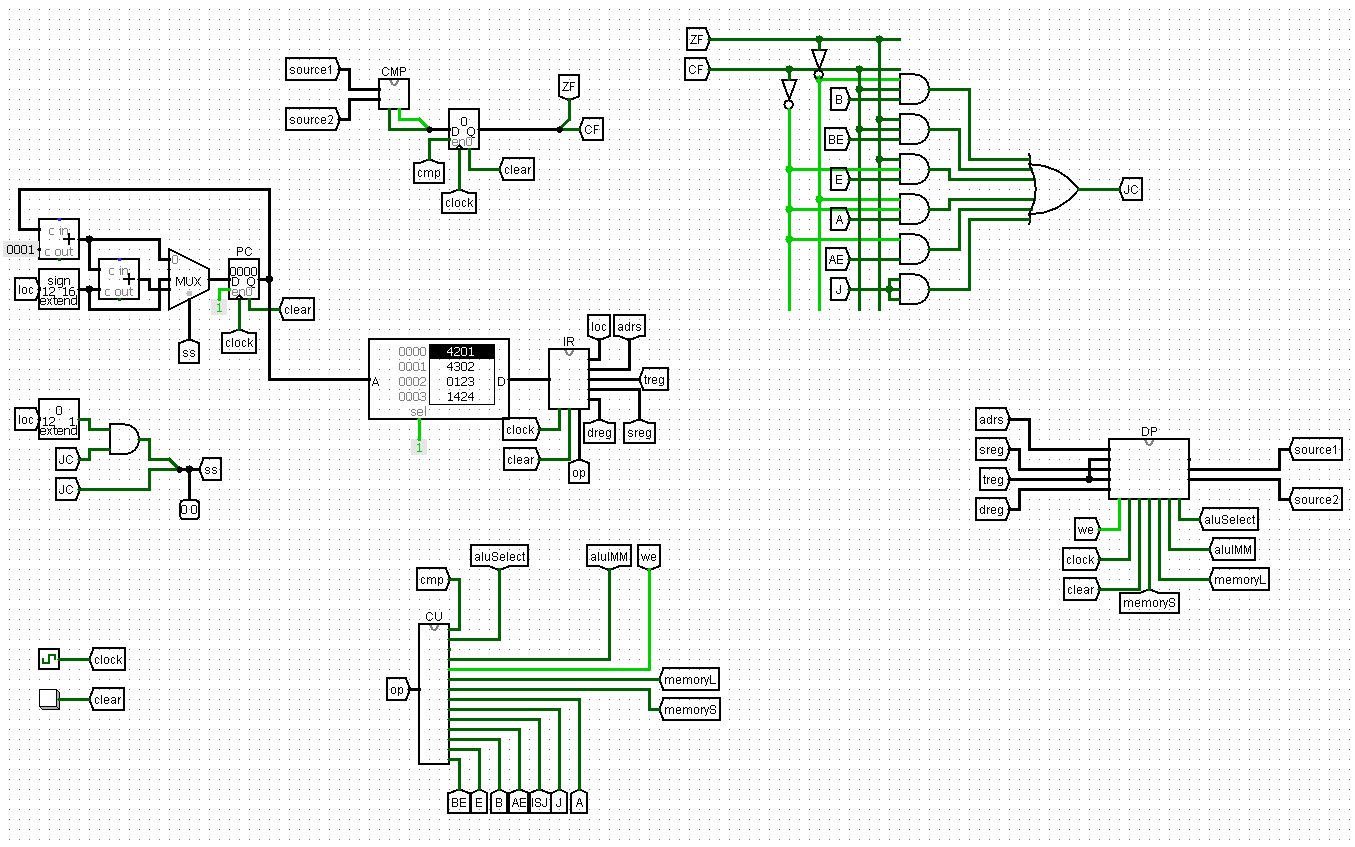
**CSE 3025 Term Project**

**150115010-Mahmut AKTAŞ**

**150115069-Muhammet ŞERAMET**

In this project we wanted to design a CPU. At first we created inside parts of the CPU. These parts are: 16-bit register, ALU, register file, comparetor, instruction register, ControlUnit and DataPath.

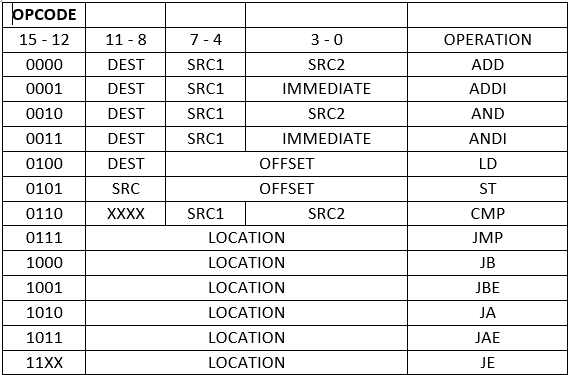
Our project’s main part is shown as below.



Our project’s datapath contains has some basic part as PC (Program Counter), Instruction Memory,ALU(Arithmetic Logic Unit), Register File,Control Unit and Data Memory. PC holds address of instruction to be executed next. This adress is sent to Instruction memory to fetch the instruction. Than fetched instruction is sent to Control Unit to produce appropriate signals for all parts in datapath. ALU handles arithmetic operationsData can be stored to or can be read from Data Memory.

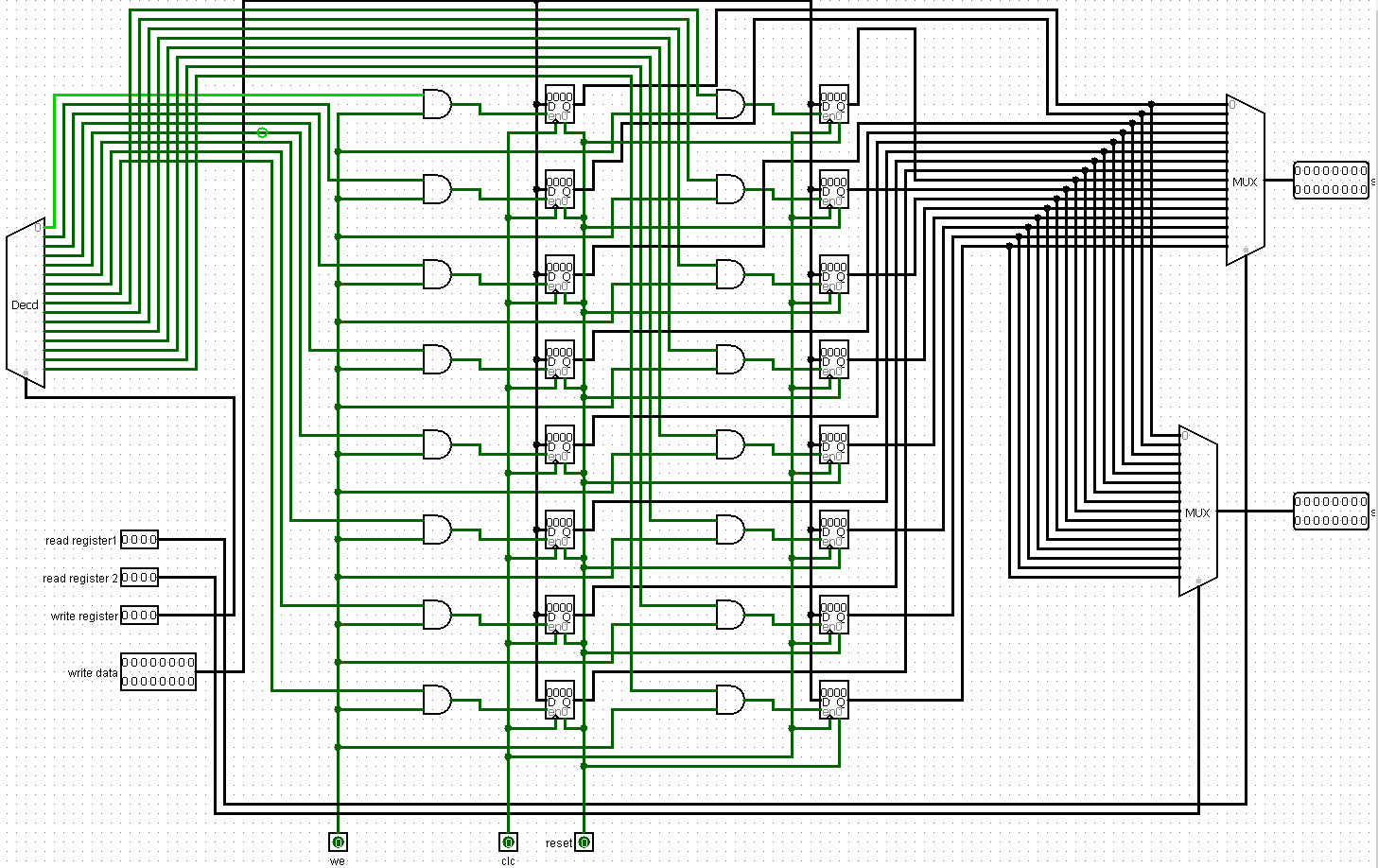
The design supports ADD, ADDI, AND, ANDI, LD, ST, JUMP, CMP, JB, JBE, JA, JAE, JE instructions.

Following table is about the details of these all of things:



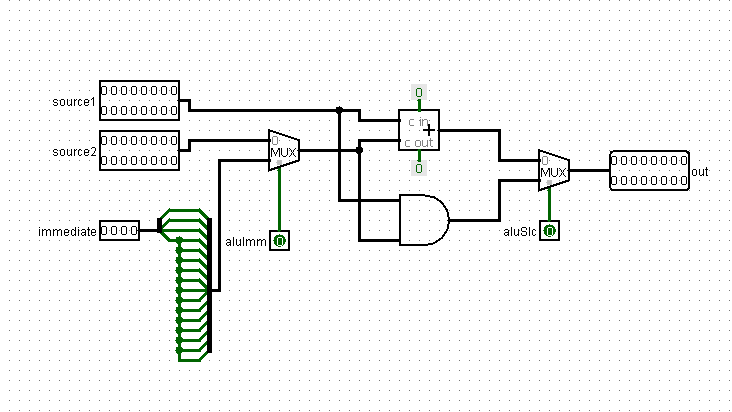
*Register File* contains 16 registers. It has 4 input port SR1,SR2,DR,inData and 2 output port

*OutData1* and *OutData2*. Its controlled by a signal WE(write enable).



*ALU* has 2 inputs *input1* and *input2* which is directed by the values read from *Register File* or

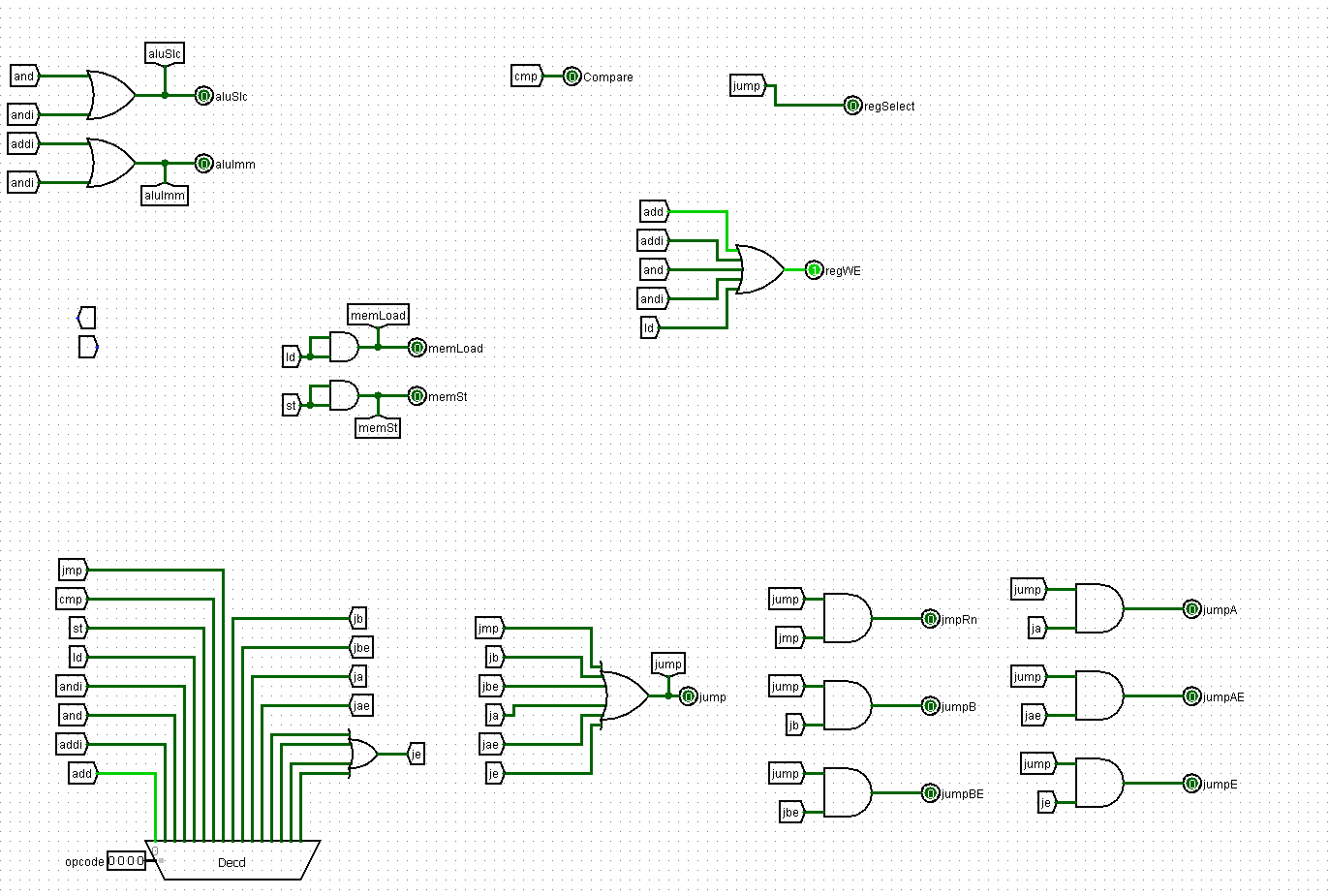
immediate part of the instruction. It has one result as *ALUout*. ALUcontrol signal decides the operation to be done in ALU(*ADD* or *AND* in our case).



*Control Unit* produces appropriate signals for all parts in datapath. Every operation of

instructions are seperated according to their relation with sequential parts in the datapath.

Our Control Unit is as shown in below.



Comments:

At first we wrote an assembly code on Java. We tranformed the instructions to hexadecimal numbers. Then we designed wanted circuit design in logism. We used our registers on this project. We learned verilog for this Project. Then we used modelsim to see our verilog code’s outputs. We just couldn’t write the main part of verilog. Other modules are working properly.