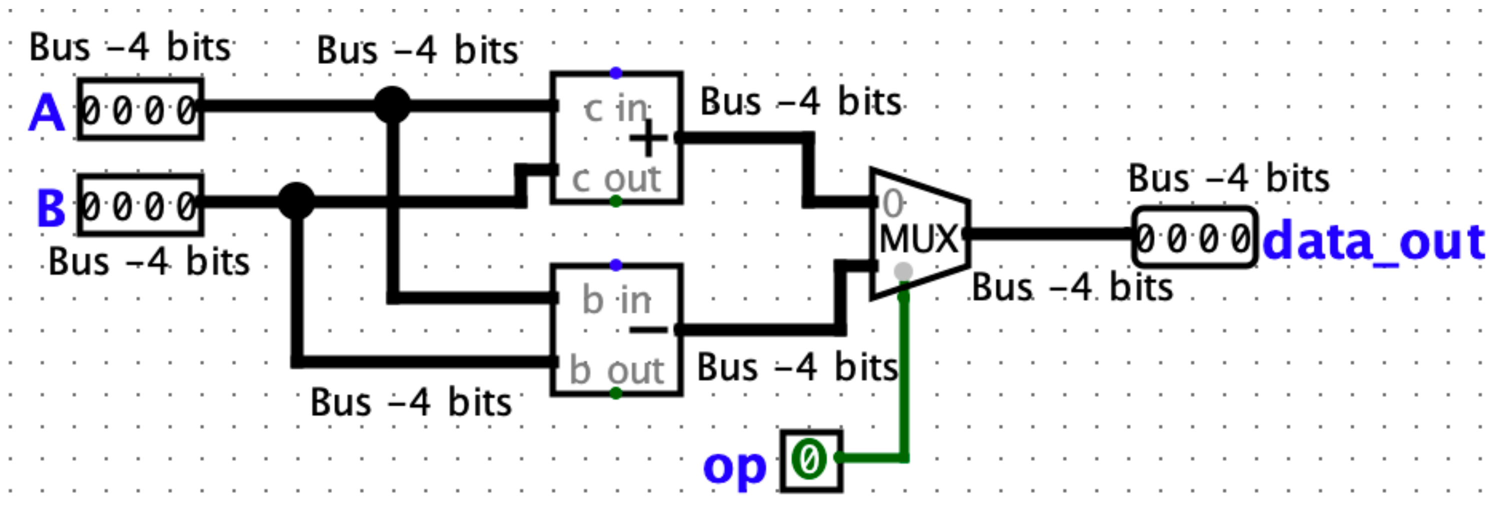
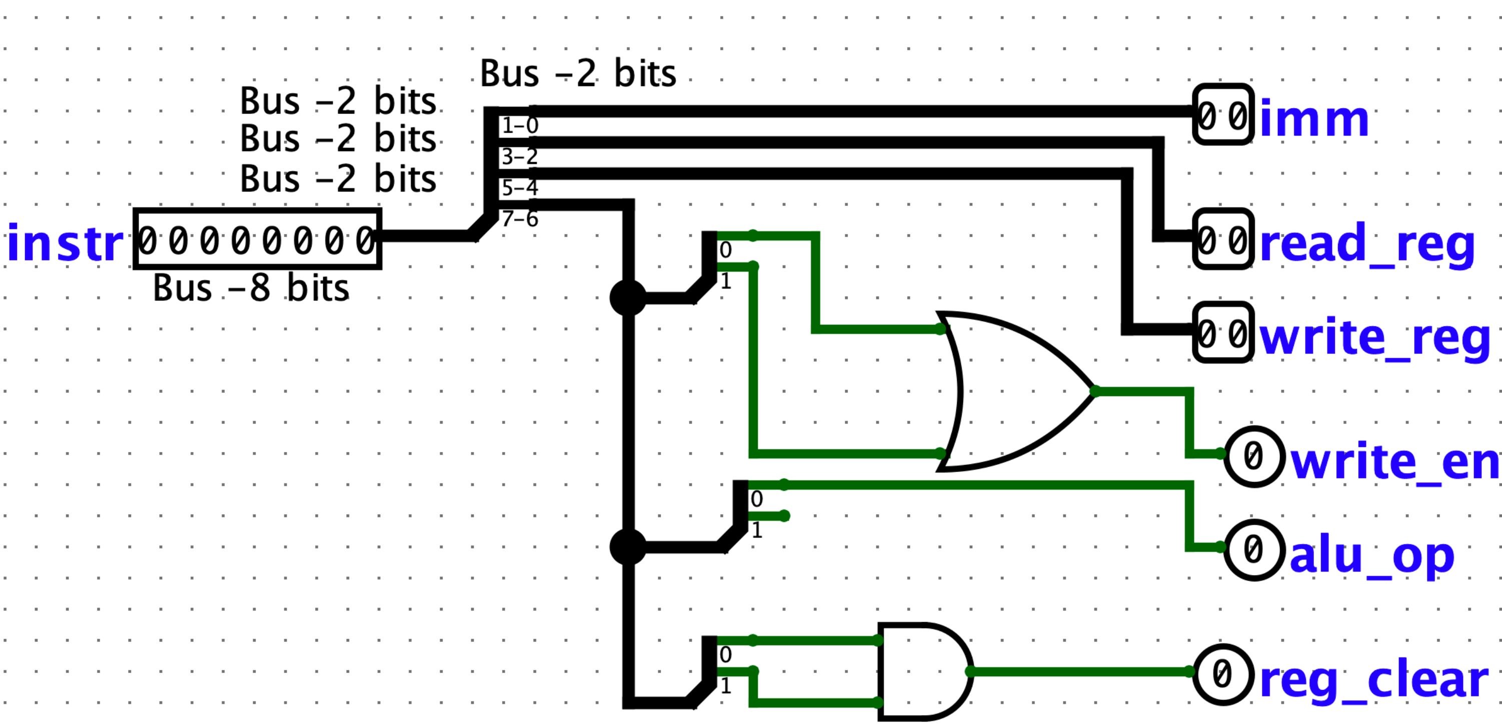
Basmala Moumneh
1006091676

Lab 7:

<u>ALU Diagram:</u>

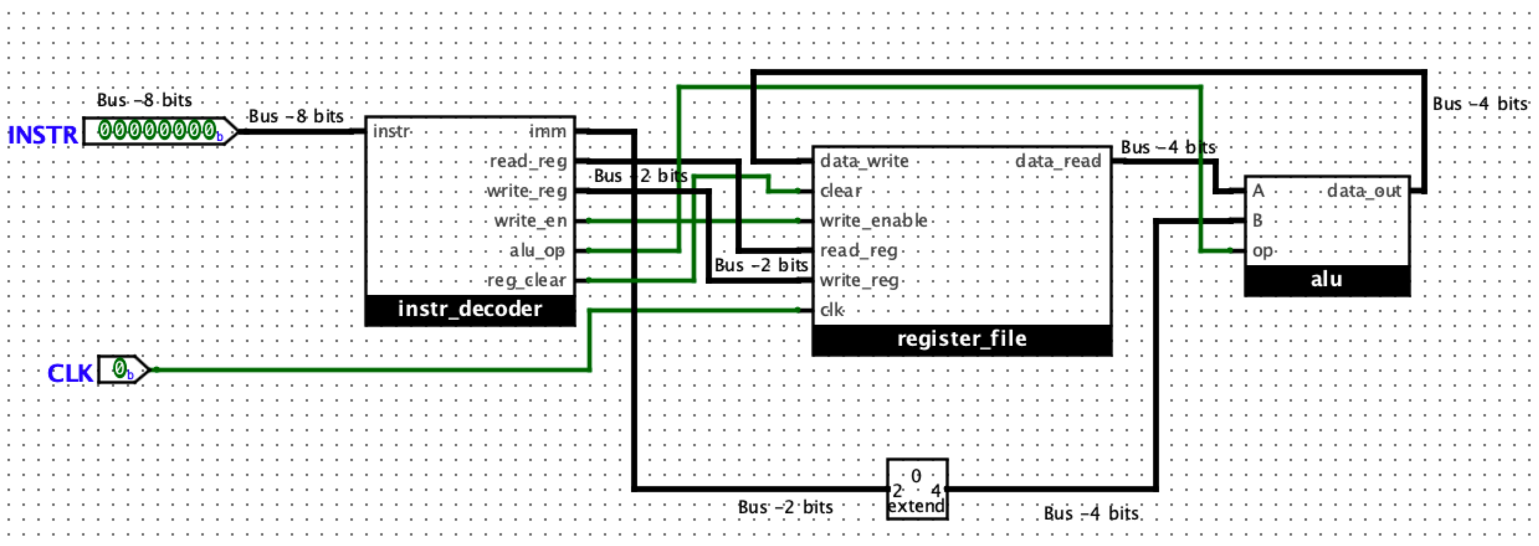


<u>Instruction Decoder:</u>



Basmala Moumneh
1006091676

High-level Logisim Circuit Diagram:



Basmala Moumneh
1006091676

<u>Example of Working Processor:</u>

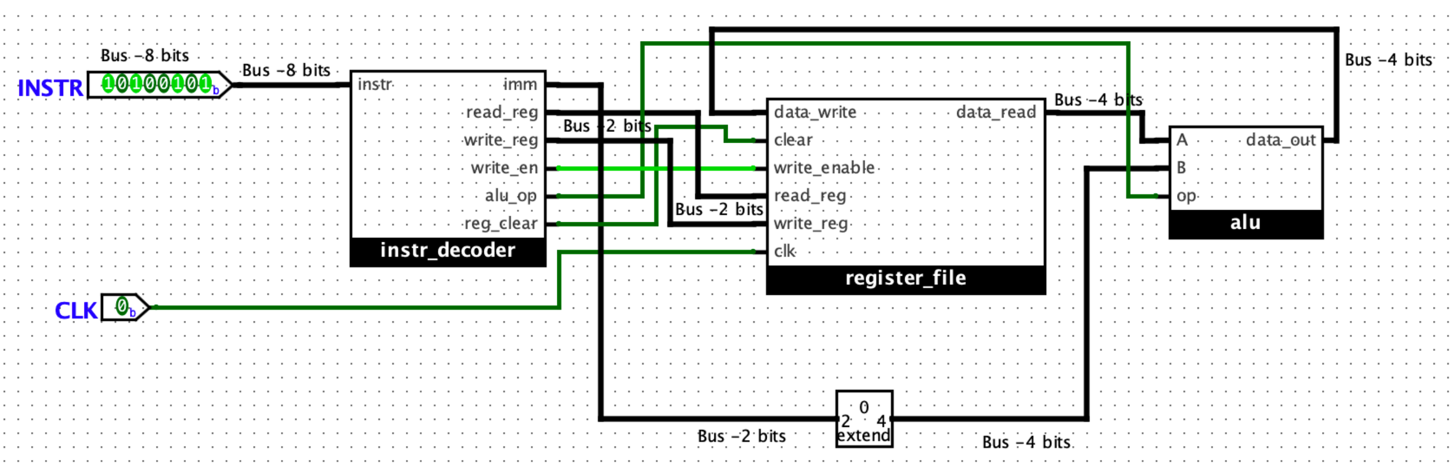
The circuit is reset, all registers have value 0000.

Instruction Code: 10100101

write_reg = register 2
read_reg = register 1
imm = 01
op_code = 10 (operation A – Addition)

→ At first the instruction code 10100101 goes through the instr_decoder and it is dissected. The instruction decoder outputs values: imm = 01, read_reg = 01, write_reg = 10, alu_op = 1, reg_clear = 0, write_en = 1 (an operation will be taking place). This information is then sent to the components that require it to function. The write_enable tells the register file that a value can be written to a register on a positive clock edge. The value imm is sent through a bit-extender (from 2 to 4 bits) so that it can be utilized in the ALU, where it is the value for the B-input (0001). Data is extracted from the read_reg, that value is outputted through data_read and sent to the ALU A-input (0000). The value of the alu_op is used in the ALU to determine which operation is used; in this case alu_op is 0, so the value from Operation A (addition, 0000 + 0001 = 0001) will be outputted through data_out and sent to data_write to be written to register 2. The value is only written to register 2 when the there is a positive edge (clock goes from 0 to 1).

When Clock is 0:

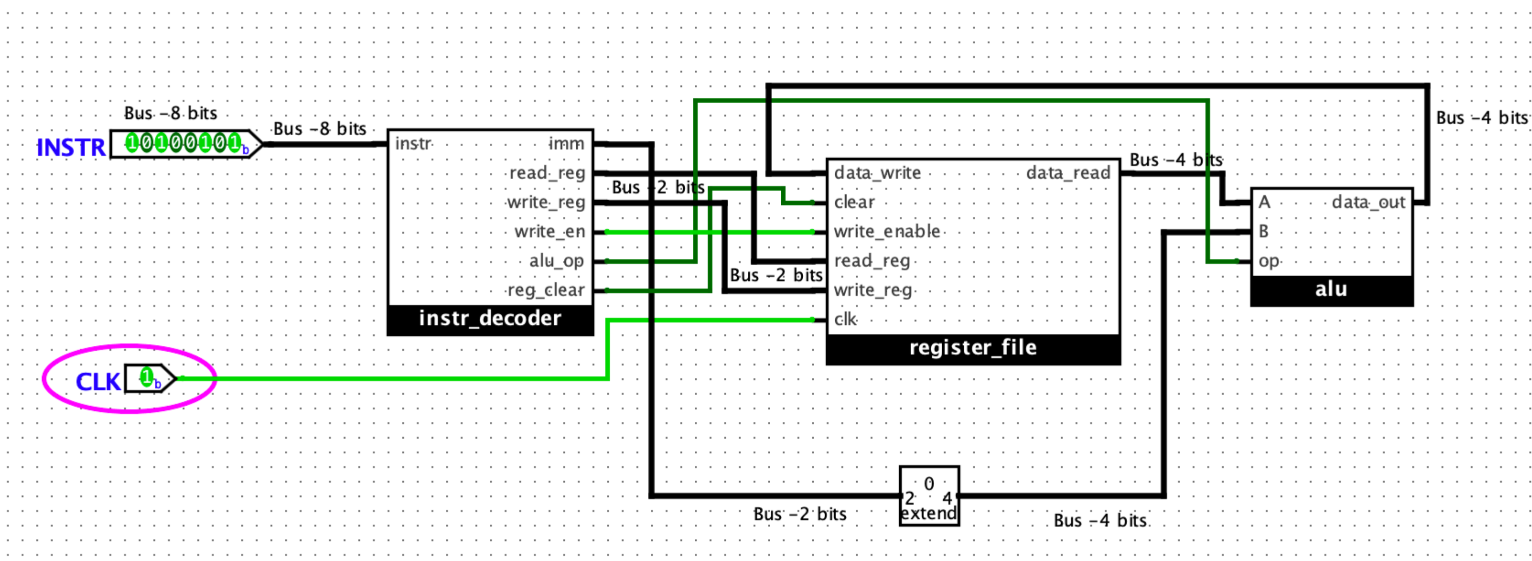


Basmala Moumneh
1006091676

Circuit	Reg name	Value
register_file	reg0	0
register_file	reg1	0
register_file	reg2	0
register_file	reg3	0

Value of register 2 is still 0.

When clock turns to 1:



Circuit	Reg name	Value
register_file	reg0	0
register_file	reg1	0
register_file	reg2	1
register_file	reg3	0

Hence, we can see that the value of register 2 has changed accordingly.