Computer Organization
Chapter 4

ID: Name: 2012/3/7

1. (5%)(Refer to the CPU architecture of Figure 1 below) Which of the following statements is correct for a load word (LW) instruction?
(A) MemtoReg should be set to 0 so that the correct ALU output can be sent to register file.
(B) MemtoReg should be set to 1 so that the Data Memory output can be sent to the register file.
(C) We do not care about the setting of MemtoReg. It can be either 0 or 1.
(D) MemWrite should be set to 1.

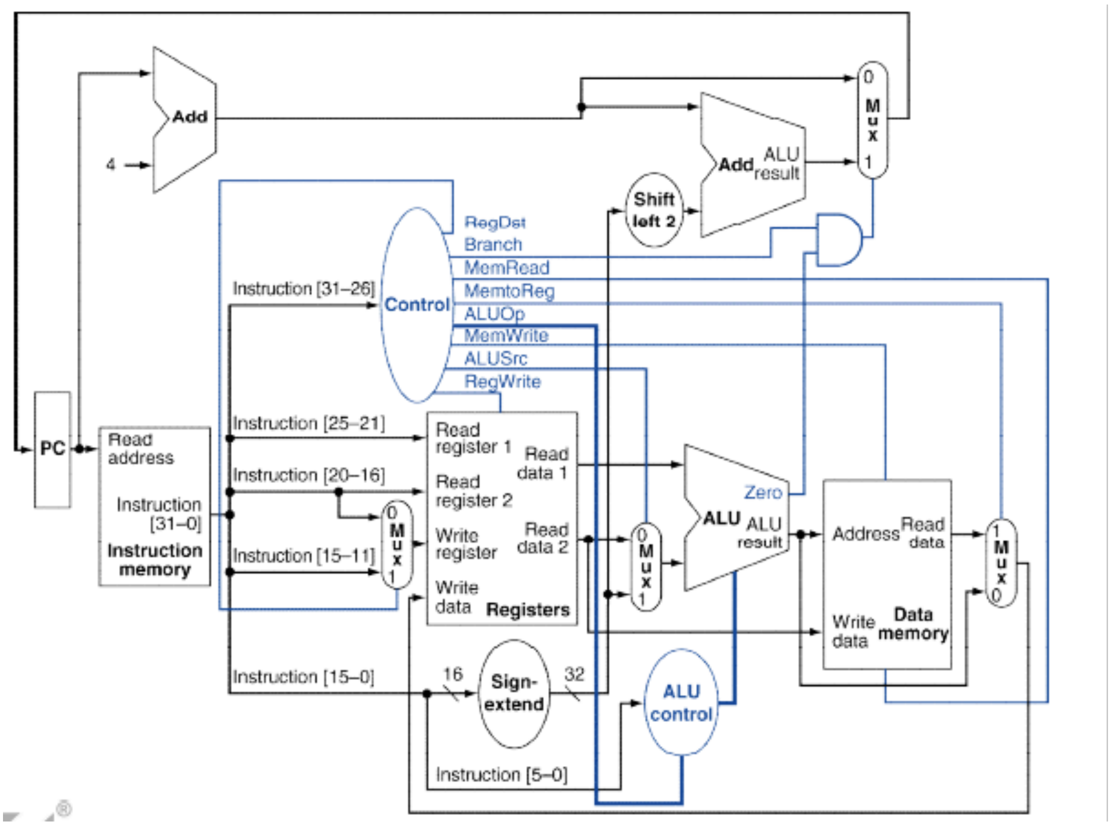


Figure 1

Ans: (B)

2. (5%)What is NOT a basic concept of pipeline?
(A) Partition instruction execution into balanced stage.
(B) Overlap execution of consecutive instruction
(C) Share hardware

Ans: (C)

3. (5%)Which one, if there is no forwarding, will cause pipeline stall?

```
(A) add $s0, $t0, $t1
    sub $s0, $t1, $t0
(B) add $t0, $t0, $t1
    sub $s0, $t1, $t0
(C) add $s1, $s0, $t1
    sub $s0, $t1, $t0
(D) add $s0, $t0, $t0
    sub $t0, $t1, $t1
```

Ans: (B)

4. (5%)Which stall can NOT be solved by data forwarding?

```
(A) sub $t3, $t0, $t1
    add $s3, $t1, $t3
(B) add $s3, $t1, $t3
    or  $s0, $s3, $s3
(C) lw  $s1, 100($0)
    sub $t3, $t0, $s1
```

Ans: (C)

5. (5%)Data hazard can be solved by compiler through

(a) data forwarding

(b) dynamic scheduling

(c) reordering instructions sequence

Ans: (c)

6. (5%)Please write down three reasons that cause EXCEPTION happen in pipeline.

Ans: I/O request, a user program call OS service(system call), overflow, undefined code, hardware breakdown

7. (30%)Base on the function of the seven control signals and the datapath of the MIPS CPU in Figure 1 (the same figure for Question 1), complete the setting of the control lines in the following Table (use 0, 1, and X (don’t care) only) for the two MIPS CPU instructions (beq and add). X (don’t care) can help to reduce the implementation complexity, you should put X whenever possible.

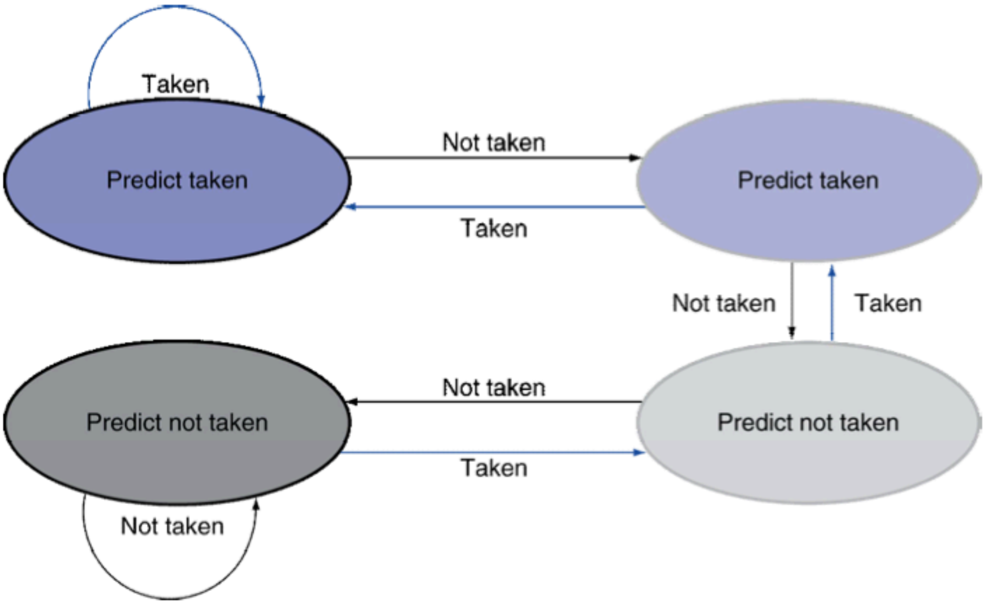
Instr.	Branch	ALU Src	Reg Write	Reg Dst	Memto Reg	Memory Write	Memory Read	ALU Op1	ALU Op2
beq	(1)	(2)	(3)	(4)	(5)	(6)	(7)	0	1
add			(8)		(9)	(10)			

(1) =
(2) =
(3) =
(4) =
(5) =
(6) =
(7) =
(8) =
(9) =
(10) =

Ans:
(1) = 1
(2) = 0
(3) = 0
(4) = X
(5) = X
(6) = 0
(7) = 0
(8) = 1
(9) = 0
(10) = 0

8. (10%) Please draw the flow chart of 2-Bit Predictor.

Ans:



9. (10%) Assume the following latencies for logic blocks in the datapath:

I-Mem	Add	Mux	ALU	Regs	D-Mem	Sign-extend	Shift-left-2
400ps	150ps	100ps	180ps	170ps	1000ps	80ps	30ps

a) (2%) What is the clock cycle time if the only type of instruction we need to support are ALU instructions (add, and, etc.)?
b) (2%) What is the clock cycle time if we only had to support lw instructions?
c) (2%) What is the clock cycle time if we must support add, beq, lw and sw instructions?
d) (4%) If we can improve the latency of one of the given datapath components by 5%, which component should it be? What is the speed-up from this improvement?

Ans:

a) 400+170+100+180+100 = 950ps
b) 400+170+100+180+1000+100 = 1950ps
c) 400+170+100+180+1000+100 = 1950ps
d) D-Mem 1000ps → 950ps
1950/1900 = 1.026

10. (10%) Assume that individual stages of the datapath have the following latencies:

	IF	ID	EX	MEM	WB
1)	200ps	300ps	500ps	600ps	200ps
2)	100ps	120ps	140ps	180ps	260ps

a) (3%) What is the clock cycle time in a pipelined and nonpipelined processor?

b) (3%) What is the total latency of a lw instruction in a pipelined and nonpipelined processor?

c) (4%) If we can split one stage of the pipelined datapath into two new stages, each with half the latency of the original stage, which stage would you split and what is the new clock cycle time of the processor?

Ans:

a) 1) 600ps, 1800ps 2) 200ps, 800ps

b) 1) 3000ps, 1800ps 2) 1000ps, 800ps

c) 1) MEM, 500ps 2) WB, 180ps

11. (10%) The condition for detecting hazards and the control signals to resolve them are as follows:

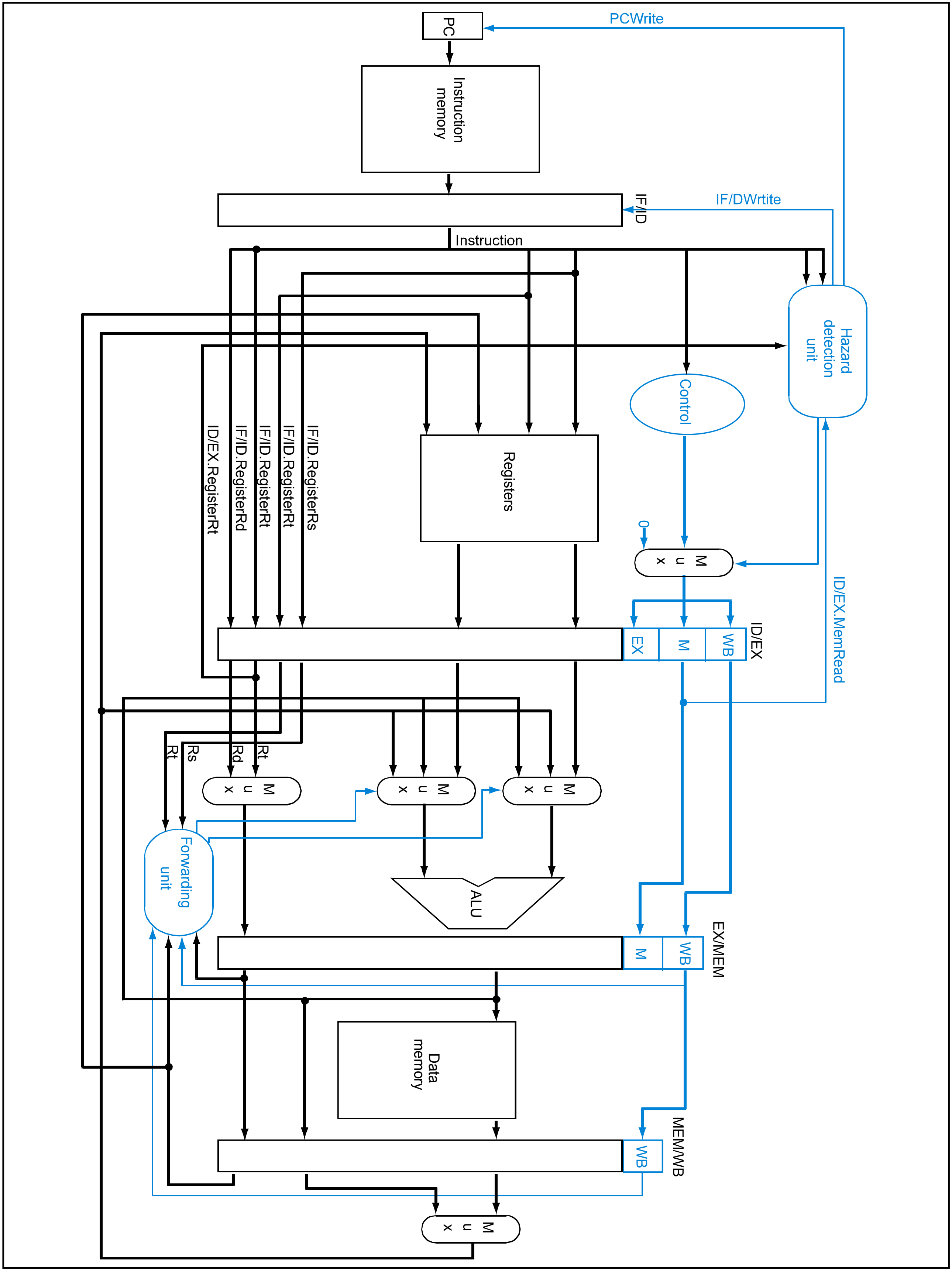
```
EX hazards:
    if( EX/MEM.RegWrite
    and (EX/MEM.RegisterRd != 0)
    and (EX/MEM.RegisterRd = ID/EX.RegisterRs)) ForwardA = 10

    if( EX/MEM.RegWrite
    and (EX/MEM.RegisterRd != 0)
    and (EX/MEM.RegisterRd = ID/EX.RegisterRt)) ForwardB = 10
MEM hazard:
    if(MEM/WB.RegWrite
    and (MEM/WB.RegisterRd != 0 )
    and (MEM/WB.RegisterRd = ID/EX.RegisterRs)) ForwardA = 01

    if(MEM/WB.RegWrite
    and (MEM/WB.RegisterRd != 0 )
    and (MEM/WB.RegisterRd = ID/EX.RegisterRs)) ForwardB = 01
```

There are some problems in this design. Please correct them. The pipelined datapath is shown in Figure 1 for your reference ans:

MEM hazard

```
   if (MEM/WB.RegWrite and (MEM/WB.RegisterRd != 0)
and not (EX/MEM.RegWrite and (EX/MEM.RegisterRd != 0)

and (EX/MEM.RegisterRd = ID/EX.RegisterRs))

and (MEM/WB.RegisterRd = ID/EX.RegisterRs)) ForwardA = 01

   if (MEM/WB.RegWrite and (MEM/WB.RegisterRd != 0)
and not (EX/MEM.RegWrite and (EX/MEM.RegisterRd != 0)

and (EX/MEM.RegisterRd = ID/EX.RegisterRt))

and (MEM/WB.RegisterRd = ID/EX.RegisterRt)) ForwardB = 01
```

Figure 1.


PC
PCWrite
Instruction memory
IF/ID
IF/DWrtite
Instruction
Hazard detection unit
Control
Registers
IF/ID.RegisterRs
IF/ID.RegisterRt
IF/ID.RegisterRt
IF/ID.RegisterRd
ID/EX.RegisterRt
0
M u x
ID/EX
EX
M
WB
ID/EX.MemRead
Rt
Rd
Rs
Rt
M u x
M u x
M u x
Forwarding unit
ALU
EX/MEM
M
WB
Data memory
MEM/WB
WB
M u x