Simulator User Manual – Project Phase II

CSCI 6461 Group 1 (2019 Fall)
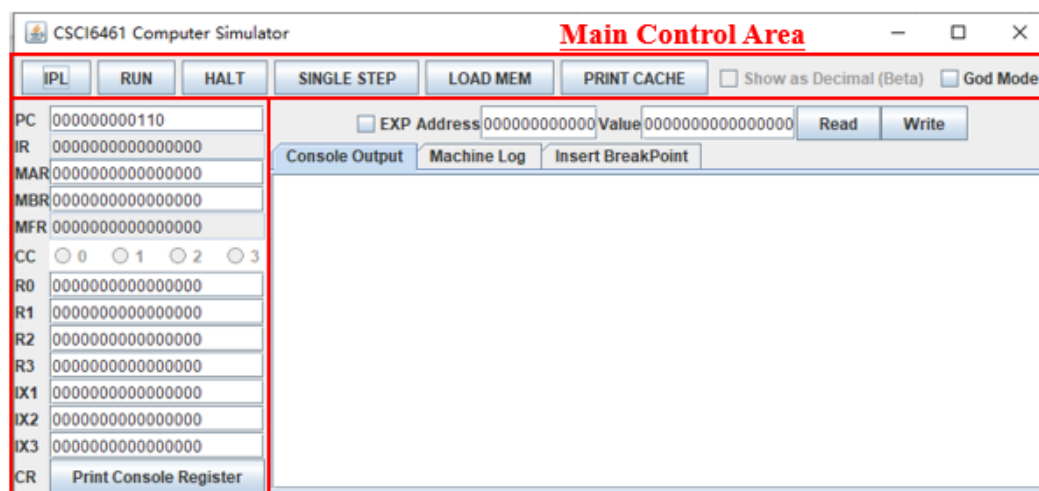

Abdulaziz Alotaibi, Jincheng Yang and Tianyi Qi

The George Washington University

Author Note

A. The User Interface

Figure 1The User Interface of the simulator

Main Control Area

- **IPL Button** will reset the entire simulator. All data inside the memory and registers will be set to initial value.

- **Run Button** will continue to execute instruction driven by PC unless it meets a HALT instruction or manually click the HALT button on the GUI.

- **HALT Button** will send a HALT.

- **SINGLE STEP Button** will execute one instruction from the PC.

- **LOAD MEM Button** will read a CSV file and load the value to the memory.

  The CSV file should have 2 fields.

  <Memory Location(Binary String)>,<Data(Binary String)>.

  Here is an example:

  > 000000000001,0000011100011111
  >
  > 000000000010,0000011100011111
  >
  > 000000000011,0000011100011111
  >
  > 000000000100,0000011100011111

The example above load data "0000011100011111" to memory location 01,10,11,100.

- **PRINT CACHE Button** will print the entire cache data to the "Machine Log" Output Space.

Register Field

- Most of the register value is editable, while some of the registers like IR, MFR, CC is not changeable.

- To change the value of the register, type the value you want to set and press <Enter>.

- **Print Console Register Button** will print all the console register data to the "Machine Log" Output Space. (Console Register is used for IN/OUT Instruction, with DEVID 3-31).

Memory Field

- EXP Click-box is the Memory Expand button. When this box is checked, the max memory is set to 4096 words, while the default is 2048 words.

- To read the memory value, just type the address you want to read, click the Read button, and the value will appear on the value field.

- To write the memory, just type the address and the value and click Write button. If your input is legit, the value will be writing to the memory location.

Console Output

- This is the place for the OUT Instruction to print something. This simulates the "Console Printer" for OUT Instruction.

Machine Log

- This is the place to display all the machine logs, which have a very specific detail to demonstrate what the simulator is doing.

You can check whether the Cache is HIT or MISS, the value of all dumped info, etc.

Insert BreakPoint (DEBUG ONLY)

- This is the place to set a breakpoint. When clicking the RUN button, when PC meets

any of the breakpoint values, the simulator will come to a stop so you can start single step for easy debugging.

Note: This program implements a field engineer's console.

B. Operate the simulator to Testing Program 1

Program 1: A program that reads 20 numbers (integers) from the keyboard, prints the numbers to the console printer, requests a number from the user and searches the 20 numbers read in for the number closest to the number entered by the user. Print the number entered by the user and the number closest to that number. Your numbers should not be 1…10, but distributed over the range of 0 … 65,535. Therefore, as you read a character in, you need to check it is a digit, convert it to a number, and assemble the integer.

To run this testing program, follow the step below:

1. Open the simulator JAR file and click the "IPL" Button.

2. Click the "LOAD MEM" Button and select the testing program 1 CSV file. (File Name: TestingProgram1(P2).csv)

3. Click "RUN"

4. It will require 20 numbers at first. When it pop-up a window to require a number, type the number directly. The range of the input is 0~65535.

5. *For Example: If you would like to input number "234" to be the first number, type "234".*

6. After the 20 numbers you input, it will require a new number. Just type the new number for the program to compare, and the simulator will display the closet number for the 20 numbers you input before and print it to the "Console Output".

Note: There are two files that are attached to this project:
   1. A PDF file to explain the machine interactions in test program1.

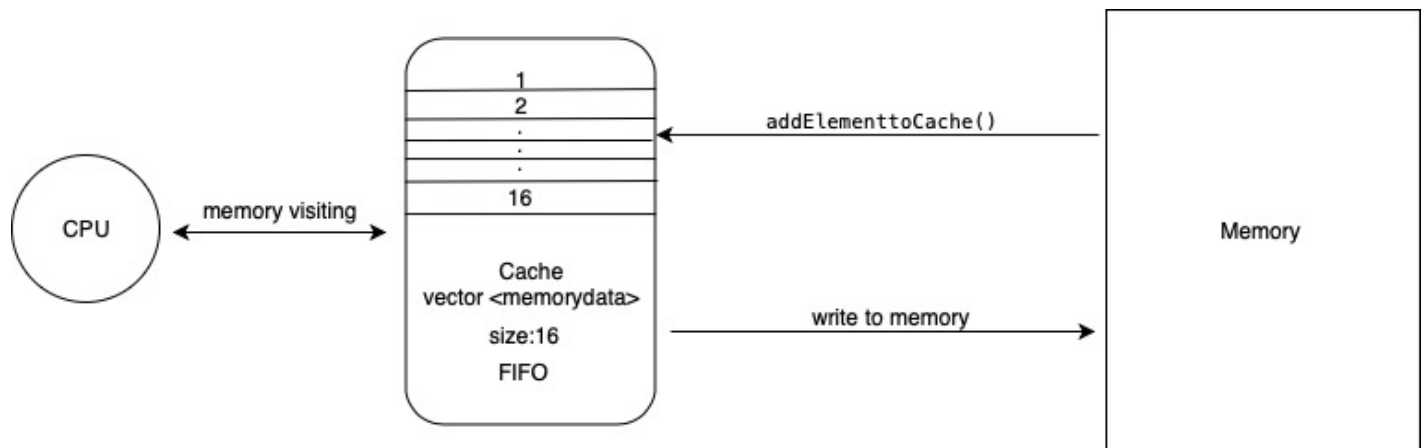2.   **A video file to demonstrate how the test program 1 working.**

C. Memory and Cache design

Figure 2 memory and cache design