

(ID)

DECODE

F1

F2

EXECUTE

(EX)

F3

MEMORY

(MEM)

WRITEBACK

(WB)

FETCH

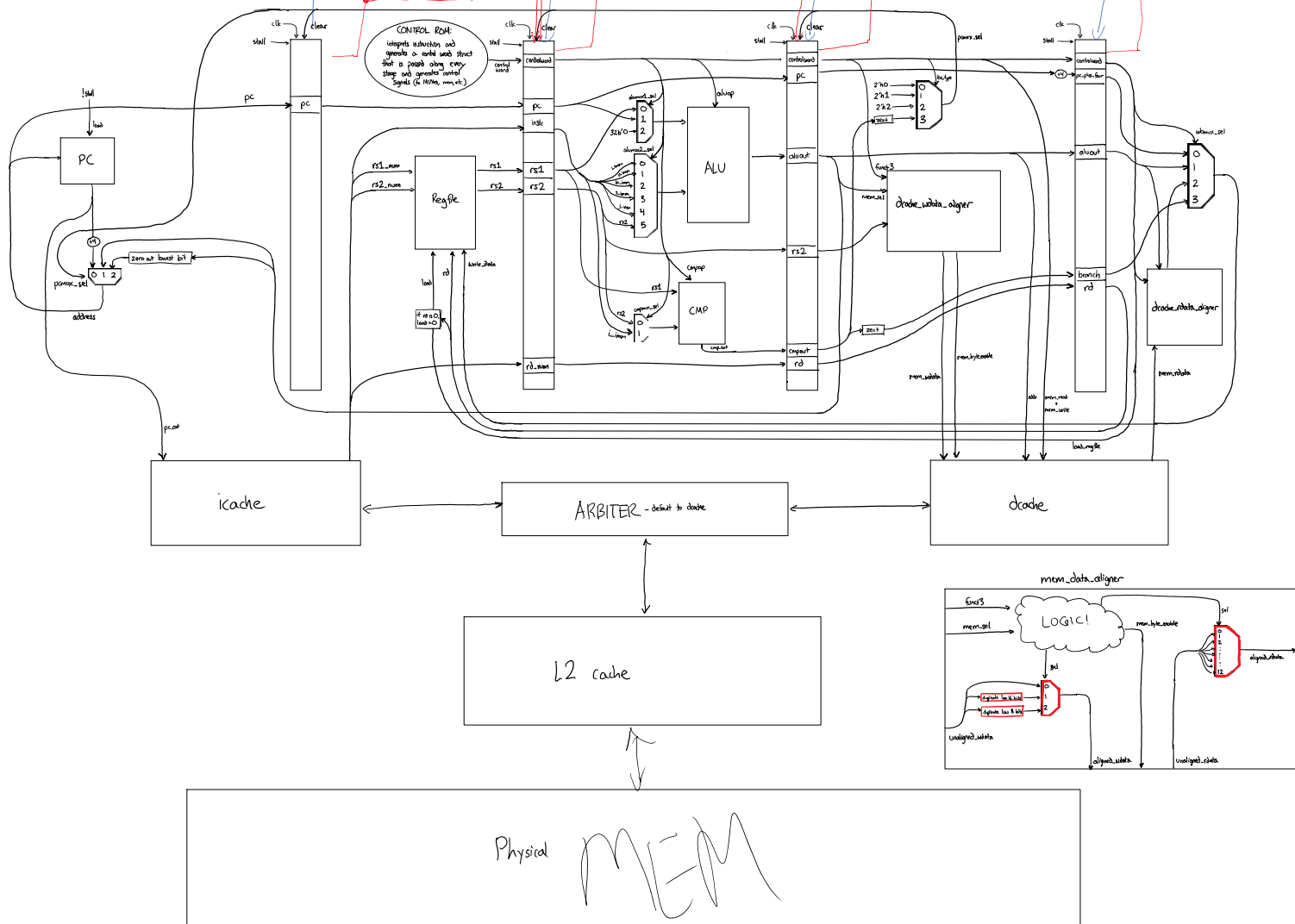
(IF)

StallA

StallB

StallC

StallD



opcodes: ALU, LX, JALR, SX // ALU means any ALU op, LX means any load, SX means any store

$StallA = [(rd == rs1) + (rd == rs2)] \times opcode \times mem\_resp + StallB + StallC + StallD$

$StallB = // + StallC + StallD$

$StallC = // + StallD$

$StallD = //$

$Stall = StallA + StallB + StallC + StallD$

$F1 = WB.ctrl.load\_regfile \ \&\& \ EX.rs == WB.rd \ \&\& \ !MEM.is\_forwarding$   
 $F2 = WB.ctrl.load\_regfile \ \&\& \ MEM.rs == WB.rd$   
 $F3 = MEM.ctrl.load\_regfile \ \&\& \ EX.rs == MEM.rd$

} Done for rs1 and rs2