Project - Y86 Simulator
Part B

Y86 Processor

Write a program that simulates the 5 sequential stages of the Y86 processor. The input to this program is the assembly code from the previous Project Part A. Your program should show how the simulator reads each line from memory and pipelines through each of the following stages:

1. Fetch
2. Decode
3. Execute
4. Memory
5. Writeback
6. PC Update

You can use the Ohio State example as the guide for your processor implementation:

http://web.cse.ohio-state.edu/~reeves.92/CSE2421sp13/PracticeProblemsY86.pdf

In particular, refer to page 5 for the organization of the various stages of the processor and page 7 for details of the steps for each kind of instruction. Your output should show the state of the key temporary storage registers after each instruction.

A summary of the various input and output temporary registers are presented below. Please note that not all registers are updated after each operation. The details of which registers are used and updated for each kind of instructions are shown on page 7 of the above Ohio-State document.

- Fetch:
    - Input – Full assembly instruction
    - Output: ifun, icode, rA, rB, valP, valC

- Decode:
    - Input: R[rA], R[rB], R[esp]
    - Output: valA, valB, valE
- Execute:
    - Input: ifun, valA, valB
    - Output: valE, CC
- Memory:
    - Input: valA, valE, valP
    - Output: valM, Memory
- Writeback
    - Input: valE, valM
    - Output: Register
- PC update
    - Input: valP, CC, valC
    - PC