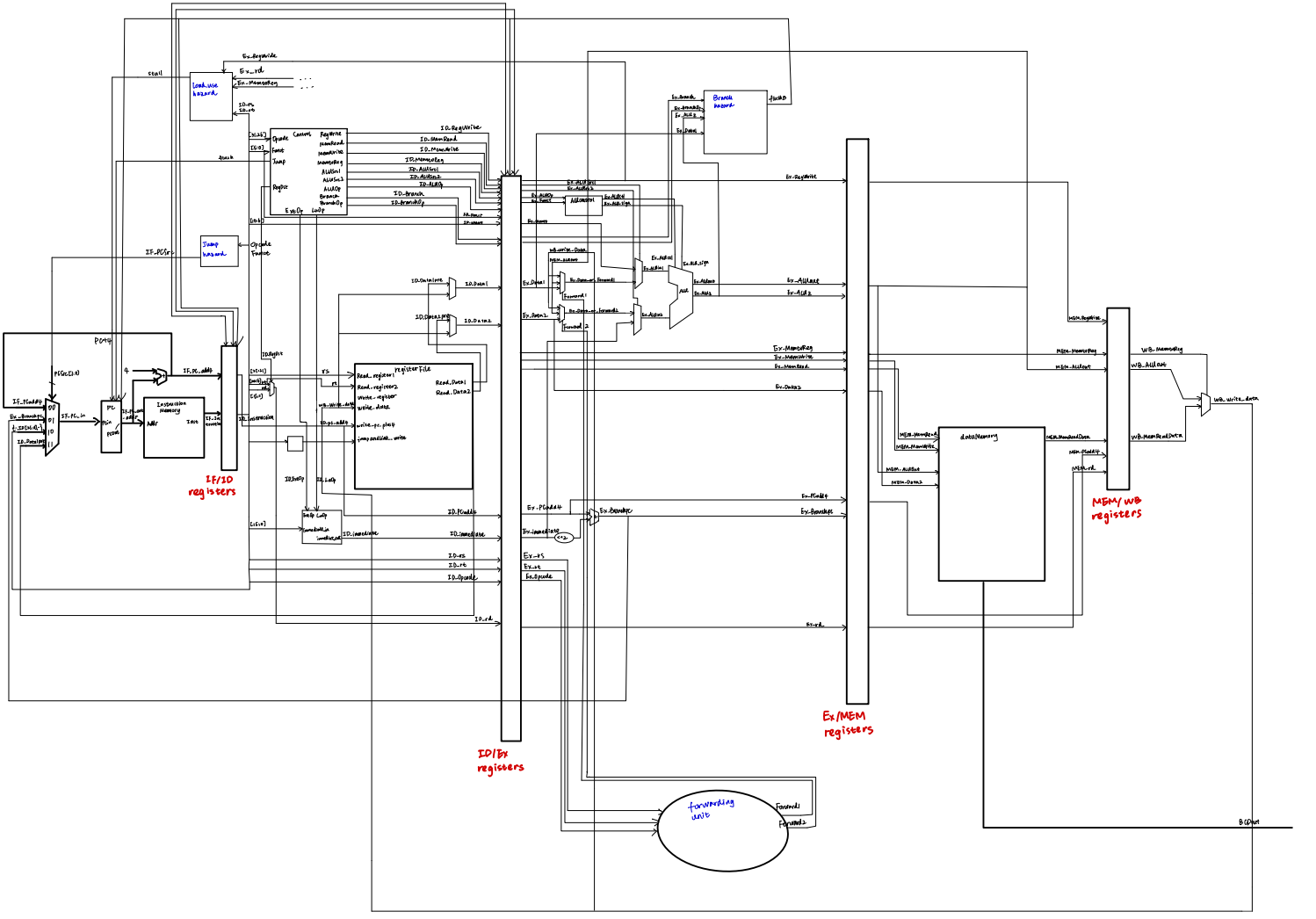
stall
Ex_RegWrite
Ex_rd
Ex_MemRead
ID.rs
ID.rt
Load-use hazard
Control
Opcode
Funct
Jump
Register
Regwrite
Memread
Memwrite
Memtoreg
Alusrc1
Alusrc2
Aluop
Branch
BranchOp
Ex op
Loop
branch
IF_PCSrc
Jump hazard
Opcode
Funct
ID.RegWrite
ID.MemRead
ID.MemWrite
ID.MemtoReg
ID.Alusrc1
ID.Alusrc2
ID.Aluop
ID.Branch
ID.BranchOp
ID.Jump
ID.Loop
Branch hazard
Ex.Branch
Ex.Alu1
Ex.Alu2
Ex.Jump
JumpAD
Ex.RegWrite
Ex.ALUSrc1
Ex.ALUSrc2
Ex.Aluop
Ex.Alusrc1
Ex.Alusrc2
Ex.Jump
WB.write_data
MEM.memdata
Ex.ALUSrc1
Ex.ALUSrc2
Ex.RA.sign
Ex.Alu1
Ex.Alu2
Alu
Ex.Alu1
Ex.Alu2
Ex.Jump1
Forward1
Ex.Jump2
Forward2
ID.Jump1
ID.Jump2
Ex.Alu1
Ex.Alu2
Ex.MemRead
Ex.MemWrite
Ex.Memtoreg
Ex.Jump
Ex.Branch
Ex.rd
MEM.RegWrite
MEM.MemtoReg
MEM.Alu.out
WB.memtoreg
WB.Alu.out
WB.MemReadData
WB.write_data
MEM.MemRead
MEM.MemWrite
MEM.Alu.out
MEM.Data2
MEM.MemReadData
MEM.MemWrite
MEM.rd
P(31:4)
P(3:2)
P(1:0)
4
IF.PC.add4
IF.Instruction
IF.PC.in
Ex.Branch
J.IDEX.Q
ID.Jump
IF.PC.add4
PC
Mux
PCOut
Instruction Memory
Addr
Instr
IF/ID registers
ID.Instruction
ID.Rs
ID.Rt
ID.Rd
ID.Opcode
ID.Shamt
ID.Funct
ID.Imm
ID.Jump
ID.Branch
ID.Loop
Register File
Read_register1
Read_register2
Write_register
Write_data
Read_Data1
Read_Data2
WB.Write.data
ID.Rd
ID.Rt
ID.Rs
Sign Loop
Zero Loop
ID.Immediate
ID.Rt
ID.Rd
ID.Rs
ID.Opcode
ID.Rd
ID/Ex registers
Ex.Branch
Ex.Immediate
<<2
Ex.rs
Ex.rt
Ex.Opcode
Ex.rd
Ex/MEM registers
dataMemory
MEM.MemRead
MEM.MemWrite
MEM.Alu.out
MEM.Data2
MEM.MemReadData
MEM.RegWrite
MEM.MemtoReg
MEM.Alu.out
MEM.rd
MEM/WB registers
WB.memtoreg
WB.Alu.out
WB.MemReadData
WB.write_data
forwarding unit
Forward1
Forward2
B(Out)