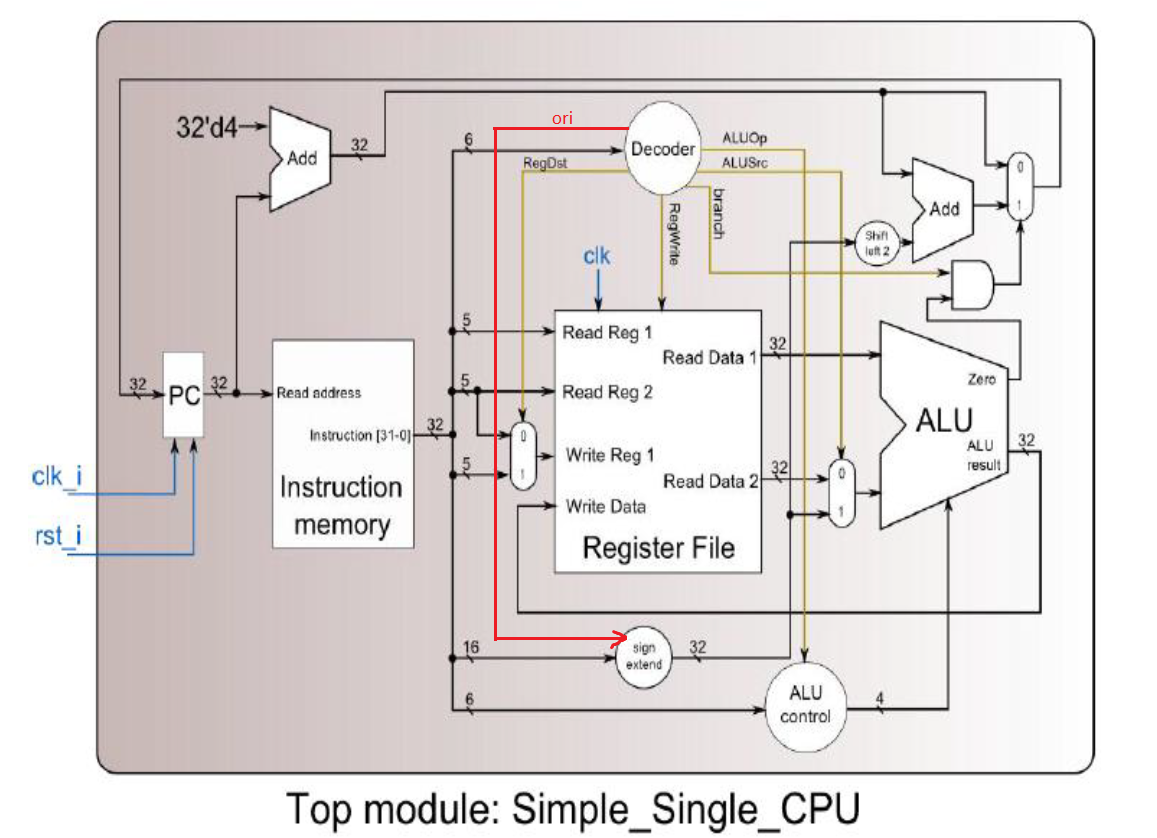
**Computer Organization**

0416021曾香耘 0416246王彥茹

**Architecture diagram:**



**Detailed description of the implementation:**

Done everything required.

The operations are mostly written in ALU, using 32-bit behavioral level design.

Encoded ALUOp and ALUCtrl for several types of instructions.

|  |  |  |
| --- | --- | --- |
|  | ALUOp | ALUCtrl |
| ADD | 010 | 0010 |
| ADDI | 100 | 0010 |
| SUB | 010 | 0110 |
| AND | 010 | 0000 |
| OR | 010 | 0001 |
| SLT | 010 | 1000 |
| SLTU | 010 | 0111 |
| BEQ | 001 | 0110 |
| SLL | 010 | 1001 |
| SLLV | 010 | 1010 |
| LUI | 101 | 1011 |
| ORI | 110 | 0001 |
| BNE | 011 | 0101 |

Two operations are otherwise special that we did some modification on the original design: for the first being ORI, and the second BNE.

For ORI, we added a new wire from the decoder to the sign extender in order to do zero extension instead of sign extension.

For BNE, we give the zero bit from ALU the opposite value such that other design could treat it as how they treat BEQ but receive an opposite result.

**Problems encountered and solutions:**

Single Cycle CPU does not have an output itself, ISE treats everything as redundant hardware and therefore could not be synthesized.

Register File had multiple single edges according to ISE, and that creates and error such that the project couldn’t be synthesized.

We gave up on synthesizing.

**Lesson learnt (if any):**

Insight into a simple single cycle cpu, for instance, the decoding mechanism of the decoder for different types of instructions.