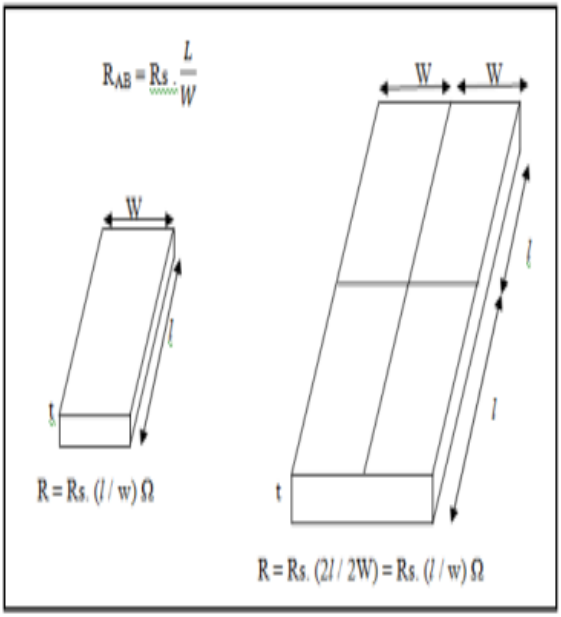
WINTER– 16 EXAMINATION (Subject Code: 17659)

Model Answer

Important Instructions to examiners:

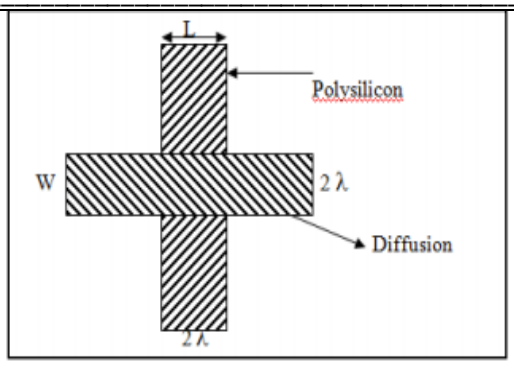
1) The answers should be examined by key words and not as word-to-word as given in the model answer scheme.
2) The model answer and the answer written by candidate may vary but the examiner may try to assess the understanding level of the candidate.
3) The language errors such as grammatical, spelling errors should not be given more Importance (Not applicable for subject English and Communication Skills.
4) While assessing figures, examiner may give credit for principal components indicated in the figure. The figures drawn by candidate and model answer may vary. The examiner may give credit for any equivalent figure drawn.
5) Credits may be given step wise for numerical problems. In some cases, the assumed constant values may vary and there may be some difference in the candidate's answers and model answer.
6) In case of some questions credit may be given by judgement on part of examiner of relevant answer based on candidate's understanding.
7) For programming language papers, credit may be given to any other program based on equivalent concept.

Q. No.	Sub Q.N.	Answer	Marking Scheme
Q.1		**Attempt any THREE of the following**	**12-Total Marks**
1	a)	**Explain the process of estimation of resistance of the channel and how it is calculated.**	**4M**
	Ans:	**Estimation of Resistance:** Consider a uniform slab of conducting material of resistivity ρ. Let W be the width, t the thickness and *L* the length of the slab. Hence the resistance between A and B terminal. $R_{AB} = \frac{\rho L}{A}$ ohms. Where A = cross-sectional area. Thus $R_{AB} = \frac{\rho L}{t.W}$ ohms. Consider the case in which L = W, that is a square of resistive material then $R_{AB} = \frac{\rho}{t} = Rs$ Where • Rs = ohm per square or sheet resistance • Therefore, $Rs = \frac{\rho}{t}$ ohm per square • Hence Rs is completely independent of the area of the square. • Thus,	**(Explaination :4 Mrks**



- Thus to obtain the resistance of a conductor on a layer multiply the sheet resistance Rs, by the ratio of length to width of the conductor as shown in the equation. For examples, resistances of the two shapes shown in the above figure are same because the length to width ratio of both the slabs is same, even though the sizes are different. Although the voltage – current characteristics of a MOS transistor are generally non-linear, it is used to approximate its behavior in terms of a 'change resistance' to estimate the performance.
- The channel resistance Rc
- $Rc = K\left(\frac{L}{W}\right)$
- Where $K = \frac{1}{\mu Cox\,(vgs - Vt)}$

  μ = surface mobility of majority carriers. (i.e. electrons in n-device and holes in p-device)
- Since mobility and threshold voltage are temperature dependent parameters, the channel resistance change with temperature. But as given in equation of Rc, channel resistance mainly depends on length to width ratio of the channel.



- In the above diagram both poly and diffusion are of 2λ widths. The overlapping region is called a 'channel', with length and width 2 λ, as shown in figure. The thinnox is only in the channel region.
- In the above example channel length L = 2 λ and width W =2 λ.
- The channel is square in shape and channel resistance.

$R = Rs\left(\frac{L}{W}\right)$

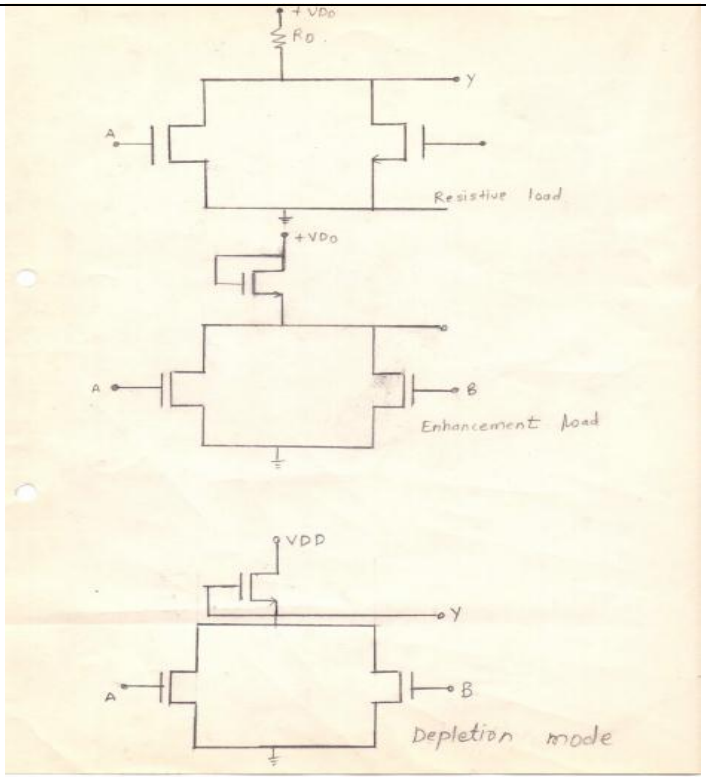
Therefore, $R = Rs\left(\frac{2\lambda}{2\lambda}\right)$

$R = 1^2$. Rs ohm / square.

Therefore, R = Rs ohms.

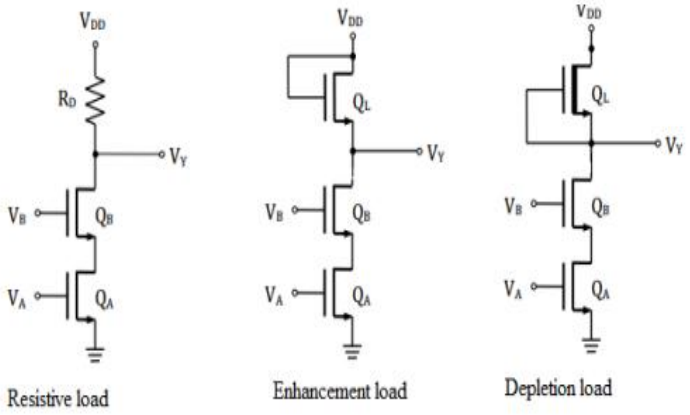
b) Draw NAND and NOR gates using NMOS. **4M**

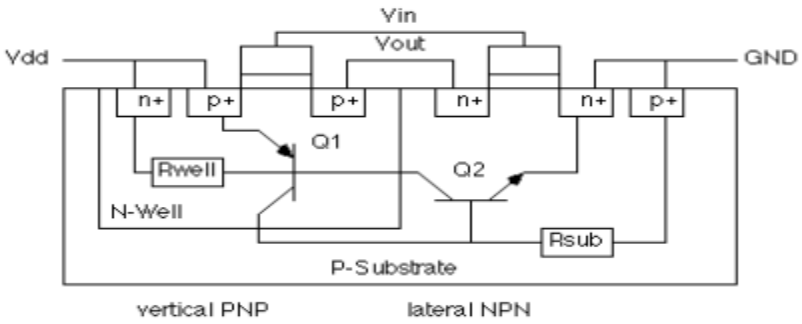
Ans:



(2 Marks each)

(Any one diagram for NAND and NOR each)



c)	**Explain Latch-up in CMOS and how it is minimized.**	**4 M**
Ans:	 **Figure: Latchup diagram** How to prevent latch up 1. Increase the doping of well and substrate 2. Increase the depth of the well region. 3. Increase the spacing of NMOS / PMOS devices. 4. The source / well contact must be kept nearly to reduce series resistance, 5. Place a substrate contact for every 5 - 10 transistors, 6. Layout of n and p transistors with packing of n devices towards gnd and p devices toward VDD. 7.Using gold doping in the well region of substrate region. 8. To avoid undesirable parasitic 9.Guard Rings / channel stops: Additional heavy diffusion region may be same type of material as of substrate or well. 10. Reduce gain: add thin epitaxy layer over substrate.	**(Diagram :2 Marks,Mini minimization :2 Mark)**
d)	**Explain any three operatiors used in VHDL.**	**4 M**
Ans:	**OPERATORS:** The predefined operators in the language are classified into the following six categories: **1. Logical operators** **2. Relational operators**	**(Any 3 operators)**

3. **Shift operators**
4. **Adding operators**
5. **Multiplying operators**
6. **Miscellaneous operators**

The operators have increasing priority from 1 to 6. The evaluation of the expression is done from left to right. Parenthesis may be used to override the left to right evaluation.

1. Logical Operators:

The seven logical operators are:

and or nand nor xor xnor not

These operators are defined for the predefined types BIT and BOOLEAN. During evaluation of logical operators, bit values '0' and '1' are treated as FALSE and TRUE values of the BOOLEAN type respectively. The not operator has the same priority as that of miscellaneous operator.

2. Relational Operators:

These are

= /= < <= > >=

The result type for all relational operations is always the predefined type Boolean. The = (equality) and the /= (inequality) operators are predefined on any type except file types. The remaining four relational operators are predefined on any scalar type (i.e. integer or enumerated types) or descreate array type (i.e. array in which element values belong to discrete type).

3. Shift Operators:

These are:

sll srl sla sra rol ror

Each of the operators takes an array of BIT or BOOLEAN as the left operand and an integer value as the right operand and performs the specified operation.

The sll operator (shift left logical) and srl operator (shift right logical) fill the vacated bits with left operand type LEFT.

The sla operator (shift left arithmetic) fills the vacated bits with rightmost bit of the left operand, while the sra operator (shift right arithmetic) fills the vacated bits with the left most of the left operand. The operator causes the vacated bits to be filled with the displaced bits in a circular fashion.

hese are "1001010" sll 2 is "0101000" -- filled with BIT LEFT, which is '0'.

"1001010" srl 3 is "0001001" – filled with '0'.

"1001010" sla 2 is "0101000" – filled with rightmost bit which is '0'.

"1001010" sra 3 is "1111001" –filled with '1'which is the leftmost bit.

"1001010" rol 2 is "0101010" –rotate left

"1001010" ror3 is "0101001" –rotate right

4. **Adding operator:**

These are:

+ – &

The operands for the addition(+) and substraction (-) operators mustbe of the same numeric type. The operands for the & (concatenation) operators can be either a one dimensional array type or an element type. the result is always an array type.

e.g. '0' & '1'

results in an array of characters "01".

5. **Multiplying Operators:**

These are:

*** / mod rem**

The multiplication and division operators are predefined for both operands being of the same integer or real type. The result is also of the same type.

For the division operator, division of a value of physical type by either an integer or a real value is allowed, and the result type is the physical type.

The rem (reminder) and mod (modulus) operators operates on operands of integer types, and the result is also of the same type. The result of a rem operation has the sign of its first operand and is defined as:

A **rem** B = A-(A/B)*B

The result of mod operator has the sign of the second operand and is defined as:

A **mod** B= A-B*N

6. **Miscellaneous Operators:**

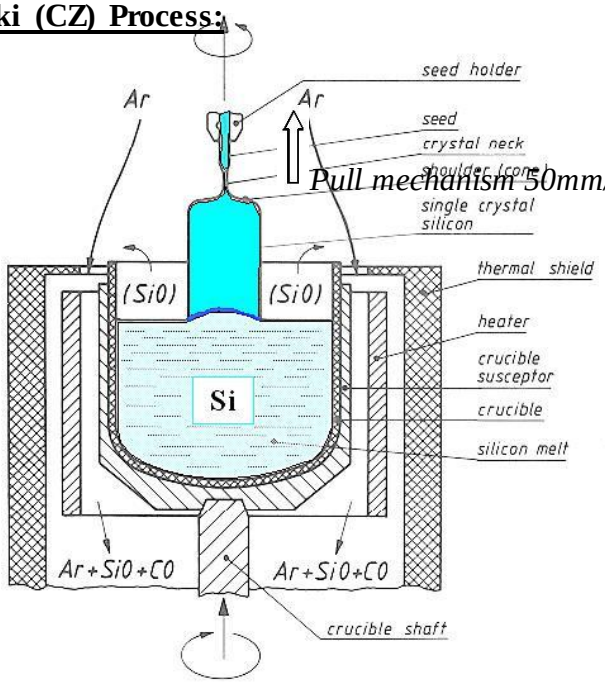
The miscellaneous operators are:

abs **

The abs (absolute) operator is defined for any numeric type.

The ** (exponentiation) operator is defined for the left operand to be of integer or floating point type and for the right operand (i.e the exponent) to be of the integer type only.

Operator Type							
logical	and	or	nand	nor	xor	xnor	not
relational	=	/=	<	<=	>	>=	
shift	sll	srl	sla	sra	rol	ror	
addition	+	-					
unary	+	-					
multiplying	*	/	mod	rem			
other	**	abs	&				

b)	**Attempt any <u>One</u> of following:**	**6 M**
(a)	**Explain CZ process for wafer fabrication, with neat diagram.**	**6M**
Ans:	**<u>Czochralski (CZ) Process:</u>**  	**(Diagram:3Mark Explanation: 3 Mark)**

It consists of Quartz crucible, which is surrounded by a graphite radiator. The graphite is heated by radio frequency induction heating and temperature maintained a few degrees above the melting point of silicon (approx. 1425^0C), the atmosphere just above the polysilicon melt is typically helium or orgon for freezing.
A polycrystalline Si is melted in the crucible and controlled amount of impurities (p type or n type) are added to the melt to provide the crystal with required electrical properties.
After the seed (single crystal silicon piece) is dipped into the melt, the seed is gradually withdrawn vertically from the melt while simultaneously being rotated. The molten polycrystalline silicon melts the tip of the seed and it is withdrawn, refreezing occurs. As the melt freezes, it assumes the single crystal form of the seed. This process is continued until the melt is consumed. The diameter of the ingot (rod of silicon) is determined by the seed withdrawn rate and seed rotation rate.
The produced crystalline silicon rod is then slicing into wafers using cutting tools like diamond blades. Following slicing at least one face of the wafer is polished to flat scratch free mirror finish surface.

(b) Define the terms: (1) Metastability (2)Noise margins (3) Skew — **6M**

Ans: — **(Each :2 Mark)**

Metastability: In digital systems, when two asynchronous signals combine in such a way that their resulting output goes to an indeterminate state or unpredictable state. This state is known as metastable state. At the end of metastable state the output settles either 0 or 1. This whole process is known as metastability.

Noise Margins: It is a measure of noise immunity of a gate or circuit (noise immunity is the ability of a gate or circuit to tolerate any noise present in a signal without performing a wrong operation).

Skew (Clock Skew): skew is defined as "the magnitude of the time difference between two events that ideally would occur simultaneously"
We say that in synchronous systems, all the flip-flops are clocked simultaneously. But this is not always done. The clock signal, which is said to be apied simultaneously to all the flip-flops, may cause a minute delay changes due to some variation in the wiring between the components. Due to this, it may happen that the clock signal may arrive at the clock inputs of different flip-flops at different times. This delay is termed as clock skew.

Q 2 — **Attempt any Four of the following** — **16 M**

a) Differentiate between Xilix and Atmel series architecture of CPLD.(four point) — **4 M**

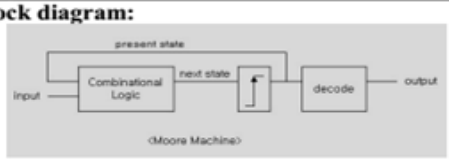
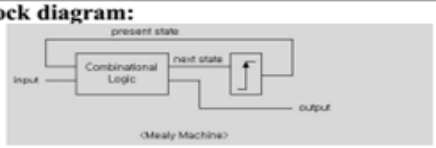
Ans: — **(4 point Each point:1 Marks)**

Sr. No.	Xilinx CPLD	ATMEL CPLD
1.	XC9536, XC9572, XC95108 these have 36, 72 and 108 macro-cells.	ATF1502, ATF1504, ATF1508 these have 32, 64 and 128 macro-cells.
2.	Available in variety of packages but 44 and 84 pins PLCC or J-lead packages are more popular.	Available in variety of packages but 44, 68 and 84 pins PLCC or J-lead packages are more popular.

	3.	Xilinx offers their web packs free download.	ATMEL offers their WinCUPL PLD compiler for free download.	
	4.	The most current version only works on windows XP.	The most current version only works on Windows XP.	
	5.	In conversion application XC4000 FPGAs the equivalents are 1. XC4002,XC4003 2. XC5200 SERIES XC5202 AND XC5204 3. XC505((SPARTAN SERIES) XC 10.	In conversion application atmel series FPGAs equivalent are 1. AT40K05,AT40K05/10 2. AT40K05,AT40K05/10 3. AT40K05,AT40K05/20	

b) Compare moore and meaiy machines.(four point) — **4 M**

Ans: — **(4 Point, Each point:1 Marks)**

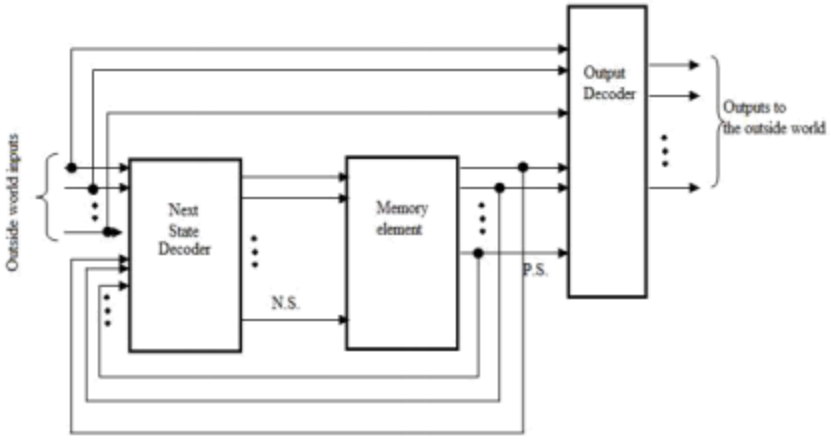
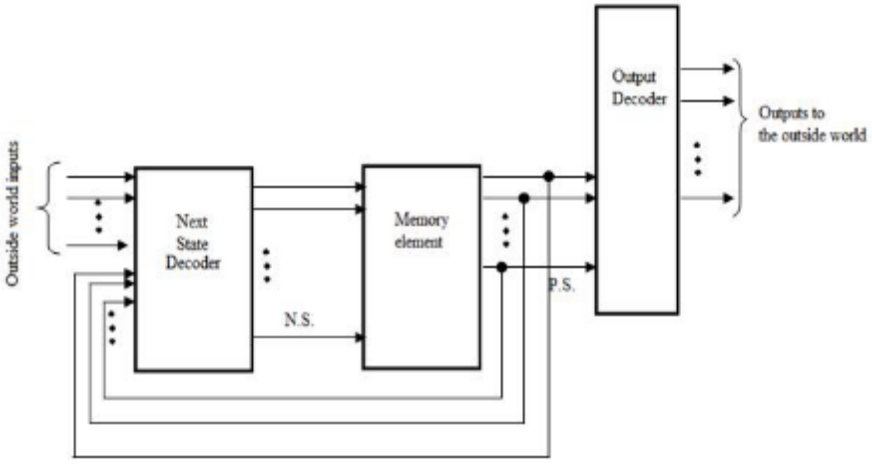
SR NO	MOORE MACHINE	MELAY MACHINE
1.	Output is function of state of machine.	Output is function of state of machine and present input condition.
2.	Requires more number of states.	Requires less number of states.
3.	Faster.	Slower.
4.	Simple design.	Complex design.
5.	Output in state.	Output is at the time of state transition.
6.	**Block diagram:**  <Moore Machine>	**Block diagram:**  <Mealy Machine>

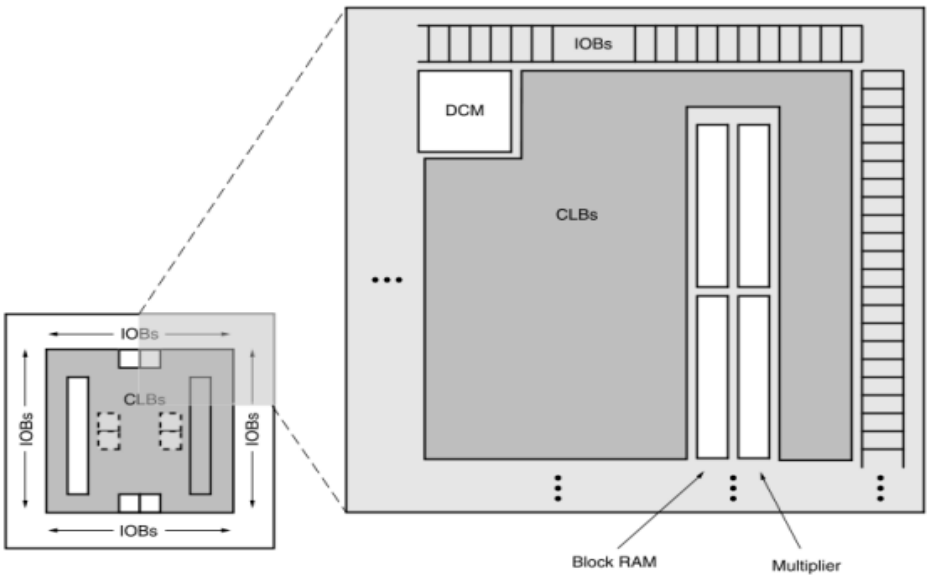
c) Write VHDL code for 3-bit up-counter. — **4 M**

Ans: — **(Entity:1 Mark Architecture :3Mark) Program using any other logic should be considered)**

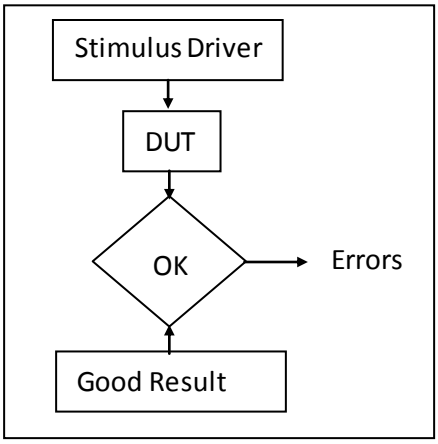
Program:

```
library IEEE;
use ieee.std_logic_1164.all;
use ieee.std_logic_unsigned.all;
entity counter is
        port(Clock, CLR : in  std_logic;
        Q : out std_logic_vector(2 downto 0));
end counter;
architecture archi of counter is
signal tmp: std_logic_vector(2 downto 0);
begin
process (Clock, CLR)
begin
        if (CLR='1') then
        tmp <= "000";
        elsif (Clock'event and Clock='1') then
        tmp <= tmp + 1;
        end if;
end process;
        Q <= tmp;
end archi;
```

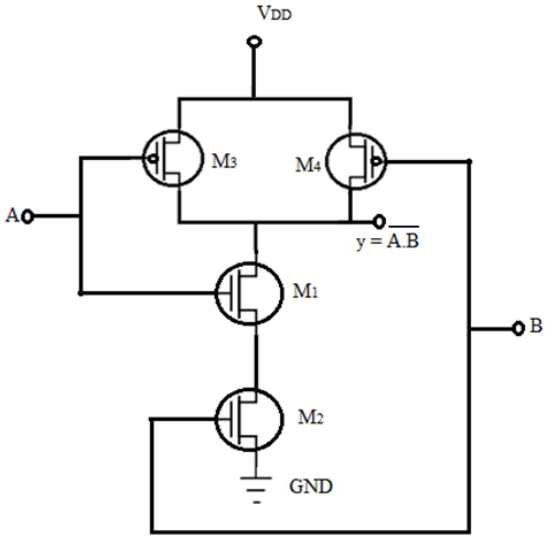
d)	**What are the advantages of twin-tub process of CMOS fabrication?**	**4 M**
Ans:	Advantages of Twin-tub process are (1) Separate optimized wells are available. (2) Balanced performance is obtained for n and p transistors. (3) Make it possible to optimize "Vt", "Body effect", and the "Gain" of n, p devices, independently. (4) The parasitic transistor is not likely to be formed. (5) No latch-up.	**(Any 4Advantages :1 Mark each)**
e)	**List the type of FSM. Draw labelled diagram of each.**	**4 M**
Ans:	**Types of FSM:** • **Melay Machine** • **Moore Machine** • **Melay Machine**  • **Moore Machine**  	**(Type:1 Mark Labelled Diagram:3 Marks)**

	f)	**Write the advantages and purpose of VHDL.**	**4 M**
	Ans:	Purpose of VHDL: VHDL was develop to overcome the flaws found in HDL. It is used to model a digital system at many levels of abstraction, ranging from the algorithm level to get level. It enables to express the concurrent or sequential behavior of digital system with and without timing. VHDL is intended for circuit synthesis as well as circuit simulation. **Advantages:** • It supports hierarchy. It uses set of components and interconnect them, each component can also be modeled as a set interconnected sub components. • It supports both synchronous and asynchronous timing models. • It supports various digital modeling techniques like finite state machine (FSM), Algorithmic description and Boolean expressions. • Concurrency, timing and clocking can be modeled in this language. • The logical operation and timing behavior of a design can be simulated. • It is not technology specific i.e. it can work with Xilinx, Lattice, Atmel series of CPLDs or FPGAs. • It is not case sensitive.	**(Purpose:2 Marks Advantages:2 Marks)**
Q. 3		**Attempt any Four of the following**	**16 M**
	a)	**Write basic architecture of spartan 3 FPGA series.**	**4 M**
	Ans:		**(2 Marks for Diagram, 2 Marks for Explanation)**

	The Spartan-3E family architecture consists of five fundamental programmable functional elements: **Configurable Logic Blocks (CLBs):** Contain flexible Look-Up Tables (LUTs) that implement logic plus storage elements used as flip-flops or latches. CLBs perform a wide variety of logical functions as well as store data. **Input/ Output Blocks (IOBs):** Control the flow of data between the I/O pins and the internal logic of the device. Each IOB supports bidirectional data flow plus 3-state operation. Double Data-Rate (DDR) registers are included. **Block RAM :** Provides data storage in the form of 18-Kbit dual-port blocks. **Multiplier Blocks :** Accept two 18-bit binary numbers as inputs and calculate the product. **Digital Clock Manager (DCM**): Blocks provide self-calibrating, fully digital solutions for distributing, delaying, multiplying, dividing, and phase-shifting clock signals.	
b)	**What is test bench and write down a typical test bench format.**	**4 M**

Ans:	A test bench is a model that is used to exercise and verify the correctness of a hardware model. The design which allows verifying the functionality of design at each step in VHDL synthesis based methodology is known as test bench.
	 
	
	A typical test bench format is
	entity TEST_ BENCH is
	end;
	architecture TB_ BEHAVIOR of TEST_ BENCH is
	component ENTITY_ UNDER_TEST
	port (list- of- ports-their-types-and-modes);
	end component;
	Local-signal-declarations ;
	begin
	Generate-waveforms-using-behavioral-constructs;
	Apply- to-entity-under-test;
	EUT: ENTITY_UNDER_TEST port map (port-associations) ;
		(Defination :1 Marks and Typical Test bench format :3 Marks, diagram in optional)

	Monitor-values-and-compare-with-expected-values; end TB_ BEHAVIOR;	
c)	**Write VHDL code to implement 4:1 multiplexer .**	**4 M**
Ans:	Library IEEE; Use IEEE. Std_logic_1164.all; Entity MUX4_1 is Port(I : in bit_vector (3 downto 0); S: in bit_vector (1 downto 0); Y: out bit); end MUX4_1; architechture MUX of MUX4_1 is begin with S Select Y <= I(0) when "00" I(1) when "01" I(2) when "10" I(3) when "11"; '0' when others; -- optional end MUX;	**(Any other statement used marks should be given)**
d)	**Explain NAND gate using CMOS transistors.**	**4 M**
Ans:	**Explanation:** A general CMOS circuit consists of an n-MOS logic block between the output and ground and a pMOS logic block between the output and V_{DD}. • In CMOS, Number of nMOS transistors = Number of pMOS transistors. For two input NAND gate, there are two nMOS transistors in pull down, and hence two pMOS transistors in pull up. Since pMOS and nMOS are complementary to each other, pMOS transistors are connected in parallel between output and V_{DD}. • Two input CMOS NAND gate is shown in figure below. The pull down sections has transistors M_1 and M_2 in series and pull up section has transistors M_3 and M_4 are in parallel. • Transistors M_1 and M_3 from one CMOS with A as input and transistors M_2 and M_4 from another CMOS with B as a input.	**(Diagram: 2 Marks, Explanation: 2 Marks)**

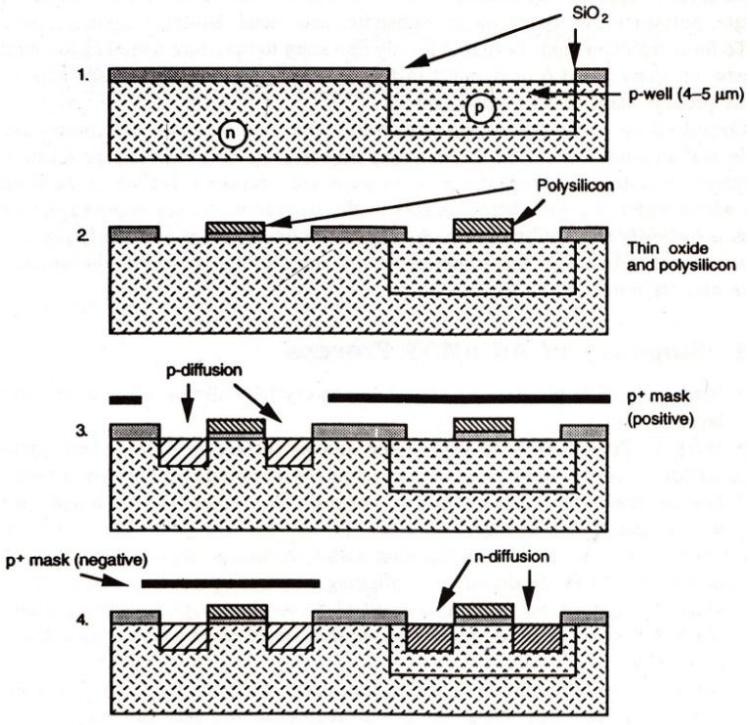


Truth Table (Optional):

Input		CMOS				Output
A	B	M1	M2	M3	M4	
0	0	OFF	OFF	ON	ON	1
0	1	OFF	ON	ON	OFF	1
1	0	ON	OFF	OFF	ON	1
1	1	ON	ON	OFF	OFF	0

e) **Explain P well process with suitable diagram.** **4 M**

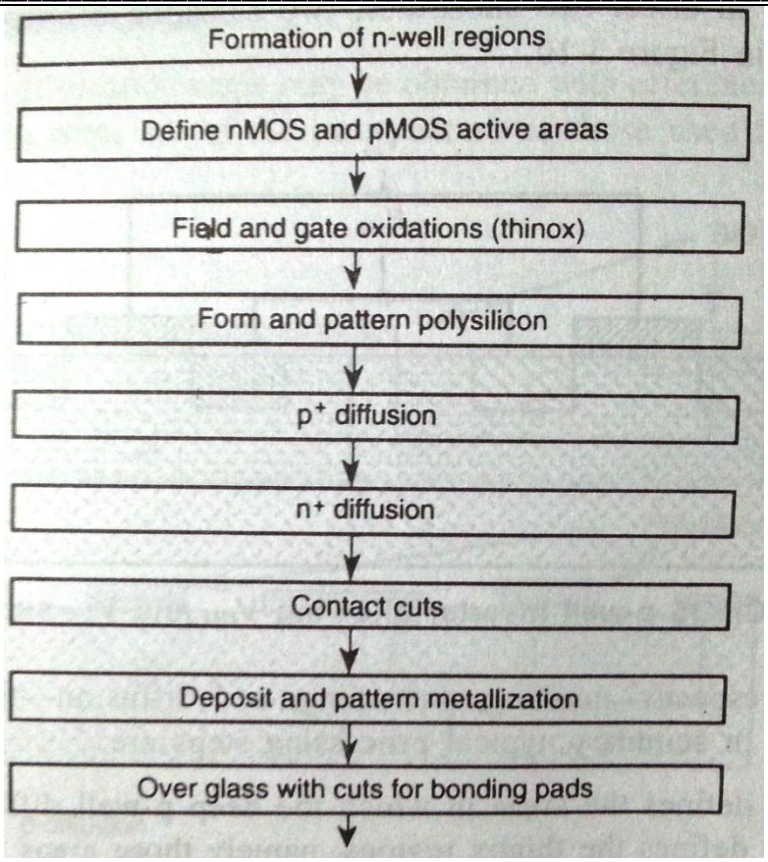
Ans:

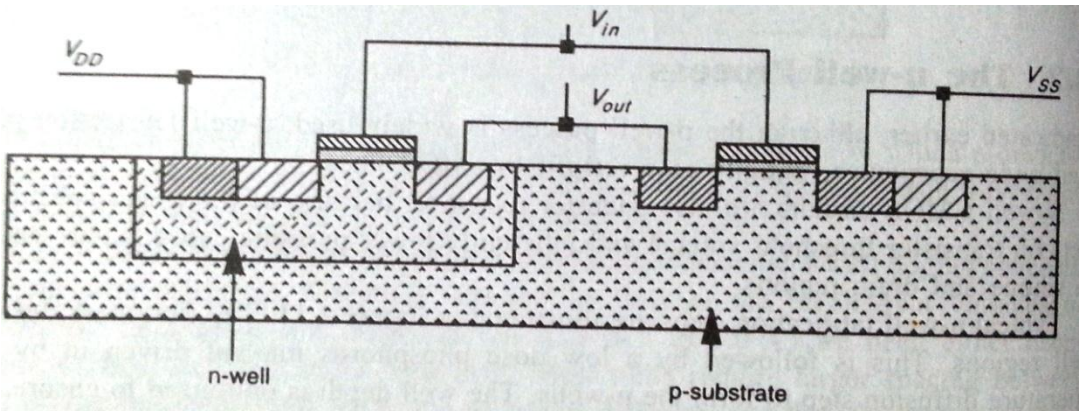


(Diagram: 2 Marks, Explanation: 2 Marks)

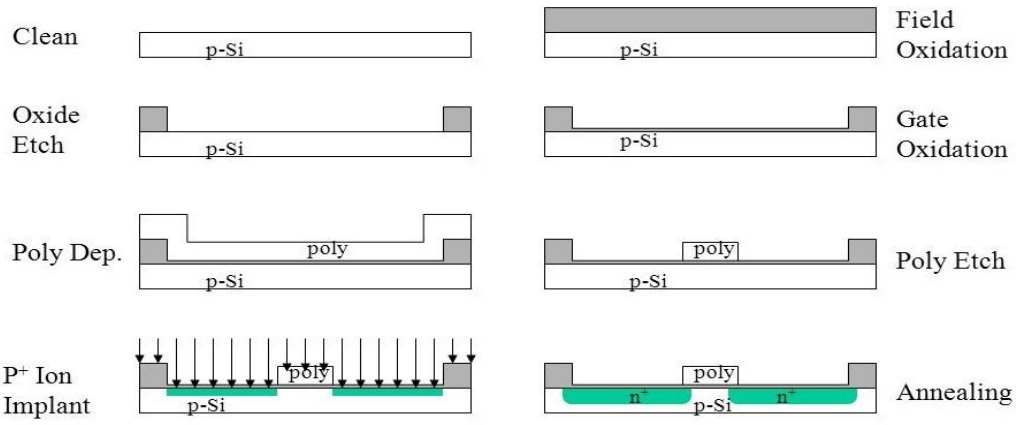
		Explanation:- ***(Explanation need not to write all the steps)*** Step1- Initially thick SiO2 layer is grown on the entire n-type silicon surface. Step2- Using photoresist layer and a mask, which defines the area of deep p-well diffusion, the SiO2 layer is etched off. Step3- Oxide in the p transistor region is removed and thin oxide layer is grown all over the surface. Step4- The polysilicon is patterned on thin oxide regions using a Mask forming pMOS and nMOS gates. Step5- The area of nMOS transistor is covered using p+ mask, and p diffusion is carried out to form source and drain of pMOS transistor. Step6- Using negative p+ mask the area of pMOS transistor is covered and n diffusion is carried out to form source and drain of nMOS transistor. Step7- Thick oxide layer is grown on the entire chip surface and the transistor areas where contact cuts are to be made are defined. Step8- The metal layer is deposited on the entire chip surface and is patterned.	
	f)	**Write the output equation of Moore and Mealy machines. List any 2 examples for FSM.**	**4 M**
	Ans:	Output Equation for Moore: f (o/p) = f (P.S.) Output Equation for Moore: f (o/p) = f (i/p, P.S.) **Examples: (Any relevant examples marks to be given)** 1. Vending machines, 2. Elevators, 3. Traffic lights, 4. Combinational Locks	**(Output equation: 2 Marks, Examples :2 Marks)**
Q. 4		**Attempt any THREE of following:**	**12 M**
	a)	**Write VHDL code to implement 4-bit adder.**	**4 M**
	Ans:	library IEEE; use IEEE.STD_LOGIC_1164.ALL; use IEEE.STD_LOGIC_ARITH.ALL; use IEEE.STD_LOGIC_UNSIGNED.ALL; entity FullAdder1 is Port (X : in std_logic; Y : in std_logic; CIN : in std_logic; SUM : out std_logic; CARRY : out std_logic); end FullAdder1; architecture Behavioral of FullAdder1 is begin Process (X,Y,CIN) begin SUM <= X XOR Y XOR CIN; CARRY<=(X AND Y) OR (X AND CIN) OR (Y AND CIN); end process; end Behavioral;	**(Any other relevant statement marks to be given)**
	b)	**Explain the following terms (1)Event scheduling (2) Simulation cycle**	**4 M**
	Ans:	**No need of writing all the points** • Events are changes to the wire or registers. Statements can schedule event to occur at particular time or to be triggered by other events in current time slot or at later simulation time.	**(Event Scheduling:2 Marks, Simulation**

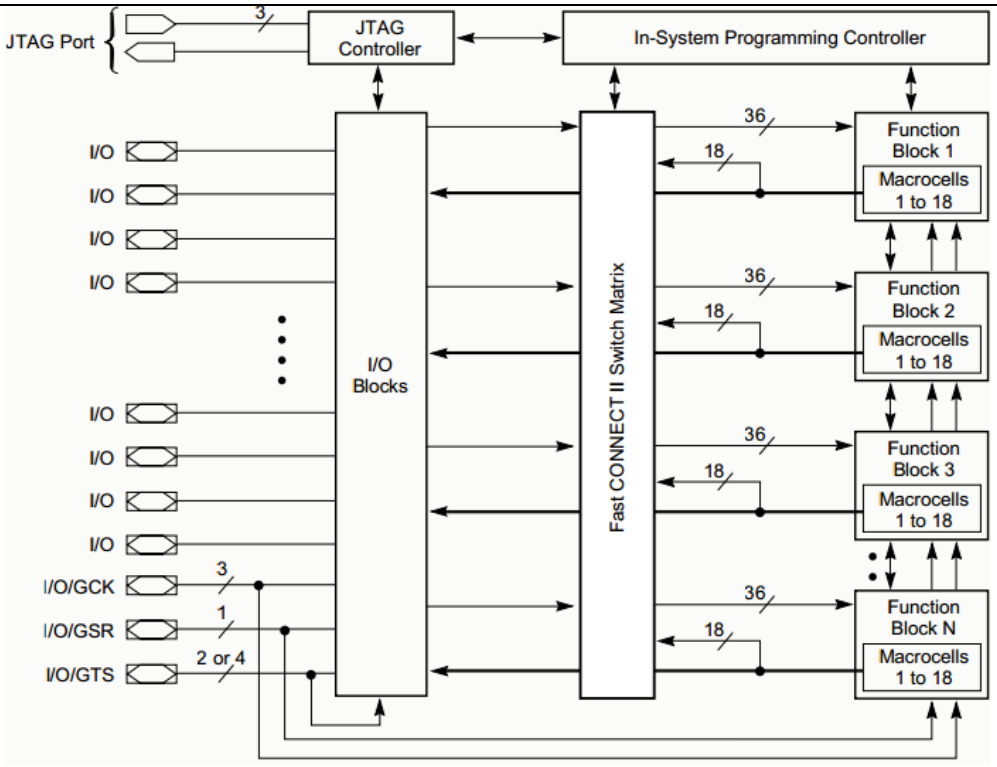
		• The events queue is segmented into five different regions. Each event will be added to one of the five regions in the queue but are only removed from the active region. 1. **Active events:** This event occurs at current simulation time and can be processed in any order. 2. **Inactive events:** This event occurs at current simulation time but shall be processed after all active events are processed. 3. **Non-blocking assign update event:** This event is evaluated during some previous simulation time, but shall be assigned at this simulation time after all active and inactive events are processed. 4. **Monitor event:** This event is processed after all active events, inactive events and non-blocking assign update events are processed. 5. **Future events:** This event occurs at some future simulation time. Future events are divided into future inactive event and future non-blocking assign update event. • Processing all active events is called simulation cycle. **Simulation cycle :** Some designs are self-simulating and do not need any external stimulus, but in most of the cases VHDL designers use VHDL test bench to drive the design being tested. Test bench is used to verify the functionality or correctness of a HDL model. Tt is a specification in HDL that plays the role of a complete simulation environment for the analyzed system. A test bench is at the highest level in the hierarchy of the de&gn. It instantiates the design under test (DUT) and provides the necessary input stimulus to DUT and examines the output from DUT. The stimulus driver drives input to the DUT. DUT responds to the input signals and produces output. Finally, it compares the output results from DUT with the expected values and reports any discrepancies.	**cycle:2 Marks)**
	c)	**Draw CMOS transistor fabrication using n-well process**	**4 M**
	Ans:	• The n-well CMOS circuits are also superior to p-well because of the lower substrate bias effect on transistor threshold voltage and inherently lower parasitic capacitance associated with source and drain regions. • The typical n-well fabrication steps are shown below. • The first mask defines the n-well regions. This is followed by a low dose phosphorous implant driven in by a high temperature diffusion step to form the n-well. • The well depth is optimized to ensure against p-substrate to p+ diffusion breakdown without compromising the n-well to n+ mask separation. • The next steps are to define the devices and diffusion paths, grow field oxide, deposit and pattern the polysilicon, carry out the diffusion, make contact cuts and finally metallise.	**(Diagram :2 Marks, Explanation : 2 Marks, Relevant steps marks can be given)**

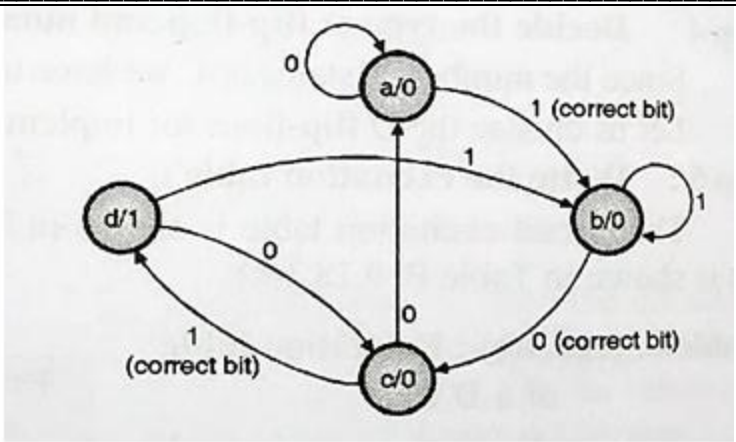




(Note:Diagram optional)



	d)	**Explain the following terms (1)Architecture (2)Configuration**	**4 M**
	Ans:	**Architecture** : • All entities that are declared have an architecture associated with it. Architecture describes the behavior of the entity. An entity can have multiple architectures. • Architecture assigned to an entity describes internal relationship between input and output of the entity. First part of the architecture may contain declaration of types, signals, constants, subprograms etc. **Configuration :** A configuration statement is used for binding a component instance to an entity architecture pair, when there are multiple architectures for a single entity.	**[Architecture : 2 Marks, Configuration:2 Marks]**
	(b)	**Attempt any One of the following**	**6 M**
	(a)	**Draw architecture of XC9500 CPLD.**	**6M**
	Ans:		**(Diagram:6 Mark)**
	(b)	**Design a sequence detector to detect the sequence 101.**	**6M**
	Ans:	**(This solution is designed by Moore logic, marks to be given if designed by Mealy logic, any other relevant method marks to be given)** • A sequence detector is a sequential state machine. In a Moore machine, output depends only on the present state and not dependent on the input (x). Hence in the diagram, the output is written with the states. • The state diagram of a moore machine for a 101 detector is:	**(Design:3 Marks, Truth Table:3 Marks)**



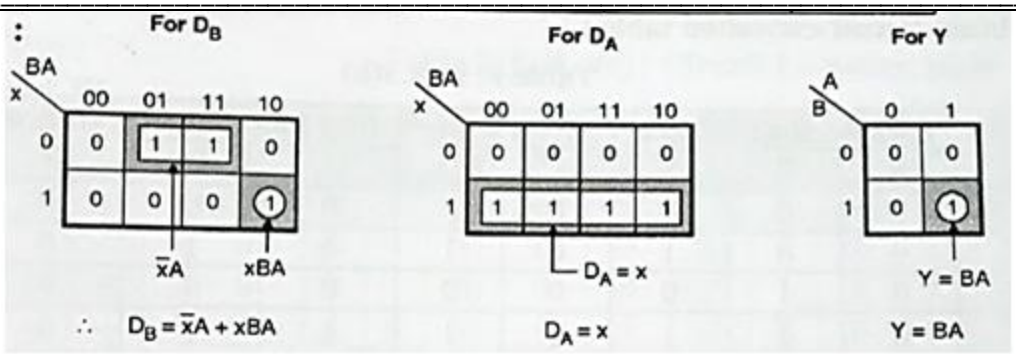
- The state table for the above diagram:

Present state	Next state		Output	
	x = 0	x = 1	x = 0	x = 1
a	a	b	0	0
b	c	b	0	0
c	a	d	0	1
d	c	b	0	0

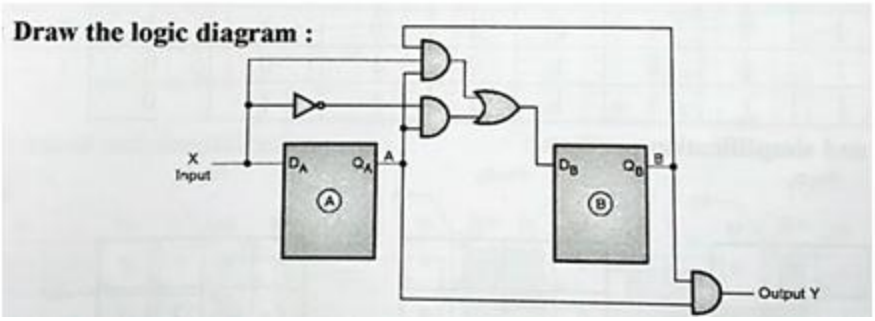
- Four states will require two flip flops. Consider two D flip flops. Their excitation table is shown below.
- Excitation table:

Input	Present state		Next state		F/F inputs		Output
x	B	A	B + 1	A + 1	D_B	D_A	Y
0	0	0 (a)	0	0 (a)	0	0	0
0	0	1 (b)	1	0 (c)	1	0	0
0	1	0 (c)	0	0 (a)	0	0	0
0	1	1 (d)	1	0 (c)	1	0	0
1	0	0 (a)	0	1 (b)	0	1	0
1	0	1 (b)	0	1 (b)	0	1	0
1	1	0 (c)	1	1 (d)	1	1	0
1	1	1 (d)	0	1 (b)	0	1	1

- K-maps to determine inputs to D Flip flop:



- Circuit diagram for the sequence detector:

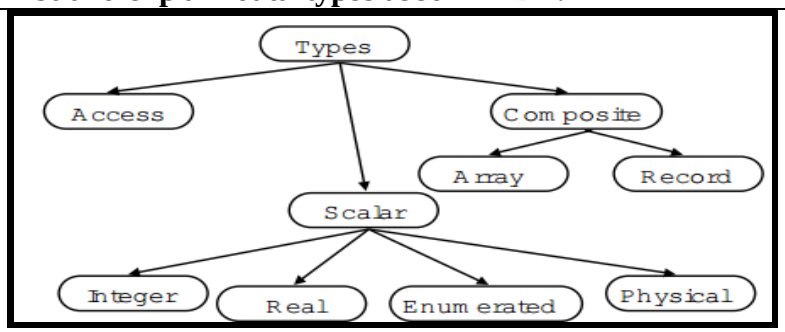


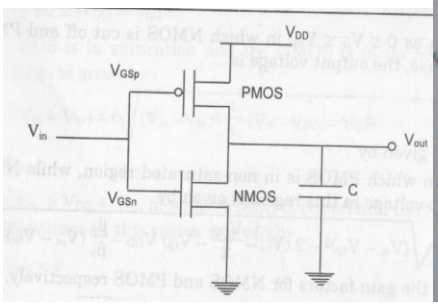
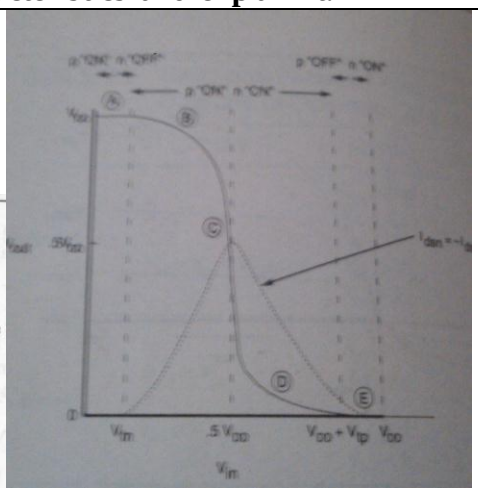
Q.5		Attempt any Four of the following:	16 M
	a)	Differentiate FPGA and CPLD.	4 M

Ans: **Comparison:**

(4 marks for any 4 points)

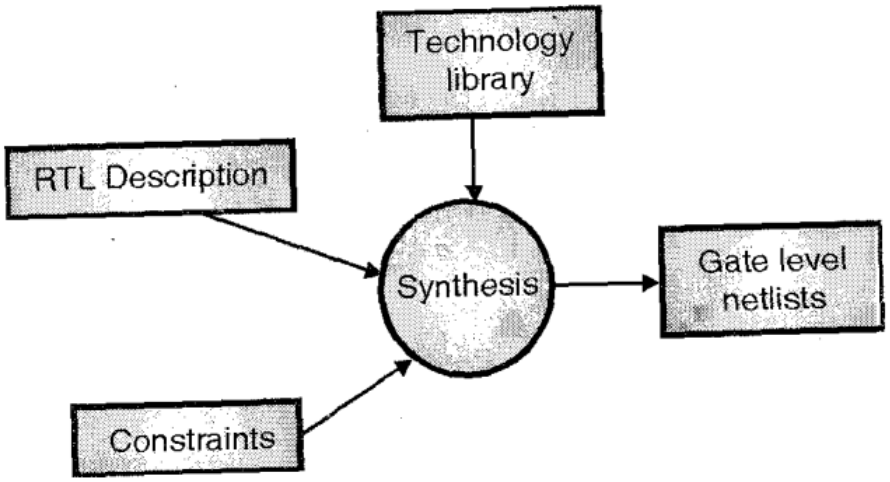
Sr. No.	FPGA	CPLD
1	It is field programmable gate arrays.	It is complex programmable logic device.
2	Capacity is defined in terms of number of gates available.	Capacity is defined in terms of number of macro-cells available.
3	FPGA consumes less power than CPLD	CPLD consumes more power than FPGA devices.
4	Numbers of input and output pins on FPGA are less than CPLD.	Numbers of input and output pins on CPLD are high.
5	FPGA is suitable for designs with large number of simple blocks with few numbers of inputs.	CPLD are ideal for complex blocks with large number of inputs.
6	FPGA based designs require more board space and layout complexity is more.	CPLD based designs need less board space and less board layout complexity.
7	It is difficult to predict the speed performance of design.	It is easier to predict speed performance of design.
8.	FPGA are available in wide density range.	CPLDs contain fewer registers but have better performance.

	b)	**List and explain data types used in HDL.**	**4 M**
	Ans:	 There are mainly two types: 1. Scalar 2. Composite Scalar Types : 1. Integer: Defines the value with Integer. (Integer Range). 2. Real: Defines the value with number. 3. Enumerated: Defines the set of user defined values consisting of identifiers and character literals 4. Physical :used to represent physical quantities e.g. distance ,time Composite : 1. Array: Contain many elements of same type. 2. Record : Contain elements of different types	**(1 Mark for List. 1.5 +1.5 Marks for Scalar and Composite)**
	c)	**Explain in cycle based simulation.**	**4 M**
	Ans:	• Cycle-based simulation ignores intra-cycle state transitions, i.e. they check the status of target signals periodically irrespective of any events. This can boost performance by 5 to 10 times compared to traditional event-driven simulators. • Cycle-based technology offers greater memory efficiency and faster simulation run-time than traditional pure event-based simulators. • Cycle-based simulators work best with synchronous design but give less timing accuracy with asynchronous design. • Signals are treated as variables. Functions such as AND, OR etc. are directly converted to program statements. • Signal level functions such as memory blocks, adders, multipliers etc. are modeled as subroutines. • For every input vector, the code is repeatedly executed until all variables have attained steady value. • Compiled code simulator is efficient when used for high-level design verification. Inefficiency is incurred by the evaluation of the design when only few inputs are changing. OR 1. All events are postponed till the end of clock Cycle. 2. All results are evaluated only once at the end of each clock cycle. 3. Timing information is not maintained. 4. Suitable for Synchronous designs.	**(Each point: 1 Marks)**

	d)	**Draw the CMOS inverter characteristics and explain it.**	**4 M**
	Ans:	 CMOS Inverter  DC Transfer Characteristic Fig shows the characteristic of CMOS inverter . **Region A** When input voltage $0 \leq Vin \leq Vin$ N device is cut off and p device in linear range.Vout= Vdd. **Region B** $Vin \leq Vin \leq Vdd/2$ Vout is as per voltage equation P device is in non saturation region n device is in saturation **Region C** N and P devices in saturation Vout= vin **Region D** $Vdd/2 < Vin \leq Vdd+Vtp$ P device is in saturation and n device is in operating in non saturation **Region E** $Vin \geq Vdd-Vtp$ P device cutoff n device in linear mode Vout =0.	**(Description: 2 Marks, Characteristics:2 Marks)**
	e)	**Explain shift operators with example.**	**4 M**
	Ans:	**Shift operators**: sll shift left logical srl shift right logical sla shift left arithmetic sra shift right arithmetic rll right left logical rrl right right logical Example : count <=count sll 1 contain of the count is shifted to left by '1'. Any such examples for other operators	**(Any four with example Each operator: 1 Marks)**
	f)	**What is event scheduling and zero modelling.**	**4 M**
	Ans:	**Event scheduling:** Event is nothing but change on target signal which is to be updated. Ex. X<= a after 0.5ns when select=0 else	**(Event scheduling :2 Marks**

		X<= b after 0.5ns The assignment to signal x does not happen instantly. Each of the values assigned to x contain an **after** clause. The mechanism for delaying the new value is called scheduling an event. By assigning port x a new value, an event was scheduled 0.5ns in the future that contains the new value for signal x. when the event matures, signal receives a new value. **Zero Modelling :** The ordering of zero delay events is handled with a fictitious unit called delta time. Delta time represents the execution of a simulation cycle without advancing Simulation time. The simulator models zero-delay events using delta time. Events scheduled at the same time are simulated in specific order during a delta time step. Related logic is then re-simulated to propagate the effects for another delta time step. Delta time steps continue until there is no activity for the same instant of simulated time. **OR** In VHDL zero delay circuits and designs that depends on zero delay components can never be built. Simulation deltas are used to order some types of events during simulation. Specifically zero delay events must be ordered to produce consistent results. If they are not properly ordered results can vary between different simulation runs.	**, zero modeling:2 Marks)**
Q.6		**Attempt any FOUR of following:**	**16 M**
	a)	**Explain oxidation and diffusion process in fabrication process.**	**4 M**
	Ans:	**OXIDATION:** Oxidation is a process by which a layer of silicon dioxide is grown on the surface of a silicon wafer. The oxidation of silicon is necessary throughout the modern integrated circuit fabrication process. The oxidation of silicon is achieved by heating silicon wafers in an oxidizing atmosphere such as oxygen or water vapor .There are two types 1. Wet oxidation 2. Dry oxidation **Diffusion :** The process of junction formation, which is transition from p to n type or vice versa. Diffusion of impurity atoms into silicon crystal takes place only at elevated temperature, typically 900 to 1100°C.The following material are used **P type : Borane(B_2H_6)** **N type: Phosphine(PH_3)**	**(Oxidation:2 Marks, Diffusion: 2 Marks)**
	b)	**Define the following terms related to VHDL (1)Package (2) Entity**	**4 M**
	Ans:	**Package :** Contains frequently used declarations, constants, functions, procedures, users data types, subprograms and components. **Entity :** A design's interface signals to the external circuitry i.e. declaration of inputs and outputs of the design .	**(Package : 2 Marks, Entity : 2 Marks)**
	c)	**Explain HDL design flow for synthesis.**	**4 M**

Ans:



Synthesis is an automatic method of converting higher level of abstraction to lower level of abstraction.

1.The process that converts user, hardware description into structural logic description. Synthesis is a means of converting hdl into real world hardware. It generates a gate level net list for the target technology. The synthesis tool converts register transfer level (RTL) description to gate level netlist. These gate level netlists consist of interconnected gate level macrocells.

2.The inputs to the synthesis process are RTL (register transfer level) VHDL description, circuit constraints and attributes for the design, and a technology library.

3.The synthesis process produces an optimized gate level net list from all these inputs. The translation from RTL description to Boolean equivalent description is usually not user controllable.

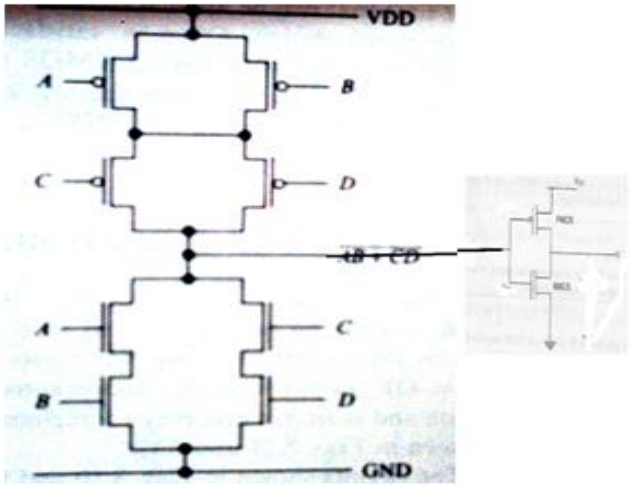
4.The intermediate form that is generated is a format that is optimized for a particular tool and may not even be viewable by the user. All the conditional signal assignments and selected signal assignment statements are converted to their boolean equivalent in this intermediate form. The optimization process takes an un optimized Boolean description and converts it to an optimized Boolean description. For this it uses number of algorithm and rules. This process aims to improve structure of Boolean equations by applying rules of boolean algebra. This removes the redundant logic and reduces the area requirement.

OR

Simple steps

1. Describe your design with HDL
2. Perform RTL simulation
3. Synthesizing your design
4. Create Xilinx Netlist Files (XNF/EDIF etc)
5. Perform Functional Simulation
6. Floor planning of design (optional)
7. Placing and routing
8. Perform a timing simulation (post layout)

(2 Marks for flow 2Marks for description)

	Note :flow may vary	
d)	**Explain the following terms (1) Delta Delay (2) Sensitivity list**	**4 M**
Ans:	**Delta Delay** The real time that the simulator takes to execute one simulation cycle is known as delta delay for simulation delta with zero simulation time.A delta delay is very small and does not correspond to any real delay and actual simulation time does not advance. Delta delay is introduced to achieve concurrency and order independency . The simulator freezes simulation time until all scheduled assignments in current simulation time is finished and there are no more events in the sensitivity list. **Sensitivity List** Every concurrent statement has a sensitivity list. Statements are executed only when there is an event or signal in the sensitivity list, otherwise they are suspended. Ex. F<=a and b; A and b are in the sensitivity list of f. the statement will execute only if one of these will change. Ex. Process(clk, RST) The process is sensitive to RST and clk signal i.e. an event on any of these signals will cause the process to resume.	**(Delta Delay :2Marks, Sensitivity List: 2Marks,)**
e)	**Execute the following equation by the circuit with CMOS logic. D =[(A.B) +(C.D)]**	**4 M**
Ans:		**(Diagram:4 Marks)**
f)	**What is meant by efficient coding style? How arithmetic expressions are optimized?**	**4 M**
Ans:	There may be more than one method to model a particular design part but only a few would yield better performance. The essence of VHDL coding lies in understanding which style yields the ultimate performance under the given set of specifications. The key to higher performance is to avoid writing code that needlessly creates additional work for the HDL compiler and synthesizer, which, in turn, generates designs with greater number of gates. Basically, any coding style that gives the HDL simulator information about the design. **Optimising Arithmetic Expressions** :(2 Marks)	**(Note:It should describe Writing Technology-Independent HDL:2 Marks, Optimising**

		Design compliers use the properties of arithmetic operators such as associative and cumulative to rearrange an expression so that it results in an optimized implementation. The uses following methods 1. Merging cascaded adders with carry 2. Arranging Expression trees for minimum delay **3.** Sharing of common sub expressions and parenthesis	**Arithmetic Expressions 2 Marks)**