

Nome e Matricula: _____

Field	0	rs	rt	rd	shamt	funct
Bit positions	31:26	25:21	20:16	15:11	10:6	5:0

a. R-type instruction

Field	35 or 43	rs	rt	address
Bit positions	31:26	25:21	20:16	15:0

b. Load or store instruction

Field	4	rs	rt	address
Bit positions	31:26	25:21	20:16	15:0

c. Branch instruction

FIGURE 4.14 The three instruction classes (R-type, load and store, and branch) use two different instruction formats. The jump instructions use another format, which we will discuss shortly.

1. (6 pontos) Sabendo que o código 35 para Load, 43 para Store, 4 para BEQ, 8 para ADDI, 2 para JUMP, 0 para R-type com func=32 para ADD, SUB (func=34), AND (func=36), OR (func=37). Codifique as instruções abaixo

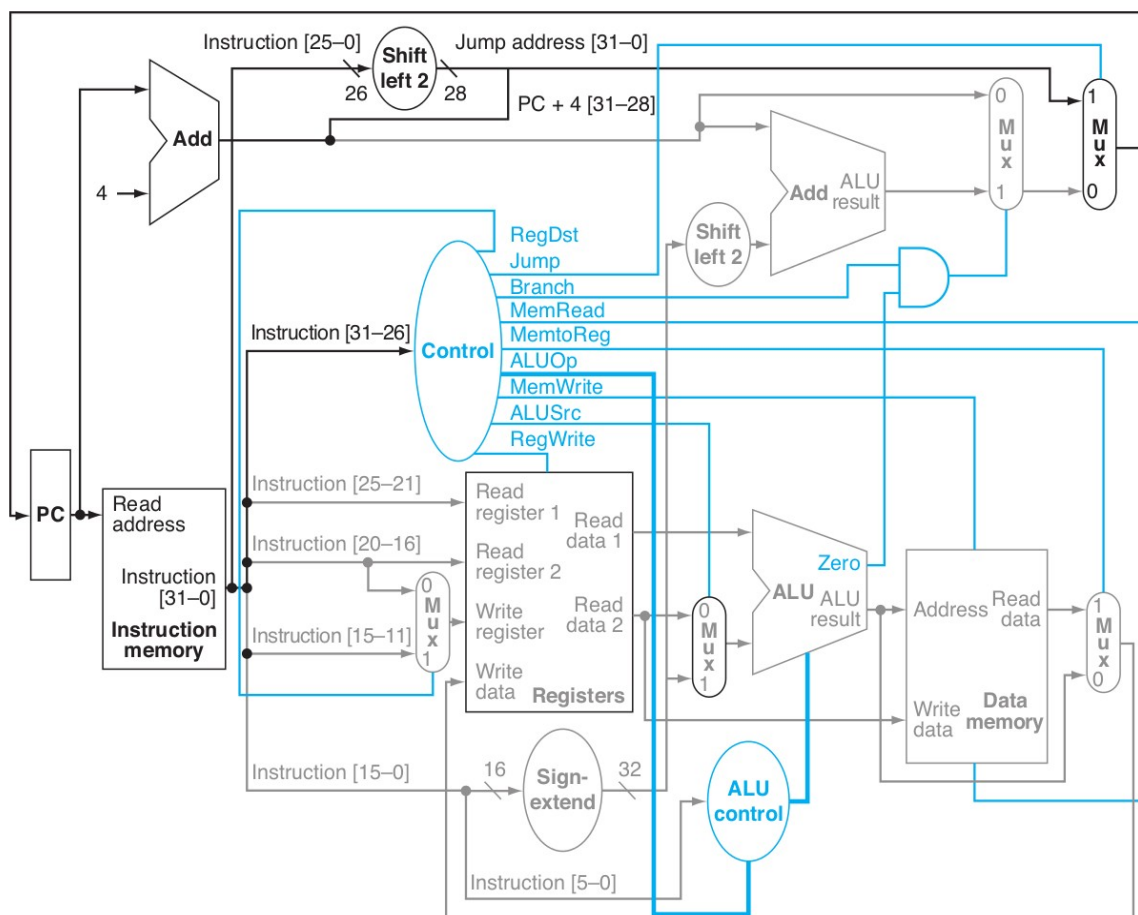
a) ADD R1,R2,R3 =

b) LD R2,8(R3) =

c) SD R3,4(R1) =

2. (6 pontos) Crie uma nova instrução BEQF Rs == Func ; Se falso, PC = PC+4, se verdadeiro PC = PC + 4 + Rt*4. Preencha os valores dos controles e desenhe o caminho e as alterações na figura do MIPS.

opcode	regdst	branch	Mem read	MemtoReg	ALU Op1	ALU Op2	Mem write	ALU src	Reg write	jump	



3) (6 pts) Suponha o código abaixo no MIPS pipeline onde os registros tem o valor inicial $R_i = i$ e a memória(i) = $2 * i$.

```

ADD R1,R2,R2
SUB R2,R8,R1
LD R3,4(R2)
ADD R5,R3,R2
OR R7,R7,R6
AND R2,R4,R7

```

Escreva os valores dos sinais de controle e os valores dos dados no ciclo 6.

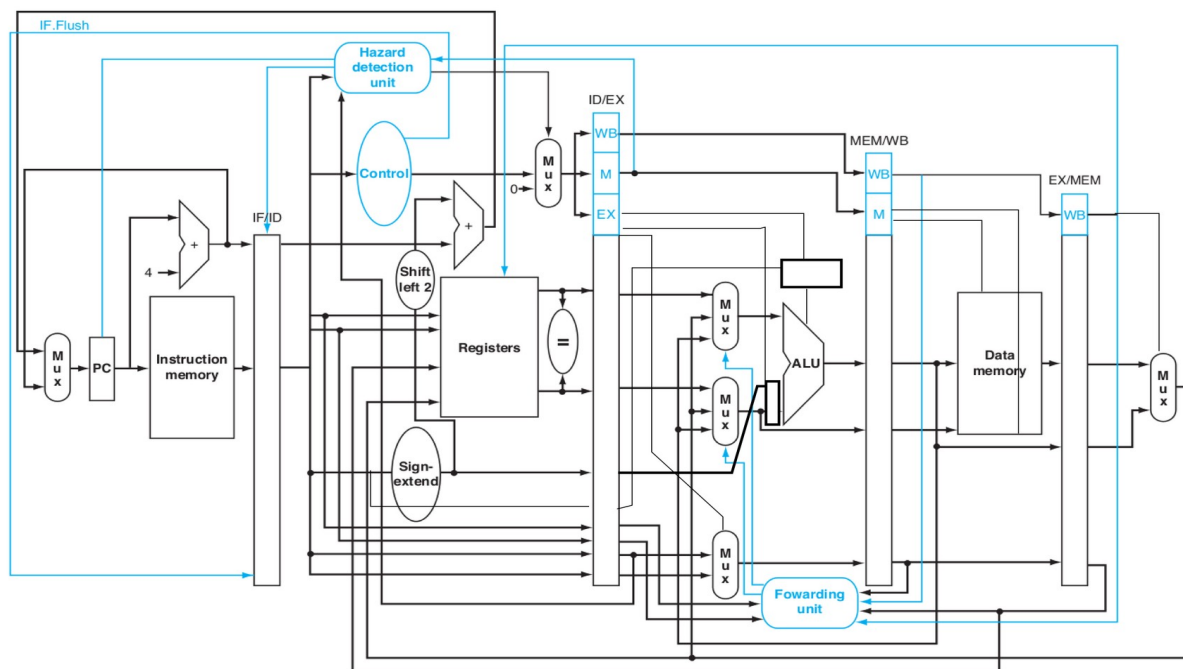


FIGURE 4.65 The final datapath and control for this chapter. Note that this is a stylized figure rather than a detailed datapath, so it's missing the ALUsrc mux from Figure 4.57 and the multiplexor controls from Figure 4.51.

4 (8 pontos) a) De um exemplo que sempre terá uma bolha no caso de dependência de Dados para um LD seguido de um SD. Explique. b) De um exemplo onde você pode modificar o MIPS pipeline para não ter que inserir a bolha. Desenhe e explique os recursos que você adicionou no MIPS abaixo:

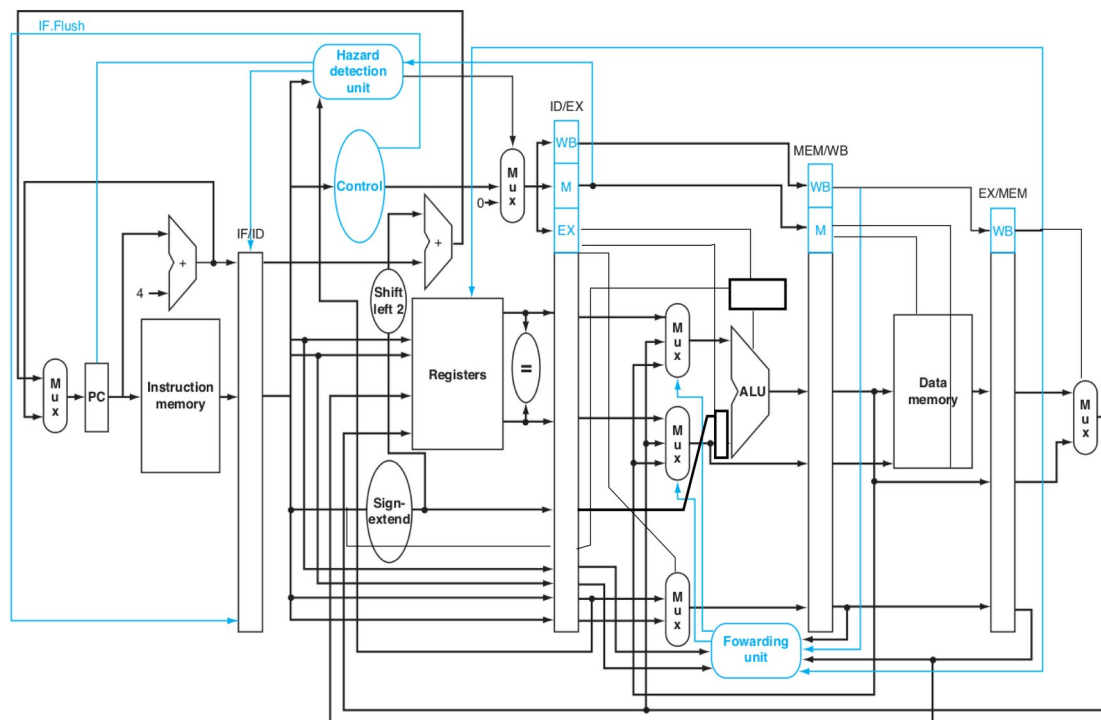


FIGURE 4.65 The final datapath and control for this chapter. Note that this is a stylized figure rather than a detailed datapath, so it's missing the ALUsrc mux from Figure 4.57 and the multiplexor controls from Figure 4.51.

5 (6 pontos) Escreva um programa MIPS para calcular o somatório $m(0) = \sum m(i) * m(i)$, para $i=2$ até $m(1)-1$.