RISC Design

N Surya Prakash Reddy(20CS10038)

Burra Nithish(20CS10018)

November 9, 2022

1 Instruction Format Encoding

1.1 R-type instructions

- For Arithmetic, Shift, Logic and Complex instructions

OP code	Dest Reg	Src Reg	Shamt	Rednt.	Func Code
6	5	5	5	5	6

Table 1: Encoding for R-type instructions

1. OP Code - Operation code: 6 bits
2. Dest Reg - Destination register: 5 bits
3. Src Reg - Source register: 5 bits
4. Shamt - Shift amount: 5 bits
5. Rednt - Redundant field: 5 bits
6. Func Code - Function code: 6 bits

1.2 Immediate Instructions

OP code	Dest Reg	Src Reg	Imm
6	5	5	16

Table 2: Encoding for Immediate instructions

1. OP Code - Operation code: 6 bits
2. Dest Reg - Destination register: 5 bits
3. Src Reg - Source register: 5 bits
4. Imm - Immediate: 16 bits

1.3 Branch Instructions

OP code	Dest Reg	Addr	Func code
6	5	15	6

Table 3: Encoding for branching instructions

1. OP Code - Operation code: 6 bits
2. Dest Reg - Destination register: 5 bits
3. Addr - Branch address: 15 bits
4. Func Code - Function code: 6 bits

Operation	OP Code	Func code
ADD	1	0
COMP	1	1
AND	2	0
XOR	2	1
SHLL	3	0
SHLLV	3	1
SHRL	3	2
SHRLV	3	3
SHRA	3	4
SHRAV	3	5
BR	4	0
BLTZ	4	1
BZ	4	2
BNZ	4	3
B	5	0
BCY	5	1
BNCY	5	2
BL	6	0
DIFF	7	0
ADDI	8	-
COMPI	9	-
LW	10	-
SW	11	-

Table 4: Encoding for each operation

2 Data path

The data path for the above set of rules would look something like Figure 1

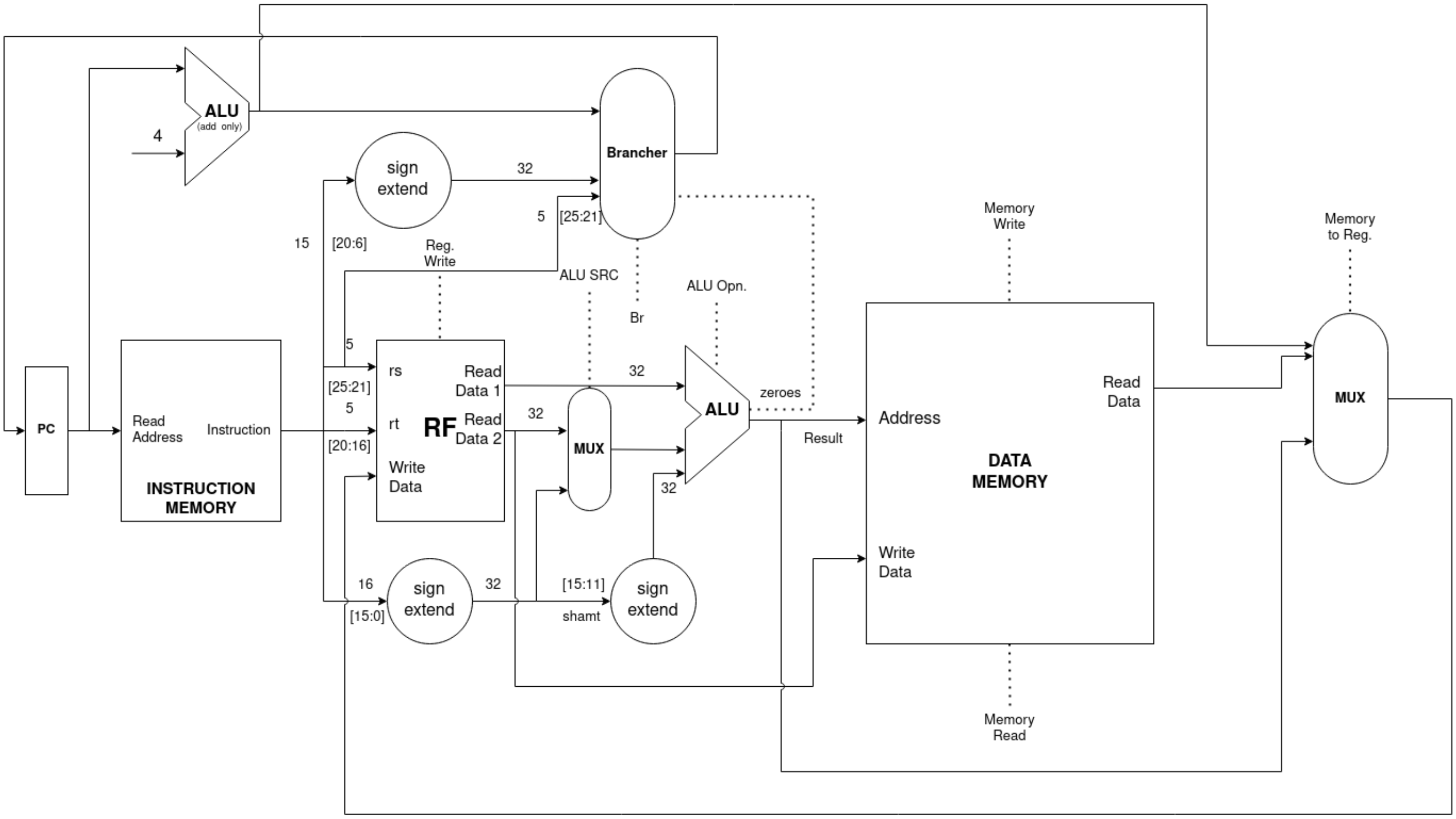

Figure 1: Datapath

3 Control Unit

3.1 Main Control Unit

The main control unit takes the OPCODE as input and outputs the different control line values

instruction type	OP Code	aluOp	aluSrc	memToReg	regWrt	memRd	memWrt	Br
Reg arith	000001	001	1	00	10	0	0	00
Reg logical	000010	010	1	00	10	0	0	00
shift	000011	011	1	00	10	0	0	00
Reg branch	000100	000	1	00	00	0	0	01
Carry branch	000101	000	1	00	00	0	0	10
branch and link	000110	000	1	01	01	0	0	11
diff	000111	100	1	00	10	0	0	00
add imm	001000	101	0	00	10	0	0	00
comp imm	001001	110	0	00	10	0	0	00
load word	001010	101	0	10	11	1	0	00
store word	001011	101	0	00	00	1	1	00

Table 5: Main Control Table

- aluOp - Control signal for alu showing the operation type
- aluSrc - For the second operand to be imm(0) or rt(1)
- memToReg - For the write data to RF, ALU result(0) or PC+4(1) or Read data(2)
- regWrt - For writing data to RF, no write(0) or reg_{31}(1) or rs(2) or rt(3)
- memRd - For reading data from memory. This control acts like the enable pin of the data RAM.
- memWrt - For writing data to memory
- Br - For branching control which gives out PC Src control signal using the zeros[2:0] output from the ALU

2b'00

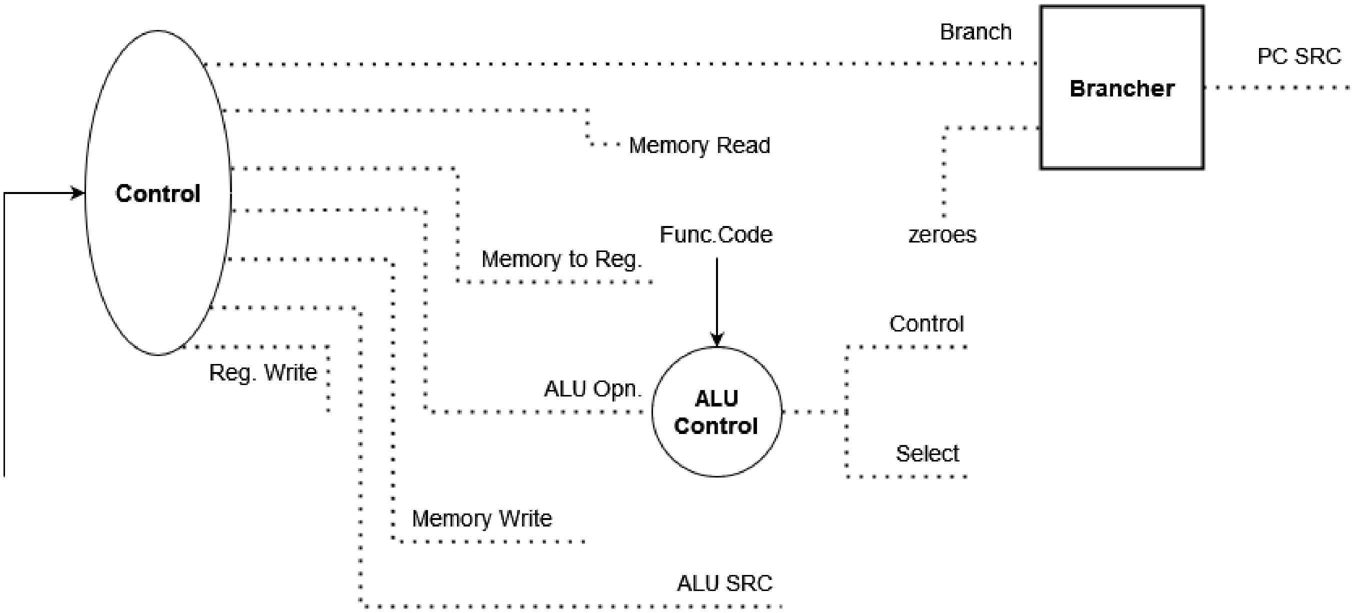

Figure 2: Control

3.2 ALU control unit

The ALU control unit takes the Alu Op as input and sends the control signal to select the required output

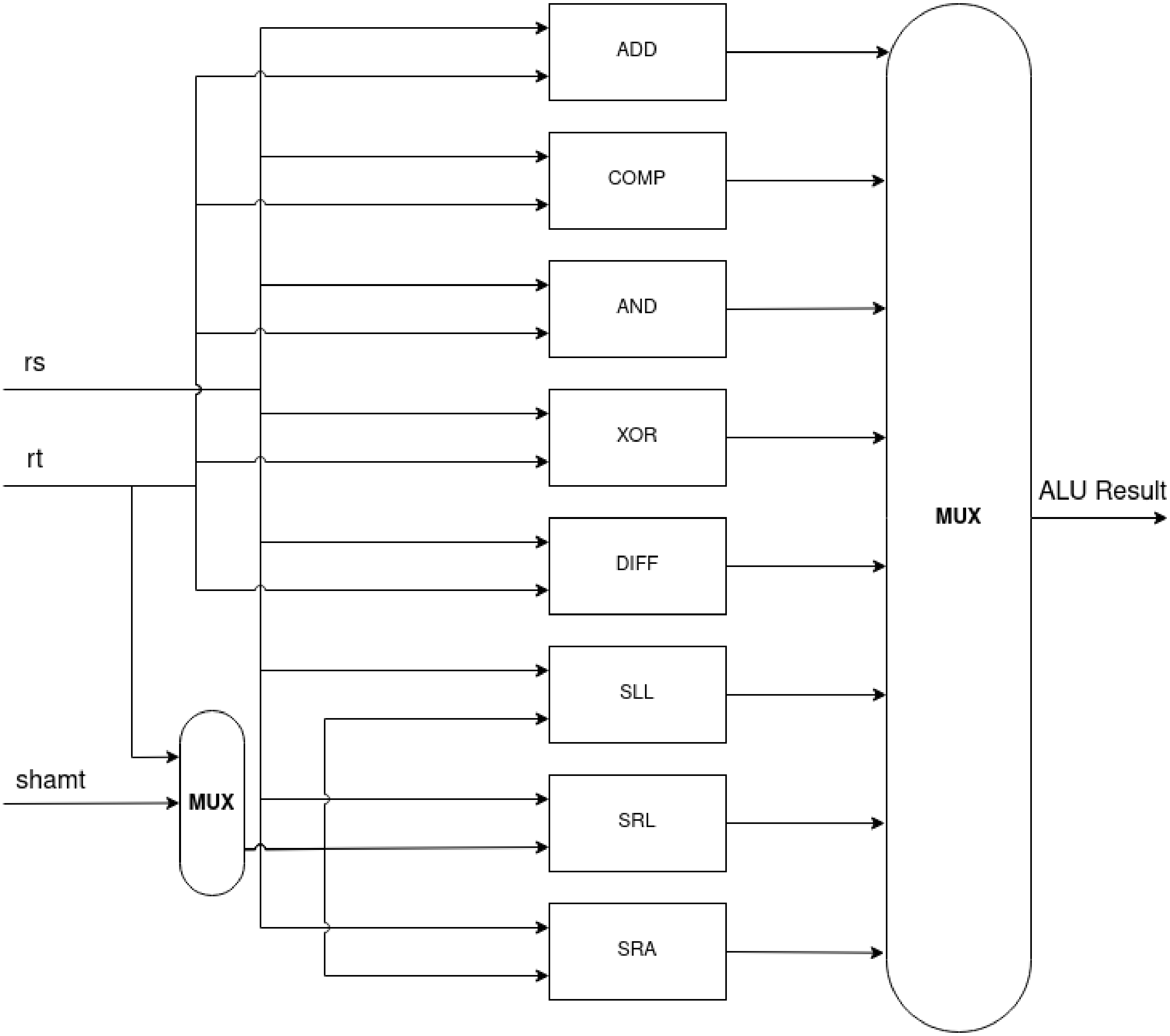

Figure 3: ALU Design

aluOp	fnCode	ctrl	select
000	-	000	0
001	000000	000	0
001	000001	001	0
010	000000	010	0
010	000001	011	0
011	000000	101	1
011	000001	101	0
011	000010	110	1
011	000011	110	0
011	000100	111	1
011	000101	111	0
100	000000	100	0
101	-	000	0
110	-	001	0

Table 6: ALU control unit