Vidyavardhini's College of Engineering and Technology

Department of Artificial Intelligence & Data Science

AY: 2024-25

Class:	SE	Semester:	III
Course Code:	CSC304	Course Name:	DIGITAL LOGIC AND COMPUTER ARCHITECTURE

Name of Student:	SHRUTI GAUCHANDRA
Roll No. :	15
Assignment No.:	03
Title of Assignment:	Analyze the truth table of digital components and identify them, their functions in processor architecture.
Date of Submission:	06/09/24
Date of Correction:	06/09/24

Evaluation

Performance Indicator	Max. Marks	Marks Obtained
Completeness	5	5
Demonstrated Knowledge	3	3
Legibility	2	2
Total	10	10

Performance Indicator	Exceed Expectations (EE)	Meet Expectations (ME)	Below Expectations (BE)
Completeness	5	3-4	1-2
Demonstrated Knowledge Legibility	3	2	1
Legibility	2	1	0

Checked by

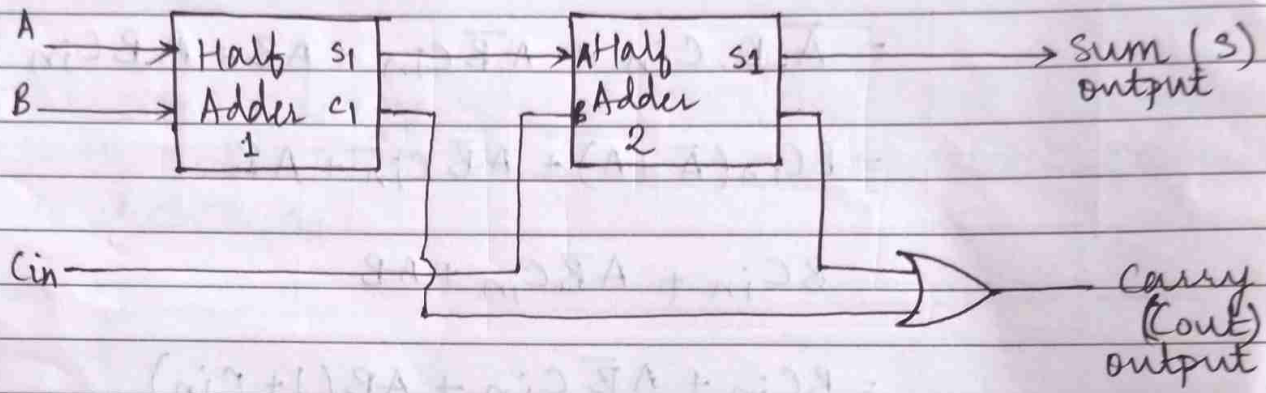
Name of Faculty : MS. KSHITIJA GHARAT

Signature : Bharat

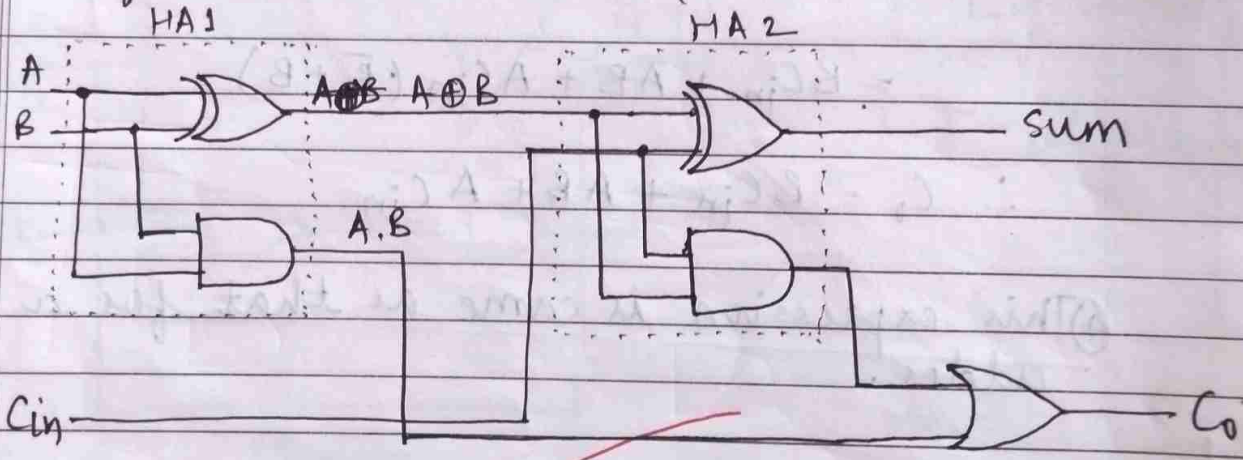
Date : 6/9/24

Q1. Apply the concept of half adder and additional gates to design full adder.

→ (1) The full adder circuit can be constructed using two half adders as shown:



(2) A full adder can be implemented using two half adders an an OR Gate as shown.



Here, $S = (A \oplus B) \oplus C_{in} = A \oplus B \oplus C_{in}$

(3) This expression is same as that obtained for the full adder. Thus the sum output has been successfully implemented by the circuit

$C_0 = (A \oplus B)\, C_{in} + AB$

$C_0 = (\overline{AB} + \overline{AB})\, C_{in} + AB = \bar{A}BC_{in} + A\bar{B}C_{in} + AB$

$= \bar{A}BC_{in} + A\bar{B}C_{in} + AB(1 + C_{in})$

$= \bar{A}.B.C_{in} + A\bar{B}C_{in} + AB + ABC_{in}$

$= BC_{in}(\bar{A} + A) + A\bar{B}C_{in} + AB$

$= BC_{in} + A\bar{B}C_{in} + AB$

$= BC_{in} + A\bar{B}C_{in} + AB(1 + C_{in})$

$= BC_{in} + A\bar{B}C_{in} + AB + ABC_{in}$

$= BC_{in} + AB + AC_{in}(\bar{B} + B)$

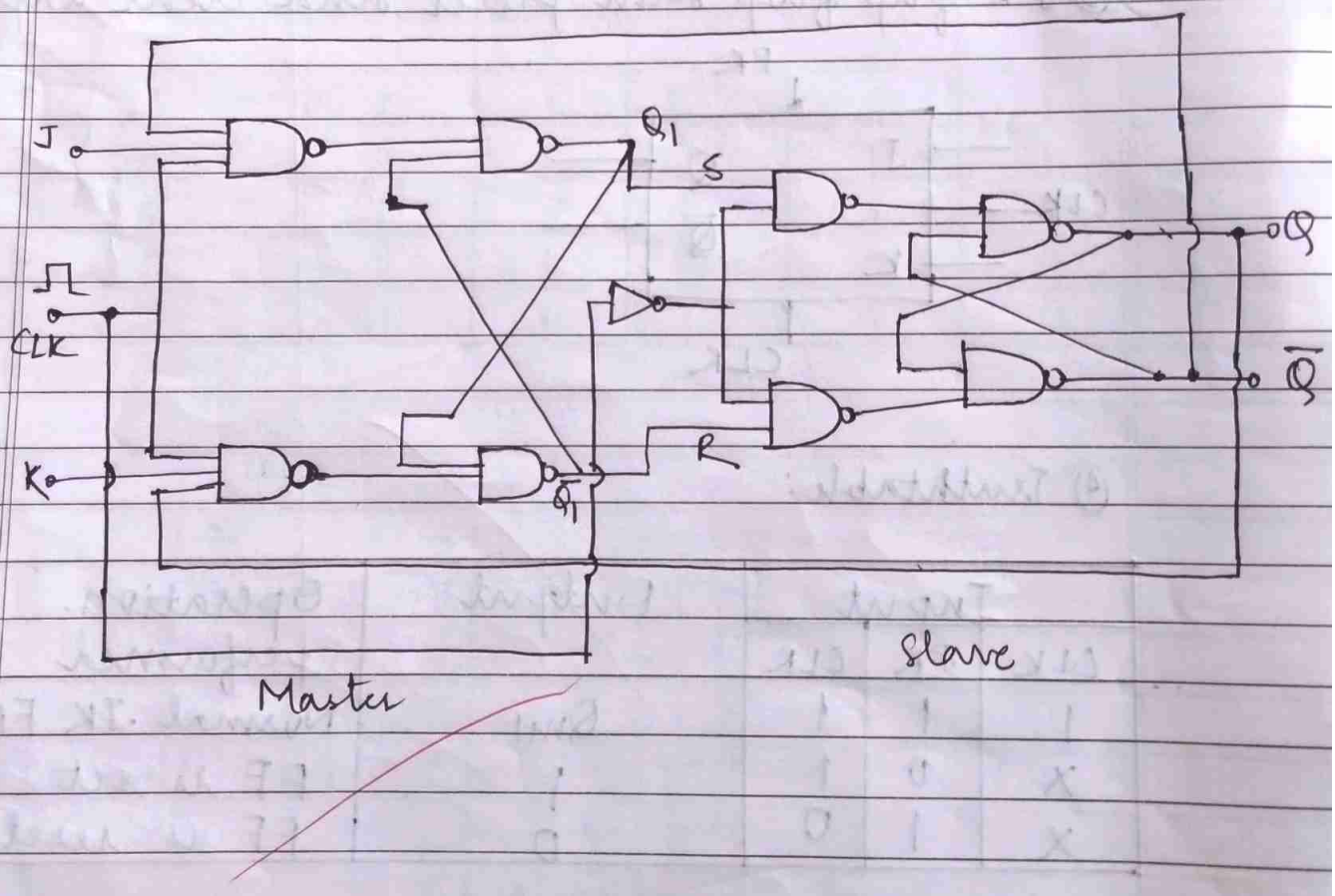
$\therefore C_0 = BC_{in} + AB + AC_{in}$.

④ This expression is same as that for a full adder.

Q2. Draw logic circuit diagram Master - Slave J-K flip-flop with PRESET and CLEAR inputs using NAND gates. Give its truth table and logic symbol.

→ ① Circuit Diagram:

The circuit diagram of Master - Slave JK flip-flop with PRESET & CLEAR inputs using NAND gates is shown as:

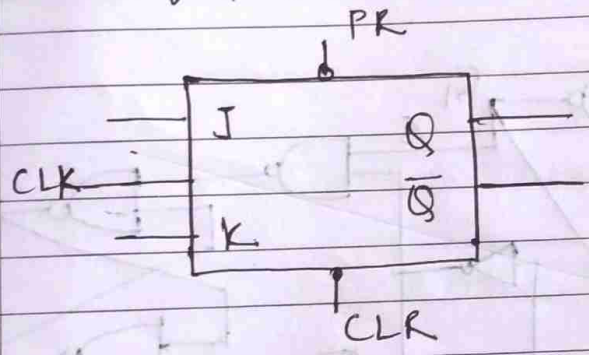


P.T.O.

② Truth Table:

Case	Inputs			Outputs		Remark
	CLK	J	K	Q_{n+1}	$\bar{Q}_{n+1}$	
I	X	0	0	Q_n	$\bar{Q}_n$	No change
II	⎍(↓)	0	0	Q_n	$\bar{Q}_n$	No change
III	⎍(↓)	0	1	0	1	Reset
IV	⎍(↓)	1	0	1	0	set
V	⎍(↓)	1	1	Q_n	$\bar{Q}_n$	Toggle

③ JK flip flop with preset and clear inputs.



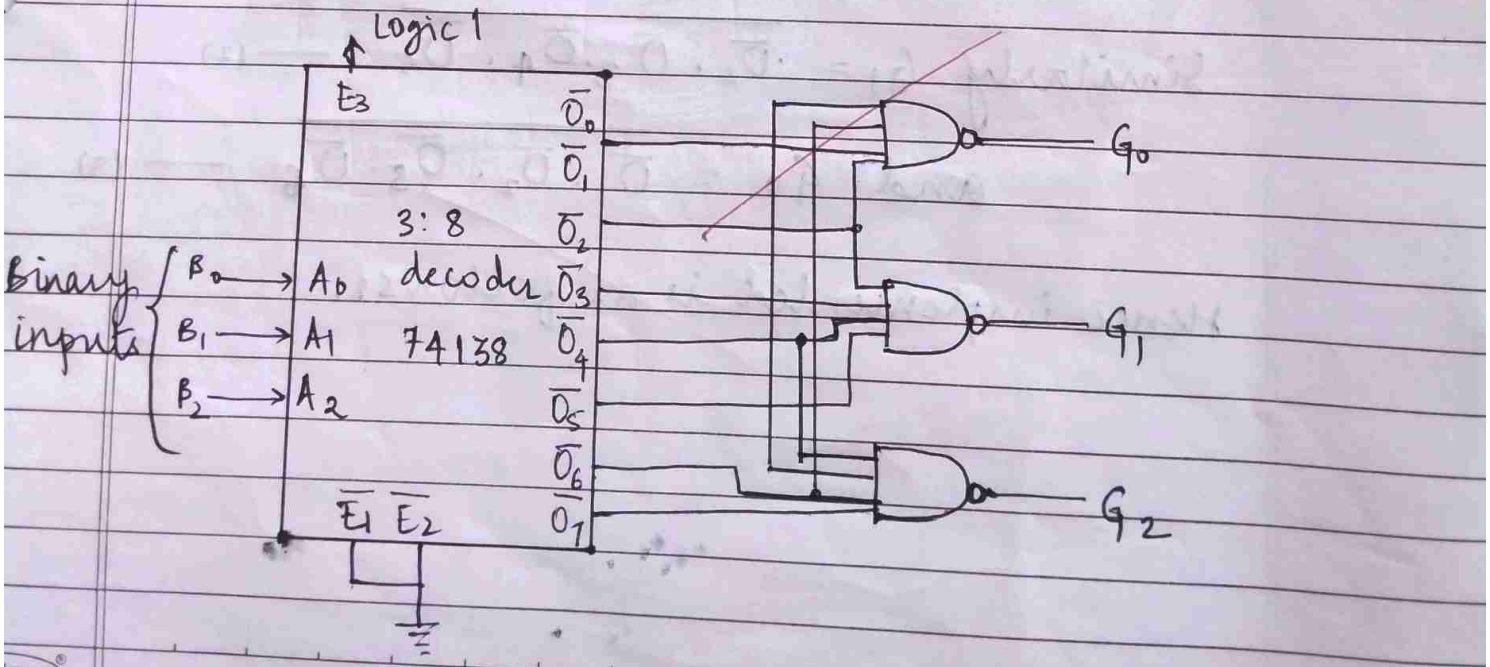
④ Truthtable:

Input			Output	Operation performed
CLK	PR	CLR		
↓	1	1	Q_{n+1}	Normal JK FF
X	0	1	1	FF is set
X	1	0	0	FF is reset

Q3. Implement a 3-bit binary to gray code converter using decoder IC 74138

→ Step 1: Truth Table relating the binary and gray codes:

Decimal	Binary inputs			Gray outputs		
	B_2	B_1	B_0	G_2	G_1	G_0
0	0	0	0	0	0	0
1	0	0	1	0	0	1
2	0	1	0	0	1	1
3	0	1	1	0	1	0
4	1	0	0	1	1	0
5	1	0	1	1	1	1
6	1	1	0	1	0	1
7	1	1	1	1	0	0



Step 2: Expressions for G_2, G_1 and G_0:

Normally the expressions for G_2, G_1 and G_0 would have been written as:

$$G_2 = O_4 + O_5 + O_6 + O_7$$
$$G_1 = O_2 + O_3 + O_4 + O_5$$
$$G_0 = O_1 + O_2 + O_5 + O_6$$

But the outputs of IC 74138 are active low. Hence we will convert these equations:

$$G_2 = O_4 + O_5 + O_6 + O_7$$
$$= \overline{\overline{O_4 + O_5 + O_6 + O_7}}$$

But $\overline{A + B} = \overline{A} \cdot \overline{B}$

$$\therefore G_2 = \overline{\overline{O_4} \cdot \overline{O_5} \cdot \overline{O_6} \cdot \overline{O_7}} \quad \text{—(1)}$$

Similarly $G_1 = \overline{\overline{O_2} \cdot \overline{O_3} \cdot \overline{O_4} \cdot \overline{O_5}}$ —(2)

and $G_0 = \overline{\overline{O_1} \cdot \overline{O_2} \cdot \overline{O_5} \cdot \overline{O_6}}$ —(3)

Hence implemented is as follows: