**CS 271 Computer Architecture and Assembly Language**

**- Assignment 2 -**

**This assignment is designed to give you practice working with control and repetition structures as well as the instruction execution cycle on the IA32 architecture. Please submit your .asm and .docx files on CANVAS.**

**Instruction Execution Cycle Section**

The instruction LDSUBSTR EAX, mem1, mem2 loads mem1 into EBX, performs a subtraction operation between the values stored in EAX and EBX (EAX – EBX), and then stores the result in mem2.

Preconditions:

* @LDSUBSTR = 0x20A0 in cache memory. 32-bit addresses are used.
* [EAX] = 5
* @mem1 = 0x1500 in main memory. [mem1] = 7
* @mem2 = 0x1504 in main memory, and its current value is unknown.
* The content of all the other registers, memory locations, etc. is unknown.

For every cycle step we will write down the content of each register, memory location, etc.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **IF** | **ID** | **OF** | **EX** | **WB** |
| EIP: 0x20A0  EAX: 5  EBX:  mem1: 7  mem2: ?  EIR:  EID:  ALU Operand 1:  ALU Operand 2:  ALU Result:  MAR:  MDR: | EIP:  EAX:  EBX:  mem1:  mem2:  EIR:  EID:  ALU Operand 1:  ALU Operand 2:  ALU Result:  MAR:  MDR: | EIP:  EAX:  EBX:  mem1:  mem2:  EIR:  EID:  ALU Operand 1:  ALU Operand 2:  ALU Result:  MAR:  MDR: | EIP:  EAX:  EBX:  mem1:  mem2:  EIR:  EID:  ALU Operand 1:  ALU Operand 2:  ALU Result:  MAR:  MDR: | EIP:  EAX:  EBX:  mem1:  mem2:  EIR:  EID:  ALU Operand 1:  ALU Operand 2:  ALU Result:  MAR:  MDR: |