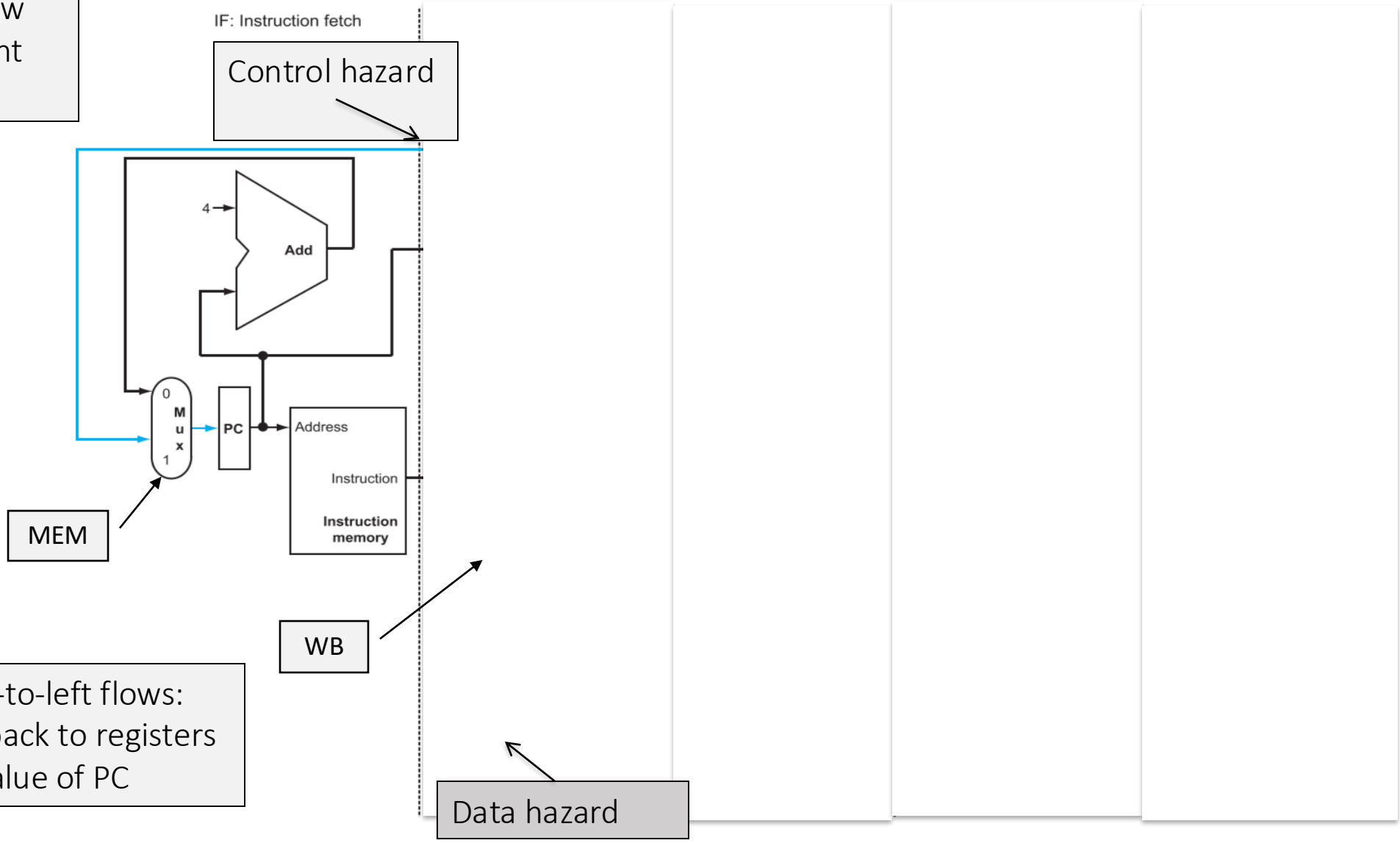
Topic V2B

Pipelined Datapath

Readings: (Sections 4.6-4.7)

RISC-V Pipelined Datapath

Signals flow left-to-right



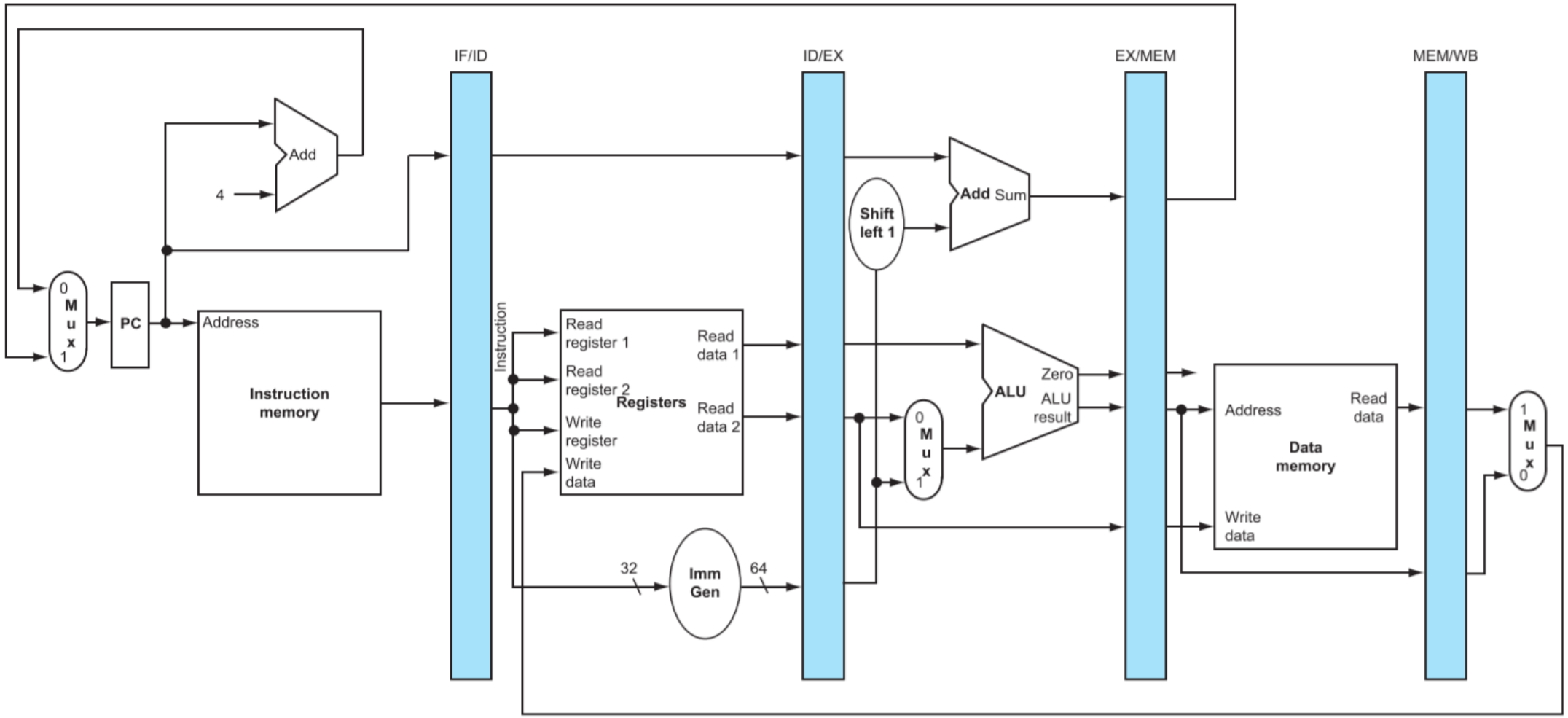
Two right-to-left flows:

- write back to registers
- next value of PC

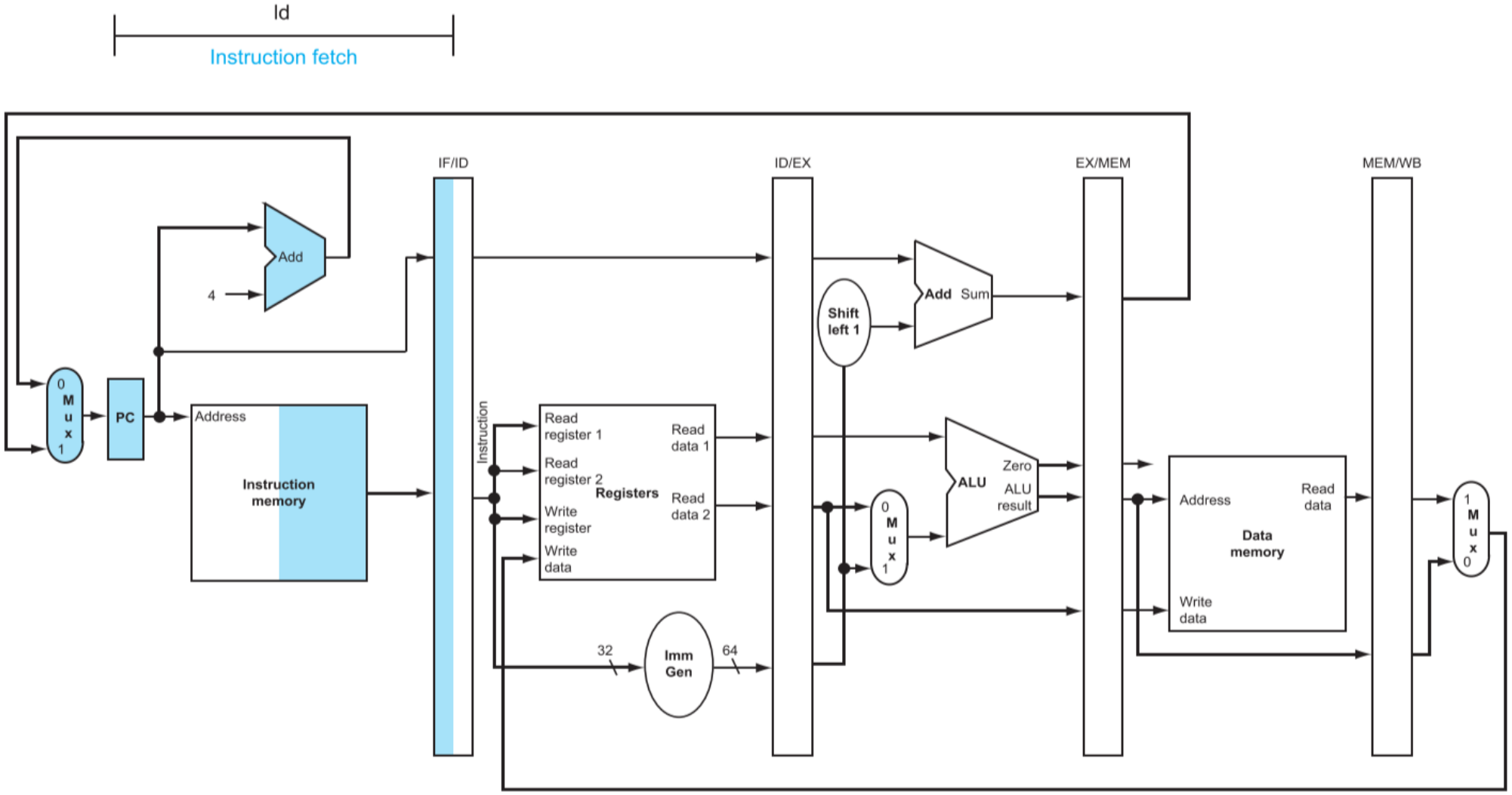
Pipeline registers

Need registers between stages

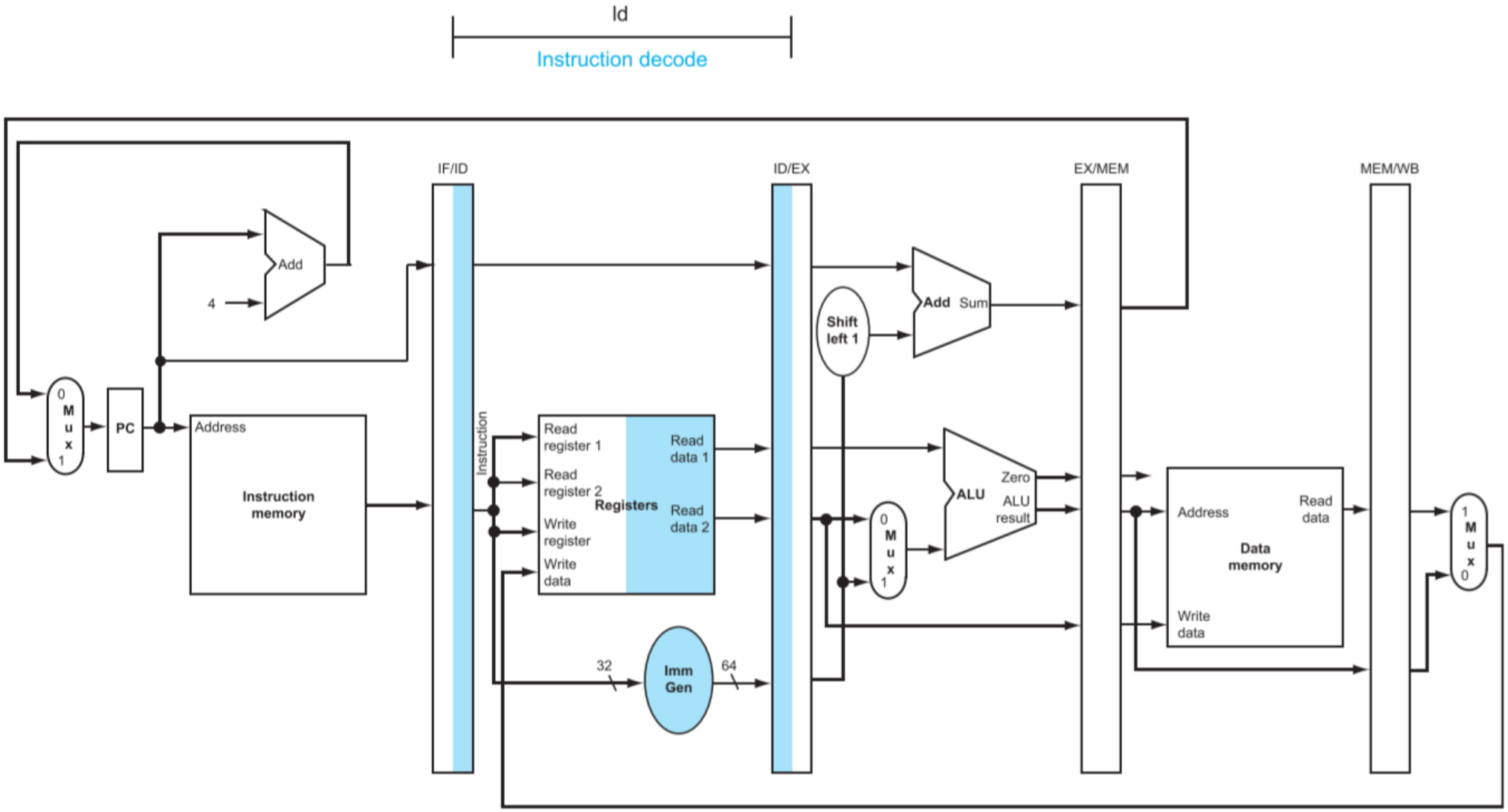
To hold information produced in previous cycle



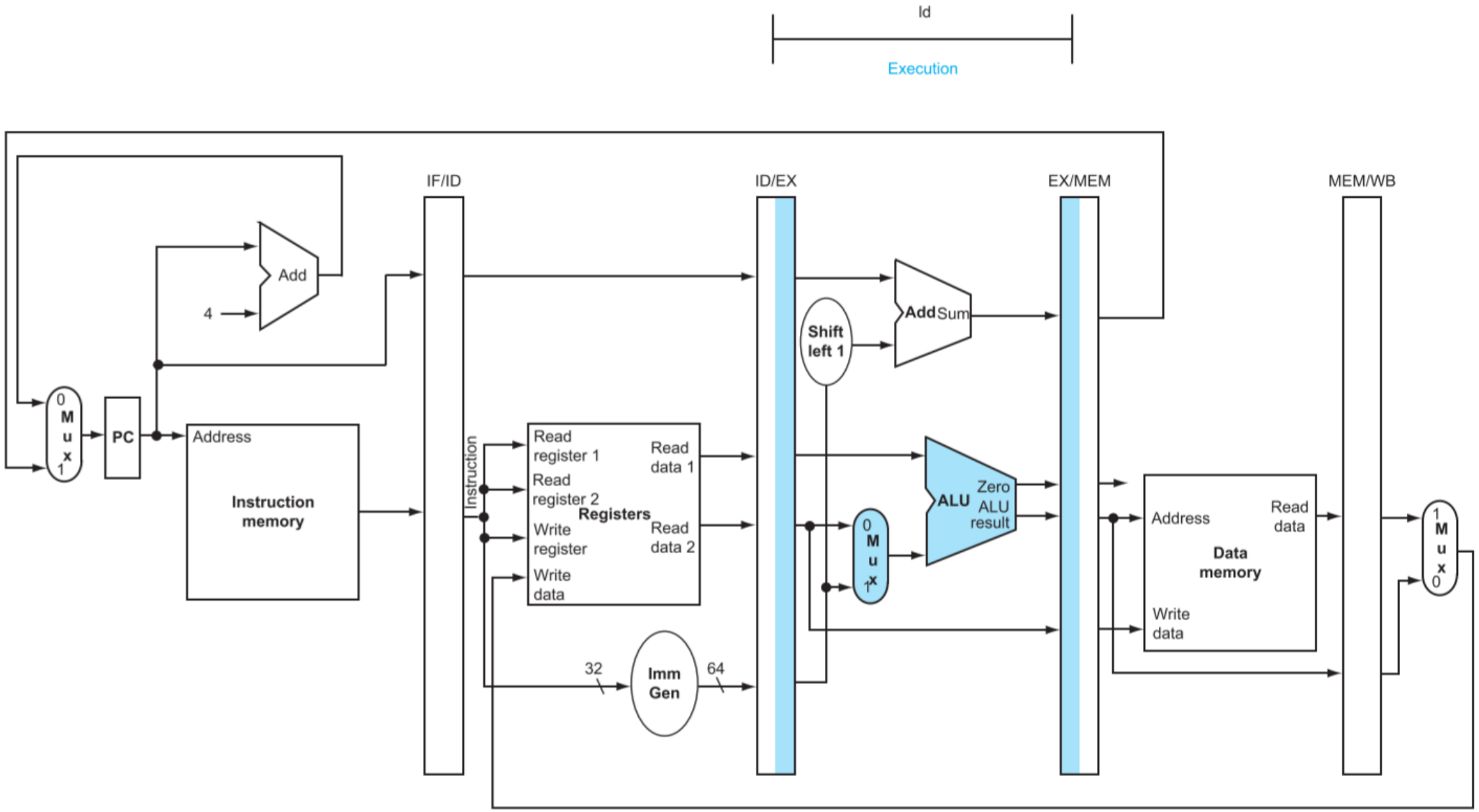
Instr. Fetch (IF) for Load, Store, …



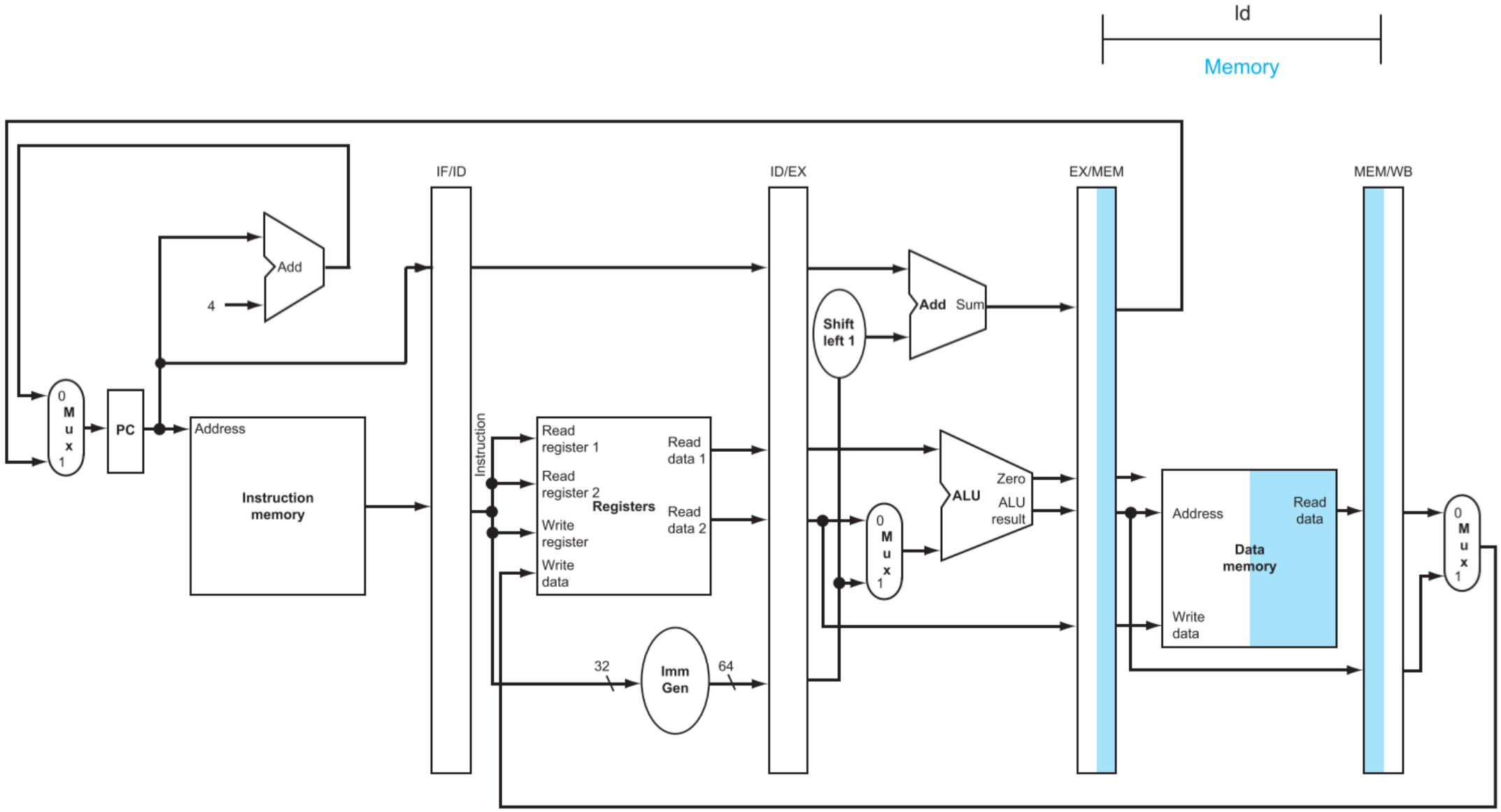
Instr. Decode (ID) for Load, Store, …



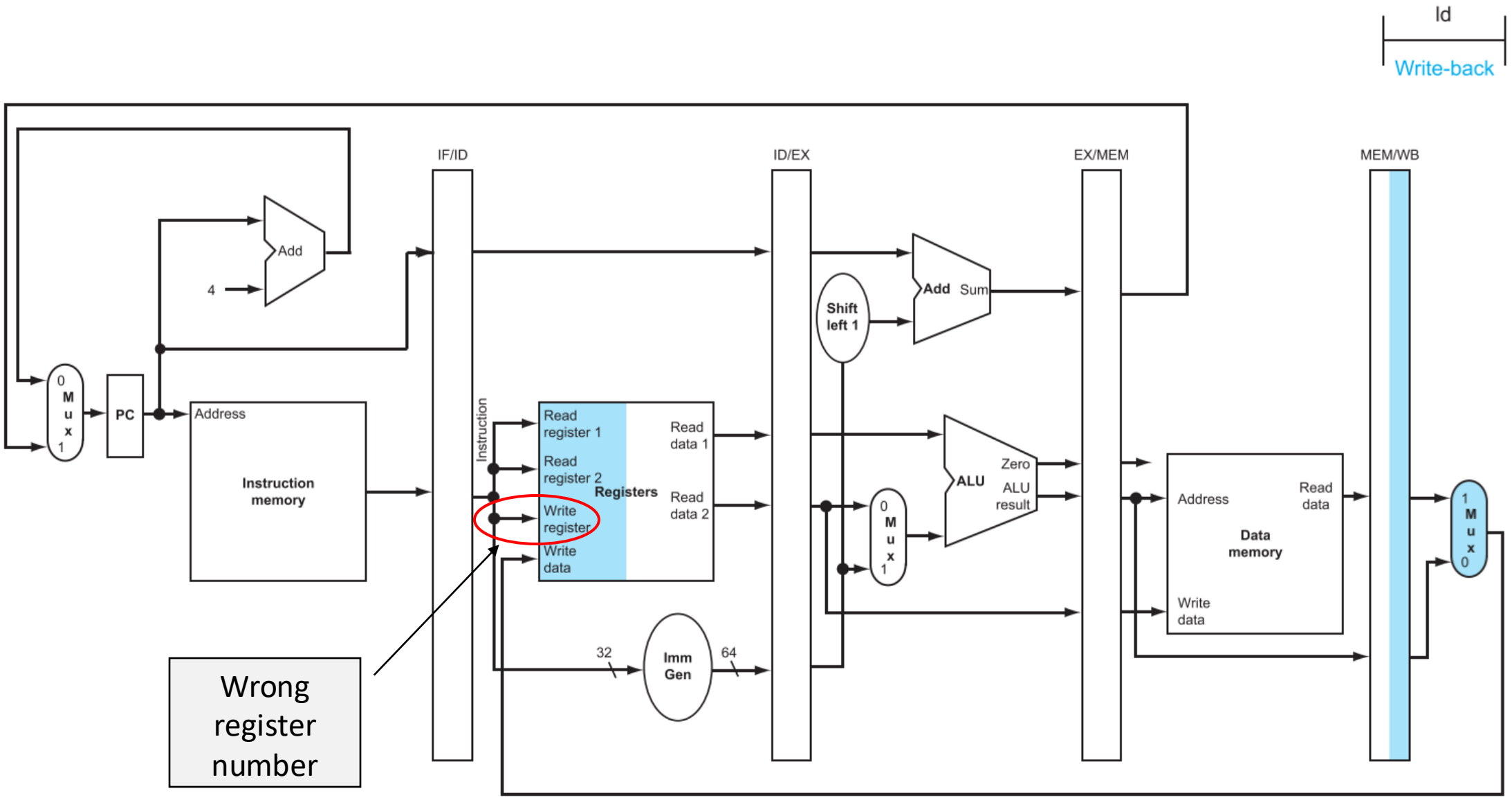
Execution (EX) for Load



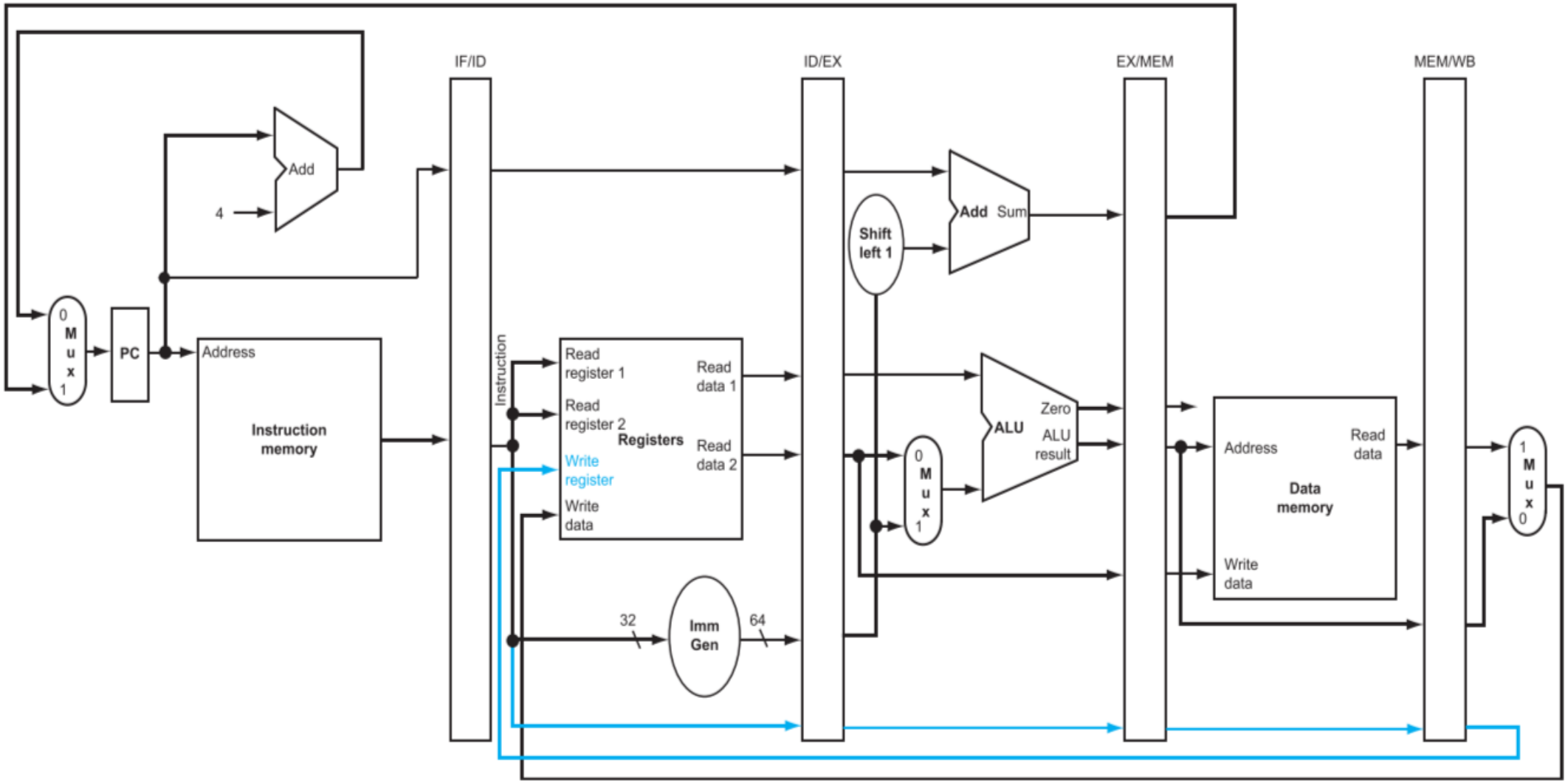
Memory (MEM) for Load



Write Back (WB) for Load

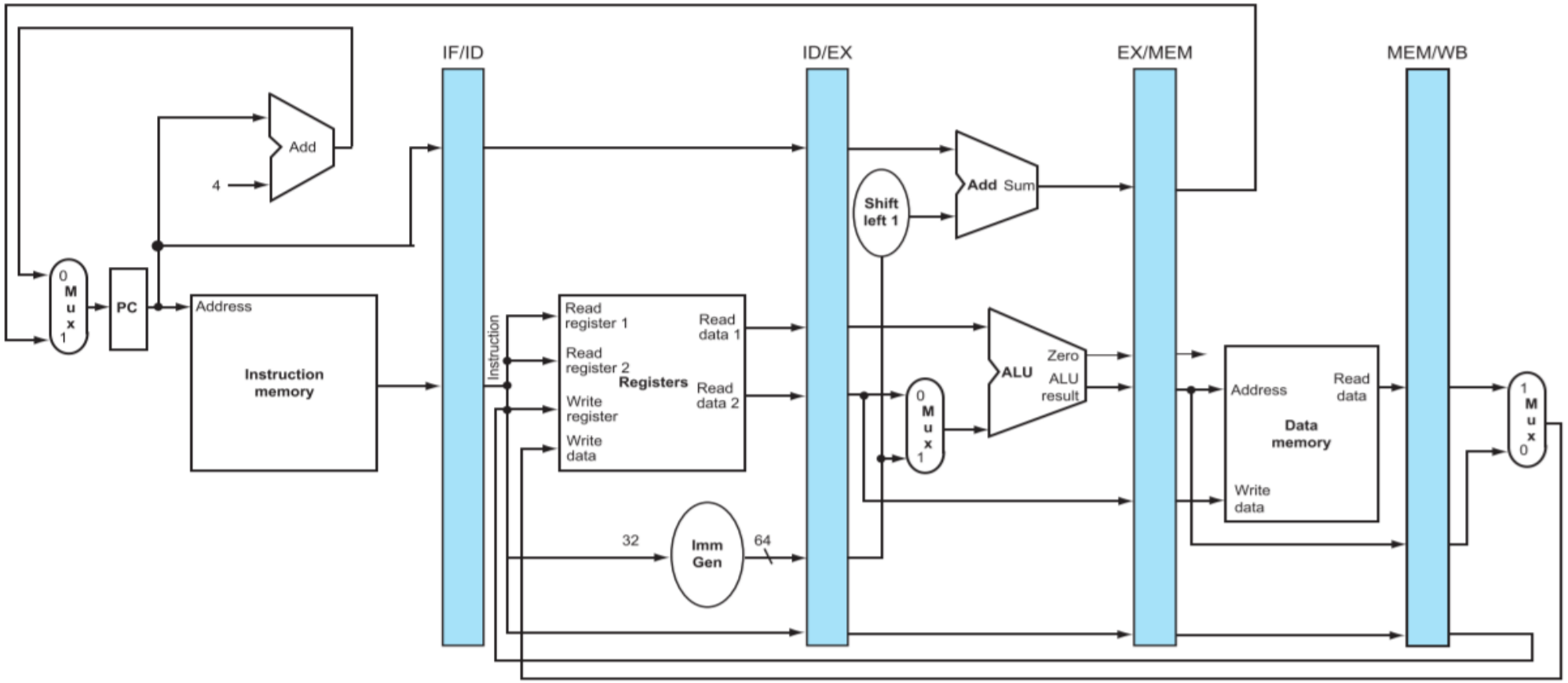


Corrected Datapath for Load

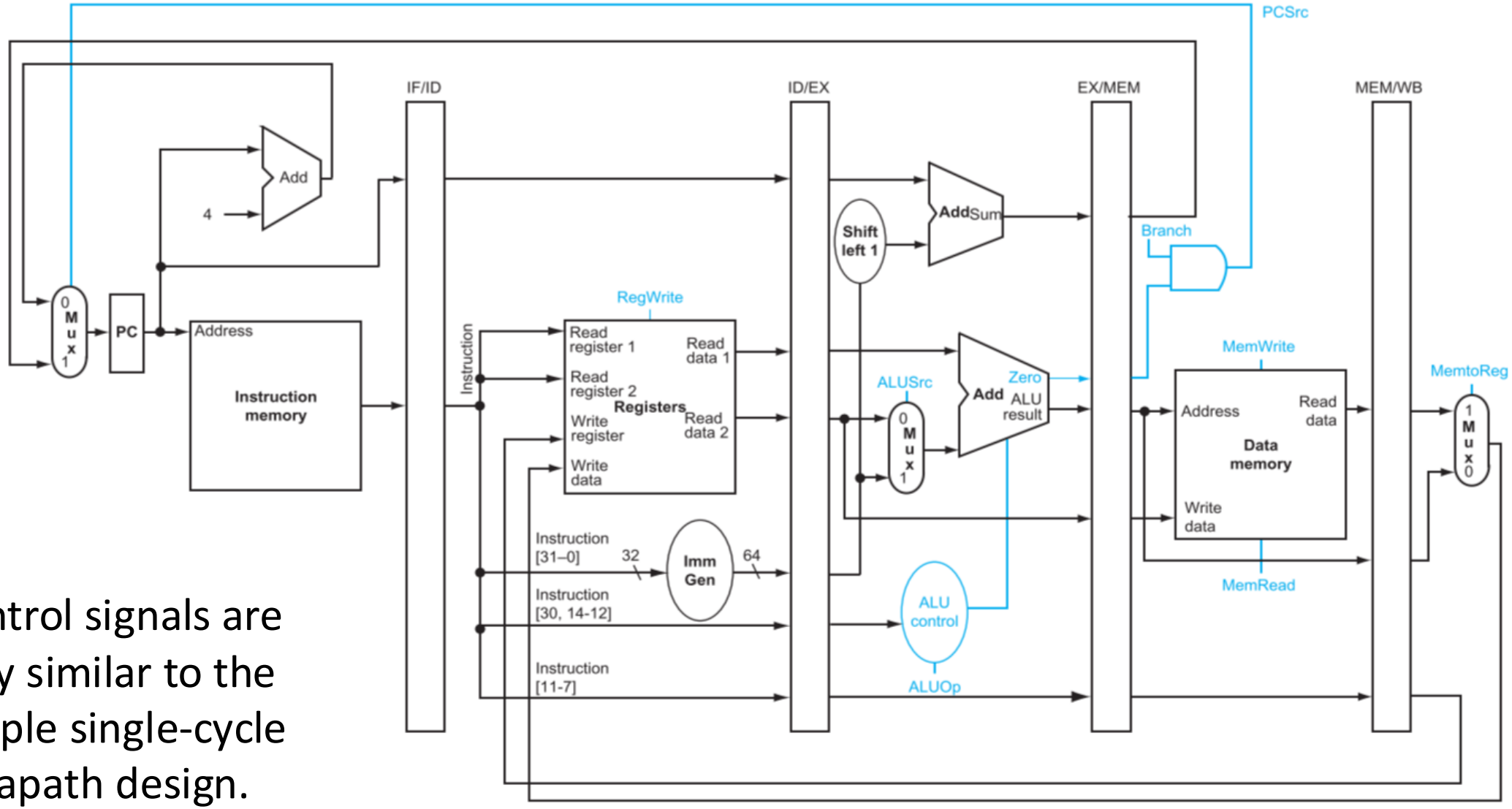


Snapshot of Pipeline State in a given Cycle

```
ld  x10, 40($1)
sub x11, x2, x3
add x12, x3, x4
ld  x13, 48(x1)
add x14, x5, x6
```



Pipelined Control (Simplified)

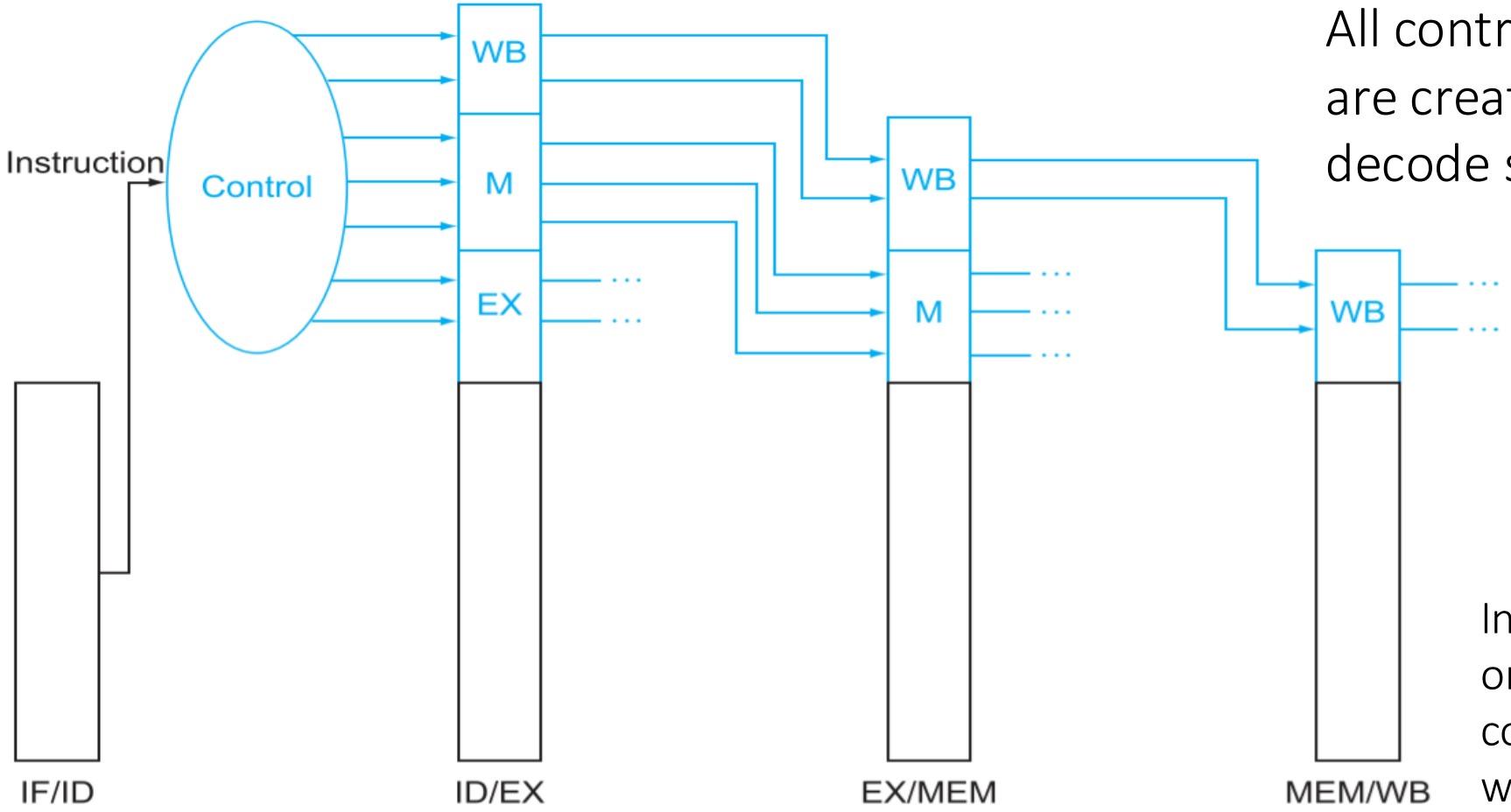


Control signals are very similar to the simple single-cycle datapath design.

Pipelined Control

Control signals derived from instruction

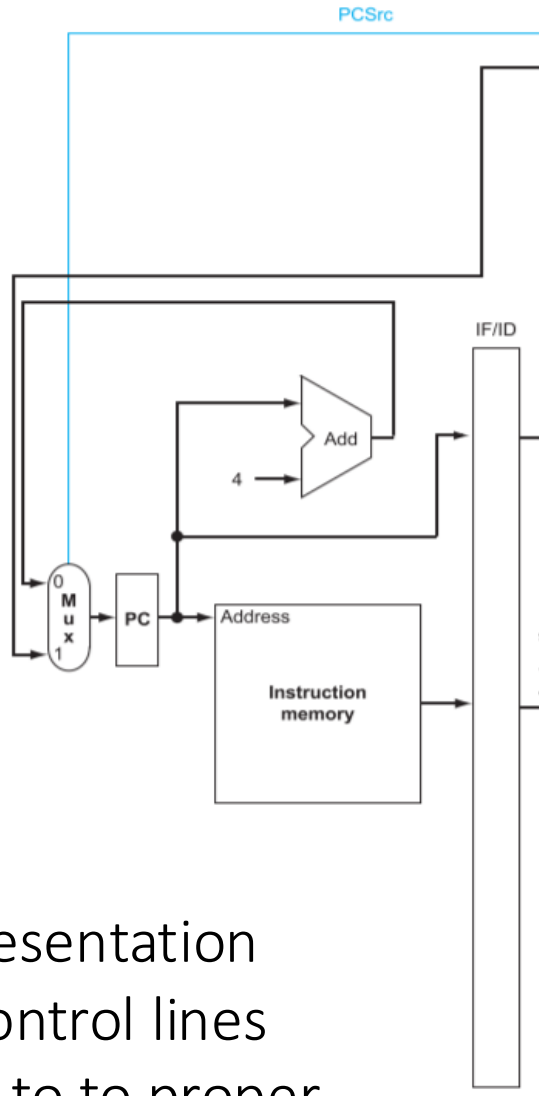
As in single-cycle implementation



All control signals are created in the decode stage.

Inter-stage registers only need to store control signals that will be used later.

Pipelined Control



Same representation but with control lines connected to to proper stages.

Data Hazards in ALU Instructions

```
sub   x2,   x1, x3
and   x12, x2, x5
or    x13, x6, x2
and   x14, x2, x2
sw    x15, 200(x2)
```

sub

and

or

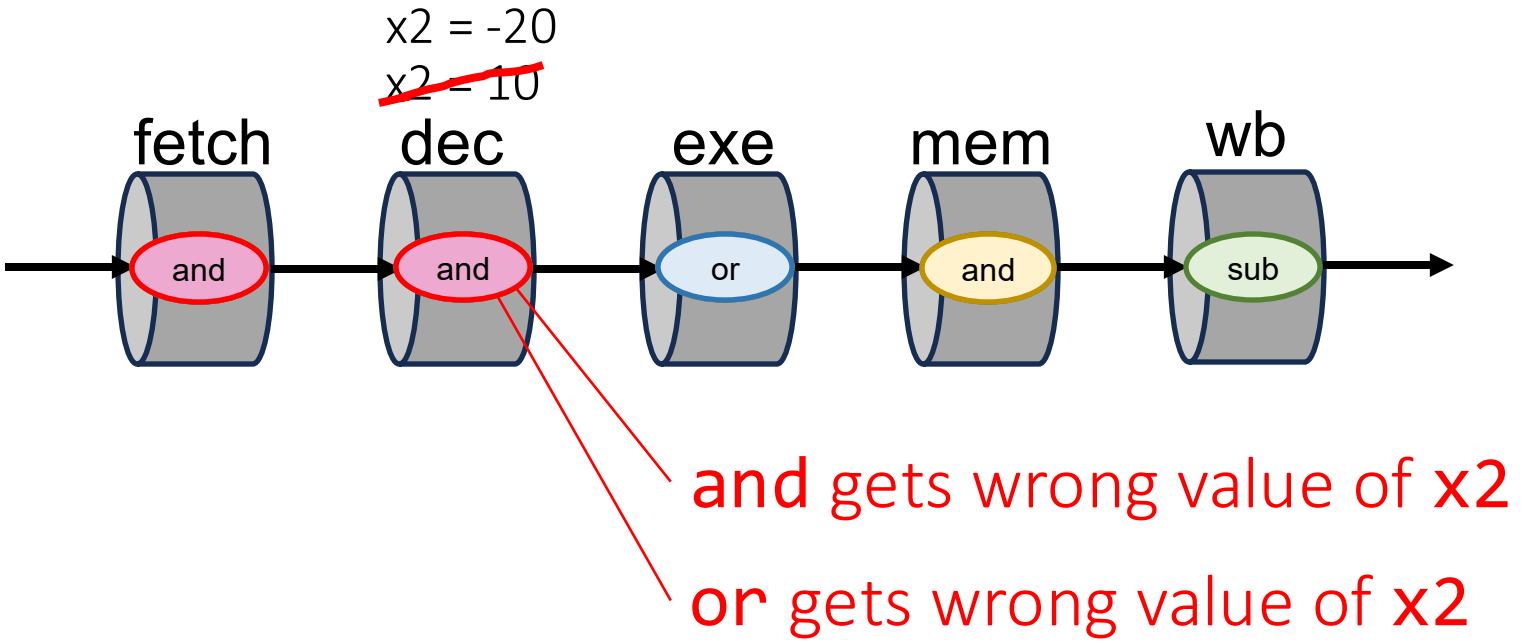
and

sw

x2 = 10 here

x2 = -20 here

Without Forwarding



```
sub  x2,  x1, x3
and  x12, x2, x5
or   x13, x6, x2
and  x14, x2, x2
sw   x15, 200(x2)
```

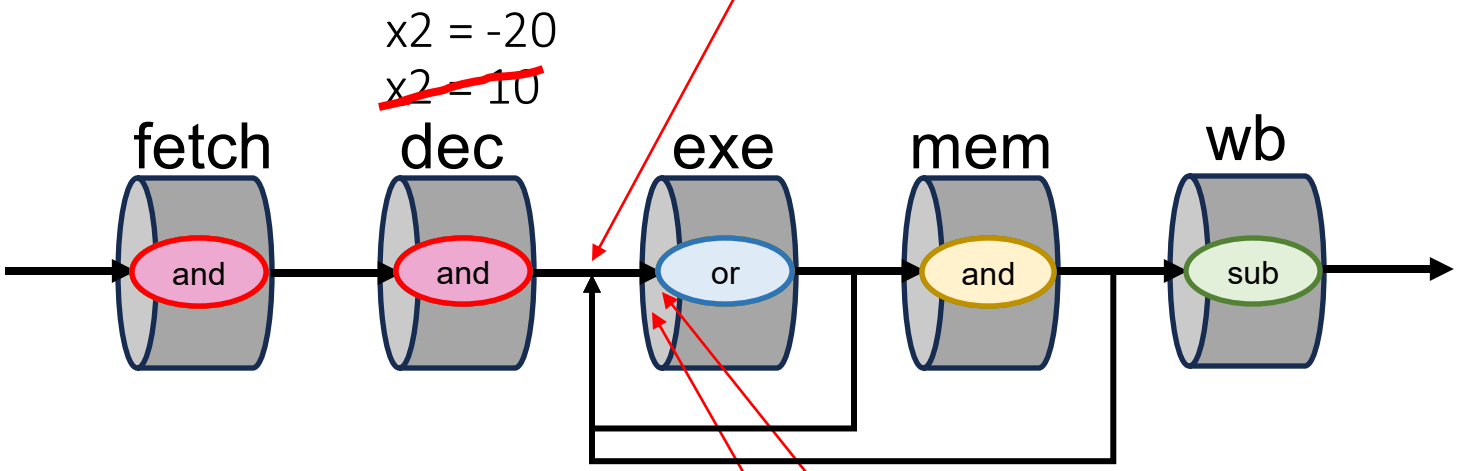
sub
and
or
and
sw

x2 = 10 here

x2 = -20 here

With Forwarding

Where forwarded value comes from?



and gets correct value of **x2**

or gets correct value of **x2**

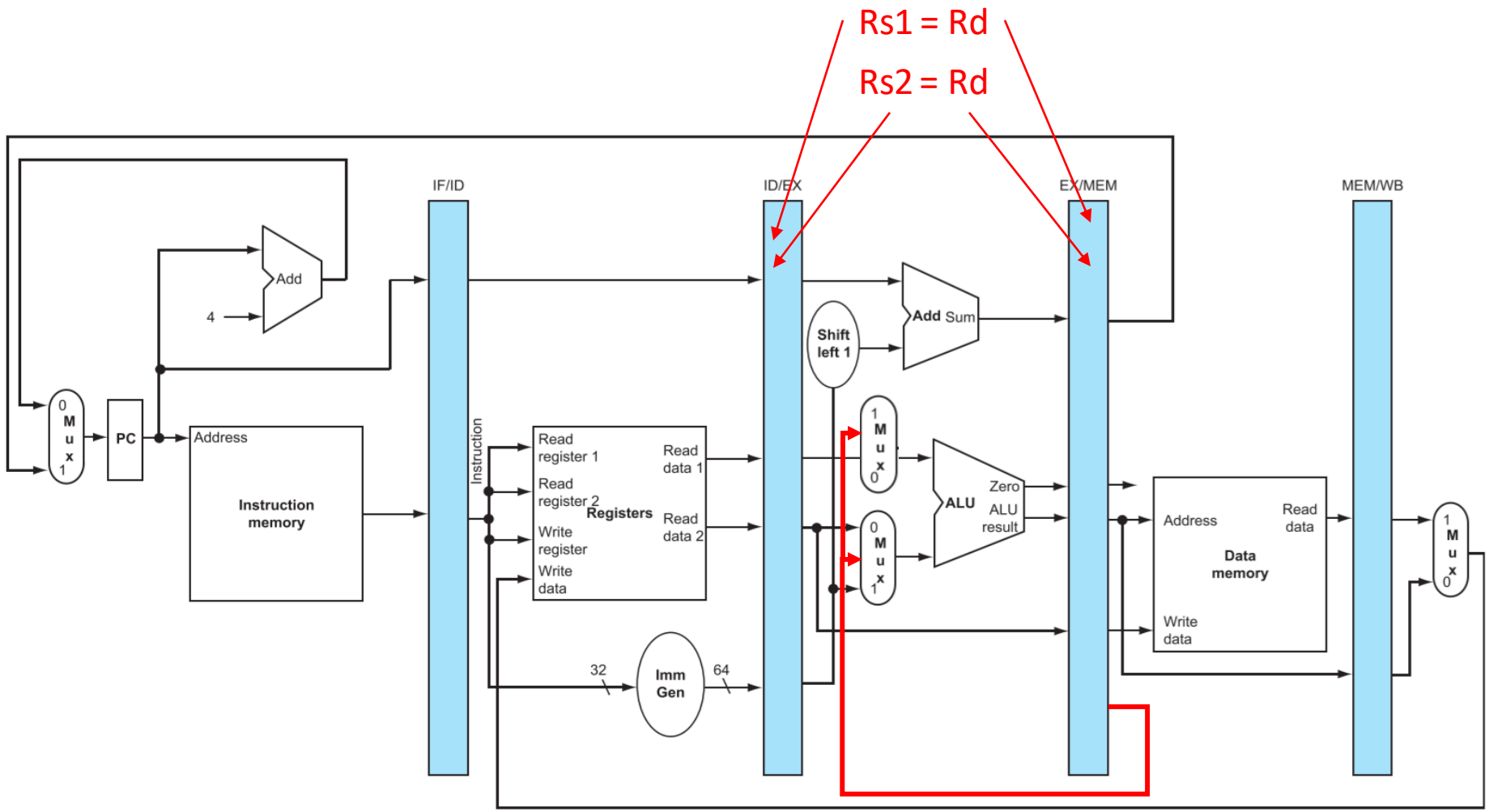
RISC-V Forwarding Unit Logic

How RISC-V detects that forwarding is needed?

Example: forwarding to EX stage

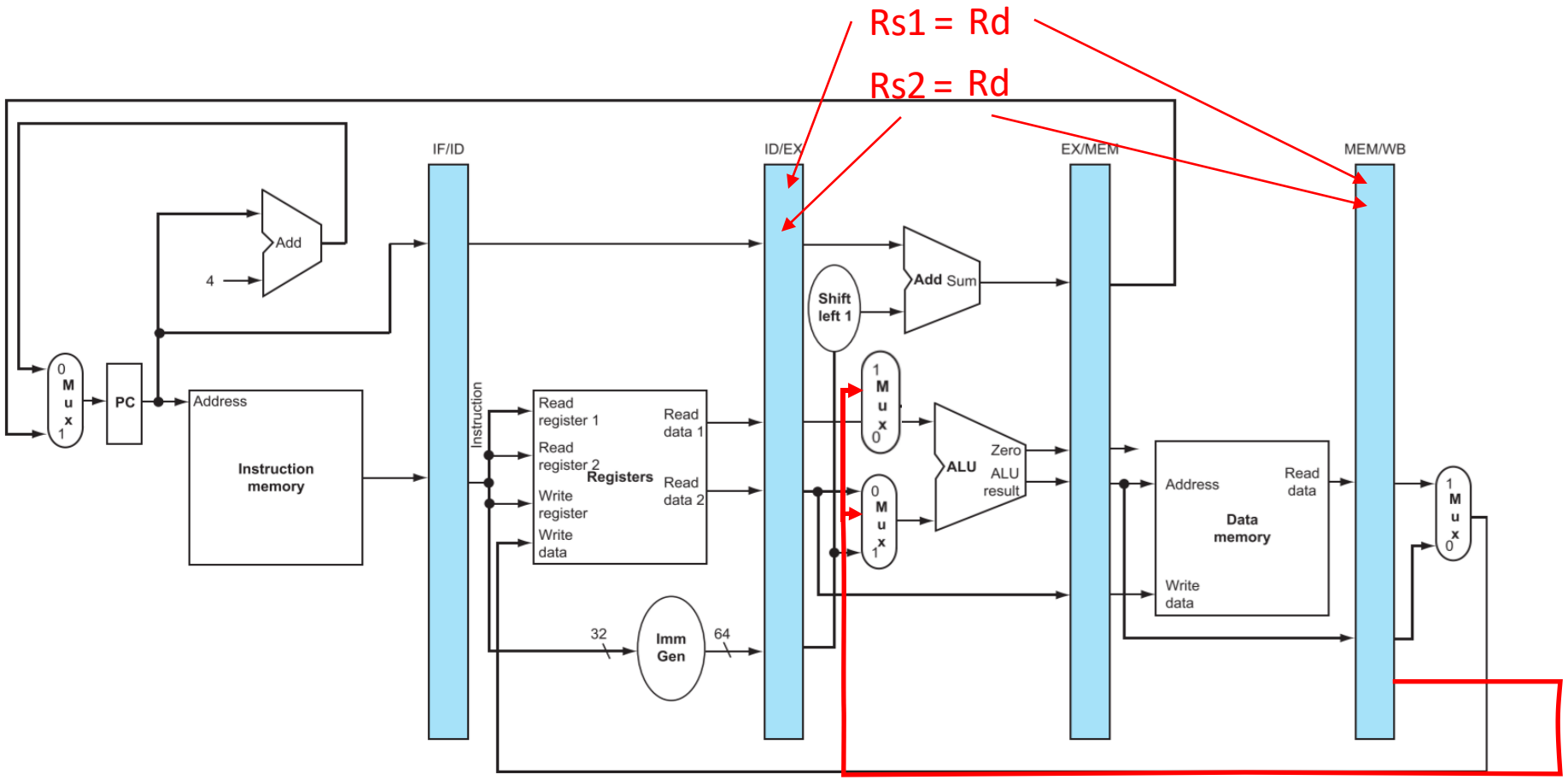
Only need to forward if instruction in EX writes to a register.

And only if Rd ≠ x0

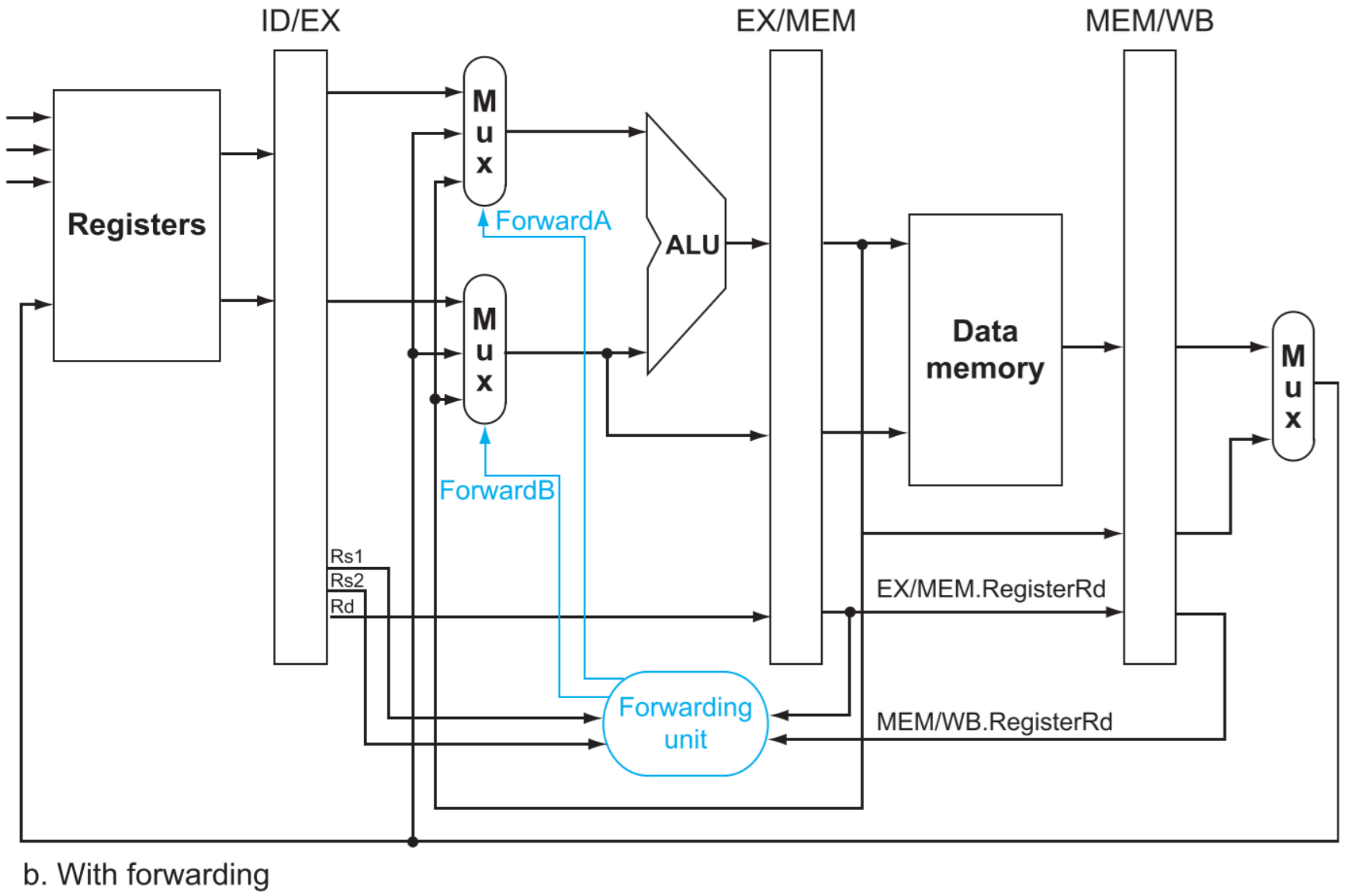


Only need to forward if instruction in MEM writes to a register.

And only if Rd ≠ x0

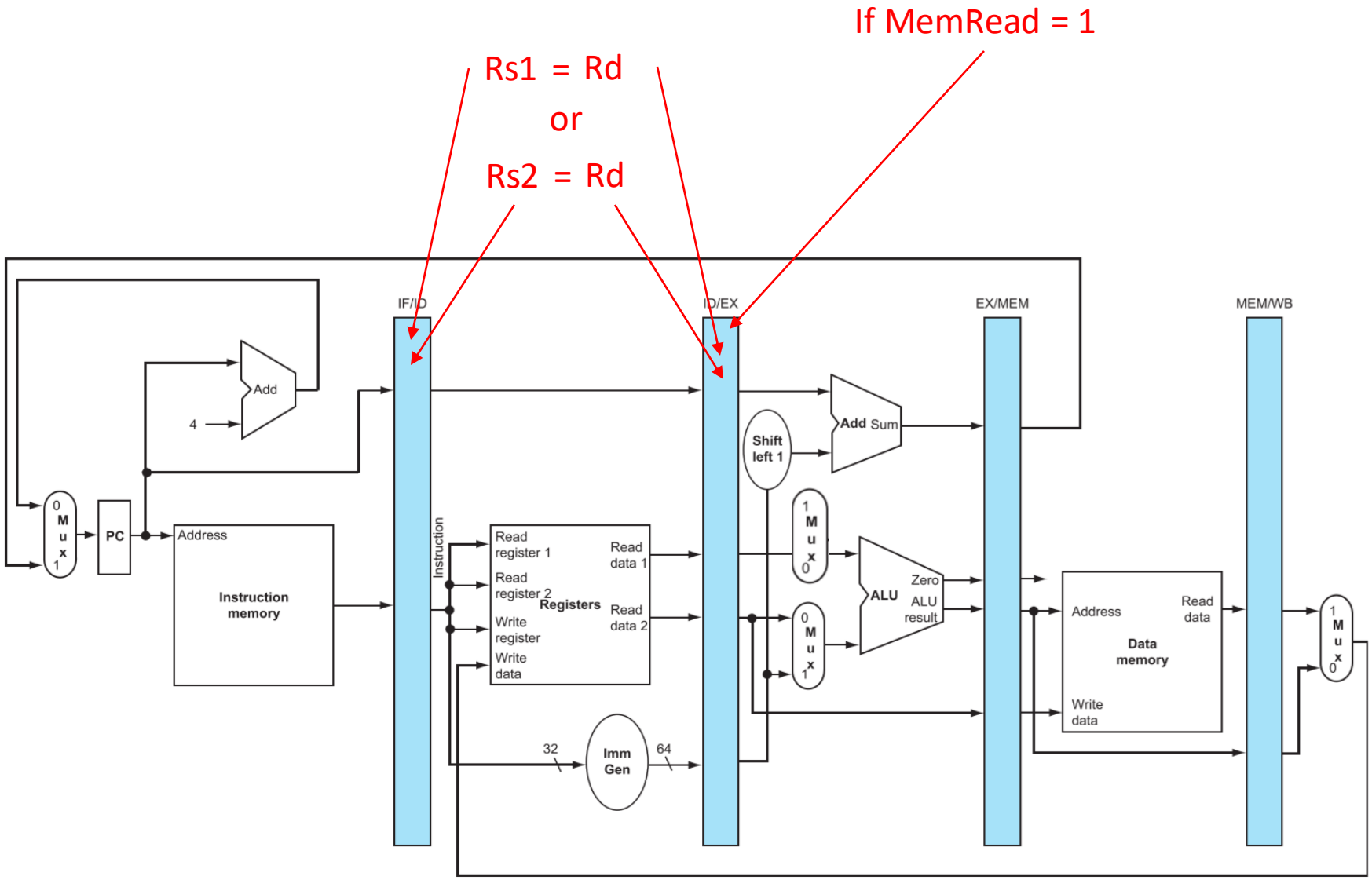


Forwarding Paths



b. With forwarding

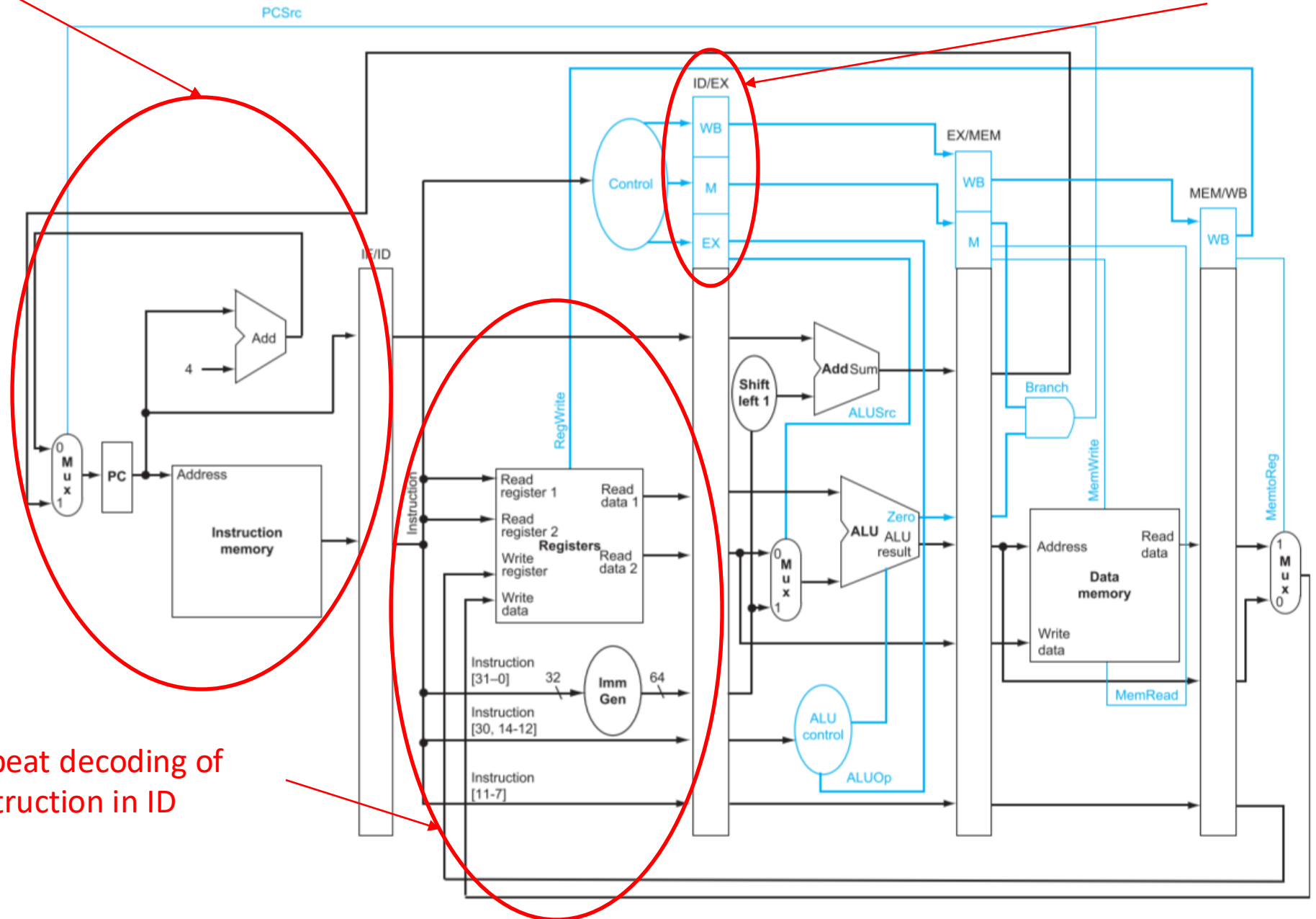
When a load/use stall is needed?



How to Stall the Pipeline?

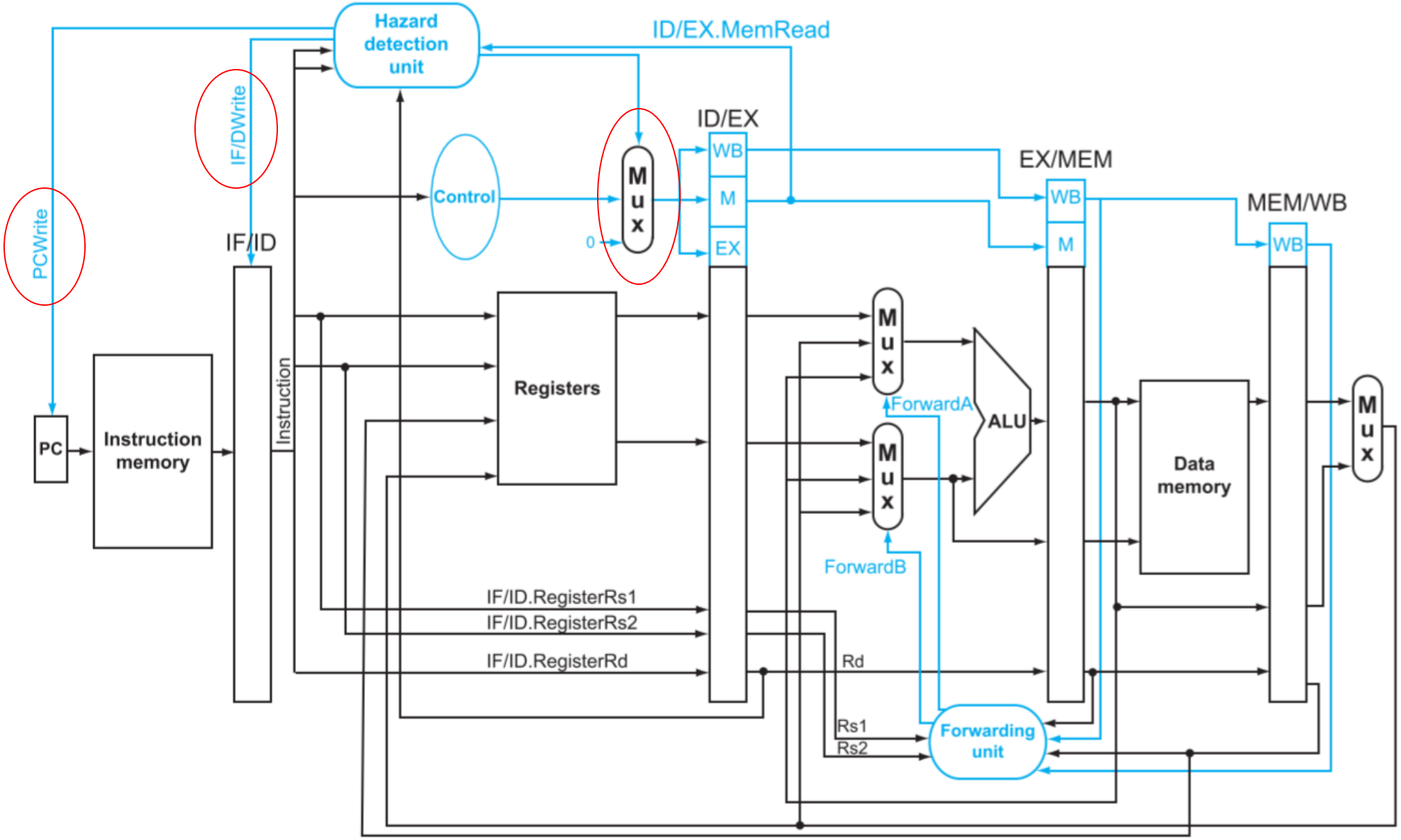
Change PC to fetch the instruction in IF again.

Force all control lines in ID/EX to 0:
EX, MEM, WB will execute nop (no operation)

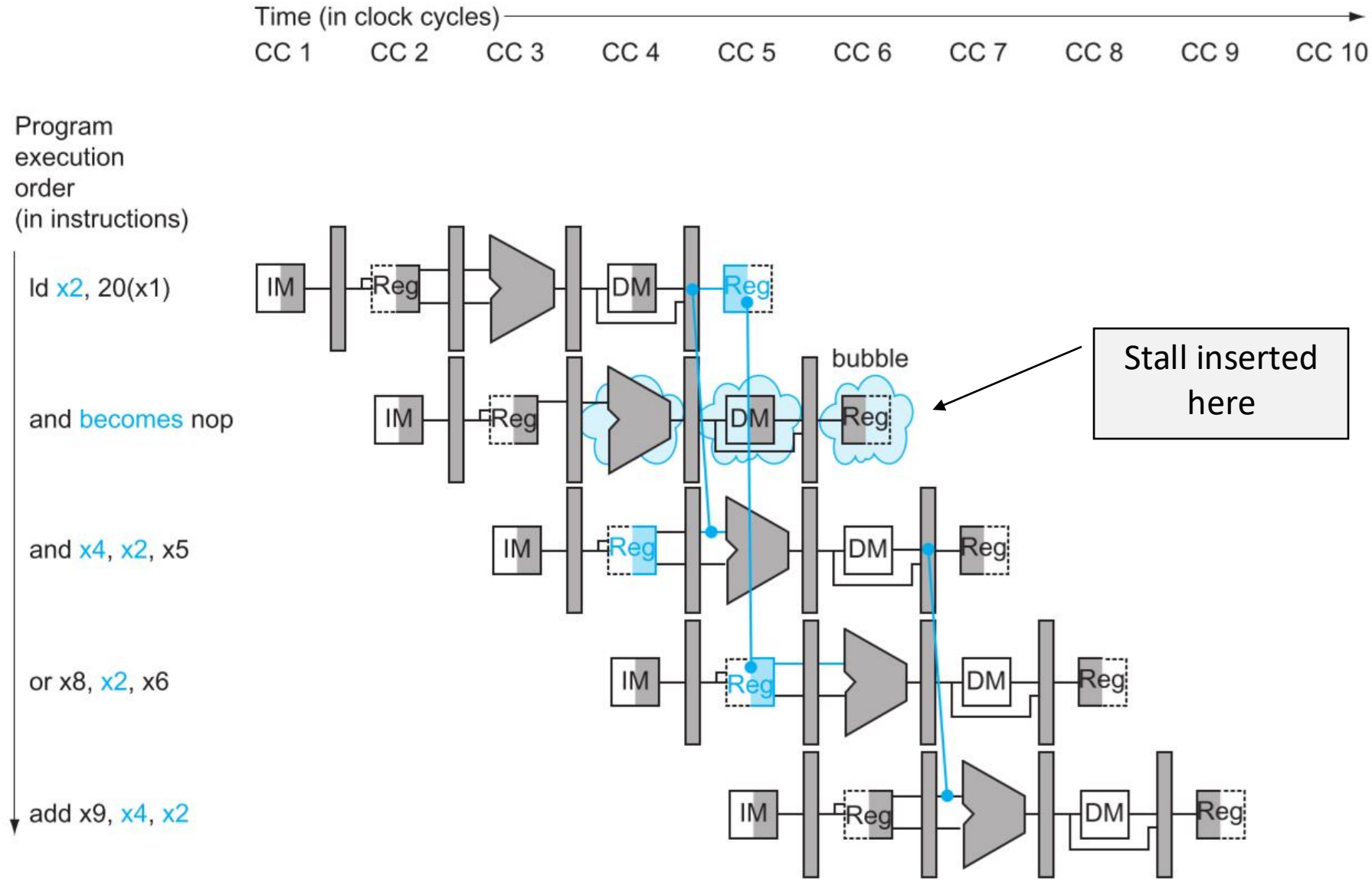


Repeat decoding of Instruction in ID

Datapath with Hazard Detection



Stall/Bubble in the Pipeline



Stall/Bubble in the Pipeline