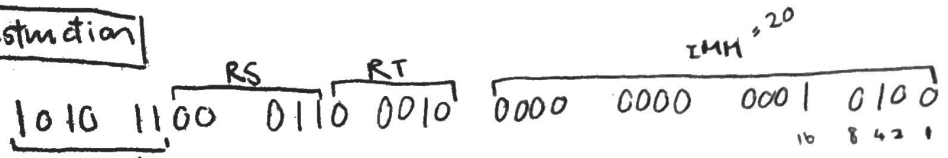
CS M151B HW 4 Junhong Wang (504941113)

4.7

Instruction




Registers

r0 = 0
r1 = -1
r2 = 2
r3 = -3
r4 = -4
r5 = 0
r6 = 6
r8 = 8
r12 = 2
r31 = -16

M[R[RS] + IMM] = R[RT]

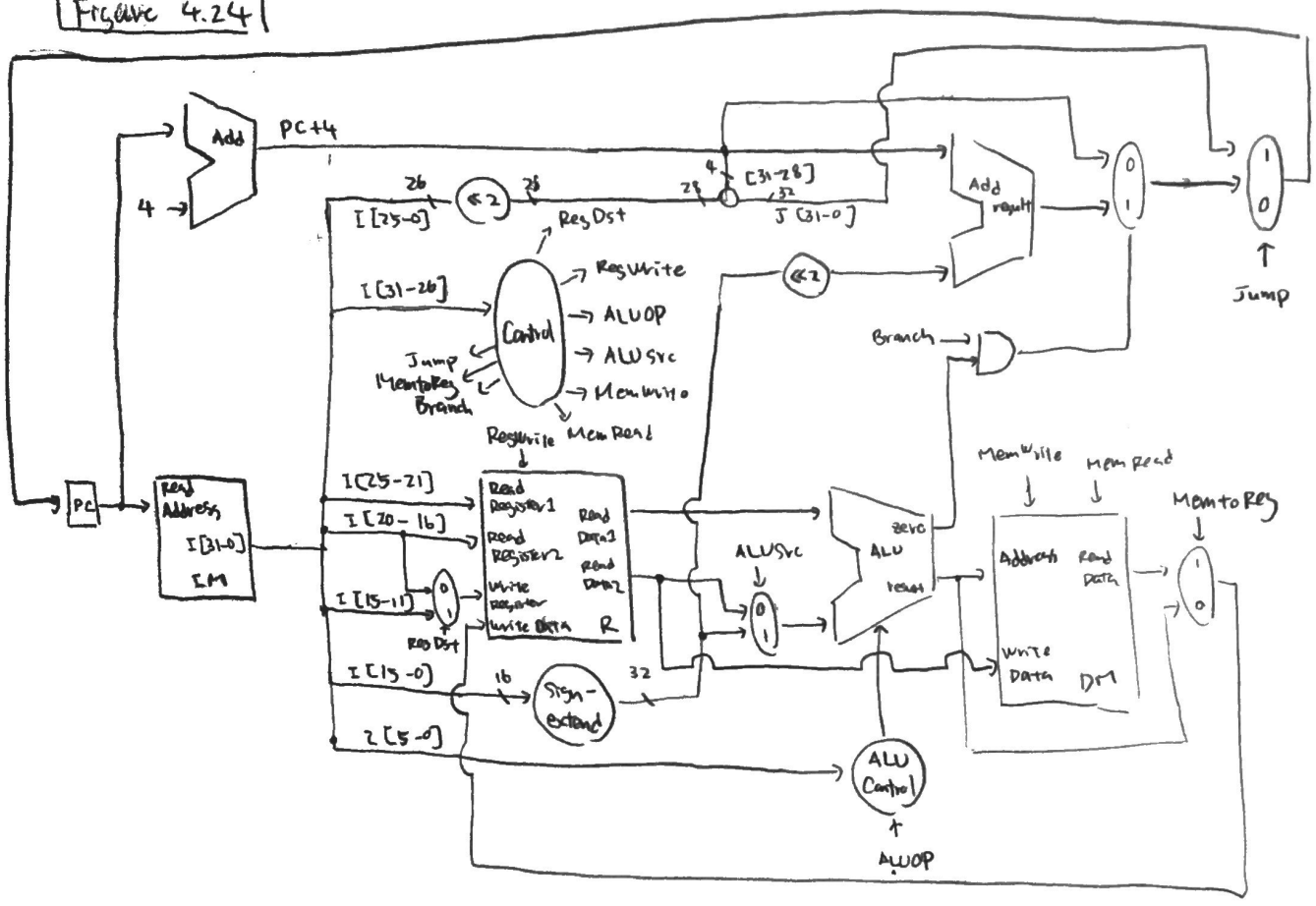
a) M[R[3] + 20] = R[2]

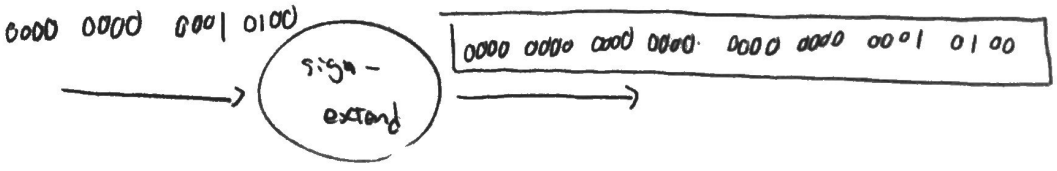
b) M[-3 + 20] = 2

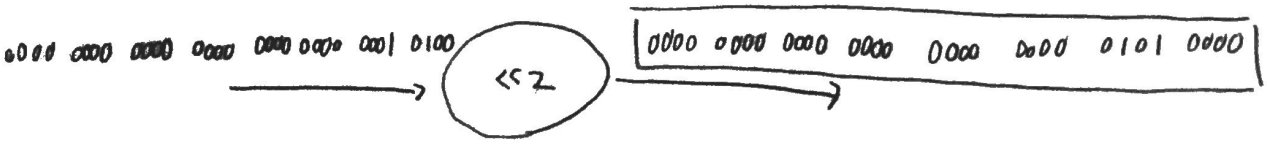
c) M[17] = 2

4.7.1

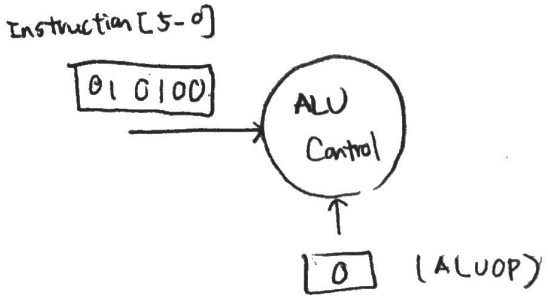
Figure 4.24



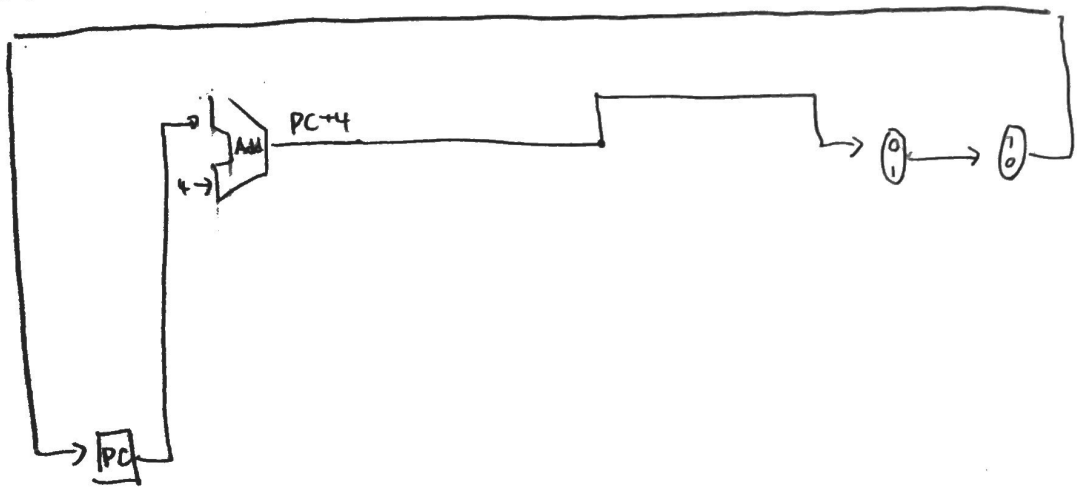



4.7.2

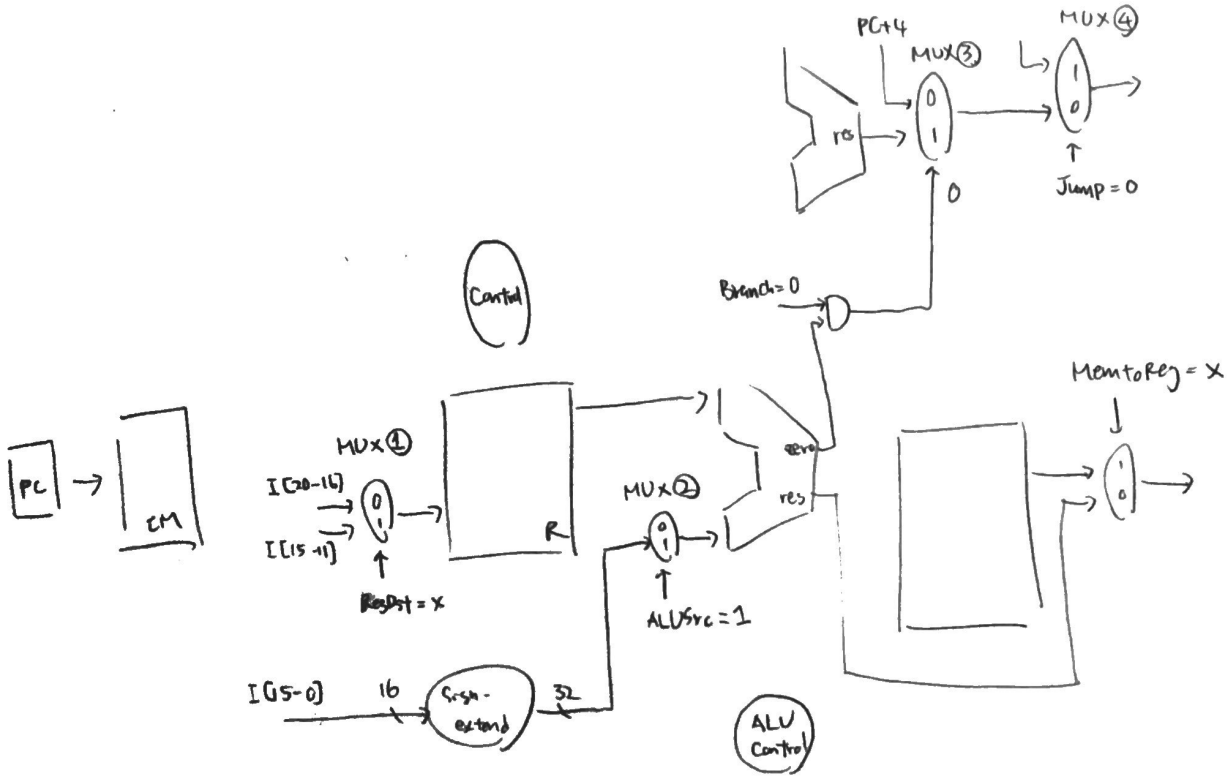


4.7.3




∴ PC ← PC + 4

4.7.4



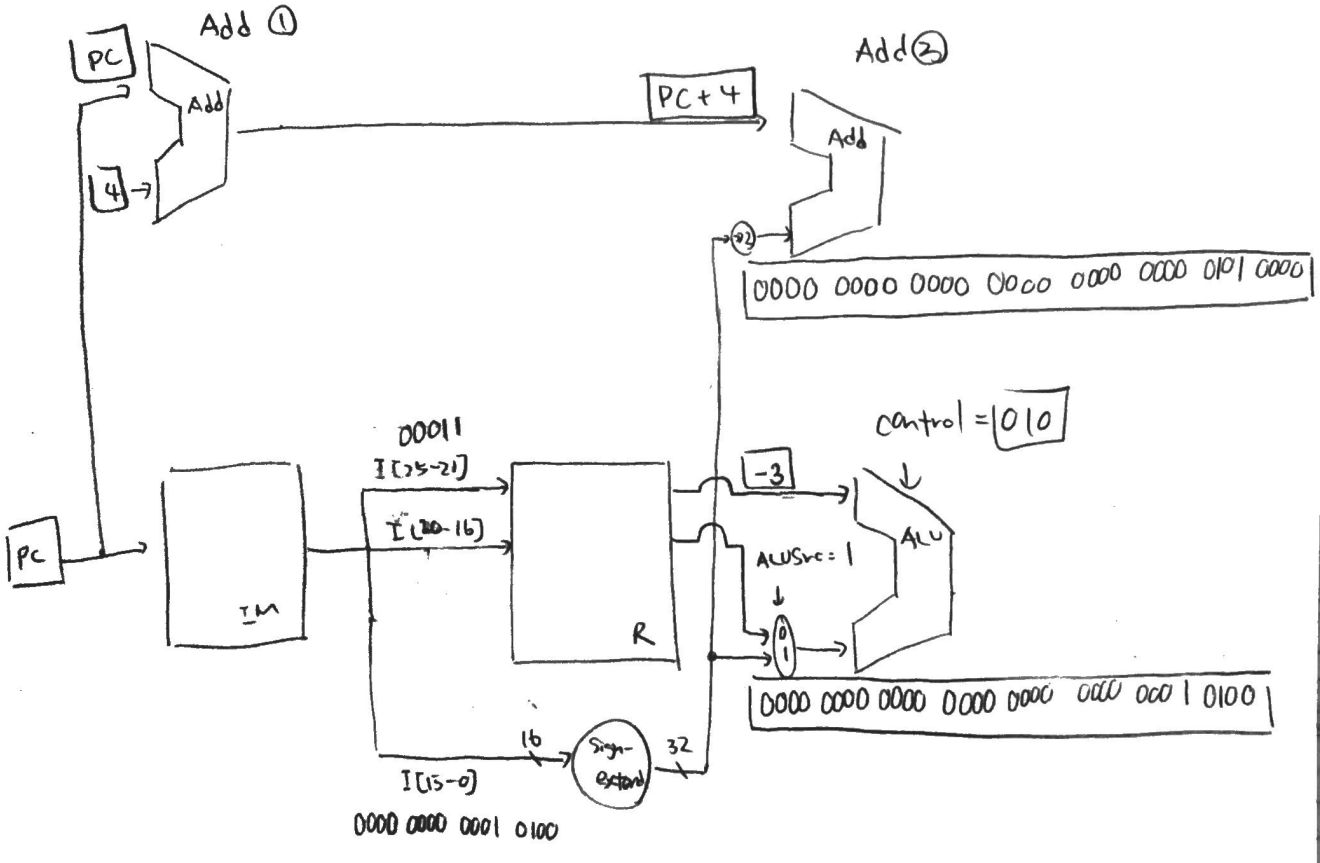

MUX ① : If RegDst = 0, then 00011.
If RegDst = 1, then 00010.

MUX ② : 0000 0000 0000 0000 0000 0000 0001 0100

MUX ③ : PC + 4

MUX ④ : PC+4

4.7.5




Add ① : Top input is PC ; Bottom input is 4.

Add ② : Top input is PC +4 ;

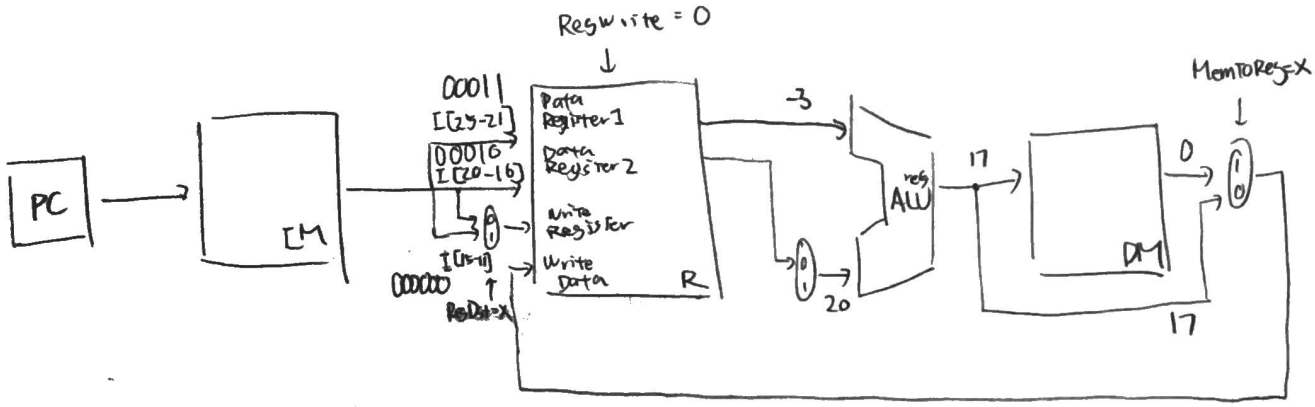
Bottom input is 0000 0000 0000 0000 0000 0000 0101 0000 .

ALU : Top input is -3 (decimal) ;

Bottom input is 0000 0000 0000 0000 0000 0000 0000 0100 ;

Control input is 010 .

4.7.6




Data Register 1 is 00011

Data Register 2 is 00010

If RegDst = 0, then Write Register is 00010

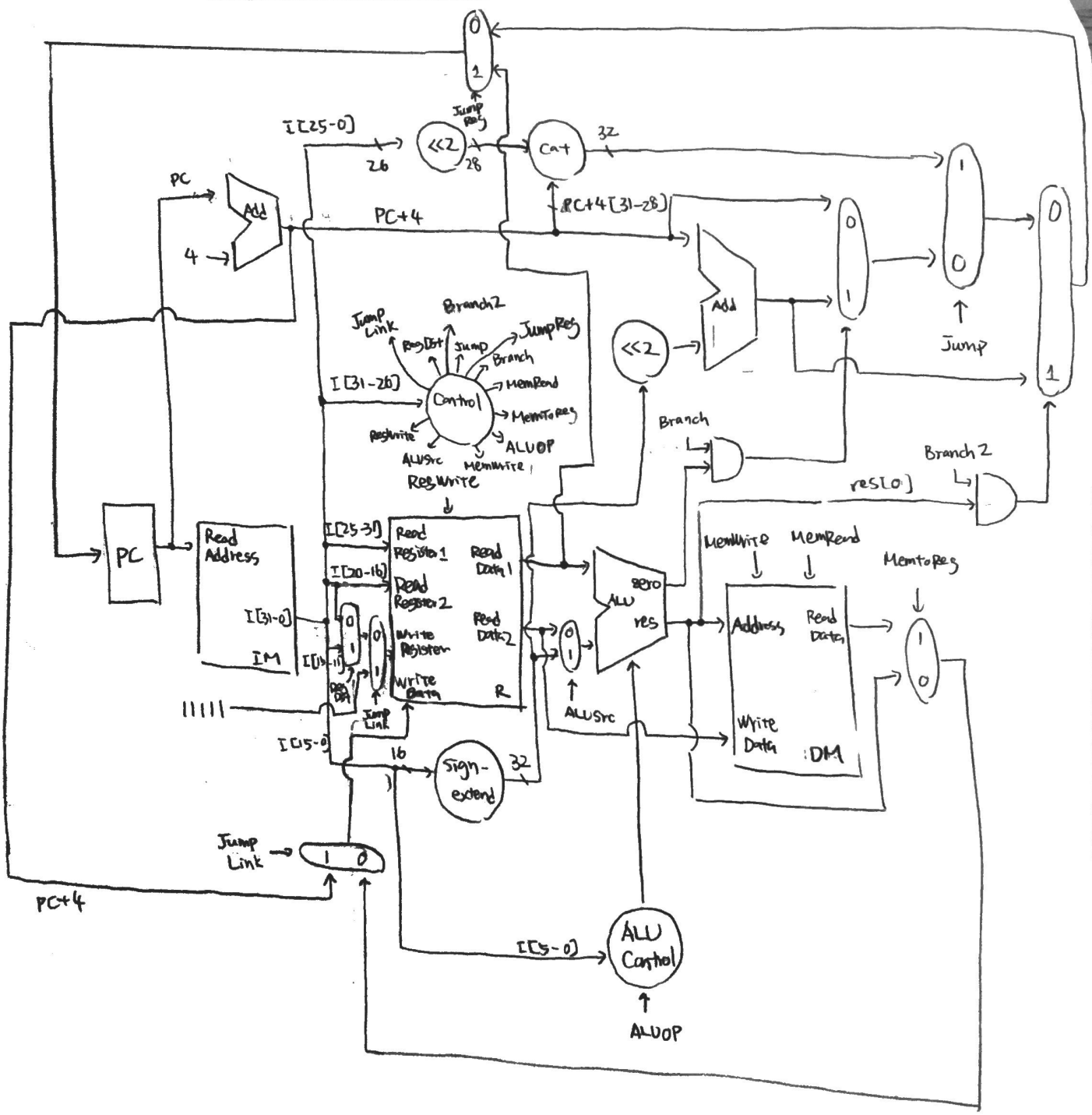
If RegDst = 1, then Write Register is 00000

If Mem to Reg = 0, then Write Data is 17 (decimal)

If Mem to Reg = 1, then Write Data is 0 (decimal).

RegWrite is 0.

# Single Cycle Datapath with blt, jal, jr instructions



Main Controller

| Input or Output | Signal Name | R-format | lw | sw | j | beq | blt | jal | jr |
|---|---|---|---|---|---|---|---|---|---|
| Input | OP code | 000000 | 100011 | 101011 | 000010 | 000100 | 000001 | 000011 | 000000 |
| Outputs | RegDst | 1 | 0 | X | X | X | X | X | X |
| | ALUSrc | 0 | 1 | 1 | X | 0 | 0 | X | X |
| | MemtoReg | 0 | 1 | X | X | X | X | X | X |
| | RegWrite | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 0 |
| | MemRead | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| | MemWrite | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| | Branch | 0 | 0 | 0 | X | 1 | 0 | X | X |
| | Jump | 0 | 0 | 0 | 1 | 0 | 0 | 1 | X |
| | ALUOp | 10 | 00 | 00 | XX | 01 | 11 | XX | XX |
| | Branch 2 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | X |
| | JumpLink | 0 | 0 | X | X | X | X | 1 | X |
| | JumpReg | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |

ALU Controller

| Opcode | ALUOp | instruction | function | ALU Action | ALU Ctrl |
|---|---|---|---|---|---|
| lw | 00 | load word | xxxxxx | add | 010 |
| sw | 00 | store word | xxxxxx | add | 010 |
| beq | 01 | branch equal | xxxxxx | subtract | 110 |
| blt | 11 | branch less than | xxxxxx | SLT | 111 |
| R-type | 10 | add | 100000 | add | 010 |
| | | subtract | 100010 | subtract | 110 |
| | | AND | 100100 | AND | 000 |
| | | OR | 100101 | OR | 001 |
| | | SLT | 101010 | SLT | 111 |