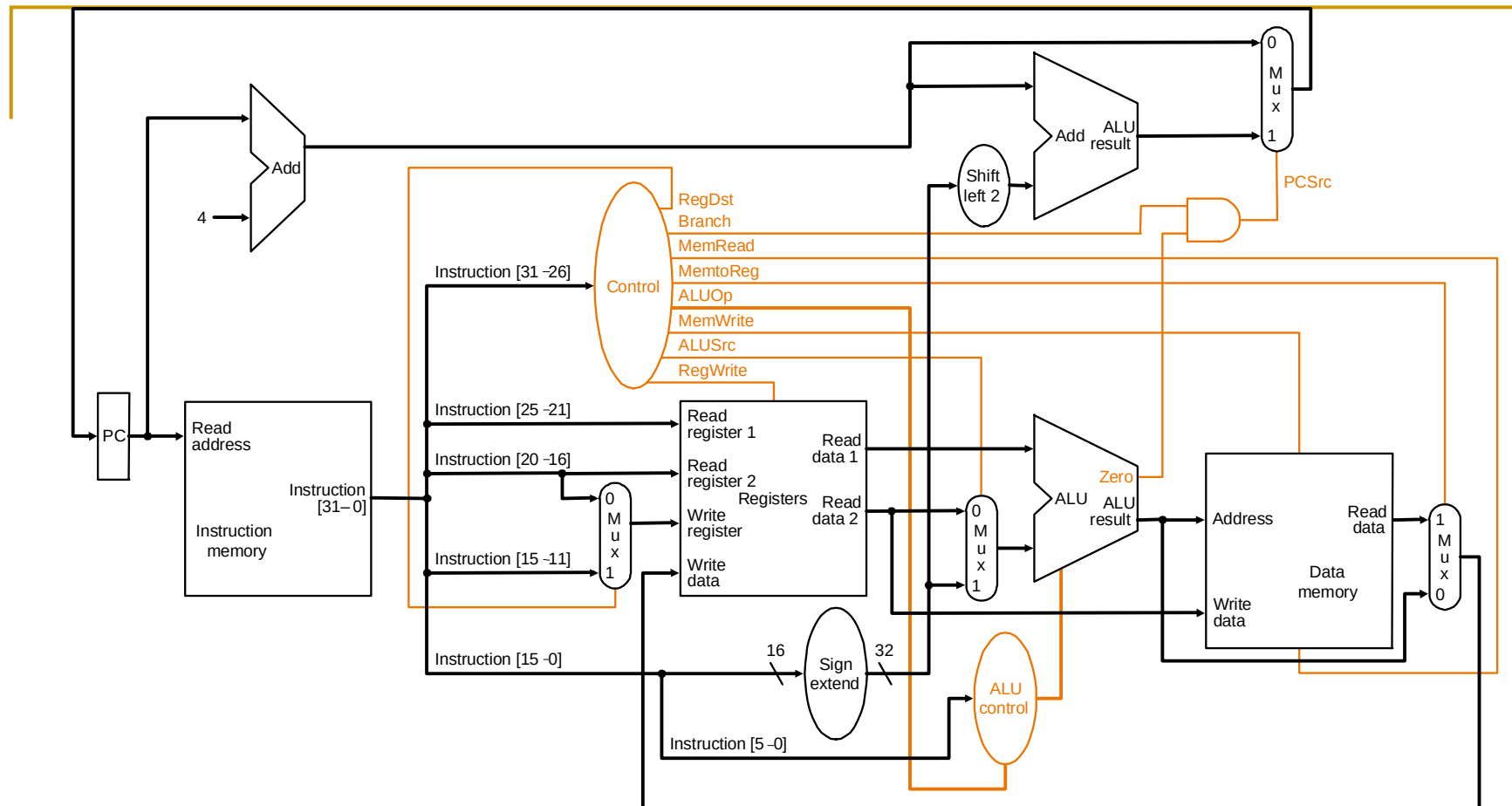

Arquiteturas de Computadores

Implementação de MIPS monociclo (cont.)

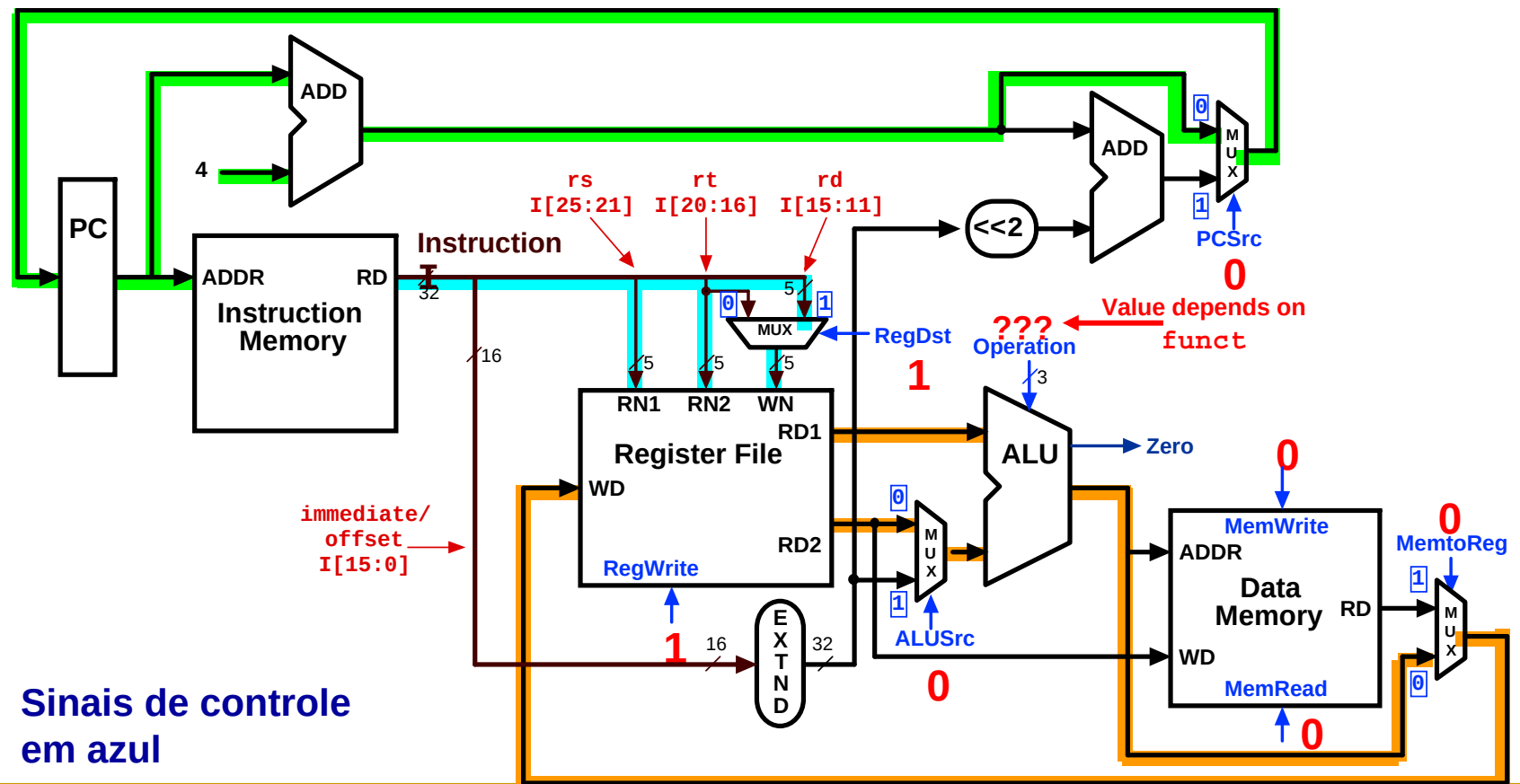
Fontes dos slides: Patterson &
Hennessy book website
(copyright Morgan Kaufmann)
e Dr. Sumanta Guha



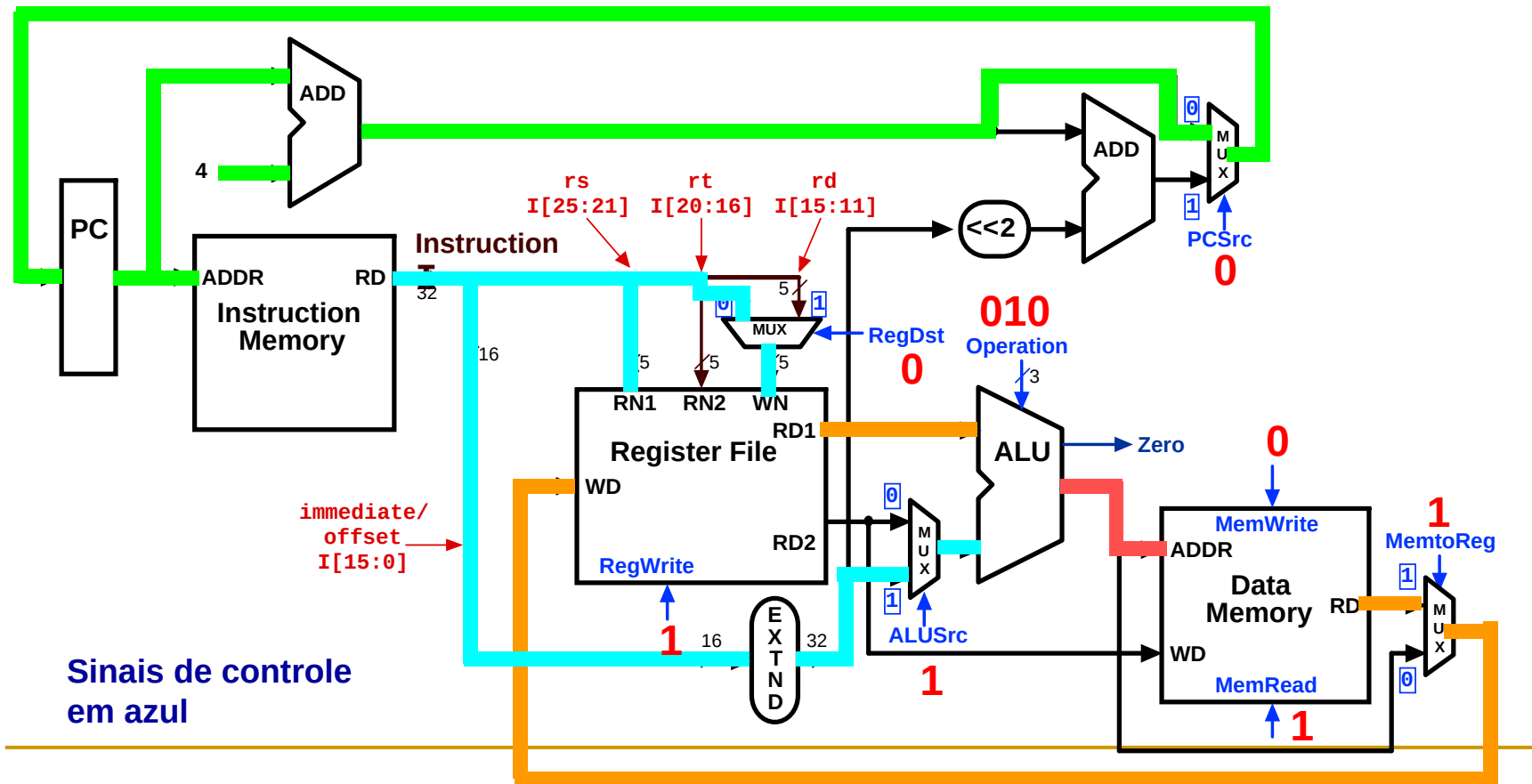
Determinando os sinais de controle a partir do código de operação opcode

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

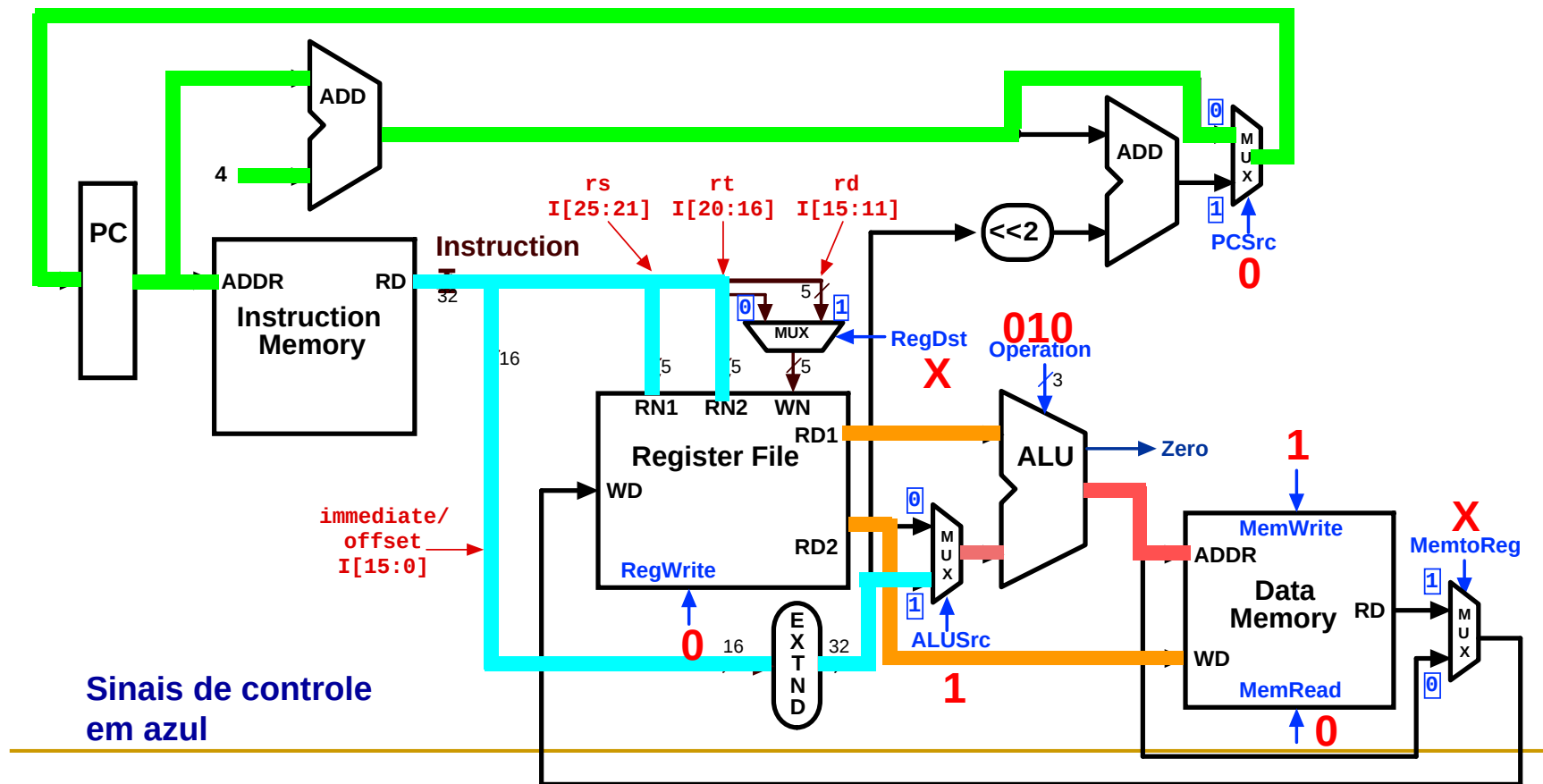
Sinais de controle: Instruções do tipo R



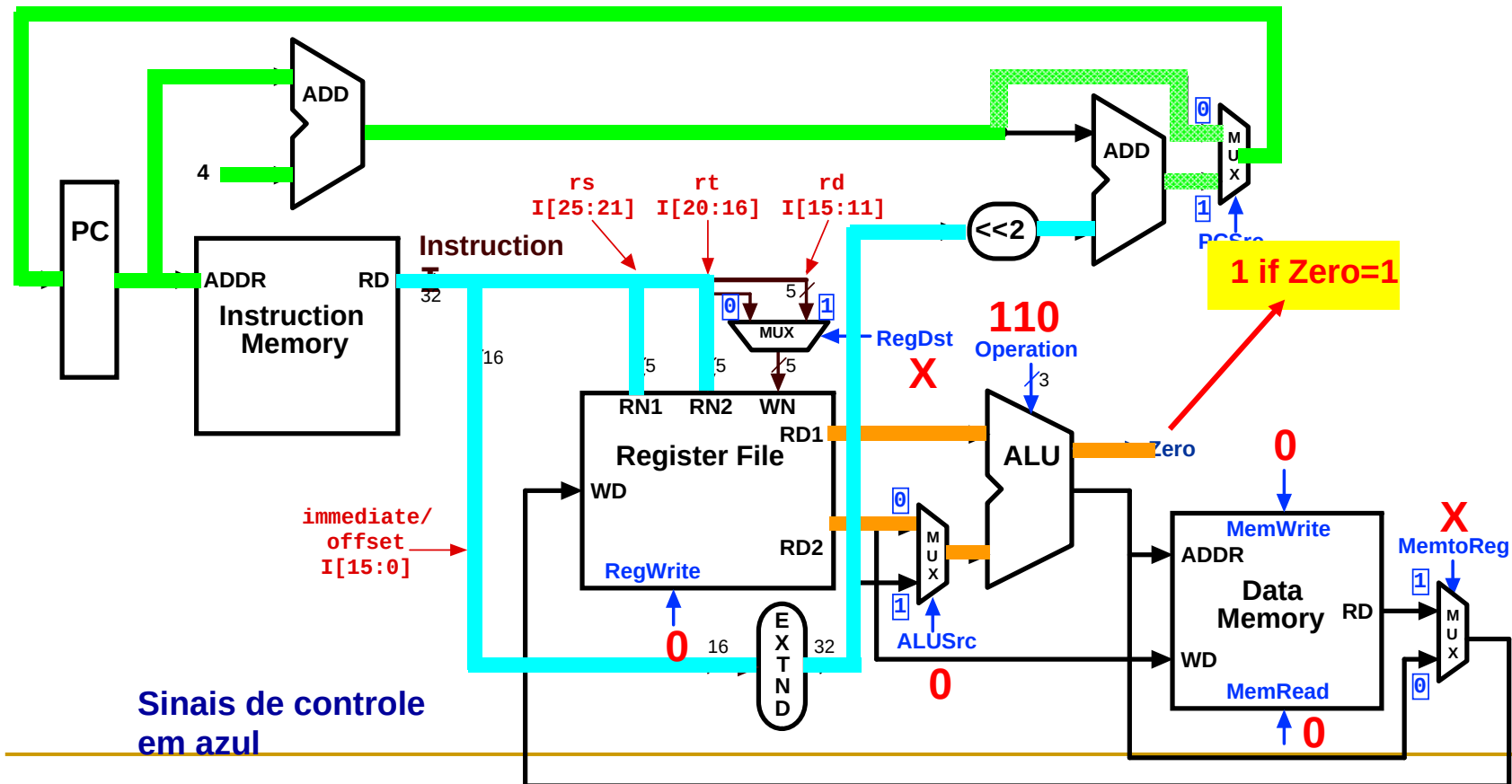
Sinais de controle: Instrução lw



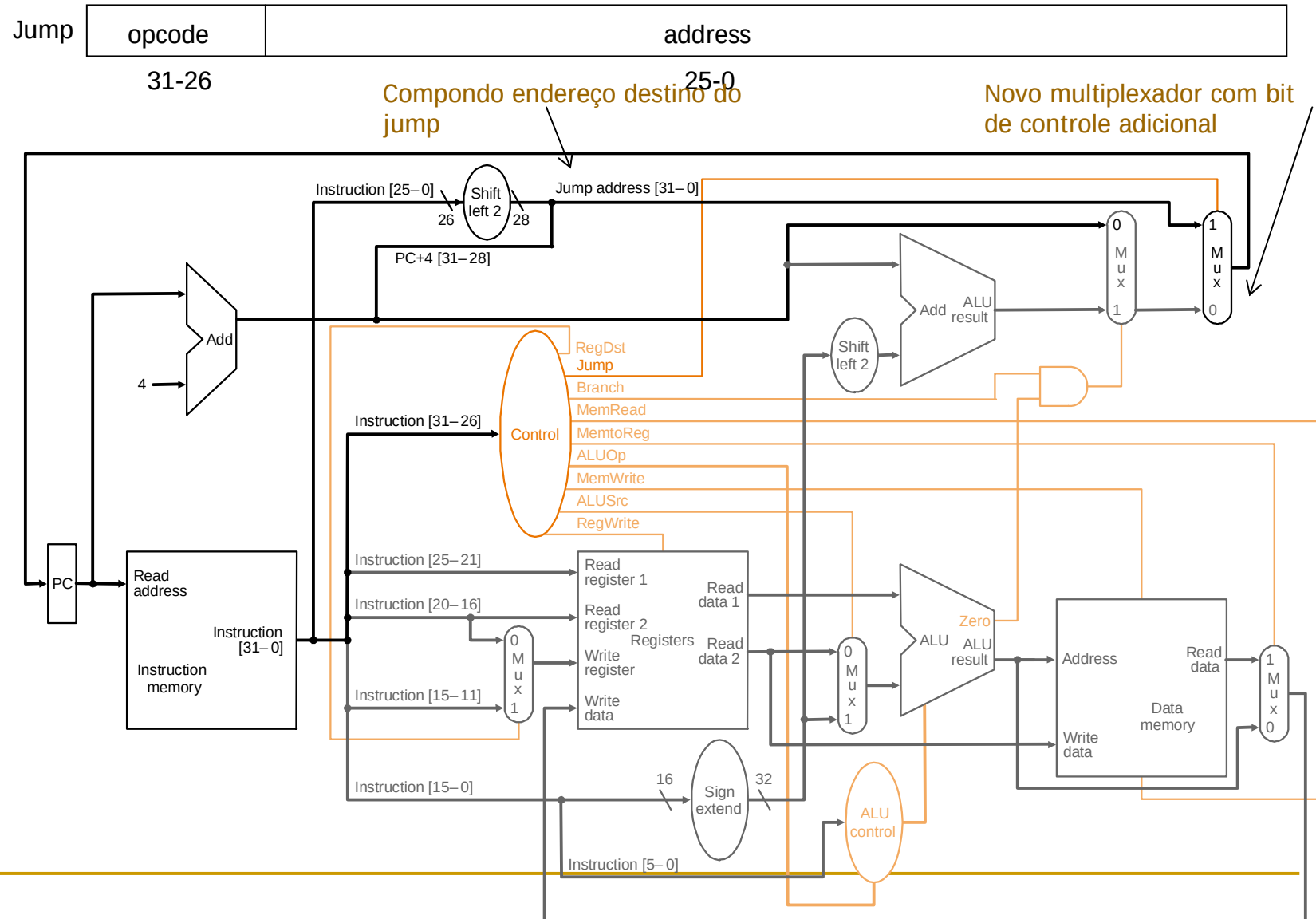
Sinais de controle: Instrução SW



Sinais de controle: Instrução beq

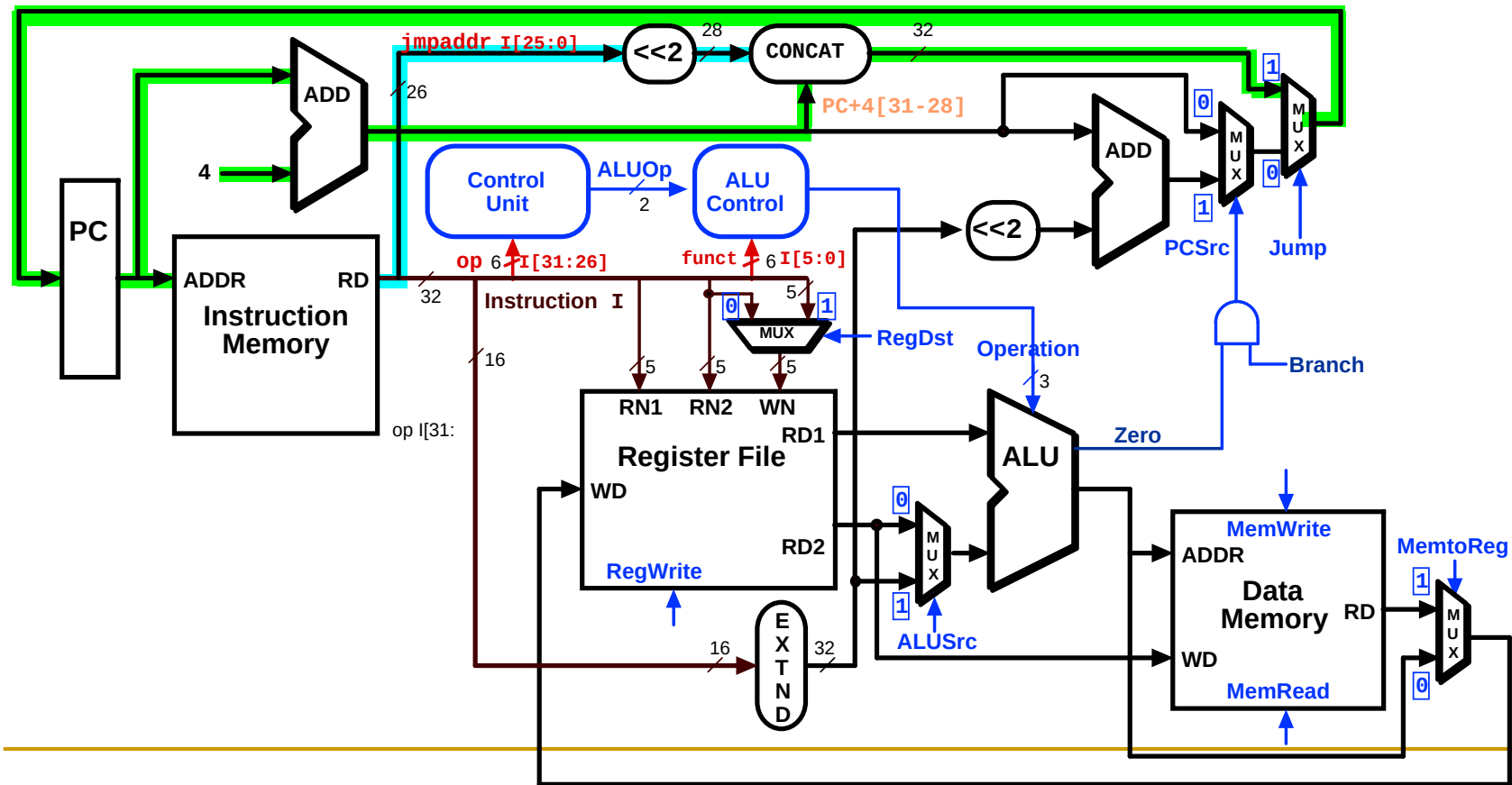


Caminho de dados com controle III



Caminho de dados estendido para jump: unidade de controle gera um novo bit para jump

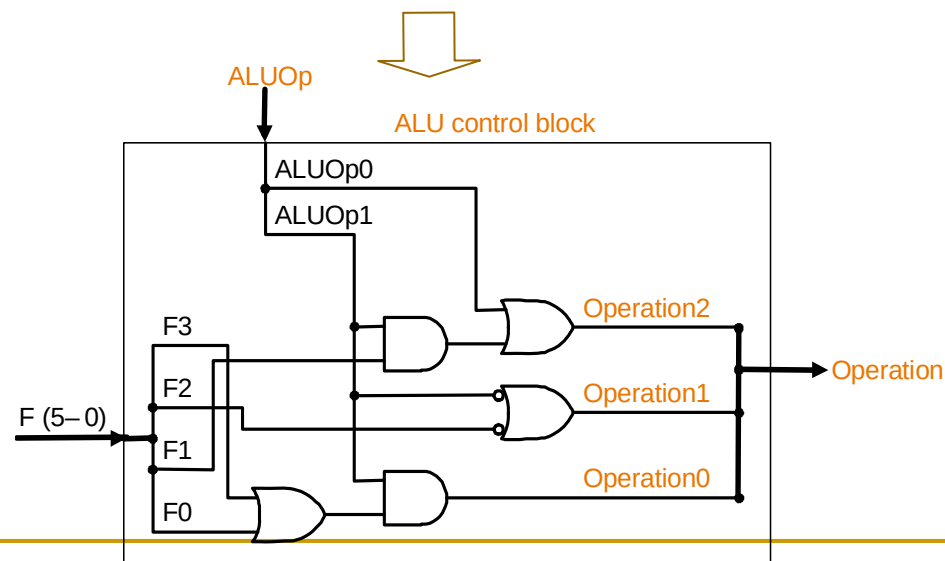
Caminho de dados para executar j



Implementação: Bloco de controle da ALU

ALUOp		Funct field						Operação
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

Tabela verdade para bits de controle da ALU

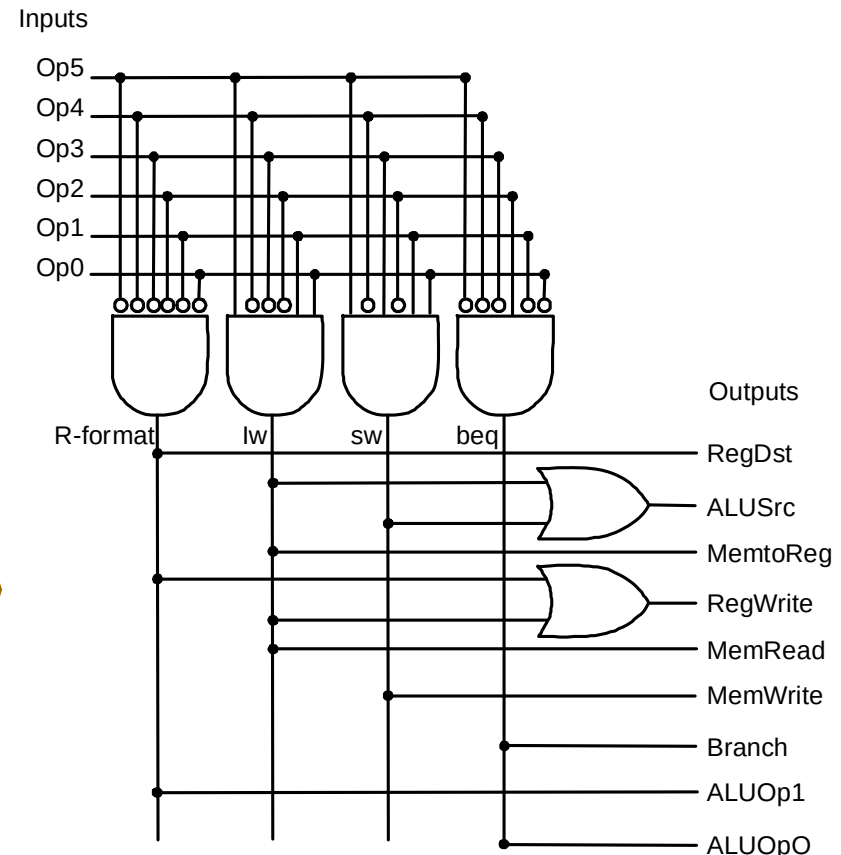


Lógica de controle da ALU

Implementação: Bloco de controle principal

	Nome do Sinal	R-format	lw	sw	beq
Inputs	Op5	0	1	1	0
	Op4	0	0	0	0
	Op3	0	0	1	0
	Op2	0	0	0	1
	Op1	0	1	1	0
	Op0	0	1	1	0
Outputs	RegDst	1	0	X	X
	ALUSrc	0	1	1	0
	MemtoReg	0	1	X	X
	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
	ALUOp2	0	0	0	1

Tabela verdade para os sinais de controle do bloco principal



Controle por PLA (programmable logic array): qualquer expressão lógica pode ser expressa por produtos