NSU Term Final Examination (Summer – 2020)

CSE: 332 (Section-3)

Total Marks = 50; Time: 1 hour 30 Minutes

Answer all questions (5x10 = 50)

* *Important Instructions: Late submission will cost 5 marks. Any submission after five minutes will not be evaluated!*
* *Question-6 is a bonus question containing 6 marks. Marks obtained in question-6 will be added to your Mid Term marks not exceed the total of 30.*

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1. | Consider a program having following sequence of instructions, where the syntax consists of an opcode followed by the destination register followed by one or two source registers:  In-order Issue and In-order execution   |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | | Instruction no | Instructions | FETCH | DECODE | EXECUTE | WRITE BACK | CLOCK |  | | 1 | ADD R3, R1, R2 | I-1 |  |  |  | 0 |  | | 2 | LOAD R6, [R3] | I-2 | I-1 |  |  | 1 |  | | 3 | AND R7, R5, 3 | I-3 | I-2 | I-1 |  | 2 |  | | 4 | ADD R1, R6, R0 | I-4 | I-3 |  | I-1 | 3 |  | | 5 | SUB R2, R1, R6 | I-5 | I-4 | I-2 |  | 4 |  | | 6 | AND R3, R7, 15 | I-6 | I-5 | I-3 |  | 5 |  | | 7 | SUB R5, R3, R4 | I-7 | I-6 |  |  | 6 |  | | 8 | ADD R0, R1, R10 | I-8 | I-7 |  |  | 7 |  | | 9 | LOAD R6, [R5] | I-9 | I-8 |  |  | 8 |  | | 10 | SRL R7, R0, 8 | I-10 | I-9 |  |  | 9 |  | |  |  |  | I-10 |  | I-2 | 10 |  | |  |  |  |  | I-4 | I-3 | 11 |  | |  |  |  |  |  | I-4 | 12 |  | |  |  |  |  | I-5 |  | 13 |  | |  |  |  |  | I-6 | I-5 | 14 |  | |  |  |  |  |  | I-6 | 15 |  | |  |  |  |  | I-7 |  | 16 |  | |  |  |  |  | I-9 | I-8 | 17 |  | |  |  |  |  | I-10 |  | 18 |  | |  |  |  |  |  |  | 19 |  | |  |  |  |  |  |  | 20 |  | |  |  |  |  |  |  | 21 |  | |  |  |  |  |  |  | 22 |  | |  |  |  |  |  |  | 23 |  | |  |  |  |  |  | I-9 | 24 |  | |  |  |  |  |  | I-10 | 25 |  |   Assume the use of a four-stage pipeline: fetch, decode/issue, execute, write back. Assume that all pipeline stages take one clock cycle except for the execute stage. For simple integer arithmetic and logical instructions, the execute stage takes one cycle, but for a LOAD from memory, Six cycles are consumed in the execute stage. If we have a simple scalar pipeline that allows only In-order Issue and In-order execution, Show the execution of the program in time steps. |
| 2. | Given the Instruction sets for Machine-1, Machine-2 and Machine-3, write programs to calculate: P= ((B×C) + (A +D)) **/** (E - C × D) and comment on relative efficiency.   |  |  |  | | --- | --- | --- | | Machine-1: One address instructions | Machine-2: Two address instructions | Machine-3: Three address instructions | | STORE X ; X<― [AC]  LOAD X ; AC<― [X]  MPY X ; AC<― [AC x X]  DIV X ; AC<― [AC / X]  ADD X ; AC<― [AC + X]  SUB X ; AC<― [AC -X]  X could be memory address/CPU register/variable | MOV X, Y ; X<― [Y]  MPY X, Y ; X<― [X x Y]  DIV X , Y ; X<― [X / Y]  ADD X, Y ; X<― [Y + X]  SUB X, Y ; X<― [X - Y]  X and Y could be memory addresses/CPU registers/variables | MOV X, Y ; X<― [Y]  MPY Z, X, Y ;Z<― [X x Y]  DIV Z, X , Y ; Z<― [X / Y]  ADD Z, X, Y ; Z<― [Y + X]  SUB Z, X, Y ; Z<― [X - Y]  X , Y and Z could be memory addresses/CPU registers/variables |   …..        …. |
| 3. | Consider a hypothetical Control Unit which supports 8 k words. The Hardware contains 64 internal control signals, 16 bus control signals, 8 Flags and 8 branch conditions. What is the size of control word and control memory in: a) Horizontal Programming b) Vertical programming  For Horizontal programming: For Vertical Programming:  8K x (64+16+3+3+13) = 8K x 99 bits = **99Kbytes** 8K x (6+4+13+3+3) = 8K x 29 bits = **29KBytes** |
| 4. | Use STACK for storing the return address for a procedure return of the following program given below. Show the contents of stack following each CALL and RETURN instructions:   |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | Main Program   |  |  | | --- | --- | | Memory Address | Instructions | | 6000H |  | | ----- |  | | 6050H | CALL Proc 1 | | ---- |  | | 6080H | CALL Proc 2 | |  |  | |  |  | |  | END | | Procedure 1   |  |  | | --- | --- | | Memory Address | Instructions | | 5000H |  | | --- |  | | 5060H | CALL Proc 2 | |  |  | |  | RETURN | | Procedure 2   |  |  | | --- | --- | | Memory Address | Instructions | | 2000H |  | | --- |  | | 2030H | CALL Proc 3 | | ----- |  | |  | RETURN | | Procedure 3   |  |  | | --- | --- | | Memory Address | Instructions | | 3000H |  | |  |  | |  | RETURN | |   Solution   |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | |  | | --- | |  | |  | |  | | --- | | |  | | --- | |  | |  | | 6051H | | --- | | |  | | --- | |  | | 5061H | | 6051H | | -- | | |  | | --- | | 2031H | | 5061H | | 6051H | | --- | | |  | | --- | |  | | 5061H | | 6051H | | …. | |   More   |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | |  | | --- | |  | |  | | 6051H | | --- | | |  | | --- | |  | |  | |  | | --- | | |  | | --- | |  | |  | | 6081H | | --- | | |  | | --- | |  | | 2031H | | 6081H | | -- | | |  | | --- | |  | |  | | 6081H | | -- | | |  | | --- | |  | |  | |  | | ---- | |   Solution ends |
|  |  |
| 5. | Given the signals Ci (i = 1, 2, 3….13) in following diagram, list the activation of control signals in sequential order to execute  ADD M1 (Contents of memory location M1 and AC are added and result is stored in AC)  STORE M2 (Store contents of AC in memory location M2)  Assume C14 is a control signal used to activate or select addition operation in ALU. Also use CR and CW signals to read from memory and write onto memory. Figure below to answer this question. |
| 6. | We assume a superscalar pipeline capable of fetching and decoding two instructions at a time, having two separate functional units (e.g., one integer arithmetic and one floating-point arithmetic), and having two instances of the write-back pipeline stage. Assume the following constraints on a six-instruction code fragment:  Inst-1 is a floating point operation  Inst-2 requires two cycles to execute and depends on output of Inst-1  Inst-3 and Inst-4 conflict for the same functional unit. Inst-3 and Inst-4 are floating point operations.  Inst-5 is an Integer operation  Inst-5 and Inst-6 conflict for a functional unit.  Integer operations can be performed on integer or floating point unit whichever found available but floating point operations must require floating point unit.  Show the states of different stages of superscalar processor at different clock cycles for any instruction issue policy. Please mention the policy you use to answer the question. |