Lec 1-2

Introduction to VLSI

1. What is circuit integration? [Marks: 1]

Circuit integration is a technique that allows to build system with many more transistors allowing much more computing power to be applied to solve a problem.

2. What is Integrated Circuit (IC)?

IC is a device built with lots of transistors for allowing much more computing power to be applied to sol problem.

3. What are the advantages / importances of VLSI?

1. Integration improves the design in the following ways
   a. Ckt contains lower parasitics, which results i higher speed.
   b. Lower power consumption.
   c. Physically smaller ckt.
2. Integration reduces manufacturing cost — (almost) manual assembly.

3. The integration of a large number of functions on a single chip usually provides:

- Less area/volume and therefore, compactness
- Less power consumption
- Less testing requirement at system level
- Higher reliability, mainly due to on-chip interconnects.
- Higher speed, due to significantly reduced interconnection length
- Significant cost savings.

4. What are the technologies used in VLSI?

1. TTL (Transistor-Transistor Logic)
2. ECL (Emitter Couple logic)
3. MOS (Metal Oxide Semi-conductor)
4. Boolean Logic

5. What are the cost factors of IC?

1. For large-volume ICs:
   a. Packaging is the largest cost.
   b. Testing is the second-largest cost.
2. For low-volume ICs, design costs may swamp all manufacturing costs.

6. Write down Moore's Law with its importance. [2007, Marks: 2]
ALSO, Explain Moore's Law in terms of Intel Microprocessors. [In course 05-06, Marks: 2]

Moore's Law:

The number of transistors per chip would grow exponentially (doubled every 18 months).

Importance of Moore's Law:*

The capabilities of many digital electronic devices are strongly linked to Moore's law: processing speed, memory capacity etc. All of these are improving at (roughly) exponential rates. This has dramatically increased the usefulness of digital electronics in nearly every segment of the world economy. Moore's law precisely describes a driving force of technological and social change in the late 20th and 21st centuries.

Moore's Law applied in Intel Microprocesors:

The number of transistors in Intel microprocessors of several generations starting from the Intel 8086 have grown exponentially almost every 18 months, which is a classic example of Moore's Law.

---

* Taken from Wikipedia.

7. What are the challenges of VLSI design? [2008, 2006. Marks: 1]

1. Multiple levels of abstraction — transistors to CPUs.
2. Multiple and conflicting ~~costs~~ constraints — low costs and high performance are often at odds.
3. Short design time — late products are often irrelevant.

8. Explain the term "multiple and conflicting costs".

[In-course 05-06. Marks: 1]

There are two cost measures of VLSI chip:

1. The speed at which the chip runs — two architectures which execute the same function (e.g. multiplication) may run at very different speeds.
2. Chip area — the cost of manufacturing a chip is exponentially related to its area.

Actual answer for 1 mark starts from here: ✓ The mentioned term means that if ~~must~~ multiple cost ~~to~~ criteria — such as area and speed requirements — must be satisfied, many design decisions will improve one cost metric at the expense of the other.

3. What do you understand by top-down design and bottom-up design?

A top-down approach ~~essentially~~ is the breaking down of a system to gain insight into its compositional subsystems (i.e. functional detail).

A bottom-up approach is the piecing together of systems to give rise to grander systems. Bottom-up design creates abstractions from low-level behavior.

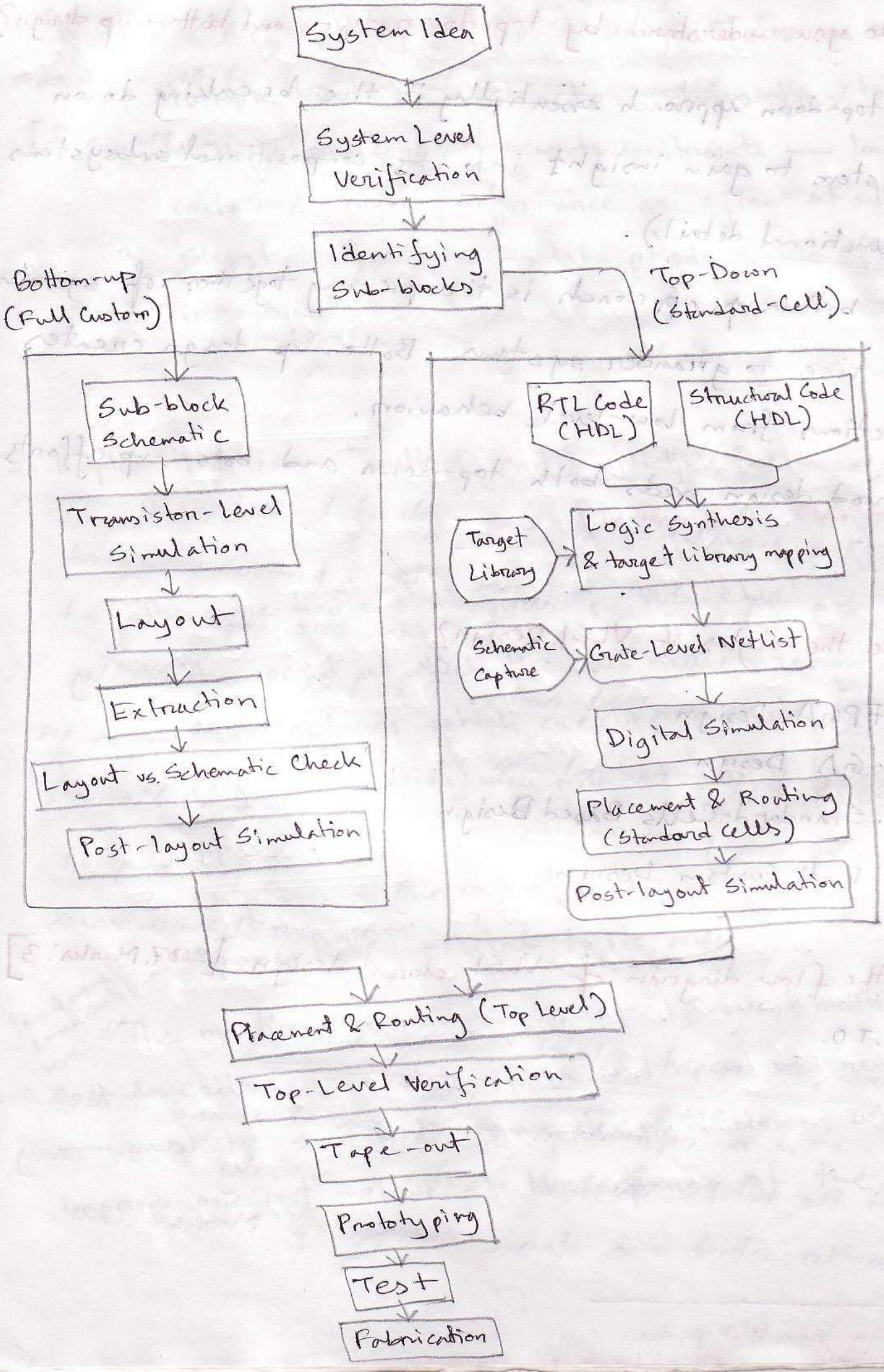
Good design needs both top-down and bottom-up efforts.

5. What are the styles of VLSI Design?

1. FPGA Design
2. GA Design
3. Standard-Cells Based Design
4. Full Custom Design

1. Draw the flow diagram of VLSI circuit design. [2007. Marks: 3]

P.T.O.


System Idea
System Level Verification
Identifying Sub-blocks
Bottom-up (Full Custom)
Top-Down (Standard-Cell)
Sub-block Schematic
Transistor-Level Simulation
Layout
Extraction
Layout vs. Schematic Check
Post-layout Simulation
RTL Code (HDL)
Structural Code (HDL)
Target Library
Logic Synthesis & target library mapping
Schematic Capture
Gate-Level Netlist
Digital Simulation
Placement & Routing (Standard cells)
Post-layout Simulation
Placement & Routing (Top Level)
Top-Level Verification
Tape-out
Prototyping
Test
Fabrication

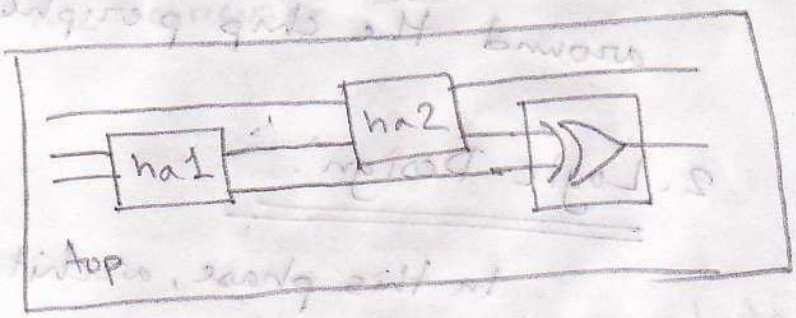
What is net list? [2007. Marks: 0.5]
What are the importances of netlist? [In course 05-06. Marks: 1]

Netlist gives for each net the terminals connected to that net.

Importance of netlist:

To understand how the system works, we only have to know each component's & netlist (i.e., input-output behavior), not how ~~that~~ the behavior is implemented inside the box.

Construct a full-adder using two half-adders and write down its netlist. [2007. Marks: 2]



net1 : top.in1 ha1.in1
leftin1 : top.in2 ha1.in2
net2 : top.in3 ha2.in1
s1 : ha1.out1 ha2.in2
net3 : ha2.out2 xor.in1
c1 : ha1.out2 xor.in2
outnet1 : ha2.out2 top.out1
outnet2 : xor.out1 top.out2

14. What are the VLSI IC design approaches?

- Design with Chips
- Full Custom Design
- Semi-custom Design

15. Describe each phases of full-custom design. [In-course. Marks: 4]

1. System Design:

The architecture is checked against the system specification to ensure that all required hardware features and data paths have been included. It is also usual to check that the ~~req~~ required number of input pins and output pins can be accommodated around the chip periphery.

2. Logic Design:

In this phase, architectural blocks are expanded into logic diagrams. Here each items drawn represents a particular logic function, such as a gate. Control and timing logic is also included at this level. Some simulation is also used to verify the logic function.

3. Circuit Design:

In this level, the logic is translated into circuits and dimension assigned to the translated circuit. Sometimes stick diagram is used to represent the circuit. Circuit simulation can verify the circuit at this level and provide an indication of power dissipation and heat.

4. Layout Design:

Once the circuit design is correct, the circuits are allocated to the positions on the silicon and geometric shapes are generated for each silicon layer corresponding to the circuits and their interconnections.

5. Fabrication:

After the geometric layout procedure is complete, the design normally passes out of the designer's hands. The data representation of the geometic layout is normally used to produce the mask of each silicon layer. The masks are then used at the different production stages of the fabrication process to produce the specified chip.

16. Describe the fabrication process. [In course 05-06. Marks: 3]

[See the prev. ques.]

17. Why logic design is placed before circuit design in full-custom design approach? Discuss with example.

[2007, 2006. Marks: 2]

Circuit design is the implementation of logic design. After ~~all~~ the logic of all the components are designed, the whole circuit is implemented.

Hence, ~~circuit design~~ logic design is placed before circuit design.

Lecture 3-4

Logic Synthesis, Queen McCluskey, BDD

1. What do you mean by logic synthesis? Write some logic synthesis techniques/tools. [2008, 2007, 2006, Marks: 2]

Logic synthesis is a process by which an abstract form of desired circuit behavior is turned into a design implementation in terms of logic gates.

Some logic synthesis techniques are – Karnaugh maps, Quine-McCluskey algorithm, Binary Decision Tree (BDT) etc.

2. What is essential prime implicant? [In-course 08-09, Marks: 1]

(Essential prime implicants are prime implicants that cover an output of the input function that no combination of other prime implicants is able to cover.)

A product term is called essential prime implicant of input function if there is a minterm that is only covered by that prime implecant.

3. Write down the advantage of Q&M minimization over K-map minimization. [2006. Marks: 1/2]

~~While K-map can handle five or six input variables so~~

1. While K-map becomes very complex for input variables greater than six, Q&M can handle any number of inputs.

2. Q-M happens to be nicely suited to a computerized solution than K-map does.

4. What do you mean by Ordered BDD and ROBDD?
[2007, Marks: 2]

Ordered BDD:

A BDD is called 'ordered' if different variables appear in the same order on all paths from the root.

ROBDD:

A BDD is said to be 'reduced ordered' if it is an ordered BDD and the following two rules have been applied to its graph:

1. Merge any isomorphic subgraphs.
2. Eliminate any node whose children are isomorphic.

5. What is the advantage of ROBDD?

The advantage of ROBDD is that it is canonical (unique) for a particular functionality. This property makes it useful in functional equivalence checking and other operations like functional technology mapping.

6. Define BDD.

A BDD is a data structure that is used to represent a boolean function.

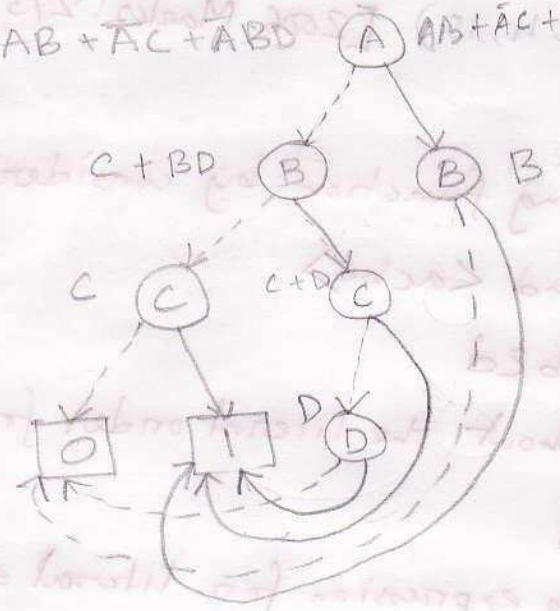
7. Draw the BDD for the following functions: [Marks: 4/each]

i) $F = AB + \bar{A}C + \bar{A}BD$ [2007]

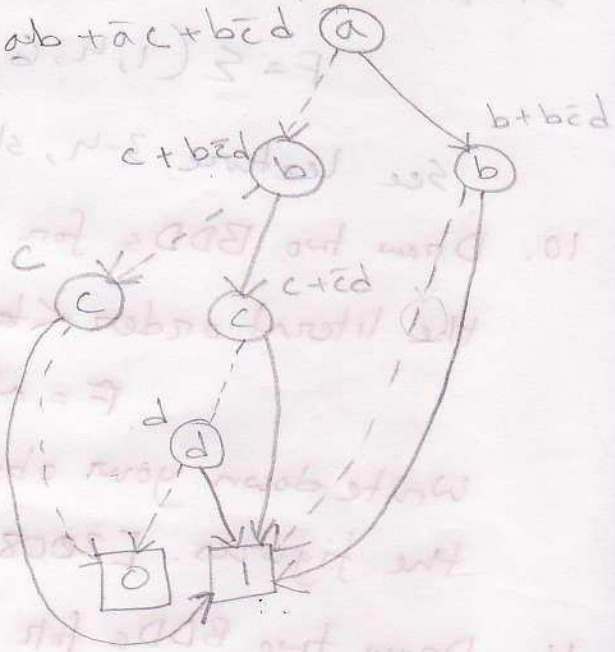
ii) $F = ab + \bar{a}c + b\bar{c}d$ [2006]

iii) $X = \bar{A}BC + A\bar{B}C + \bar{B}\bar{C}D$ [Inverse]

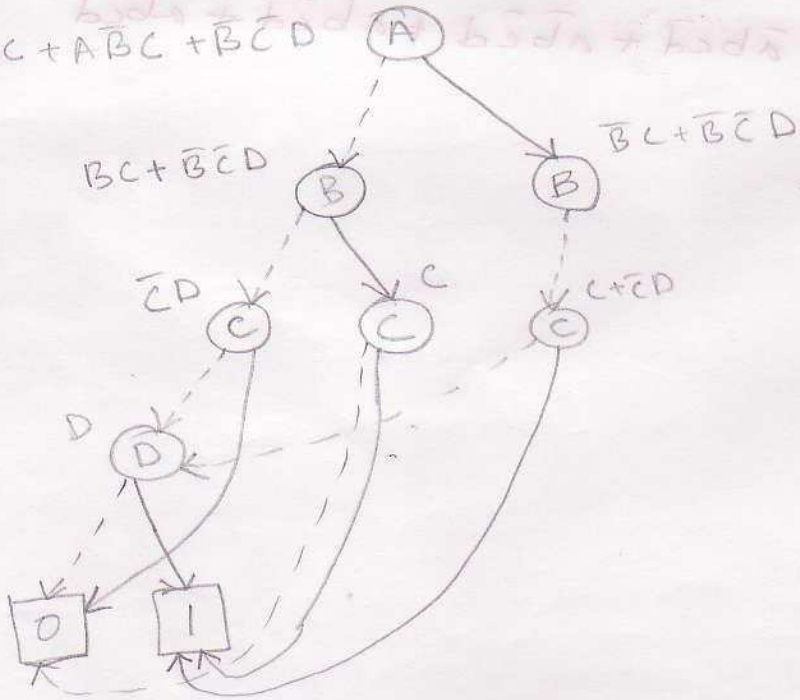
(i) $AB + \bar{A}C + \bar{A}BD$



(ii) $ab + \bar{a}c + b\bar{c}d$



(iii) $\bar{A}BC + A\bar{B}C + \bar{B}\bar{C}D$



8. Using Q-M algorithm, minimize the following function:

$F = \Sigma(0, 1, 2, 8, 10, 11, 14, 15)$ [2007. Marks: 5]

See lecture 3-4, slide No. 11.

9. Calculate the prime implicant for the following function:

$F = \Sigma(1, 4, 6, 7, 8, 9, 10, 11, 15)$ [2006. Marks: 2/3]

See lecture 3-4, slide no. 16.

10. Draw two BDDs for the following function by considering the literal order <~~bad~~ bcda> and <acbd>

$F = abc + \bar{a}c + b\bar{c}d$

Write down your observation about the literal order from the figures. [2008. Marks: 4]

11. Draw two BDDs for the following expression for literal sequence <a, b, d, c> and <b, d, c, a>. [in-course 08-09. Marks: 4]

$f_x = \bar{a}bc\bar{d} + a\bar{b}\bar{c}d + ab\bar{c}\bar{d} + abcd$

Lecture 5-6

Reversible Logic Synthesis

1. What is reversible logic (RL)? and reversible gate/circuit?

~~An RL~~

A logic expression is 'reversible' if each value in the ~~the~~ result of the expression has a unique input value combination.

A logic gate/~~circuit~~ circuit is L is reversible if, for any output y, there is a unique input x such that applying $L(x) = y$.

2. What is a garbage in reversible logic gate? [In-course 08-09. Marks: 0.5]

Every gate output that is not used as input to another gate or as a primary output, is called garbage.

In other words, the unutilized outputs from a gate are called garbage.

3. What are the advantages of reversible logic? [2008, 2006. Marks: 0.5]

Reversible logic is a potential way to improve the energy efficiency of a circuit, as no input information is lost and consequently no power is lost.

4. Write down the limitations of reversible logic. [In-course]

1. Garbage is the most significant limitation of RL.
2. Feedback loop is strictly restricted in RL.

5. Prove that at least two garbage output will be generated to realize a reversible full-adder circuit. [2008, Marks: 2]

From the following truth table of a full-adder circuit, we can observe that for the 2nd, 3rd and 5th or the 4th, 6th and 7th output combinations, we have no unique input combinations can be found.

Input			Output	
A	B	C_{in}	$Sum = A \oplus B \oplus C_{in}$	$C_{out} = AB \oplus BC_{in} \oplus C_{in}A$
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Let's consider the case of 2nd, 3rd and 5th rows.

To make the output combinations unique, let's consider the first input bit A (arbitrarily) and take it as a garbage output. Then the input-output combination will look as follows:

Input			Output		
A	B	C_i	S	Cout	$G_1 = A$
0	0	1	1	0	0
0	1	0	1	0	0
1	0	0	1	0	1

Still, the 1st and 2nd output combinations are the same. So, we take another input bit B as a second garbage:

Input			Output			
A	B	C_i	S	Cout	$G_1 = A$	$G_2 = B$
0	0	1	1	0	0	0
0	1	0	1	0	0	1
1	0	0	1	0	1	0

Now, the output combinations are unique and the unique combination for each output combination can be found. Similarly, the 4th, 6th and 7th output combinations of the aforementioned full-adder circuit can be made unique by taking these two garbages.

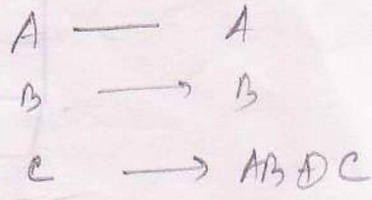
Thus, it can be said that to realize a reversible full-adder ckt, a minimum of two garbage output will be generated.

6. Realize the following expressions ~~with~~ using Toffoli and NOT gate only:

iii) $X = \overline{ab+bc} \oplus \overline{cd+e}$ [2008]

i) $X = (A+B) \oplus (CD+\bar{E}F)$ [In-course]

ii) $X = \overline{ab+bc} \oplus (cd+e)$ [2006]



[For Toffoli gate, $I_v = (A, B, C)$

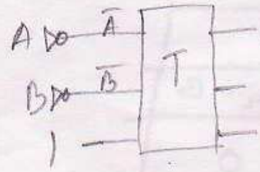
$O_v = (P, Q, R)$

where $P = A$,

$Q = B$,

$R = AB \oplus C$]

i)



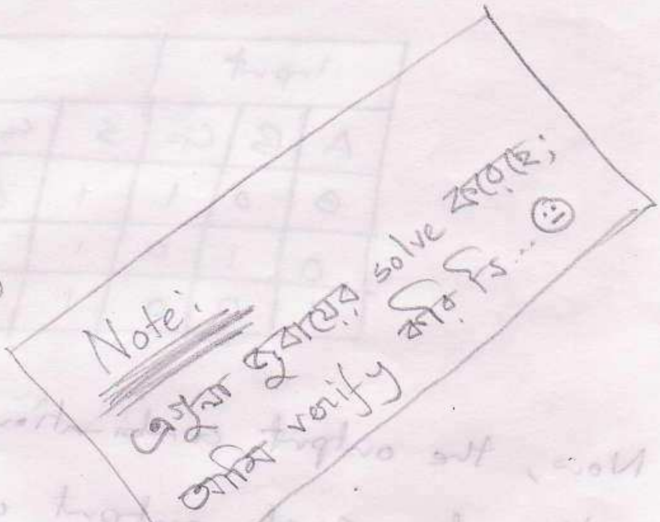
$P = \bar{A}$

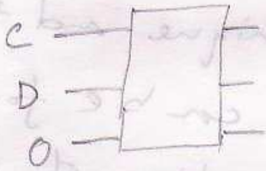
$Q = \bar{B}$

$R = \bar{A}\bar{B} \oplus 1$

$= \overline{\bar{A}\bar{B}}$

$= A+B$



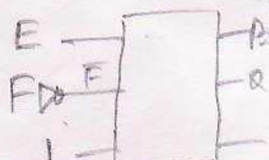


$P = C$

$Q = D$

$R = CD \oplus 0$

$= CD$



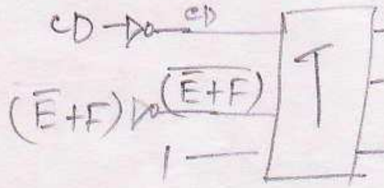
$P = E$

$Q = \bar{F}$

$E\bar{F} \oplus 1$

$= \overline{E\bar{F}}$

$= \bar{E}+F$



$P = \overline{CD}$

$Q = \overline{(\bar{E}+F)}$

$R = \overline{CD}.\overline{(\bar{E}+F)} \oplus 1$

$= CD + (\bar{E}+F)$

7. Realize the following expression with Peres and NOT gate only: [2007. Marks: 3]

$$X = \overline{ab+bc} \text{ EXOR } (cd+e)$$

[For Peres Gate, $I_v = (A, B, C)$

$O_v = (P = A,$
$Q = A \oplus B,$
$R = AB \oplus C)$]

A — A
B — A⊕B
C → AB⊕C

8. Draw the reversible ckt for the following expression using Fredkin Gates and Not gates only. [In-course 08-09. Marks: 4]

$$F = \overline{\bar{A}\bar{B} \oplus \overline{CD}} + BCD + \overline{\overline{\bar{C}D} \oplus EF}$$

[For Fredkin Gate, $I_v = (A, B, C)$

$O_v = (P = A,$
$Q = \bar{A}B \oplus AC,$
$R = \bar{A}C \oplus AB)$]

8. What are the challenges of Reversible Logic Design? [In-course 08-09. Marks: 0.5]

Lecture - 7

Simulation & Combinational Network Delay

1. What are the combinational network delays? How can we remove ~~it~~ them? [2007. Marks: 2]

The combinational network delays consist of

1. Gate delay:

Transistors that are too small to drive ~~the~~ a gate's load, particularly if the gate fans out to a number of other gates, may cause one gate to run much more slowly than all other gates in the system.

2. Path delay:

If some paths are significantly longer than ~~the~~ others, the long paths will determine maximum clock rate.

How to eliminate delays:

1. Removing Gate delay:

1. The transistors of the driving gate can be enlarged.
2. The logic can be redesigned to reduce the gate's fanout.

2. Removing Path delay:

1. Speed up a ~~delay~~ gate on the critical path (the path which creates the longest delay).
2. If there is more the one critical paths, speed up all of them
3. Speed up cutset through critical path.

2. How can we measure the delay of a gate? [In-course Marks: 2]

Delay of a gate $T_{gate} = T_{intrinsic} + C_{load} * T_{load}$

where, ~~T_{gate} = The t~~

$T_{intrinsic}$ = Intrinsic gate delay (caused by parasitics)

C_{load} = The actual load in some units, e.g. pF

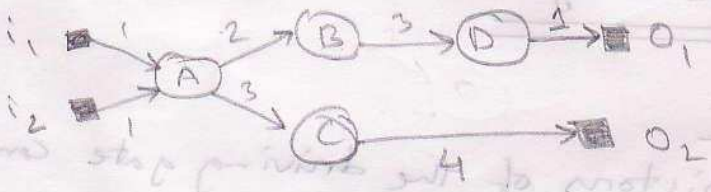
T_{load} = The delays per load in some units, e.g. ns/pF.

3. Discuss critical path and false path with examples. [2008, 2007, 2006. Marks: 2]

Critical path:

The longest delay path in a ckt is known as the critical path, since that path limits system performance.

For example, in the following delay graph, the critical paths are $i_1 \to A \to C \to O_2$ and $i_2 \to A \to C \to O_2$



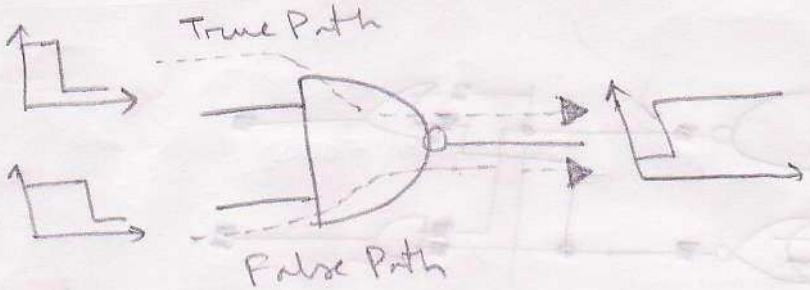
False Path:

A false path is a path which cannot be exercised due to Boolean gate conditions.

For example, in the following figure, the upper input goes low followed by the lower input. As either input going low causes the NAND's output to go high, the high-to-low transition of the latter input doesn't affect the gate's output

and is called a false path.

False paths cause pessimistic delay estimates.



4, Define Simulation. Differentiate between switch level simulation and circuit simulation. [2008. Marks: 2] In course 08-09.

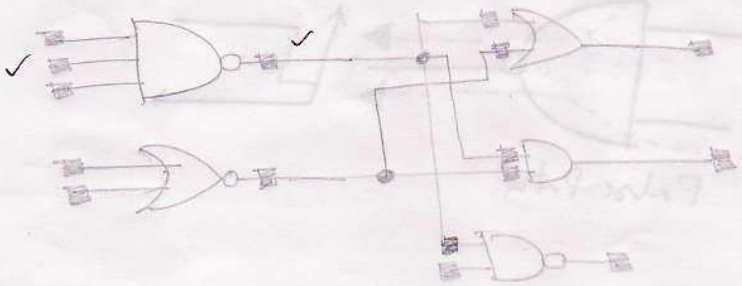
Simulation:

Simulation is a ~~to~~ design validation tool for both checking a circuit's function and other parameters such as performance.

Diff. between switch level & ckt simulation:

Switch level simulation	Ckt. simulation
1. Used for function verification of transistors in a ckt.	1. Used to verify the entire CMOS ckt. and to detect errors in the ckt.
2. Can accurately simulate a ckt. consisting of hundreds / thousands of transistors.	2. Can operate accurately only for small ckts.

5. Define fan out. Calculate the fan-in and fan-out of the following ckt.:



Fan out:

~~The fan out of a gate is the number of~~

The number of inputs that a gate's output drives ~~and~~ along with the primary ~~input~~ output (if exists) is called the fan-out of a gate.

~~The maximum number fan out of the g~~

The fan-out of the gate which has the maximum fan-out is called the fan-out of a circuit.

The fan-in of the ckt. is 3. The fan out is also 3.

6. Define fan-in.

The number of inputs of a gate is called its fan in.

The ~~nuber~~ fan-in of a gate which has the maximum number of inputs as compared to other ~~gathe~~ gates in the ckt. is called the fan in of the ckt.

Lecture 27 (2nd Part)

How to deal with fan-out problem? [2008. Marks: 2]
OR, Discuss about the ways to drive large fan-out in a circuit. [In-course 08-09. Marks: 2]

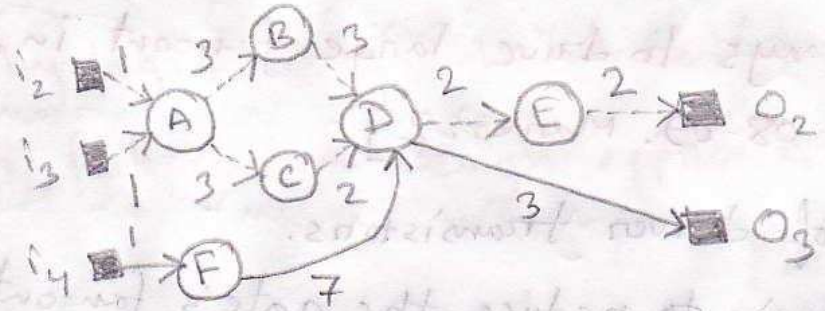
1. Increase sizes of driver transistors.
2. Redesign the logic to reduce the gate's fan-out by adding intermediate buffers.

Why and how do we reduce critical path length?

We reduce critical path length to improve system performance.
Critical path length can be reduced by:

~~1. Speed up~~
1. speeding up gates on the critical path by increasing transistor sizes or reducing wiring capacitance.
2. Redesigning the logic along the critical path to use a faster gate configuration by rewriting the function ≠ to reduce the number of logic levels.

9. What is critical path delay? Calculate the critical path delay for the following circuit: [In course 05-06. Marks: 2]



The delay along the critical path is called the critical path delay.

The critical path in the given circuit is $i_4 \rightarrow F \rightarrow D \rightarrow E \rightarrow O_2$ and the critical path delay is $(1+7+2+2) = 12$.

10. What are the advantages and disadvantages of the following simulators?

1. Circuit Simulator
2. Switch Level Simulator
3. Timing Simulator
4. Logic/Gate Level Simulator
5. Mixed Mode Simulator

[See lecture 7 slides 4 to 8.]

Lecture 8

Static Complementary Gate Structure

1. What is combinational logic?

Combinational logic is a type of digital logic which is implemented by boolean circuits, where the output is a pure function of the present input only.

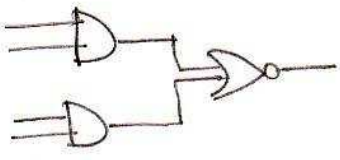
2. What is sequential logic?

Sequential logic is a type of logic ckt whose output depends on not only the present input but also on the history of the input.

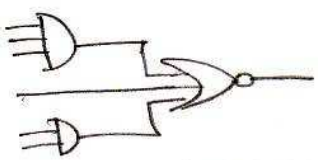
3. What is AOI? Give example. [In-course Marks: 2]

AOI (AND-OR-Invert) logic and AOI gates are two-level compound (or complex) logic functions constructed from the combination of one or more AND gates followed by a NOR gate.

Examples of AOI:



2-2 AOI



3-1-2 AOI

4. Construct the pull-up and pull-down of the following expressions: [Marks: 1.5/2 each]

i) $X = \overline{A}$ (Not Gate/Inverter)

ii) $X = \overline{AB}$ (NAND Gate)

iii) $X = \overline{A+B}$ (NOR Gate)

iv) $X = \overline{AB+BC+DEF}$

v) $X = \overline{(A+B)CD(E+F)}$ [2008], 2007]

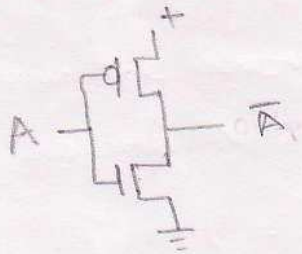
vi) $X = \overline{abc+de+f}$ [2006]

(vii) $F = \overline{(A+B+C)(B+D)(A+D)}$ [In-course 08-09]

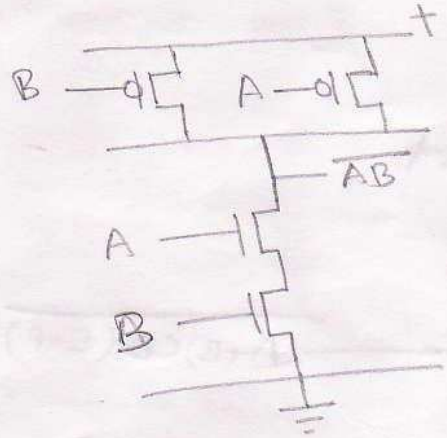
Note:

1. $+V_{cc}$ & Ground সবসময় উপরে থাকবে।
2. Pull-up-এ pMos-এ bubble সবসময় ইনপুট থাকবে।
3. Series connection-এ (AND) input সবসময় উপরে থাকবে, parallel-এ (OR) সবসময় উপরে থাকবে।

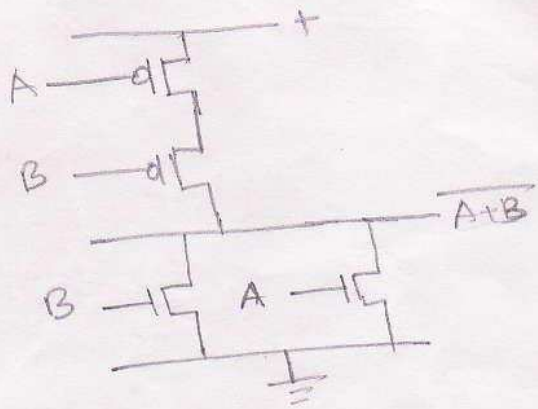
i)



ii)

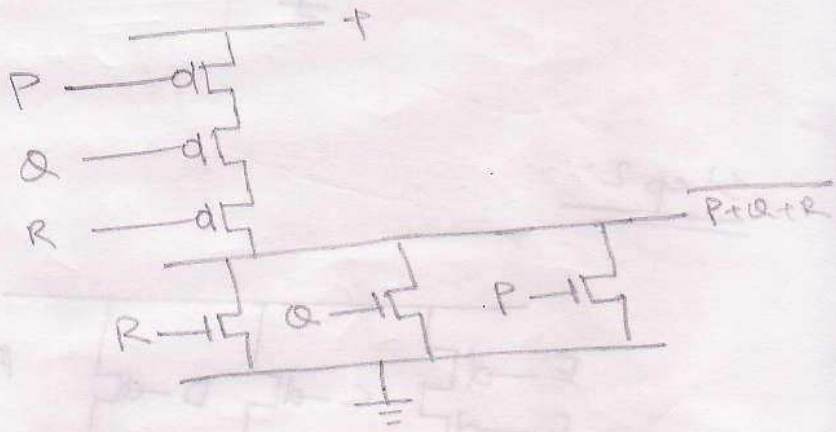


iii)

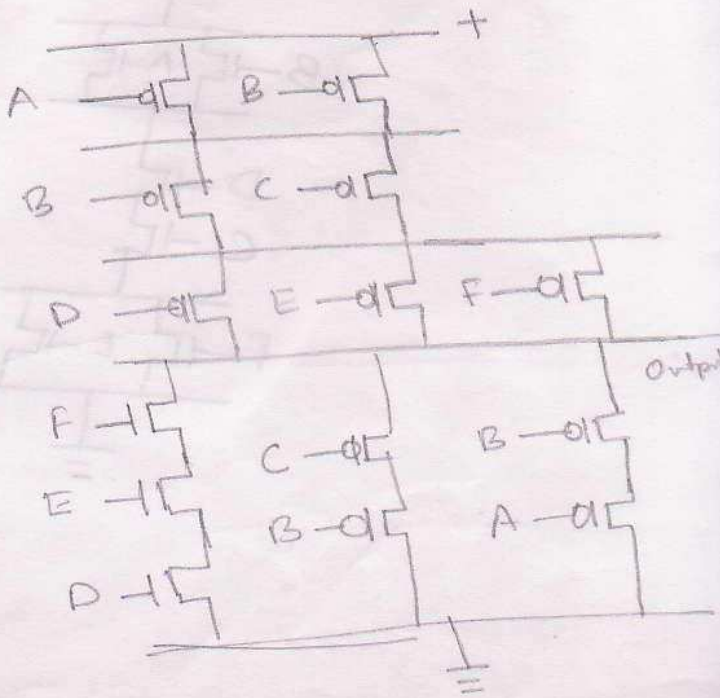


iv)

Step 1: Let, $X = \overline{P+Q+R}$

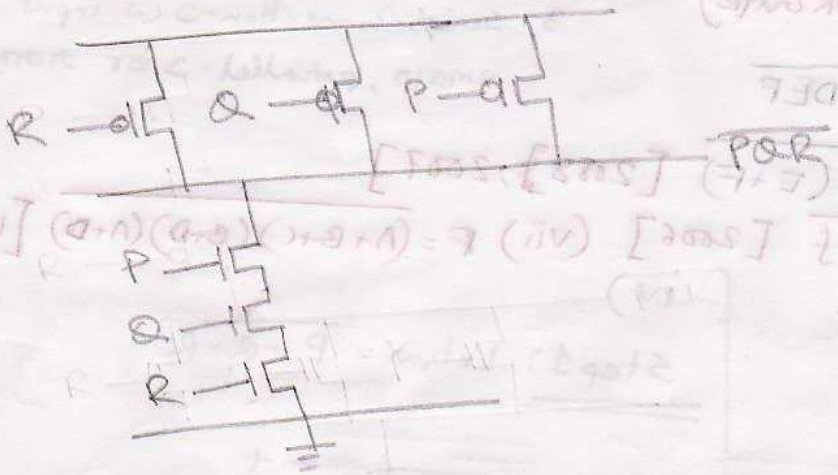


Step 2:

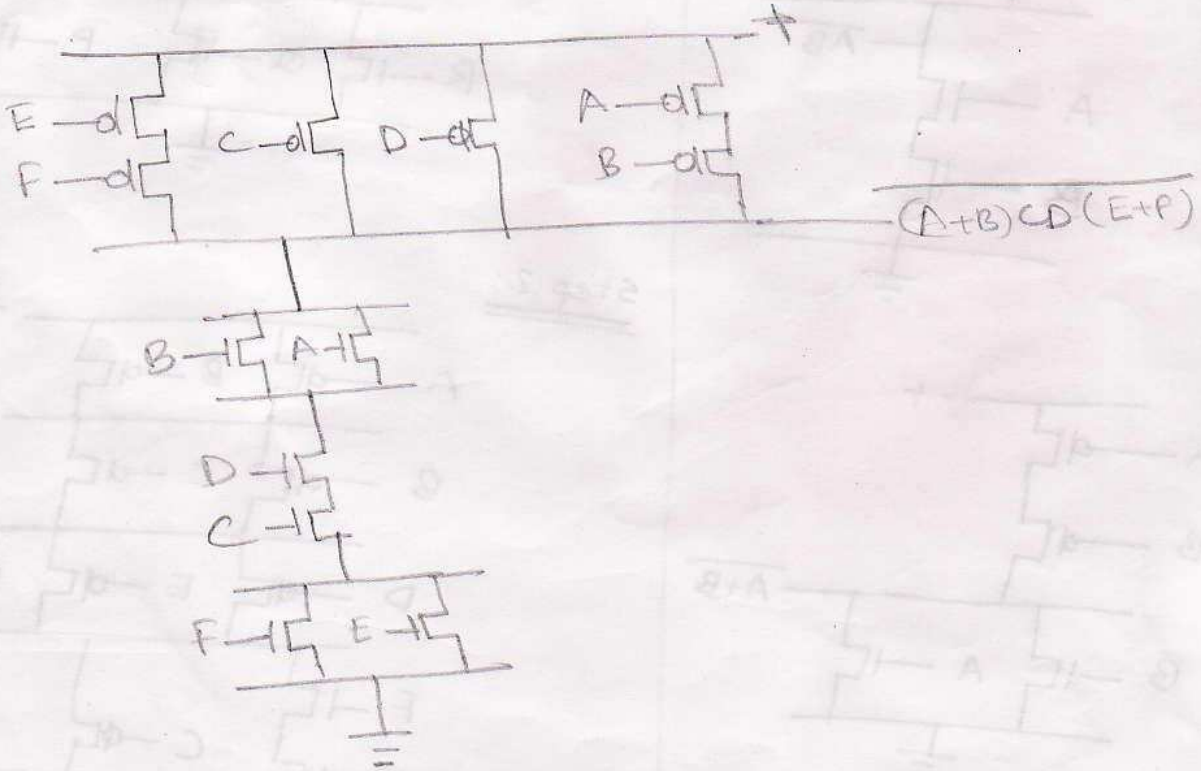


v) $X = \overline{(A+B)CD(E+F)}$

Step 1: Let, $X = \overline{PQR}$



Step 2:



Lecture 8 (2nd Part)

Why designing gates for logic function is non-trivial?

Because,

1. Logic gates may not be available in the library for all logic expressions.
2. A logic expression may map into gates that consume a lot of area, delay on power.

What do you understand by completeness of functions?

~~A functi~~ A set of functions $f_1, f_2, \ldots$ is complete if and only if every Boolean function can be generated by a combination of the functions.

For example, NAND and NOR are complete sets, whereas {AND, OR} is not complete.

Lecture 9

MOS

1. Write down the properties of enhancement mode of N-MOS transistor. [2006. Marks: 3]

   1. $V_{threshold} = 1V$
   2. The semiconductor surface between the source and drain is doped with p-type impurity.
   3. The n-MOS device is OFF when $V_{gs} = 0V$.

2. Write down the properties of ~~enhancement~~ depletion mode of nMOS transistor. [In-course: Marks: 2]

   1. $V_{threshold} = -4V$
   2. The semiconductor surface between the source and drain is doped with n-type impurity.
   3. The nMOS device is turned ON when $V_{gs} = 0V$.

3. Clearly describe the basic operations of nMOS enhancement mode. [2006. Marks: 2]
   OR, Discuss about different types of status of nMOS enhancement mode with appropriate diagrams.

   Three basic operations:

   1. Cut-off / subthreshold / weak-inversion mode
   2. Non-saturated / triode / linear region / ohmic mode
   3. Saturated / active mode.

1. Cut-off: $V_{gs} > V_t$ and $V_{ds} = 0$ | $V_t \neq 0$

Since $V_{ds} = 0$, no current will flow through the channel.

2. Non-saturated: $V_{gs} > V_t$ and $V_{ds} < V_{gs} - V_t$ | $V_{ds} \neq 0$

Channel exists as well as electricity is flowing, but not in the maximum level. So, voltage should be increased. The current flow will be continued until $V_{ds} > V_{gs} - V_t$.

3. Saturated: $V_{gs} > V_t$ and $V_{ds} \geq V_{gs} - V_t$

This state will be continued until $V_{ds} < V_{gs} - V_t$.

4. What is threshold voltage? [In course 08-09. Marks: 1]

Threshold voltage is the voltage at which a MOS device begins to conduct (Turn ON).

The input potential is called threshold voltage, V_t.

5. What is transit time? Establish the relationship between V_{ds} and E_{ds}. [2008 2006. Marks: 2]

Transit time:

It is the time which the electron takes to flow from drain to source through the channel of a MOS transistor. This time increases when the voltage between the drain and source decreases.

Relationship among V_{ds}, E_{ds}, I_{ds}:

We know, transit time $\lambda = L/V$ [where L = length of the channel, μ = Electron or hole mobility]

$$= L/(\mu E_{ds})$$

$$= L^2/(\mu V_{ds})$$

$$\therefore \frac{L}{\mu E_{ds}} = \frac{L^2}{\mu V_{ds}}$$

$$\Rightarrow V_{ds} = L E_{ds}$$

Again,

$$I_{ds} = \frac{\text{Charge induced into channel } (Q_c)}{\text{Transit time } (\lambda)}$$

$$\Rightarrow \lambda = \frac{Q_c}{I_{ds}}$$

$$\Rightarrow \frac{L^2}{\mu V_{ds}} = \frac{Q_c}{I_{ds}}$$

$$\Rightarrow I_{ds} = \frac{Q_c \mu V_{ds}}{L^2}$$

$$\Rightarrow I_{ds} = \left(\frac{Q_c \mu}{L}\right) \cdot E_{ds}$$

6. What are the differences between BJT and MOS?
[2006. Marks: 2]

BJT	MOS
1. High static power dissipation.	1. Low static power dissipation.
2. Low packing density.	2. High packing density.
3. Complex structure.	3. Simpler structure.
4. Implementation of dynamic logic is not permitted.	4. Implementation of dynamic logic is permitted.
5. Faster	5. Slower.
6. Requires large area.	6. Requires small area.
7. According to power-speed product, BJT is not preferable.	7. According to power-speed product, MOS is preferable.

7. Show the supremacy of MOS over BJT with an appropriate example. [2008, 2007, Marks: 2]
OR, with appropriate example, prove the supremacy of MOS over BJT.

Though the speed of BJT is better than the MOS, MOS has some other primitives that lead to prefer MOS over BJT.

MOS consumes less power than BJT and requires less area. Its structure is simpler than BJT. Speed-power factor is the most important point to select MOS instead of BJT.