|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Name:**  **Enrolment No:** | | |  | | | | |
| Odd Semester Mid Term Examination, Oct 2023Faculty of Engineering| School of Computer and Communication EngineeringDepartment of IoT and ISB. TECH – CSE (IoT & IS ) | | | | | | | |
| Course Code: IS2101 | | Course: Digital Design and Computer Architecture | | | | Semester: III | |
| **Time: 1.5 hrs. Max. Marks: 30** | | | | | | | |
| **Instructions: All questions are compulsory.**  **Missing data, if any, may be assumed suitably.**  **Calculator is not allowed.** | | | | | | | |
| **SECTION A** | | | | | | | |
| S.No. |  | | | | **Marks** | | **CO** |
| Q A1 | Write down “two address code” using register for the following equation:  X=A+B/C  Solution:  LOAD B, R0  DIV C, R0  ADD A, R0  STORE R0, X | | | | 2 | | CO3 |
| Q A2 | Suppose a system is 32-bit and memory is byte-addressable. Memory address begin with 1000. If the location has the value in (FF00AA11). Find the value at location 1002 if the value is represented in little endian and big-endian.  Solution:  FF00AA11  Big Endian: AA   |  |  |  |  | | --- | --- | --- | --- | | 1000 | 1001 | 1002 | 1003 | | FF | 00 | AA | 11 |   Little Endian: 00   |  |  |  |  | | --- | --- | --- | --- | | 1000 | 1001 | 1002 | 1003 | | 11 | AA | 00 | FF | | | | | 2 | | CO2 |
| Q A3 | What are universal gates and why they are called as universal gates.  A universal gate is a gate which can implement any Boolean function without need to use any other gate type. The NAND and NOR gates are universal gates. In practice, this is advantageous since NAND and NOR gates are economical and easier to fabricate and are the basic gates used in all IC digital logic families. | | | | 2 | | CO1 |
| **SECTION B** | | | | | | | |
| Q B1 | Design a 3 X 8 Decoder using two 2 X 4 Decoders.  Digital Circuits - Decoders | | | | 4 | | **CO1** |
| Q B2 | Perform addition using 7-bit signed, 2’s complement binary number representation, on the following pairs of decimal numbers. State whether overflow occurs in each case.   1. (-32)10 and (-32)10 2. (32)10 and (32)10   Solution:   |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | | 1 | 1 | 0 | 0 | 0 | 0 | 0 | -32 | | 1 | 1 | 0 | 0 | 0 | 0 | 0 | -32 | | 1 | 0 | 0 | 0 | 0 | 0 | 0 | -64 |   Overflow= NO   |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 32 | | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 32 | | 1 | 0 | 0 | 0 | 0 | 0 | 0 | +64 |   Overflow= YES  “ Out of Range answer is arithmetically incorrect . | | | | 4 | | **CO2** |
| Q B3 | Identify the addressing mode in the following assembly instructions: | | | | 0.5 X 8=4 | | **CO3** |
| 1. LOAD 20(R1), R5 2. MOVE #3000, R6 3. ADD A, B 4. CLEAR R0 | | | 1. MOVE R1, R2 2. BRANCH > 0 LOOP 3. ADD (R1), R2 4. ADD A, R0 |
|  | 1. LOAD 20(R1), R5: Indexed Mode 2. MOVE #3000, R6: Immediate Mode 3. ADD A, B: Absolute/Direct Mode 4. CLEAR R0: Register Mode 5. MOVE R1, R2: Register Mode 6. BRANCH > 0 LOOP: Direct Mode 7. ADD (R1), R2: Indirect Mode 8. ADD A, R0: Direct and Register Mode Both | | | |  | |  |
| Q B4 | The shift register shown below is initially loaded with 1010. If the clock is applied continuously then after how many clocks pulses, the content of the shift register becomes 1010 again.  A diagram of a flowchart  Description automatically generated  Solution  Solution: 7 clock pulses   |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | CLK | Q3 | Q2 | Q1 | Q0 | Q1 XOR Q0 | S1=Q2 XORY | | X | 1 | 0 | 1 | 0 | 1 | 1 | | 1 | 1 | 1 | 0 | 1 | 1 | 0 | | 2 | 0 | 1 | 1 | 0 | 1 | 0 | | 3 | 0 | 0 | 1 | 1 | 0 | 0 | | 4 | 0 | 0 | 0 | 1 | 1 | 1 | | 5 | 1 | 0 | 0 | 0 | 0 | 0 | | 6 | 0 | 1 | 0 | 0 | 0 | 1 | | 7 | 1 | 0 | 1 | 0 |  |  | | | | | 4 | | **CO1** |
| **SECTION-C** | | | | | | | |
| Q C1 | * A K-map with of three variable is shown below:      1. Write the function for above mentioned k-map in the form of Sum of Min-terms. 2. Write simplified function. 3. Draw the Logic Diagram. 4. Identify gate delay for the same.   Solution:   * Write the function for above mentioned k-map in the form of Sum of Min-terms   A’B’C’+AB’C’+AB’C+A’BC+A’BC’+ABC’   * Write simplified function.   C’+A xor B   * Draw the Logic Diagram. * Identify gate delay for the same.   Total Gate Delay: 2 | | | | 2  2  2  2 | | CO1 |