

COMPUTER ORGANIZATION AND DESIGN

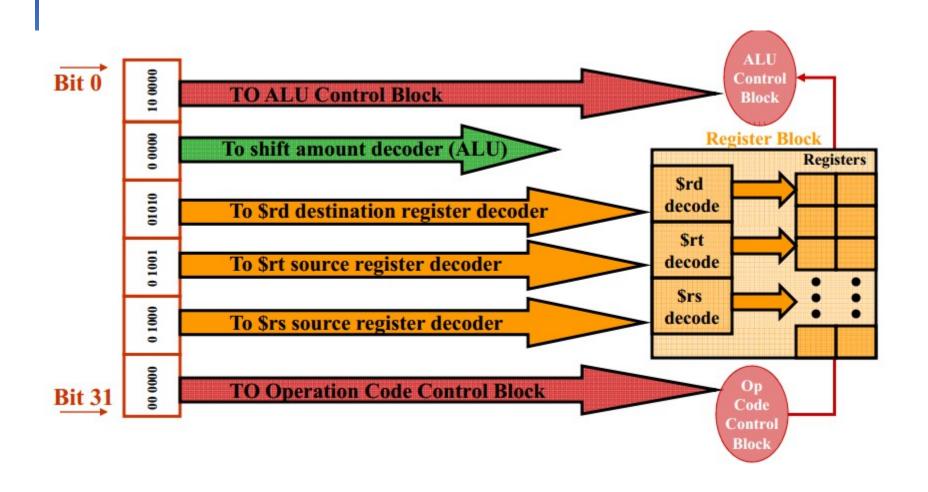


The Hardware/Software Interface

Chapter 4

The Processor

Instruction disposition







Control unit design

- The control unit must be able to take inputs and generate
 - A write signal for each state element,
 - Selector control for each multiplexor and
 - The ALU control.



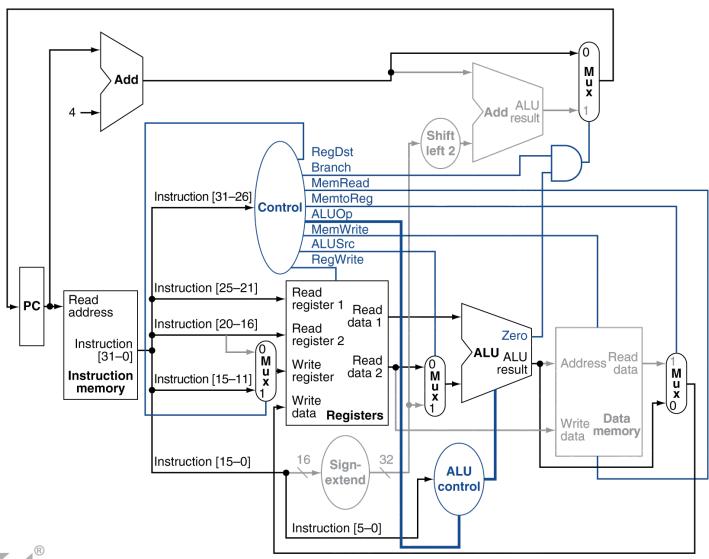
ALU Control

- ALU used for
 - Load/Store: F = add
 - Branch: F = subtract
 - R-type: F depends on funct field

ALU control	Function
0000	AND
0001	OR
0010	add
0110	subtract
0111	set-on-less-than



R-Type Instruction





ALU Control Unit

- Must describe hardware to compute 4-bit ALU control input
 - given instruction type

function code for arithmetic



ALU Control

- Assume 2-bit ALUOp derived from opcode
 - Combinational logic derives ALU control

Opcode	ALUOp	Operation	funct	ALU function	ALU control
lw	00	load word	XXXXXX	add	0010
SW	00	store word	XXXXXX	add	0010
Beq	01	branch equal	XXXXXX	subtract	0110
R-type	10	add	100000	add	0010
		subtract	100010	subtract	0110
		AND	100100	AND	0000
		OR	100101	OR	0001
		set-on-less-than	101010	set-on-less-than	0111



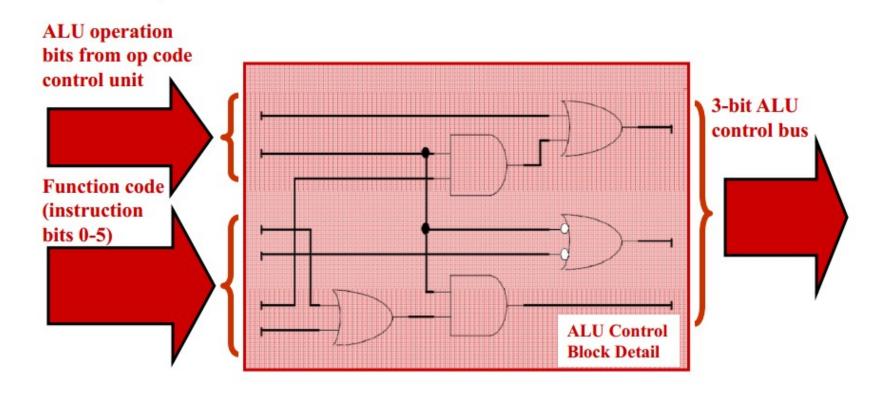
Truth table for ALU control bits

ALUOp		Funct field						
ALUOp1	ALUOp0	F5	F4	F3	F2	F1	F0	Operation
0	0	X	X	Х	Х	X	Х	0010
Х	1	Х	X	Х	Х	X	Х	0110
1	X	Х	X	0	0	0	0	0010
1	Х	Х	X	0	0	1	0	0110
1	X	Х	X	0	1	0	0	0000
1	Х	X	Х	0	1	0	1	0001
1	X	Х	Х	1	0	1	0	0111



Sample ckt for a 3 bit ALU cntrl

ALU Control Block





The Main Control Unit

Control signals derived from instruction

R-type	0	rs	rt	r	rd	shar	mt	funct
	31:26	25:21	20:16	15	5:11	10:	6	5:0
Load/ Store	35 or 43	rs	rt	address				
Otoro	31:26	25:21	20:16		15:0			
Branch	4	rs	rt		address			
	31:26	25:21	20:16		15:0			↑
				/	11			
	opcode	always	read,		write	e for		sign-extend
		read	except			ype		and add
			for load		and	load		



Main control unit

"RegDst" – Register destination control (MUX input)

"Branch" – Activates branch address change function

"MemRead" – Signals read cycle to data memory circuits

"MemtoReg" – Selects ALU or memory write to register

"ALUOp" (2 lines) – Used with function code in ALU control

"MemWrite" – Signals write cycle to data memory circuits

"ALUSrc" – Selects register and immediate ALU operand

"RegWrite" – Activates write function to register block



Effect of control signals

Signal Name	When Signal = 1	When Signal = 0
RegDst	Write reg. = \$rd (bits 11–15)	Write reg. = \$rt (bits 16-20)
Branch	ALU branch compare activated	No branch activated
MemRead	Memory data → write register	No data read from memory
MemtoReg	Memory data → write register	ALU results → write register
ALUOp	NA; lines go to ALU control block	NA; lines go to ALU control block
MemWrite	ALU or register data → memory	No data written to memory
ALUSrc	2nd ALU operand is immediate (sign-extended instr. bits 0-15)	2 nd ALU operand is from \$rt (instruction bits 16-20)
RegWrite	Memory/ALU data → write reg.	No input to register block



Setting of control signals through opcode fields

Instruction	RegDst	ALUSrc	Memto- Reg		Mem Read		Branch	ALUOp1	ALUOp0
R-format	1	0	0	1	0	0	0	1	0
1w	0	1	1	1	1	0	0	0	0
SW	χ	1	Х	0	0	1	0	0	0
beq	X	0	Х	0	0	0	1	0	1

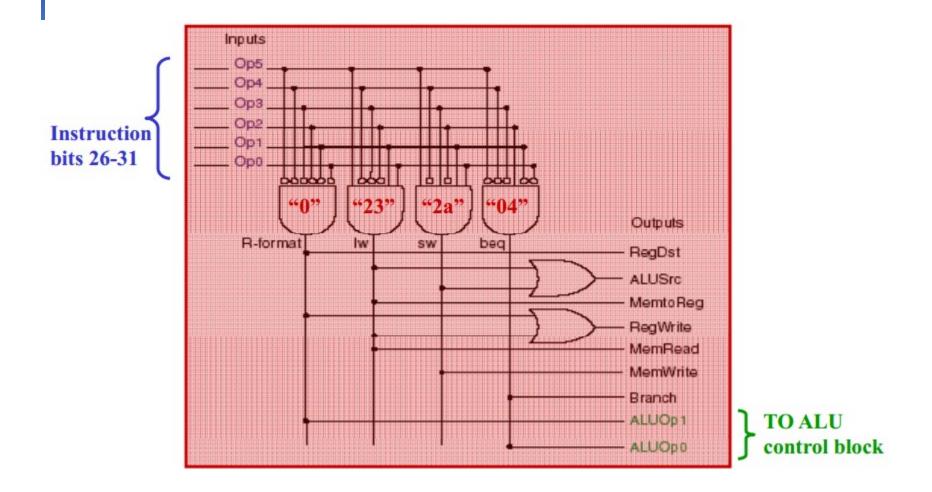


Truth table for control functions

Input or output	Signal name	R-format	1w	SW	beq
Inputs	Op5	0	1	1	0
	Op4	0	0	0	0
	0р3	0	0	1	0
	Op2	0	0	0	1
	Op1	0	1	1	0
	Op0	0	1	1	0
Outputs	RegDst	1	0	Χ	Х
	ALUSrc	0	1	1	0
	MemtoReg	0	1	Χ	Χ
	RegWrite	1	1	0	0
	MemRead	0	1	0	0
	MemWrite	0	0	1	0
	Branch	0	0	0	1
	ALUOp1	1	0	0	0
	ALUOp0	0	0	0	1

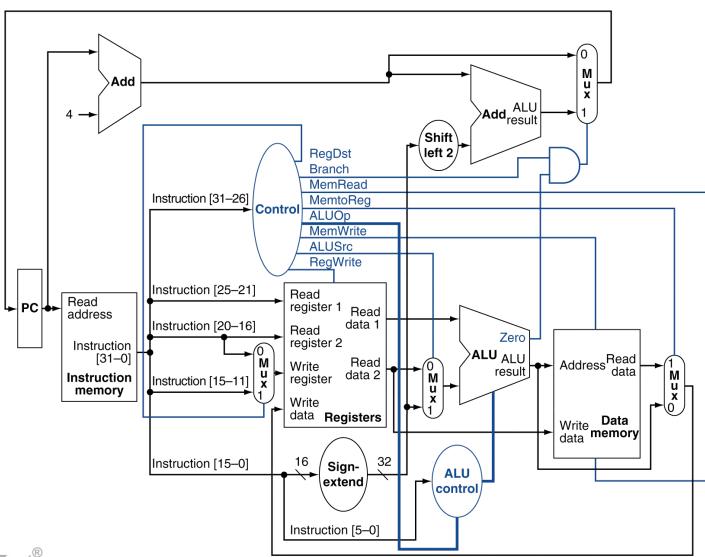


Main control unit



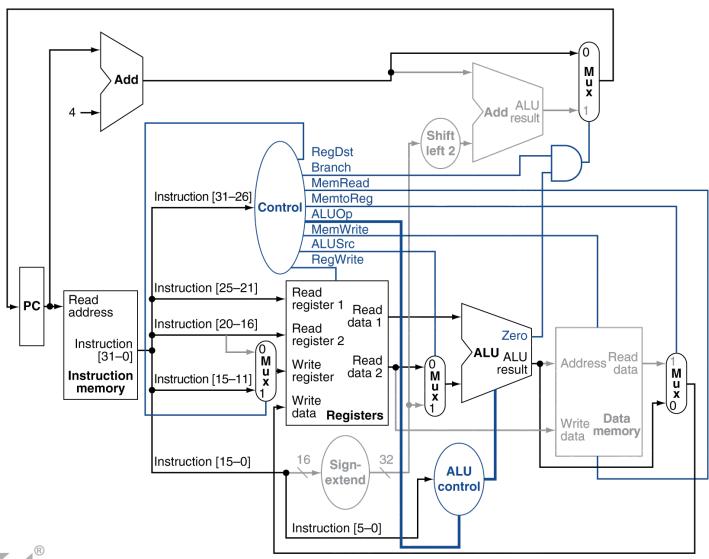


Datapath With Control





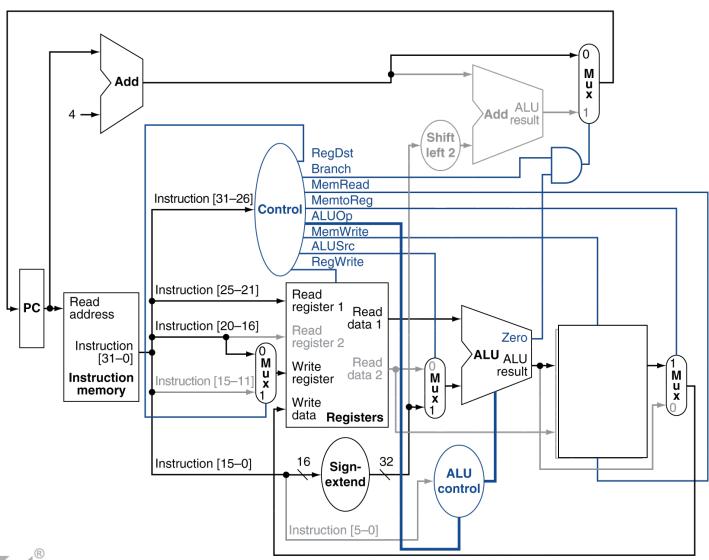
R-Type Instruction







Load Instruction







Branch-on-Equal Instruction

