A

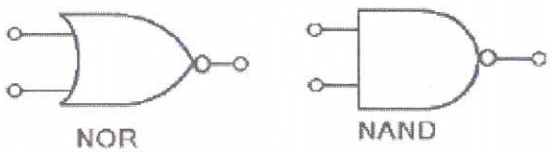
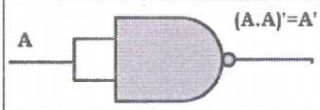
209 16/11/23

12 Nov-23

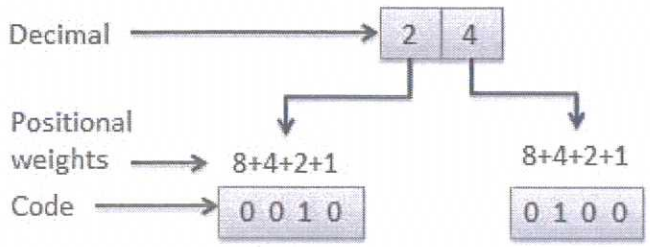
Scoring Indicators

COURSE NAME : DIGITAL COMPUTER FUNDAMENTALS

COURSE CODE : TED (21) – 3134 **QID : 2110220127**

<table>
<tr><th>Q No</th><th>Scoring Indicators</th><th>Split score</th><th>Sub Total</th><th>Total score</th></tr>
<tr><td></td><td>PART A</td><td></td><td></td><td>9</td></tr>
<tr><td>I. 1</td><td>American Standard Code for Information Interchange</td><td>1</td><td>1</td><td rowspan="10"></td></tr>
<tr><td>I. 2</td><td>0101 1001</td><td>1</td><td>1</td></tr>
<tr><td>I. 3</td><td> 
</td><td>2 x 0.5</td><td>1</td></tr>
<tr><td>I. 4</td><td>$\overline{(A \cdot B)} = \overline{A} + \overline{B}$
$\overline{A + B} = \overline{A} \cdot \overline{B}$</td><td>2 x 0.5</td><td>1</td></tr>
<tr><td>I. 5</td><td> 
</td><td>1</td><td>1</td></tr>
<tr><td>I. 6</td><td>An Encoder is a combinational circuit that has maximum of 2n input lines and ‘n’ output lines.</td><td>1</td><td>1</td></tr>
<tr><td>I. 7</td><td>0100 0101</td><td>1</td><td>1</td></tr>
<tr><td>I. 8</td><td><table><tr><th>A</th><th>B</th><th>C</th><th>S</th></tr><tr><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>1</td><td>0</td></tr></table></td><td>1</td><td>1</td></tr>
<tr><td>I. 9</td><td>Registers - a group of flip flops used to store multiple bits of data</td><td>1</td><td>1</td></tr>
</table>

(Sajan.D)

				24
	PART B			
II. 1	Binary – base2 – maximum 2 symbols 0,1 Octal - – base8 – maximum 8 symbols 0,7 B F A 6 = 1011 1111 1010 0110 (Binary) = 1 011 111 110 100 110 = 1 3 7 6 4 6 (Octal)	1 1 1	3	
II. 2	**In Weighted binary codes,** each position of the number represents a specific weight. Several systems of the codes(eg 8421) are used to express the decimal digits 0 through 9. In these codes each decimal digit is represented by a group of four bits.  **In Non-Weighted Code,** the positional weights are not assigned. The examples of non-weighted codes are Excess-3 code and Gray code. Gray code of 9 is 1101	1.5 1.5	3	
II. 3	Even parity : Number of 1's in a binary data must be even including parity bit. Consider the data 1010001, which has three ones in it. By adding a one, making the sequence 1 1010001, so that there are four ones (an even number). Consider the data 1010011, which has four ones in it. making the sequence 0 1010001, maintaining even number of 1's	1 1 1	3	
II. 4	The SOP (Sum of Product) and POS (Product of Sum) are the methods for deducing a particular logic function SOP : Adding Product terms (minterm) – minterms are taken where the input set produces a value 1. Eg : Y = A'B'C'+A'BC'+A'BC+ABC'+ABC POS : Multiplying only Sum terms (Maxterm) - Maxterms are taken where the input set generate a value 0 Eg : Y = (A+B+C').(A'+B+C).(A'+B+C')	1 1 1	3	

Sign B

II. 5			3	

A	B	A XOR B
0	0	0
0	1	1
1	0	1
1	1	0

From truth table high outputs are taken for minterms…
A XOR B = A'B + AB'

II. 6			3
	A' + B'		
	= A'.1 + 1.B'	Multiplying all terms by 1 for all missing literals	1
	= A'(B+B') + (A+A').B'	X + X' = 1	1
	= A'B + A'B' + AB' + A'B'		1
	= A'B' + A'B + AB'	Eliminated duplicate terms	

II. 7			3
	A'B'C' + A'BC' + AB'C' + ABC'		
	= A'C'(B'+B) + AC'(B'+B)		1
	= A'C' + AC'		1
	= C' (A'+A)		0.5
	= C'		0.5

II. 8			3

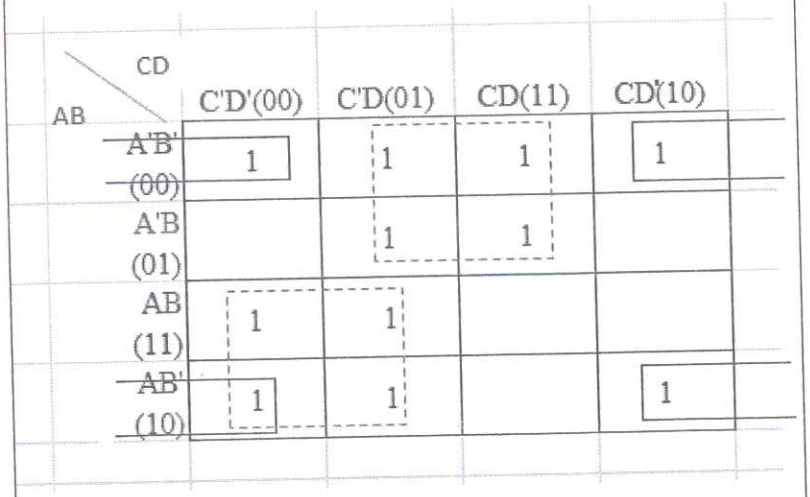
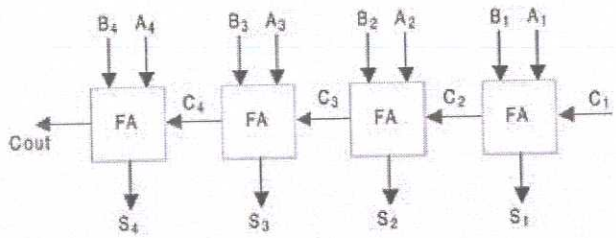
A \ BC	B'C'	B'C	BC	BC'
A'		1		1
A		1	1	1

3 input K Map draw	1	
Writing Labels above columns and left of row	1	
Marking minterms in k-map		

3 RB.. Sym B

		1		
II.9	Step1 : Identify the problem Step2 : write truth table Step3 : Identify high outputs and write output functions for sum and carry in SOP form Step4 : Simplify Standard SOP form to Simplified expressions. Step5 : Draw the logic diagram of ouput functions and Implement the circuit.	0.5 1 0.5 0.5 0.5	3	
II.10	Clocked flip-flops as memory elements in Synchronous where as unclocked flip-flops asynchronous sequential circuits. in Asynchronous : The status of memory element(Flip flop) change any time as soon as input is changed. in Synchronous : The status of memory element(Flip flop) change only at the active clock edge A race condition exists in an asynchronous circuit : two or more binary state variables may change value in response to a change in an input variable.	1 1 1	3	
	PART C			42
III. 1	Gray code(10111011): 11100110 (XORing steps + Answer) Hex(1011 1011.1010) : $(BB.A)_{16}$ ((Grouping by 4 + Answer)) Oct(10 111 011.101) :$(273.5)_8$ ((Grouping by 3 + Answer)) Oct(10 111 011.101) 187.625 (By Power of 2)	2 2 1 2	7	7
III. 2	Convert to Binary 1's Complement 2's Complement Binary Addition Identifying Positive or Negative Convert to Decimal	1 1 1 2 1 1	7	7
III. 3	List Gates Logic Expression Logic Symbol Truthtable	1 2 2 2	7	7

Sajin. B

III. 4	 Answer is :A'D + AC' + B'D'		7	7
	4 Variable K Map Drawing and Marking Labels in Rows and Columns	1		
	Mapping minterms in K Map	2		
	Grouping	2		
	Identifying common terms and write to SOP	2		
III. 5		3	7	7
	4- bit Parallel Adder is designed using 4 Full Adders FA0, FA1, FA2, FA3 .	1		
	Full Adder FA_0 adds A_0, B_0 along with carry Cin to generate Sum S_0 and Carry bit C_1 and this Carry bit is connected to FA1. FA_1 accepts this Carry C_1 and adds with its inputs A_1 and B_1 to generate Sum S_1 and Carry C_2.	3		

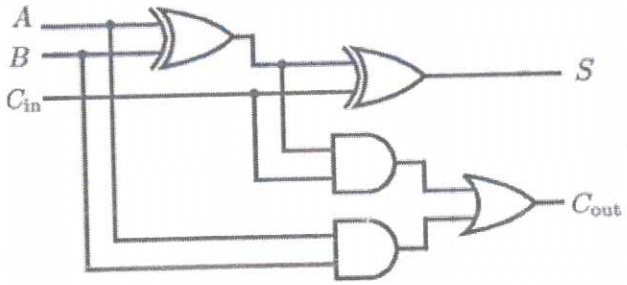
Sajis. B

III. 6

Inputs			Outputs	
A	B	C_{in}	Sum	Carry
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

$S = A \text{ XOR } B \text{ XOR } C_{in}$

$C_{out} = C_{in}.\ (A \text{ XOR } B) + AB$



Truth table — 2

Sum and Carry in SOP — 2

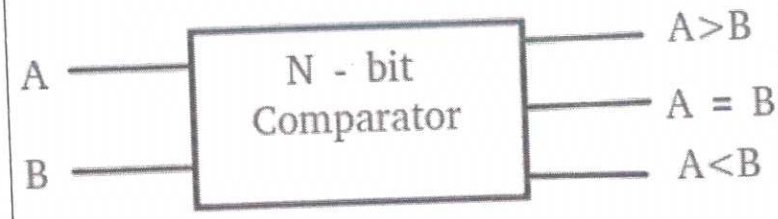
Simplify — 1

Circuit Diagram — 2

7 | 7

6 Sajis. B

III. 7 A magnitude digital Comparator is a combinational circuit that compares two digital or binary numbers in order to find out whether one binary number is equal, less than, or greater than the other binary number.
It have two inputs one for A and the other for B and have three output terminals, one for A > B condition, one for A = B condition, and one for A < B condition. 1 | 7 | 7



A comparator used to compare two binary numbers each of two bits is called a 2-bit Magnitude comparator. It consists of four inputs and three outputs to generate less than, equal to, and greater than between two binary numbers. 2

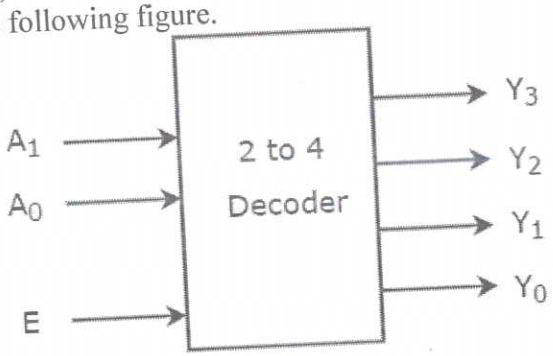
The truth table for a 2-bit comparator

INPUT				OUTPUT		
A1	A0	B1	B0	A>B	A=B	A<B
0	0	0	0	0	1	0
0	0	0	1	0	0	1
0	0	1	0	0	0	1
0	0	1	1	0	0	1
0	1	0	0	1	0	0
0	1	0	1	0	1	0
0	1	1	0	0	0	1
0	1	1	1	0	0	1
1	0	0	0	1	0	0
1	0	0	1	1	0	0
1	0	1	0	0	1	0
1	0	1	1	0	0	1
1	1	0	0	1	0	0
1	1	0	1	1	0	0
1	1	1	0	1	0	0
1	1	1	1	0	1	0

4

Sept. B

III. 8 | 2 to 4 Decoder has two inputs A1 & A0 and four outputs Y3, Y2, Y1 & Y0. The block diagram of 2 to 4 decoder is shown in the following figure. | 2 | 7 | 7



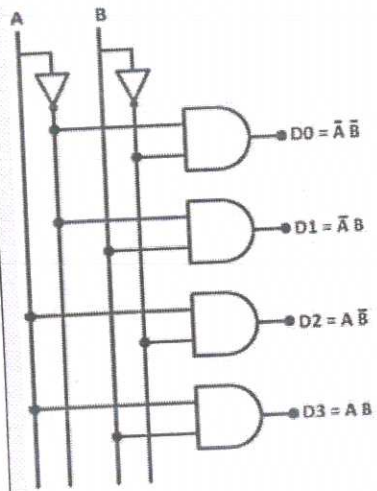
1

One of these four outputs will be '1' for each combination of inputs

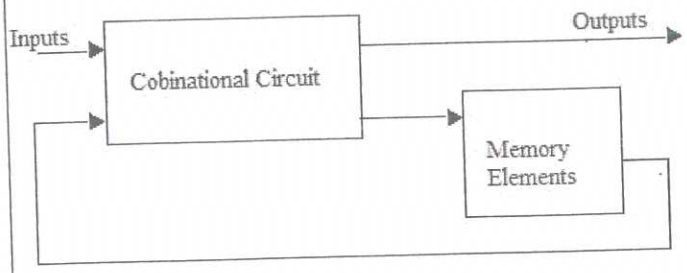
Inputs		Outputs			
A	B	d_0	d_1	d_2	d_3
0	0	1	0	0	0
0	1	0	1	0	0
1	0	0	0	1	0
1	1	0	0	0	1

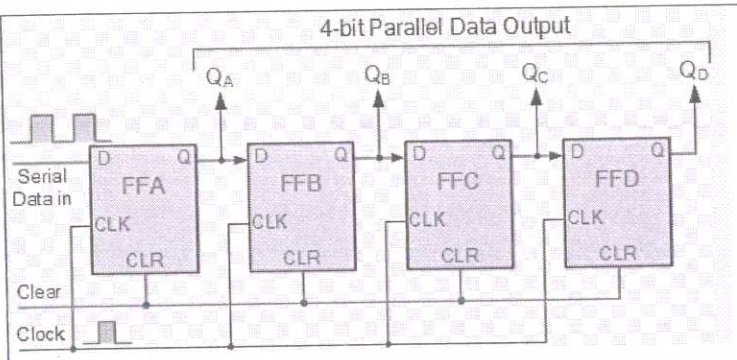
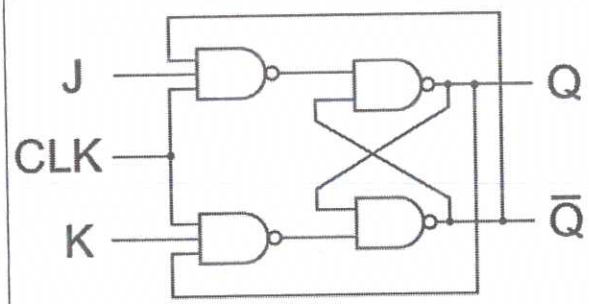
2

The outputs of 2 to 4 decoder are nothing but the min terms of two input variables A1 & A0



2

III. 9	A combinational circuit produces an output based on input variables only, but a **sequential circuit** produces an output based on **current input and previous output variables**. That means sequential circuits include memory elements that are capable of storing binary information. A latch capable of storing one bit of information  As shown in the figure, there are two types of input to the combinational logic : External inputs which are not controlled by the circuit. Internal inputs, which are a function of a previous output state. Asynchronous sequential circuit: do not use a clock signal – faster - change their state immediately when there is a change in the input signal. Synchronous sequential circuit: uses clock signal - slower compared to asynchronous. output changes state at the start of an input pulse and remains in that until the next input or clock pulse.	1 1 2 1 1 1	7	7
III. 10	Shift registers hold the data in their memory which is moved or "shifted" to their required positions on each clock pulse The shift register, which allows serial input (one bit after the other through a single data line) and produces a parallel output is known as Serial-In Parallel-Out shift register. The logic circuit given below shows a serial-in-parallel-out shift register.	1 1 3	7	7

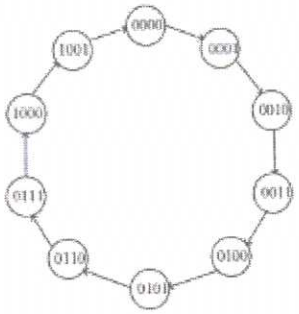
	 The circuit consists of four D flip-flops which are connected. The clear (CLR) signal is connected in addition to the clock signal to all the 4 flip flops in order to RESET them. The output of the first flip flop is connected to the input of the next flip flop and so on. All these flip-flops are synchronous with each other since the same clock signal is applied to each flip flop.	2		
III. 11	A gated S R flip flop with the addition of a clock input circuitry is basically the J K flip flop. This circuit prevents the invalid output condition which occurs when both inputs are high. Most important characteristics is it toggles the previous output when J=K=1.  In the above circuit, we can see that the output is fed back to the enabling NAND gates. This makes the toggling action possible at J=K=1. Truth Table for J K Flip Flop	1 1 2 1 2	7	7

Clock	J	K	Q_{n+1}	State
0	x	x	Q_n	
1	0	0	Q_n	Hold
1	0	1	0	Reset
1	1	1	1	Set
1	1	1	$\overline{Q}_n$	Toggle

JK flip-flops are used as one-bit storage elements, clock dividers and constructing counters and shift registers.

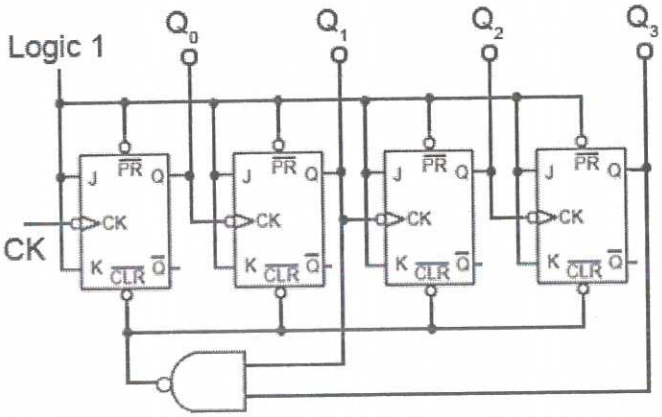
III. 12 BCD counter, also known as decade counter, is one of the types of most widely used digital counters, which counts up to 10 with an applied clock signal. 1

It is a 4-bit binary digital counter, counts from 0000 (0) to 1001 (9) in decimal form on the application of the clock signal. 1



When the count reaches the predetermined count value(10), it resets all the flip-flops and starts to count again from 0. 1

BCD or decade counter circuit is designed by using JK flip flops and NAND gate. The BCD counter it requires 4 JK flip flops because it is a 4-bit binary counter. The design of the decade counter is shown below. 1



2

7 7

Sagi D

The truth table of the decade counter is shown below.

Input pulses/clock pulses	Q3	Q2	Q1	Q0
0	0	0	0	0(resets)
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	0	0	0	0

1

It counts the decimal input pulses and displays the output in binary form. The output of the NAND gate is zero when the input pulse count reaches 9 (1001).

1