First discussion 09/07/2014

* Discuss the ClassProjectDescription.doc

1. What to do

Design a simulator of a simple computer, which mainly consists of a CPU, a Memory.

We should design a Graphic User Interface to access the simulator.

1. What to submit

For the fist assignment: a jar file, the source code, a manual for the simulator and group design notes

1. When due

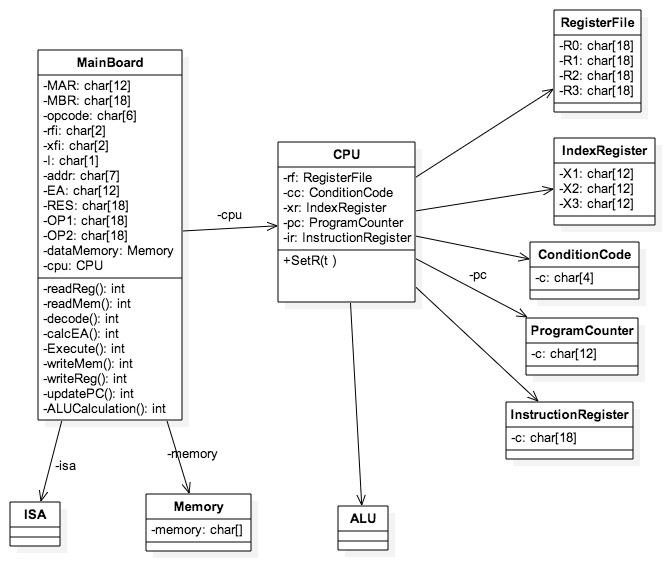
The first assignment due at Sep 17

* Discuss the details of the project
* Discuss the decode processes based on the example from the professor

For the simplicity, we need to change the processed a little and group the whole process into several stages: Decode, Address Calculation, ALU calculation, Execution and PC update.

Second discussion 09/10/2014

* Discuss the class diagram



* Define some terminologies:

R(registerID): read the content of the register

M(memAddr): read the content of the memory

For simplicity, we use one byte, that is, a char, to represent a bit in memory.

* Decode stages specification: decode, address calculation, ALU calculation, execution, PC update

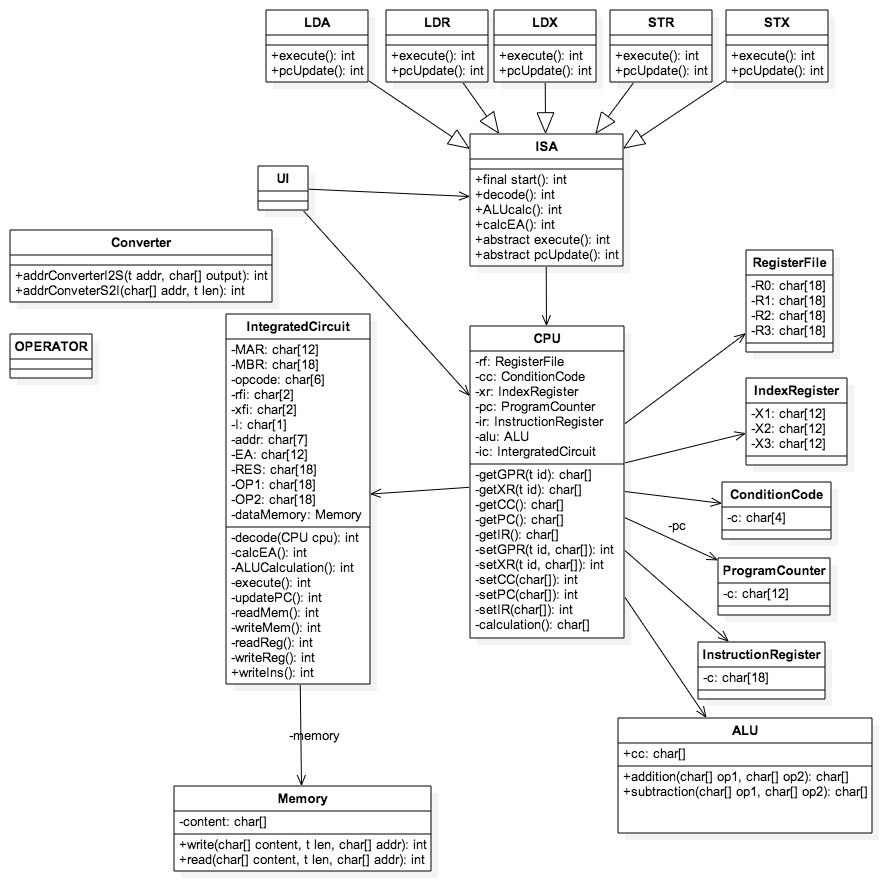
For more decode process, see the file *decode process v0.1.docx*

|  |  |
| --- | --- |
|  | JZ 0, 1, 1, 123 |
|  | 001000 00 01 1 1111011 |
| Decode | MAR <— R(PC)  MBR <— M(MAR)  IR <— R(MBR)  OPCODE <— IR0-5  RFI <— IR6-7  XFI <— IR8-9  I <— IR10  ADDR <— IR11-17 |
| Address Calculation | IF I=0  IF XFI = 0  EA <— ADDR  IF XFI = 1, 2, 3  EA <— R(XFI) + ADDR  IF I=1  IF XFI = 0  EA <— M(ADDR)  IF XFI = 1, 2, 3  EA <— M(R(XFI) + ADDR) |
| ALU Calculation |  |
| Execution | IF R(RFI) = 0  ADDR <— EA  IF R(RFI) != 0  ADDR <— PC + 1 |
| PC Update | PC <— ADDR |

* Summarize the registers needed in every state
* Why this design.

Third discussion 09/14/2014

A new, more complete class diagram

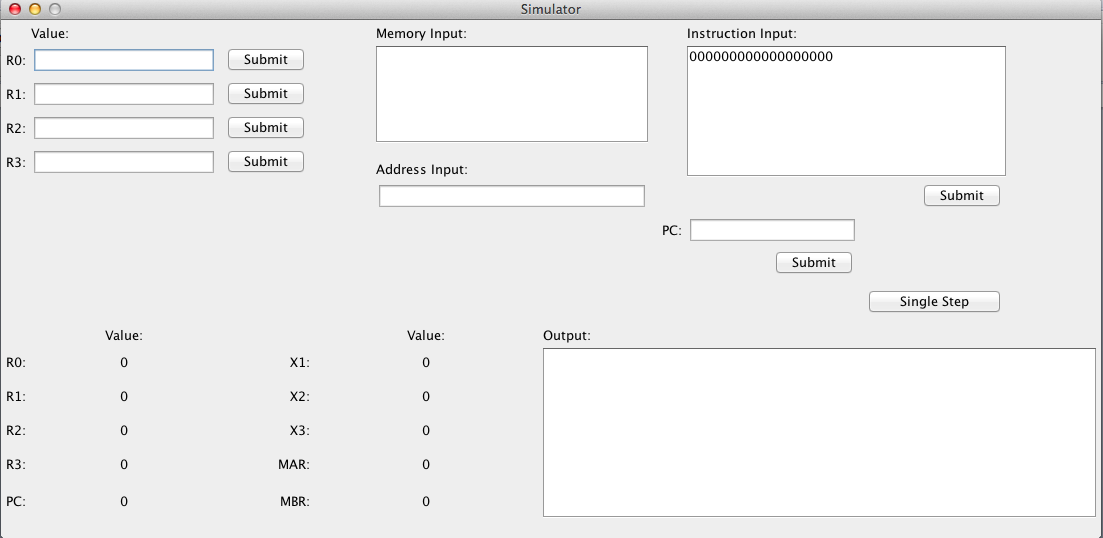


Coding, coding, and debuging…

Discuss the UI

Fourth discussion 09/16/2014

Complete the UI



Integrate the UI and the background logical program.