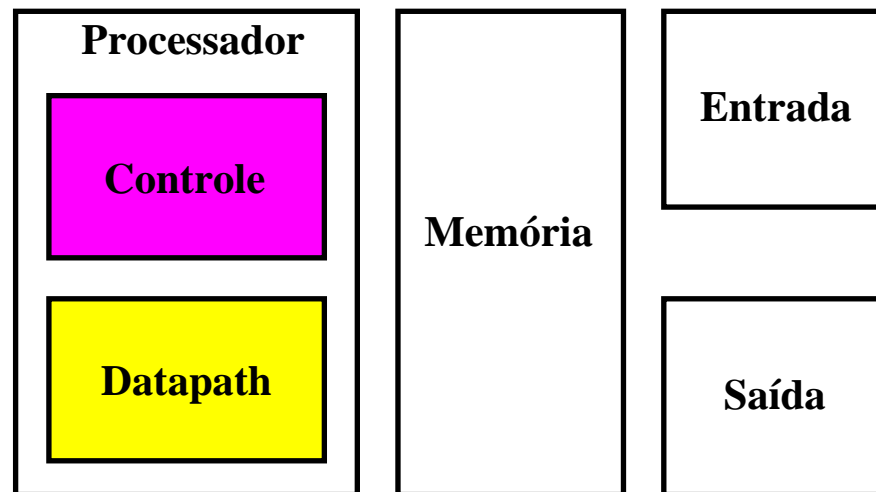
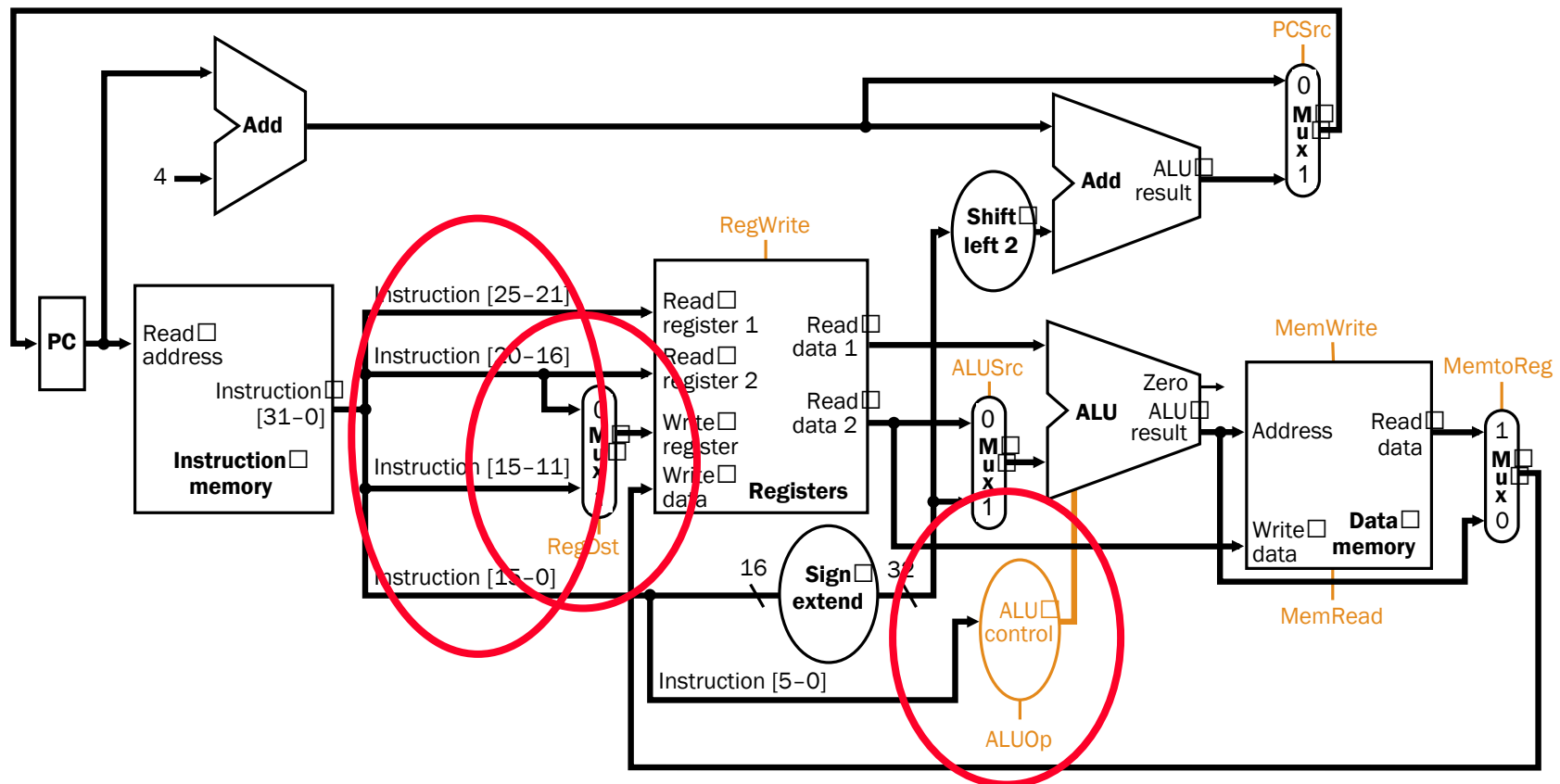


# CPU: comportamento de um datapath



# Incluindo o controle da ALU

- Campos do IR + controle ALU + mux p/ reg. destino

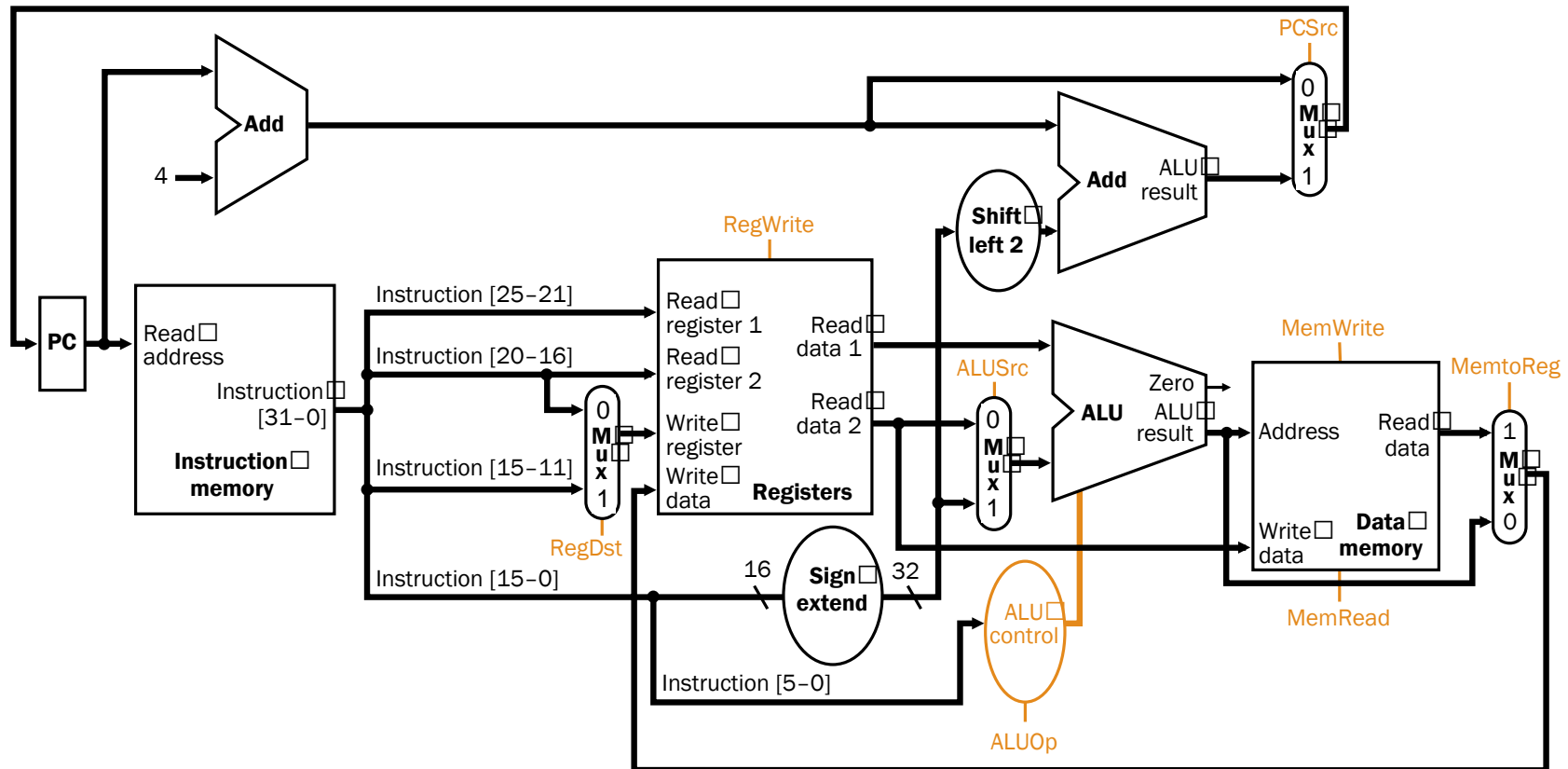


# A função de cada sinal de controle

Nome	Efeito desativado	Efeito ativado
RegDst	Número do reg. destino vem do campo rt (bits 20-16)	Número do reg. destino vem do campo rd (bits 15-11)
RegWrite	<u>Nenhum</u>	Reg. Indicado em write reg é escrito com valor na entrada de dados
ALUSrc	O 2º operando da ALU vem da porta de leitura n. 2	O 2º operando da ALU vem da extensão de sinal
MemRead	Nenhum	A conteúdo indicado pelo endereço é colocado na saída de dados
PCSrc	O PC recebe a saída do somador que calcula PC+4	O PC recebe a saída do somador que calcula o endereço-alvo p/ salto
MemWrite	<u>Nenhum</u>	O valor na entrada de dados é armazenado na posição indicada pelo endereço
MemtoReg	O valor realimentado para a porta de escrita vem da ALU	O valor realimentado para a porta de escrita vem da memória de dados

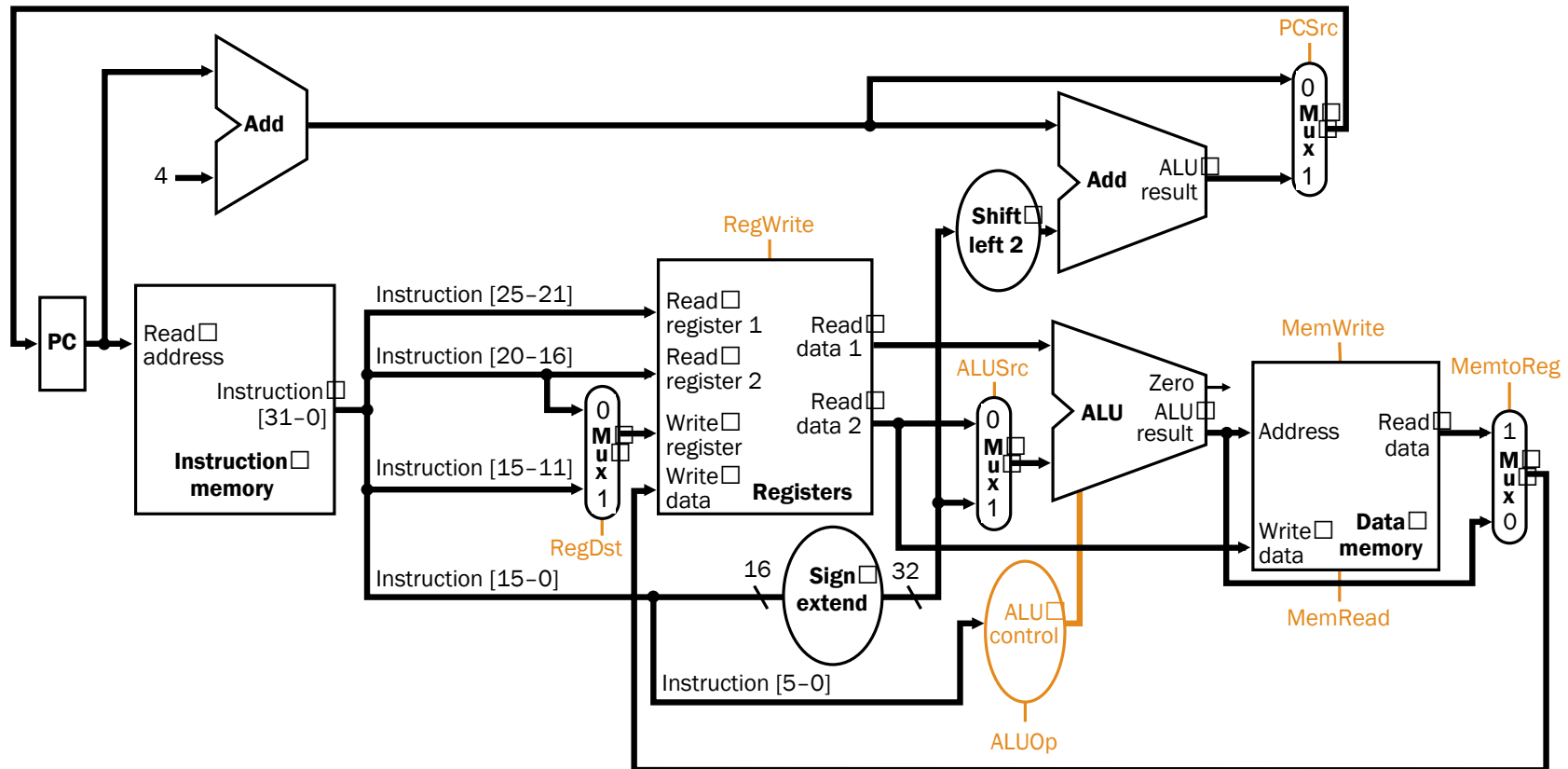
# Implementação do controle

Instr	Reg Dst	ALU Src	Memto-Reg	Reg Write	Mem Read	Mem Write	Branch	ALU Op1	ALU Op2
R	1	0	0	1	0	0	0	1	0



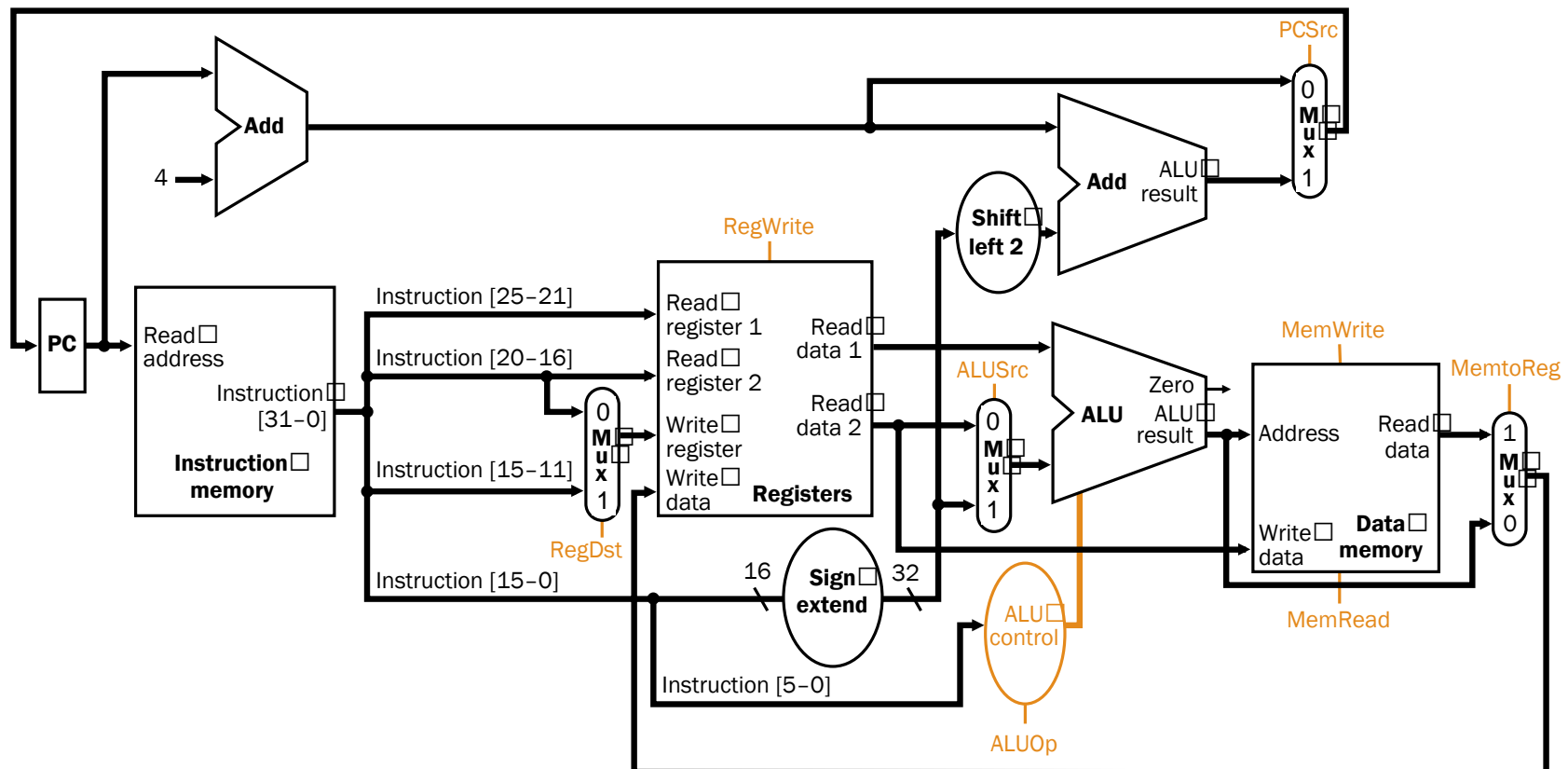
# Implementação do controle

Instr	Reg Dst	ALU Src	Memto-Reg	Reg Write	Mem Read	Mem Write	Branch	ALU Op1	ALU Op2
lw	0	1	1	1	1	0	0	0	0



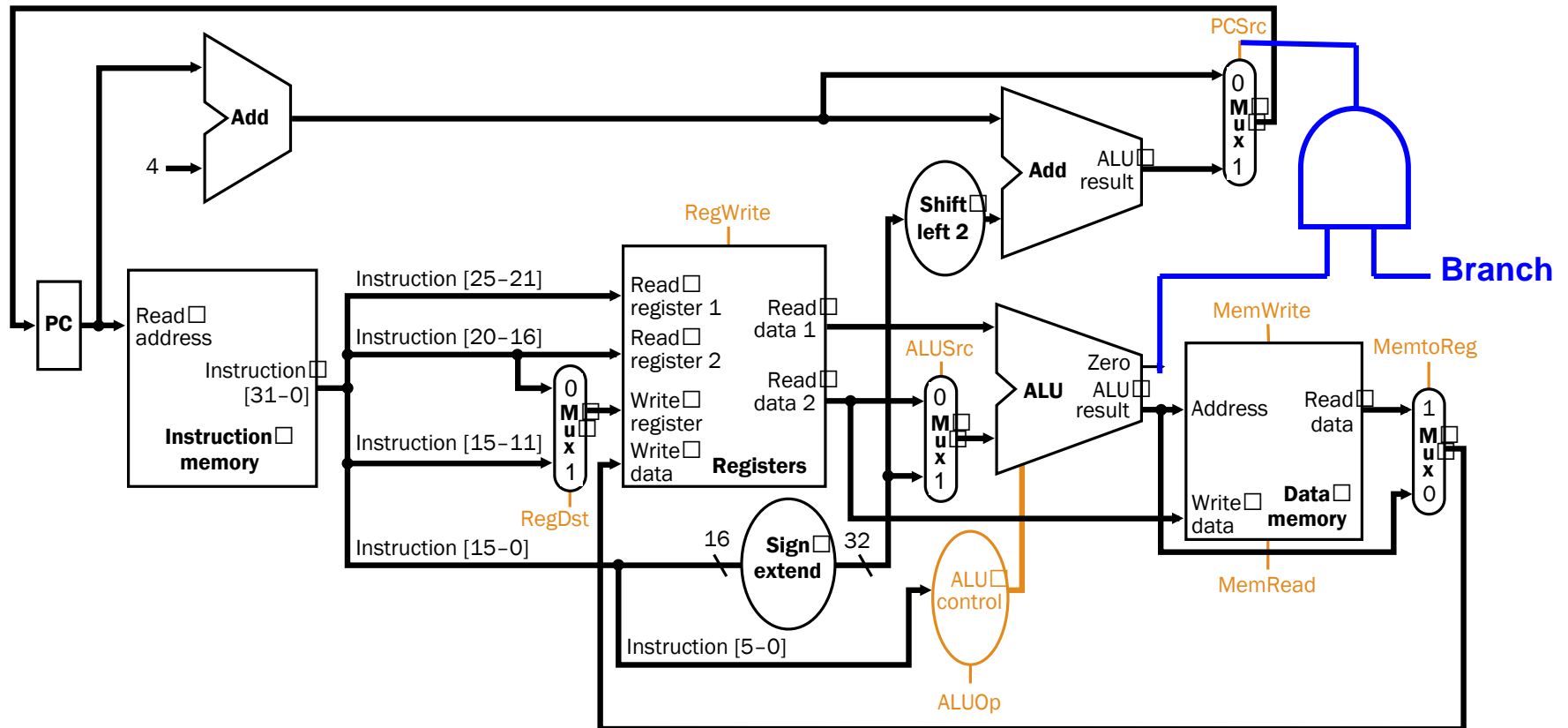
# Implementação do controle

Instr	Reg Dst	ALU Src	Memto-Reg	Reg Write	Mem Read	Mem Write	Branch	ALU Op1	ALU Op2
sw	X	1	X	0	0	1	0	0	0



# Implementação do controle

Instr	Reg Dst	ALU Src	Memto-Reg	Reg Write	Mem Read	Mem Write	Branch	ALU Op1	ALU Op2
beq	X	0	X	0	0	0	1	0	1



# Implementação do controle

Instr	Reg Dst	ALU Src	Memto-Reg	Reg Write	Mem Read	Mem Write	Branch	ALU Op1	ALU Op2
R	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



# Comportamento da unidade de controle

Instr	Reg Dst	ALU Src	Memto-Reg	Reg Write	Mem Read	Mem Write	Branch	ALU Op1	ALU Op2
000000	1	0	0	1	0	0	0	1	0
100011	0	1	1	1	1	0	0	0	0
101011	X	1	X	0	0	1	0	0	0
000100	X	0	X	0	0	0	1	0	1



entradas

# Comportamento da unidade de controle

Instr	Reg Dst	ALU Src	Memto-Reg	Reg Write	Mem Read	Mem Write	Branch	ALU Op1	ALU Op2
000000	1	0	0	1	0	0	0	1	0
100011	0	1	1	1	1	0	0	0	0
101011	X	1	X	0	0	1	0	0	0
000100	X	0	X	0	0	0	1	0	1



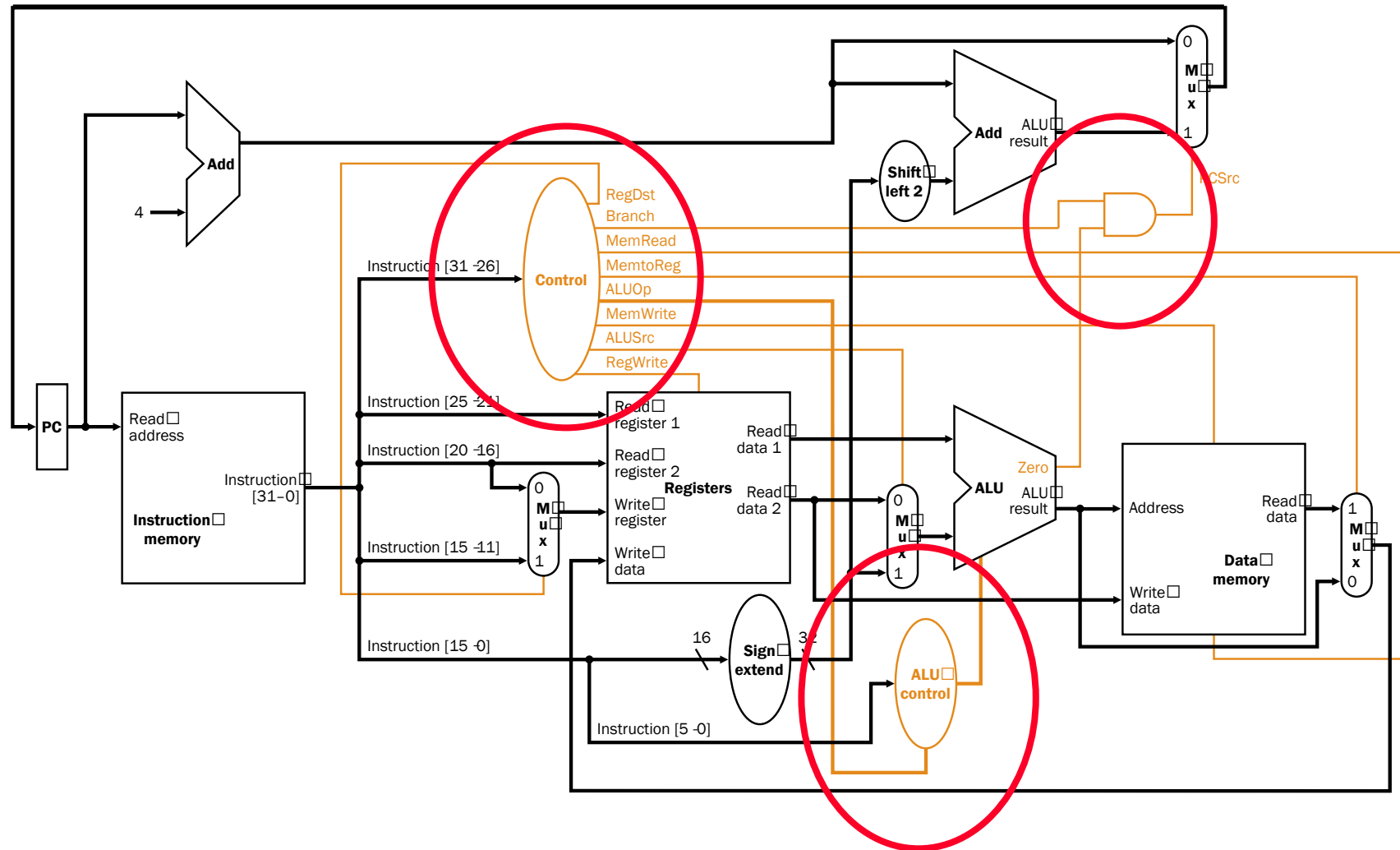
entradas



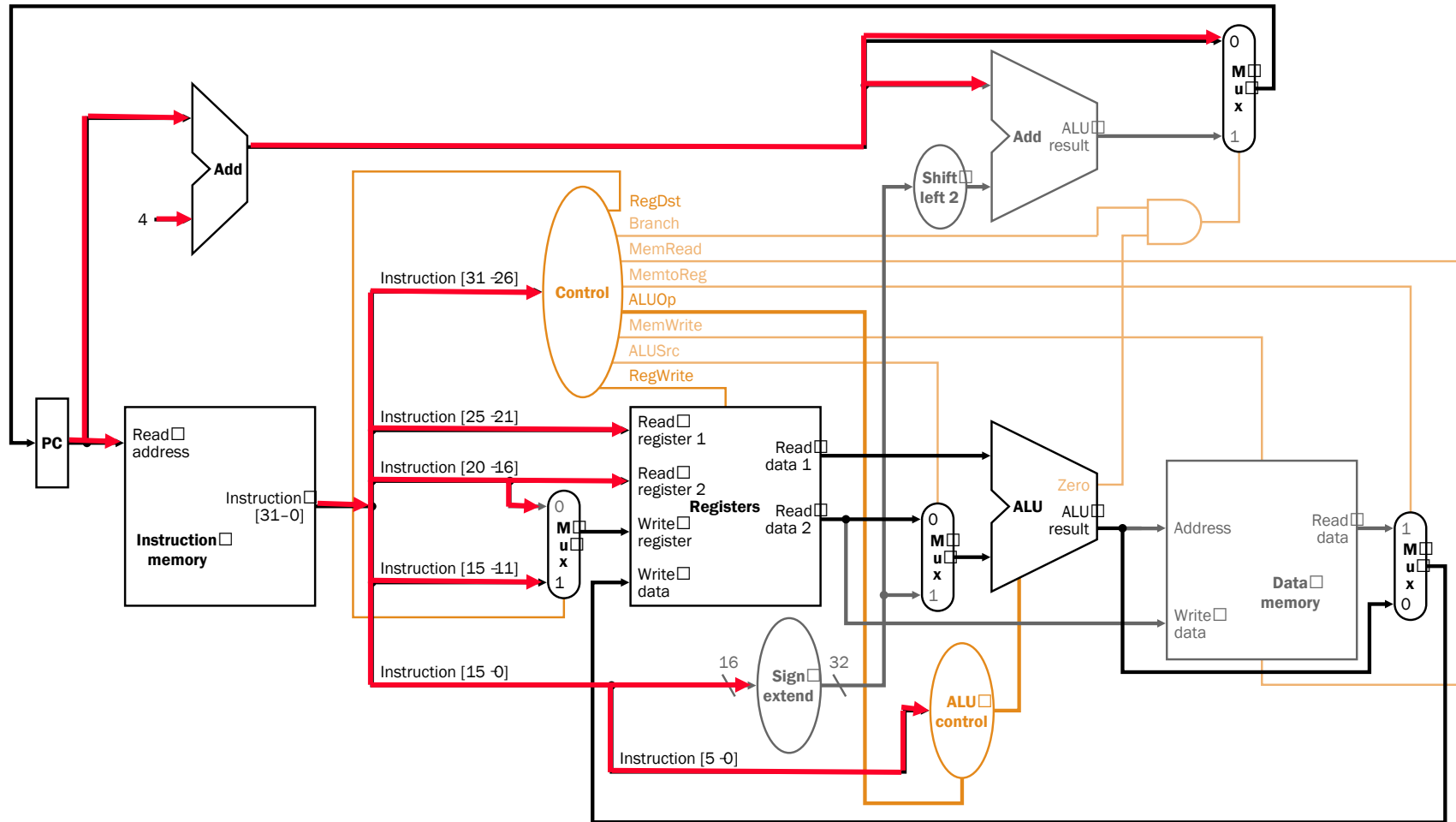
saídas

**Estrutura da unidade de controle (portas lógicas): veja Apêndice C.2**

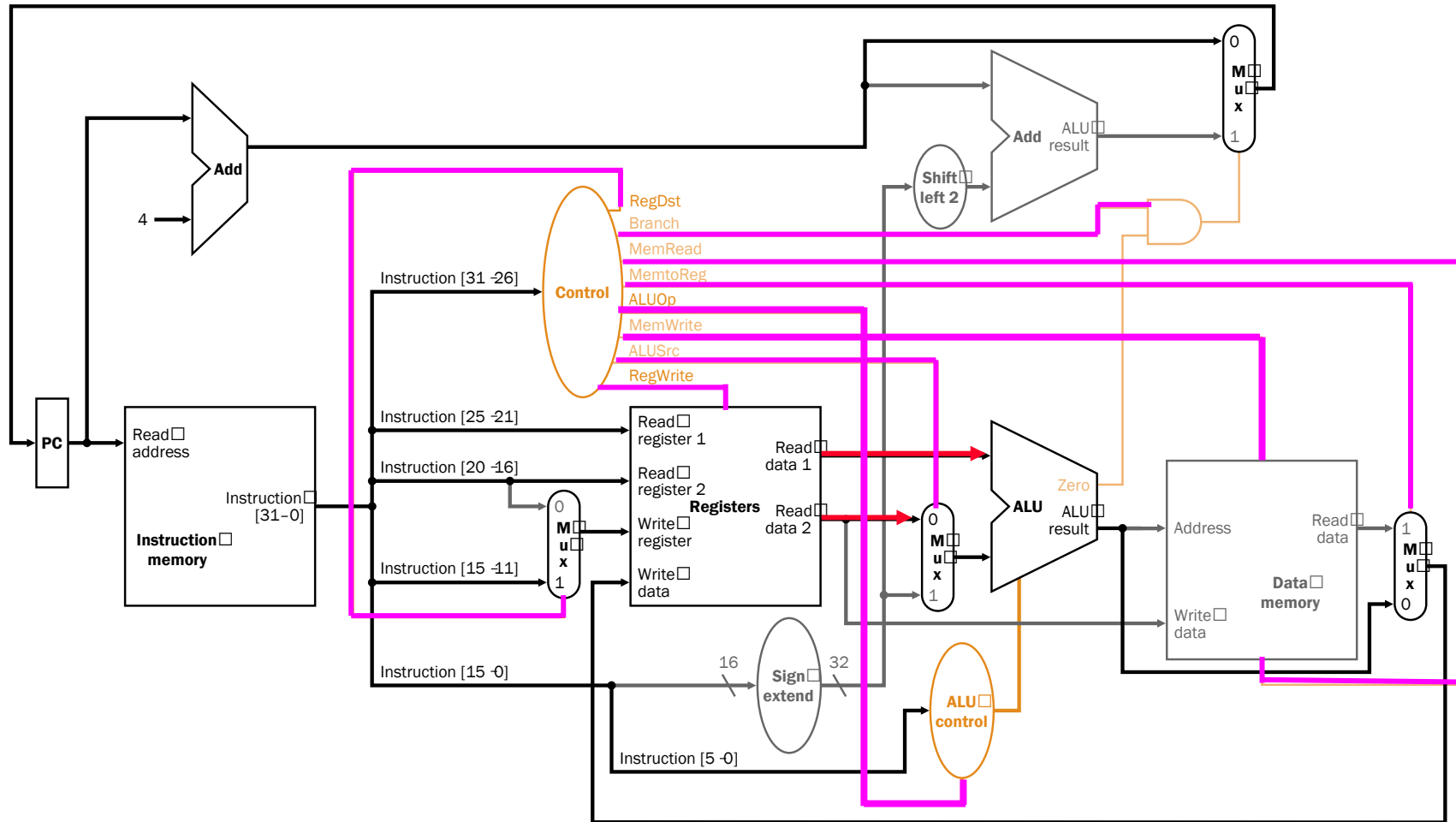
# Incluindo a unidade de controle



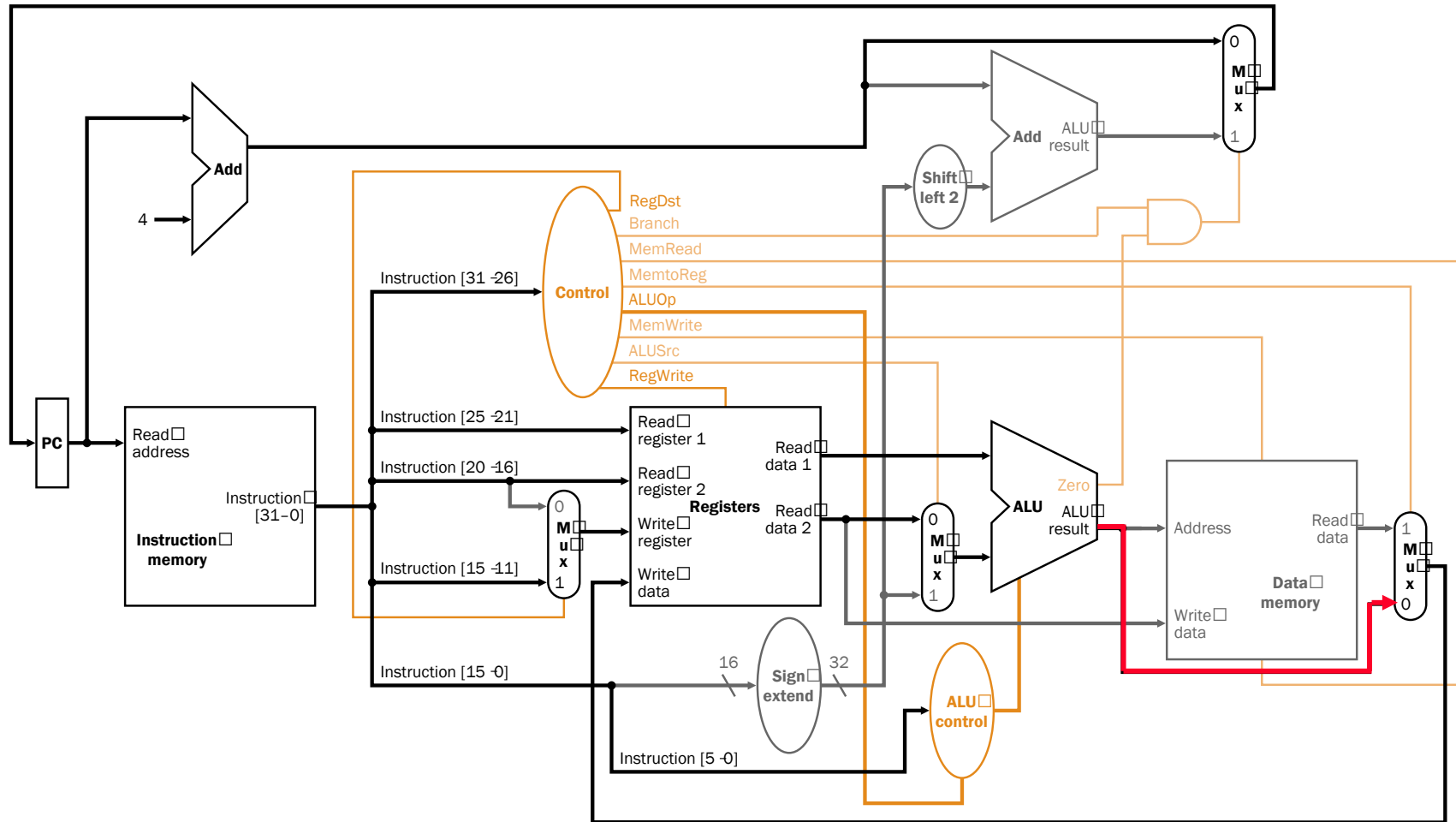
# Instrução tipo R: 1ª fase



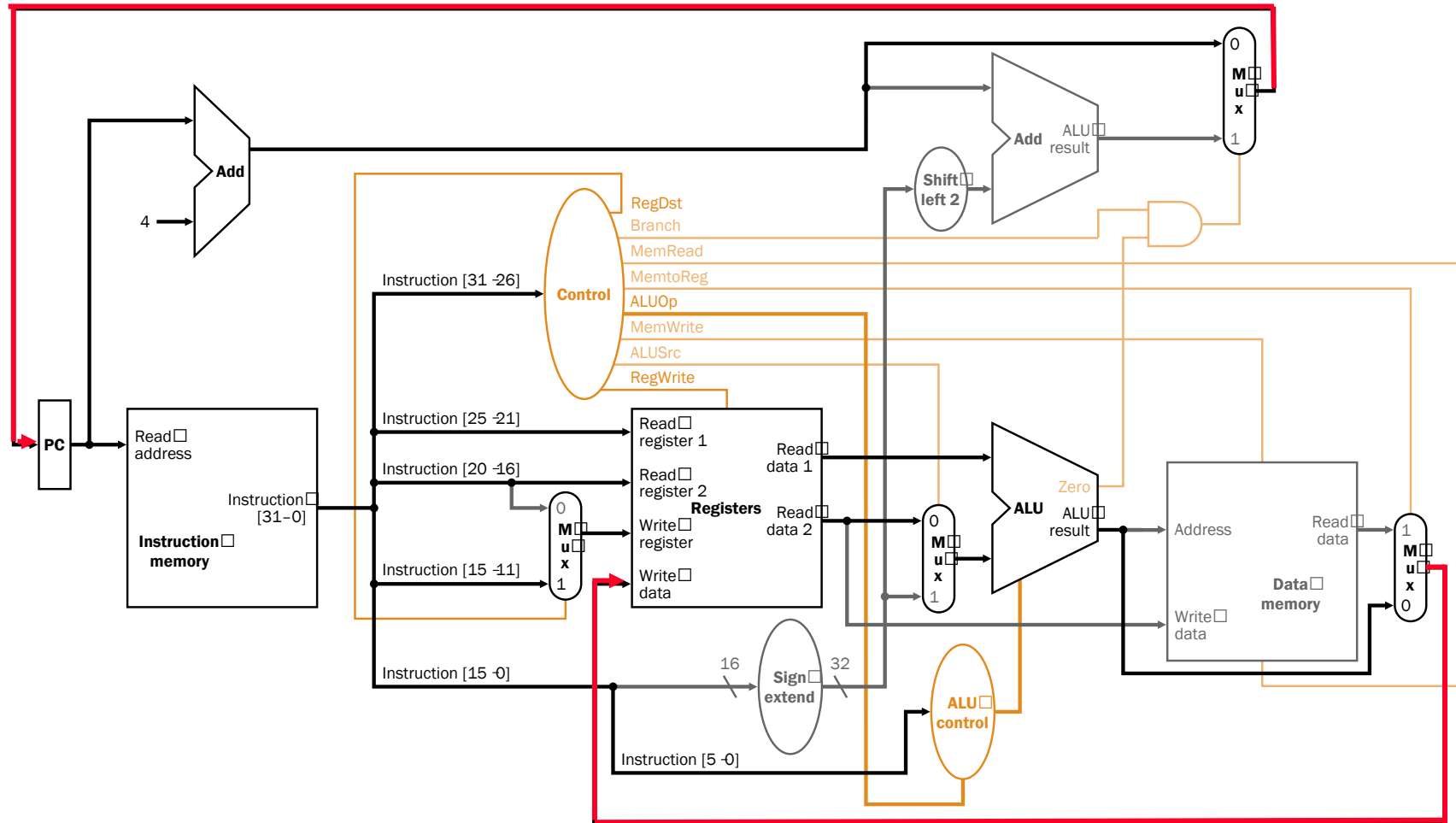
# Instrução tipo R: 2ª fase



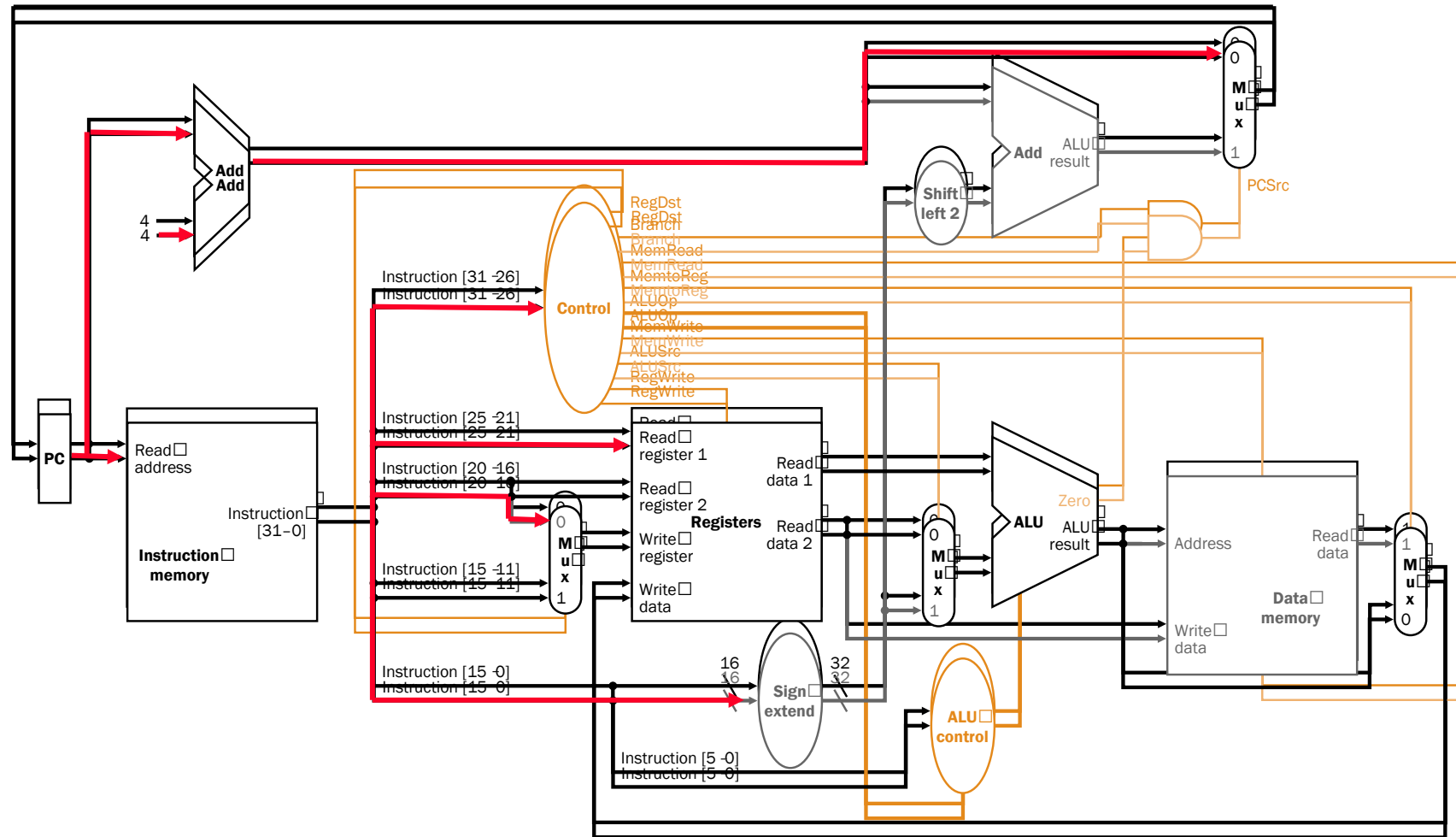
## Instrução tipo R: 3ª fase



# Instrução tipo R: 4ª fase

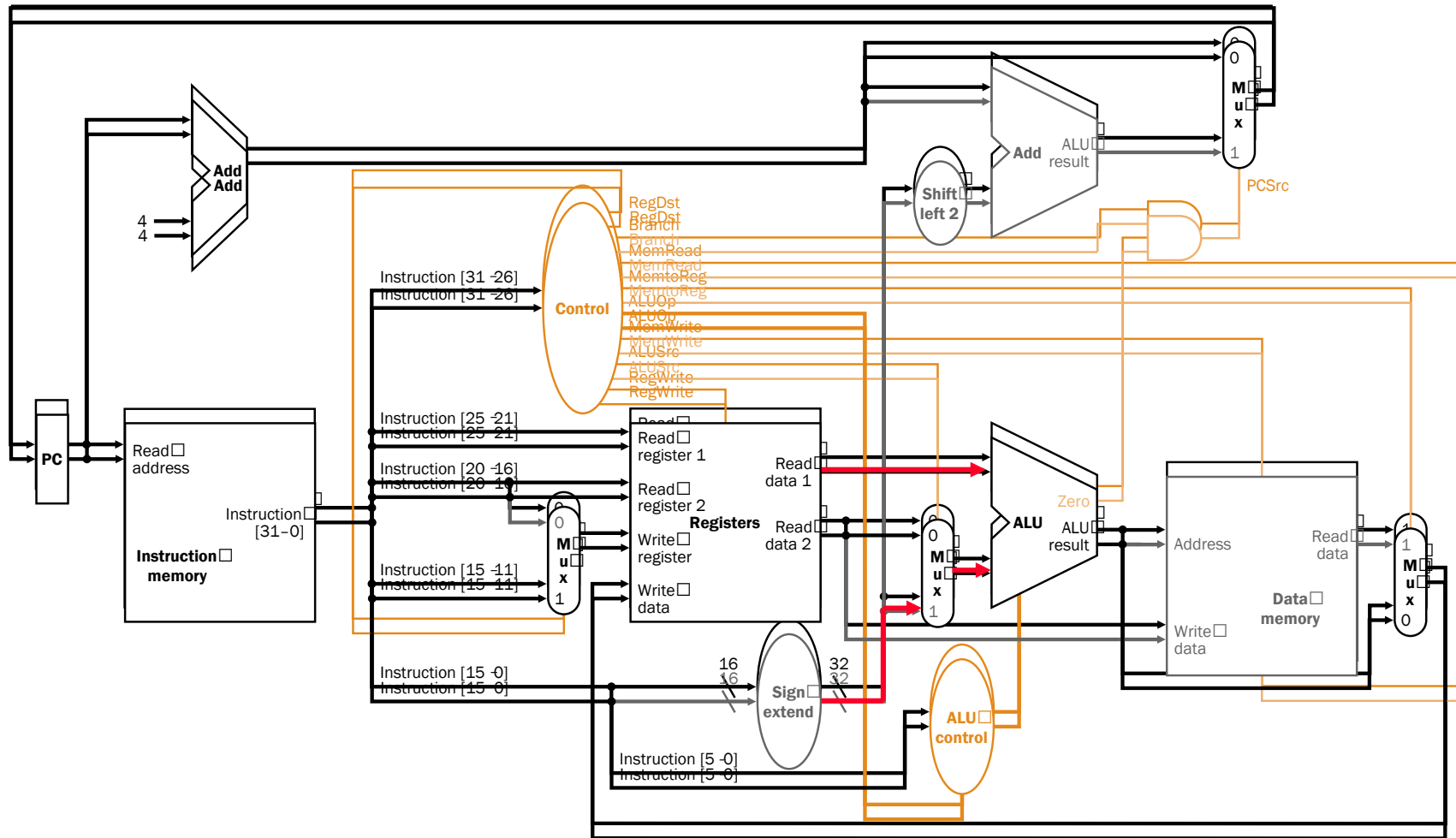


# Instrução load: 1ª fase

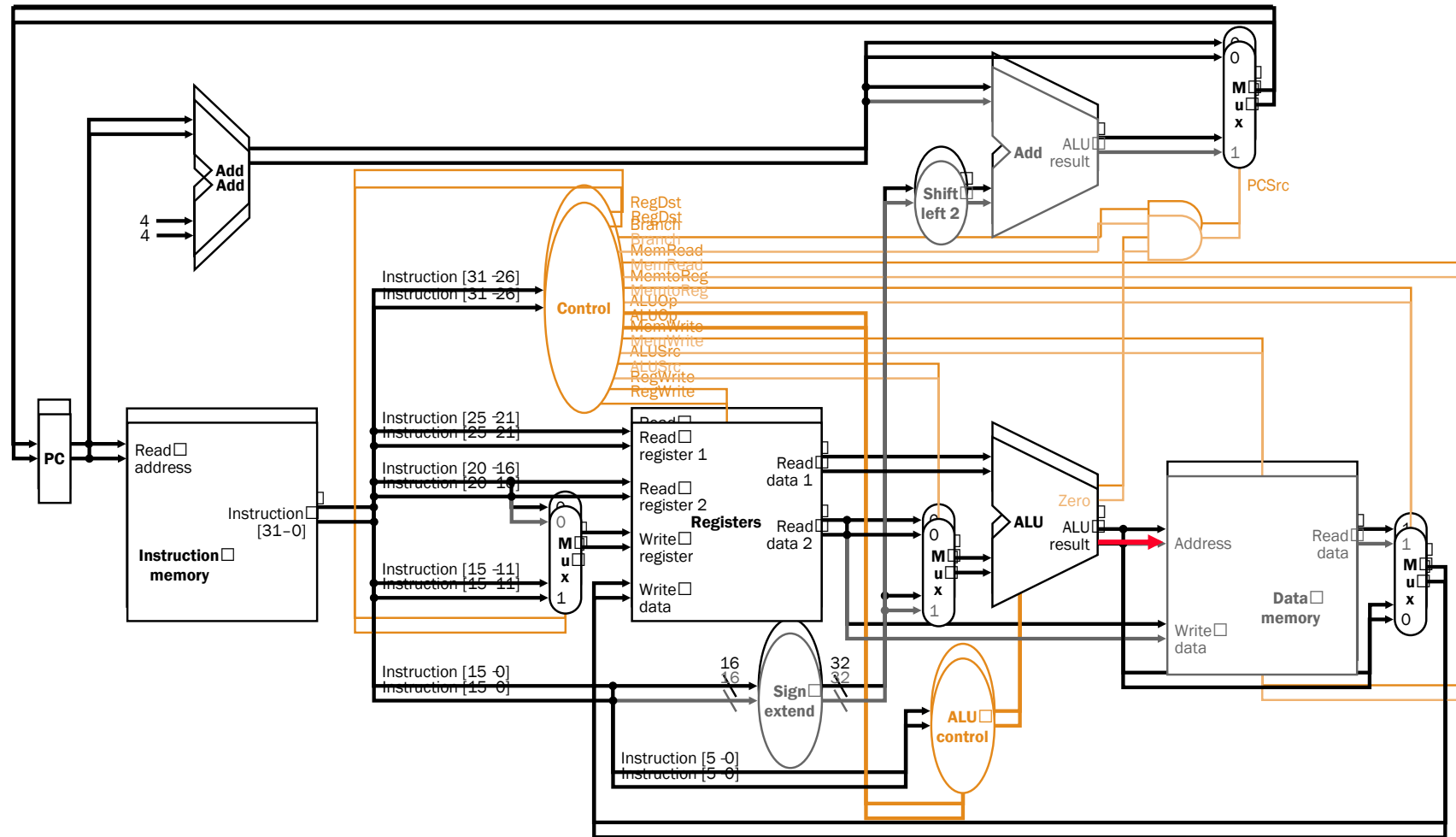




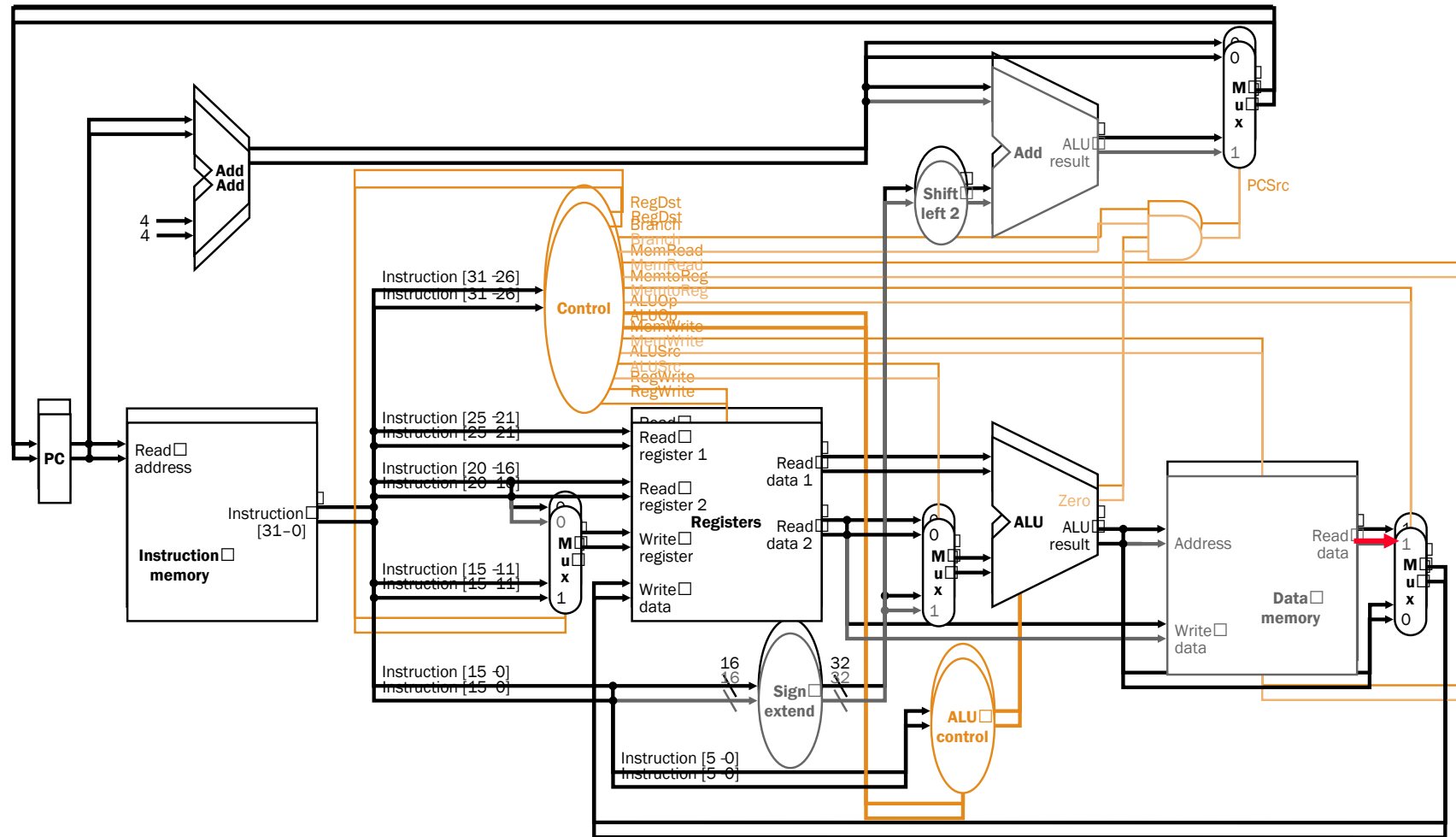
# Instrução load: 2ª. fase



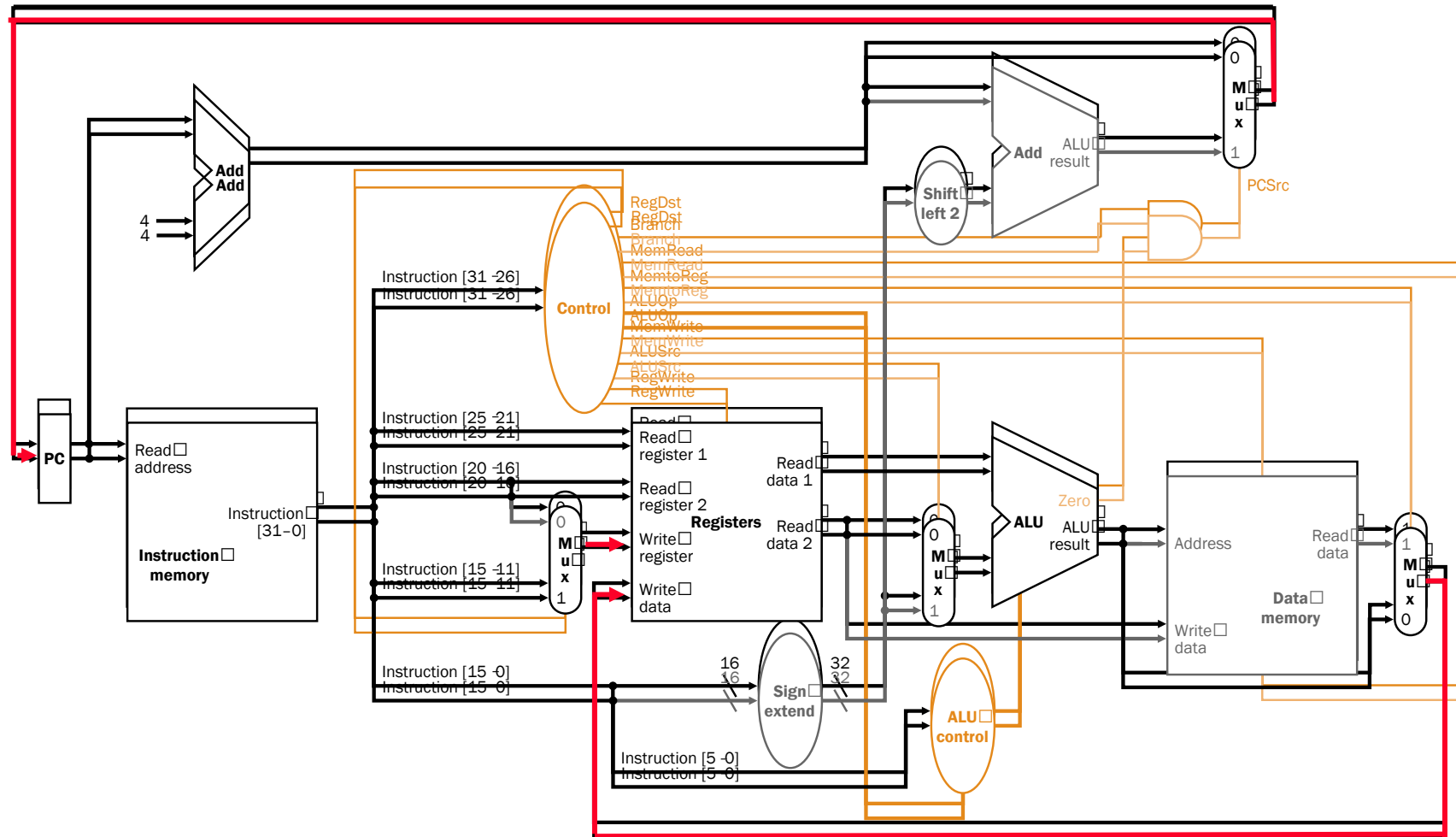
# Instrução load: 3ª fase



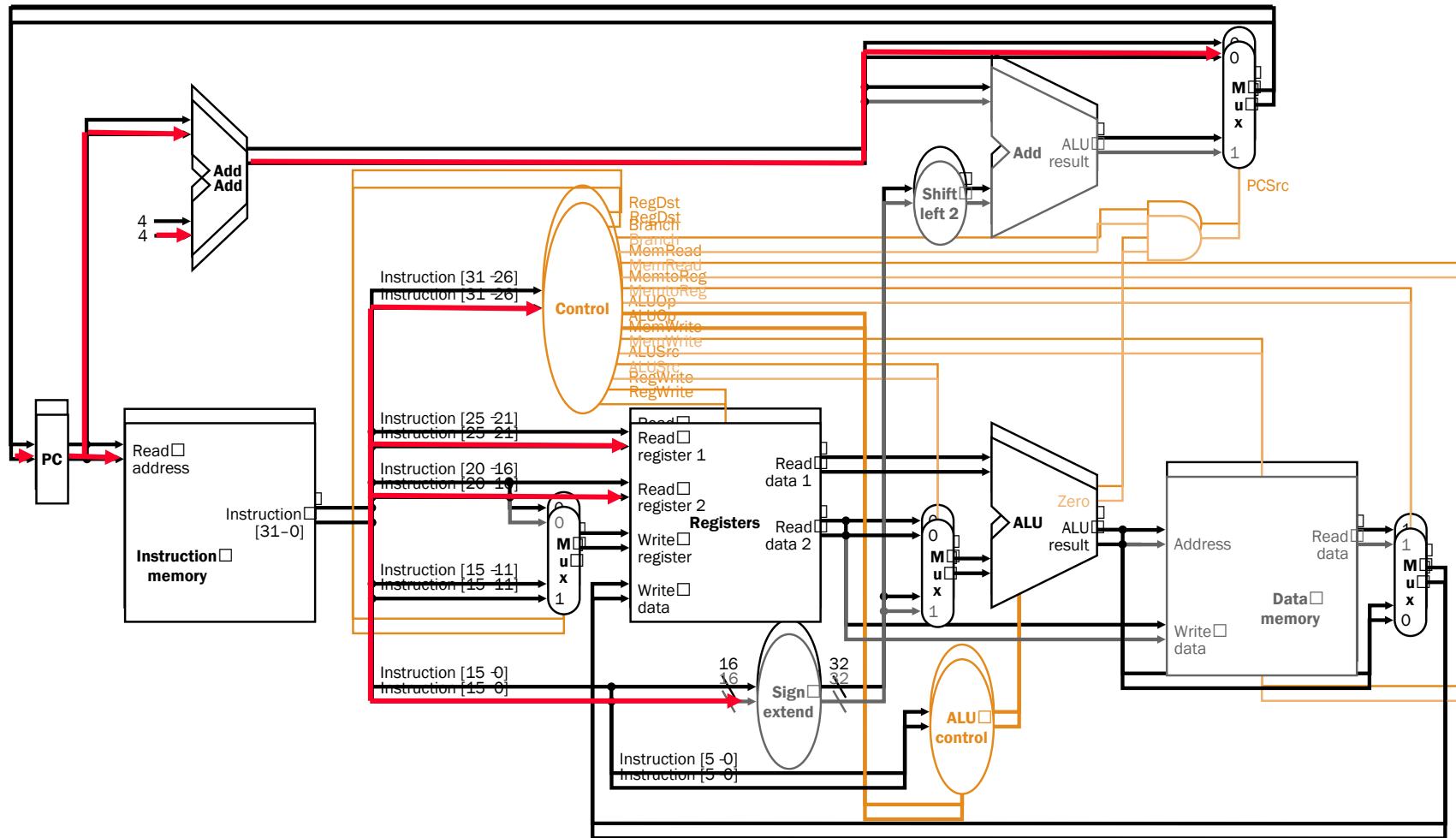
# Instrução load: 4ª fase



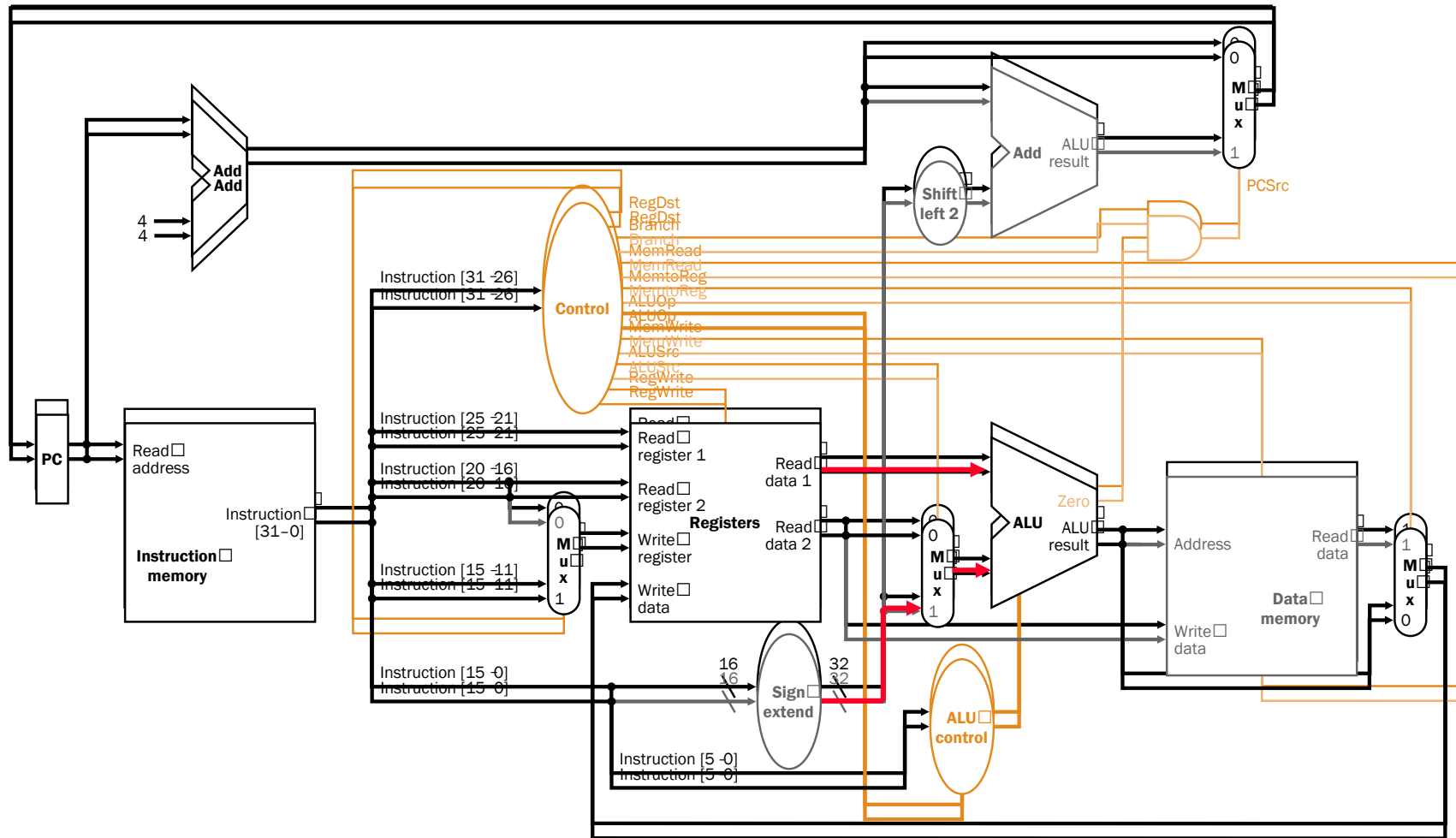
# Instrução load: 5ª. fase



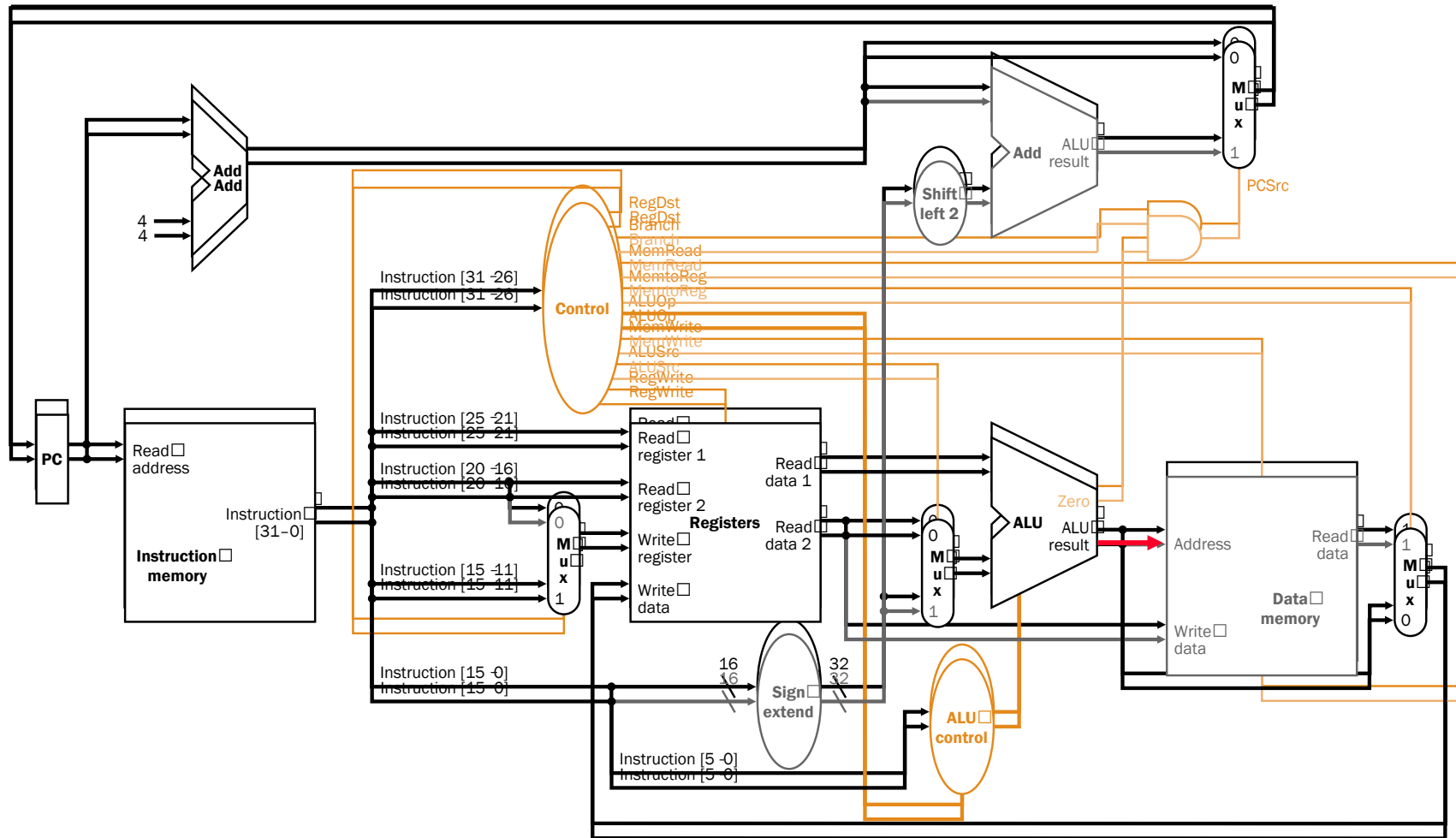
# Instrução store: 1ª fase



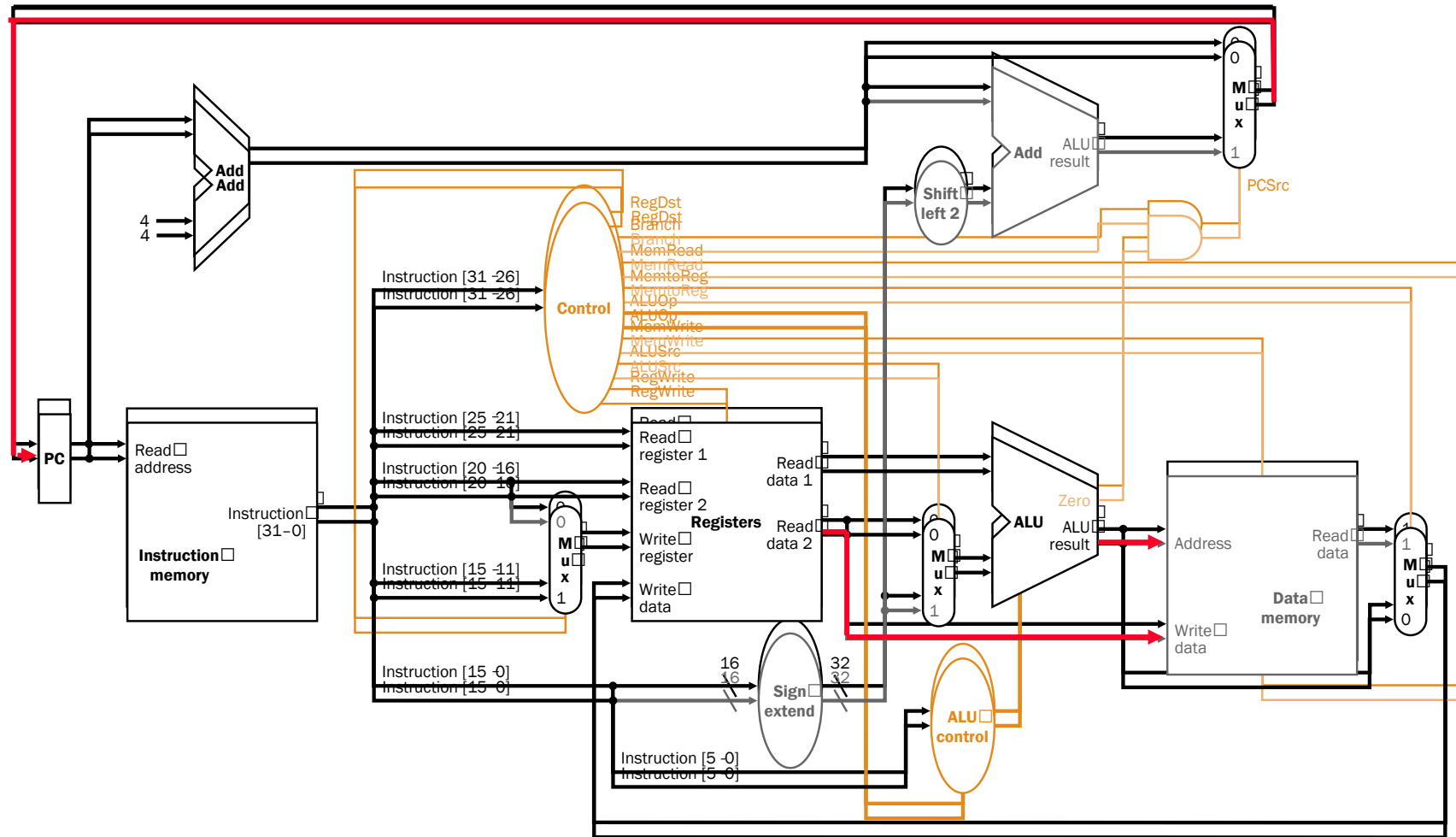
# Instrução store: 2ª fase



# Instrução store: 3ª fase

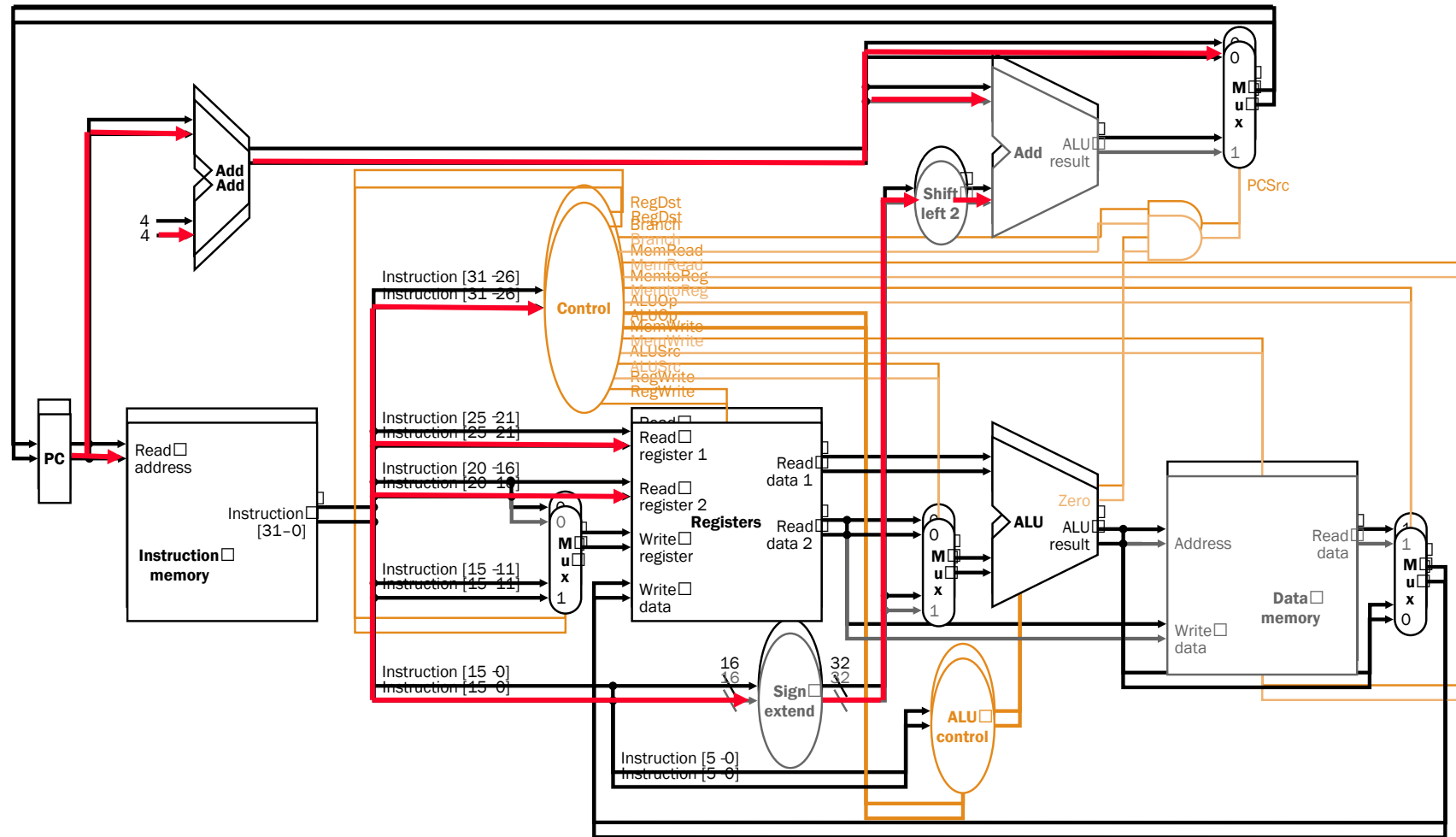


# Instrução store: 4ª fase

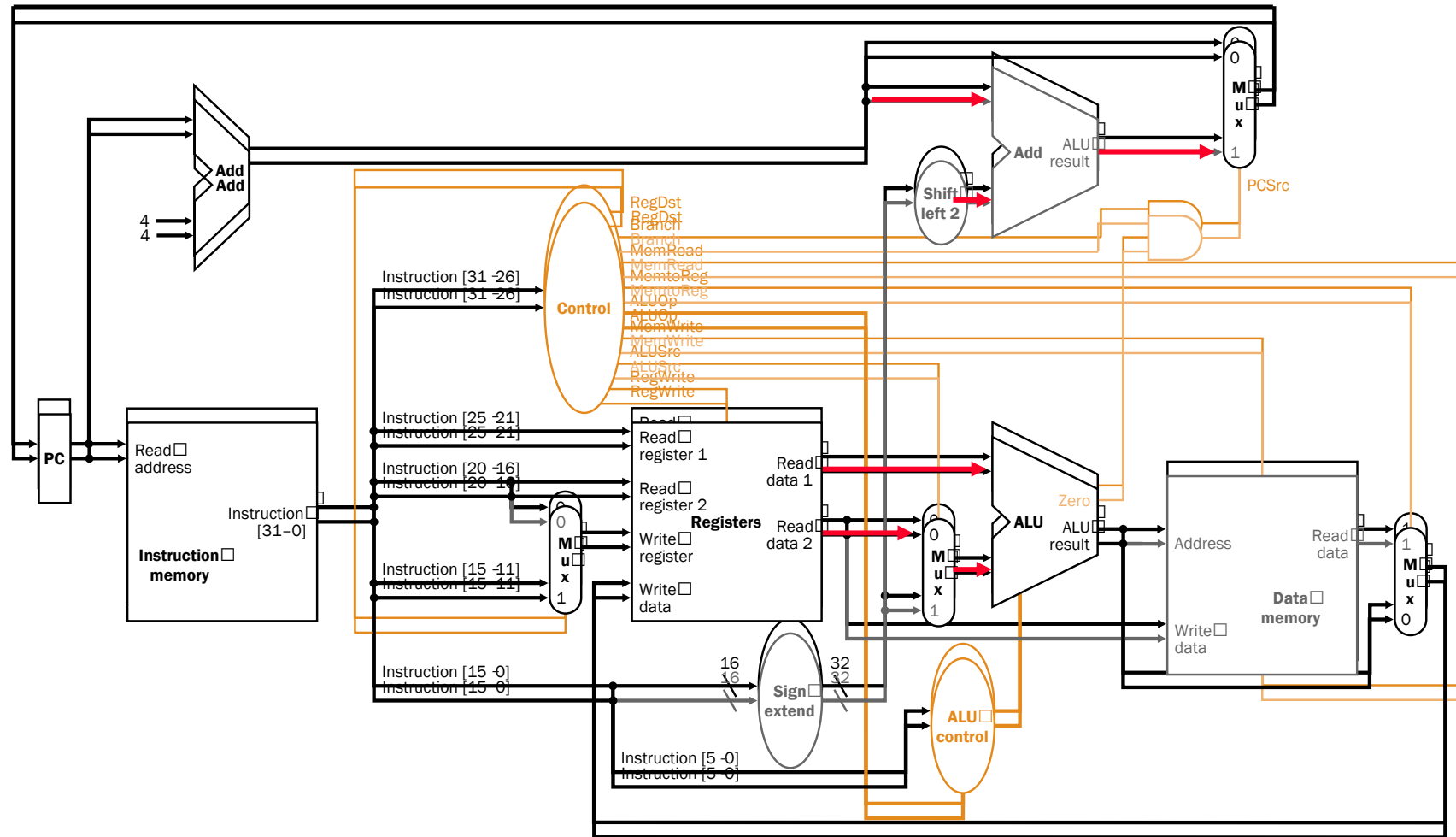




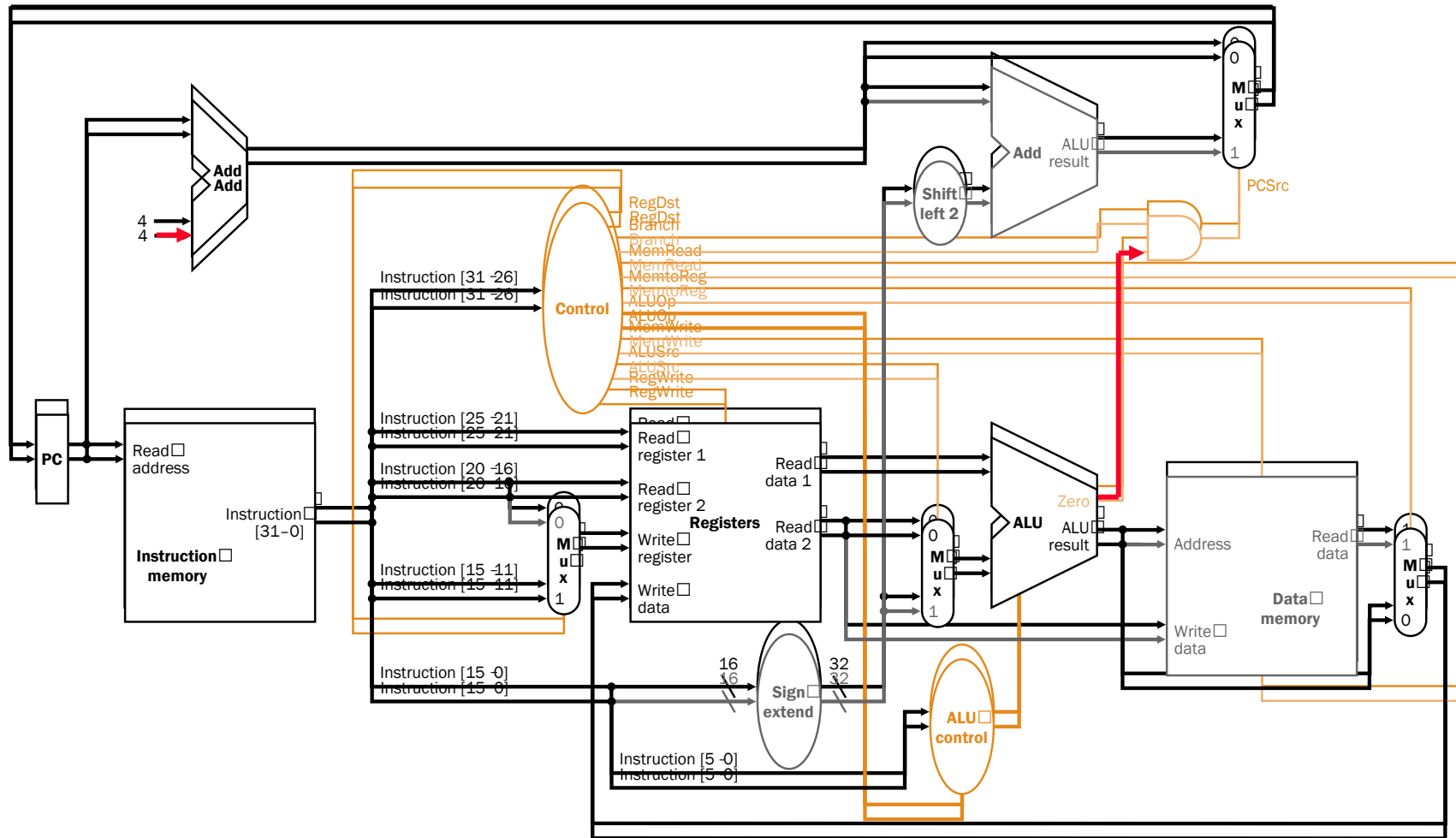
# Instrução beq: 1ª fase



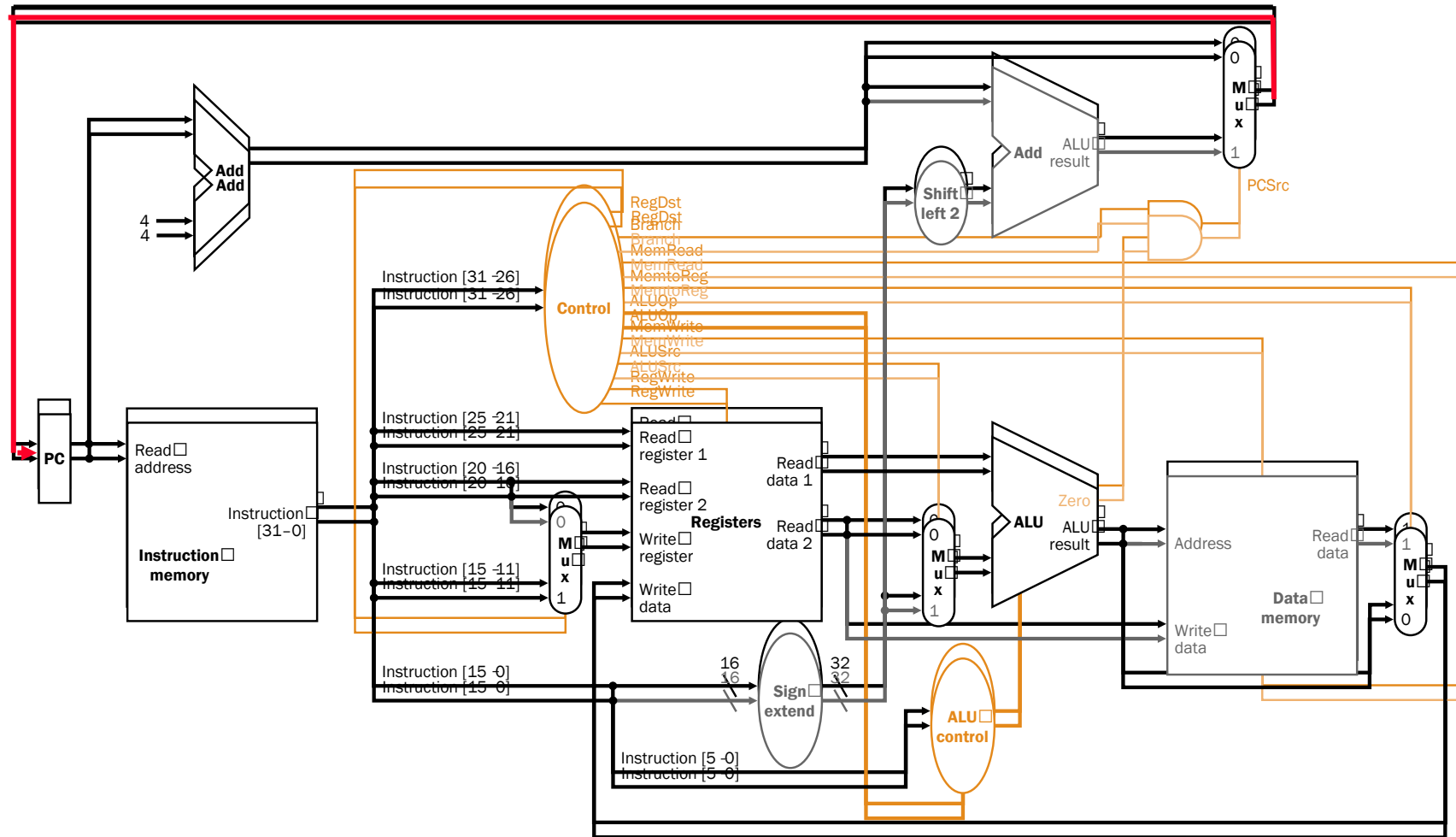
# Instrução beq: 2ª fase



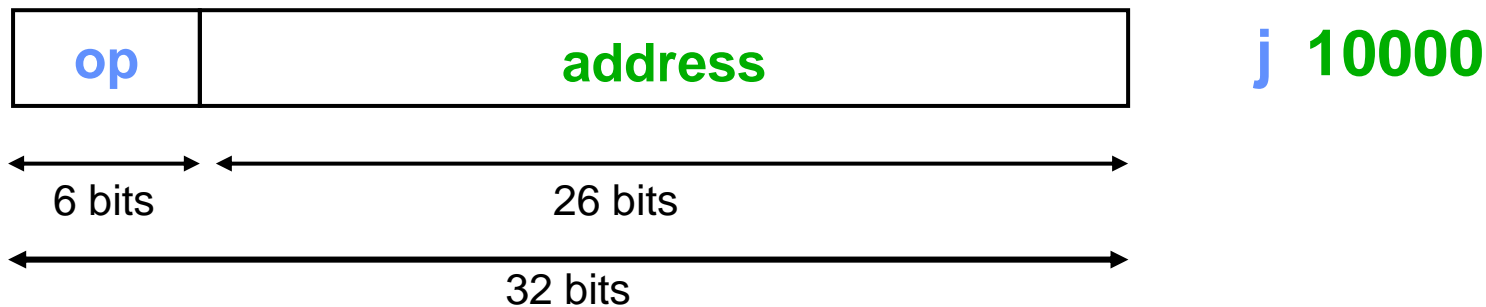
# Instrução beq: 3ª fase



# Instrução beq: 4ª fase



# Incluindo o desvio incondicional



- **Concatenar:**
  - 4 MSBs do PC+4 • 26 bits do campo imediato • 00

# Incluindo suporte para jump

