Project: Design of 24-bit MIPS CPU

Course Code: CSE332
Course Title: Computer Organization and Architecture
Semester: Summer 2024
Section: 04
Group: 10

Group Members:

Samir Mahmud	***2212429042***
Mohaimen Al Mamun	***2221726642***

Operands

In our project there are different operands like:

Rs- Source register 1.
Rt- Source register 2.
Rd- Destination register.
shampt- Shift amount.
Function bits- Identifies operation in r-type instructions.
Immediate- Used for constant values in I-type

Types of operands

The operand types in our instruction set are:

- **Register-based**: Most of the instructions work with registers (rs, rt, rd).
- **Memory-based**: Only LW and SW involve memory access.

Number of operations

Our project is assigned with 15 operations in total. Those are:

1. ADDi
2. NOP
3. NOR
4. BEQ
5. OR
6. ADD
7. BNE
8. SRL
9. SW
10. SLT
11. SLL
12. SUB
13. JMP
14. AND
15. LW

Types of operations

Name	Category	Operation	Type/Format
ADDi	Arithmetic	Add immediate	I
NOP	No Operation	No Operation	R
NOR	Logical	NOR	R
BEQ	Conditional	Branch On Equal	I
OR	Logical	Or	R
ADD	Arithmetic	Add	R
BNE	Conditional	Branch on Not Equal	I
SRL	Shift	Shift Right Logical	R
SW	Memory	Store Word	I
SLT	Comparison	Shift Less Than	R
SLL	Shift	Shift Left Logical	R
SUB	Arithmetic	Subtract	R
JMP	Unconditional	Jump	J
AND	Logical	And	R
LW	Memory	Load Word	I

No. of the format of instructions

There are 3 different instruction formats used in our design:

- **R-type format:**

    1. ADD
    2. SUB
    3. AND
    4. OR
    5. NOR
    6. SLT
    7. SLL
    8. SRL.
    9. NOP

- **I-type format:**

    1. ADDi
    2. LW
    3. SW
    4. BEQ
    5. BNE.

- **J-type format:**

    1. JMP.

Description of each of the formats

3 formats with fields and field length shown below-

R-Type:

Opcode	Rs	Rt	Rd	Shampt	Function
4	5	5	5	1	4

I-type:

Opcode	Rs	Rt	Immediate
4	5	5	10

J-type:

Opcode	Address
4	20

Control Signals Table

Instruction	Opcode (4 bits)	ALUOp (2 bits)	Func (4 bits)	ALUSrc	RegDest	MemRead	MemWrite	MemtoReg	RegWrite	Branch	Jump
ADDi	0001	00	xxxx	1	0	0	0	0	1	0	0
NOP	0000	10	0001	0	0	0	0	0	0	0	0
NOR	0000	10	0010	0	1	0	0	0	1	0	0
BEQ	0010	01	xxxx	0	x	0	0	x	0	1	0
OR	0000	10	0011	0	1	0	0	0	1	0	0
ADD	0000	10	0100	0	1	0	0	0	1	0	0
BNE	0011	01	xxxx	0	0	0	0	x	0	1	0
SRL	0000	10	0101	0	1	0	0	0	1	0	0
SW	0100	00	xxxx	1	x	0	1	x	0	0	0
SLT	0000	10	0110	0	1	0	0	0	1	0	0
SLL	0000	10	0111	0	1	0	0	0	1	0	0
SUB	0000	10	1000	0	1	0	0	0	1	0	0
JMP	0101	xx	xxxx	0	0	0	0	0	0	0	1
AND	0000	10	1001	0	1	0	0	0	1	0	0
LW	0110	00	xxxx	1	0	1	0	1	1	0	0

ALUop and different operations in the ALU

ALUop	Operation
00	Lw, sw, addi
01	Beq, bne
10	R type
xx	jump

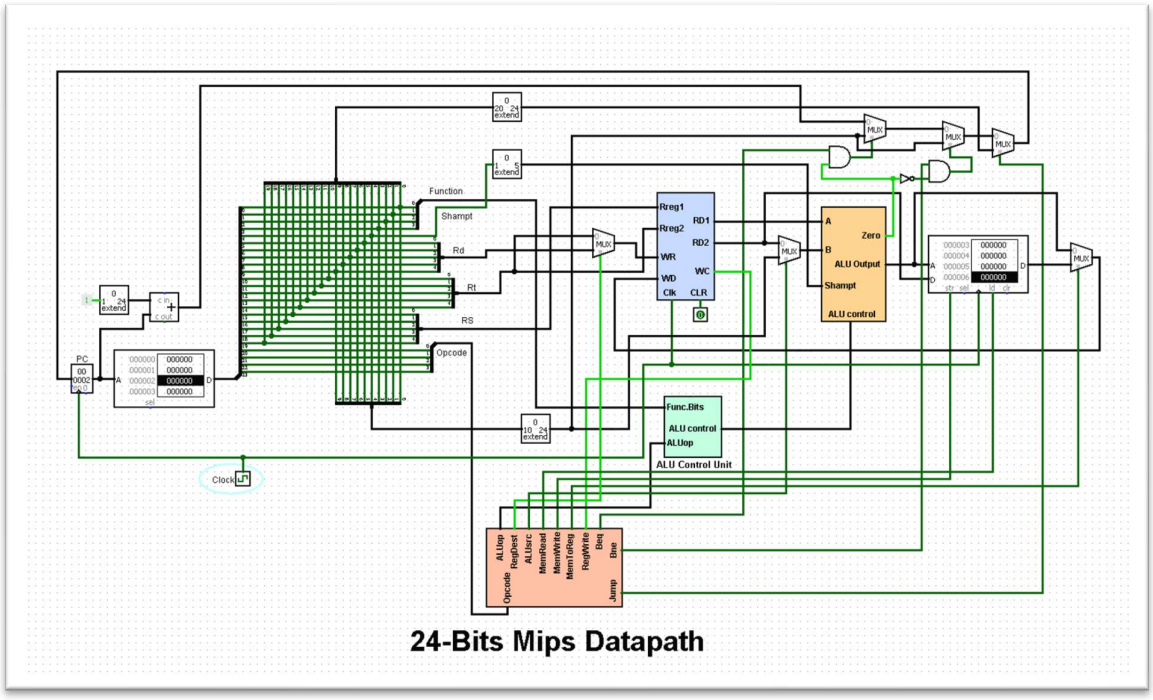
ALU control Unit output(ALU control signals)

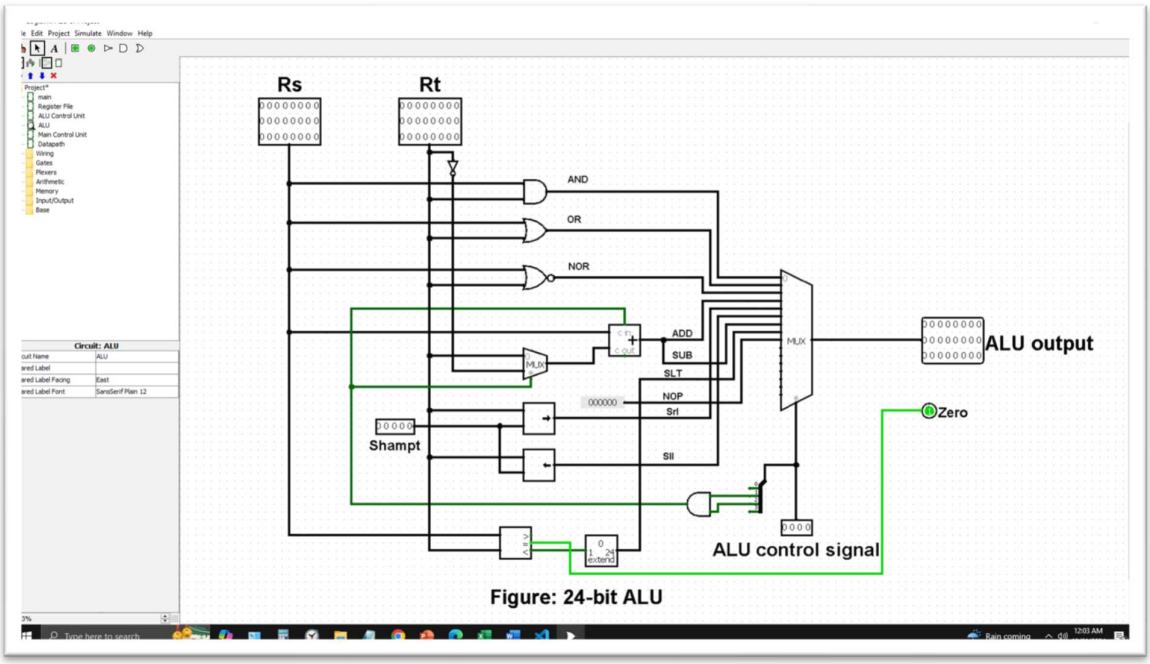
ALU Operation	ALU Control Output (4-bit)
AND	0000
OR	0001
NOR	0010
ADD	0011
SRL (Shift Right)	0100
SLL (Shift Left)	0101
Subtract	0110
SLT (Set Less Than)	0111
NOP	1000

Combined Table for ALU Control Unit

Instructions	Type	ALUOp	Function Field (4 bits)	Operation in ALU	ALU Control Input (4 bits)
ADDi	I-type	00	xxxx	ADD	0011
NOP	R-type	xx	0001	NOP	1000
NOR	R-type	10	0010	NOR	0010
BEQ	I-type	01	xxxx	Subtract	0110
OR	R-type	10	0011	OR	0001
ADD	R-type	10	0100	ADD	0011
BNE	I-type	01	xxxx	Subtract	0110
SRL	R-type	10	0101	Shift Right Logical	0100
SW	I-type	00	xxxx	ADD	0011
SLT	R-type	10	0110	Set Less Than	0111
SLL	R-type	10	0111	Shift Left Logical	0101
SUB	R-type	10	1000	Subtract	0110
JMP	J-type	00	xxxx	Jump	1111
AND	R-type	10	1001	AND	0000
LW	I-type	00	xxxx	ADD	0011

Screenshots of the circuits:





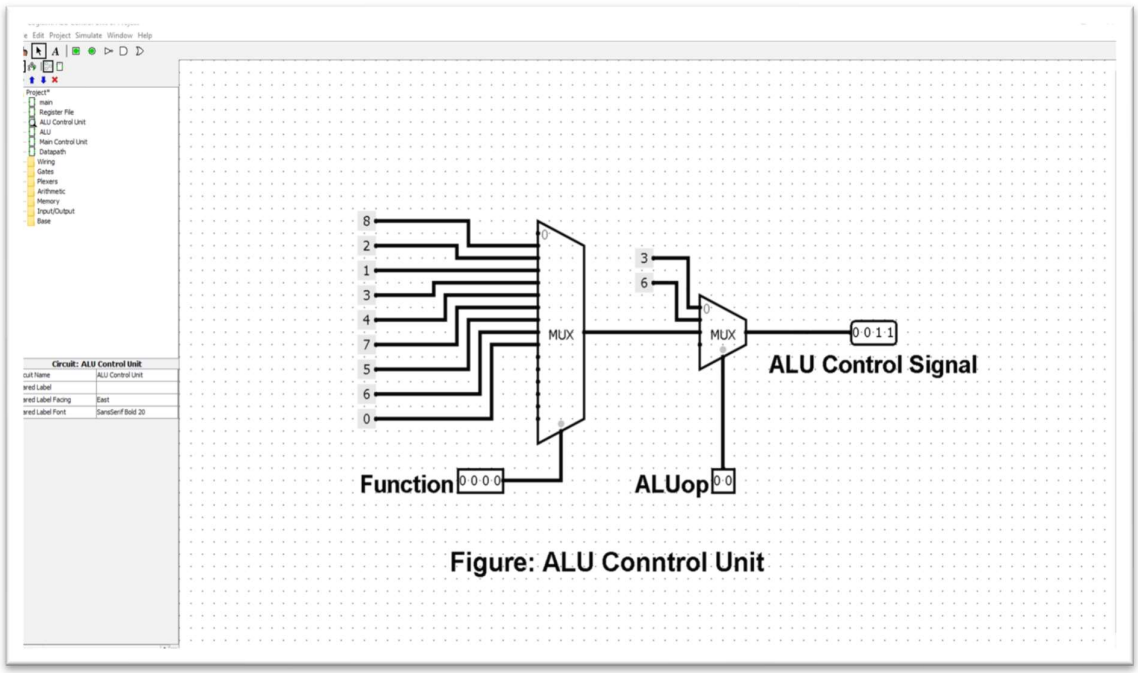


Figure: ALU Conntrol Unit

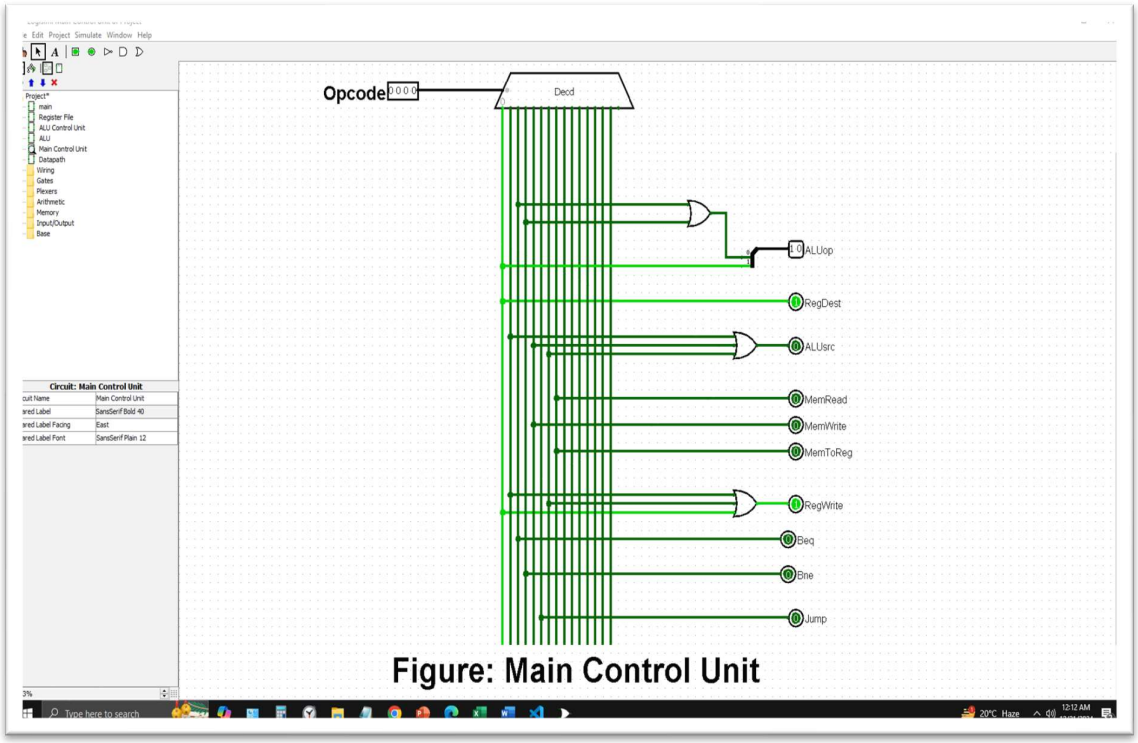


Figure: Main Control Unit

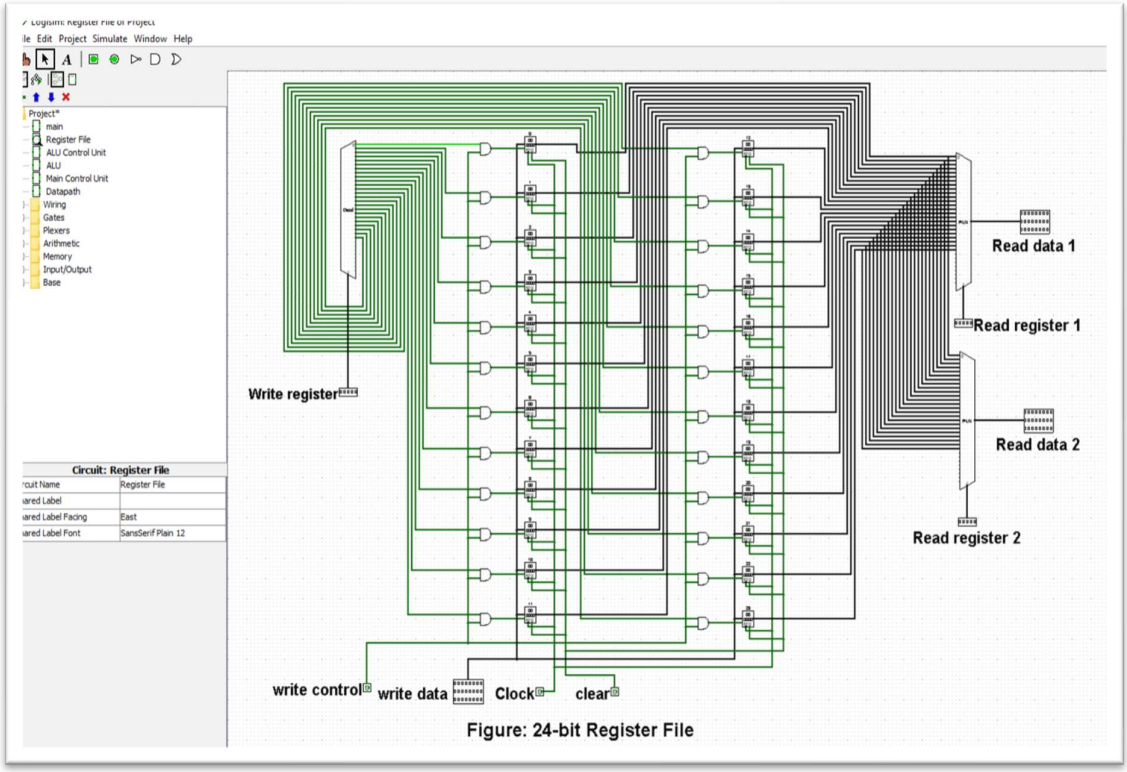


Figure: 24-bit Register File