User Guide - Machine Simulator

Team 4 -

Members

- Robert Liu
- Rihan Krithik Ramesh
- Samyukta Giridhar

Usage Instructions

1. Download the following files from the blackboard

Download the 'CISC-Simu.jar' for execution of the machine.

In order to read the code, or understand our design notes, you can download the rest of the files available on the folder.

2. Preparation instructions

a. **How to unzip, start or prepare the environment for execution.**

- Once the zip folder has been downloaded to preferred location, user can extract the files in the zip.
- The 'CISC-Simu.jar' can then be executed on your system directly.

> 💡 **Note :** *To execute the jar file with expected output, it is important to have java version 1.8.XX installed.*

b. Locations of other files :

i. The IPL.txt file can be any text file in the users system that needs to be read and executed. There is no need for a specific location.

ii. Code specifically related to the front panel, can be found at src→simulator.

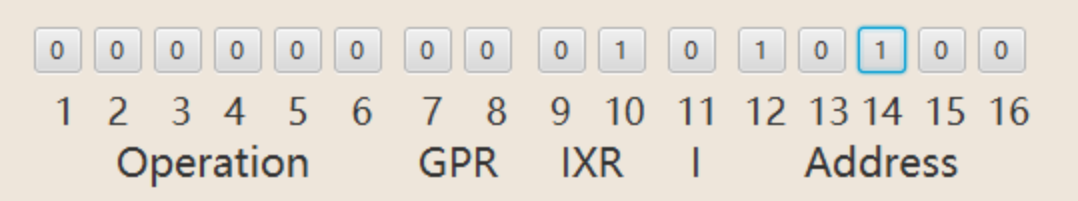
iii. The source java code can be found at src→simulator. It is named as 'Main.java'.

iv. The other files present at src→simulator are :

- FXMLController.java : It contains the functions that take place during an action on the GUI buttons.
- GUI.java : The file contains the code which makes up the interface.
- Instructions.java : This file is responsible for reading the instruction and tear them down into respective codes.
- SimFunc.java : This file contains all the logic implements for the available functions.

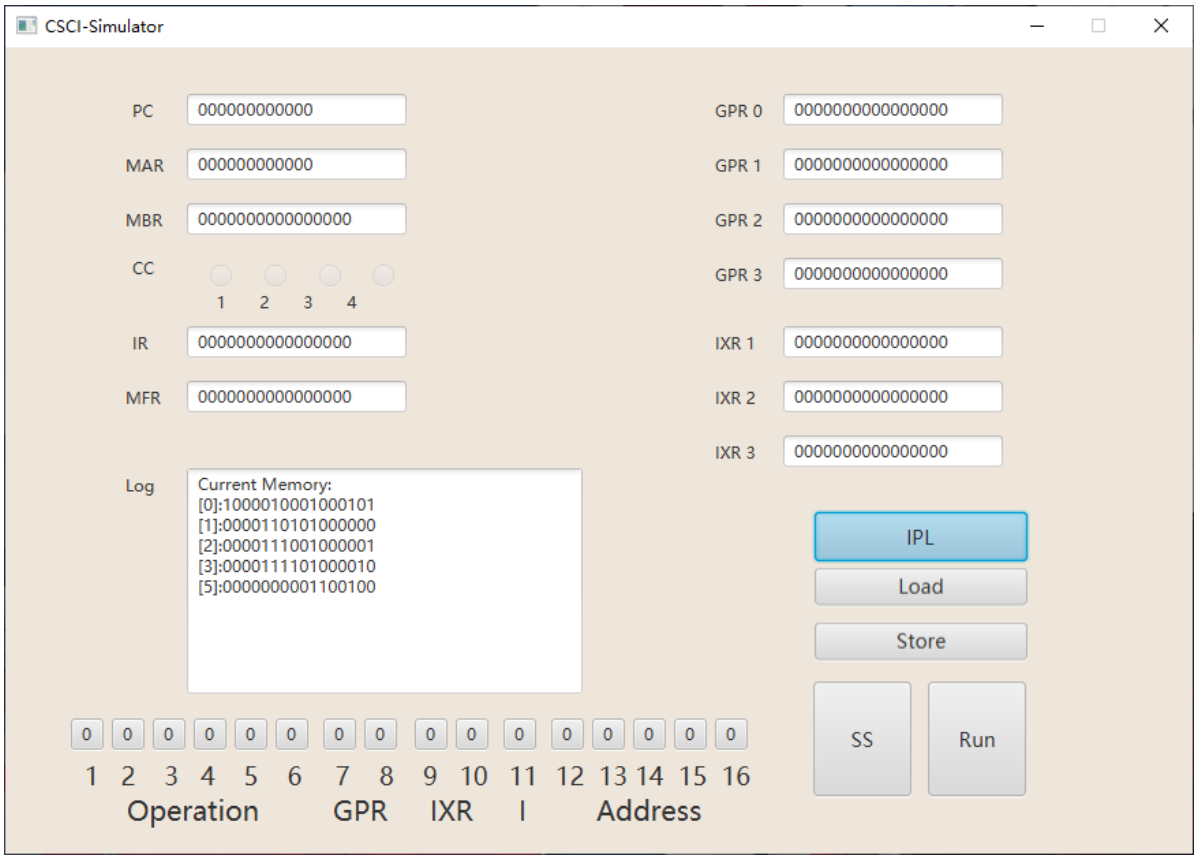
3. **Functions :**

iv. **Manual Instruction :** User can provide manual instruction to specified locations using the bottom values present on your front panel. In a 16 bit input, the first 6 bits are used to provide the operation; bit 7 and 8 are to specify the General Purpose Register named GPR0, GPR1, GPR2 and GPR3; to choose the index register, user can use the 9th and 10th bit, and finally use the 11th bit to execute indirect addressing (setting I bit to 1 makes the instruction as indirect addressing). The last 5 bits (11, 12, 13, 14, 15 and 16) are used to specify the address at which the instruction has to be executed.



v. **Read from external file (IPL) :** The machine can also read instructions from an external file by clicking on 'IPL' and choosing the specific file from where the instruction has to be read. Execution of the instruction takes place the same way as

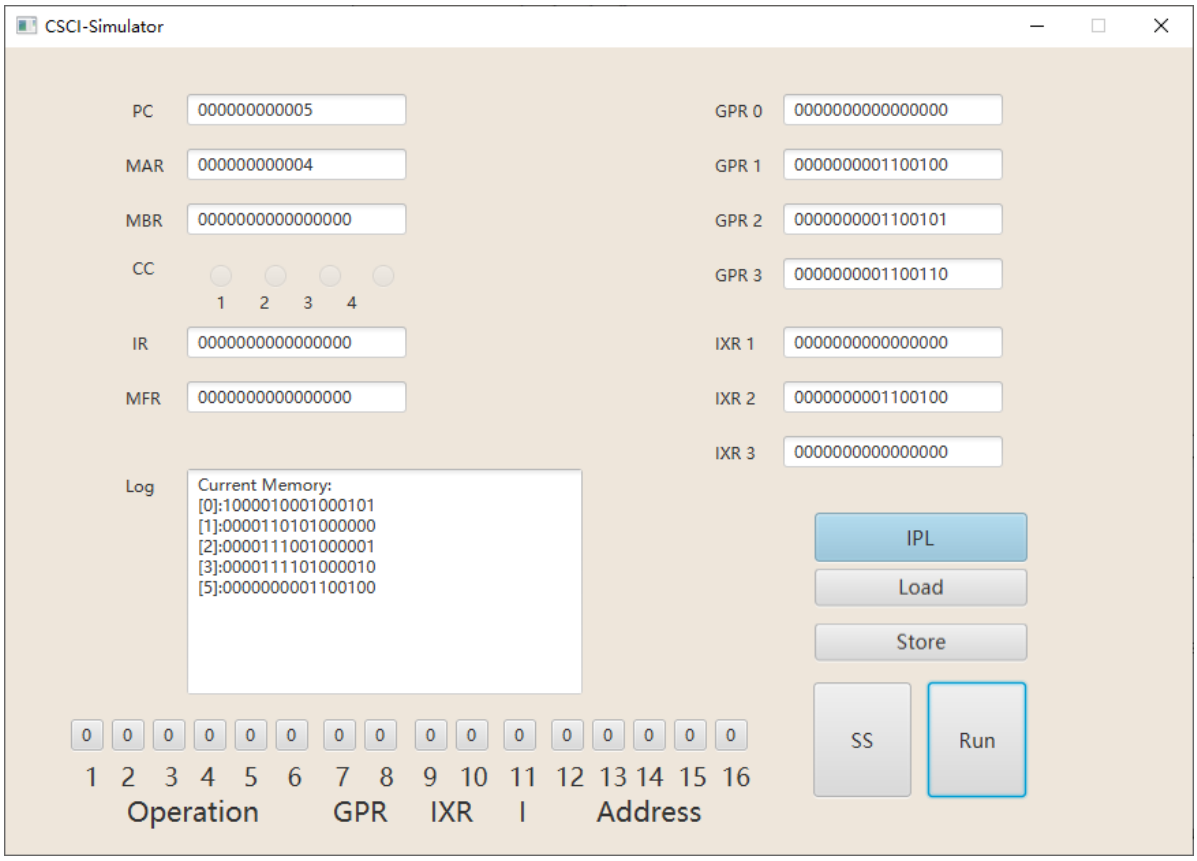
a manual instruction. The file is limited to text format, and upon execution of the function, the system initializes.



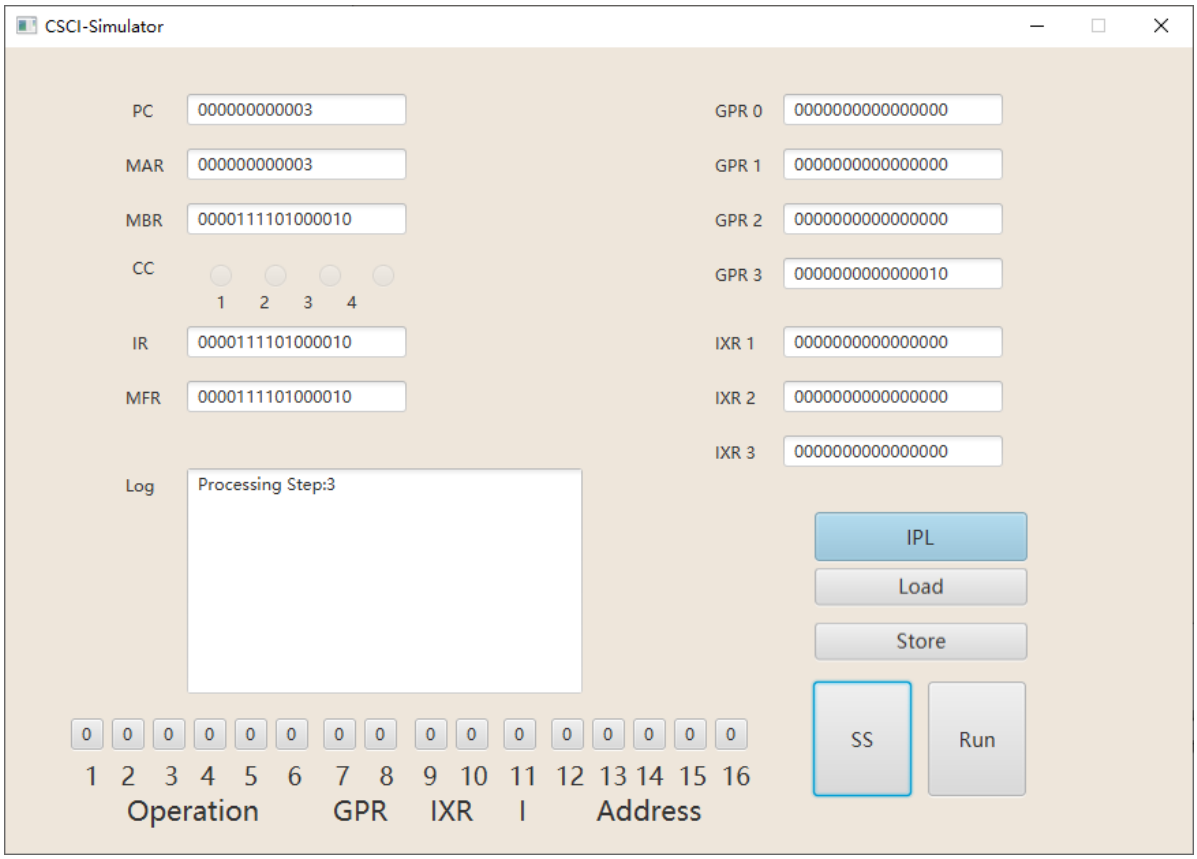
vi. **Store :** The store function can be used for manual inputs, where the value gets stored the selected Register (either GPR or IXR) to memory.

vii. **Load :** Load function helps in loading the registers with the manually inputted 16 bit instruction.

viii. **Run :** The run button will execute the all the instructions that has been loaded into the memory and provide the output on Program Counter (PC), Memory Address Register (MAR), Memory Buffer Register (MBR), Instruction Register(IR), Memory Fault Register(MFR) and the Result in the front panel.



ix. **Single Step :** Single step helps in running only one cycle execution, where only one line of instruction gets executed and the result is shown like how the RUN function works. The user can still modify the 12bit value directly on the PC text-field and the system would execute the instruction. (example: manually change pc's default value of 0 to 1 or above may impact the result)


CSCI-Simulator
PC
000000000003
MAR
000000000003
MBR
0000111101000010
CC
1 2 3 4
IR
0000111101000010
MFR
0000111101000010
GPR 0
0000000000000000
GPR 1
0000000000000000
GPR 2
0000000000000000
GPR 3
0000000000000010
IXR 1
0000000000000000
IXR 2
0000000000000000
IXR 3
0000000000000000
Log
Processing Step:3
IPL
Load
Store
SS
Run
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
Operation
GPR
IXR
I
Address