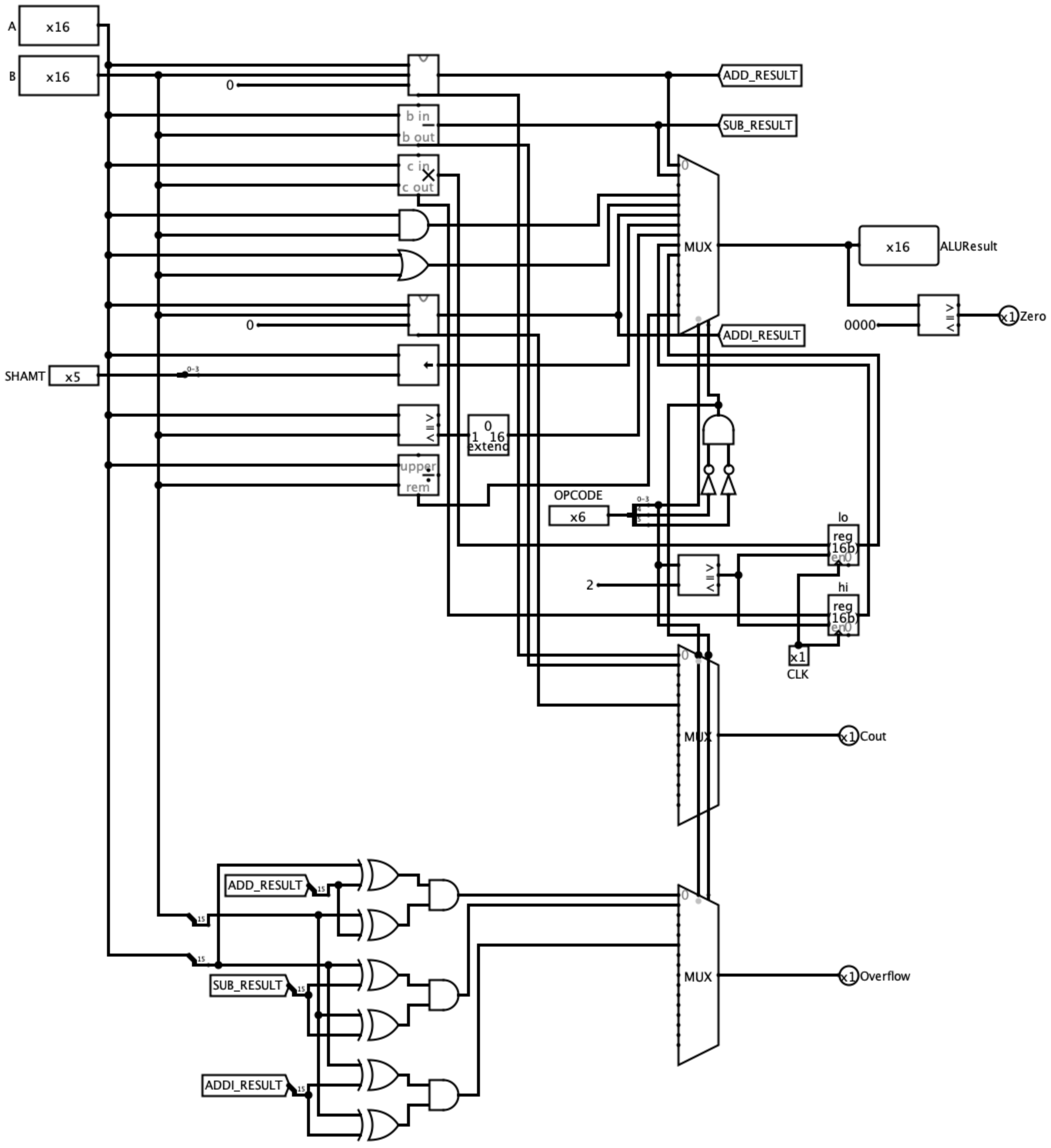
COMP 303 Project

Single Cycle Processor

Hakan Şahin 64355
Yasemin Melek 63929

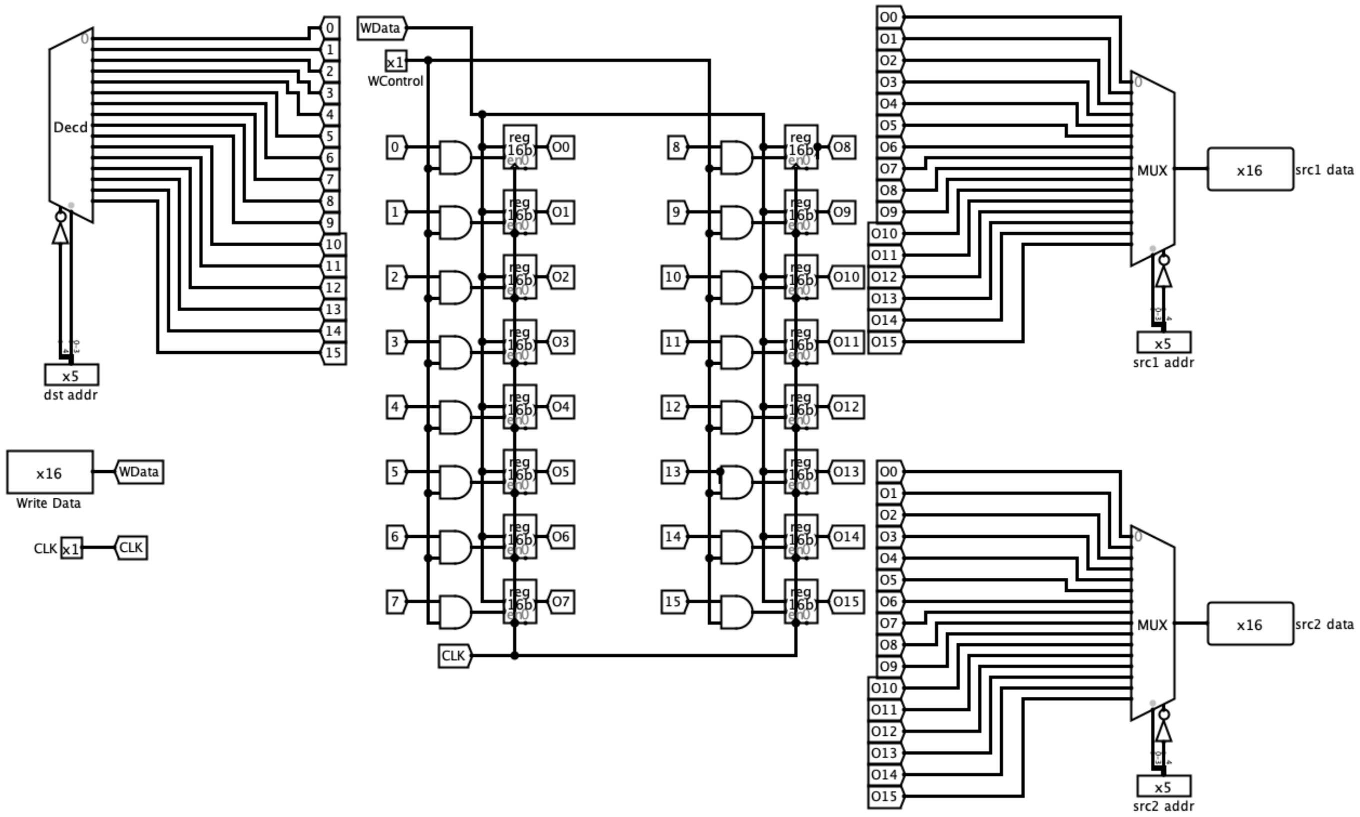
ALU

Hakan Şahin - 64355 / Yasemin Melek - 63929

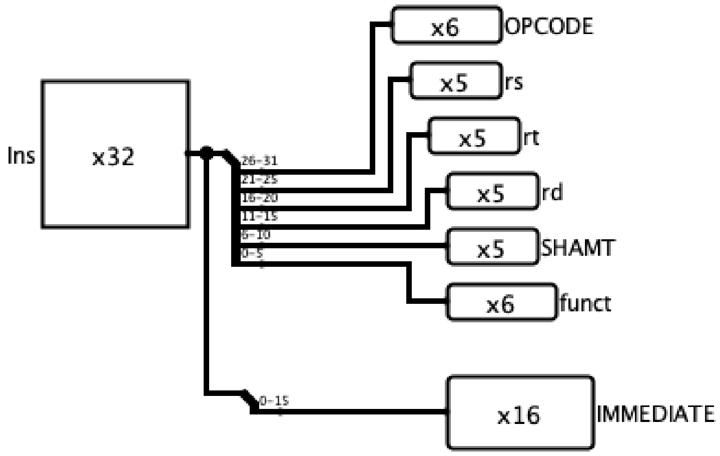


Register File

Hakan Şahin - 64355 / Yasemin Melek - 63929

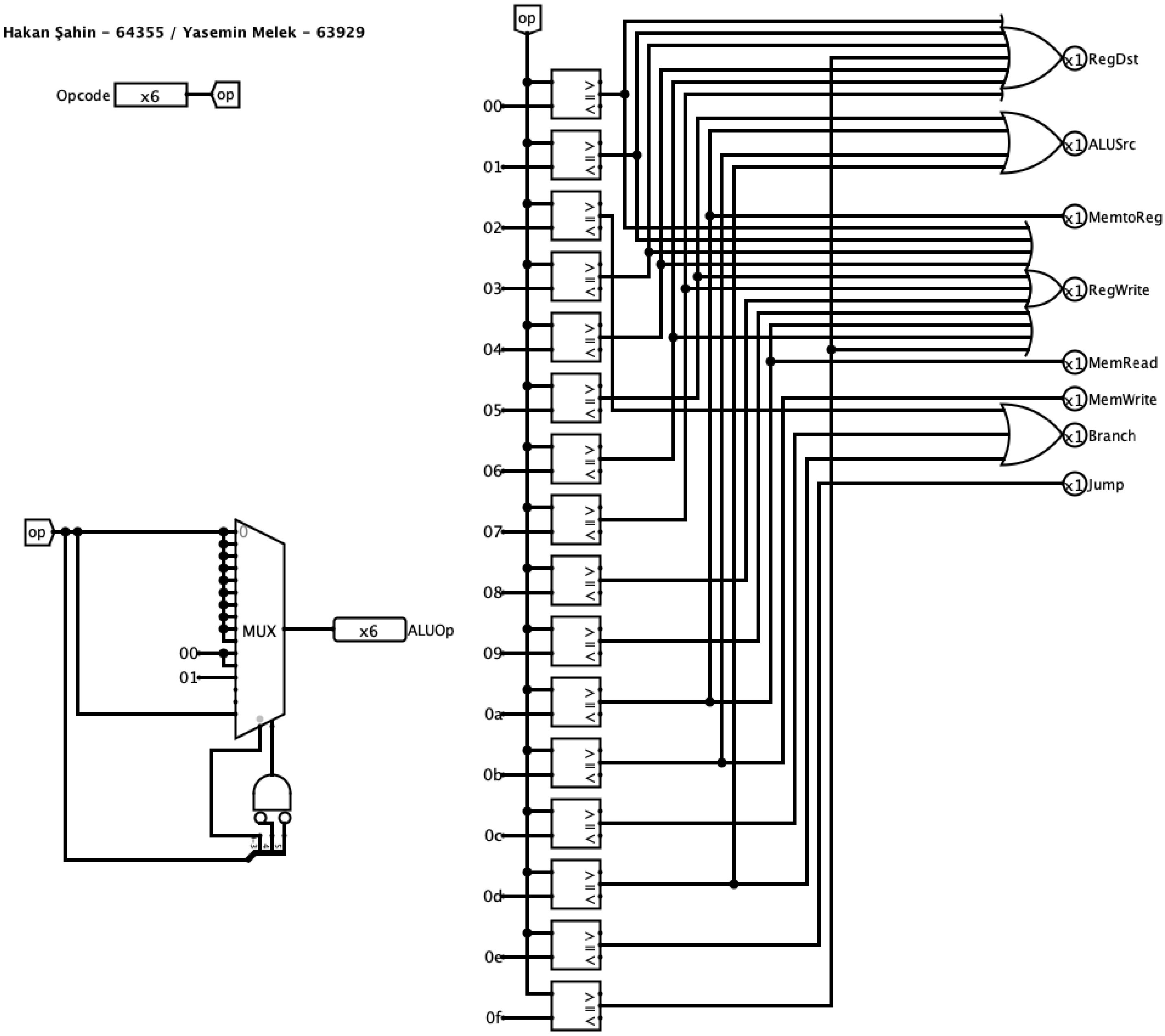


Instruction Splitter

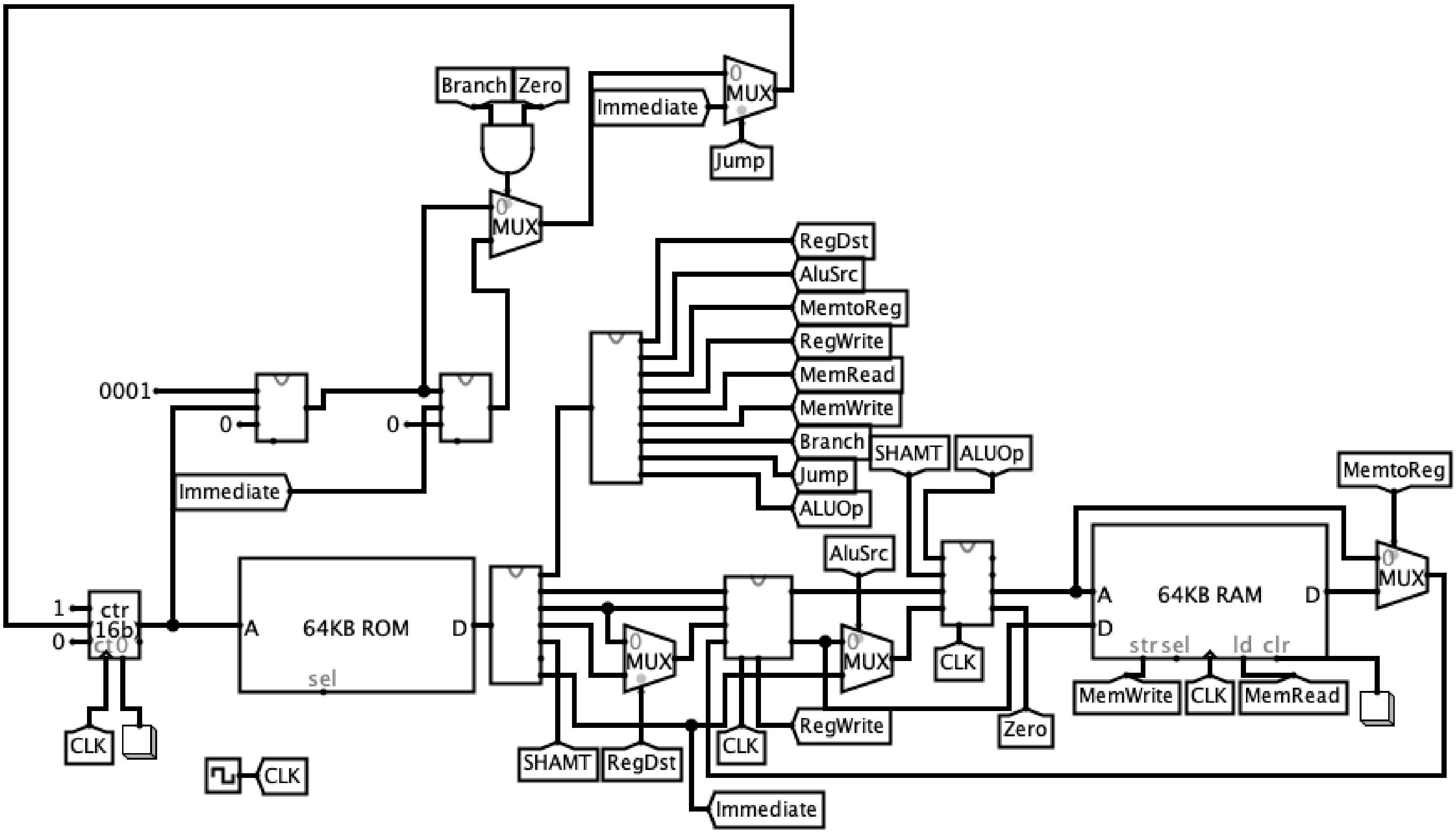


Control Unit

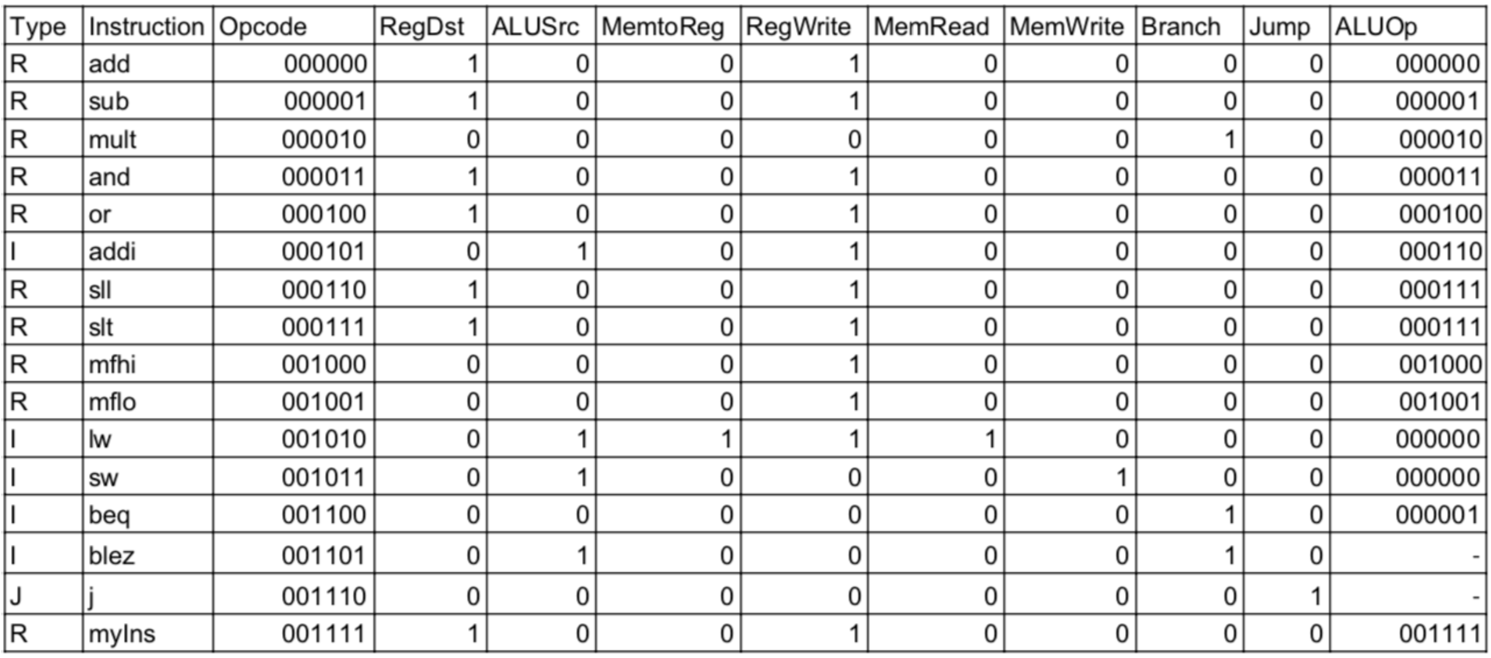
Hakan Şahin - 64355 / Yasemin Melek - 63929



Single Cycle Processor



Control Table



Type	Instruction	Opcode	RegDst	ALUSrc	MemtoReg	RegWrite	MemRead	MemWrite	Branch	Jump	ALUOp
R	add	000000	1	0	0	1	0	0	0	0	000000
R	sub	000001	1	0	0	1	0	0	0	0	000001
R	mult	000010	0	0	0	0	0	0	1	0	000010
R	and	000011	1	0	0	1	0	0	0	0	000011
R	or	000100	1	0	0	1	0	0	0	0	000100
I	addi	000101	0	1	0	1	0	0	0	0	000110
R	sll	000110	1	0	0	1	0	0	0	0	000111
R	slt	000111	1	0	0	1	0	0	0	0	000111
R	mfhi	001000	0	0	0	1	0	0	0	0	001000
R	mflo	001001	0	0	0	1	0	0	0	0	001001
I	lw	001010	0	1	1	1	1	0	0	0	000000
I	sw	001011	0	1	0	0	0	1	0	0	000000
I	beq	001100	0	0	0	0	0	0	1	0	000001
I	blez	001101	0	1	0	0	0	0	1	0	-
J	j	001110	0	0	0	0	0	0	0	1	-
R	myIns	001111	1	0	0	1	0	0	0	0	001111

Our Design and Custom Instruction

In our single cycle processor design, we mostly used the examples and information from the lecture slides. The design works for the instructions in the table above. We have used an instruction splitter unit for separating the opcode, rs,rt,rd, shamt, funct and Immediate. For the arithmetic instructions in the ALU except the adder we used the built-in arithmetic library in logisim.

In the selection process of our control unit we used the opcode. Based on the opcode of the instruction the necessary signals output. If the instruction is an R-type instruction, the ALUOp is the same as the opcode of the instruction. If the instruction is I-type, ALUOp becomes 000000 or 000001 depending on the instruction.

Our custom instruction is an R-type instruction. For two inputs A and B, the instruction does the remainder operation which is A%B. Therefore the instruction performs rd = rs % rt. For this operation, we used the division operation from the built-in arithmethic library in logisim.