VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

Programme	:	**B.Tech**	Semester	:	**WIN 22-23**
Course	:	**Computer Architecture And Organization**	Code	:	**BCSE205L**
Faculty	:	**Dr. Ilavarasi A K**	Slot	:	
Date	:	**08-02-2023**	Marks	:	**10**

Note:

DIGITAL ASSIGNMENT I

Submit before 20th Feb 2023

- *You can refer any material but cite the material referred.*
- *Solve the answers in hard copy sheet with each page of your answer sheet containing your hand written Name and Register number.*
- *Upon completion of the assignment scan each page with clarity.*
- *The scanned assignment sheets must be converted into single PDF file.*
- *Plagiarism will be checked. Copy/paste solutions are not accepted which will lead to awarding 0/10 in the assignment.*
- *Submission allowed only in PDF format in the LMS portal.*

Attempt all the Questions

1. Illustrate in detail the operational concepts of Von Neumann interconnection architecture for the **addition of the last four digits of your register number**. Example: For 21BCE3028, your analysis should add 3,0,2,8. Assume that these numbers are stored at some known locations in the memory separately. Mention clearly the current status of the memory at the beginning of your analysis andthen proceed.

   i. Write the Assembly code that does the given task.
   ii. Trace the values of Program Counter (PC), Memory Address Register (MAR), Memory Buffer Register (MBR), Instruction Register (IR), Instruction Buffer Register (IBR) and Accumulator (AC) at the end of executing each instruction.
   iii. Write down the register transfer operations involved in executing each instruction in the code.

2. Consider four digits of your register number as N1,N2,N3,N4 such that
   N1= first digit of your register number
   N2= second digit of your register number
   N3=third digit of your register number
   N4=fourth digit of your register number.

   They are stored in consecutive memory locations A and B such that A= N1+N2 and B= N3+N4. Write control sequence to **multiply A and B** using single cycle data path (with diagram) following cases:

   Case 1:

   If A is odd number and B is even number, write the control sequence to perform the task (A is fetched indirectly and B is available in register)

   Case 2:

   If A is even and B is odd, write the control sequence to perform the task (A is available in register and B is fetched indirectly.

   Case 3:

   When both A and B are odd/even, both operands are available in registers and result to be written in a memory location contained in register R3.

   For example:

   If your register number is "21BCE1007" then A= 1+0 and B=0+7. Since A and B are odd numbers you should perform case 3

   If your register number is "21BCE1110" A= 1+1 and B=1+0. Since A is even number and B is odd, you should perform case 2.