UNIVERSITÀ DEGLI STUDI DI TRIESTE

dipartimento di ingegneria e architettura

Combinational Logic Gates in CMOS

A.Carini – Digital Integrated Circuits

Digital electronics

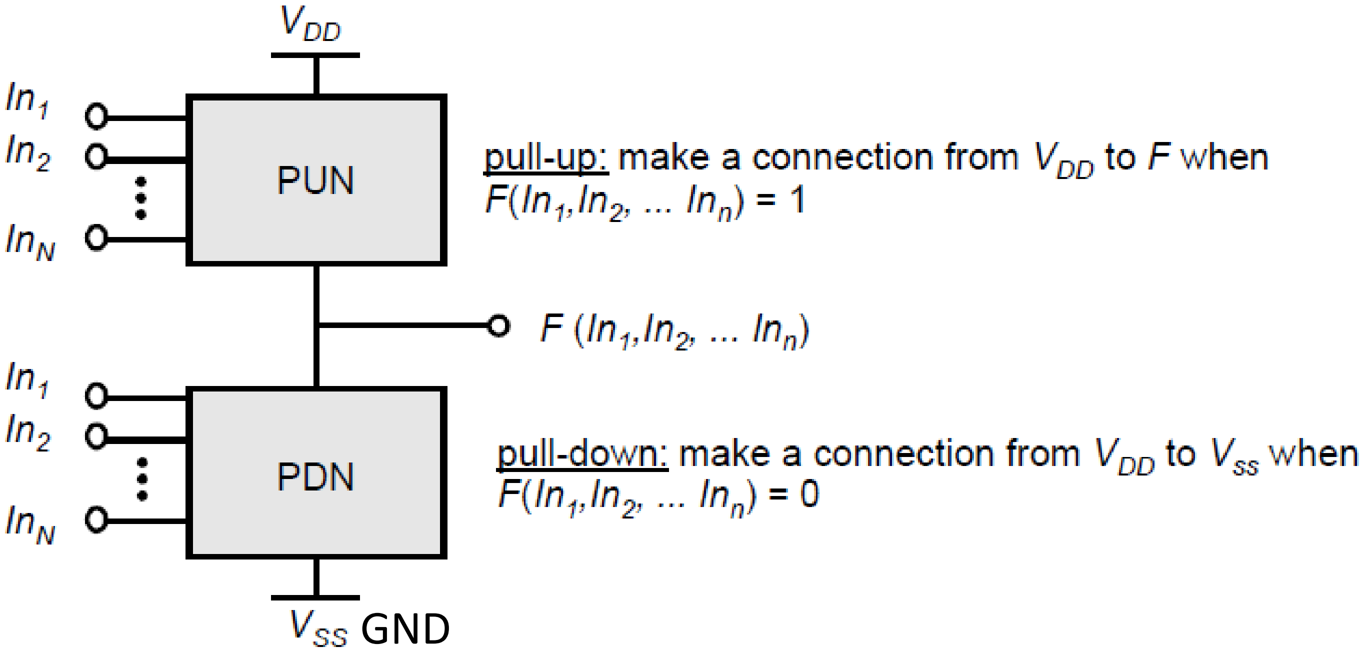
- The focus is on *combinational logic* circuits, that have the property that at any point in time, the output of the circuit is related to its current input signals by some Boolean expression.
- There are numerous circuit styles to implement a given logic function.
- The common design metrics by which a gate is evaluated are area, speed, energy and power.
- Depending on the application, the emphasis will be on different metrics.
- The switching speed of digital circuits is the primary metric in a high-performance processor, while it is energy dissipation in a battery operated circuit.
- Robustness to noise and reliability are also very important considerations.

Static CMOS Design

- The most widely used logic style is static complementary CMOS.
- The complementary CMOS falls under a broad class of logic circuits called *static* circuits in which at every point in time (except during transients), each gate output is connected to either V_{DD} or *GND* via a low-resistance path.
- This is in contrast to the *dynamic* circuit class, which relies on temporary storage of signal values on the capacitance of high-impedance circuit nodes.
- The latter approach has the advantage that the resulting gate is simpler and faster.
- Its design and operation are however more involved and prone to failure due to an increased sensitivity to noise.

Static CMOS Design

- A static CMOS gate is a combination of two networks, called the *pull-up network* (PUN) and the *pull-down* network (PDN).



pull-up: make a connection from V_{DD} to F when $F(In_1, In_2, \dots In_n) = 1$

pull-down: make a connection from V_{DD} to V_{ss} when $F(In_1, In_2, \dots In_n) = 0$

Static CMOS Design

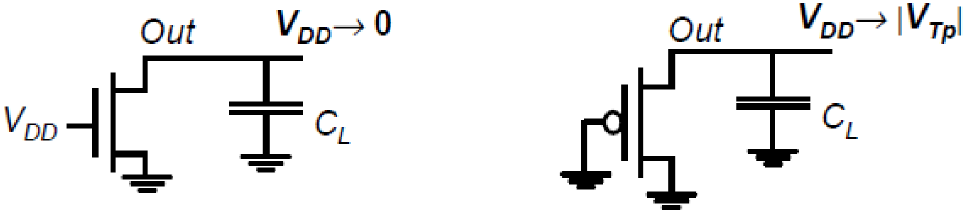
- The PUN and PDN networks are complementary: *one and only one* of the networks is conducting in steady state.
- Once the transients have settled, a path always exists between V_{DD} and the output *F*, realizing a high output (“one”), or, alternatively, between *GND* and *F* for a low output (“zero”).
- This is equivalent to stating that the output node is always a *low-impedance* node in steady state.

Static CMOS Design

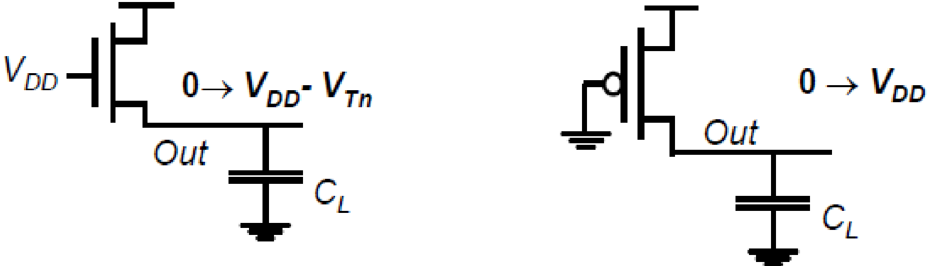
In constructing the PDN and PUN networks, the following observations apply:

- A transistor can be thought of as a switch controlled by its gate signal. An NMOS switch is *on* when the controlling signal is high and is *off* when the controlling signal is low. A PMOS transistor acts as an inverse switch, with dual controls.
- The PDN is constructed using NMOS devices, while PMOS transistors are used in the PUN. The primary reason for this choice is that NMOS transistors produce “strong zeros,” and PMOS devices generate “strong ones”.

Static CMOS Design



(a) pulling down a node using NMOS and PMOS switches

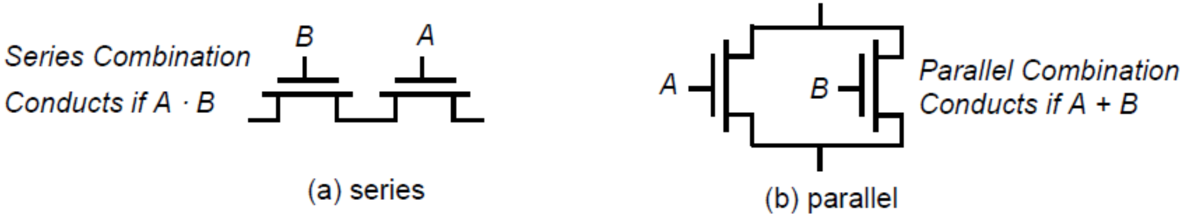


(b) pulling down a node using NMOS and PMOS switches

Figure 6.3 Simple examples illustrate why an NMOS should be used as a pull-down, and a PMOS should be used as a pull-up device.

Static CMOS Design

- A set of construction rules can be derived to construct logic functions:
  - NMOS devices connected in series corresponds to an AND function.
  - NMOS transistors connected in parallel represent an OR function.
  - A series connection of PMOS conducts if both inputs are low, representing a NOR function: $\overline{A} \cdot \overline{B} = \overline{A + B}$
  - PMOS transistors in parallel implement a NAND function: $\overline{A} + \overline{B} = \overline{A \cdot B}$
- Using De Morgan’s theorems, it can be shown that the pull-up and pull-down networks of a complementary CMOS structure are *dual* networks.



(a) series

(b) parallel

Figure 6.4 NMOS logic rules — series devices implement an AND, and parallel devices implement an OR.

Static CMOS Design

- To construct a CMOS gate, one of the networks (e.g., PDN) is implemented using combinations of series and parallel devices.
- The other network (i.e., PUN) is obtained using duality principle by walking the hierarchy, replacing series sub-nets with parallel sub-nets, and parallel sub-nets with series sub-nets.
- The complete CMOS gate is constructed by combining the PDN with the PUN.
- The complementary gate is naturally *inverting*, implementing only functions such as NAND, NOR, and XNOR.
- The number of transistors required to implement an *N*-input logic gate is 2*N*.

Static CMOS Design

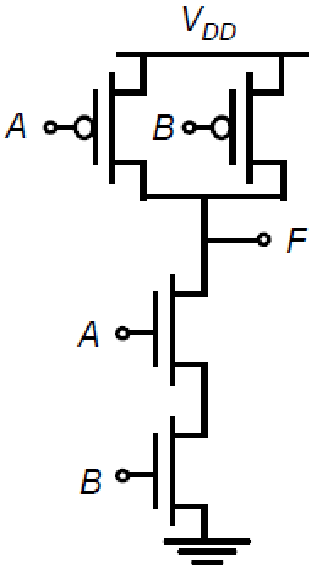


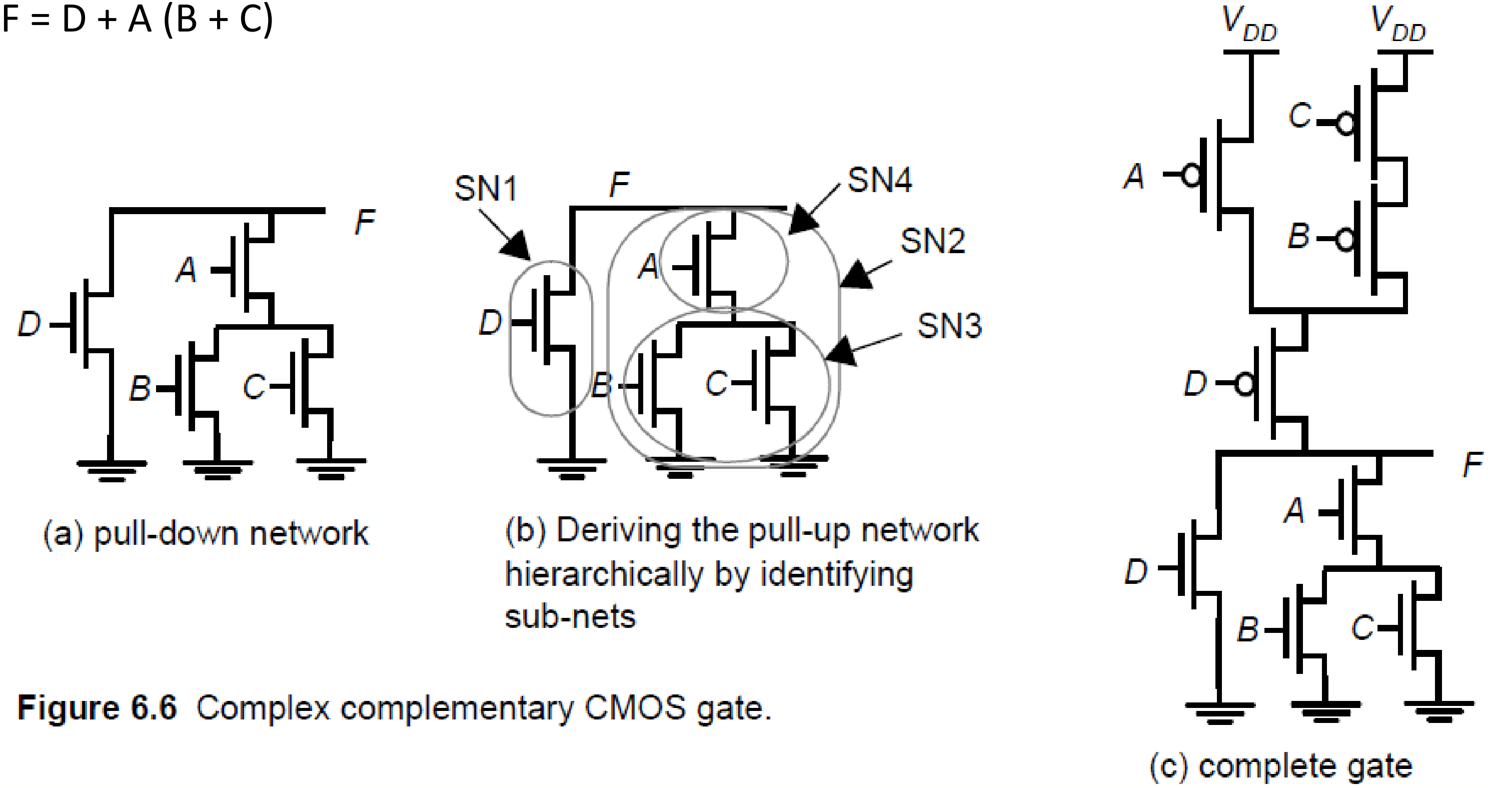
Table 6.1Truth Table for 2 input NAND

A	B	F
0	0	1
0	1	1
1	0	1
1	1	0

Figure 6.5 Two-input NAND gate in complementary static CMOS style.

Static CMOS Design

- $F = \overline{D + A\,(B + C)}$



(a) pull-down network

(b) Deriving the pull-up network hierarchically by identifying sub-nets

(c) complete gate

Figure 6.6 Complex complementary CMOS gate.

Static Properties of Complementary CMOS Gates

- Complementary CMOS gates inherit all the nice properties of the basic CMOS inverter.
- They exhibit rail to rail swing with $V_{OH} = V_{DD}$ and $V_{OL} = GND$.
- The circuits also have no static power dissipation, since the circuits are designed such that the pull-down and pull-up networks are mutually exclusive.
- The analysis of the DC voltage transfer characteristics and the noise margins is more complicated than for the inverter, as these parameters depend upon the data input patterns applied to gate.

Static Properties of Complementary CMOS Gates

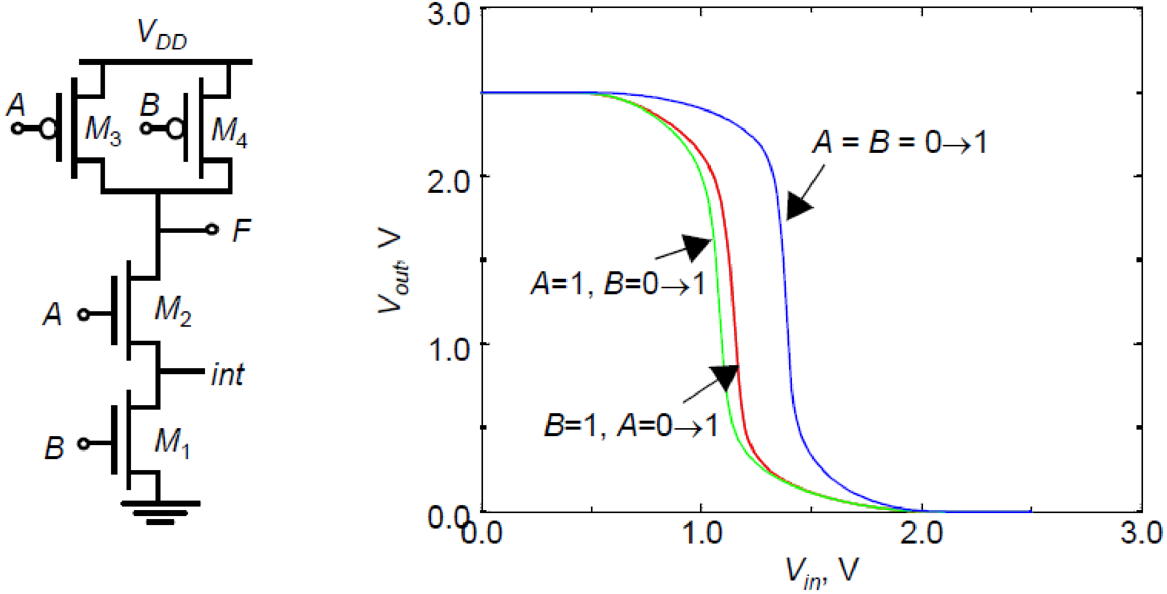
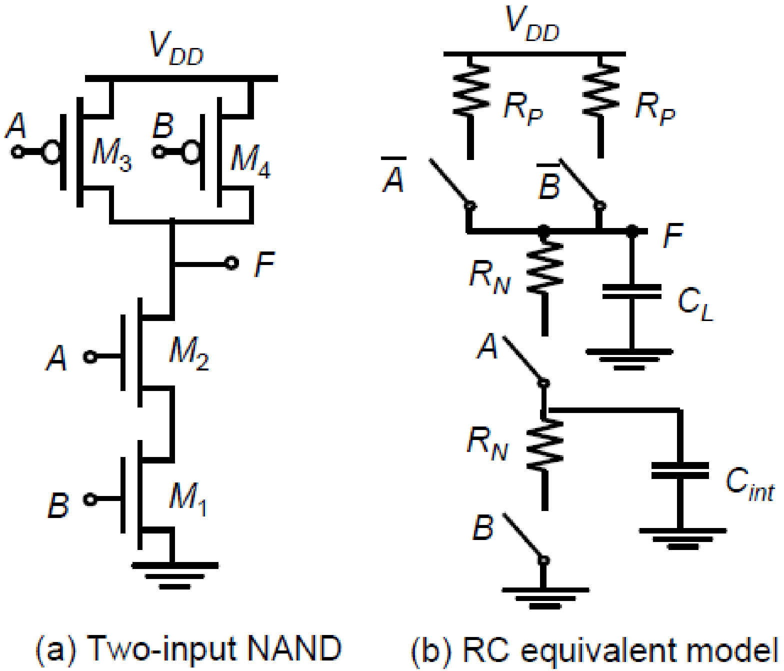


Figure 6.7 The VTC of a two-input NAND is data-dependent. NMOS devices are 0.5µm/0.25µm while the PMOS devices are sized at 0.75µm/0.25µm.

Propagation Delay of Complementary CMOS Gates

- The computation of propagation delay proceeds in a fashion similar to the static inverter.
- Each transistor is modeled as a resistor in series with an ideal switch.
- The value of the resistance is dependent on the power supply voltage and an equivalent large signal resistance, scaled by the ratio of device width over length, must be used.
- The logic is transformed into an equivalent RC network that includes the effect of internal node capacitances.
- Similar to the noise margins—**the propagation delay depends upon the input patterns**.

Propagation Delay of Complementary CMOS Gates



(a) Two-input NAND (b) RC equivalent model

Figure 6.8 Equivalent RC model for a 2-input NAND gate.

$$t_{pLH1} = 0.69 \times (R_p/2) \times C_L,$$

$$t_{pLH2} = 0.69 \times R_p \times C_L,$$

$$t_{pHL} = 0.69 \times (2\ R_n) \times C_L,$$

Propagation Delay of Complementary CMOS Gates

- When sizing the transistors in a gate with multiple fan-in’s, we should pick the combination of inputs that triggers the worst-case conditions.
- For example, for a NAND gate to have the same pull-down delay (t_{phl}) as a minimum-sized inverter, the NMOS devices in the NAND stack must be made twice as wide so that the equivalent resistance the NAND pull-down is the same as the inverter. The PMOS devices can remain unchanged.
- This first-order analysis assumes that the extra capacitance introduced by widening the transistors can be ignored.

Propagation Delay of Complementary CMOS Gates

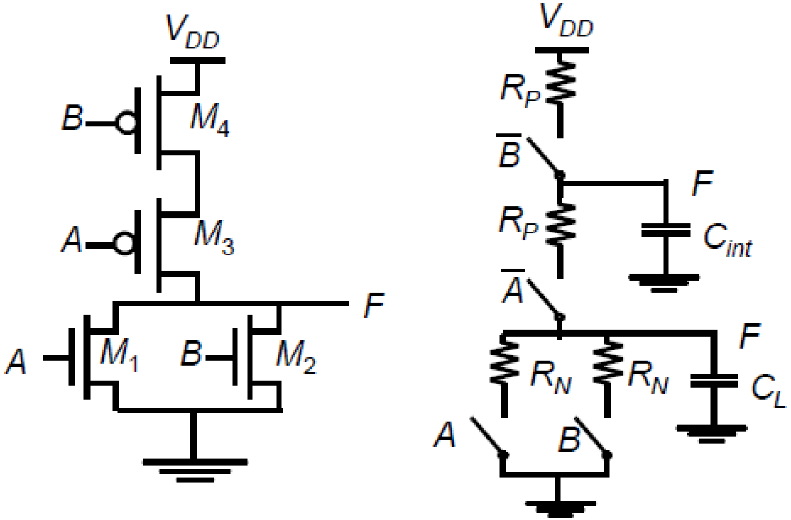


Figure 6.10 Sizing of a NOR gate to produce the same delay as an inverter with size of NMOS: 0.5µm/0.25µm and PMOS: 1.5µm/0.25µm.

Propagation Delay of Complementary CMOS Gates

- The worst-case pull-down transition happens when only one of the NMOS devices turns on (i.e., if either *A* or *B* is high).
- Assume that the goal is to size the NOR gate such that it has approximately the same delay as a given inverter.
- Since the pull-down path in the worst case is a single device, the NMOS devices (*M*1 and *M*2) can have the same device widths as the NMOS device in the inverter.
- For the output to be pulled high, both devices must be turned on. Since the resistances add, the devices must be made two times larger compared to the PMOS in the inverter.
- Since PMOS devices have a lower mobility relative to NMOS devices, stacking devices in series must be avoided as much as possible.
- A **NAND implementation** is clearly **preferred over a NOR implementation** for implementing generic logic.

Propagation Delay of Complementary CMOS Gates

- So far in the analysis of propagation delay, we have ignored the effect of internal node capacitances.
- In complex logic gates that have large *fan-in*, the internal node capacitances can become significant.
- The internal capacitances consist of the junction capacitance of the transistors, as well as the gate-to-source and gate-to-drain capacitances.
- The latter are turned into capacitances to ground using the Miller equivalence.
- The delay analysis for such a circuit involves solving concentrated RC networks.

Propagation Delay of Complementary CMOS Gates

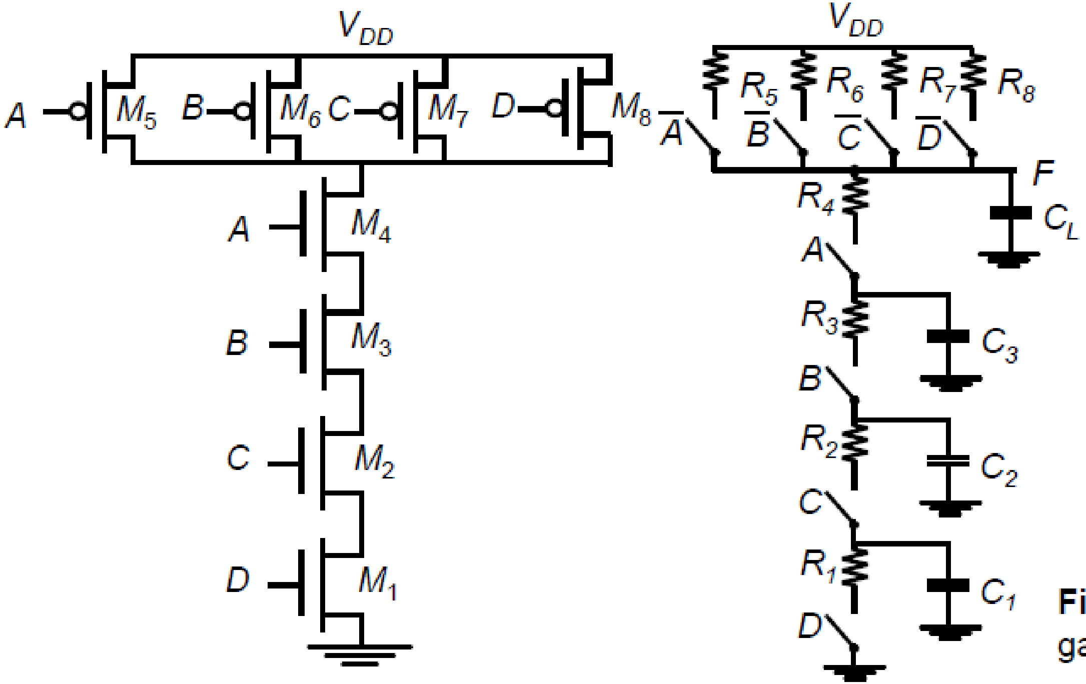


Figure 6.11 Four input NAND gate and its RC model.

Propagation Delay of Complementary CMOS Gates

- The propagation delay can be computed using the Elmore delay model and is approximated as:

$$t_{pHL} = 0.69(R_1 \cdot C_1 + (R_1 + R_2) \cdot C_2 + (R_1 + R_2 + R_3) \cdot C_3 + (R_1 + R_2 + R_3 + R_4) \cdot C_L)$$

- Notice that the resistance of *M*1 appears in all the terms, which makes this device especially important when attempting to minimize delay.
- Assuming that all NMOS devices have an equal size

$$t_{pHL} = 0.69R_N(C_1 + 2 \cdot C_2 + 3 \cdot C_3 + 4 \cdot C_L)$$

Propagation Delay of Complementary CMOS Gates

- While complementary CMOS is a very robust and simple approach for implementing logic gates, there are two major problems as the *fan-in* increases.
- First, the number of transistors required to implement an *N* fan-in gate is *2N*.
- The second problem is that propagation delay of a complementary CMOS gate deteriorates rapidly as a function of the fan-in.
- The large number of transistors (*2N*) increases the overall capacitance of the gate.
- For an *N*-input NAND gate, the output capacitance increases linearly with the fan-in since the number of PMOS devices connected to the output node increases linearly with the fan-in.
- Also, a series connection of transistors slows the gate as well, because the effective (dis)charging resistance is increased. For the *N*-input NAND gate, the effective resistance of the PDN path increases linearly with the fan-in.
- Since the output capacitance increase linearly and the pull-down resistance increases linearly, **the high-to-low delay can increase in a quadratic fashion**.

Design Techniques for Large Fan-in

Several approaches may be used to reduce delays in large fan-in circuits.

1. **Transistor Sizing**

- The most obvious solution is to increase the overall transistor size.
- This lowers the resistance of devices in series and lowers the time constant.
- However, increasing the transistor size, results in larger parasitic capacitors, which do not only affect the *propagation delay* of the gate in question, but also present a larger load to the preceding gate.
- This technique should, therefore, be used with caution.
- If the load capacitance is dominated by the intrinsic capacitance of the gate, widening the device only creates a “self-loading” effect, and the *propagation delay* is unaffected.

Design Techniques for Large Fan-in

2. Progressive Transistor Sizing

- We saw that the resistance of $M1$ ($R1$) appears N times in the delay equation, the resistance of $M2$ ($R2$) appears N-1 times, etc.
- It is clear that $R1$ should be made the smallest, $R2$ the next smallest, etc.
- A progressive scaling of the transistors is beneficial: $M1 > M2 > M3 > M_N$.

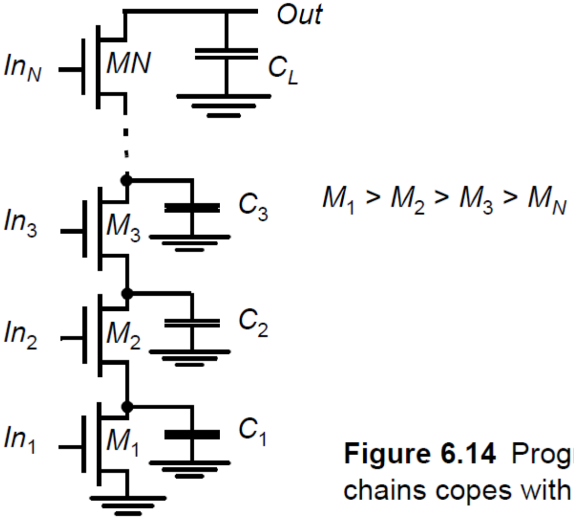


Figure 6.14 Progressive sizing of transistors in large transistor chains copes with the extra load of internal capacitances.

Design Techniques for Large Fan-in

3. Input Re-Ordering

- Some signals in combinational logic blocks might be more critical than others.
- An input signal is called *critical* if it is the last signal to assume a stable value.
- The path through the logic which determines the ultimate speed of the structure is called the *critical path*.
- Putting the critical-path transistors closer to the gate output can result in a speedup.

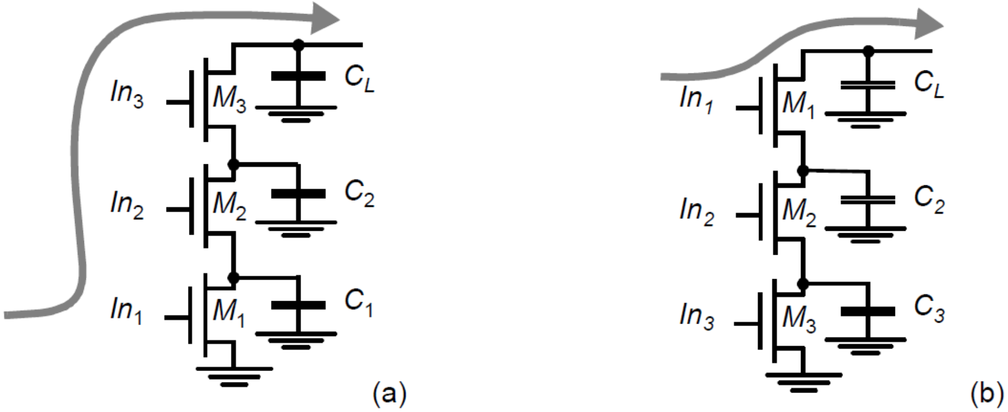


Figure 6.15 Influence of transistor ordering on delay. Signal In_1 is the critical signal.

Design Techniques for Large Fan-in

4. Logic Restructuring

- Manipulating the logic equations can reduce the fan-in requirements and hence reduce the gate delay.

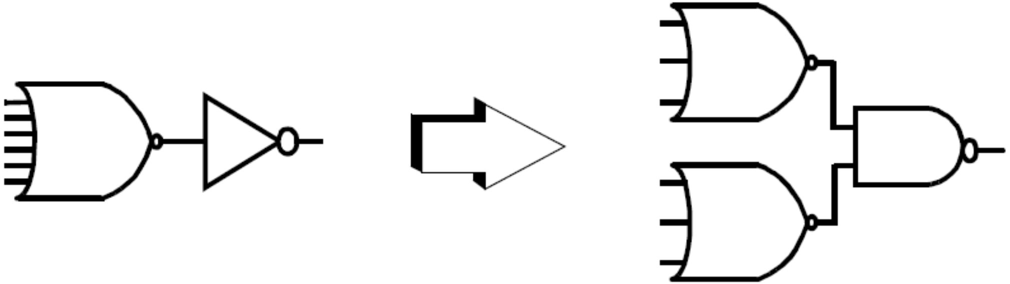


Figure 6.16 Logic restructuring can reduce the gate fan-in.

Ratioed Logic or Pseudo-NMOS Logic

- Ratioed logic reduces the number of transistors required to implement a given logic function, at the cost of reduced robustness and extra power dissipation.
- The purpose of the PUN in complementary CMOS is to provide a conditional path between V_{DD} and the output when the PDN is turned *off*.
- In ratioed logic, the entire PUN is replaced with a single unconditional load device that pulls up the output for a high output.
- The gate consists of an NMOS pull-down network that realizes the *logic function*, and a simple *load device*.

Ratioed Logic or Pseudo-NMOS Logic

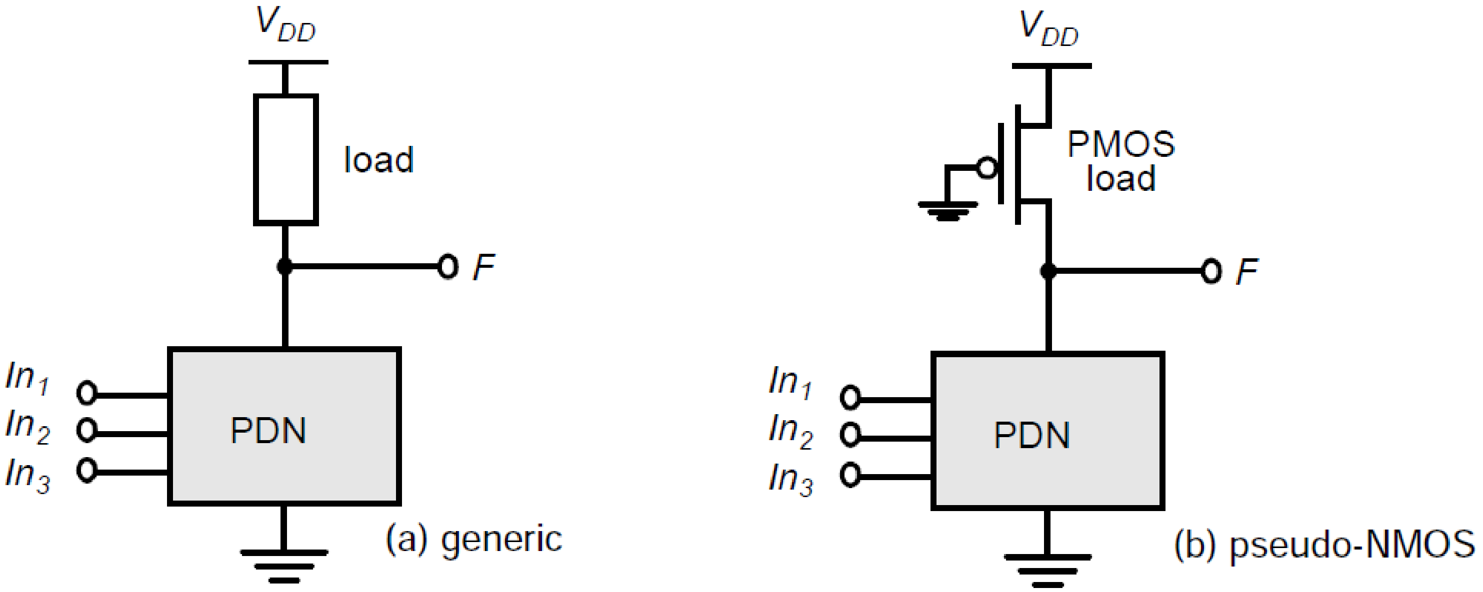


Figure 6.27 Ratioed logic gate.

Ratioed Logic or Pseudo-NMOS Logic

- The clear advantage of pseudo-NMOS is the reduced number of transistors (N+1 versus 2N for complementary CMOS).
- The nominal high output voltage (V_{OH}) for this gate is $V_{DD.}$
- On the other hand, the **nominal low output voltage is not 0 V.**
- This results in reduced noise margins and more importantly static power dissipation.
- The sizing of the load device relative to the pull-down devices can be used to trade-off parameters such a *noise margin*, *propagation delay* and *power dissipation*.
- Since the output voltage swing depends upon the ratio between the NMOS and PMOS sizes, the circuit is called ***ratioed***.

Ratioed Logic or Pseudo-NMOS Logic

- The value of V_{OL} is obtained by equating the currents through the driver and load devices for $V_{in} = V_{DD}$. At this operation point, it is reasonable to assume that the NMOS device resides in linear mode, while the PMOS load is saturated.

$$k_n\left((V_{DD}-V_{Tn})V_{OL}-\frac{V_{OL}^2}{2}\right) = k_p\left((-V_{DD}-V_{Tp})\cdot V_{DSATp}-\frac{V_{DSATp}^2}{2}\right)$$

- Assuming that $V_{OL} << V_{DD}-V_T$ and $V_{Tn} = |V_{Tp}|$, V_{OL} can be approximated as:

$$V_{OL} \approx \frac{k_p(+V_{DD}+V_{Tp})\cdot|V_{DSATp}|}{k_n(V_{DD}-V_{Tn})} \approx \frac{\mu_p\cdot W_p}{\mu_n\cdot W_n}\cdot|V_{DSATp}|$$

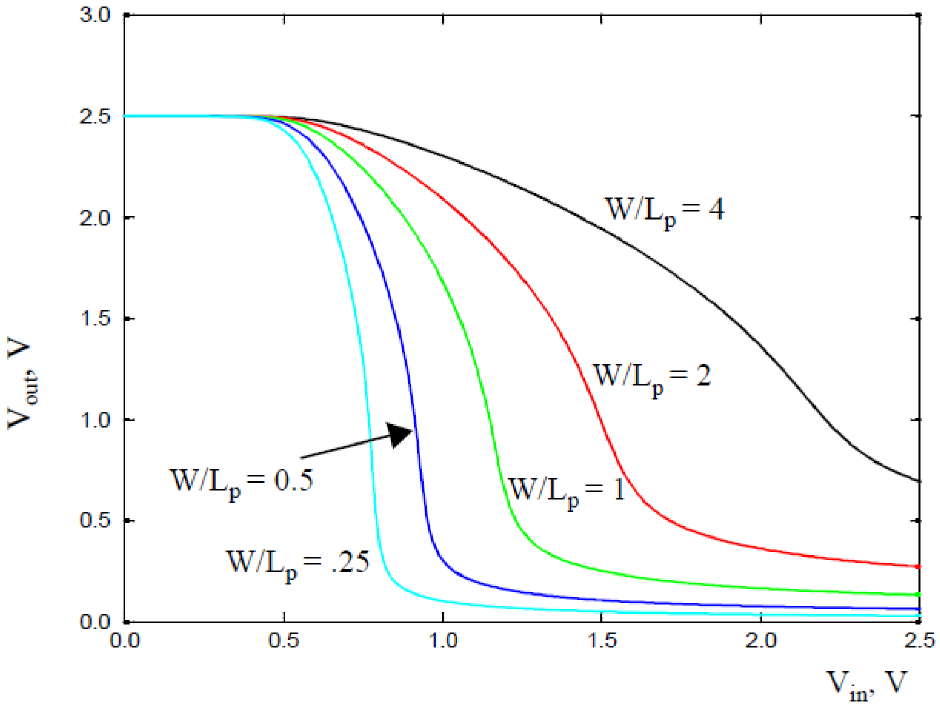
- In order to make V_{OL} as small as possible, the PMOS device should be sized much smaller than the NMOS pull-down devices. Unfortunately, this has a negative impact on the *propagation delay* for charging up the output node.

Ratioed Logic or Pseudo-NMOS Logic

- A major disadvantage of the pseudo-NMOS gate is the static power that is dissipated when the output is low through the direct current path that exists between V_{DD} and *GND*.

$$P_{low} = V_{DD}I_{low} \approx V_{DD} \cdot k_p\left((-V_{DD} - V_{Tp}) \cdot V_{DSATp} - \frac{V_{DSATp}^2}{2}\right)$$

Ratioed Logic or Pseudo-NMOS Logic



W_N/L_N=0.5 um/0.25 um

Figure 6.28 Voltage-transfer curves of the pseudo-NMOS inverter as a function of the PMOS size.

Ratioed Logic or Pseudo-NMOS Logic

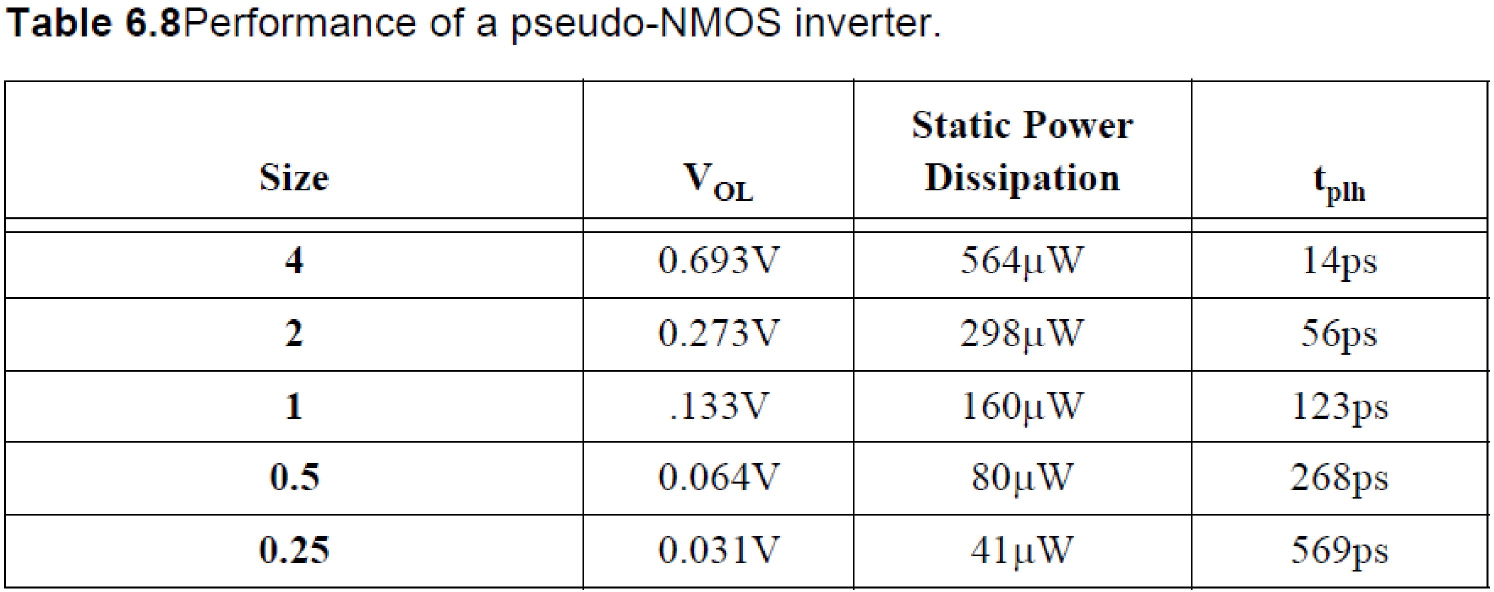
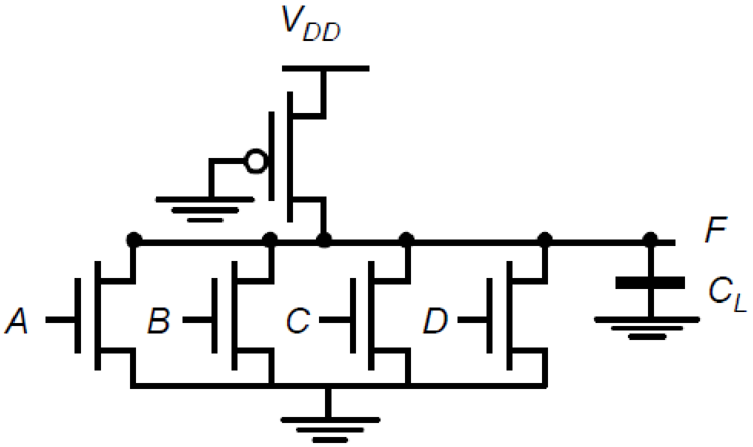


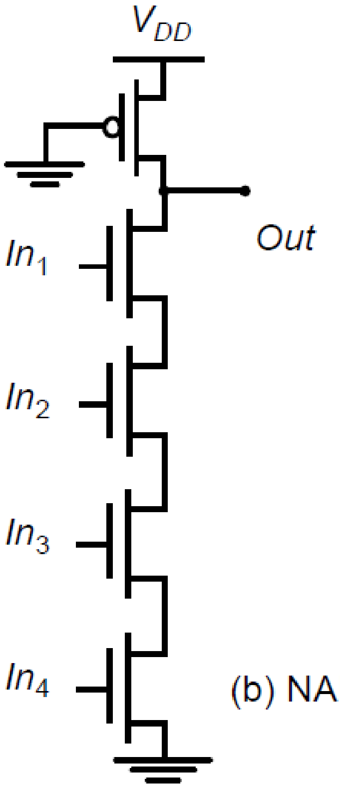
Table 6.8 Performance of a pseudo-NMOS inverter.

Size	V_{OL}	Static Power Dissipation	t_{plh}
4	0.693V	564µW	14ps
2	0.273V	298µW	56ps
1	.133V	160µW	123ps
0.5	0.064V	80µW	268ps
0.25	0.031V	41µW	569ps

Ratioed Logic or Pseudo-NMOS Logic



(a) NOR



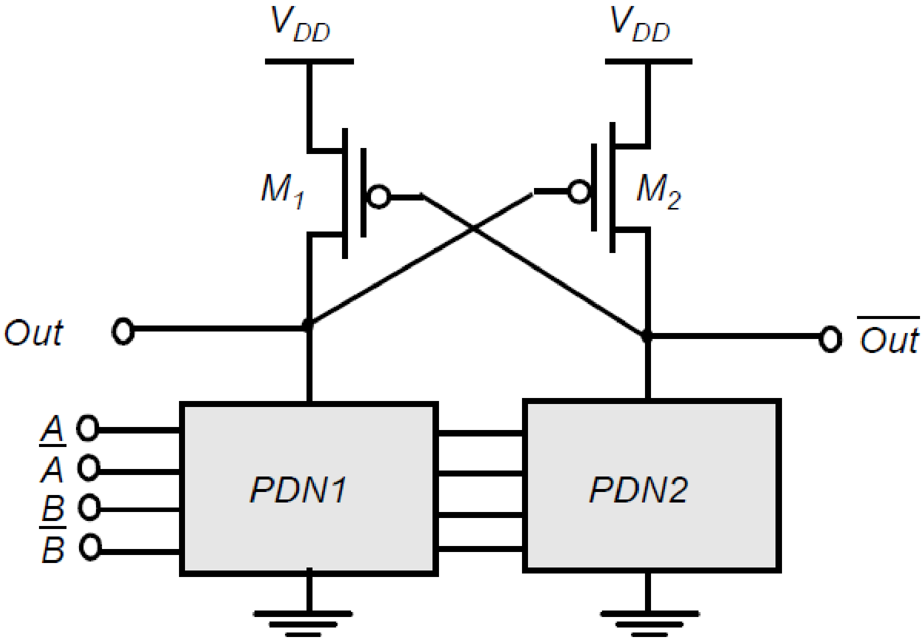
(b) NAND

Figure 6.29 Four-input pseudo-NMOS NOR and NAND gates.

Differential Cascode Voltage Switch Logic (or DCVSL)

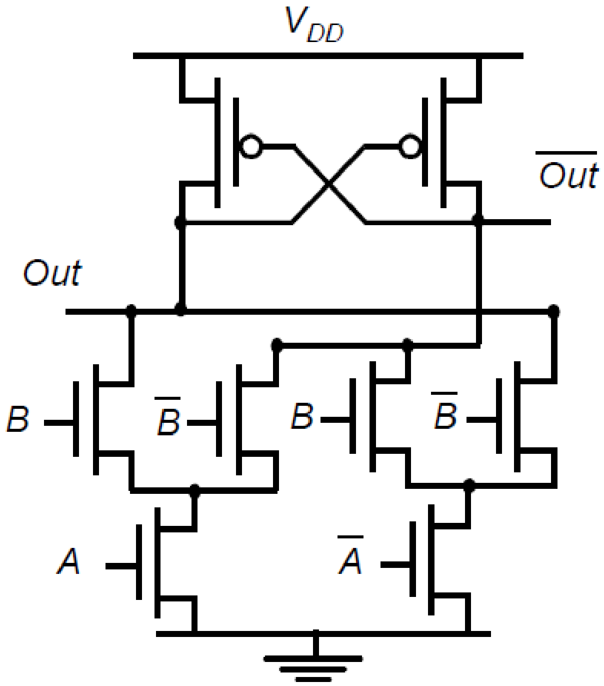
- It is possible to create a ratioed logic style that completely eliminates static currents and provides rail-to-rail swing.
- Such a gate combines two concepts: *differential logic* and *positive feedback*.
- A differential gate requires that each input is provided in complementary format, and produces complementary outputs in turn.
- The feedback mechanism ensures that the load device is turned off when not needed.
- An example of such a logic family is the *Differential Cascode Voltage Switch Logic* (or DCVSL) one.

Differential Cascode Voltage Switch Logic (or DCVSL)



(a) Basic principle

Differential Cascode Voltage Switch Logic (or DCVSL)



(b) XOR-XNOR gate

Differential Cascode Voltage Switch Logic (or DCVSL)

- The resulting circuit exhibits a rail-to-rail swing, and the static power dissipation is eliminated: in steady state, none of the stacked pull-down networks and load devices are simultaneously conducting.
- However, the circuit is still ratioed since the sizing of the PMOS devices relative to the pull-down devices is critical to functionality, not just performance.
- In addition to the problem of increase complexity in design, this circuit style still has a power-dissipation problem that is due to cross-over currents.
- During the transition, there is a period of time when PMOS and PDN are turned on simultaneously, producing a short circuit path.

Differential Cascode Voltage Switch Logic (or DCVSL)

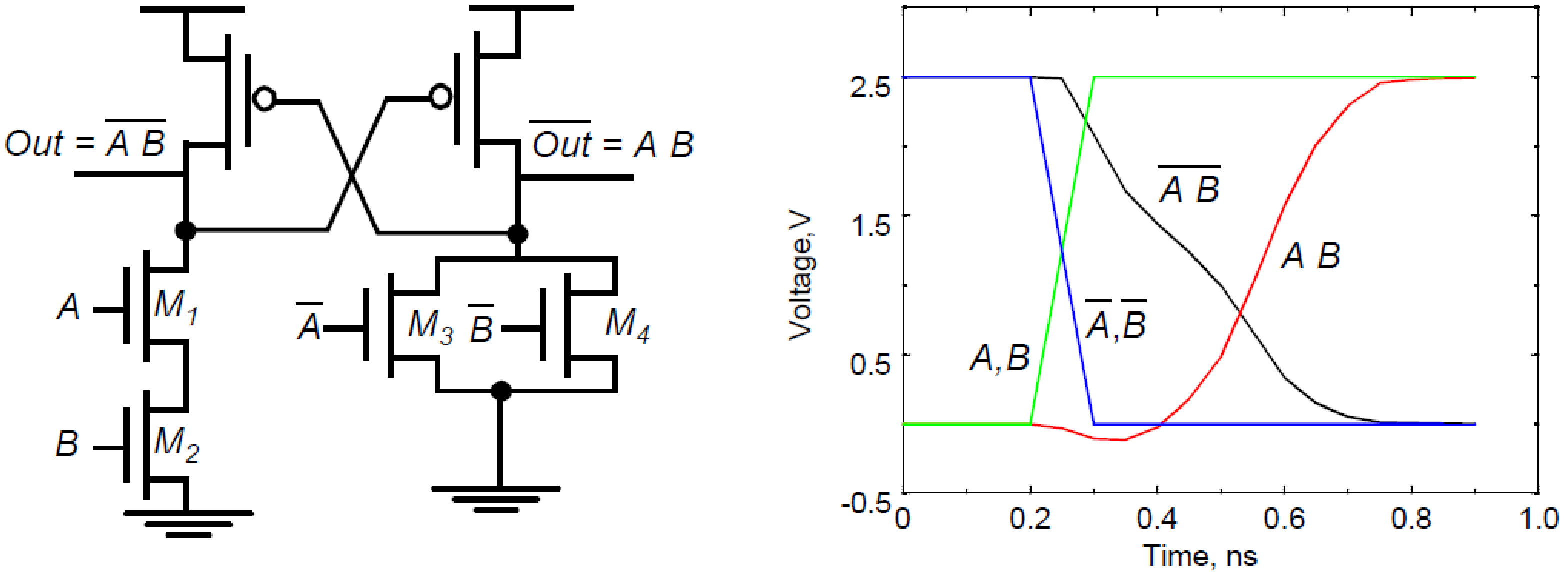


Figure 6.31 Transient response of a simple AND/NAND DCVSL gate. M_1 and M_2 1μm/0.25μm, M_3 and M_4 are 0.5μm/0.25μm and the cross-coupled PMOS devices are 1.5μm/0.25μm.

See:

- J. M. Rabaey, A. Chandrakasan, B. Nikolic, «Digital Integrated Circuits: A Design Perspective», Pearson, 2003
  - Cap. 6.2.1-6.2.2