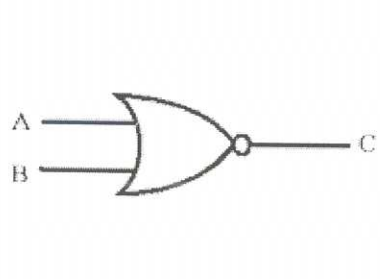
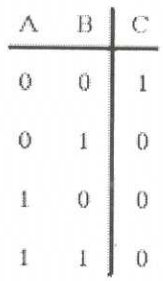
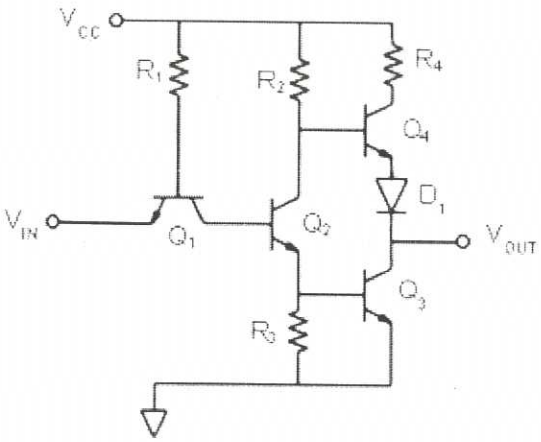
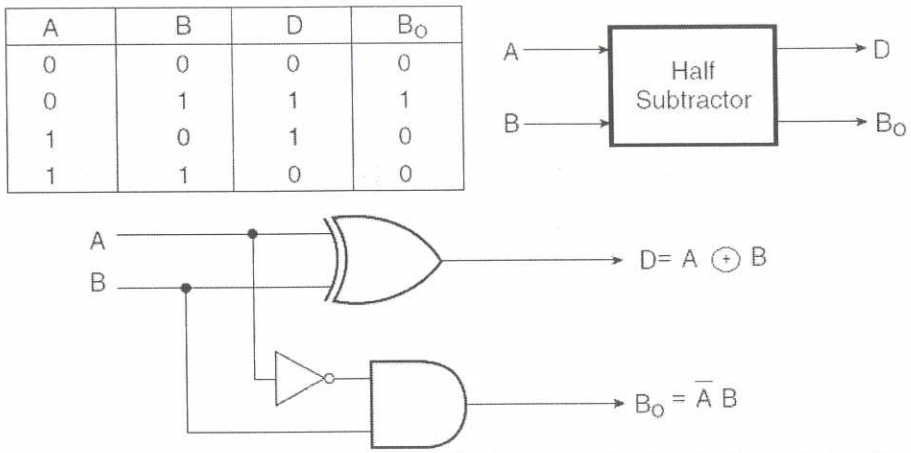
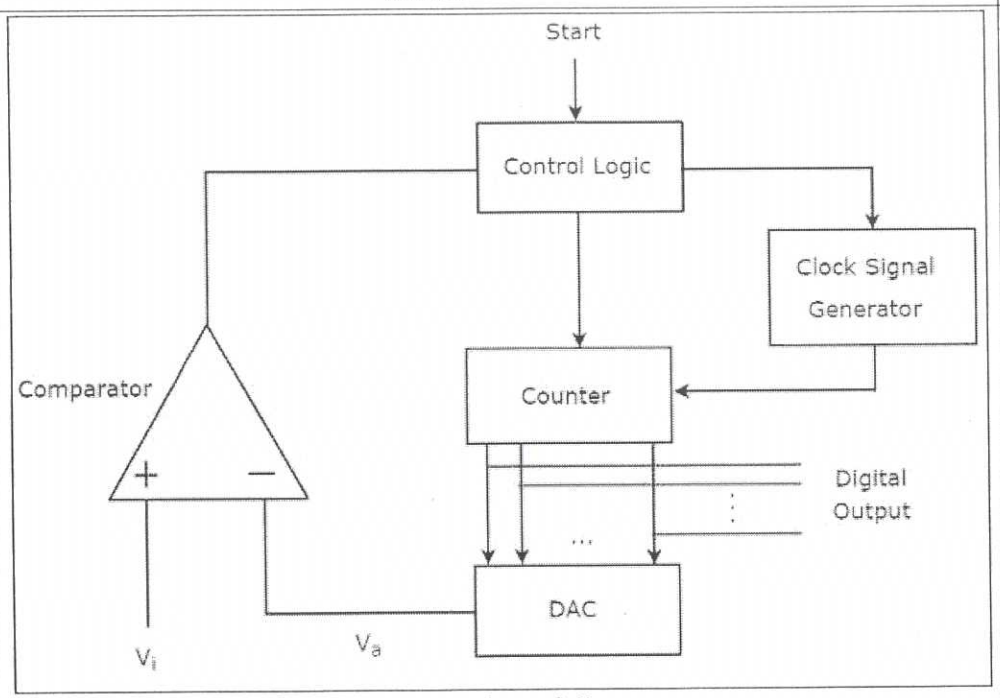
Scheme of evaluation

Course : Digital Electronics
Version : 2015
Code : 3042

Scoring indicators

Qn No.	Scoring indicator	Split Score	Sub total	Total
I 1	ASCII & EBCDIC	1+1	2	10
I 2	**Propagation delay**, t_{pd}, is the time required for a digital signal to travel from the input of a logic **gate** to the output.		2	
I 3	Frequency counters Analog to digital convertors. Digital clocks Digital triangular wave generator. Generating staircase voltage Frequency divider circuits of the clock signals.	Any two	2	
I 4	Settling time represents the time it takes for the output to settle within a specified band ±(1/2) LSB of its final value, after the change in digital input.		2	
I 5	 	1+1	2	
II 1	 When Vin =0, Q1 forward biased, so Q2 is off, Q4's base gets pulled up by R2, so Q4 is on (but not saturated) and Q3 is off as it gets no base current and its base is pulled down by R3. Therefore output will be 1. When Vin = 1, Q1 is off, Q2 is on. It sends its emitter current into Q3, turning it on Q4 is off. So output will be 0	Diagram (3) + explanation (3)	6	30

II 2	BCD: In this code each decimal digit is represented by a 4-bit binary number. BCD is a way to express each of the decimal digits with a binary code. In the BCD, with four bits we can represent sixteen numbers (0000 to 1111). But in BCD code only first ten of these are used (0000 to 1001). The remaining six code combinations i.e. 1010 to 1111 are invalid in BCD XS-3: Excess-3 codes are unweighted and can be obtained by adding 3 to each decimal digit then it can be represented by using 4 bit binary number for each digit. The codes 0000,0001,0010,1101,1110 and 1111 are not used for any digit. Gray Code: It is the non-weighted code and it is not arithmetic codes. That means there are no specific weights assigned to the bit position. It has a very special feature that, only one bit will change each time the decimal number is incremented. As only one bit changes at a time, the gray code is called as a unit distance code. The gray code is a cyclic code. Gray code cannot be used for arithmetic operation.	2+2+2	6	
II 3		Diagram (2) + Explanation (2)+ TT(2)	6	
II 4	 The **working** of a counter type ADC is as follows – The **control logic** resets the counter and enables the clock signal generator in order to send the clock pulses to the counter, when it received the start commanding signal. The **counter** gets incremented by one for every clock pulse and its value will be in binary (digital) format. This output of the counter is applied as an input of DAC. **DAC** converts the received binary (digital) input, which is the output of counter, into an analog output. Comparator compares this analog value, V_a with the external analog input value V_i. The **output of comparator** will be **'1'** as long as Vi is greater than Va. The operations	Diagram (3) + Explanation (3)	6	

mentioned in above two steps will be continued as long as the control logic receives '1' from the output of comparator.

The **output of comparator** will be **'0'** when V_i is less than or equal to V_a. So, the control logic receives '0' from the output of comparator. Then, the control logic disables the clock signal generator so that it doesn't send any clock pulse to the counter. At this instant, the output of the counter will be displayed as the **digital output**. It is almost equivalent to the corresponding external analog input value V_i

II 5.

A basic 3-bit up/down synchronous counter.

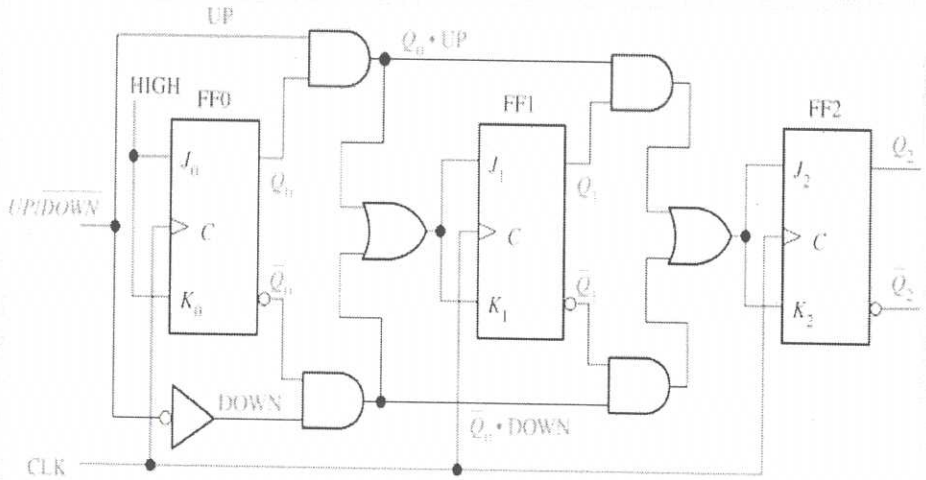
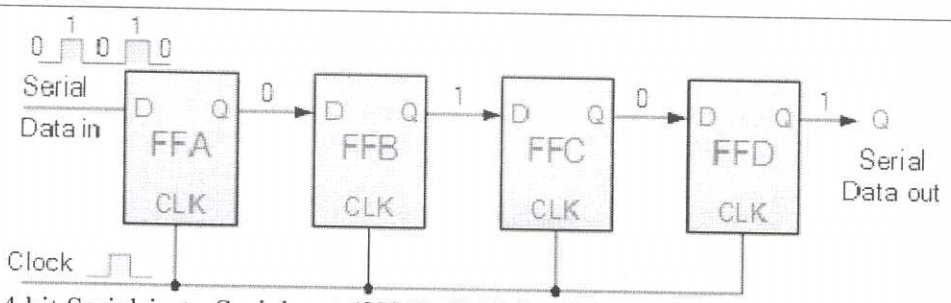


Diagram (3) + Explanation (3)

6

II 6.



4 bit Serial-in to Serial-out (SISO) Shift Register

In this shift register the data is allowed to flow straight through the register and out of the other end. Since there is only one output, the DATA leaves the shift register one bit at a time in a serial pattern, hence the name Serial-in to Serial-Out Shift Register or SISO.

The SISO shift register is one of the simplest of the four configurations as it has only three connections, the serial input (SI) which determines what enters the left hand flip-flop, the serial output (SO) which is taken from the output of the right hand flip-flop and the sequencing clock signal (Clk). The logic circuit diagram above shows a generalized serial-in serial-out shift register.

Diagram (3) + explanation (3)

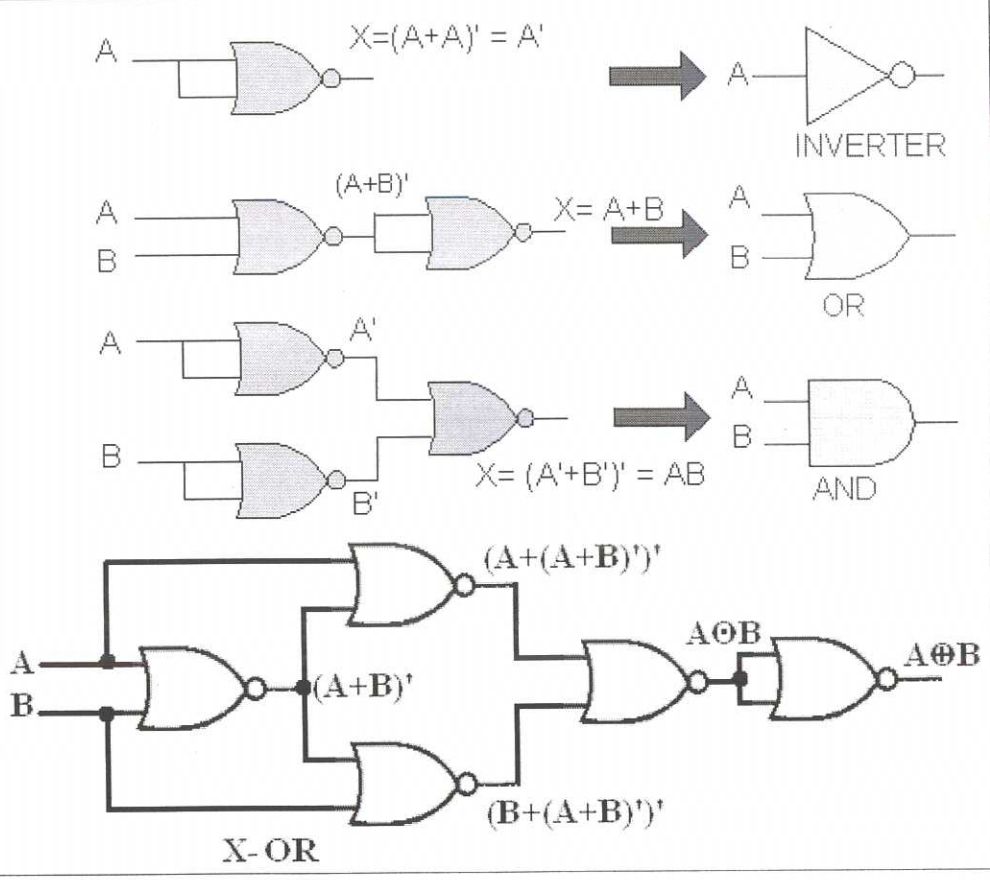
6

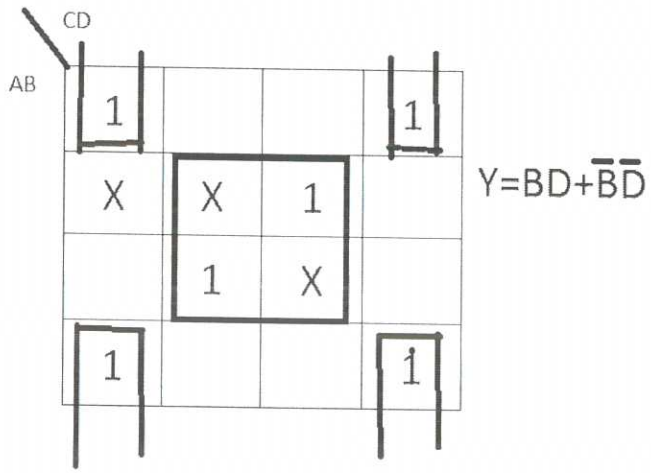
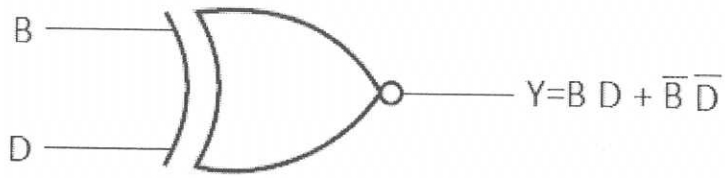
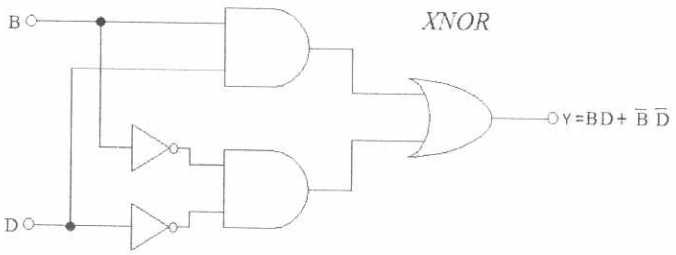
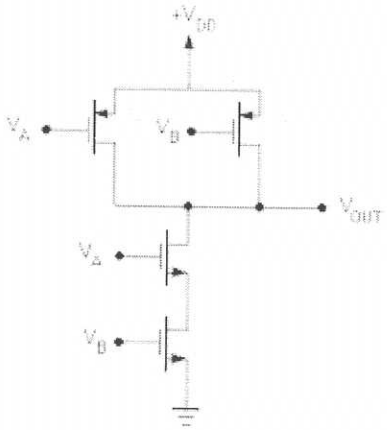
II 7.

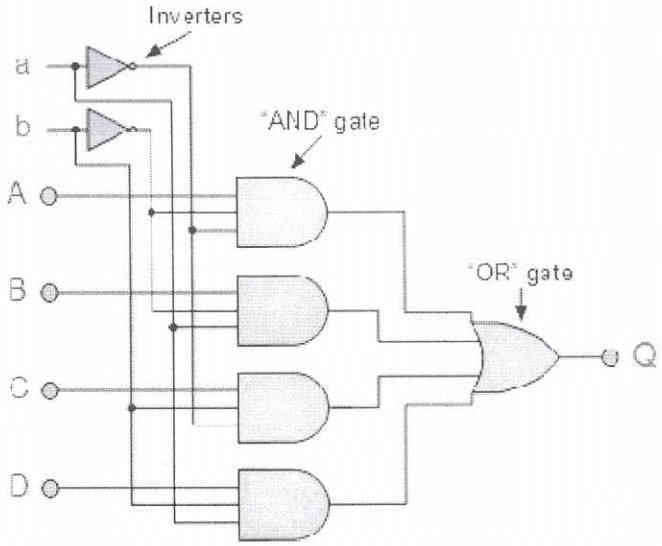
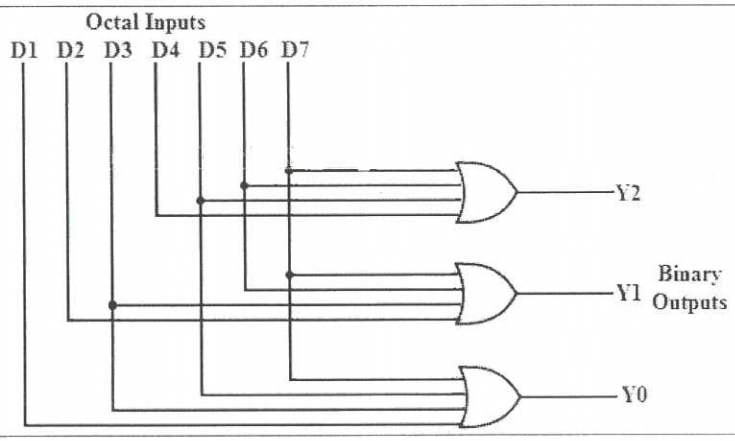
Synchronous Sequential Circuit	*Asynchronous Sequential Circuit*
Synchronous sequential circuits are digital circuits governed by clock signals.	Asynchronous sequential circuits are digital circuits that are not driven by clock. They can be called as *self-timed circuits.*
The transition from one state to another takes place only by the application of specified clock signal, even if the inputs change.	The transition from one state to another takes place immediately once the inputs change.
The states of synchronous sequential circuits are always predictable and thus reliable.	There are chances for the asynchronous circuits to enter into a wrong state because of the time difference between the arrivals of

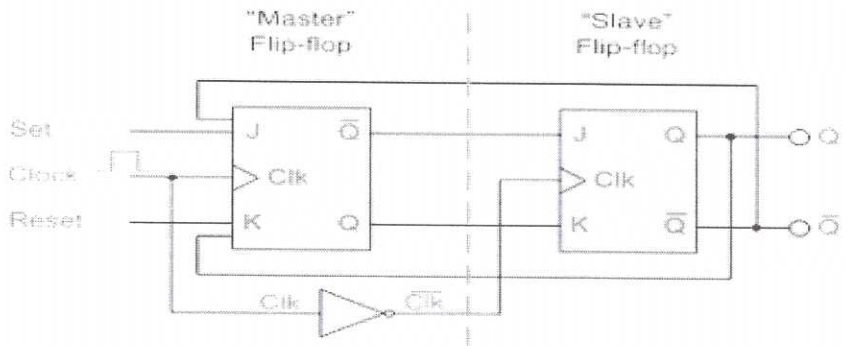
Any 6

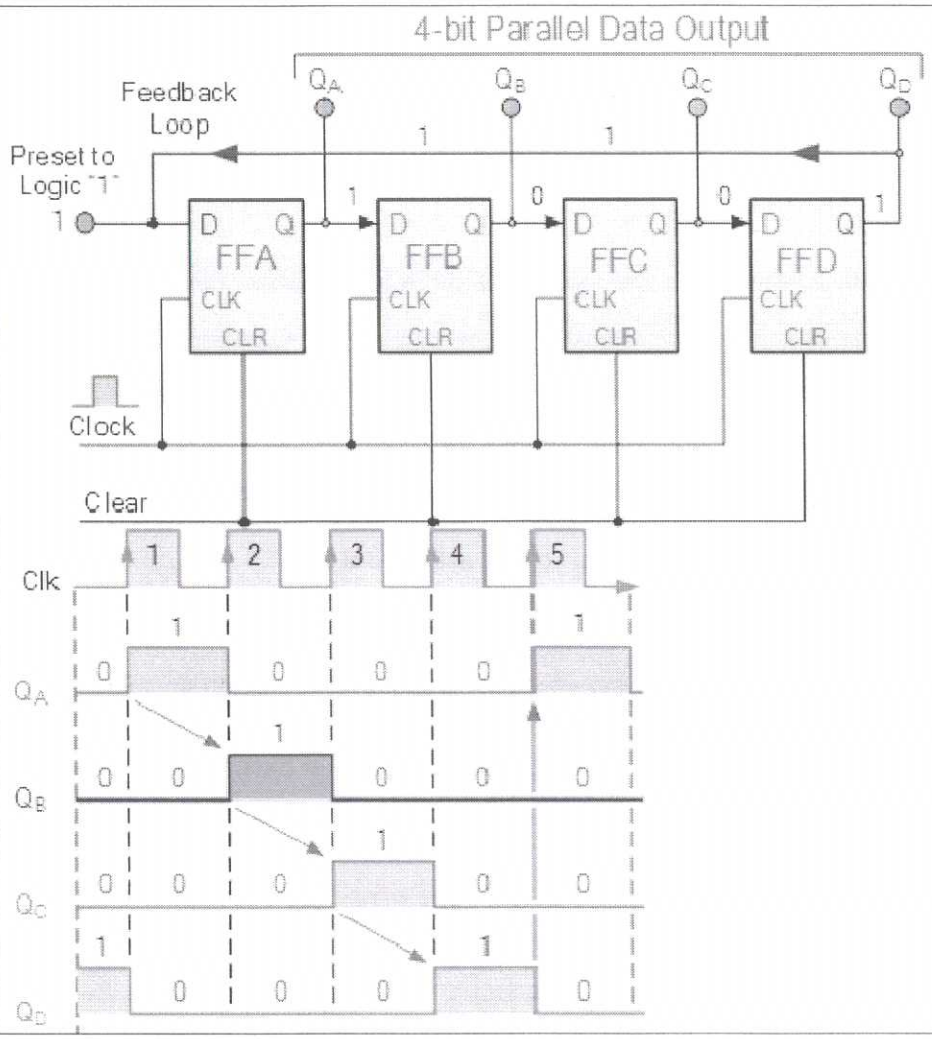
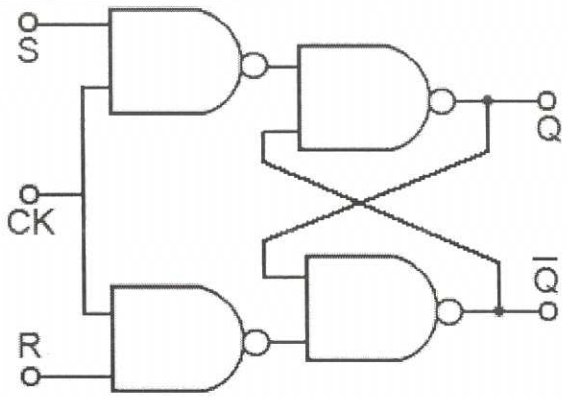
6

		inputs. This is called as *race condition.*			
	It is easy to design synchronous sequential circuits.	The presence of feedback among logic gates causes instability issues making the design of asynchronous sequential circuits difficult.			
	The synchronous sequential circuits are slower in its operation speed. This is due to the propagation delay of clock signal in reaching all elements of the circuit.	The asynchronous sequential circuits are comparatively faster. Here, there is no clock signal but only the propagation delay of logic gates.			
	The distributed clock signal consumes large power and dissipates large amount of heat.	Power consumption and heat dissipation are comparatively lower.			
	Synchronous circuits are used in counters, shift registers, memory unit. Synchronous counters are used in digital clocks, digital locking system, keyboard controller, frequency counter etc. Shift registers are used in data transfer and also to introduce time delay into circuits.	Asynchronous circuits are used in low power and high speed operations. They are employed in simple microprocessors, digital signal processing units and in communication systems for email applications, internet access and networking.			
	PART C				
III (a)			NOT (1) + remaining (2 each)	7	15
III (b)	i)13.625 ii)110011100 iii)001111111101 iv)11000.10011		4×2	8	

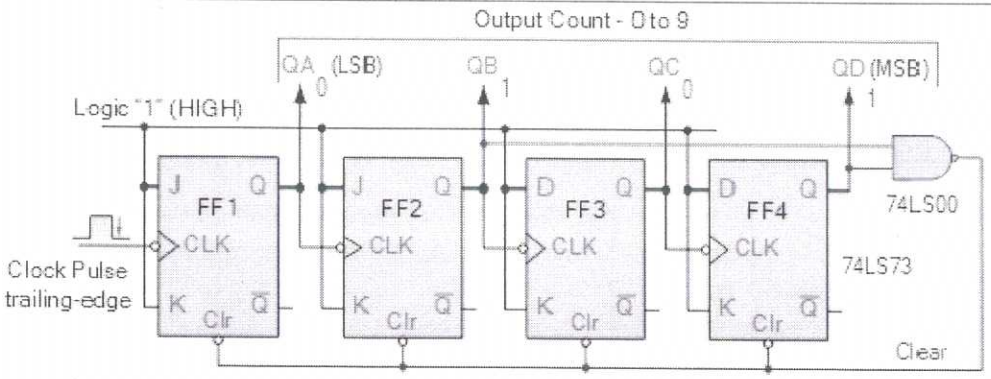
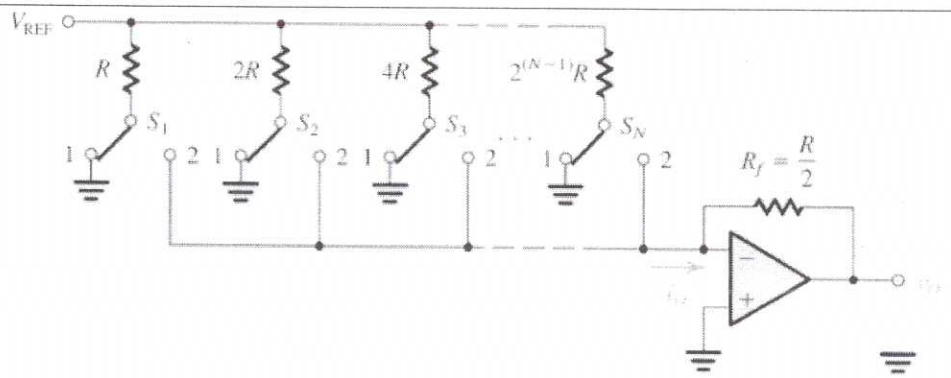
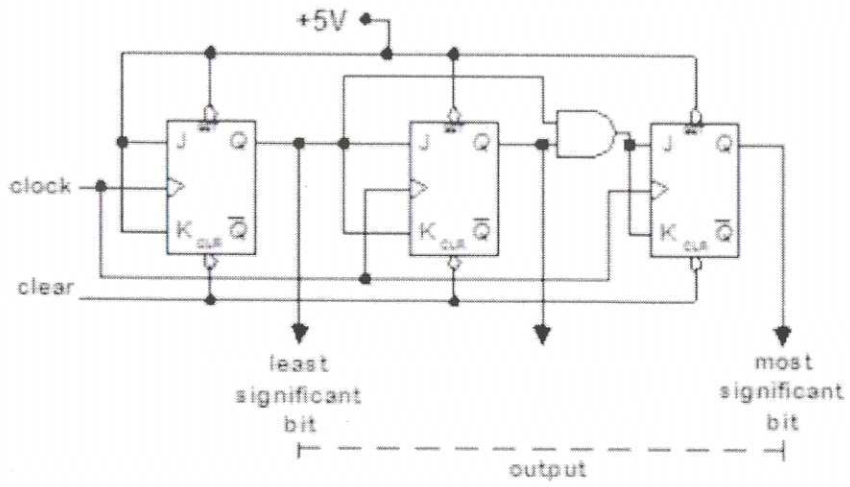
IV (a)	  	Mapping(2) +Grouping (2) + Reduction (3) + Implementation (2)	9	15
IV (b)	(i) $\overline{A}+B$ (ii) A(B+C) (iii) B+AC	3x2	6	
V (a)	**CMOS NAND Gate** The below figure shows a 2-input Complementary MOS NAND gate. It consists of two series NMOS transistors between V_{OUT} and Ground and two parallel PMOS transistors between V_{OUT} and VDD.  CMOS NAND Gate If either input A or B is logic 0, at least one of the NMOS transistors will be OFF, breaking the path from Y to Ground. But at least one of the pMOS transistors will be ON, creating a path from Y to VDD. Hence, the output Y will be high. If both inputs are high, both of the nMOS	Diag (4) + Exp (4)	8	15

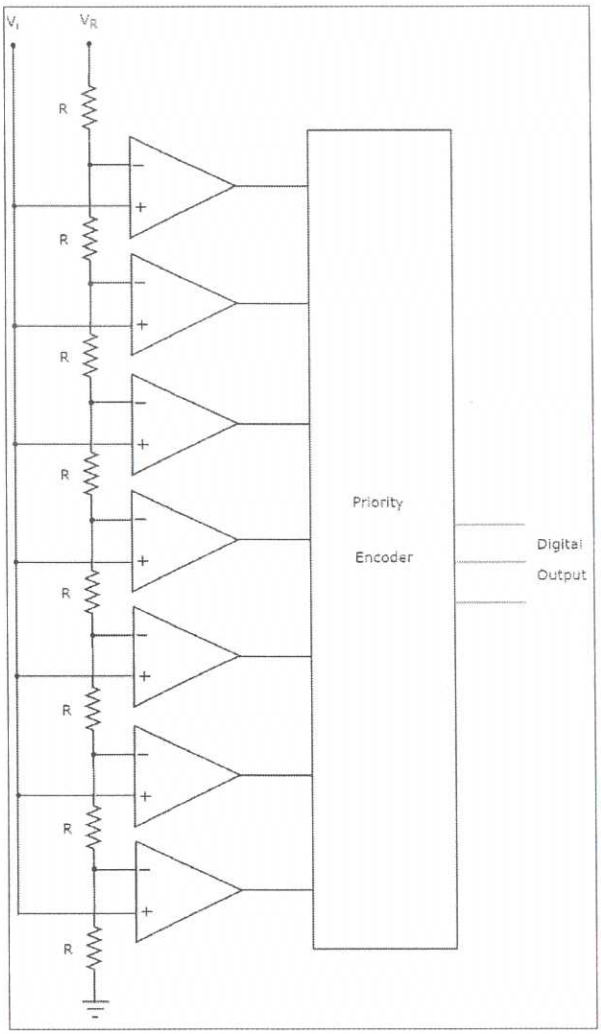
	transistors will be ON and both of the pMOS transistors will be OFF. Hence, the output will be logic low. The truth table of NAND logic gate given in below table. V_A | V_B | V_{OUT} 0 | 0 | 1 0 | 1 | 1 1 | 0 | 1 1 | 1 | 0			
V (b)	The Boolean expression for this 4-to-1 **Multiplexer** above with inputs A to D and data select lines a, b is given as: $Q = abA + abB + abC + abD$ In this example at any one instant in time only ONE of the four analogue switches is closed, connecting only one of the input lines A to D to the single output at Q. As to which switch is closed depends upon the addressing input code on lines "a" and "b". So for this example to select input B to the output at Q, the binary input address would need to be "a" = logic "1" and "b" = logic "0". Thus we can show the selection of the data through the multiplexer as a function of the data select bits as shown. 	Diag+ TT (4) + Exp (3)	7	
VI (a)		Diag (4) + Exp (4)	8	15

	<table><tr><th colspan="8">Input</th><th colspan="3">Output</th></tr><tr><th>D0</th><th>D1</th><th>D2</th><th>D3</th><th>D4</th><th>D5</th><th>D6</th><th>D7</th><th>Y2</th><th>Y1</th><th>Y0</th></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td></tr><tr><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td></tr><tr><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td></tr><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td></tr><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td></tr><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>1</td><td>0</td></tr><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td><td>1</td><td>1</td></tr></table>			
VI (b)	**High-level input voltage, VIH** : This is the minimum input voltage which is recognized by the gate as logic 1. **Low-level input voltage, VIL:** This is the maximum input voltage which is recognized by the gate as logic 0. **High-level output voltage, VOH:** This is the minimum voltage available at the output corresponding to logic 1. **Low-level output voltage, VOL:** This is the maximum voltage available at the output corresponding to logic 0. **Noise Margin**: Ability of the gate to tolerate fluctuations of the voltage levels. The input and output voltage levels defined above point. **Fan-in**: *Fan-in* is the number of input signals that can be connected to a gate without causing it to operate outside its intended operating range. **Fan-out**: *Fan-out* is the maximum number of inputs that can be driven by a logic gate.	7×1	7	
VII (a)	 Master Slave Flip Flop <table><tr><th rowspan="2">Case</th><th colspan="3">Inputs</th><th colspan="2">Outputs</th><th rowspan="2">Remark</th></tr><tr><th>CLK</th><th>J</th><th>K</th><th>Q_{n+1}</th><th>$\bar{Q}_{n+1}$</th></tr><tr><td>I</td><td>×</td><td>0</td><td>0</td><td>Q_n</td><td>$\bar{Q}_n$</td><td>No change</td></tr><tr><td>II</td><td>(1)</td><td>0</td><td>0</td><td>Q_n</td><td>$\bar{Q}_n$</td><td>No change</td></tr><tr><td>III</td><td>(1)</td><td>0</td><td>1</td><td>0</td><td>1</td><td>Reset</td></tr><tr><td>IV</td><td>(1)</td><td>1</td><td>0</td><td>1</td><td>0</td><td>Set</td></tr><tr><td>V</td><td>(1)</td><td>1</td><td>1</td><td>$\bar{Q}_n$</td><td>Q_n</td><td>Toggle</td></tr></table>	Diag (3) + TT (2) + Exp (3)	8	15

VII (b)		Diag (2) + TT (2) +Exp (3)	7	
VIII (a)		Diag (4) + TT (2) + Exp (3)	9	15
VIII (b)	When J=K=1, and frequency of the clock pulse is very low, the output of JK flip flop toggles more than once in same clock pulse. This condition is known as	3+3	6	

INPUTS			OUTPUT	STATE
CLK	S	R	Q	
X	0	0	No Change	Previous
↑	0	1	0	Reset
↑	1	0	1	Set
↑	1	1	-	Forbidden

	Race-around condition. **Steps to avoid racing condition in JK Flip flop:** 1. If the Clock On or High time is less than the propagation delay of the flip flop then racing can be avoided. This is done by using edge triggering rather than level triggering. 2. If the flip flop is made to toggle over one clock period then racing can be avoided. This introduced the concept of Master Slave JK flip flop. 3. Using high frequency clock pulses			
IX (a)		Diag (4) + ST (2) + Exp (2)	8	15
IX (b)	 $D = \frac{b_1}{2^1} + \frac{b_2}{2^2} + \cdots + \frac{b_N}{2^N}$ $\quad i_O = \frac{2V_{REF}}{R}D$ $\quad v_O = -i_O R_f = -V_{REF}D$	Diag (4) + Exp (3)	7	
X (a)		Diag (3) + ST (2) + Exp (3)	8	15

X (b)	 The **working** of a 3-bit flash type ADC is as follows. • The **voltage divider network** contains 8 equal resistors. A reference voltage V_R is applied across that entire network with respect to the ground. The voltage drop across each resistor from bottom to top with respect to ground will be the integer multiples (from 1 to 8) of V_R8. • The external **input voltage** V_i is applied to the non-inverting terminal of all comparators. The voltage drop across each resistor from bottom to top with respect to ground is applied to the inverting terminal of comparators from bottom to top. • At a time, all the comparators compare the external input voltage with the voltage drops present at the respective other input terminal. That means, the comparison operations take place by each comparator **parallelly**. • The **output of the comparator** will be '1' as long as V_i is greater than the voltage drop present at the respective other input terminal. Similarly, the output of comparator will be '0', when, V_i is less than or equal to the voltage drop present at the respective other input terminal. • All the outputs of comparators are connected as the inputs of **priority encoder**. This priority encoder produces a binary code (digital output), which is corresponding to the high priority input that has '1'. • Therefore, the output of priority encoder is nothing but the binary equivalent **(digital output)** of external analog input voltage, V_i. The flash type ADC is used in the applications where the conversion speed of analog input into digital data should be very high.	Diag (4) + exp (3)	7	