



Project Topic:

Design a 12-bit Custom RISC-Microprocessor

Project Members:

1. Md. Samin Israk 2021362642
2. Shahriyar Zaman Ridoy 2022299642
3. Jannat Sultana 2022217642
4. Zinnat Fowzia Ria 1931343042



There are mainly two parts in our project —

1. Logisim Simulation
2. Assembler (Coding)



Logisim Simulation:

For the logisim simulation, we have built 12-bit register, 12-bit ALU and a Control unit. And then we connected register file, ALU and Control unit in a single file, also we added a 12-bit ROM and a 12-bit RAM in that file. In the ROM we will write the opcode of all the instructions.



Assembler (coding)

By using C++ language , we wrote code for different instructions. Firstly, we converted Assembler to Binary. Then we converted Binary to Hexadecimal. We used input file to take the input and output file to show the outputs.



Q: How many types of instruction?

Answer: There are 3 types of instructions. R-type, I-type(Immediate ,Load, Store) , J-type

Q: Describe each of the formats.

Answer:

R -Type formats and field length

OP	RS	RT	RD	Shamt	Funct
3	3	3	3	0	0



I- Type:

OP	RS	RT	Immediate
3	3	3	3

J-Type:

OP	Address
3	9



Q: How many Operands?

Answer:

R-Type : 3 operands

I-Type : 3 operands

J-type : 1 operand



Q: How many operations?

Answer:

R-type: 2 - Arithmetic operations(Add, Sub)

I-type: 4 - Arithmetic operations(Addi, Subi, LW,SW)

1- logical operations (ANDi)

J-type: 1-operations

Information of All Instructions

Instructions	Opcode	RD/RT (Reg.Dst)	Reg. Write.En	ALU Src	AluOP 1	AluOP 0	C.in	B.Invert	lw_enb	sw_enb	RAM to Reg	Jump
Add	000	0	1	0	1	0	0	0	0	0	0	0
Sub	001	0	1	0	1	0	1	1	0	0	0	0
Lw	010	1	1	1	1	0	0	0	1	0	1	0
Sw	011	0	0	1	1	0	0	0	0	1	0	0
Addi	100	1	1	1	1	0	0	0	0	0	0	0
Subi	101	1	1	1	1	0	1	1	0	0	0	0
Andi	110	1	1	1	0	0	0	0	0	0	0	0
Jump	111	0	0	0	0	0	0	0	0	0	0	1

R- Format

Opcode (3 bits)

RS (3 bits)

RT (3 bits)

RD (3 bits)

I- Format

Opcode (3 bits)

RS (3 bits)

RT (3 bits)

Immediate (3 bits)

J- Format

Opcode (3 bits)

Target (9 bits)

