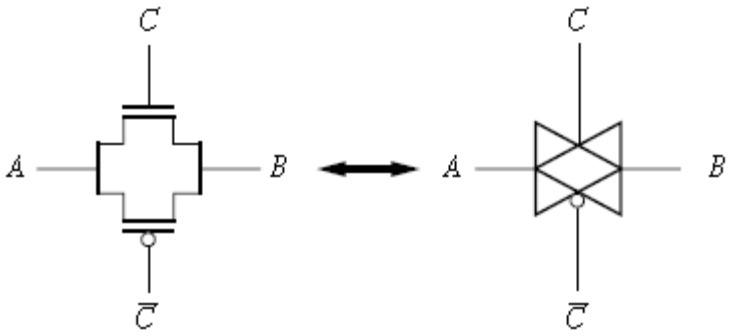
(b) $V^* = \dfrac{(C_1V_1 + C_2V_2)}{C_1 + C_2} = \dfrac{(20\text{ fF} \times 0 + 20\text{ fF} \times 1.2)}{(20\text{ fF} + 20\text{ fF})} = 0.6$ V

(c) $V^* = \dfrac{(C_1V_1 + C_2V_2)}{C_1 + C_2} = \dfrac{(20\text{ fF} \times 0 + 100\text{ fF} \times 1.2)}{(20\text{ fF} + 100\text{ fF})} = 1.0$ V

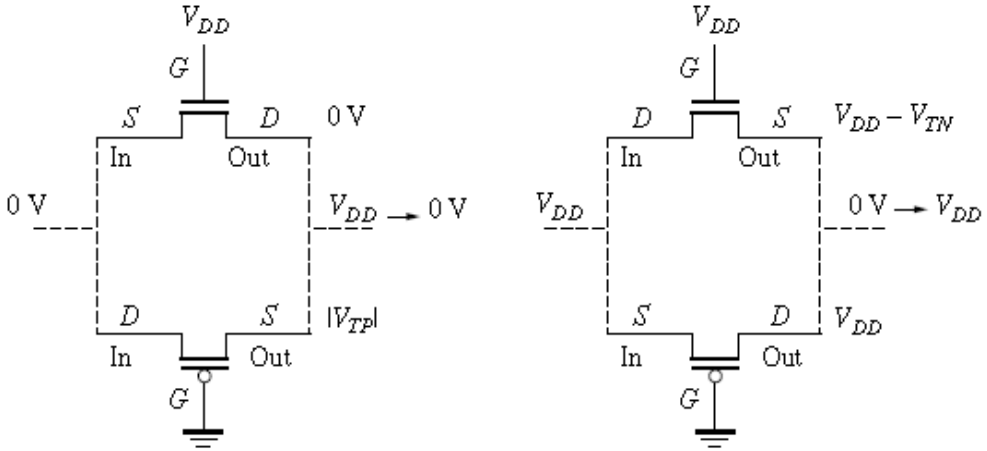
(Board Notes)

That is more than $V_{DD} - V_T$, therefore One side clips to V_{DD}-V_T and the rest of charge determines the voltage at other node

CMOS Transmission Gate



Using both control (C)signal and its complementary (C_bar) to pass the signal



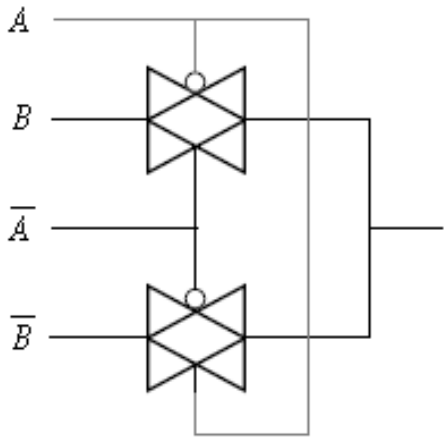
(Board Notes)

How about clock Feedthrough?

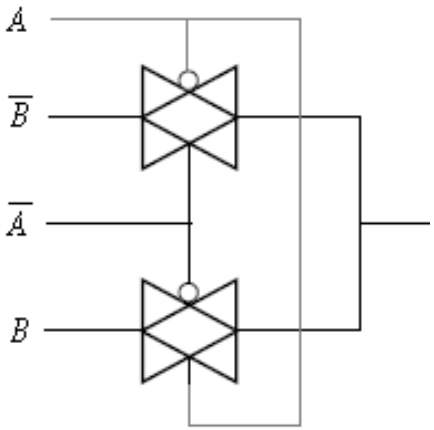
Is it better or worse than single

The combination of both devices allows proper passage of signal (always one Strong and one Weak device)

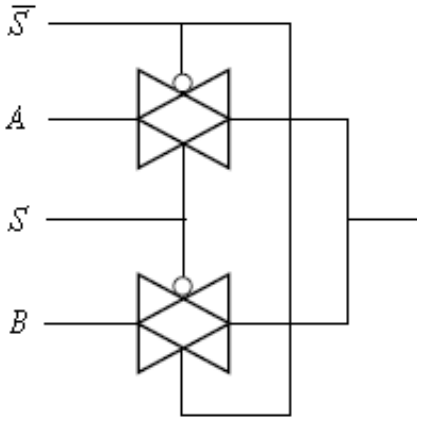
Logic Circuit Design Using TGs



XOR



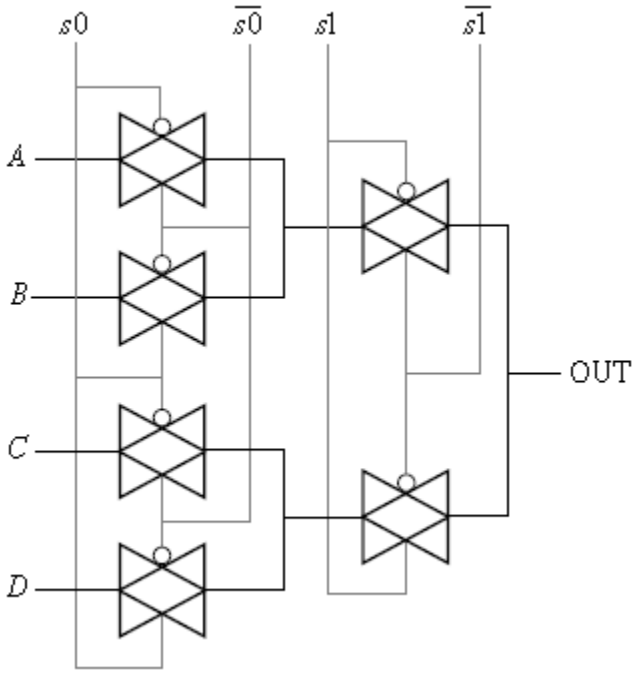
XNOR

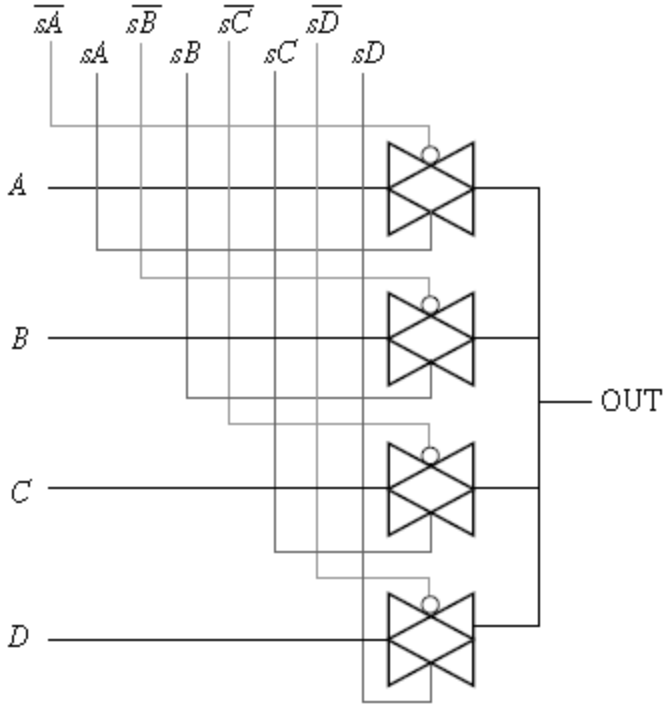


MUX2

- When two TGs are shorted at output, it does '+' operation (like OR)
- When a TG passes 'A' signal with 'C' control, it does '." operation (like AND)

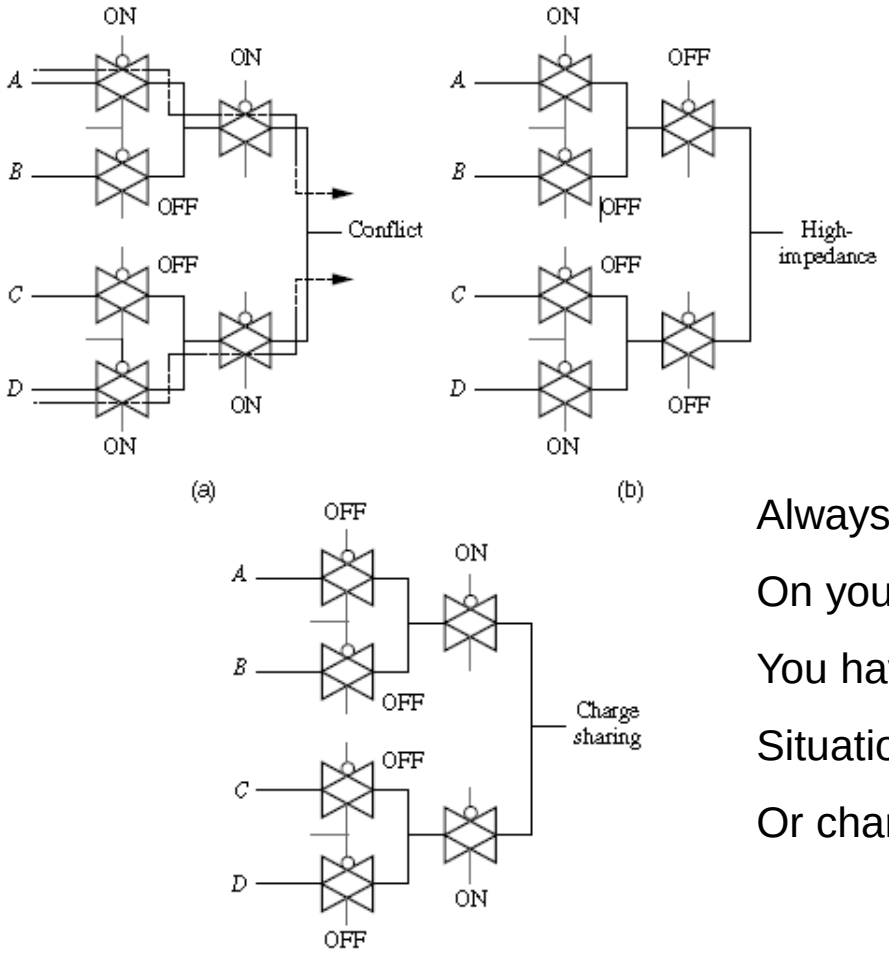
Example – MUX4 implementation





Which one is faster? We need to quantify the speed just like what we did with CMOS Logic gates

Important Design Note



Always do a sanity check

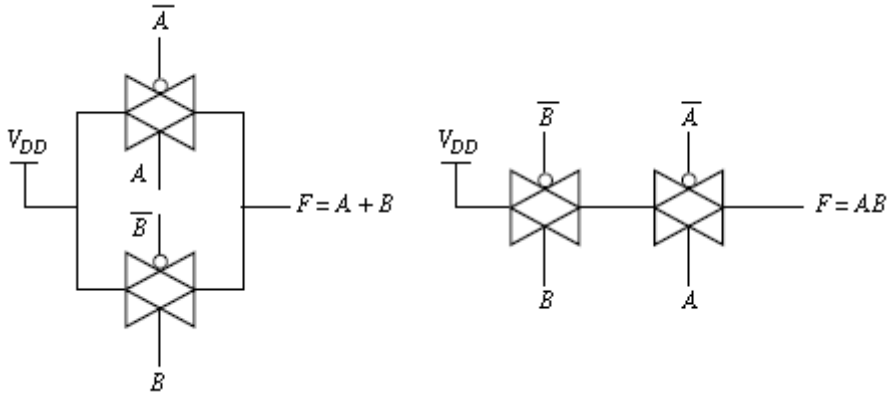
On your TG design to ensure

You have avoided all problematic

Situations (multiple path, no path

Or charge sharing scenario)

Custom Logic Design Using TGs



(Board notes)

Q: Is NMOS of TG needed in OR circuit?

Strategy: 1. Pick a few of your variables as control signal

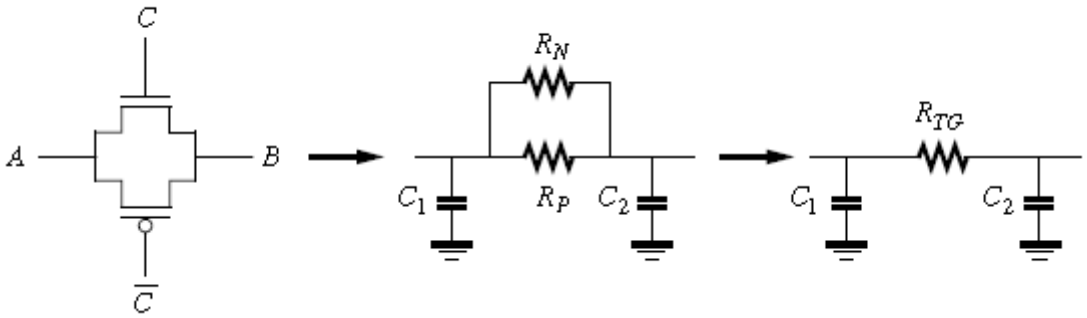
2. Build a truth table

3. Implement using simple AND/OR/MUX structures introduced

4. Simplify the circuit (if possible)

Example: Implement the function $F = AB + A\overline{B}C + \overline{A}\,\overline{C}$ using transmission gates.

TG Delay Calculation



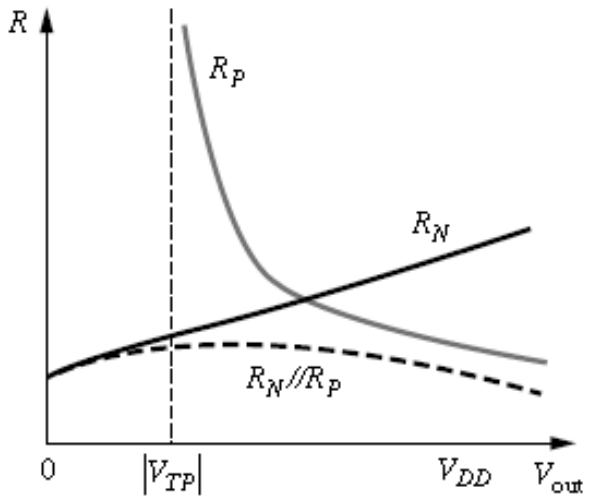
We need to calculate R_N and R_P and their equivalent parallel resistance

Let's start with the case that we are propagating 0 (A is 0 and B is at V_{DD} but transitioning To 0),

NMOS is partly in Saturation, partly in linear region $R_N = \frac{V_{out}}{I_{Dsat,n}}$ $R_N = \frac{V_{out}}{I_{Dlin,n}}$

PMOS is in saturation till output reaches V_T and then is off) $R_p = \frac{V_{out}}{I_{Dsat,p}}$

TG Delay Calculation - II



(a) V_{in} = 0 V, V_{out} = $V_{DD} \rightarrow$ 0 V

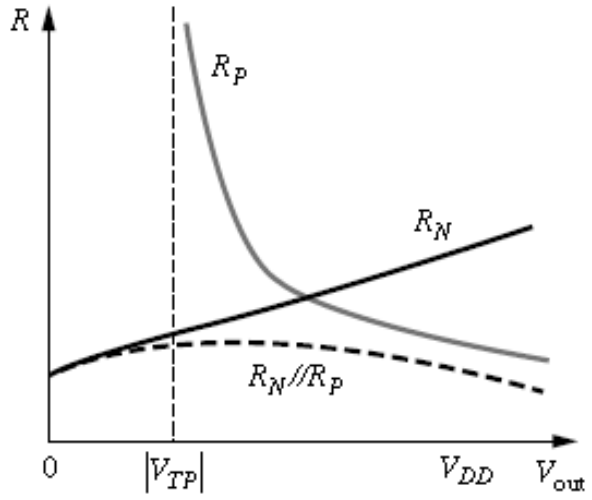
-NMOS resistance decreases as device enters linear from saturation (remember $I_{deminas}$ vs V_{ds} curve

- PMOS resistance goes to infinity at VT (switch turns-off)

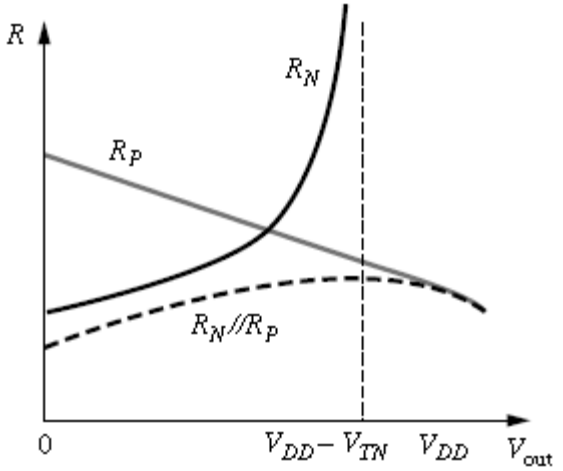
- The average of value of PMOS resistance roughly is twice that of its saturation value, $R_P \sim 2R_{eqp}$

- The parallel combination of the two resistances remains constant!

TG Delay Calculation - Resistance



(a) V_{in} = 0 V, V_{out} = V_{DD} → 0 V



(b) V_{in} = V_{DD}, V_{out} = 0 V → V_{DD}

$$R_{TG} = R_N//R_P = R_{eqn}//2R_{eqp} \approx R_{eqn}//4.8R_{eqn} = 0.83R_{eqn} \approx R_{eqn}$$

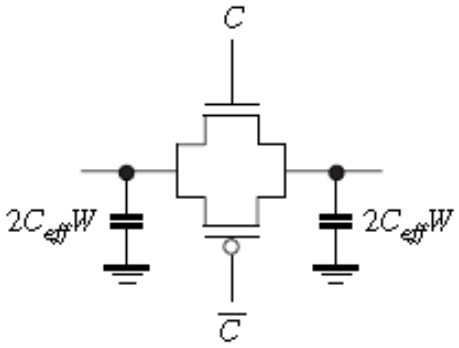
$$R_{TG} = R_N//R_P = 2R_{eqn}//R_{eqp} \approx 2R_{eqn}//2.4R_{eqn} = 1.1R_{eqn} \approx R_{eqn}$$

Similar Unit resistance due to parallel combination of structures

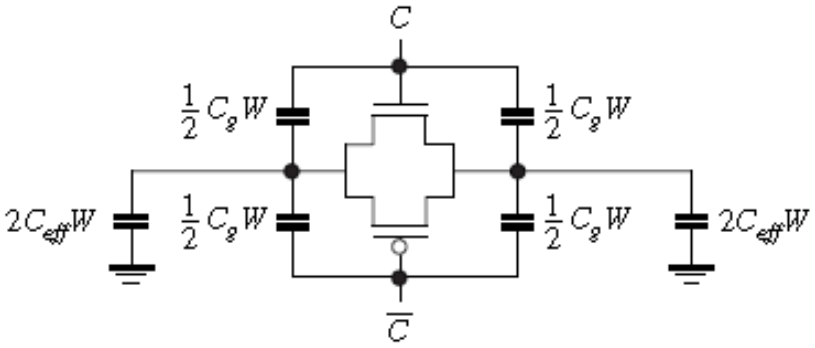
PMOS and NMOS can be chosen equal size in TG

$$R_{TG} = R_{eqn}\left(\frac{L}{W}\right)$$ Scale with proper W/L

TG Delay Calculation - Capacitance



(a) *OFF* state



(b) *ON* state

$$C_{\text{in}} = C_{\text{out}} = C_{eff}(W_n + W_p)$$

$$C_{\text{in}} = C_{\text{out}} = C_{eff}2W$$

Devices are off, no contribution from gate Capacitance, Equal devices

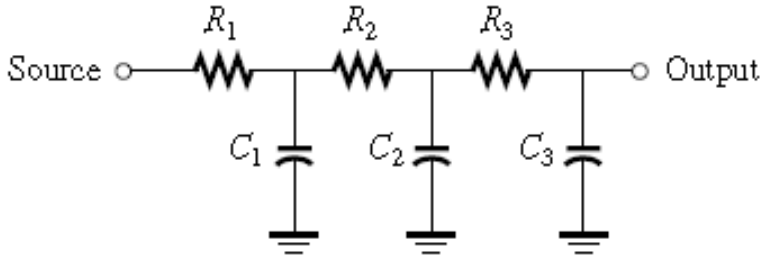
$$C_{\text{in}} = C_{\text{out}} = C_{eff}(W_n + W_p) + \frac{1}{2}(C_gW_n + C_gW_p)$$

$$C_{\text{in}} = C_{\text{out}} = C_{eff}2W + C_gW$$

Devices are in linear, half the gate capacitance to Source & drain, Equal devices

Elmore Delay - Review

Useful technique to find a quick hand calculation for RC delay for long chains



$$\tau = R_1C_1 + (R_1 + R_2)C_2 + (R_1 + R_2 + R_3)C_3$$

$$\tau_i = \sum_k (C_k \times R_{ik})$$

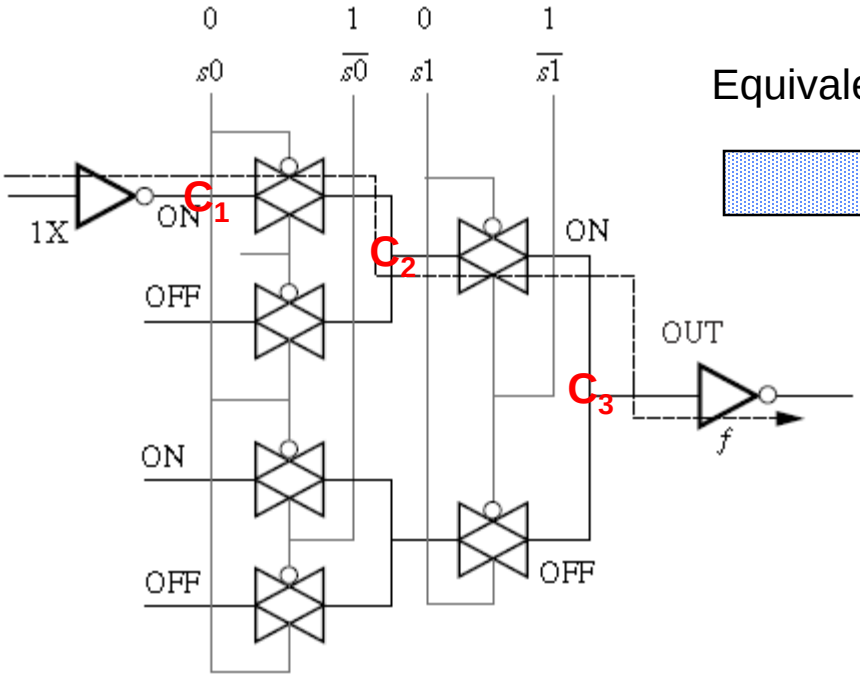
Where Γ_i is the total delay, you have to sum all the RC components in the following steps:

1. Highlight the main path, sum all R's and multiply up by the final C_{main}
2. Find the common resistors between the main path and any other path (R_X) that contains a C_X
3. Sum all the components $\Gamma_i = \sum R_X C_X + R_{main} C_{main}$

TG Delay Calculation – Cont'd

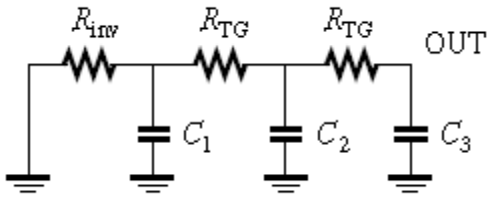
Now using Elmore delay and TG delay calculations we can answer last the question

Which of the two MUX4 implementations (slide 12) are faster(assume fan-out of f times1x inverter)?



Implementation I

Equivalent model



$t_1 = R_{inv}C_1 + (R_{inv} + R_{TG})C_2 + (R_{inv} + 2R_{TG})C_3$ From Elmore

$C_1 = 3C_{eff}W + C_gW + 2C_{eff}W$

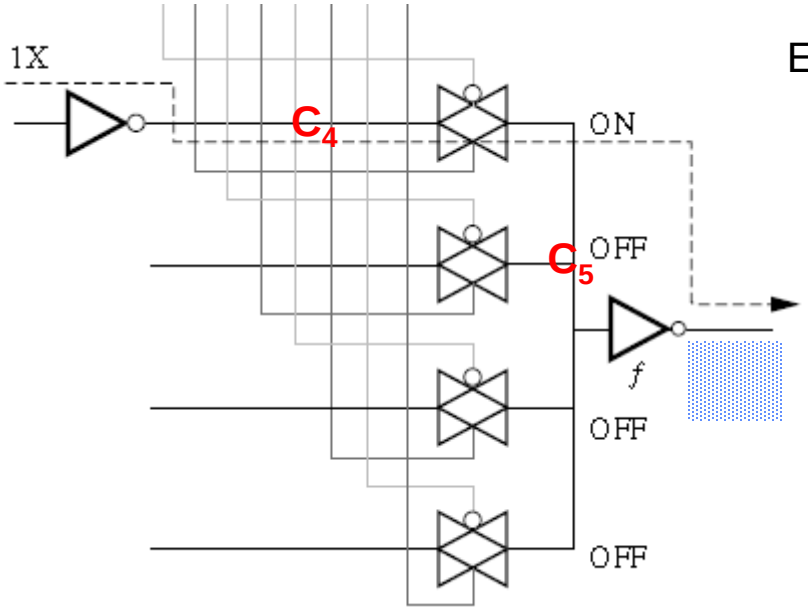
$C_2 = C_gW \times 2 + 2C_{eff}W \times 3$

$C_3 = 2C_{eff}W \times 2 + C_gW + 3fC_gW$

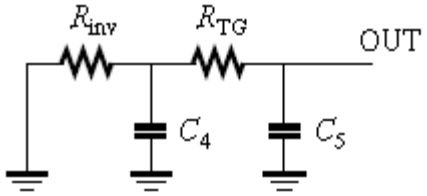
$t_1 = R(29C_{eff} + 8C_g + 9fC_g)W$

$R_{inv}=R_{TG} = R$

TG Delay Calculation – Cont'd



Equivalent model



$$t_2 = R_{inv}C_4 + (R_{inv} + R_{TG})C_5$$ From Elmore

$$C_4 = 3C_{eff}W + 2C_{eff}W + C_gW$$

$$C_5 = 2C_{eff}W \times 4 + C_gW + 3fC_gW$$

$$t_2 = R(21C_{eff} + 3C_g + 6fC_g)W$$

Implementation II

$R_{inv}=R_{TG} = R$

Faster

Note that the loading of ON and OFF transmission gates are different!