

Month and Year : September 2019	Roll Number :
Programme : B.E/B.Tech Branch : CSE/ IT Semester : III	Date : 19.09.19 Time : 9.15 AM to 10.45 AM
Course Code : 18CST32 Course Name : Computer Organization	Duration : 1 ½ Hours Max. Marks : 50

PART - A ($10 \times 2 = 20$ Marks)

ANSWER ALL THE QUESTIONS

- Assume the following numbers are in 2's complement representation [CO2,K3]
i)11100111 ii)11100100
Among these two numbers which number is divisible and not divisible by 11111011?
- Convert the decimal number – 14.25 into IEEE-754 32 bit format. [CO2,K3]
- Find the decimal value of the following bit pattern that represents a floating point number in IEEE-754 single precision format [CO2,K3]
11000001110100000000000000000000
- What is the minimum number of registers needed to evaluate the expression [CO3,K3]
 $(a - b) + (e - (c + d))$
- A processor has 40 distinct instructions and 24 general purpose registers. A 32-bit instruction word has an opcode, two register operands and an immediate operand. Find the number of bits available for the immediate operand field. [CO3,K3]
- The main memory that has memory addresses of 400,500,600,700 respectively (All values in decimal). A CPU has 24 bit instructions and a program starts at address 300 (in decimal) and the program follows sequential execution. Among these addresses, identify the legal values for the program counter. [CO3,K3]
- Differentiate between RISC and CISC processor. [CO3,K2]
- Write the 5 stages of a RISC processor. [CO3,K1]
- Convert the decimal fraction 0.1 to a binary fraction. If the conversion is not exact, give the binary fraction approximation to 8 bits after the binary point using the three truncation (chopping, Von Neumann and rounding) techniques. [CO2,K3]
- What is microprogrammed control? And where it is used? [CO3,K1]

PART – B ($3 \times 10 = 30$ Marks)
ANSWER ANY THREE QUESTIONS

- Apply restoring and non restoring division algorithms for the dividend 21 and the divisor 5 (Both values are in decimal). Also write the algorithm. (10) [CO2,K3]
- i) Illustrate with suitable examples about single precision and double precision floating point formats.
ii) Perform subtract operation for the 12 bit floating point numbers $A=1.011011 \times 2^2$ and $B = -1.10101 \times 2^0$ based on subtract rule. Also apply three truncation techniques to 3 bits after the binary point.
- With suitable diagram, summarize about hardware components of a processor. (10) [CO3,K2]
- Draw the datapath diagram and write the sequence of actions of 5 stages to fetch and execute the instruction ADD R3, R4, R5 with control signals. (10) [CO3,K3]

Bloom's Taxonomy Level	Remembering (K1)	Understanding (K2)	Applying (K3)	Analysing (K4)	Evaluating (K5)	Creating (K6)
Percentage	7	20	73	-	-	-

ANSWER KEY

PART - A

i) divisor = $\begin{array}{r} 1111 \\ \times 1011 \\ \hline 11110111 \end{array}$ $\Rightarrow (-5)$ (2 marks)
 $= \begin{array}{r} 0000 \\ \times 0100 \\ \hline 00000100 \end{array}$
 $\begin{array}{r} 00000100 \\ \times 101 \\ \hline 00000101 \end{array}$ (5)

ii) $\begin{array}{r} 11100111 \\ \times 1011 \\ \hline 11111011 \end{array}$ $\Rightarrow (-25)$
 $\begin{array}{r} 11111011 \\ \times 101 \\ \hline 11111001 \end{array}$
 $\begin{array}{r} 11111001 \\ \times 101 \\ \hline 00011001 \end{array}$ (168) $\Rightarrow (25)$

So $\frac{-25}{-5} = 5$ it is divisible by (-5)
(1 mark)

iii) $\begin{array}{r} 11100100 \\ \times 1011 \\ \hline 11111011 \end{array}$ $\Rightarrow (-28)$
 $\begin{array}{r} 11100100 \\ \times 1011 \\ \hline 00011100 \end{array}$
 $\begin{array}{r} 11100100 \\ \times 1011 \\ \hline 00011100 \end{array}$ (1684) $\Rightarrow (28)$

So $\frac{-28}{-5}$ is not divisible by (-5)
(1 mark)

2). -14.25 (2 marks)
 STEP 1: Take binary value for $(14)_{10}$

$$\begin{array}{r} \checkmark \checkmark \checkmark \times \\ 8 \ 4 \ 2 \ 1 \\ \hline \end{array}$$

$$(14)_{10} \Rightarrow (1110)_2$$

STEP 2: Take binary value for $(0.25)_6$

$$0.25 \times 2 = \begin{array}{r} 0.50 \\ \times 2 \\ \hline 1.00 \end{array}$$

$$(0.25)_{10} = (0.01)_2$$

STEP 3: Write in Normalized form (32-bit)

$$(14.25)_{10} = (1110.01)_2$$

$$= 1.11001 \times 2^3$$

$$\text{So } E = 3 \quad \rightarrow M$$

(1 mark)

STEP 4: Find E' (Exponent value in Excess - 127 format)

$$E' = E + 127$$

$$3 + 127 = 130 \quad \begin{matrix} 128 & 64 & 32 & 16 & 8 & 4 & 2 & 1 \end{matrix}$$

$$E' = 130 \Rightarrow 10000010$$

STEP 5: Represent $(-14.25)_{10}$ in 32-bit format

$E_{15\text{bit}}$	$E'_{8\text{bit}}$	$M_{23\text{bit}}$
1	10000010	110010000-----0000

18 zeros
(1 mark)

3). S E1 M (2 marks)

1	1000 0011	1010 0000 ... 0
---	-----------	-----------------

STEP 1: Find Exponent (E)

$$E^1 = E + 127$$

$$E^1 = 1000 0011$$

$$131 = E + 127 \quad 128 \quad 21 \Rightarrow 31$$

$$\boxed{E = 4}$$

STEP 2: Representation of Normalized form
mantissa Normalized from

$$1.1010 00 ... 0 \times 2^4 \quad (1 \text{ mark})$$

STEP 3: Find the decimal value by adjusting the exponent value upto 0.

$$11010.00 ... 0 \times 2^0$$

$$80 \quad (11010.0)_2$$

STEP 4: Convert Binary upto decimal

$$(11010)_2 \\ \text{to } 8 \quad 2 \Rightarrow (26)$$

$$80 - (26.0)_0 \quad (1 \text{ mark})$$

6

$$4) (a-b) + (e - (c+d))$$

(2 marks)

$$R_1 \leftarrow c;$$

$$R_2 \leftarrow d;$$

$$R_2 \leftarrow R_1 + R_2; (R_2 \leftarrow (c+d))$$

$$R_1 \leftarrow e;$$

$$R_2 \leftarrow R_1 - R_2; (R_2 \leftarrow (e - (c+d)))$$

$$R_1 \leftarrow a;$$

$$R_3 \leftarrow b;$$

$$R_1 \leftarrow R_1 - R_3; (R_1 \leftarrow (a-b))$$

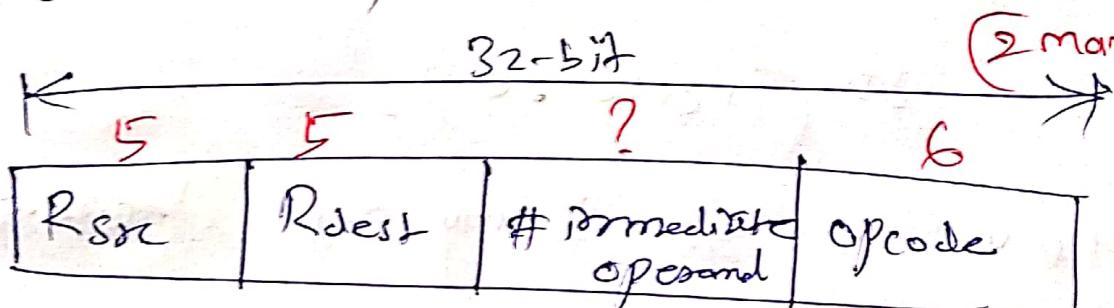
$$R_1 \leftarrow R_1 + R_2; (R_1 \leftarrow (a-b) + (e - (c+d)))$$

So totally 3 Registers used.

5

32-bit

(2 marks)



40 distinct Instructions
(opcode) $\{ = 2^6 \} (64 > 40)$

24 GPR $\Rightarrow 2^5 \} (32 > 24) \text{ (marks)}$

$$\text{So Immediate operand value} = 32\text{-bit} - 5 + 5 + 6 \\ = 32 - 16 \\ = 16 \rightarrow 5\text{-bit} \text{ (marks)}$$

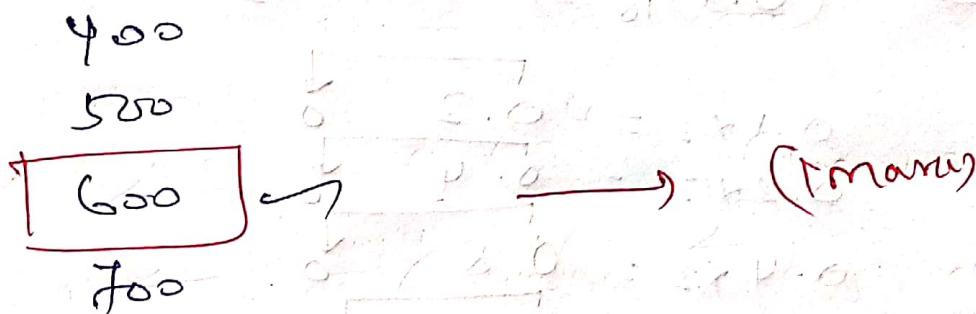
6). A CPU has 24-bit Instruction

(2 marks)

- * So total no. of bytes is 3 Bytes - (marks)
Required for a Single Instruction

* Program Starts at address 300

- * Among the following addresses,



only 600 is divisible by 3. So

600 is an legal value for the PC.

7)

RISC

CISC (marks)

(Any 2 point)

- * Simple Addressing Modes
- * All instructions fitting in a single word
- * ALU operations are performed only on process registers
- * Load / Store Instructions are used
- * More Instructions in need.
- * More Complex addressing modes
- * An Instruction may span multiple words
- * Here, it is performed on memory operand as well as process Register operands.

Q8). 5-Stages of a RISC processor.

(2marks)

1. Instruction fetch

2. Source Registers

3. ALU

4. Memory access

5. Destination Registers

$$Q9). (0.1)_{10} \Rightarrow (?)_2 \quad (2marks)$$

$$0.1 \times 2 = 0.2$$

$$0.2 \times 2 = 0.4$$

$$0.4 \times 2 = 0.8$$

$$0.8 \times 2 = 1.6$$

$$0.6 \times 2 = 1.2$$

$$0.2 \times 2 = 0.4$$

$$0.4 \times 2 = 0.8$$

$$0.8 \times 2 = 1.6$$

$$0.6 \times 2 = 1.2$$

$$0.2 \times 2 = 0.4$$

$$0.4 \times 2 = 0.8$$

$$0.8 \times 2 = 1.6$$

$$0.6 \times 2 = 1.2$$

Repeating
Pattern

So the binary value for

$$(0.1)_{10} = (0.0001100110011\dots)_2 \quad (1mark)$$

Truncation into 8-bits after the fraction point
by using Three Truncation Methods is

0.0001100110011.....
8bit

1) Chopping

(Forwrd)

0.00011001

2) Von-Neumann Rounding

0.00011001

3) Rounding

0.00011001
0.00000001(+)
0.00011010

⑥

Microprogrammed Control :

(AnsweR)

* Control signals are generated by a program stored in a special memory. This technique is called microprogrammed control.

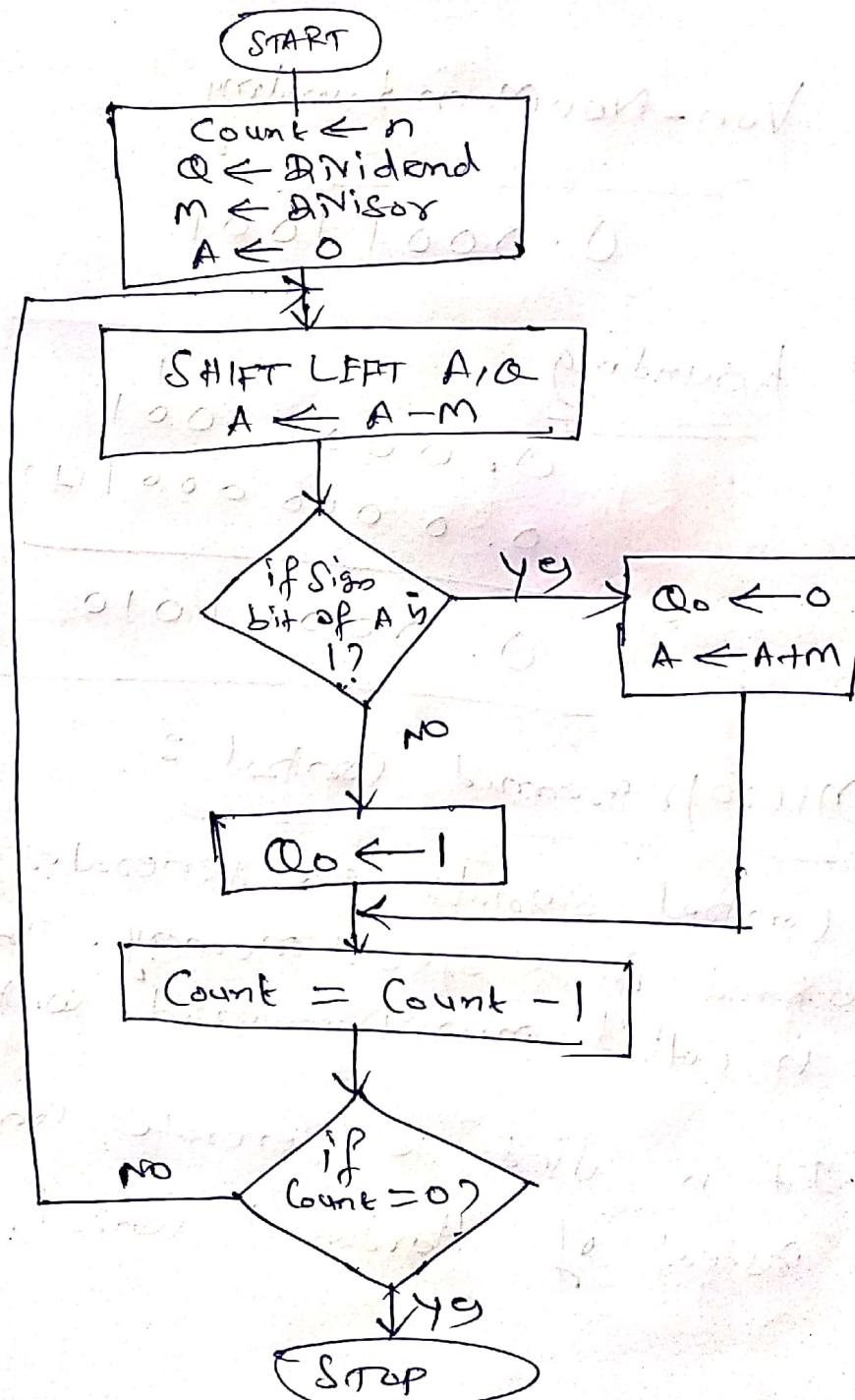
* It is used to generate control signals instead of Hardwired Control.

PART-B

11)

$$\begin{array}{r} 4 \rightarrow \text{Quotient} \\ 5 \overline{)21} \\ 20 \\ \hline 1 \rightarrow \text{Remainder} \end{array}$$

Algorithm for Restoring Division:



168421

$$\begin{array}{r} Q = 21 \\ M = 5 \end{array}$$

$$Q = 10101 \text{ (21)}$$

$$M = 0010100 (5)$$

(To mark)

$$M = 000101$$

$$111010$$

$$2^5 \Rightarrow \underline{111011}$$

$$M \quad \boxed{000101}$$

$$A \quad \boxed{000000}$$

$$\boxed{10101}$$

Q Count=5

$$Q) \quad \boxed{000000} \quad 0101 \quad \square$$

$$111011$$

SL A, Q;

A \leftarrow A-M;

Q₀ \leftarrow 0

A \leftarrow A+M

Restoring
Previous
"A" value

$$\boxed{1} \quad 11100$$

$$000101$$

$$\boxed{1} \quad 000001$$

$$01010$$

$$Q) \quad \boxed{0000010} \quad 1010 \quad \square$$

$$111011$$

$$\boxed{1} \quad 11101$$

$$10100$$

$$Q) \quad \boxed{0000010} \quad 10100$$

$$111011$$

$$\boxed{1} \quad 000000$$

$$0100 \quad \square$$

$$Q) \quad \boxed{0000000} \quad 1001 \quad \square$$

$$111011$$

$$\boxed{1} \quad 111011$$

$$1001 \quad \square$$

$$Q) \quad \boxed{0000000} \quad 10010$$

$$count=1$$

Count=4

SL A, Q;

A \leftarrow A-M;

Q₀ \leftarrow 0

A \leftarrow A+M

Count=3

SL A, Q;

A \leftarrow A-M

Q₀ \leftarrow 1

Count=2

SL A, Q;

A \leftarrow A-M

Q₀ \leftarrow 0

A \leftarrow A+M

$$\begin{array}{r}
 000000 \quad | \quad 10010 \\
 0) 000001 \quad | \quad 0010 \boxed{0} \\
 111011 \\
 \hline
 11100000010 \boxed{0}
 \end{array}$$

Count = 1
 SCA/A
 $A \leftarrow A - m$
 $Q_0 \leftarrow 0$
 $A \leftarrow A + m$

$$\begin{array}{r}
 000101 \quad | \quad 10010 \\
 0) 000001 \quad | \quad 00100
 \end{array}$$

Count = 0

Quotient Remainder

$$\text{Quotient} = 00100 \quad (4)$$

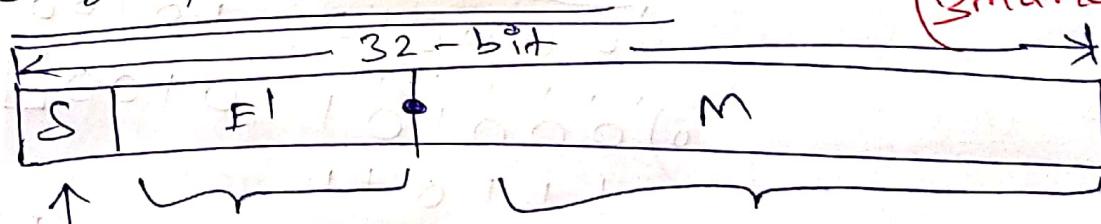
$$\text{Remainder} = 00001 \quad (1)$$

Quotient = 00100

12) (i) Single precision format:

(5 marks)

(3 marks)



Sign of

Number:

$0 \leftarrow (+)$

$1 \leftarrow (-)$

8-bit Signed

Exponent in

format

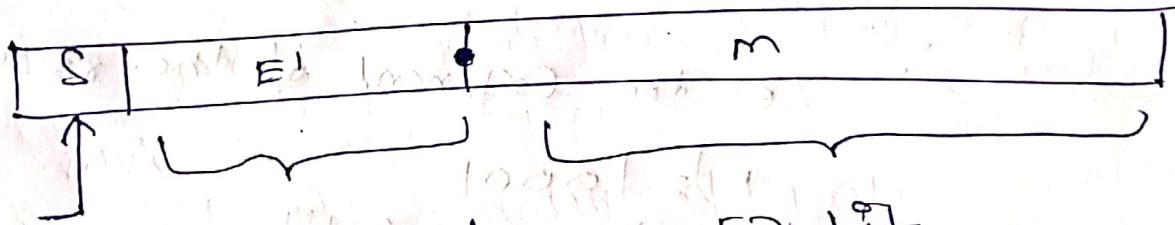
23-bit Mantissa fraction.

Value represented = $\pm 1.m \times 2^e$

Example :

0	00101001	110011.....0
---	----------	--------------

Double Precision Format: (2 marks)



Sign of no.: 11-bit signed
 $+ \Rightarrow 0$ Exponent E
 $- \Rightarrow 1$ Format

52-bit mantissa fraction

Value Represented = $\pm 1.m \times 2^e$

Example :

1	1001101111	110111.....0
---	------------	--------------

12) ii) (5 marks)

$$A = 1.011011 \times 2^2$$

$$B = -1.101010 \times 2^0$$

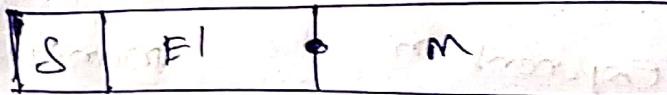
STEP 1: SHIFT the mantissa of B. to the right
by 2 bit positions

$$B = 0.01101010 \times 2^2$$

STEP 2: Set the exponent of the result to

$$E_1 = 10001$$

1-bit 5-bit 6-bit



↓ 12-bit

$$E_1 = E + 15$$

$$E_1 = 2 + 15 = 17 \Rightarrow 10001$$

STEP 3: Subtract the mantissa of B from the
mantissa of A by adding mantissa,
(Because B is negative)

$$A \Rightarrow 1.01101100$$

$$B \Rightarrow 0.01101010 (+)$$

$$A - B \Rightarrow A + B \Rightarrow \underline{1.11010110} \quad (\text{Borrowed})$$

Set the sign of the result in O

STEP 4: Normalize the result

We need to be truncate the mantissa
into six bits by using Rounding

0. 11010110
6 bits

$$\begin{array}{r} 0.110101 \\ - 0.000001 \\ \hline 0.110110 \end{array}$$

STEP 5: Represent in 12-bit format

(2 marks)

0	1000	110110
---	------	--------

If we apply truncation technique into 3 bits
after the binary point means

- 1) Chopping $\Rightarrow 0.110$
- 2) Von-Neumann $\Rightarrow 0.111$
- 3) Rounding $\Rightarrow 0.111$

13)

Main Hardware Components Diagram - 5 Marks

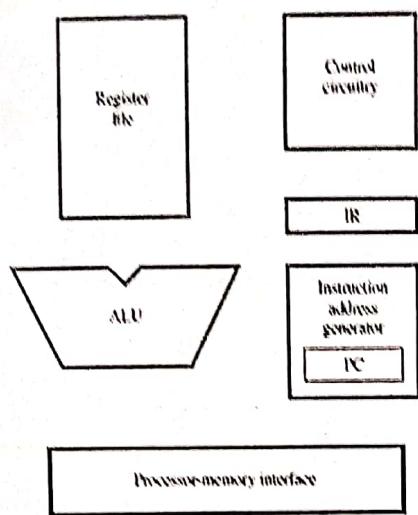


Figure 5.1 Main hardware components of a processor.

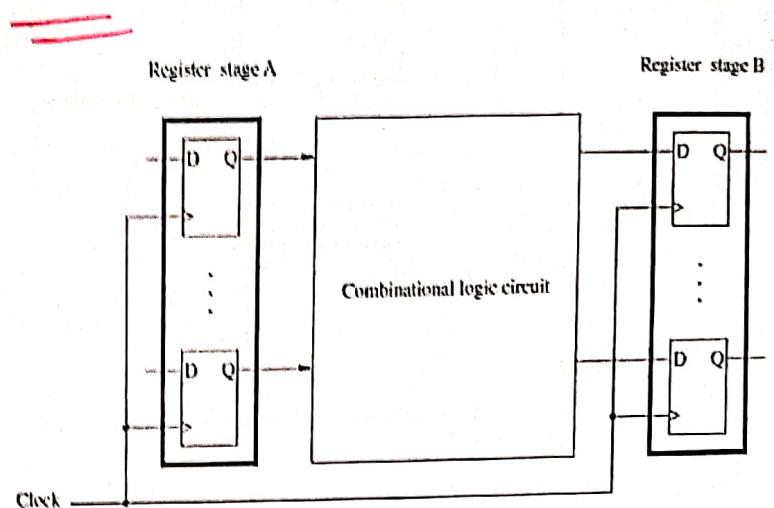
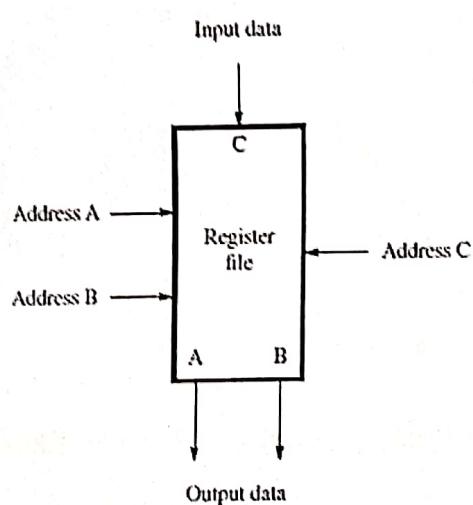
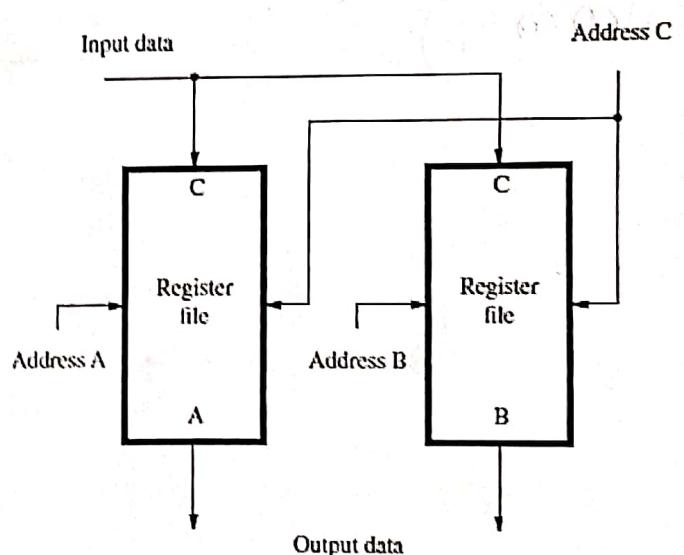


Figure 5.2 Basic structure for data processing.

(OR)



(a) Single memory block



(b) Two memory blocks

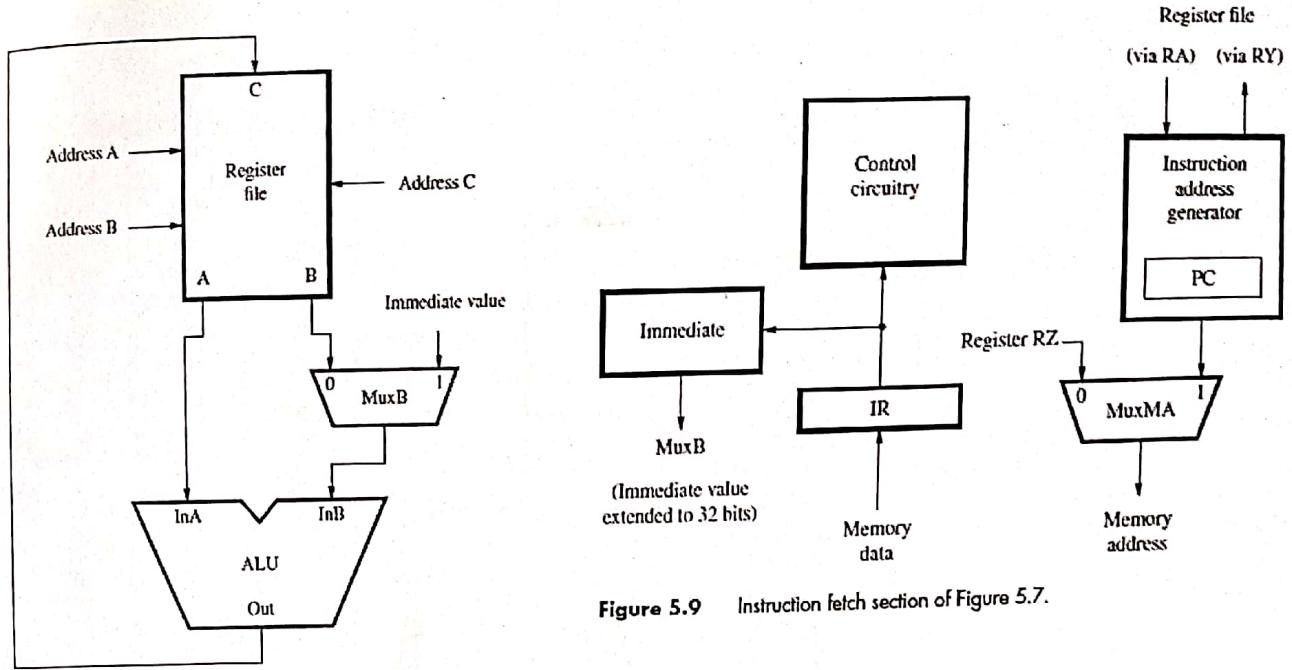


Figure 5.9 Instruction fetch section of Figure 5.7.

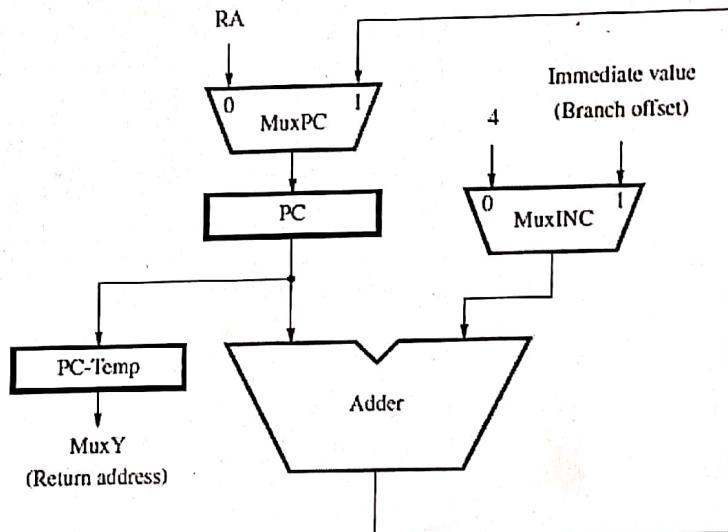
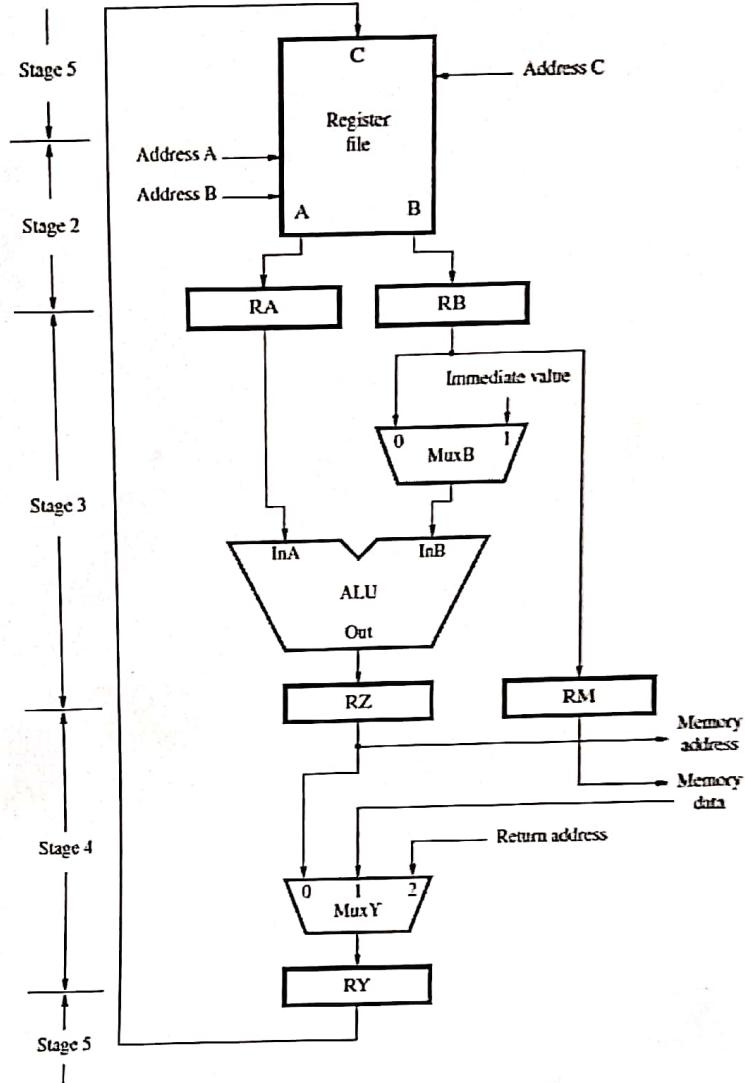


Figure 5.10 Instruction address generator.

Summarization about the following Components: – 5 Marks (Each carry 1 Marks)

1. Register File
2. ALU
3. IR (Instruction Fetch Section)
4. PC & Instruction Address Generator
5. Control Circuitry

14) Datapath Diagram: 5 Marks



Sequence of Steps : 5 Marks Each carry 1 Marks

Step	Action
1	Memory address \leftarrow [PC], Read memory, IR \leftarrow Memory data, PC \leftarrow [PC] + 4
2	Decode instruction, RA \leftarrow [R4], RB \leftarrow [R5]
3	RZ \leftarrow [RA] + [RB]
4	RY \leftarrow [RZ]
5	R3 \leftarrow [RY]

Figure 5.11 Sequence of actions needed to fetch and execute the instruction: Add R3, R4, R5.

DEPT. OF COMPUTER SCIENCE & ENGG.
KONGU ENGINEERING COLLEGE,
THOOPPALAYAM (PO)
PERUNDURAI (TA), ERODE - 638 060



KONGU ENGINEERING COLLEGE

PERUNDURAI ERODE - 638 060.
(Autonomous)



18CSPE151
Name and signature of Hall Supdt. with Date

Name of the Student	S. Raghunathan	Register No.	18CSPE151
Programme	BE	Branch & Semester	Computer Science Engineering Sem - III
Course Code and Name	18C6T02 Computer Organization	Date	19/08/19 No. of Pages Used 10

MARKS TO BE FILLED IN BY THE EXAMINER

PART - A		PART - B		Grand Total Max. Marks : 50
Question No.	Max Marks : 2	Question No.	Max Marks : 10	
1	✓	11	i) ✓	
2	✓	ii) 3		
3	1	12	i) ✓	
4	✓	ii) ✓		
5	✓	13	i) ✓	
6	✓	ii) 2		
7		14	i) ✓	
8	0	ii) ✓		
9	0			
10	0			
TOTAL	10	TOTAL	15	25

Total marks in words :

Twenty five only

Verified by,
D. Mallya

INSTRUCTION TO THE CANDIDATE

- Check the Question Paper, Programme, Course Code, Branch Name etc., before answering the questions.
- Use both sides of the paper for answering questions.
- POSSESSION OF ANY INCRIMINATING MATERIAL AND MALPRACTICE OF ANY NATURE IS PUNISHABLE AS PER RULES.

D. Mallya
Name of the Examiner

Aug 2018
Signature of the Examiner
with Date

1

001001

start - b

$$14. \quad A = 010111, B = 110110$$

Booth algorithm :-

Multiplica \Rightarrow 110110

Multiplicand \Rightarrow 010111

Booth code for multiplicand,

~~000110~~
~~1-1100-1~~

Booth code = 1 - 1100 - 1

Booth multiplication,

110110 * 1-1100 -
00000 0001010
00000 0000000 *
00000 00000 * *
11110110 * * *
0001010 * * * *
110110 * * * *

01100 0011010

$$A * B = 11000011010$$

0111110010
01111100110

1100011010

0	0	1	1	1	0	0	1	1	0
1	1	0	0	0	1	1	0	0	1
<hr/>									
1	1	0	0	0	1	1	0	1	1

Bit pair Recoding :-

Multiplexer \Rightarrow 110110

Multiplicand \Rightarrow 010111

Bit pair code,

010110

Bit pair code = +1 +2 -1

Bit-pair multiplication,

$$\begin{array}{r} 110110 \\ \times +1 +2 -1 \\ \hline 0000001010 \end{array}$$

11101100**

110110***

1100011010

$$A * B = 1100011010$$

11. i) Basic functional units of a computer:-

* Input unit

* Output unit

* Memory

* Arithmetic Logical Unit

* Control unit

Input unit:-

Input unit gets or scans the input from the user via keyboard, mouse, scanner etc.

Output unit:-

Output unit is used to deliver the output to the user through monitor, printer, projector etc.

Arithmetic & logical unit :-

ALU is an part of processor. It performs arithmetic & logical operations and returns the value to memory.

Control unit:-

Control unit is a part of processor. It gives the command for all the other units and is interconnected.

Memory:-

It is classified into primary memory and secondary memory. Primary memory is RAM (Random Access memory) and is volatile. Secondary memory is ROM (Read only Memory) and is non-volatile.

ii. ii) a) 7 and 9

$$7 \Rightarrow 00111$$

$$9 \Rightarrow 01001$$

b) -13 and 14

$$-13 = 101100$$

$$14 = 01110$$

Addition :-

$$7 \Rightarrow 00111$$

$$9 \Rightarrow 01001$$

$$16 \Rightarrow 10000$$

Ans = 010000 (Overflow)

$$-13 \Rightarrow 11101$$

$$14 \Rightarrow 01110$$

$$101001$$

Subtraction :-

$$7 \Rightarrow 00111$$

$$9 \Rightarrow 01001$$

$$-2 \Rightarrow 11110$$

Ans = 1110 (no overflow)

$$011 + 100 = 001$$

$$011 + 100 = 001$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

$$011$$

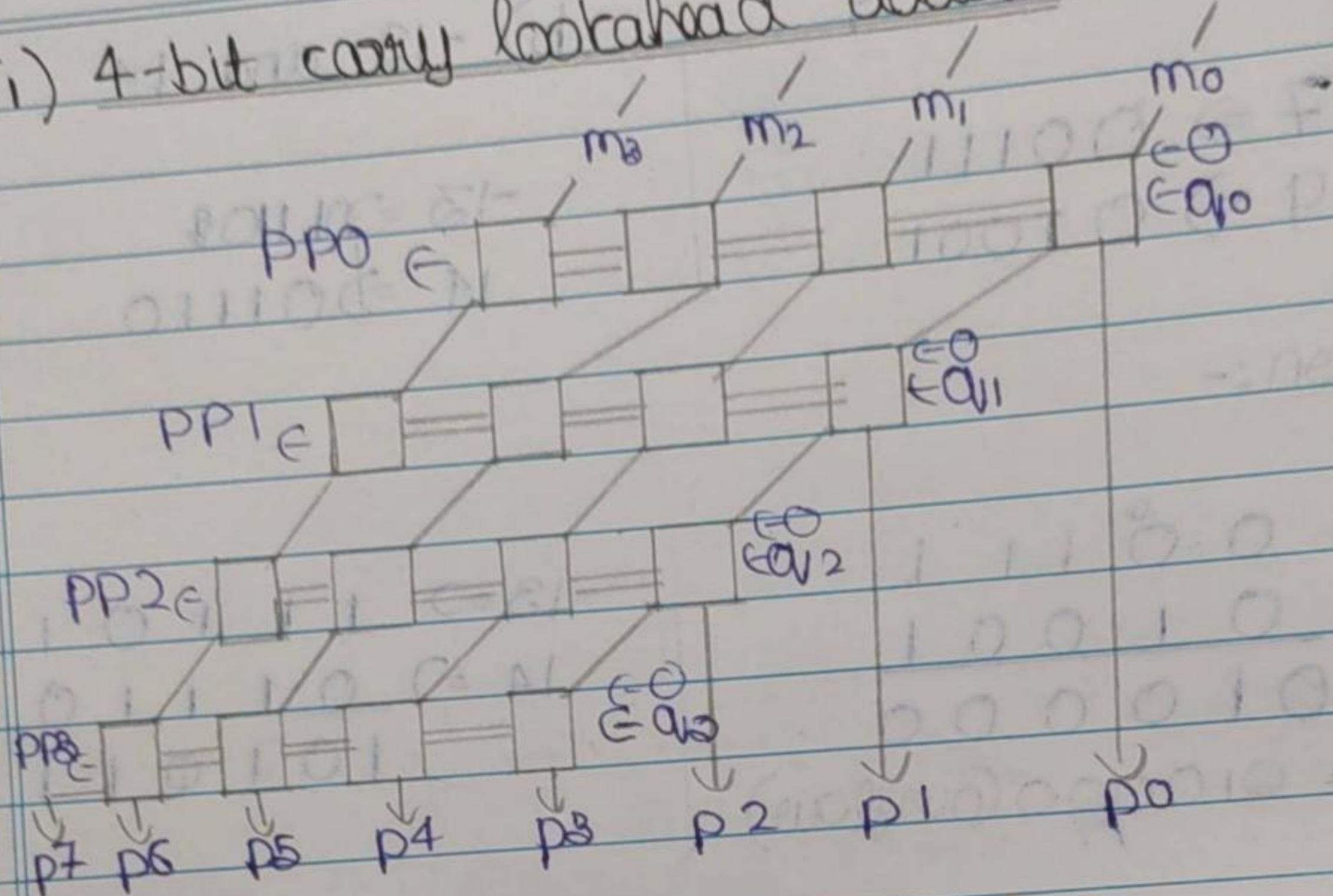
$$011$$

$$011$$

$$011$$

$$01$$

13. i) 4-bit carry lookahead adder



Logical circuit

$$D_0 = \sum_i y_i + (g_i + y_i)c_i$$

$$C_0 = g_i + p_i c_i$$

Delay 0 $D_0 = g_0 + p_0 c_0$

Delay 1 $D_1 = p_1 + p_1 g_0 + p_1 p_0 c_0$

Delay 2 $D_2 = p_2 + p_2 p_1 + p_2 p_1 g_0 + p_2 p_1 p_0 c_0$

Delay 3 $D_3 = p_3 + p_3 p_2 + p_3 p_2 p_1 + p_3 p_2 p_1 g_0 + p_3 p_2 p_1 p_0 c_0$

13 ii) $45 \Rightarrow 101101$

$63 \Rightarrow 111111$

2835

101101 * 11111
 1111101101010 0110000
 1111101101100 1000000
 111101101 * * 1000000
 11101101 * * * 1000000
 1101101 ** * * 1000000
 101101 * * * * 1000000
 1011100010011
 2048 1024 512 256 128 64 32 16 8 4

Ans : 0101100010011

Q → 101101 * 11111①

R → 0000000

101101	11111	Add
010110	11111①	Right shift
000011	11111	Add
000001	11111①	CS
101110	11111	Add
010111	01111①	CS
000100	01111	Add
000010	00111①	CS
101111	001111	Add
010111	10011①	CS
000100	100111	Add
000010	01001①	CS
101111	010011	Add
010111	10100①	CS

0,84P81
010111
0000100

(7)

000100 11010010 Add 01
 000010 01010010821
000001 00101001101

000101101
 000101101
 000101101
 000101101
 000101101

Result: 101001101

Binary: 101001101

Binary: 000000000

Binary: 101001101

Binary: 0110100

Binary: 1100000

Binary: 1000000

Binary: 0111101

Binary: 111010

Binary: 0010000

Binary: 0100000

Binary: 111101

Binary: 111010

Binary: 0010000

Binary: 0100000

Binary: 111101

Binary: 111010

Binary: 0010000

Binary: 0100000

Binary: 111101

Binary: 111010

Binary: 0010000

134980

111010

0100000

Point A

Q.P is the Q.P

Q.

1. Range of integer by n bit 2's complement number system is -2^{n-1} to $2^{n-1} - 1$.

2. 8 bit 2's complement representation :-

$$43 \Rightarrow 00101011$$

3. $I = X - Y$. To avoid overflow 'n' bits of I would require.

- 4.
- i) Array = Index mode
 - ii) LOOPS = Base with index or offset
 - iii) Pointers = Register indirect mode
 - iv) Constant = Immediate mode

5.

x_i	y_i	c_i	c_{i+1}
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	1

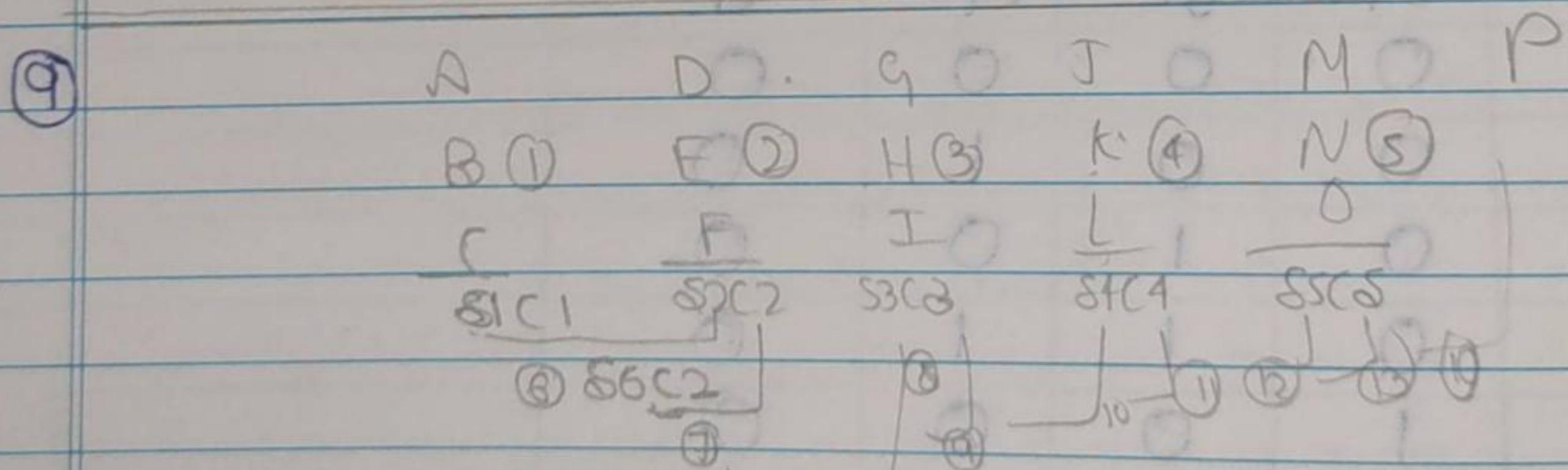
(9)

⑥ $A = 1111 \ 1010$
 $B = 0000 \ 1010$

$$\begin{array}{r}
 1111 \ 1010 * 0000 \ 1010 \\
 000000000000 \\
 111111111010 * \\
 000000000000 * \\
 11111111010 * \\
 000000000000 * \\
 000000000000 * \\
 \hline
 000000000000 * \\
 000000000000 *
 \end{array}$$

$1010010100 = BA$

⑦ 11111110 - 8 addition operation
 For hundred ones, 100 operations are reduced.



14 levels
 comparators are required

10.

