Thank you for the amazing semester!! ☺

Assignment 7 Sarah Mentel

① Exercise 7.3

Modify single-cycle MIPS processor to implement one of the following instruction.

① indicate changes to datapath

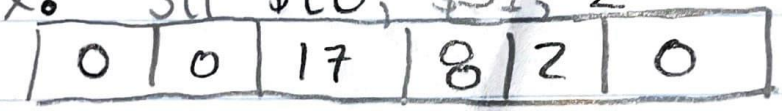
② name any new control signals

③ show changes to main decoder

... describe any other required changes

a.) sll rd, rt, shamt ① attached in file

ex: sll $t0, $s1, 2



R-type

sll shifts $rt w/ shamt, so kinda padded w/ zeros

② Left shift register added, amount specified by shamt

EXERCISE 7.3A

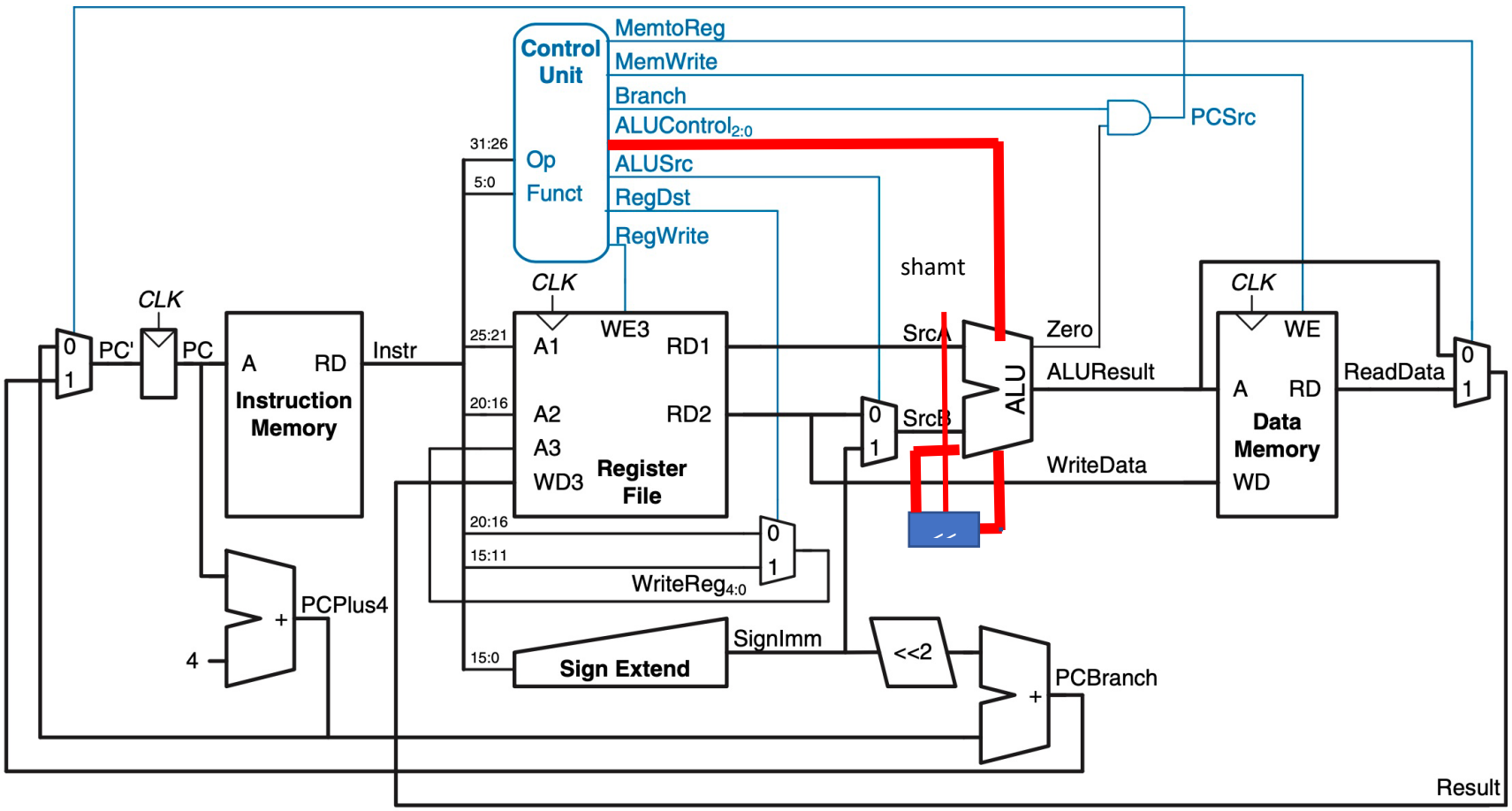


Figure 7.11 Complete single-cycle MIPS processor

b.) lui Ex lui $S, 0($S1)
op rt ↑ rs
imm

① attached pic..

② using appendix B. imm: 16-bit imm field for I instr. (added)

③

Instr.	OP	RegWrite	RegDst	ALUSrc	Branch	MemWrite
lui	001111	1	0	1	0	0

(Stored rt; No rd)

MemtoReg	ALU OP
0	00

EXERCISE7.3B

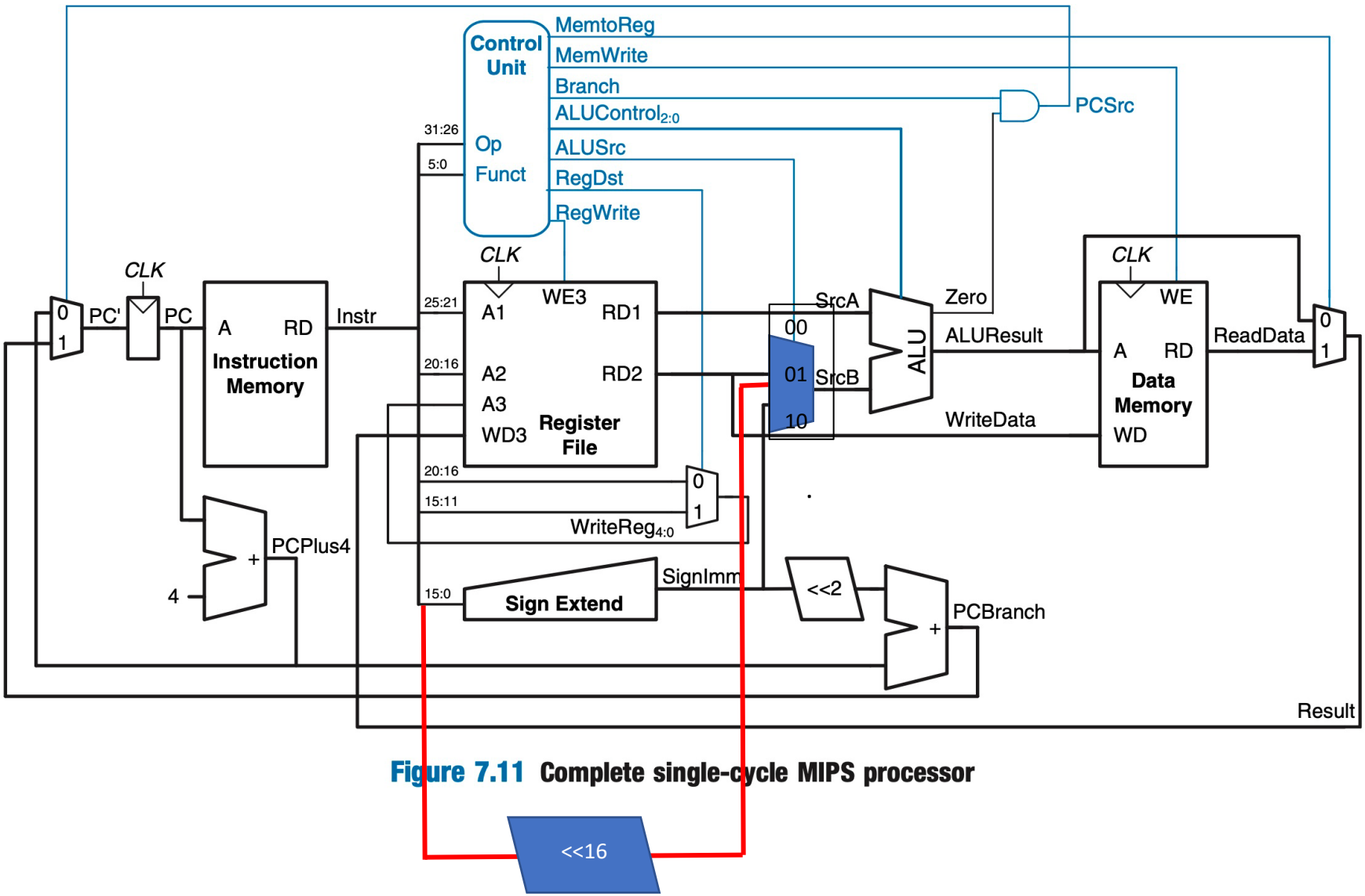


Figure 7.11 Complete single-cycle MIPS processor

Exercise 7.13

b) ori ① image

② Zero extend added ₿multiplexer size increased

ALU control 001

~~Instr OP | RegWrite | RegDst | ALUSrc | Branch | MemtoReg~~

③ Main controller unmodified ~~MemtoReg | ALUOp~~

d) ① data path remains unchanged but Diagram modified attatched pic

Exercise 7.13B

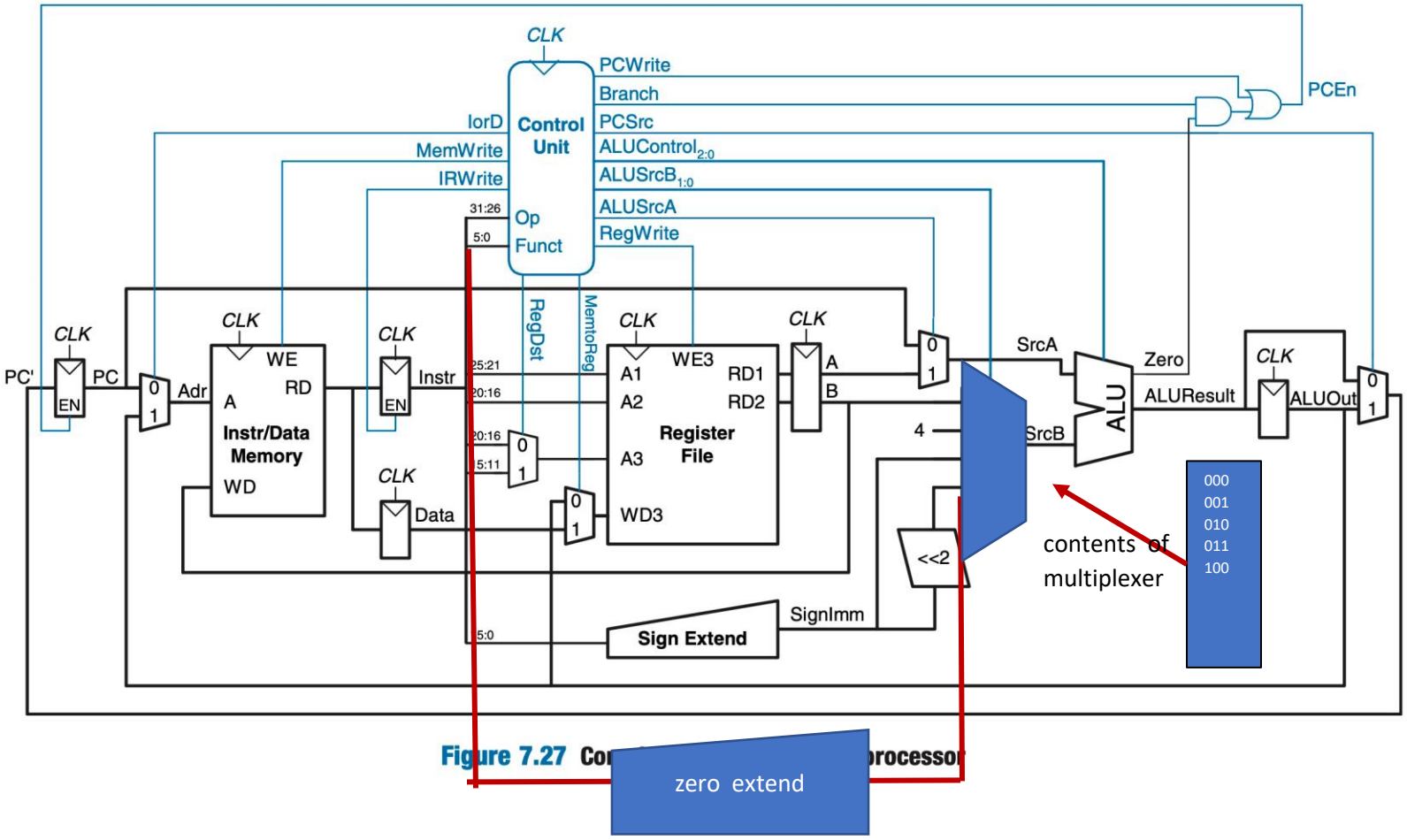
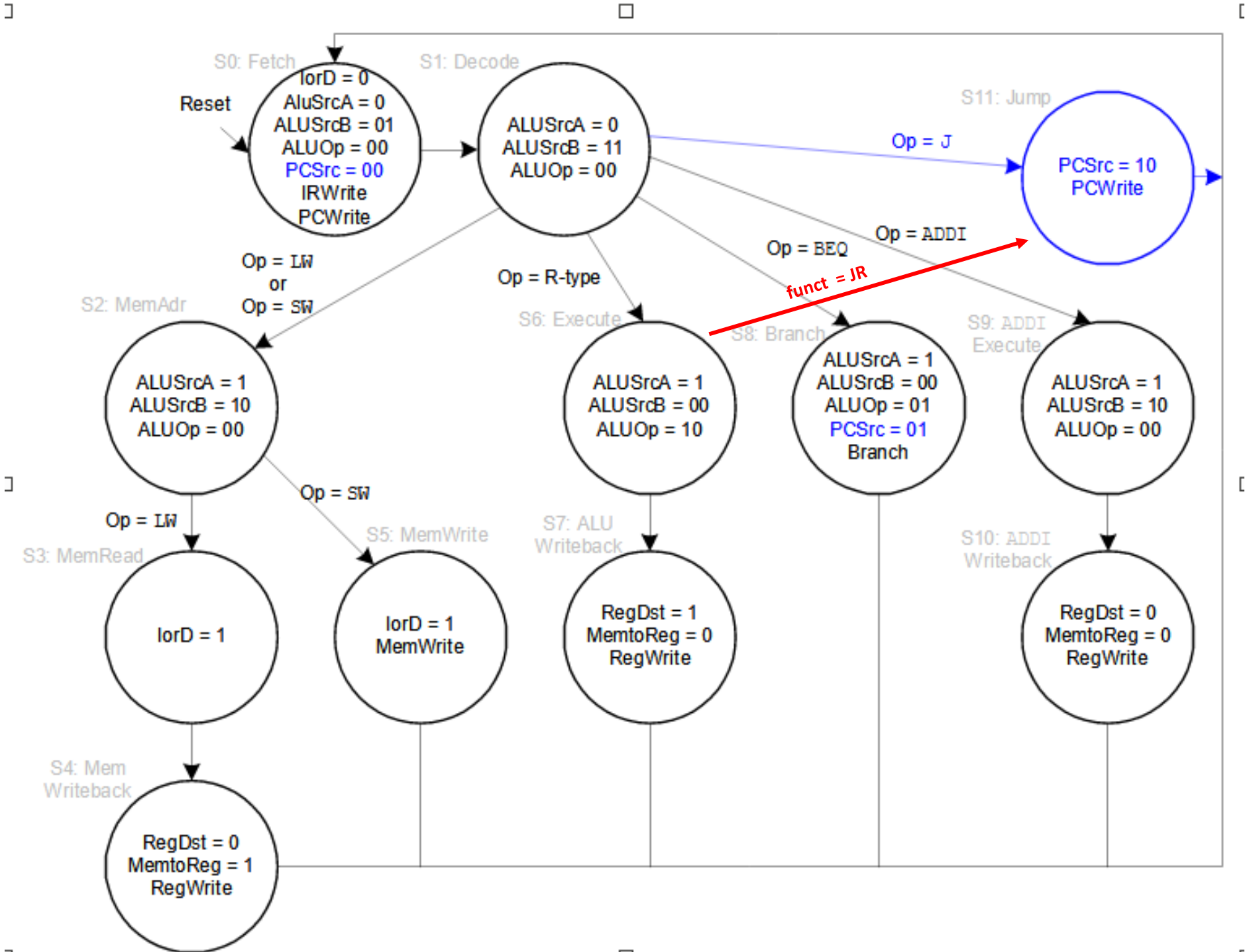


Figure 7.27 Co[illegible]rocesso[illegible]

EXERCISE 7.13 D


S0: Fetch
IorD = 0
AluSrcA = 0
ALUSrcB = 01
ALUOp = 00
PCSrc = 00
IRWrite
PCWrite
Reset
S1: Decode
ALUSrcA = 0
ALUSrcB = 11
ALUOp = 00
S11: Jump
Op = J
PCSrc = 10
PCWrite
Op = ADDI
Op = BEQ
funct = JR
Op = LW
or
Op = SW
Op = R-type
S2: MemAdr
ALUSrcA = 1
ALUSrcB = 10
ALUOp = 00
S6: Execute
ALUSrcA = 1
ALUSrcB = 00
ALUOp = 10
S8: Branch
ALUSrcA = 1
ALUSrcB = 00
ALUOp = 01
PCSrc = 01
Branch
S9: ADDI
Execute
ALUSrcA = 1
ALUSrcB = 10
ALUOp = 00
Op = LW
Op = SW
S3: MemRead
IorD = 1
S5: MemWrite
IorD = 1
MemWrite
S7: ALU
Writeback
RegDst = 1
MemtoReg = 0
RegWrite
S10: ADDI
Writeback
RegDst = 0
MemtoReg = 0
RegWrite
S4: Mem
Writeback
RegDst = 0
MemtoReg = 1
RegWrite