EE2003: Computer Organization

Assignment - 6 : Microprogram Simulation

Due date: Oct 30th 2017, 11:55pm IST

Note:

1. Please use moodle dicussion threads for posting your doubts after checking if the same question has been asked earlier.
2. Submit a single zip file in the moodle named as A6_Rollno.zip containing the .txt file for program execution along with .pdf file containing the required explanation and also the coding sheet.
3. Read the problem to fully to understand the whole procedure.

PROBLEM

The goal of this assignment is to understand how microprocessor instructions are executed at micro level. For this assignment we will be using mic-1 simulator. The simulator works only in Windows OS and doesn't require installation, instead just run mic-1.exe file.

In this assignment you are asked to execute one of the high level instruction such as ADD, SUBTRACT, LOAD etc similar to 8085 instruction set using one of the addressing modes given. The detailed explanation is provided below.

Things to Remember

1. For details regarding the architecture, programming guidelines, instruction set kindly go the link given below
   http://cis.stvincent.edu/carlsond/cs330/mic1/mic1.html

2. Understand the problem statement first and write the program using MAC-1 Instruction set. Check mic1doc.txt in the link given above to see the complete Instruction set. The program has to be included in the .pdf file to be submitted.

3. Once the program is written using MAC-1 instruction set, convert them into Tanenbaum notation. Also include the program in Tanenbaum notation in the .pdf file.

4. Finally convert the program into microinstructions using the Microinstruction Coding Sheet given in the link above. Attach this coding sheet in the .zip file to be submitted

5. Now .txt file has to be manually prepared using the coding sheet. Kindly check the example programs in the above link for more clarity. This .txt file should be submitted with explanatory pdf and the coding sheet in a single .zip file.

PROCEDURE

You are asked to implement one of the high level instructions similar to 8085 instructions using mic-1 microinstructions in the example processor architecture. One of the operands have to be fetched using different addressing modes. Detailed explanation is given below. To know what instruction to implement and what addressing mode to use, kindly check the attached excel sheet select the ones corresponding to your roll number. This assignment is to be done in groups of two.

INSTRUCTION

1. **ADD** Perform addition of numbers. Assume first operand is in Register C with value 7. Fetch the second operand using one of the addressing modes corresponding to your roll number.

2. **SUBTRACT** Perform subtraction of two numbers. Assume first operand is in Register C with value 2. Fetch the second operand using one of the addressing modes corresponding to your roll number.

3. **Logical AND** Perform logical AND of two numbers.Assume first operand is in Register C with value 6. Fetch the second operand using one of the addressing modes corresponding to your roll number.

4. **PUSH** Perform pushing a value to stack. Fetch the operand to be pushed using one of the addressing modes corresponding to your roll number.

5. **LOAD** Load contents of memory location and store the same in A register. Get the memory location using one of the addressing modes.

6. **EXCHANGE** Assume memory location 45 has value 15 initially. Swap the values with another memory location. Second operand has to be fetched using one of the addressing modes corresponding to your roll number.

7. **RIGHT SHIFT** Fetch the required value into the accumulator using one of the addressing modes and perform right shift.

ADDRESSING MODES

1. **Direct Addressing :** Value of 11 should be stored in memory location 49. The memory address 49 should be passed along with the required command.

2. **Indirect Addressing with Auto pre-increment :** Value of 11,12 and 13 should be stored in memory location 49,50 and 51. Now the memory address 50 has to be stored in register B initially. Pass the register B along with the required command.

3. **Indirect Addressing with Auto post-increment :** Value of 11,12 and 13 should be stored in memory location 49,50 and 51. Now the memory address 50 has to be stored in register B initially. Pass the register B along with the required command.

4. **Indirect Addressing with Auto pre-decrement :** Value of 11,12 and 13 should be stored in memory location 49,50 and 51. Now the memory address 50 has to be stored in register B initially. Pass the register B along with the required command.

5. **Indirect Addressing with Auto post-decrement :** Value of 11,12 and 13 should be stored in memory location 49,50 and 51. Now the memory address 50 has to be stored in register B initially. Pass the register B along with the required command.

6. **Program Counter Relative :** Assume the PC value is 45. Store value of 5 in register B. The value in PC should be used as base memory address and value in B as offset address. Values of 11,12 and 13 should be stored in memory location 49,50 and 51. Now pass the register B along with the required command.

7. **Base Register with Offset :** Store that value 45 in D register and use this as base memory address. Store value of 5 in register B. Values of 11,12 and 13 should be stored in memory location 49,50 and 51. Now pass the register B along with the required command.

–end–