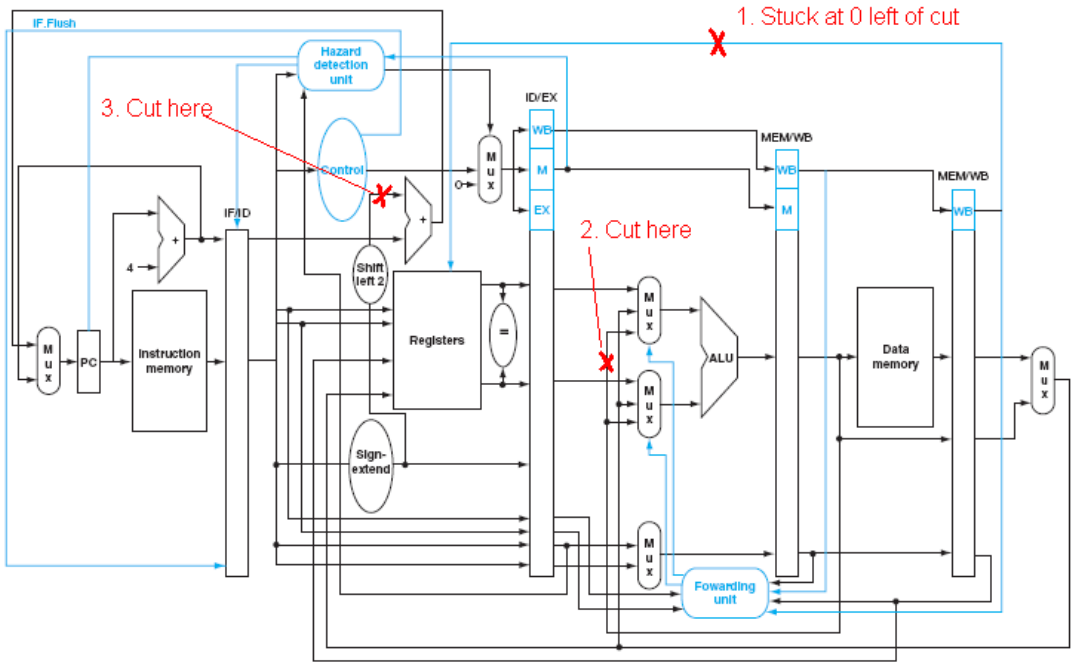
CMPEN 331 – Computer Organization and Design, Chapter 4 Review Questions

1. For the MIPS datapath shown below, several lines are marked with "X". For each one:



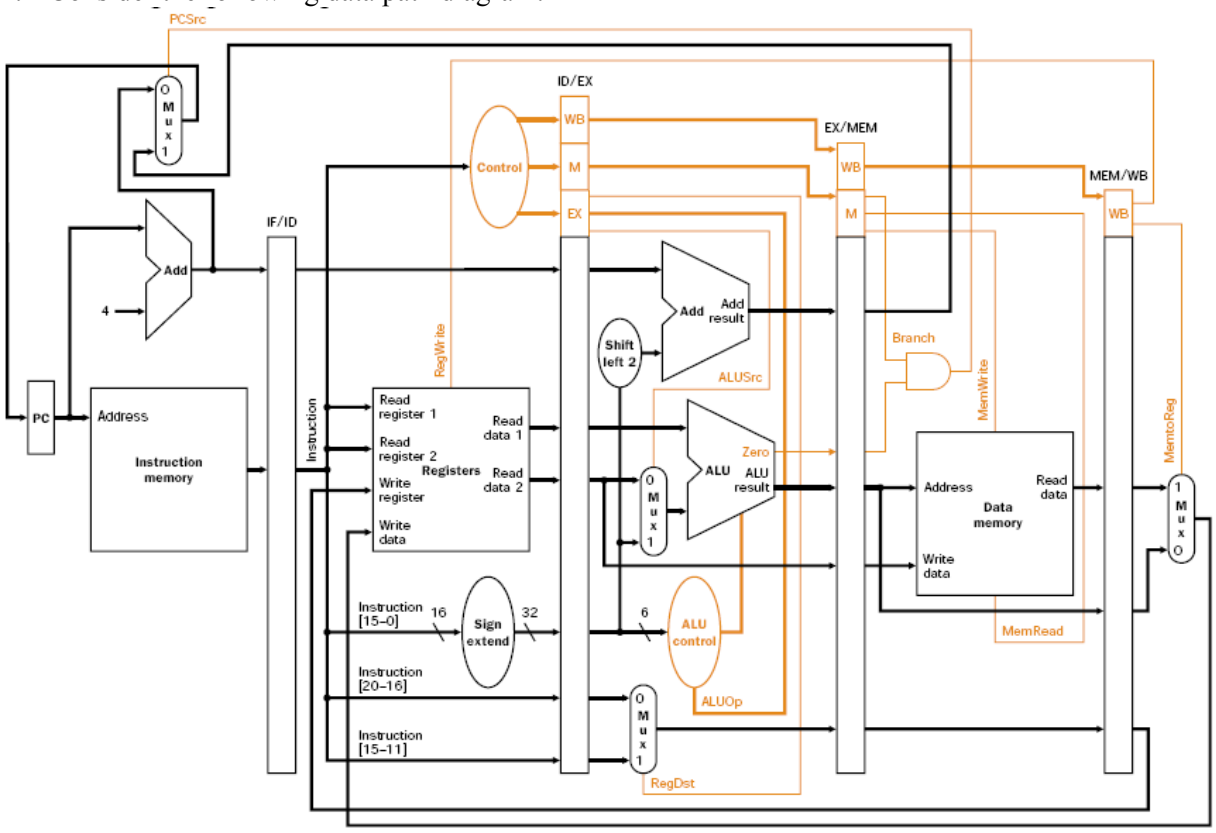
a. Describe in words the negative consequence of cutting line 1 relative to the working, unmodified processor. Give an example for a code that would fail and another one for a code that will not fail.

b. Describe in words the negative consequence of cutting line 2. Provide a snippet of code that will fail and a code that will still work.

c. Describe in words the negative consequence of cutting line 3. Provide a snippet of code that will fail and a code that will still work.

d. Can you show one mistake in the figure.

2. Consider the following data path diagram:



a. Modify the diagram to indicate the datapath changes (and any additional multiplexing) needed to provide bypassing from EX to EX for all possible hazards on arithmetic instructions. How does ALUSrc change when bypassing is added?

3. Consider the following assembly language code:

I0: ADD R4 = R1 + R0;
I1: SUB R9 = R3 - R4;
I2: ADD R4 = R5 + R6;
I3: LDW R2 = MEM[R3 + 100];
I4: LDW R2 = MEM[R2 + 0];
I5: STW MEM[R4 + 100] = R2;
I6: AND R2 = R2 & R1;
I7: BEQ R9 == R1, Target;
I8: AND R9 = R9 & R1;

Consider a pipeline with forwarding, hazard detection, and 1 delay slot for branches. The pipeline is the typical 5-stage IF, ID, EX, MEM, WB MIPS design. For the above code, complete the pipeline diagram below (instructions on the left, cycles on top) for the code. Insert the characters IF, ID, EX, MEM, WB for each instruction in the boxes. Assume that there two levels of bypassing, that the second half of the decode stage performs a read of source registers, and that the first half of the write-back stage writes to the register file. Label all data stalls (Draw an X in the box). Label all data forwards that the forwarding unit detects (arrow between the stages handing off the data and the stages receiving the data). What is the final execution time of the code?

	T0	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15	T16	T17
I0																		
I1																		
I2																		
I3																		
I4																		
I5																		
I6																		
I7																		
I8																		

4. Consider the 5-stage single-issued pipelined MIPS datapath consisting of Instruction Fetch (IF), Instruction Decode (ID), Execution (EX), Memory (MEM), and Write-Back (WB).

   You are given the following MIPS instruction sequence:

```
# $s0 to $s3 = 56, 30, 30, 7
# $t0 to $t4 = 7, 7, 7, 7, 7
      add $t0, $s0, $0
      and $t1, $t0, $s1
      or $t2, $t0, $s2
      sub $t3, $t0, $s3
      srl $t4, $t0, 2
```

Start numbering the cycles with 1 when the add instruction enters the IF stage.

For part i. to iii. assume that the datapath is broken and there is no forwarding and no stalling.

i. What are the values of $t0 to $t4 at the end of cycle 7? **(2 points)**
(a) 56, 24, 62, 7, 7

(b) 56, 24, 62, 49, 7

(c) 56, 6, 31, 7, 7

(d) 56, 6, 7, 7, 7

(e) None of the above

ii. What are the values of $t0 to $t4 at the end of cycle 8? **(2 points)**
(a) 56, 24, 62, 49, 7

(b) 56, 24, 62, 49, 14

(c) 56, 6, 31, 49, 7

(d) 56, 6, 31, 7, 7

(e) None of the above

iii. What are the values of $t0 to $t4 at the end of cycle 9? **(2 points)**
(a) 56, 24, 62, 49, 7

(b) 56, 24, 62, 49, 14

(c) 56, 6, 31, 49, 14

(d) 56, 6, 31, 0, 7

(e) None of the above

iv. What instruction(s) is/are computing the wrong result(s) (choose the answer that includes ALL faulty instructions)? **(2 points)**
(a) add

(b) and, or

(c) and, or, sub

(d) and, or, sub, srl

(e) add, and, or, sub, srl

v. Say we want to completely fix the problem from part iv. using forwarding. Which forwarding path(s) do we need to provide in order to execute the code sequence correctly (it is implied that multiplexers are inserted to join the forwarded signals with the original signals)? **(2 points)**

(a) Output of ALU in the EX stage back to the input of the ALU in the EX stage.

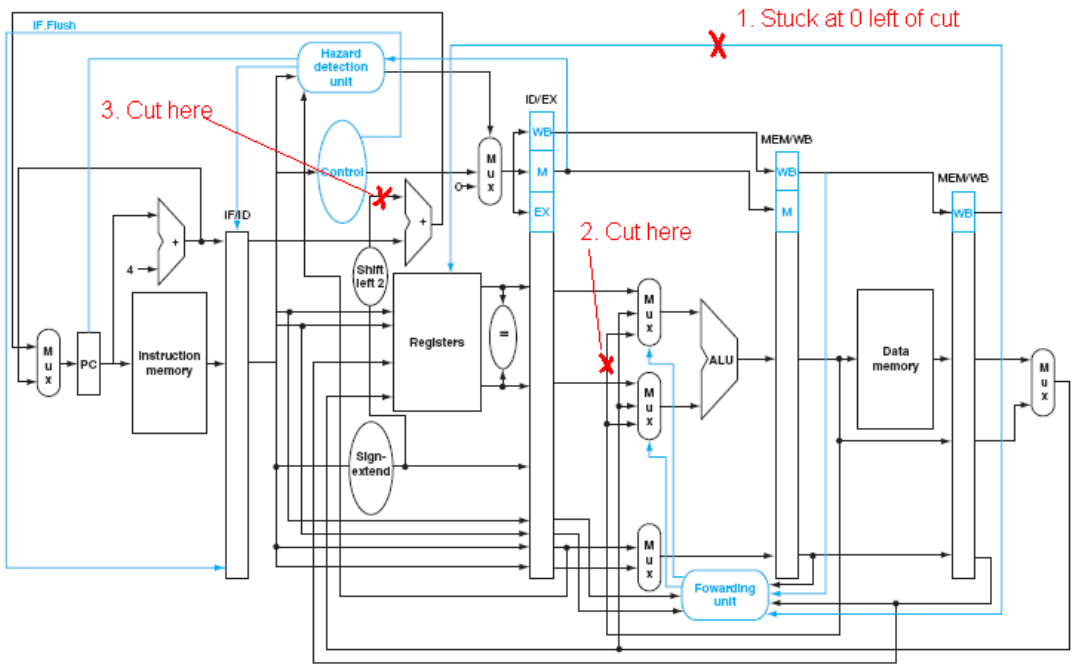
(b) Output of ALU in the MEM stage back to the output of Register File in the ID stage.

(c) Output of ALU in the MEM stage back to the input of ALU in the EX stage.

(d) Both (a) and (b).

(e) Both (a) and (c).

1. For the MIPS datapath shown below, several lines are marked with "X". For each one:



e. Describe in words the negative consequence of cutting line 1 relative to the working, unmodified processor. Give an example for a code that would fail and another one for a code that will not fail.

a. Cannot write to register file. This means that R-type and any instruction with write back to register file will fail. An example of code snippet that would fail is:
add $s1, $s2, $s3

An example of a code snippet that will not fail is:
sw $s1, 0($s2)

f. Describe in words the negative consequence of cutting line 2. Provide a snippet of code that will fail and a code that will still work.

Forwarding of the first operand fails. An example of code snippet that would fail is:
add $s1, $t0, $t1

add $s1, $s1, $s1

An example of code snippet that will not fail is:
add $s1, $t0, $t1
add $s1, $t2, $s1 # Here the second operand is forwarded correctly

g. Describe in words the negative consequence of cutting line 3. Provide a snippet of code that will fail and a code that will still work.

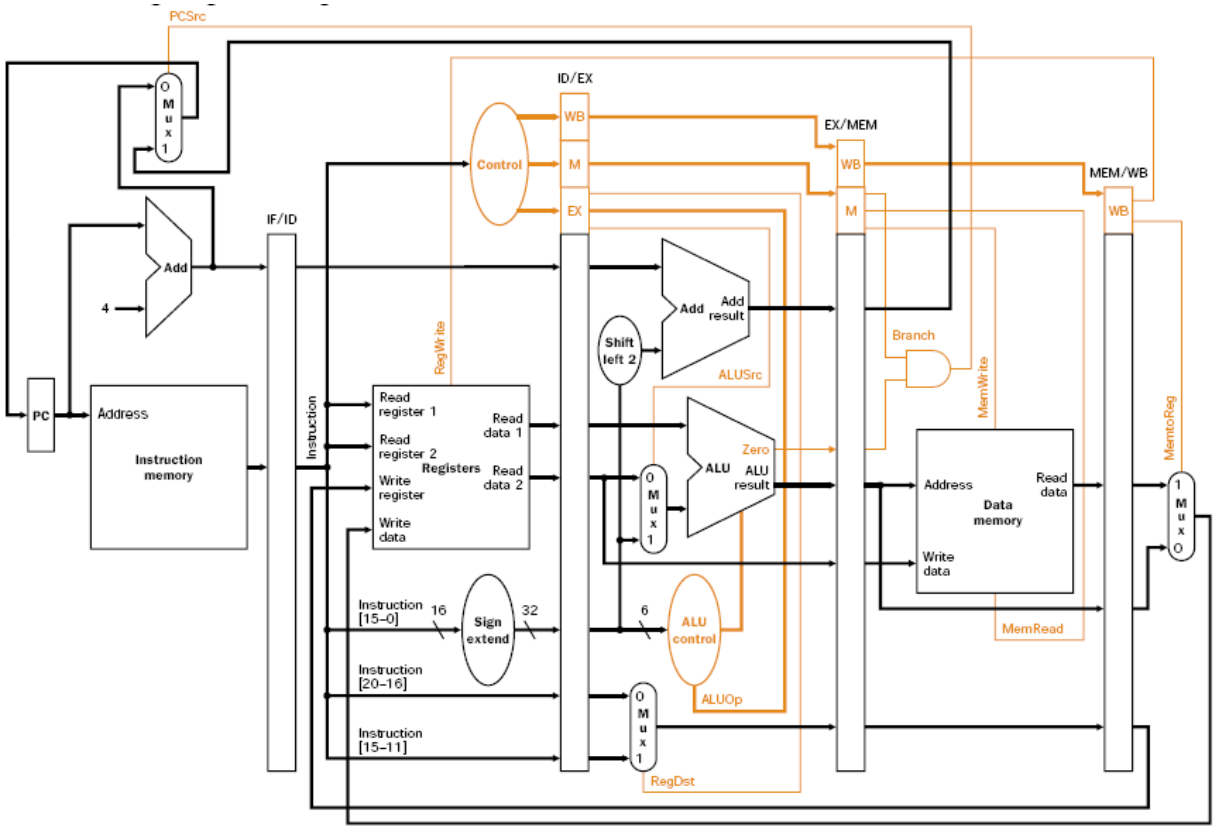
Jumping to a branch target does not work. Example of code that fails:
addi $s1, $zero, 2
addi $s2, $zero, 2
beq $s1, $s2, exit

Code that will still work:
addi $s1, $zero, 10
addi $s2, $zero, 20
beq $s1, $s2, exit

h. Can you show one mistake in the figure.

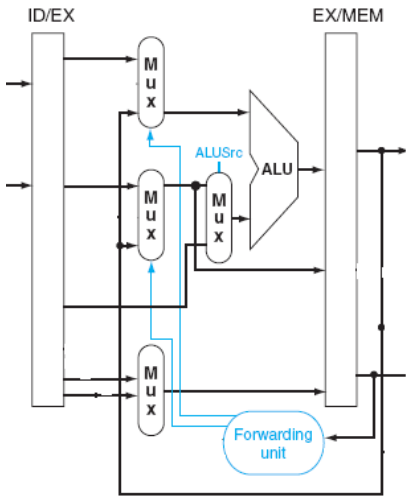
MEM/WB is shown on top of the EX/MEM

2. Consider the following data path diagram:



b. Modify the diagram to indicate the datapath changes (and any additional multiplexing) needed to provide bypassing from EX to EX for all possible hazards on arithmetic instructions. How does ALUSrc change when bypassing is added?

The ID/EX and EX/MEM latches will have a new forwarding unit between them directing the outputs of the EX stage into the MUX's that feed the inputs into the ALU. The ALUSrc control signal will not have to be modified since it is already the output from one of the MUX's. The modification (only the relevant part of the diagram is shown below) is as follows:



3. Consider the following assembly language code:

I0: ADD R4 = R1 + R0;
I1: SUB R9 = R3 - R4;
I2: ADD R4 = R5 + R6;
I3: LDW R2 = MEM[R3 + 100];
I4: LDW R2 = MEM[R2 + 0];
I5: STW MEM[R4 + 100] = R2;
I6: AND R2 = R2 & R1;
I7: BEQ R9 == R1, Target;
I8: AND R9 = R9 & R1;

Consider a pipeline with forwarding, hazard detection, and 1 delay slot for branches. The pipeline is the typical 5-stage IF, ID, EX, MEM, WB MIPS design. For the above code, complete the pipeline diagram below (instructions on the left, cycles on top) for the code. Insert the characters IF, ID, EX, MEM, WB for each instruction in the boxes. Assume that there two levels of bypassing, that the second half of the decode stage performs a read of source registers, and that the first half of the write-back stage writes to the register file. Label all data stalls (Draw an X in the box). Label all data forwards that the forwarding unit detects (arrow between the stages handing off the data and the stages receiving the data). What is the final execution time of the code?

	T0	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15	T16	T17
I0																		
I1																		
I2																		
I3																		
I4																		
I5																		
I6																		
I7																		
I8																		

	T0	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15	T16	T17
I0	IF	ID	EX	MEM	WB													
I1		IF	ID	EX	MEM	WB												
I2			IF	ID	EX	MEM	WB											
I3				IF	ID	EX	MEM	WB										
I4					X	IF	ID	EX	MEM	WB								
I5						X	IF	ID	EX	MEM	WB							
I6							X	IF	ID	EX	MEM	WB						
I7									IF	ID	EX	MEM	WB					
I8										IF	ID	EX	MEM	WB				

The final execution time of the code is 14 cycles (T0-T13).
Check if there is any missing forwarding.

4. Consider the 5-stage single-issued pipelined MIPS datapath consisting of Instruction Fetch (IF), Instruction Decode (ID), Execution (EX), Memory (MEM), and Write-Back (WB). **(10 points)**

You are given the following MIPS instruction sequence:

```
# $s0 to $s3 = 56, 30, 30, 7

# $t0 to $t4 = 7, 7, 7, 7, 7

      add $t0, $s0, $0
```

```
and $t1, $t0, $s1

or $t2, $t0, $s2

sub $t3, $t0, $s3

srl $t4, $t0, 2
```

Start numbering the cycles with 1 when the add instruction enters the IF stage.

For part i. to iii. assume that the datapath is broken and there is no forwarding and no stalling.

vi. What are the values of $t0 to $t4 at the end of cycle 7? **(2 points)**
(a) 56, 24, 62, 7, 7

(b) 56, 24, 62, 49, 7

(c) 56, 6, 31, 7, 7

(d) 56, 6, 7, 7, 7

(e) None of the above

c

vii. What are the values of $t0 to $t4 at the end of cycle 8? **(2 points)**
(a) 56, 24, 62, 49, 7

(b) 56, 24, 62, 49, 14

(c) 56, 6, 31, 49, 7

(d) 56, 6, 31, 7, 7

(e) None of the above

c

viii. What are the values of $t0 to $t4 at the end of cycle 9? **(2 points)**
(a) 56, 24, 62, 49, 7

(b) 56, 24, 62, 49, 14

(c) 56, 6, 31, 49, 14

(d) 56, 6, 31, 0, 7

(e) None of the above

c

ix. What instruction(s) is/are computing the wrong result(s) (choose the answer that includes ALL faulty instructions)? **(2 points)**

(a) add

(b) and, or

(c) and, or, sub

(d) and, or, sub, srl

(e) add, and, or, sub, srl

b

x. Say we want to completely fix the problem from part iv. using forwarding. Which forwarding path(s) do we need to provide in order to execute the code sequence correctly (it is implied that multiplexers are inserted to join the forwarded signals with the original signals)? **(2 points)**

(a) Output of ALU in the EX stage back to the input of the ALU in the EX stage.

(b) Output of ALU in the MEM stage back to the output of Register File in the ID stage.

(c) Output of ALU in the MEM stage back to the input of ALU in the EX stage.

(d) Both (a) and (b).

(e) Both (a) and (c).

e