– Module wires will get worse, but only slowly

– You don’t think to rethink your wires in your adder, memory

Or even your super-scalar processor core

- It does let you design more modules
- Continued scaling of uniprocessor performance is getting hard

-Machines using global resources run into wire limitations

-Machines will have to become more explicitly parallel

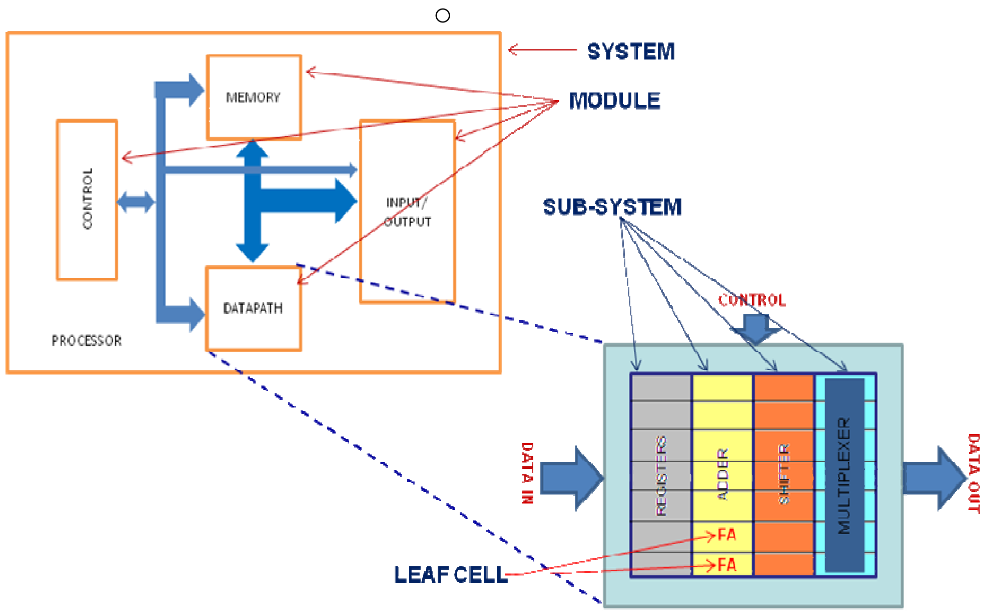
CMOS SUBSYSTEM DESIGN

CONTENTS

1.What is a *System*?
A *system* is a set of interacting or interdependent entities forming and integrate whole.
Common characteristics of a system are

- o Systems have *structure* - defined by parts and their composition
- o Systems have *behavior* – involves inputs, processing and outputs (of material, information or energy)
- o Systems have *interconnectivity* the various parts of the system functional as well as structural relationships between each other

1.1Decomposition of a *System*: A Processor



5. VLSI Design Flow

- The electronics industry has achieved a phenomenal growth –mainly due to the rapid advances in integration technologies, large scale systems design-in short due to VLSI.
- Number applications of integrated circuits in high-performance computing, telecommunications, and consumer electronics has been rising steadily.
- Current leading-edge technology trend –expected to continue with very important implications on VLSI and systems design.
- The design process, at various levels, is evolutionary in nature.
- Y-Chart (first introduced by D. Gajski) as shown in Figure1 illustrates the design flow for mast logic chips, using design activities.
- Three different axes (domains) which resemble the letter Y.
- Three major domains, namely
  Behavioral domain
  Structural domain

Geometrical domain

- Design flow starts from the algorithm that describes the behavior of target chip.

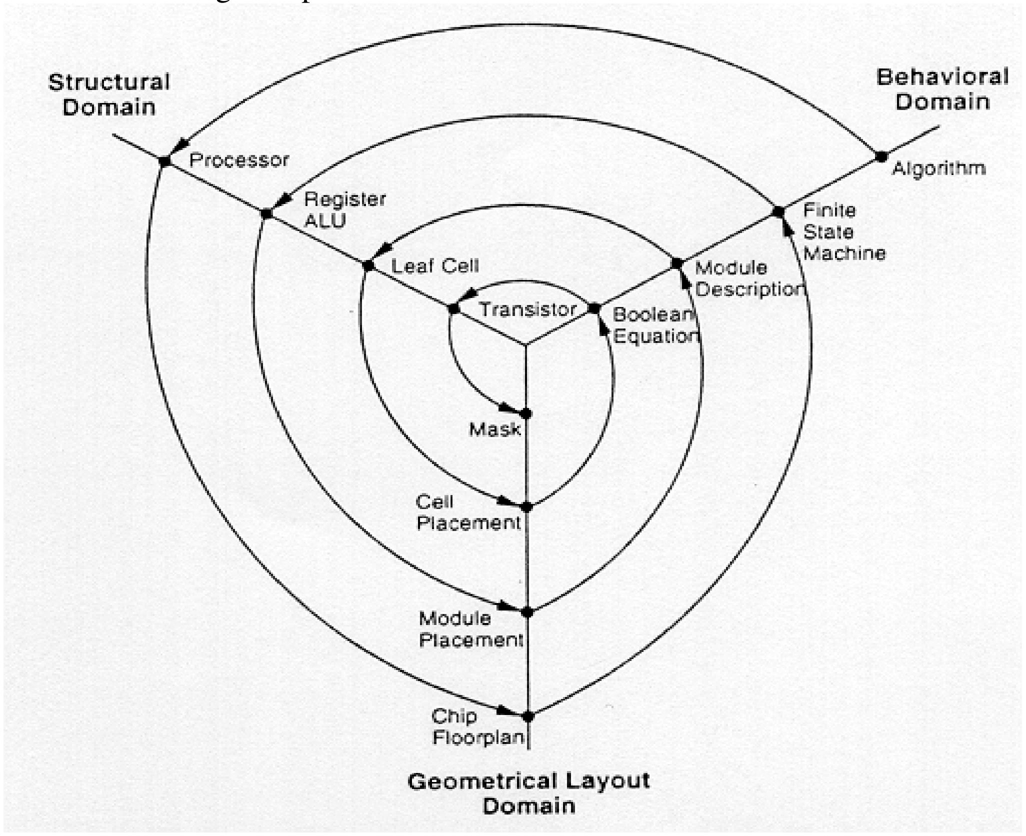


Figure 1. Typical VLSI design flow in three domains(Y-chart)

VLSI design flow, taking in to account the various representations, or abstractions of design are

Behavioural,logic,circuit and mask layout.

Verification of design plays very important role in every step during process.
Two approaches for design flow as shown in Figure 2 are

Top-down
Bottom-up

Top-down design flow- excellent design process control
In reality, both top-down and bottom-up approaches have to be combined.
Figure 3 explains the typical full custom design flow.

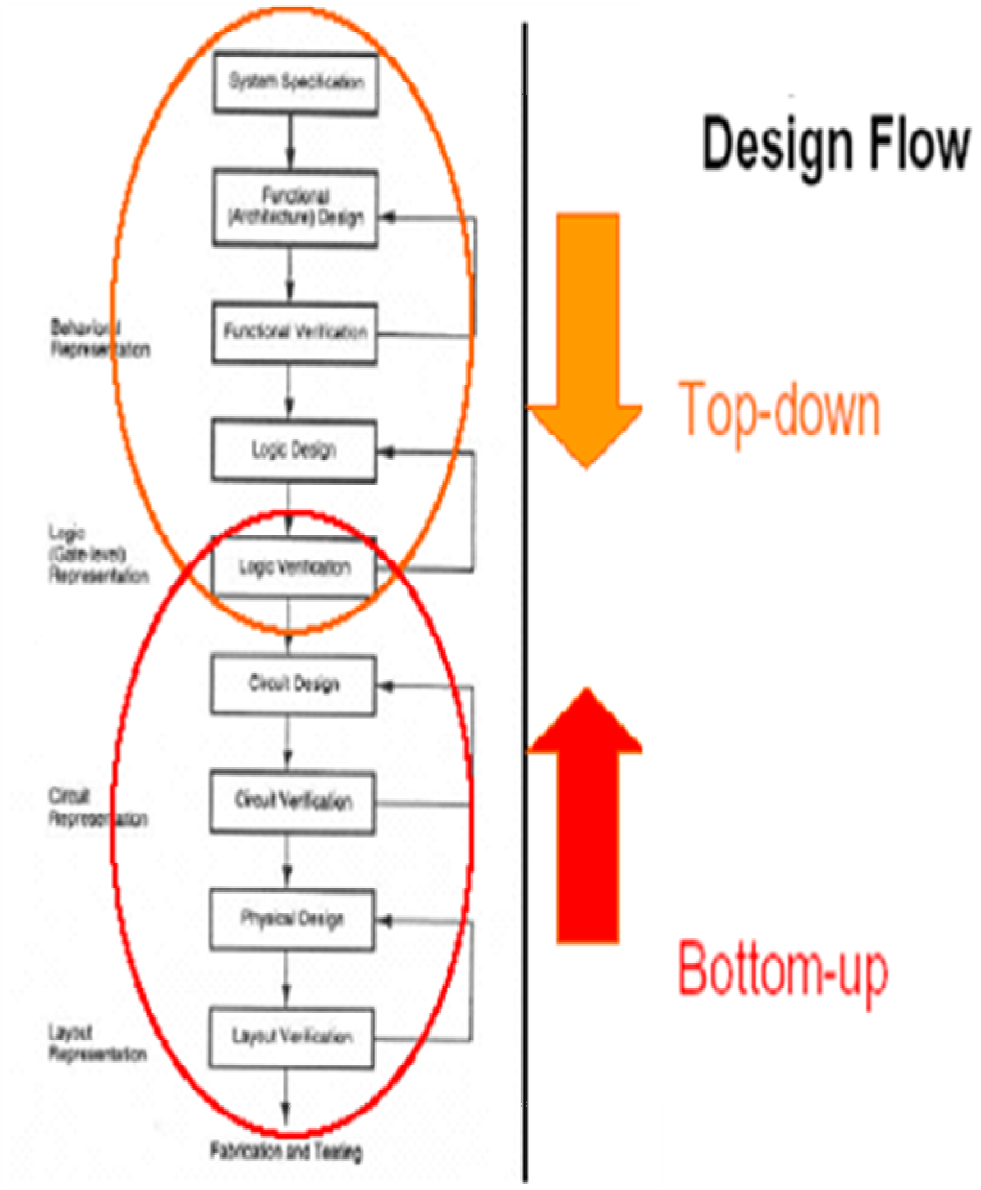


Figure 2. Typical VLSI design flow

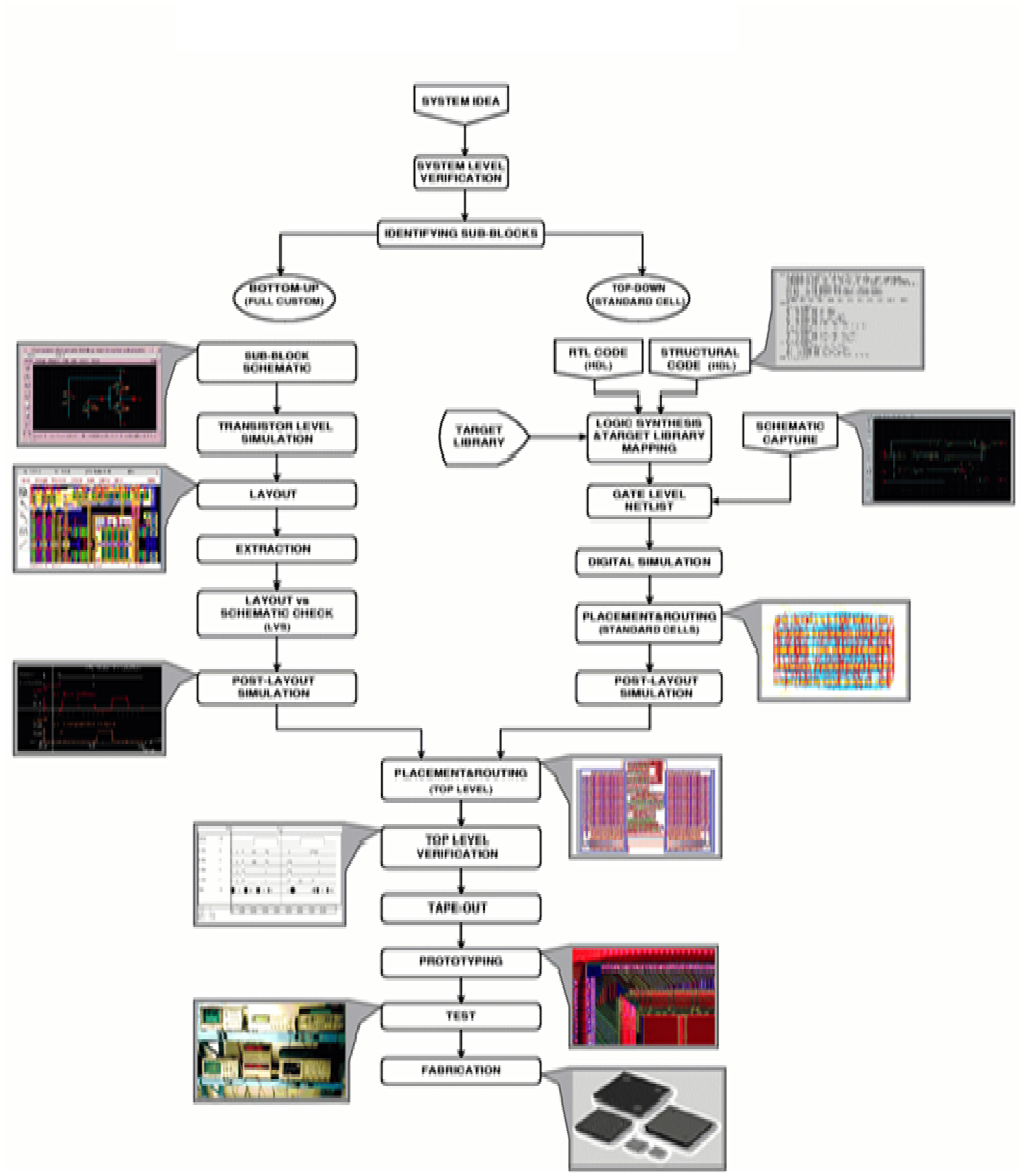
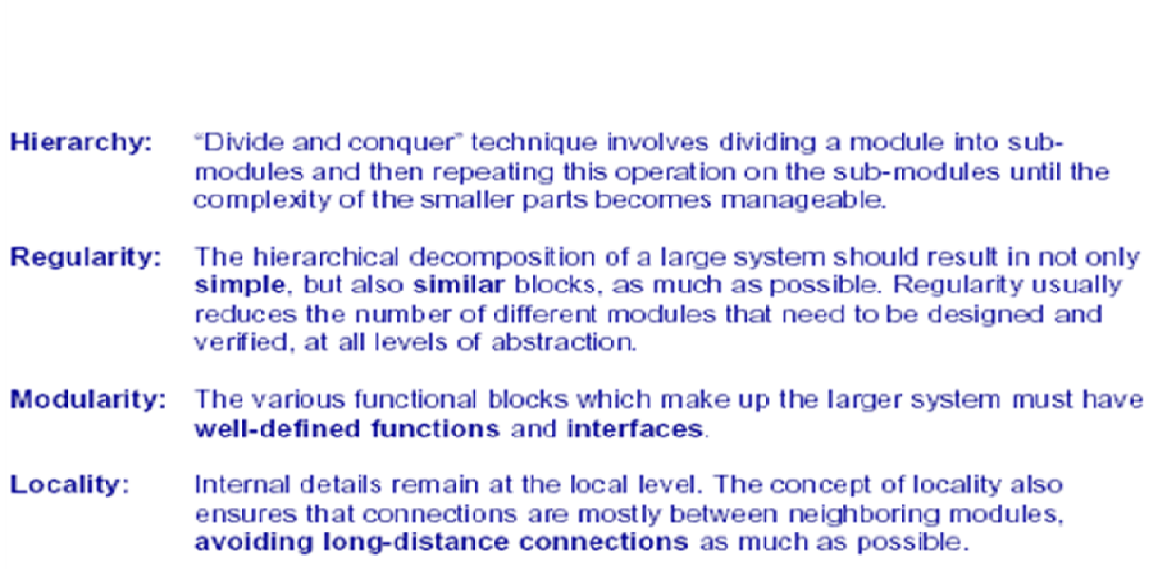


Figure 3. Typical ASIC/Custom design flow

3 Structured Design Approach

- Design methodologies and structured approaches developed with complex hardware and software.
- Regardless of the actual size of the project, basic principles of structured design- improve the prospects of success.
- Classical techniques for reducing the complexity of IC design are:
  - Hierarchy
  - Regularity
  - Modularity
  - Locality



Hierarchy:	"Divide and conquer" technique involves dividing a module into sub-modules and then repeating this operation on the sub-modules until the complexity of the smaller parts becomes manageable.
Regularity:	The hierarchical decomposition of a large system should result in not only **simple**, but also **similar** blocks, as much as possible. Regularity usually reduces the number of different modules that need to be designed and verified, at all levels of abstraction.
Modularity:	The various functional blocks which make up the larger system must have **well-defined functions** and **interfaces**.
Locality:	Internal details remain at the local level. The concept of locality also ensures that connections are mostly between neighboring modules, **avoiding long-distance connections** as much as possible.

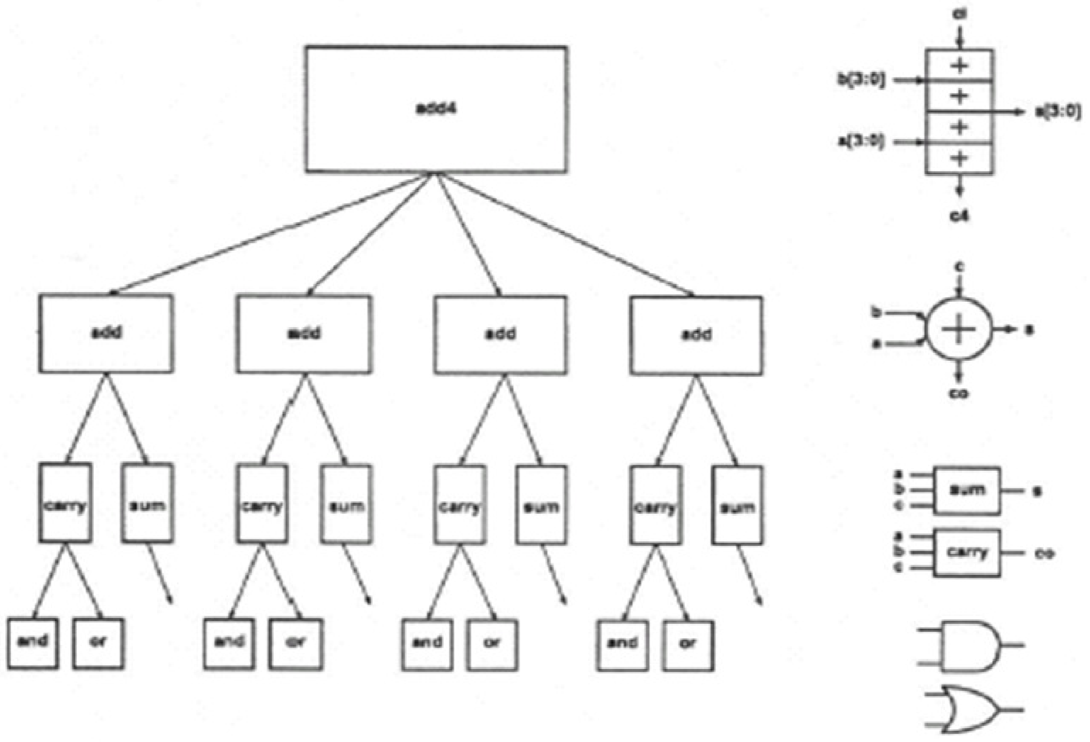


Figure 4-Structured Design Approach –Hierarchy

Regularity

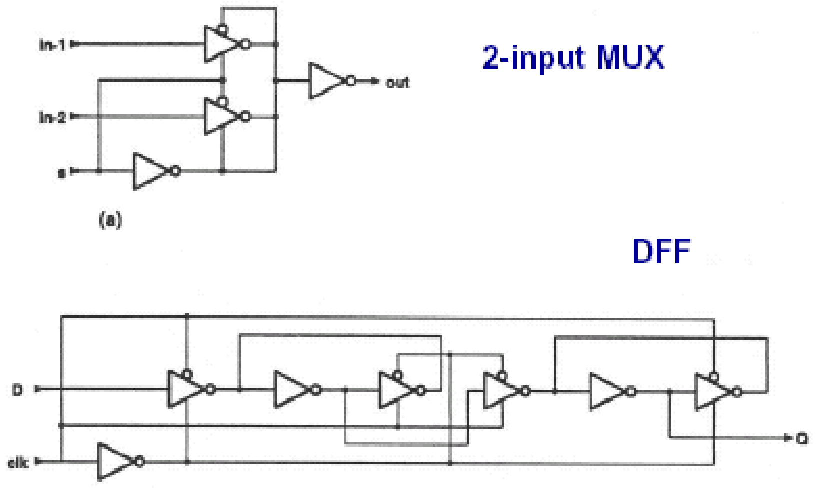


Figure5-.Structured Design Approach –Regularity

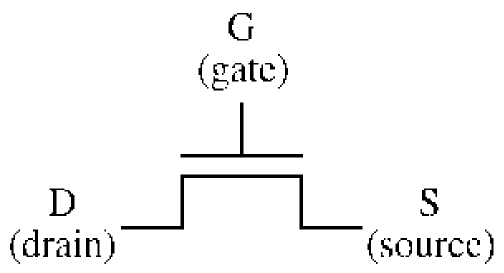
- Design of array structures consisting of identical cells.-such as parallel multiplication array.
- Exist at all levels of abstraction:
  transistor level-uniformly sized.
  logic level- identical gate structures
- 2:1 MUX, D-F/F- inverters and tri state buffers
- Library-well defined and well-characterized basic building block.
- Modularity: enables parallelization and allows plug-and-play
- Locality: Internals of each module unimportant to exterior modules and internal details remain at local level.

Figure 4 and Figure 5 illustrates these design approaches with an example.

4 Architectural issues

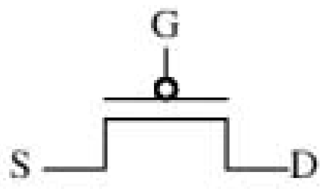
- Design time increases exponentially with increased complexity
- Define the requirements
- Partition the overall architecture into subsystems.
- Consider the communication paths
- Draw the floor plan
- Aim for regularity and modularity
- convert each cell into layout
- Carry out DRC check and simulate the performance

5. MOSFET as a Switch



*n*MOS transistor:
Closed (conducting) when
Gate = 1 (Vdd, 5V)

Open (non-conducting) when
Gate = 0 (ground, 0V)



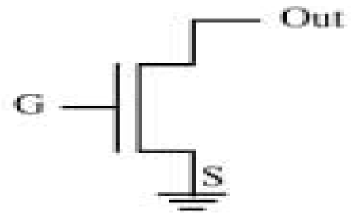
*p*MOS transistor:
Closed (conducting) when
Gate = 0 (ground, 0V)

Open (non-conducting) when
Gate = 1 (Vdd, 5V)

Note: The MOS transistor is a symmetric device. This means that the drain and source terminals are interchangeable. For a conducting *n*MOS transistor, $V_{DS} >$ 0V; for the *p*MOS transistor, $V_{DS} < 0V$ (or $V_{SD} > 0V$).

- We can view MOS transistors as electrically controlled switches
- Voltage at gate controls path from source to drain

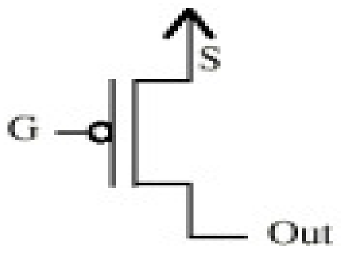
For *n*MOS switch, source is typically tied to ground and is used to *pull-down* signals:



when Gate = 1, Out = 0, (OV)

when Gate = 0, Out = Z (high impedance)

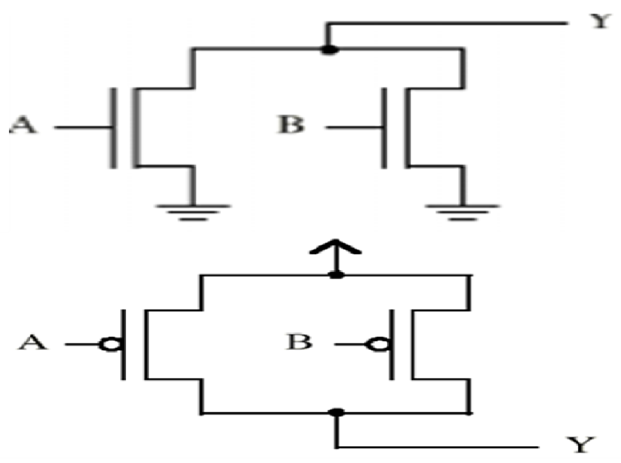
For *p*MOS switch, source is typically tied to Vdd, used to *pull* signals *up*:



when Gate = 0, Out = 1 (Vdd)

when Gate = 1, Out = Z (high impedance)

5.1 Parallel connection of Switches..



$Y = 0$, if A or $B = 1$

$\updownarrow$

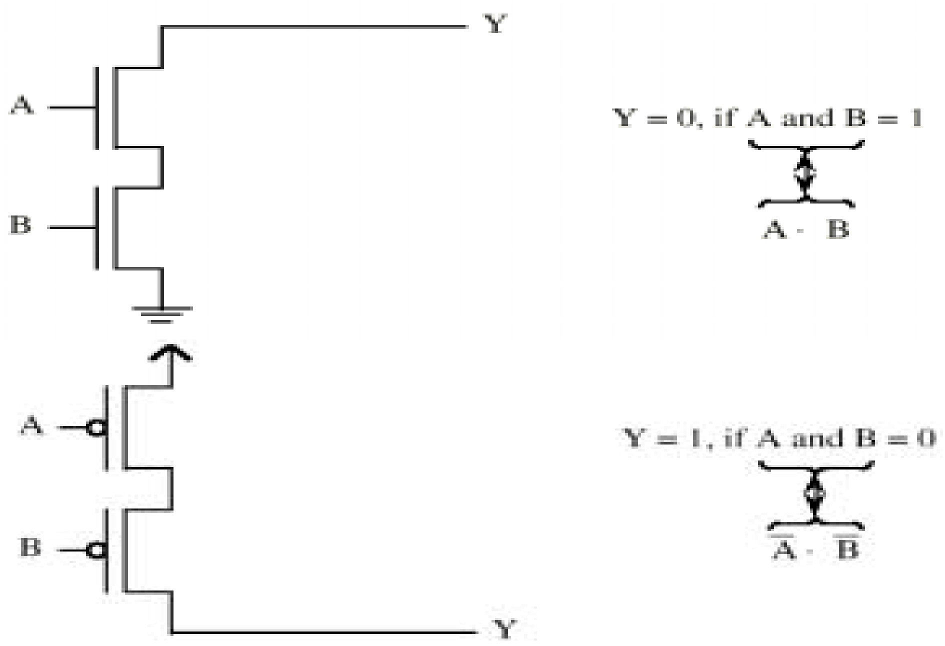
$\overline{A + B}$

$Y = 1$ if A or $B = 0$

$\updownarrow$

$\overline{A} + \overline{B}$

5.2 Series connection of Switches..



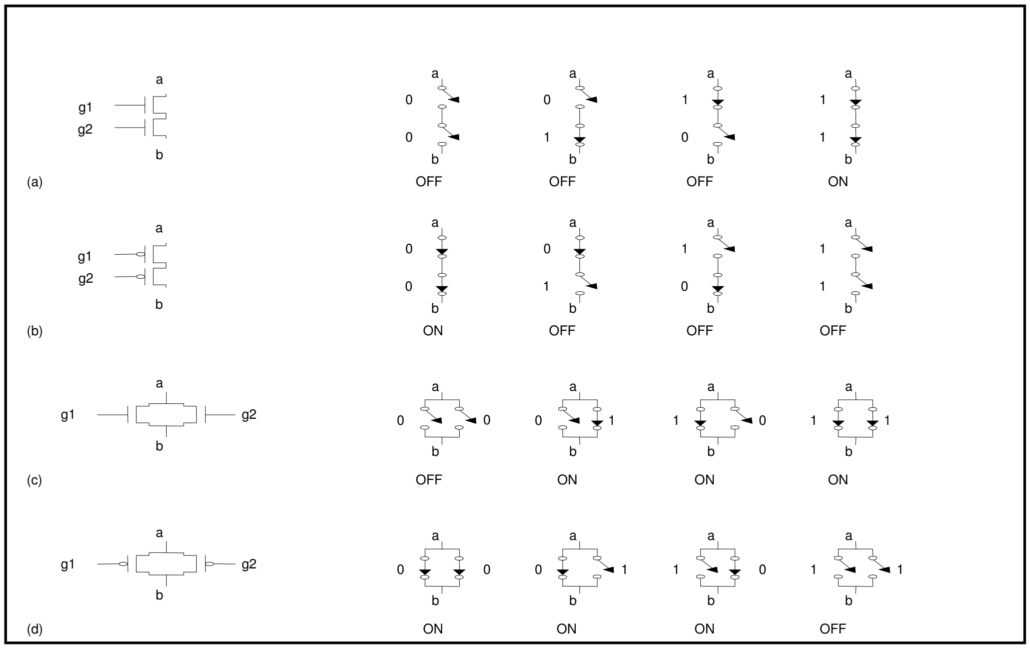
5.3 Series and parallel connection of Switches..

nMOS: 1 = ON

pMOS: 0 = ON

Series: both must be ON

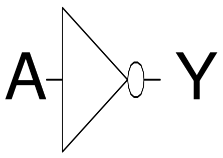
Parallel: either can be ON

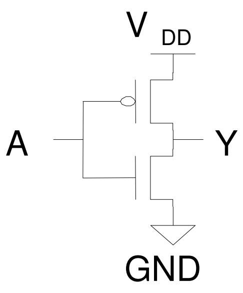


6. Circuit Families : Restoring logic

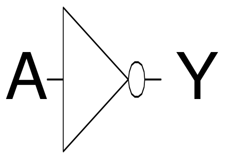
CMOS INVERTER

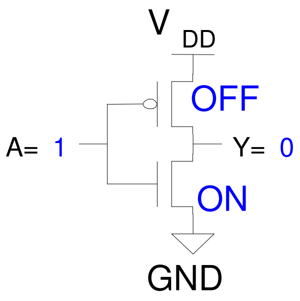
A	Y
0	
1	



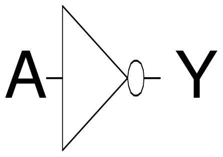


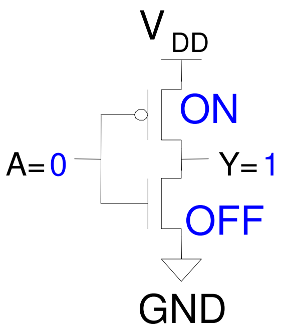
A	Y
0	
1	**0**





A	Y
0	**1**
1	0





6.1 NAND gate Design..

NAND Gate Design

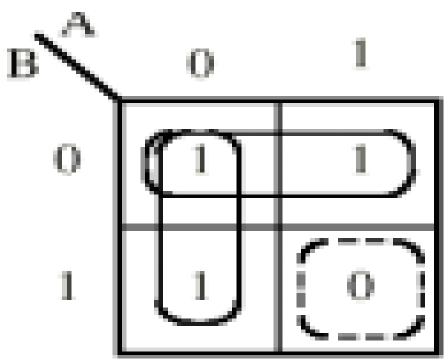
p-type transistor tree will provide "1" values of logic function

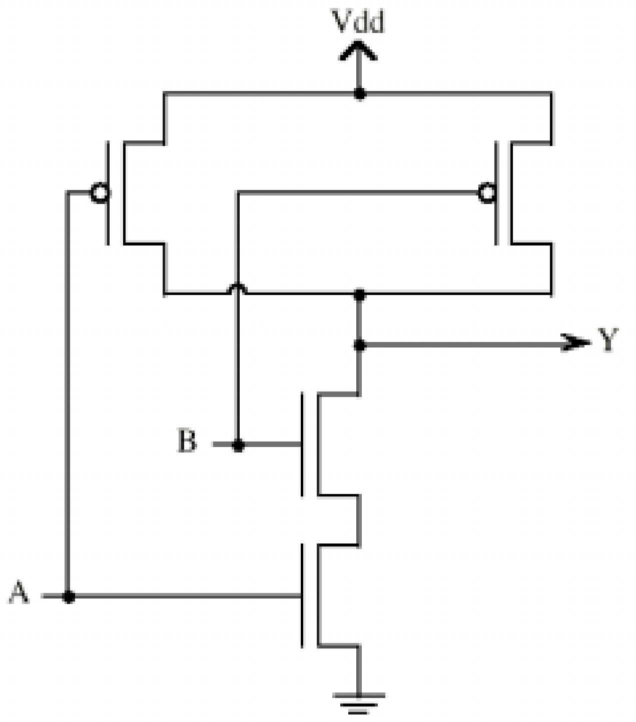
n-type transistor tree will provide "0" values of logic function

Truth Table (NAND):

AB	
00	1
01	1
10	1
11	0

K-map (NAND):

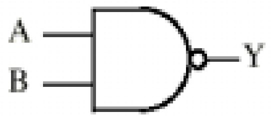




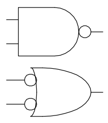
$P_{tree} = \overline{A} + \overline{B}$

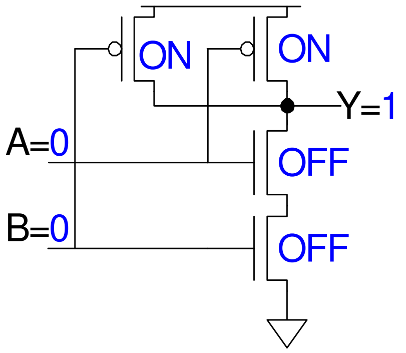
$N_{tree} = A \cdot B$

$\Leftrightarrow$

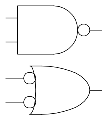


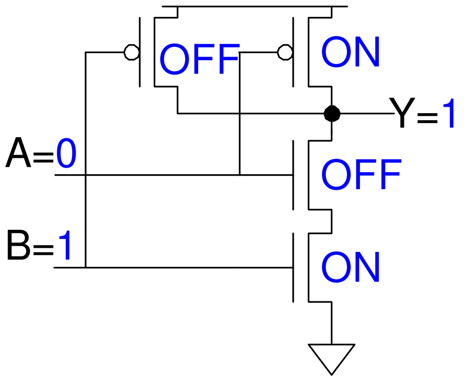
A	B	Y
0	0	1
0	1	
1	0	
1	1	



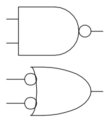


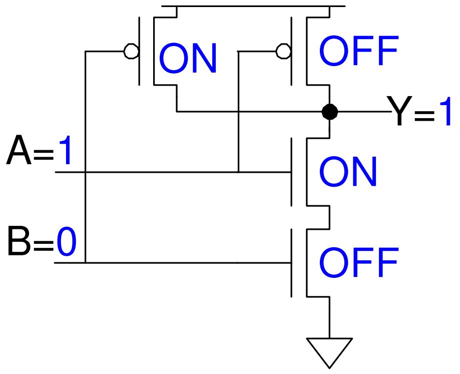
A	B	Y
0	0	1
0	1	1
1	0	
1	1	





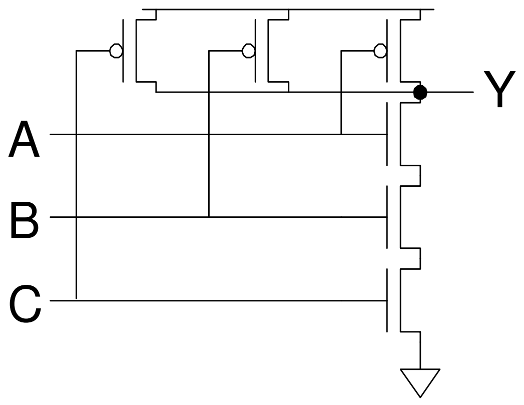
A	B	Y
0	0	1
0	1	1
1	0	1
1	1	0





NAND gate Design..

Y pulls low if ALL inputs are 1
Y pulls high if ANY input is 0



6.2 NOR gate Design..

NOR Gate Design

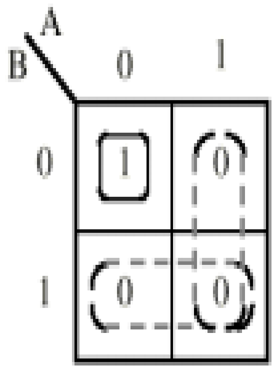
p-type transistor tree will provide "1" values of logic function

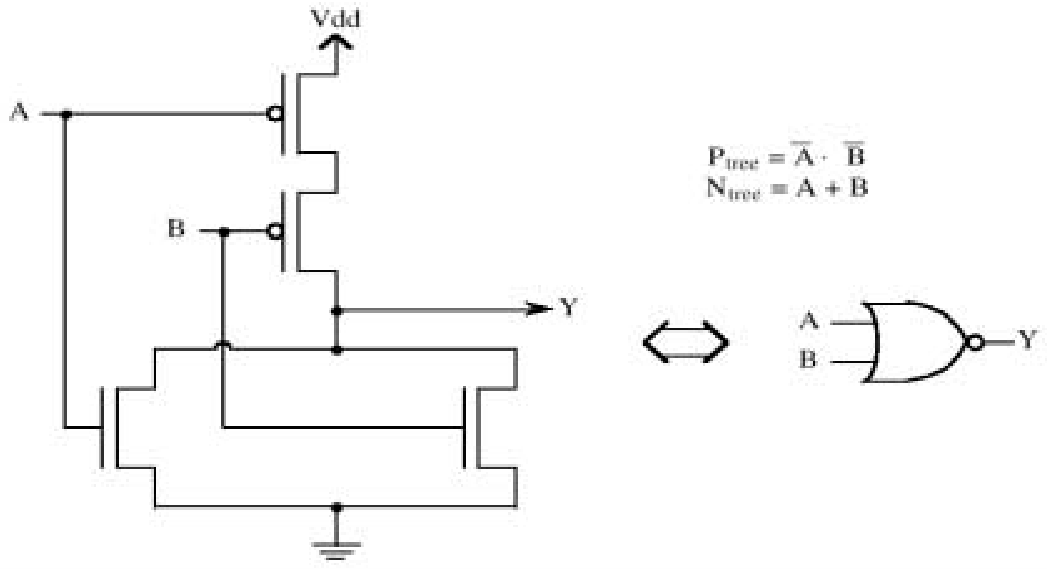
n-type transistor tree will provide "0" values of logic function

Truth Table:

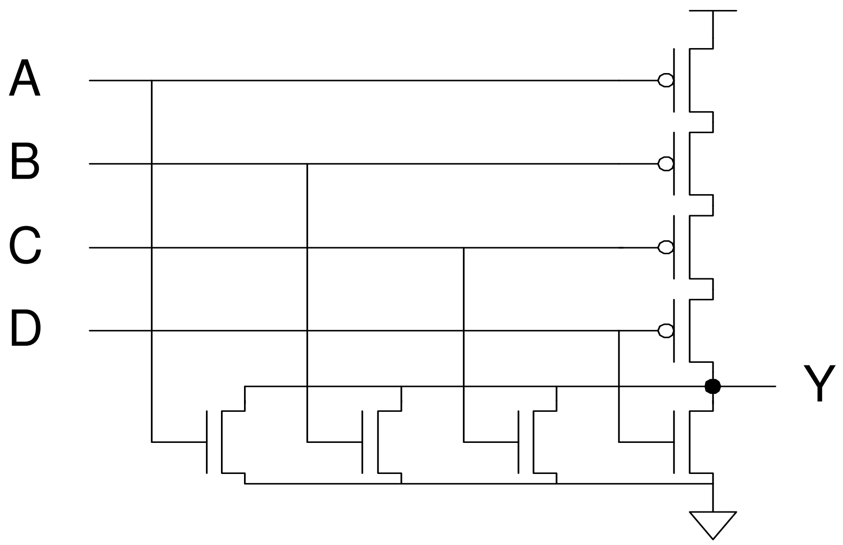
K-map:

AB	
00	1
01	0
10	0
11	0





4-input CMOS NOR gate



CMOS INVERTER

Note: Ideally there is <u>no</u> static power dissipation. When "I" is fully is *high* or fully *low*, <u>no</u> current path between Vdd and GND exists (the output is usually tied to the gate of another MOS transistor which has a very high input impedance).

Power is dissipated as "I" transistions from 0 →1 and 1→0 and a momentory current path exists between Vdd and GND. Power is also dissipated in the charging and discharging of gate capacitances.

6.3 CMOS Properties

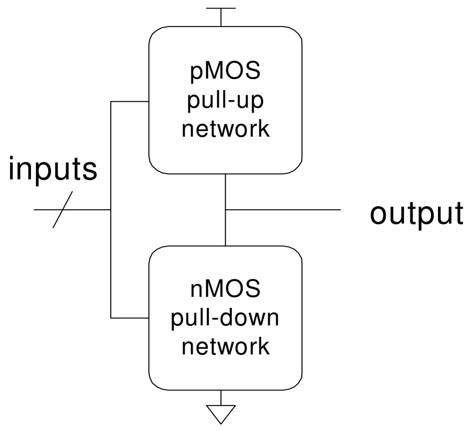
Complementary CMOS logic gates

nMOS *pull-down network*

pMOS *pull-up netw* **CMOS Properties** *ork*

a.k.a. static CMOS ,steady state is reached to 0 or 1.(no dc path from Vdd to gnd)

	Pull-up OFF	**Pull-up ON**
Pull-down OFF	**Z (float)**	**1**
Pull-down ON	**0**	**X (crowbar)**



- Complementary CMOS gates always produce 0 or 1
- Ex: NAND gate
- Series nMOS: Y=0 when both inputs are 1
- Thus Y=1 when either input is 0
- Requires parallel pMOS
- Rule of *Conduction Complements*

- Pull-up network is complement of pull-down
- Parallel -> series, series -> parallel
- Output signal strength is independent of input.-level restoring
- Restoring logic. Ouput signal strength is either V_{oh} (output high) or V_{ol}. (output low).
- Ratio less logic :output signal strength is independent of pMOS device size to nMOS size ratio.
- significant current only during the transition from one state to another and - hence power is conserved..
- Rise and fall transition times are of the same order,
- Very high levels of integration,
- High performance.

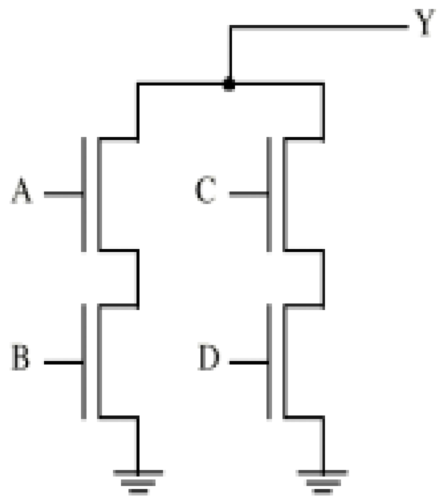
6.4 Complex gates..

$F = \overline{AB + CD} \Rightarrow N_{tree}$ will provide 0's, P_{tree} will provide 1's

0's of function F is $\overline{F}$, $\Rightarrow \overline{F} = \overline{\overline{AB + CD}} = AB + CD$

*n*MOS transistors need high true inputs, so it is desirable for all input variables to be high true, just as above.

$AB + CD \Rightarrow$



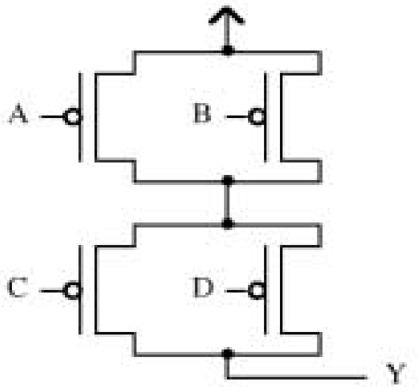
Likewise, a P_{tree} will provide 1's.

$F = \overline{AB + CD}$, need a form involving $\overline{A}$, $\overline{B}$, $\overline{C}$, $\overline{D}$

Apply DeMorgan's Theorem:

$F = \overline{AB} \cdot \overline{CD} = (\overline{A} + \overline{B}) \cdot (\overline{C} + \overline{D})$

Implementation ⇒



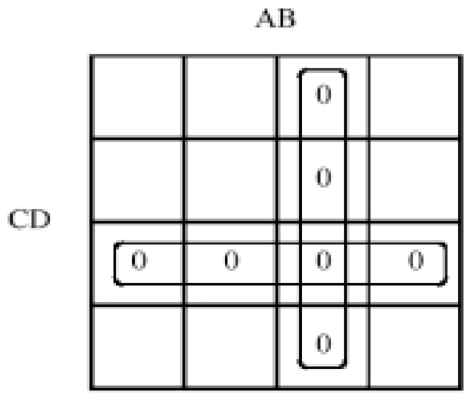
Can also use K-maps:

$F = \overline{AB + CD}$

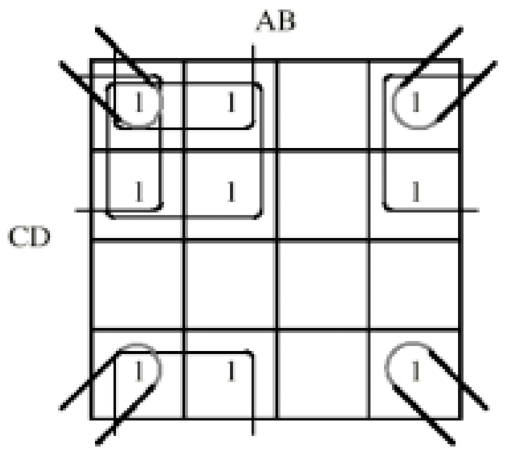
AB

CD				
	1	1	0	1
	1	1	0	1
CD	0	0	0	0
	1	1	0	1

For N_{tree}, minimize 0's; for P_{tree}, minimize 1's



$N_{tree} = AB + CD$

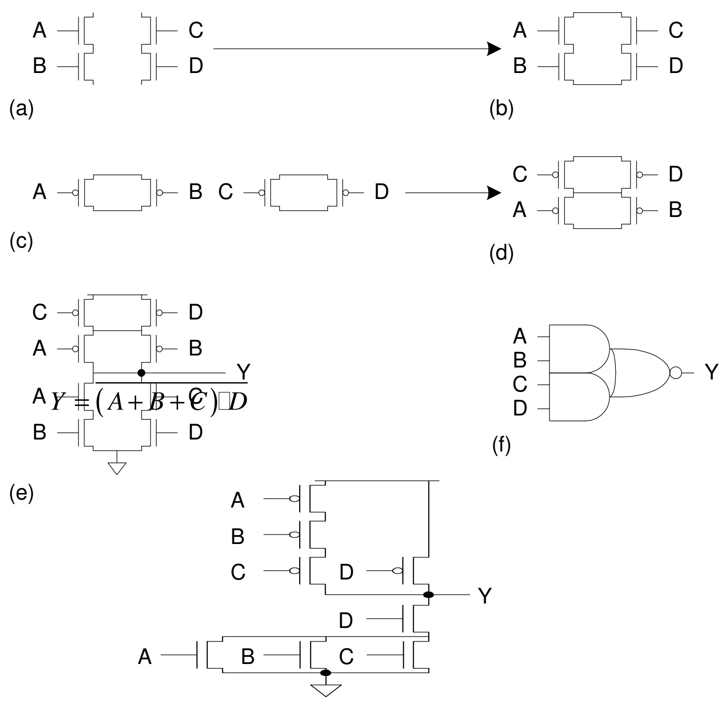


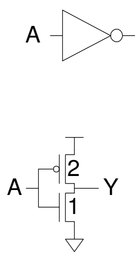
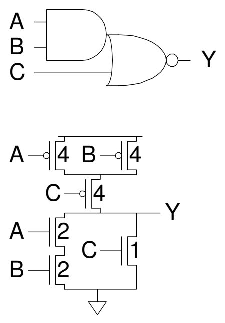
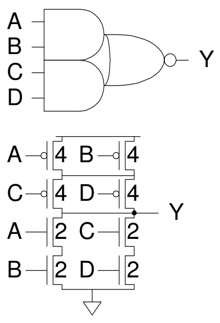
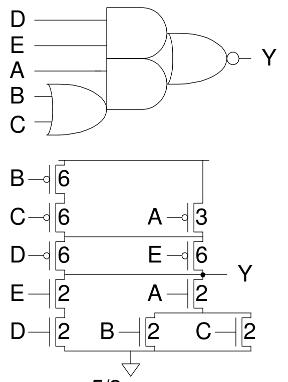
$$P_{tree} = \overline{A}\cdot\overline{C} + \overline{A}\cdot\overline{D} + \overline{B}\cdot\overline{C} + \overline{B}\cdot\overline{D}$$
$$= \overline{A}\,(\overline{C} + \overline{D}) + \overline{B}\,(\overline{C} + \overline{D})$$
$$= (\overline{A} + \overline{B}) \cdot (\overline{C} + \overline{D})$$

6.5 Complex gates AOI..

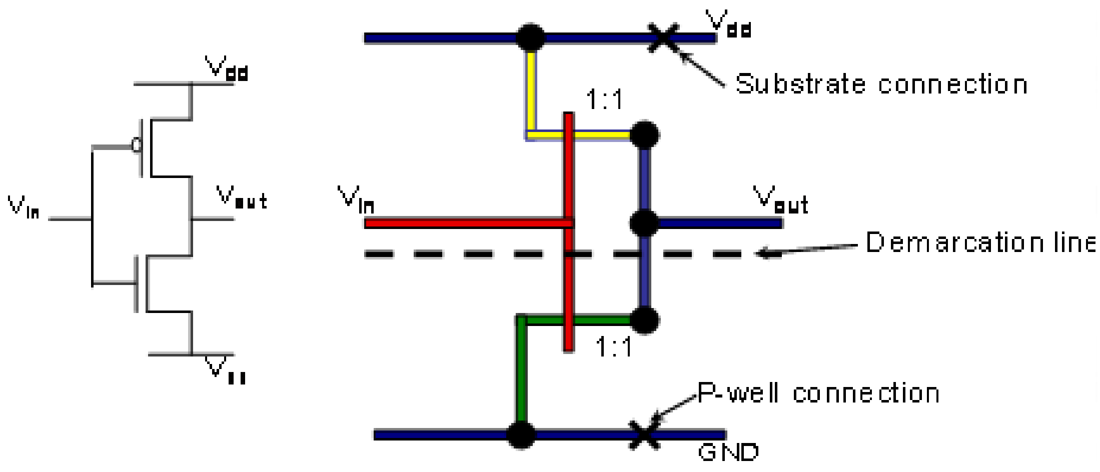
Compound gates can do any inverting function

$$Y = \overline{A\cdot B + C\cdot D} \text{ (AND-AND-OR-INVERT, AOI22)}$$

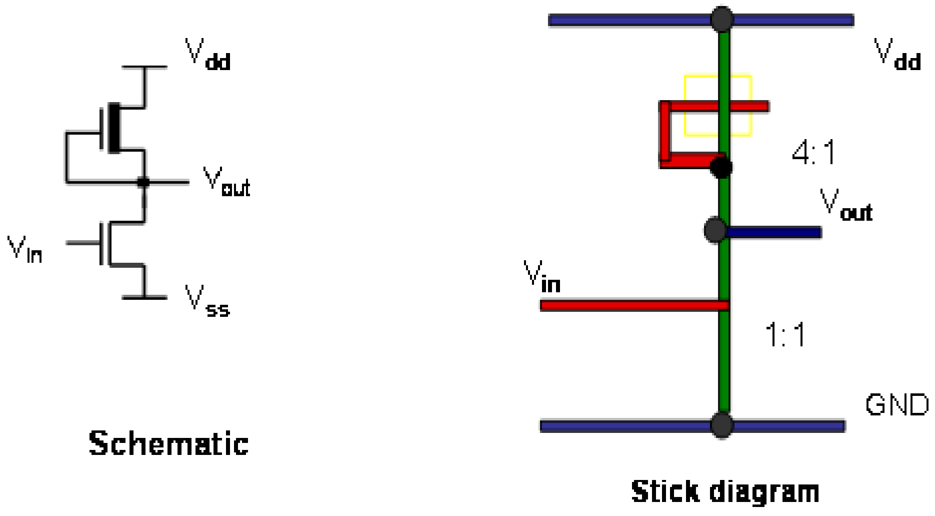


unit inverter	AOI21	AOI22	Complex AOI
$Y=\overline{A}$	$Y=\overline{A\cdot B+C}$	$Y=\overline{A\cdot B+C\cdot D}$	$Y=\overline{A\cdot(B+C)+D\cdot E}$
			
$g_A = 3/3$	$g_A = 6/3$	$g_A = 6/3$	$g_A = 5/3$
$p = 3/3$	$g_B = 6/3$	$g_B = 6/3$	$g_B = 8/3$
	$g_C = 5/3$	$g_C = 6/3$	$g_C = 8/3$
	$p = 7/3$	$g_D = 6/3$	$g_D = 8/3$
		$p = 12/3$	$g_E = 8/3$
			$p = 16/3$

6.6 Circuit Families : Restoring logic CMOS Inverter- Stick diagram



6.7 Restoring logic CMOS Variants: nMOS Inverter-stick diagram



- Basic inverter circuit: load replaced by depletion mode transistor
- With no current drawn from output, the current I_{ds} for both transistor must be same.
- For the depletion mode transistor, gate is connected to the source so it is always on and only the characteristic curve V_{gs}=0 is relevant.

- Depletion mode is called pull-up and the enhancement mode device pull-down.
- Obtain the transfer characteristics.
- As V_{in} exceeds the p.d. threshold voltage current begins to flow, V_{out} thus decreases and further increase will cause p.d transistor to come out of saturation and become resistive.
- p.u transistor is initially resistive as the p.d is turned on.
- Point at which $V_{out} = V_{in}$ is denoted as V_{inv}
- Can be shifted by variation of the ratio of pull-up to pull-down resistances $–Z_{p.u} / Z_{p.d}$
- Z- ratio of channel length to width for each transistor

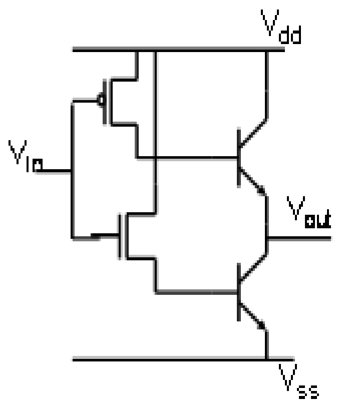
For 8:1 nMOS Inverter

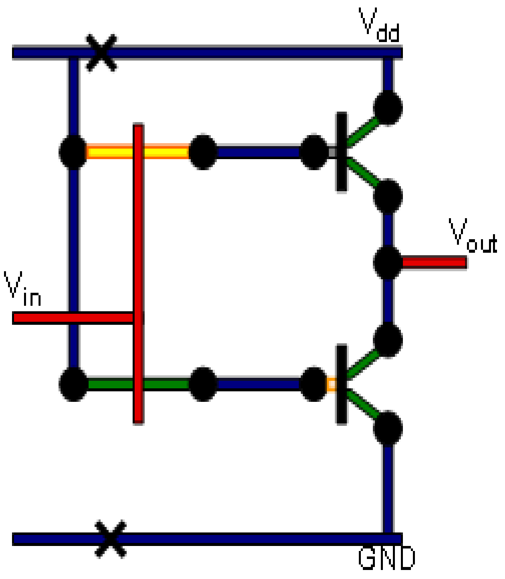
$Z_{p.u.} = L_{p.u.} / W_{p.u} = 8$
$R_{p.u} = Z_{p.u.} * R_s = 80K$
similarly
$R_{p.d} = Z_{p.d} * R_s = 10K$
Power dissipation(on) $P_d = V^2 / R_{p.u} + R_{p.d} = 0.28mV$
Input capacitance = 1 C_g

For 4:1 nMOS Inverter

$Z_{p.u.} = L_{p.u.} / W_{p.u} = 4$
$R_{p.u} = Z_{p.u.} * R_s = 40K$
similarly
$R_{p.d} = Z_{p.d} * R_s = 5K$
Power dissipation(on) $P_d = V^2 / R_{p.u} + R_{p.d} = 0.56mV$
Input capacitance = $2C_g$

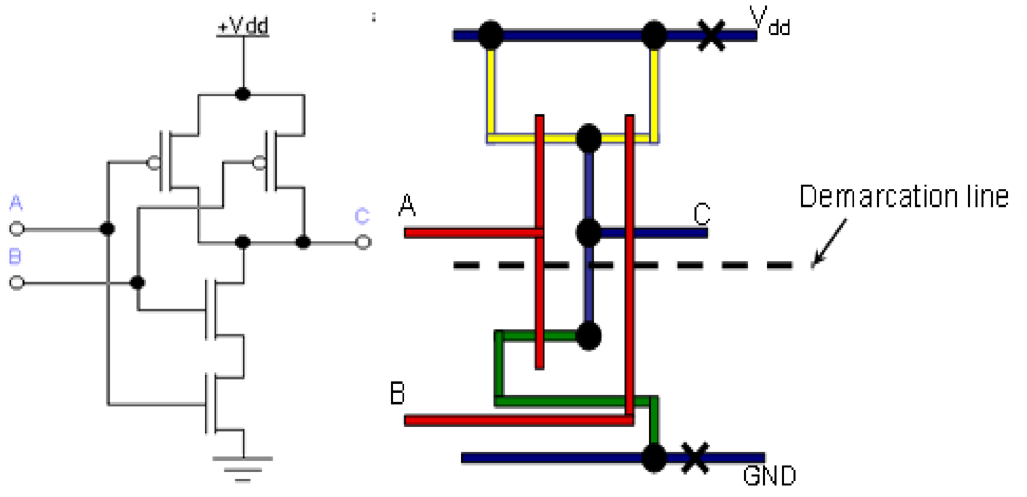
6.8Restoring logic CMOS Variants: BiCMOS Inverter-stick diagram



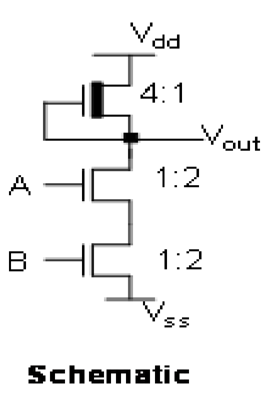


- A known deficiency of MOS technology is its limited load driving capabilities (due to limited current sourcing and sinking abilities of pMOS and nMOS transistors.)
- Output logic levels good-close to rail voltages
- High input impedance
- Low output impedance
- High drive capability but occupies a relatively small area.
- High noise margin
- Bipolar transistors have
    - higher gain
    - better noise characteristics
    - better high frequency characteristics
- BiCMOS gates can be an efficient way of speeding up VLSI circuits
- CMOS fabrication process can be extended for BiCMOS
- Example Applications
  CMOS- Logic
  BiCMOS- I/O and driver circuits
  ECL- critical high speed parts of the system

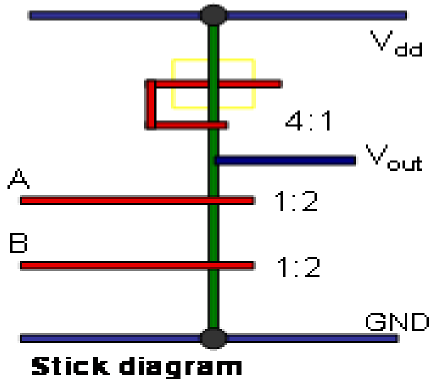
6.9 Circuit Families : Restoring logic CMOS NAND gate



6.10 Restoring logic CMOS Variants:_nMOS NAND gate

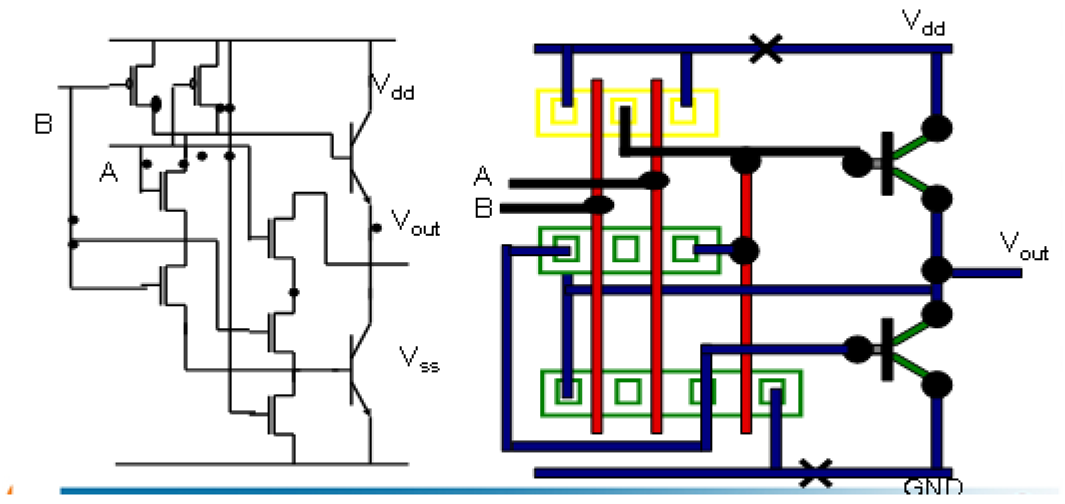


Schematic



Stick diagram

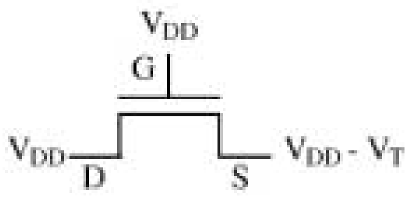
6.11 Restoring logic CMOS Variants: BiCMOS NAND gate



- For nMOS Nand-gate, the ratio between pull-up and sum of all pull-downs must be 4:1.
- nMOS Nand-gate area requirements are considerably greater than corresponding nMOS inverter
- nMOS Nand-gate delay is equal to number of input times inverter delay.
- Hence nMOS Nand-gates are used very rarely
- CMOS Nand-gate has no such restrictions
- BiCMOS gate is more complex and has larger fan-out.

7.Circuit Families :Switch logic: Pass Transistor

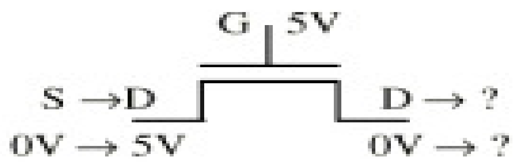
Why? *n*MOS switches cannot pass a logic "1" without a threshold voltage (V_T) drop.



where $V_T = 0.7V$ to $1.0V$ (i.e., threshold voltage will vary)

output voltage = 4.3V to 4.0V, a *weak* "1"

The *n*MOS transistor will stop conducting if $V_{GS} < V_T$. Let $V_T = 0.7V$,

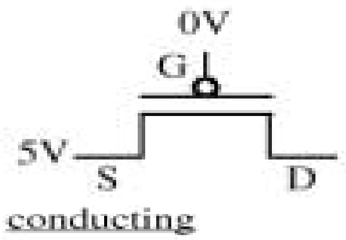


As source goes from 0V → 5V, V_{GS} goes from 5V → 0V.

When $V_S > 4.3V$, then $V_{GS} < V_T$, so switch stops conducting.

V_D left at 5V - V_T = 5V - 0.7V = 4.3V or Vdd - V_T.

For *p*MOS transistor, V_T is <u>negative</u>.



$V_{Tp} = -0.7V$

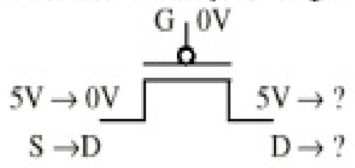
$V_{GS} = 0V - 5V = -5V$

$V_{GS} < V_{Tp}$ or $|V_{GS}| > |V_{Tp}|$

$-5V < -0.7V$ $\quad$ $5V > 0.7V$

How will *p*MOS pass a "0"?

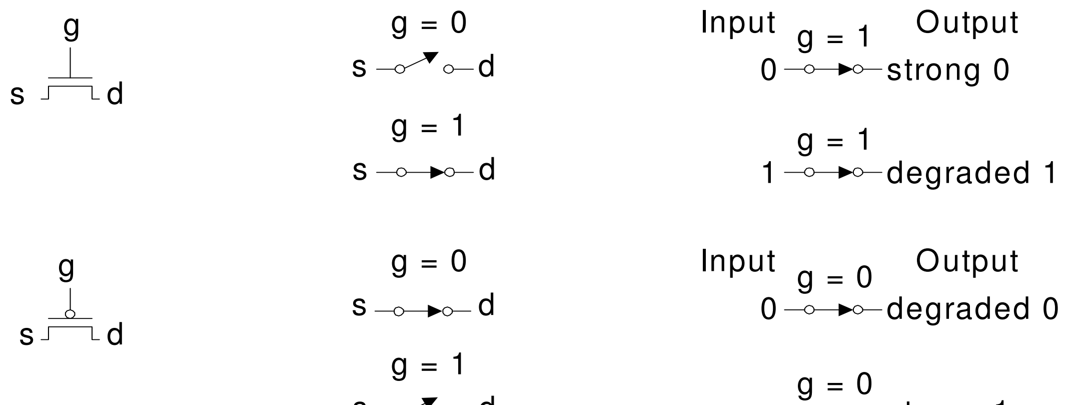
When $|V_{GS}| < |V_{Tp}|$, stop conducting



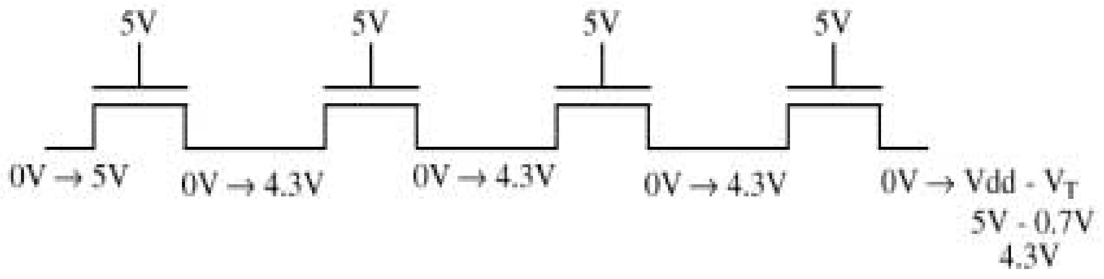
So when $|V_{GS}| < |-0.7V|$, V_D will go from 5V → **0.7V**,

a *weak* "<u>0</u>"

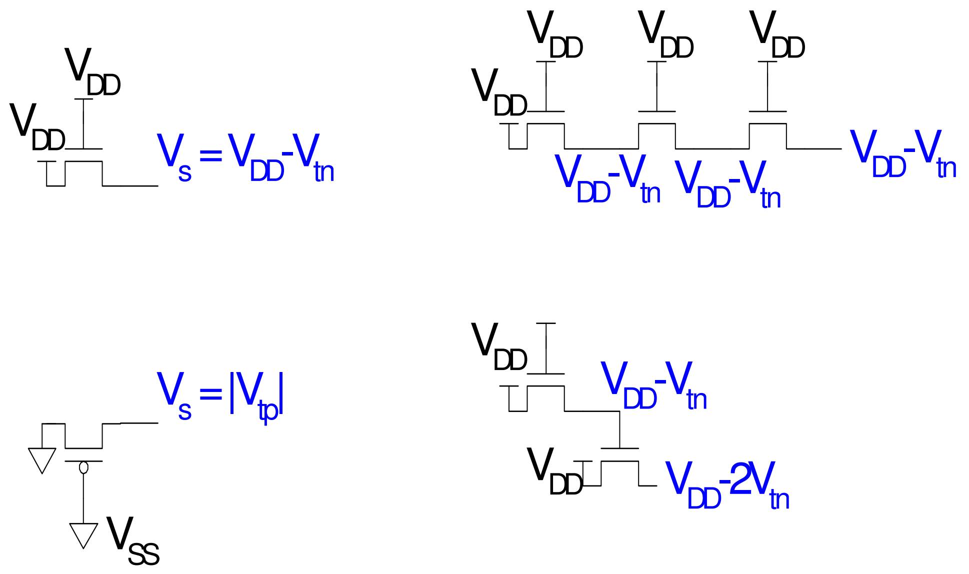
7.1 Switch logic: Pass Transistor



7.1 Switch logic: Pass Transistor-nMOS in series



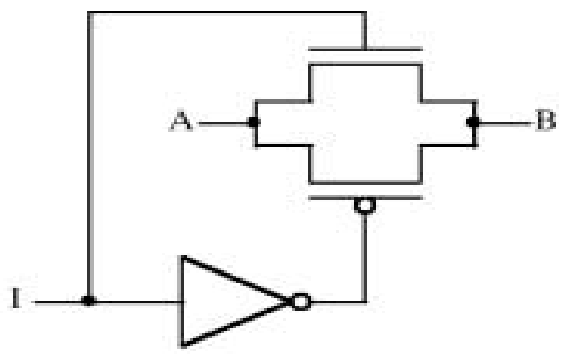
Only one threshold voltage drop across series of *n*MOS transistors



7.2 :Switch logic: Transmission gates

How are both a strong "1" and a strong "0" passed?

Transmission gate pass transistor configuration



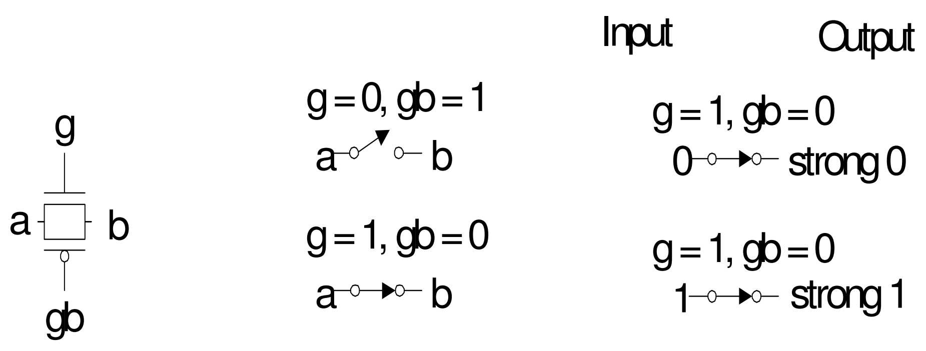
When I = 1,

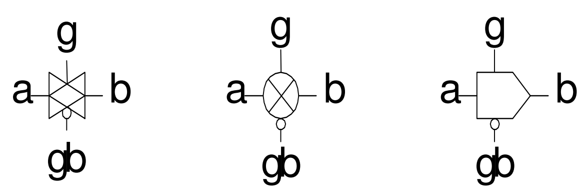
B = strong 1, if A = 1;

B = strong 0, if A = 0

When I = 0, non-conducting

Pass transistors produce degraded outputs
Transmission gates pass both 0 and 1 well

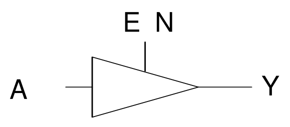




8 Structured Design-Tristate

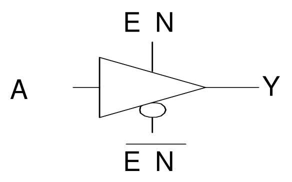
- *Tristate buffer* produces Z when not enabled

EN	A	Y
0	0	
0	1	
1	0	
1	1	



Tristate buffer produces Z when not enabled

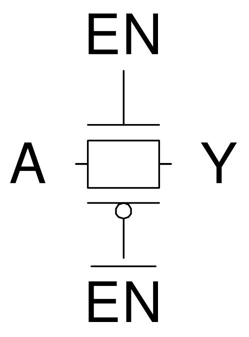
EN	A	Y
0	0	Z
0	1	Z
1	0	0
1	1	1



8.1 Structured Design-Nonrestoring Tristate

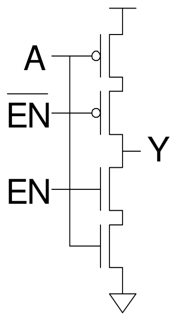
- Transmission gate acts as tristate buffer
  - Only two transistors
  - But *nonrestoring*
    - Noise on A is passed on to Y

  + No V_t drop
  - Requires inverted clock

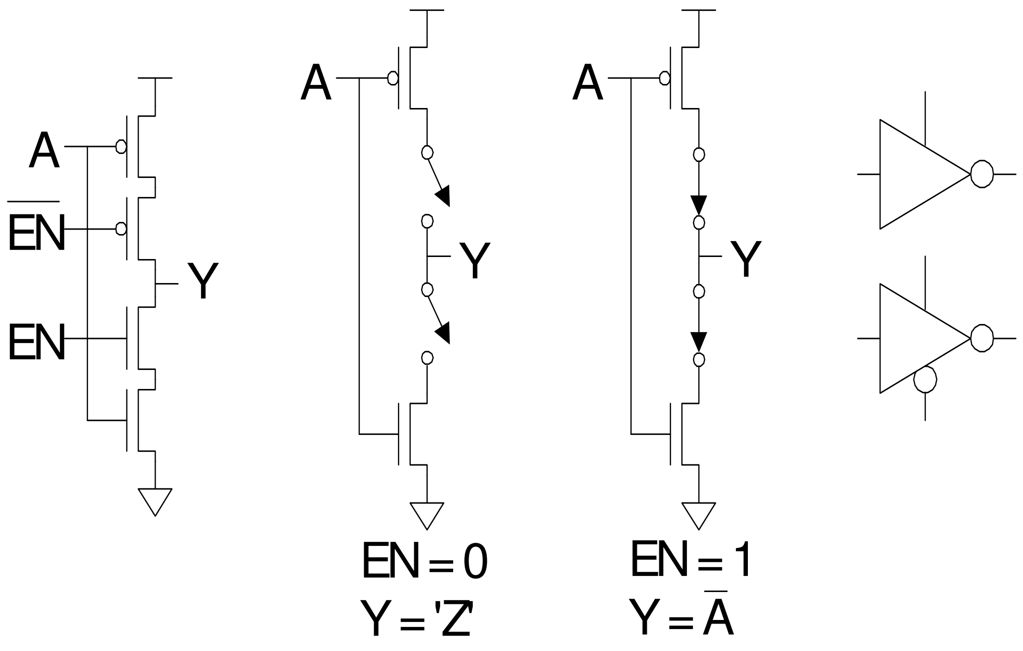


8.3 Structured Design-Tristate Inverter

- Tristate inverter produces restored output
  - Violates conduction complement rule
  - Because we want a Z output



- **Tristate inverter produces restored output**
  - **Violates conduction complement rule**
  - **Because we want a Z output**

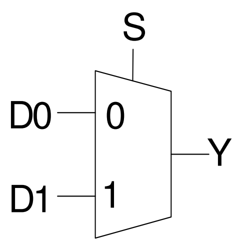


8.4 Structured Design-Multiplexers

- **2:1 *multiplexer* chooses between two inputs**

S	D1	D0	Y
0	X	0	0
0	X	1	1
1	0	X	0
1	1	X	1

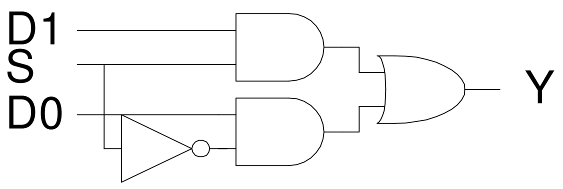
S	D1	D0	Y
0	X	0	
0	X	1	
1	0	X	
1	1	X	

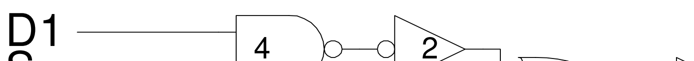


8. 5 Structured Design-Mux Design.. Gate-Level

$$Y = SD_1 + \overline{S}D_0 \text{ (too many transistors)}$$

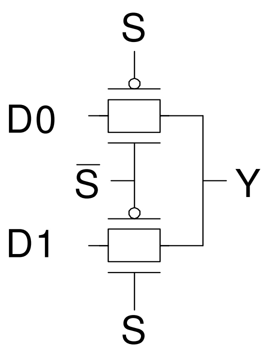
- **How many transistors are needed?**
- **How many transistors are needed? 20**





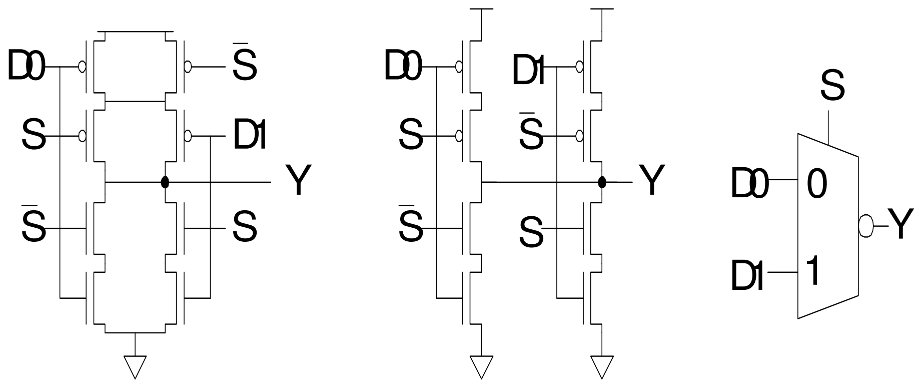
8.6 Structured Design-Mux Design-Transmission Gate

- Nonrestoring mux uses two transmission gates
  - Only 4 transistors



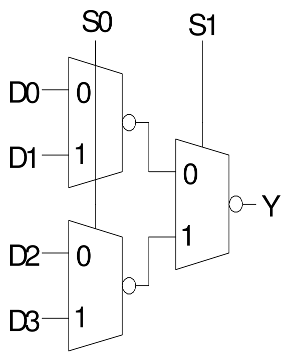
Inverting Mux

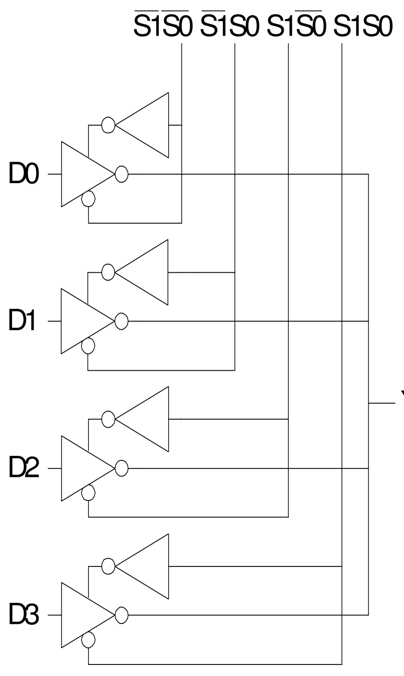
- Inverting multiplexer
  - Use compound AOI22
  - Or pair of tristate inverters
- **Noninverting multiplexer adds an inverter**



8.7 Design-4:1 Multiplexer

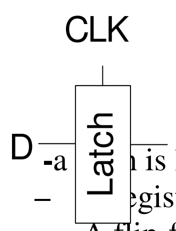
- 4:1 mux chooses one of 4 inputs using two selects
  Two levels of 2:1 muxes
  Or four tristates

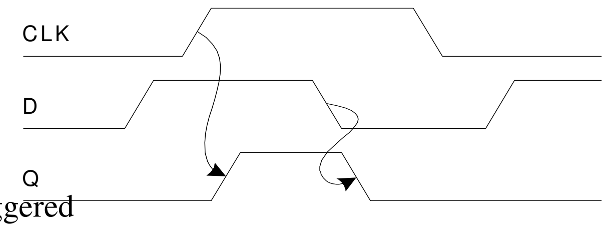




9 Structured Design-D Latch

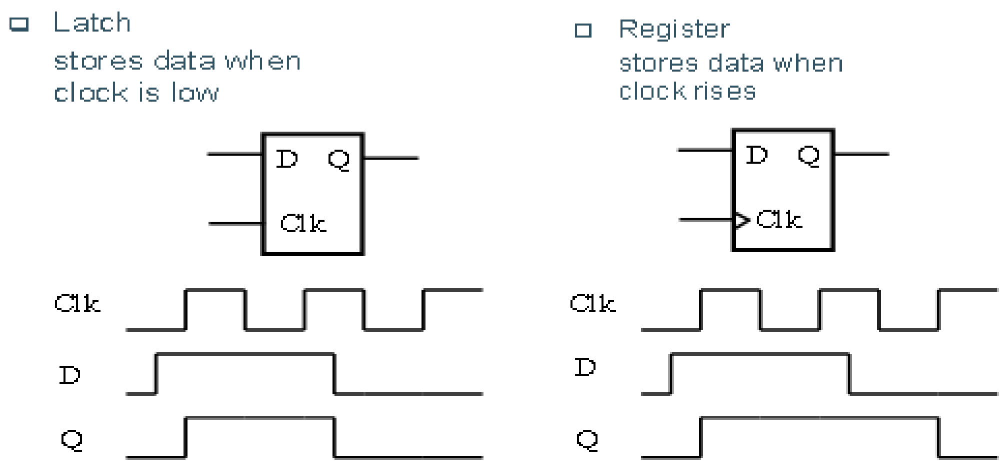
- When CLK = 1, latch is *transparent*
  - D flows through to Q like a buffer
- When CLK = 0, the latch is *opaque*
  - Q holds its old value independent of D
- a.k.a. *transparent latch* or *level-sensitive latch*





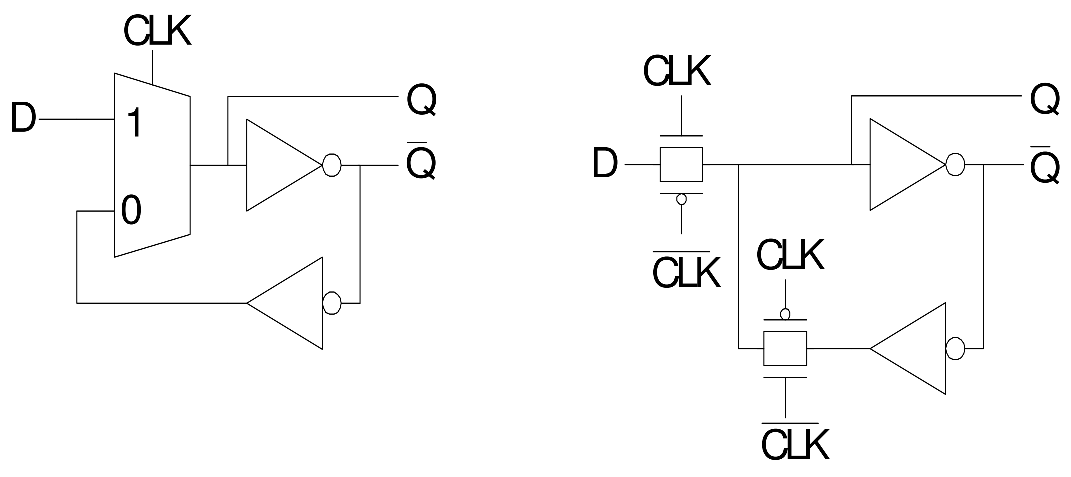
-a n is level sensitive
– egister is edge-triggered
– A flip-flop is a bi-stable element

–

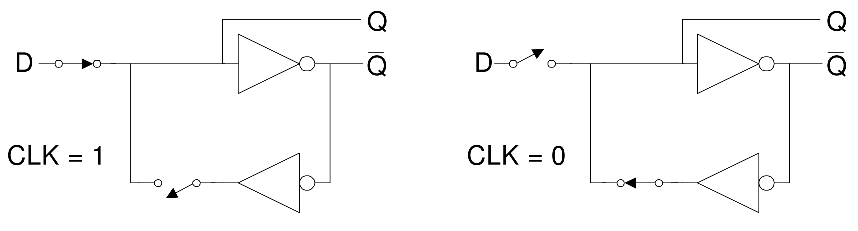


9.1 D Latch Design

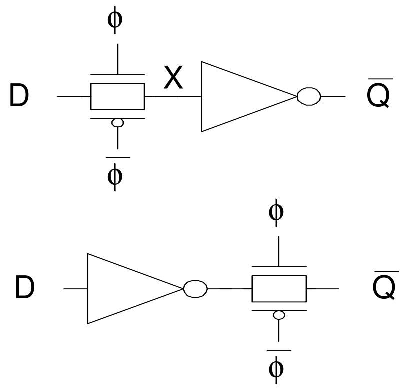
- Multiplexer chooses D or old Q



9.2 D Latch Operation



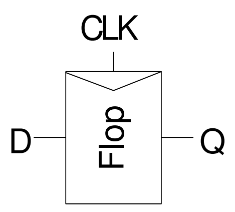
Structured Design-Latch Design

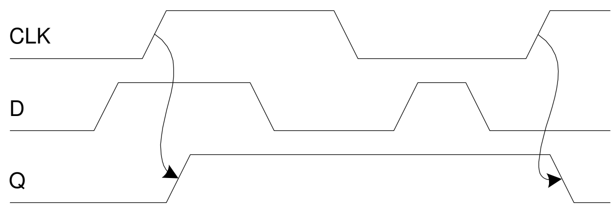


- Inverting buffer
  Restoring
  No backdriving
  Fixes either
  Output noise sensitivity
  Or diffusion input
  Inverted output

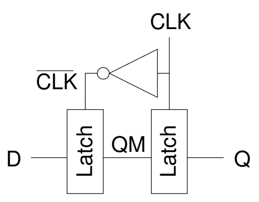
9.3 Structured Design-D Flip-flop

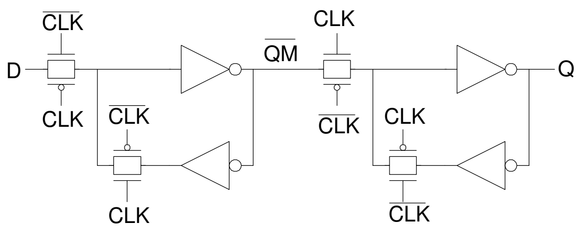
- When CLK rises, D is copied to Q
- At all other times, Q holds its value
- a.k.a. *positive edge-triggered flip-flop*, *master-slave flip-flop*



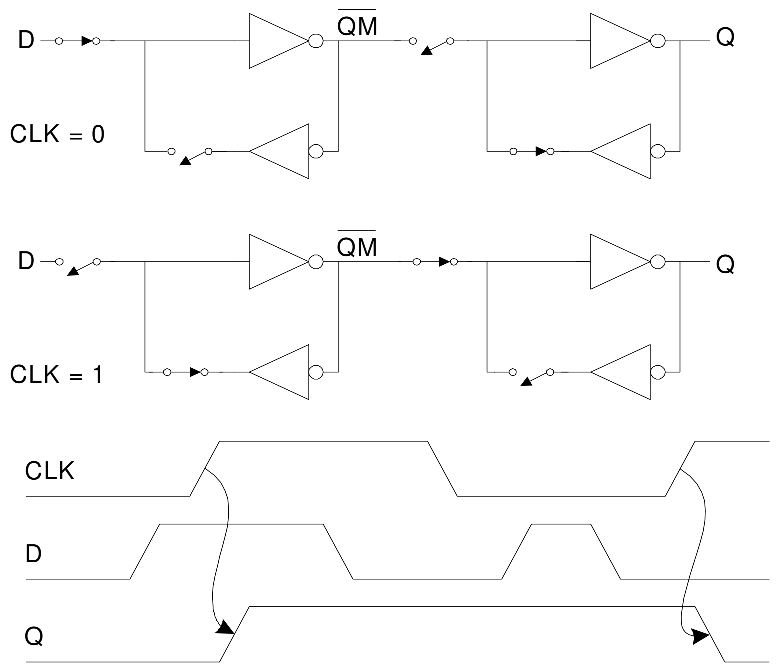


- Structured Design-D Flip-flop Design
- Built from master and slave D latches



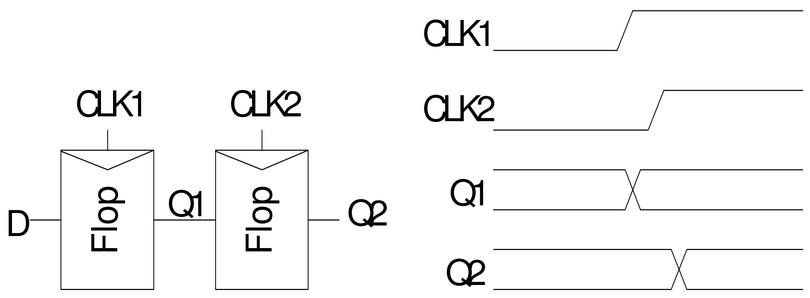


9.4 D Flip-flop Operation



9.5 Race Condition

- Back-to-back flops can malfunction from clock skew
  - Second flip-flop fires late
  - Sees first flip-flop change and captures its result
  - Called *hold-time failure* or *race condition*


CLK1
CLK2
D
Flop
Q1
Flop
Q2
CLK1
CLK2
Q1
Q2