

**352 Final Exam
Spring 2012**

Name: _____

Email: _____

Question 1: _____/15

Question 2: _____/15

Question 3: _____/15

Question 4: _____/15

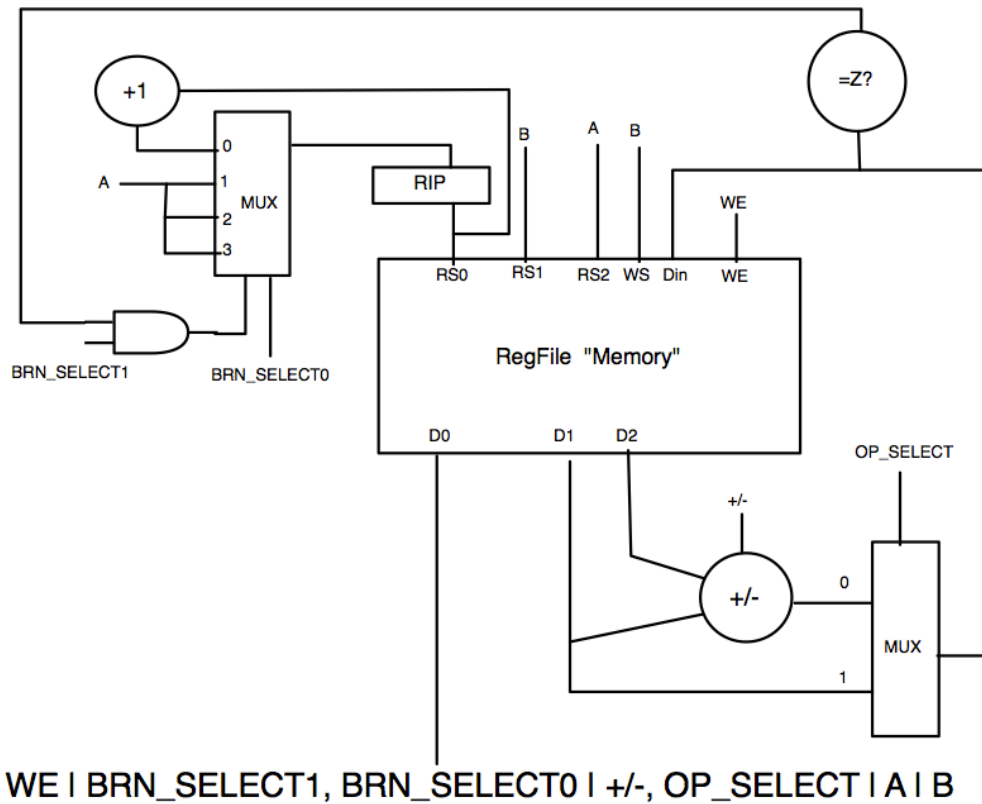
Question 5: _____/10

Question 6: _____/10

Question 7: _____/20

Summer gift: As you may get bored this summer enjoying all that beautiful sunshine, I leave for you the following gift. The gift is a challenge! Remember constructions from high school geometry / trigonometry? This is where you try and draw various shapes using only a drafting compass and a straight-edge. You may recall that drawing a hexagon is pretty simple: draw circle, use the drafting compass to score the circumference into six evenly sided pieces of the same size as the radius of the circle. Connect those dots. The challenge is to draw a pentagram. If you solve this (without using a search engine to help you) I will buy you a cup of coffee and be suitably impressed -- it is not an easy problem to solve!

Question 1) In class we studied this very simple single-cycle computer.



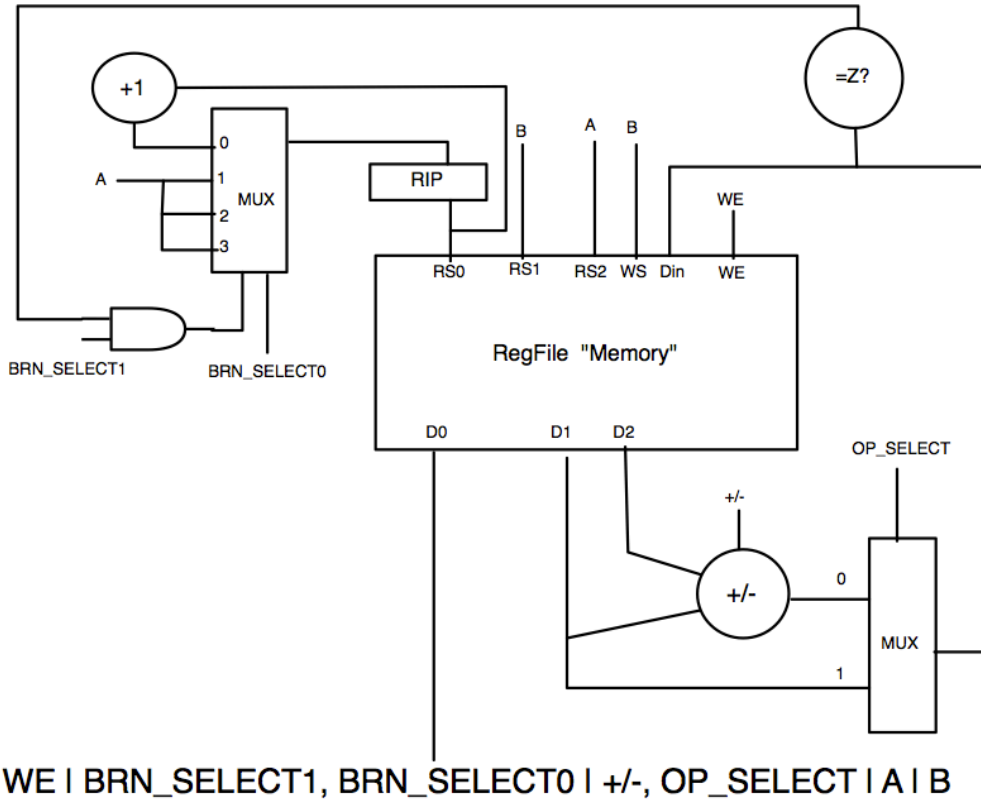
Write out the instruction encoding for the following instructions:

SUB source, source/destination

JUMP label

JUMP_IF_ZERO operand, label

Question 2)

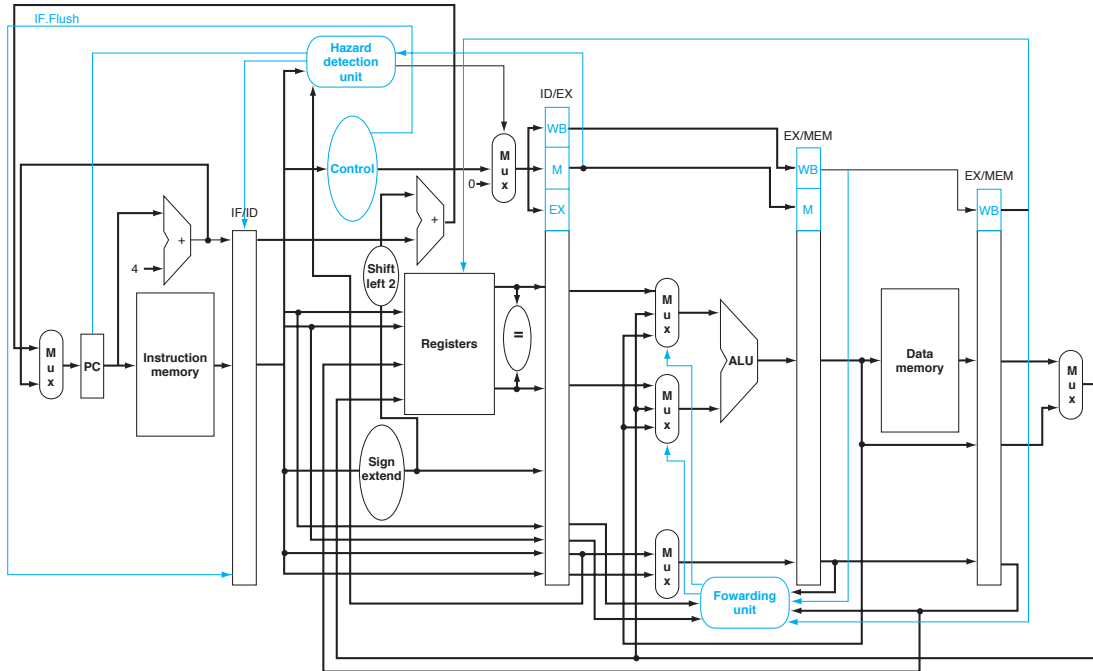


Modify the microarchitecture above to implement the following instruction:

DECREMENT_JUMP_IF_NOT_ZERO SRC/Destination, Label

This instruction should take an operand, decrement it by 1, write it back and then jump to a location if the result of the subtract is not zero. Your solution should modify the microarchitecture (add wires and any additional hardware needed) and show the bit encoding for an example DECREMENT_JUMP_IF_NOT_ZERO instruction.

Question 3)

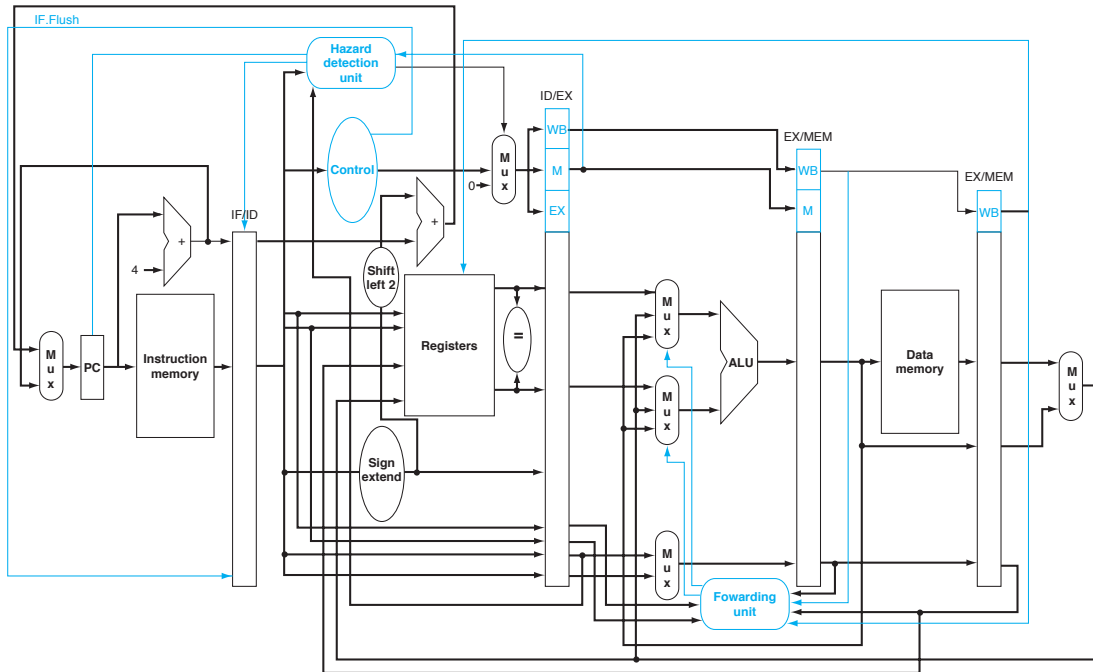


Given the above pipelined architecture, fill in the following table for the following code sequence:

here: LOAD \$r4, 0(\$r5)
 BEQ \$r4, \$r0, here
 SUB \$r5, \$r2, \$r5

	Fetch	Decode	Ex	Mem	WB
Clock 0	Load				
Clock 1					
Clock 2					
Clock 3					
Clock 4					
Clock 5					
Clock 6					
Clock 7					

Question 4)



Given the following code sequence:

here:

```

LOAD $r8, 0($r9)
STORE $r9, 0($r10)
ADD $r8, $r9, $r7
BEQ $r8, $r0, here
SUB $r8, $r9, $r7
    
```

Assume the LOAD is in the write-back stage of the processor. Show where each additional instruction in the sequence is in the processor and high-light any forwarding busses that are in use and what register values are being forward on them.

Question 5) Draw the CMOS transistor level circuit that implements the following function: $ABCD$ **Use as few transistors as possible.** This problem will be graded on the quality of the circuit as well as its correctness.

Question 6, part 1): Draw the gate-level circuit for a 4 bit multiplier. Assume you have a full-adder component given to you.

part 2) Suppose you wanted to pipeline your solution so it would be roughly 2X as fast. Draw the solution for a pipelined 4 bit multiplier. Make sure your solution can handle dependent back to back multiplies!

Question 7)

a) What is meant by “precise” in the term precise interrupt?

b) What is memory mapped I/O?

c) What is meant by the term “semi-custom” in chip design?

d) Write the Verilog code for a flip flop with asynchronous reset:

e) Which statement of the following is patently false?

- Forwarding networks can go from the beginning of one pipeline stage to the beginning of another pipeline stage
- Forwarding networks can go from the end of one pipeline stage to the end of another pipeline stage.
- Forwarding networks can go from the beginning of one pipeline stage to the end of another pipeline stage.
- Forwarding networks can go from the end of one pipeline stage to the beginning of another pipeline stage.

f) What is a glitch? Is it good or bad or indifferent? Why?

g) What are the benefits of out of order execution?

h) What are the benefits of concurrent execution on multiple cores?

i-j) Draw the transistor circuit for a positive edge triggered static flip-flop (as would be used in a register file, not a pipeline register):