MANIPAL INSTITUTE OF TECHNOLOGY
MANIPAL
(A constituent unit of MAHE, Manipal)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

III SEMESTER B.TECH. (CSE)

IN SEMESTER EXAMINATION

SUBJECT: COMPUTER ORGANIZATION AND ARCHITECTURE (CSE 2151)

Time:10.30AM- 12.00 NOON Date: 15/12/2021 MAX.MARKS: 20

Scheme of Evaluation Set1

1 Calculate the sum of 14.5 and 23.5 in IEEE 32-bit format representation. Show all the steps clearly and represent the final answer in both IEEE 32-bit format and decimal representation. 4

Step i: Convert 14.5 to binary representation **(0.5)**

1110.1→ After normalizing: 1.1101×2^3

$E' = 3+127 = 130 = 10000010_{(2)}$

In IEEE 32-bit format: 0 10000010 11010000000000…….

Step ii: Convert 23.5 to binary representation **(0.5)**

10111.1→ After normalizing: 1.01111×2^4

$E' = 4+127 = 131 = 10000011_{(2)}$

In IEEE 32-bit format: 0 10000011 011110000000000…….

Step 1 : **(0.5)**

Choose the number with the smaller exponent and shift its mantissa right a number of steps equal to the difference in exponents.

14.5 is with smaller mantissa, hence: 1. 1101000…→0.11101000…

Step 2: **(0.5)**

Set the exponent of the result equal to the larger exponent

10000011 (exponent of 23.5)

Step 3: **(0.5)**

Perform addition on the mantissas and determine the sign of the result

$$\begin{array}{r} 1.011110000000000\ + \\ \underline{0.111010000000000} \\ 10.011000000000000 \end{array}$$

Step 4: **(0.5)**

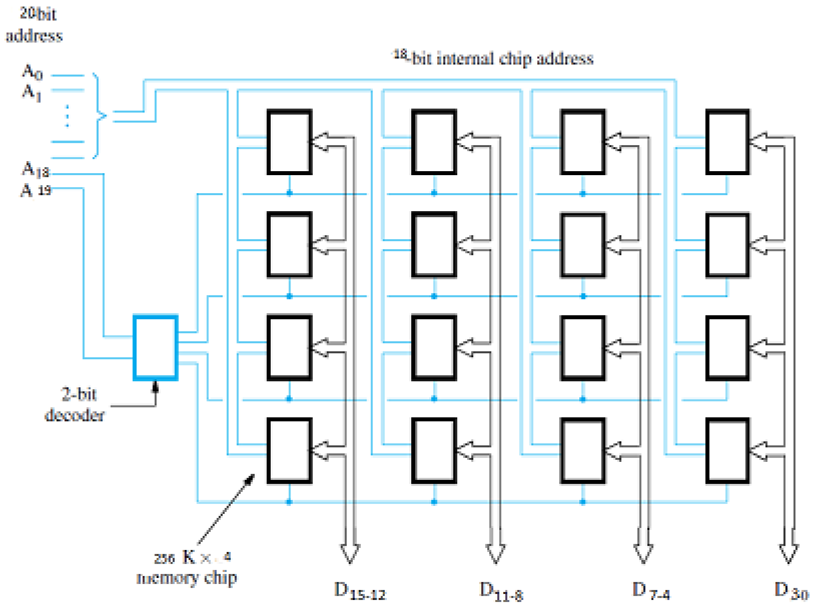
Normalize the resulting value: $1.0011000\ldots \times 2^1$

$E' = 10000011+00000001=10000100_{(2)}$

In 32-bit format: 0 10000100 001100000000… **(0.5)**

In decimal representation: 38.0 **(0.5)**

2 A. Construct a large memory of capacity 1Mx16 using 256K x 4 chips. Draw the structure and indicate clearly, the number of address lines and the data lines in the diagram 3



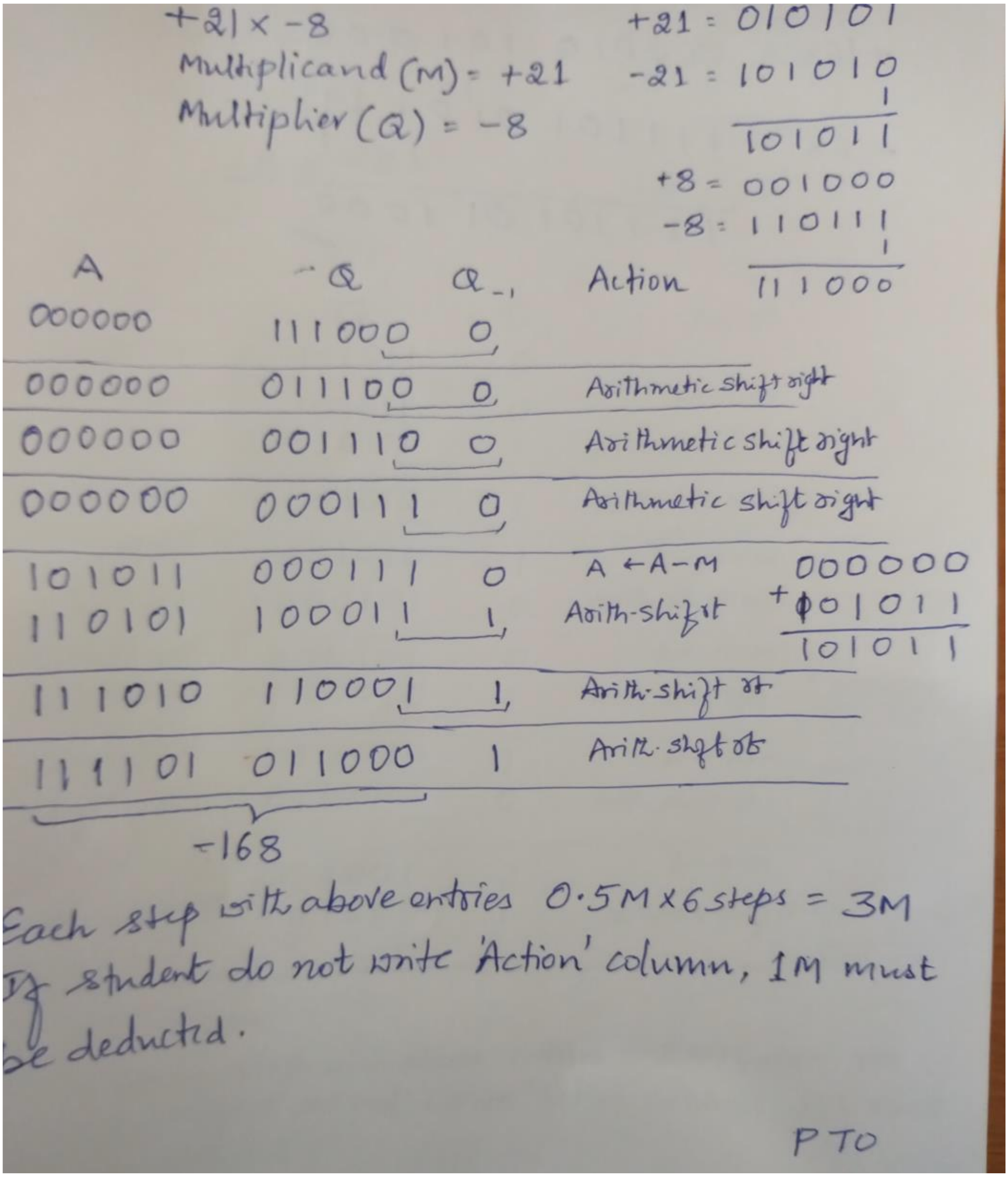
Identifying correct number of rows and colums -> 0.5m
Identifying no. of address lines for Large memory-0.5M
Identifying no. of address lines for small memory-0.5M
Writing decoder with proper inputs and outputs-0.5M
Writing proper data lines : 0.5M
Overall diagram: 0.5M

B. Mention any two comparison between hardwired control unit and microprogrammed control unit 1

Any 2= 0.5*2=1M

1. Microprogrammed approach is more expensive.
2. Control memory may reduce the overall speed of the machine, since microinsructions retrieval process takes significant amount of time.
3. Microprogramming provides a well structured control organization.
4. With Microprogramming, many additions and changes are made by simply changing the microprogram in Control memory, where as a small change in hardwired approach may lead to redesign the entire system.
5. Cost of the control logic increases with system complexity though hardwired logic is economical for simple control algorithm. In microprogrammed implementation cost of the simplest system is higher though adding new features requires additional control memory.

3 Using Booth's algorithm for 2's complement multiplication, show how to multiply the multiplicand (+21) by the multiplier (-8). 3



+21 × −8

Multiplicand (M) = +21

Multiplier (Q) = −8

+21 = 010101

−21 = 101010

+ 1

101011

+8 = 001000

−8 = 110111

+ 1

111000

A	Q	Q_{-1}	Action
000000	111000	0	
000000	011100	0	Arithmetic shift right
000000	001110	0	Arithmetic shift right
000000	000111	0	Arithmetic shift right
101011	000111	0	A ← A−M
110101	100011	1	Arith-shift rt
111010	110001	1	Arith-shift rt
111101	011000	1	Arith. shift rt

000000

+101011

101011

−168

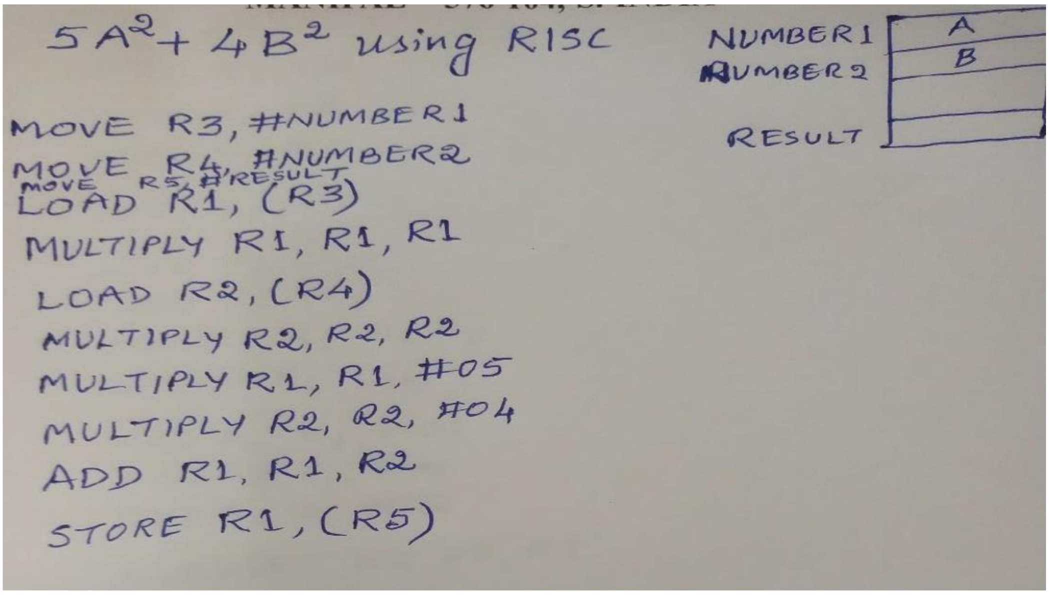
Each step with above entries 0.5M × 6 steps = 3M

If student do not write 'Action' column, 1M must be deducted.

PTO

If student do not write the 'Action steps' then 1M may be deducted.
0.5M for each step X 6 = 3M

4 Assume that an A and B are the two unsigned 8-bit numbers which are stored in consecutive memory locations NUMBER1 and NUMBER2. Write a RISC-style program to evaluate $5A^2 + 4B^2$ and store the result in memory location RESULT assuming that the result do not exceed 8-bit. Use memory pointers to access the data from / to memory. Assume that along with the other instructions, *Multiply* instruction is also defined in the instruction set. Given that the system memory is byte addressable. 3



5A^2 + 4B^2 using RISC

```
MOVE R3, #NUMBER1                    NUMBER1 | A |
MOVE R4, #NUMBER2                    NUMBER2 | B |
MOVE R5, #RESULT
LOAD R1, (R3)                        RESULT  |   |
MULTIPLY R1, R1, R1
LOAD R2, (R4)
MULTIPLY R2, R2, R2
MULTIPLY R1, R1, #05
MULTIPLY R2, R2, #04
ADD R1, R1, R2
STORE R1, (R5)
```

1

1

1

5 Divide 30 by 6 using Restoring division method. Indicate all the steps clearly 3

A	Q	
00000	11110	Initial values M=00110 -M=11010
00001 11010 11011 00110 00001	11100 11100	Shift A,M to the left A=A-M Set Q0=0, Restore A I Cycle
00011 11010 11101 00110 00011	11000 11000	Shift A,M to the left A=A-M Set Q0=0, Restore A II Cycle
00111 11010 00001	10000 10001	Shift A,M to the left A=A-M Set Q0=1 III Cycle
00011 11010 11101 00110 00011	00010 00010	Shift A,M to the left A=A-M Set Q0=0, Restore A IV Cycle
00110 11010 00000	00100	Shift A,M to the left A=A-M Set Q0=0, Restore A V Cycle

	00101	
Remainder	Quotient	

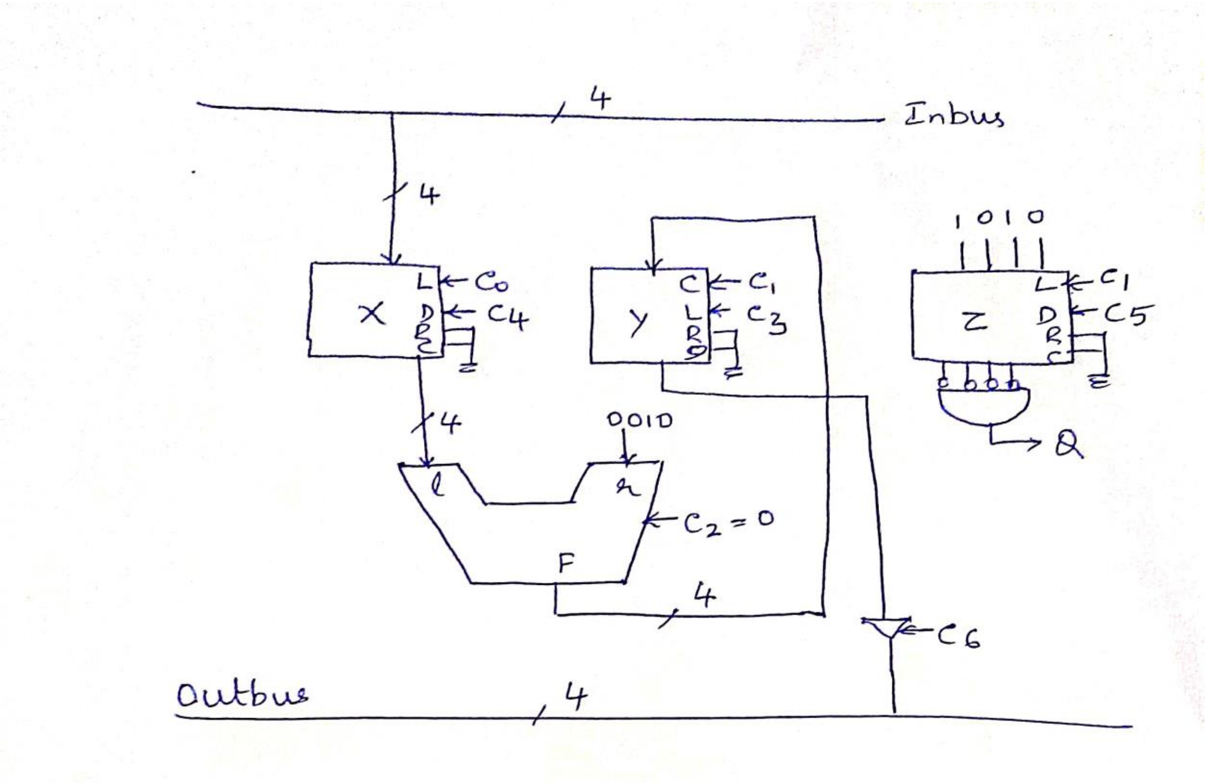
Each step : 0.5 X 5 = 2.5M
Indicating quotient and remainder 0.5M

6 Consider the Register Transfer Description given below: 3

Declare Registers X[4],Y[4],Z[4]
Declare buses Inbus[4], Outbus[4
Start: X←Inbus, Y←0, Z←10;

```
Loop:      Y←X-2
           X←X-1
           Z←Z-1;
      If Z<> 0 then go to Loop
      Outbus = Y;
      Halt: go to Halt
```

Design a neat processing section identifying all the control points



Writing 4 blocks: 0.5M
Showing width of data path and inputs: 0.5M
Identifying control signals and output = 2M
For any mistake : 0.5M deducted