

Control

- ❑ Selecting the operations to perform (ALU, read/write, etc.)
- ❑ Controlling the flow of data (multiplexor inputs)
- ❑ Information comes from the 32 bits of the instruction
- ❑ Example:

add \$8, \$17, \$18

Instruction Format:

000000	10001	10010	01000	00000	100000
op	rs	rt	rd	shamt	funct

- ❑ ALU's operation based on instruction type and function code

Control

- ❑ e.g., what should the ALU do with this instruction
- ❑ Example: lw \$1, 100(\$2)

35	2	1	100
----	---	---	-----

op	rs	rt	16 bit offset
----	----	----	---------------

- ❑ ALU control input

000	AND
001	OR
010	add
110	subtract
111	set-on-less-than

- ❑ Multi-level control is common: main control unit generates ALUop (load/store, branch, arithmetic) bits, which are then used as input to the ALU control that generates the actual signals (3 bits shown above) to control the ALU unit

Control

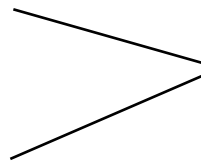
❑ Must describe hardware to compute 3-bit ALU control input

- given instruction type

00 = lw, sw

01 = beq,

10 = arithmetic



ALUOp

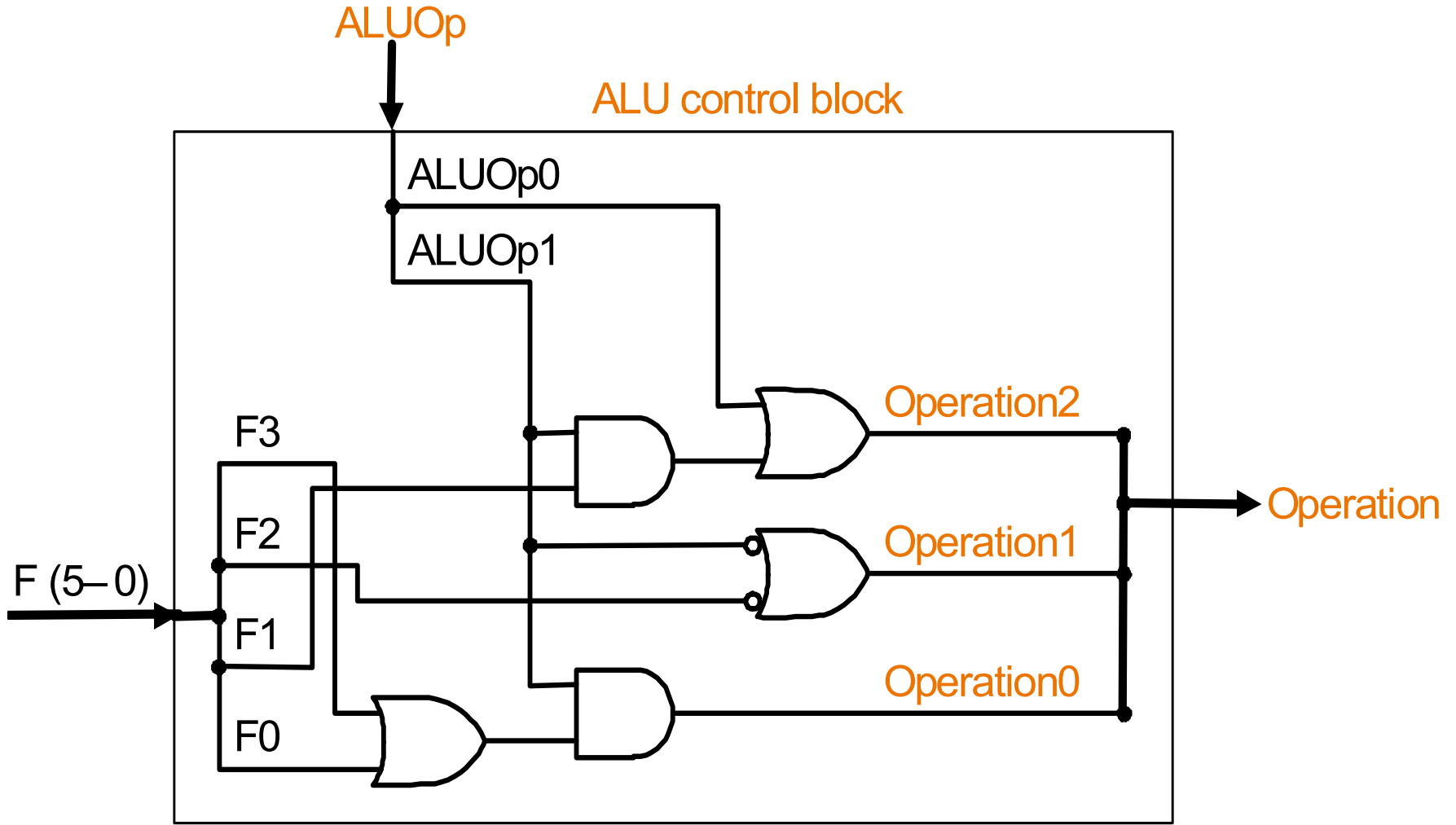
computed from instruction type

- function code for arithmetic

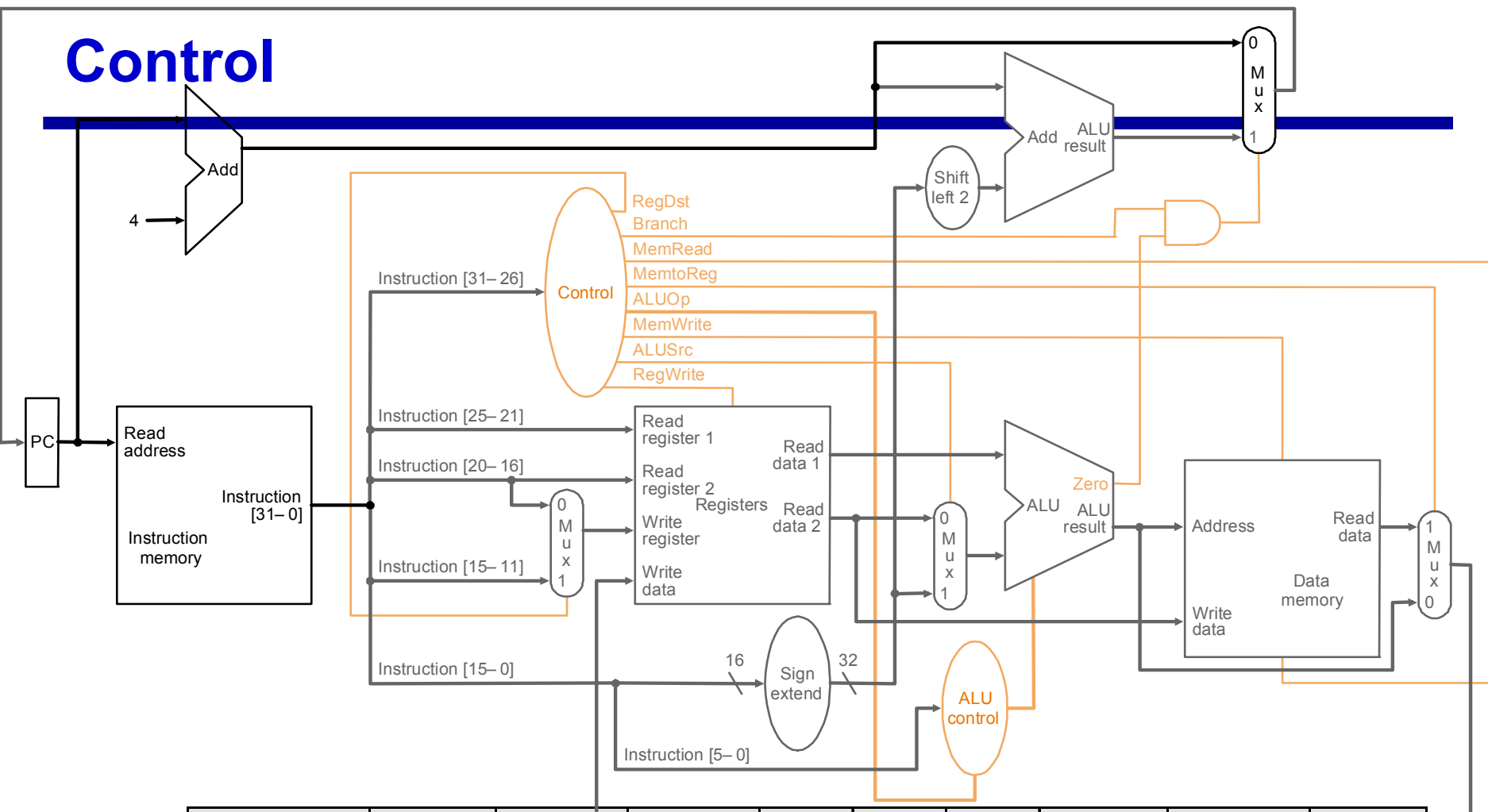
❑ Describe it using a truth table (can turn into gates):

ALUOp		Funct field						Operation
ALUOp1	ALUOp0	F5	F4	F3	F2	F1	F0	
0	0	X	X	X	X	X	X	010
X	1	X	X	X	X	X	X	110
1	X	X	X	0	0	0	0	010
1	X	X	X	0	0	1	0	110
1	X	X	X	0	1	0	0	000
1	X	X	X	0	1	0	1	001
1	X	X	X	1	0	1	0	111

ALU Control Schematic Diagram

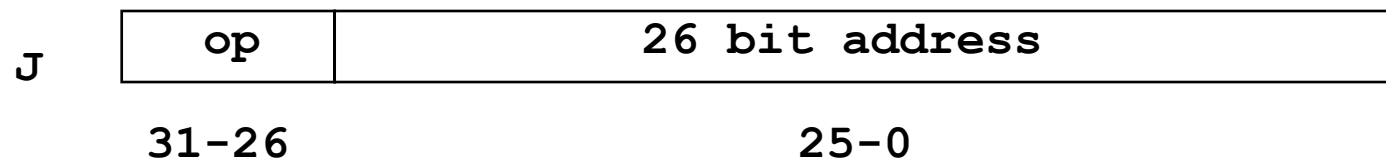
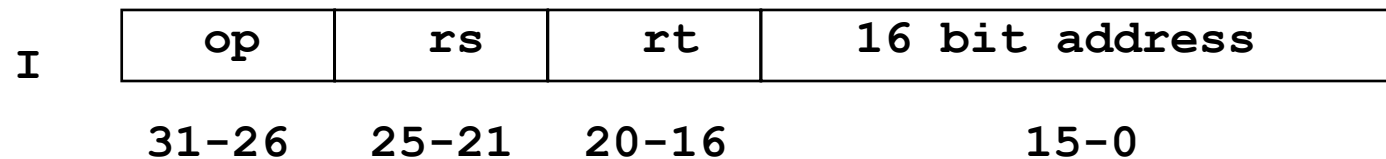
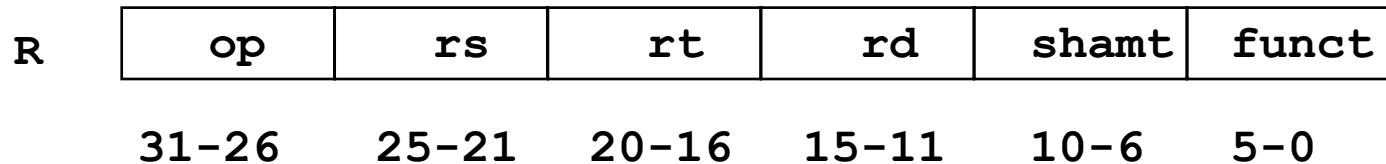


Control



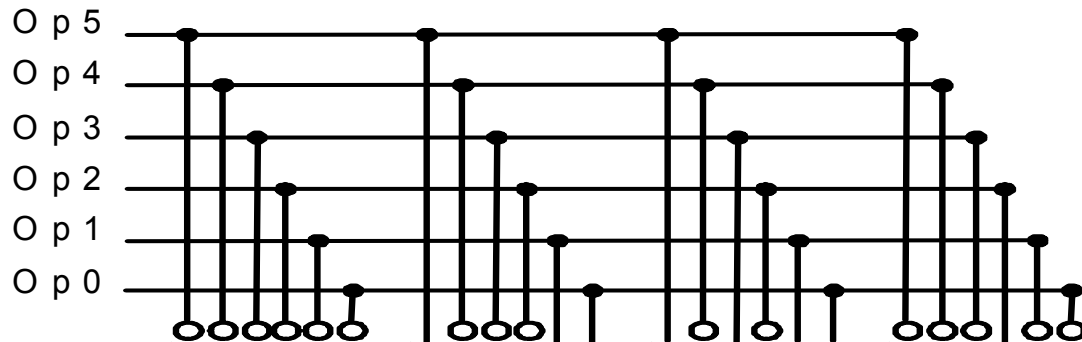
Instruction	RegDst	ALUSrc	Memto-Reg	Reg Write	Mem Read	Mem Write	Branch	ALUOp1	ALUp0
R-format	1	0	0	1	0	0	0	1	0
lw	0	1	1	1	1	0	0	0	0
sw	X	1	X	0	0	1	0	0	0
beq	X	0	X	0	0	0	1	0	1

Recall Instruction Format



Main Control Schematic Diagram

Inputs



R-format

lw

sw

beq

Outputs

- RegDst
- ALUSrc
- MemtoReg
- RegWrite
- MemRead
- MemWrite
- Branch
- ALUOp1
- ALUOp0