6.823 Computer System Architecture
Prerequisite Assessment Spring 2002
Problem Set #0

Problem Set 0 is intended as a prerequisite assessment to determine if you have the background necessary to take 6.823. This course assumes that you have a solid understanding of the material presented in 6.004 or an equivalent course.

For each question, we ask that you fill out the table at the end of the problem set handout indicating your level of confidence with each assigned problem and hand this in with your solutions. If you have never seen the material before, then please enter "0". If you have seen the material, and think you should know it, but can't answer the question without spending time studying your old notes, then please enter "1". If you are comfortable with the material, then enter "2". You should turn in solutions for problems where you entered "1" and "2", but do not have to turn in solutions for problems for which you entered a "0".

Unlike recent offerings of 6.823, we will only have a short review of this earlier material this time. If there are too many "0"s in the table then it is unlikely you will be able to keep up with 6.823 this term, as the pace is such that you will not be able to take prerequisite material concurrently. If you are not sure whether you have the background to take this class, please feel free to discuss your particular situation with a TA.

For this problem set only, you must work individually and turn in your own solutions. Do not discuss the problems with others.

Problem 1

Construct the following logic functions using only two-input NAND gates (please use hierarchy where possible to simplify your designs):

a) inverter
b) two-input XOR gate
c) 2-to-1 multiplexer
d) 2-to-4 decoder
e) transparent latch (transparent on clock low)
f) flip-flop (positive-edge triggered)

Problem 2

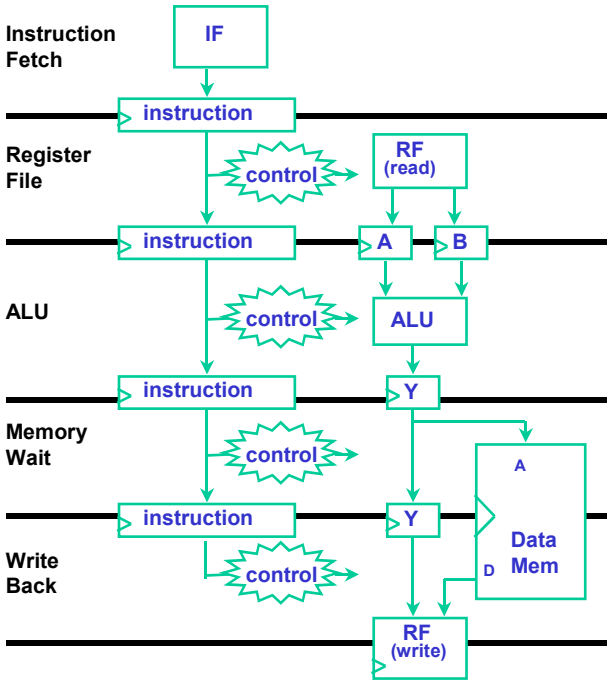
A finite-state machine built from flip-flops and combinational logic runs at a maximum clock speed of X MHz. The flip-flops have setup time, hold time, and clock-to-data time equal to zero.

a) How will the maximum clock speed change if the hold time becomes positive?
b) How will the maximum clock speed change if the setup time becomes negative?

c) If the setup time remains zero, but the hold time becomes positive, how could this affect circuit operation?
d) If the setup time remains zero, but the hold time becomes negative, how could this affect circuit operation?

Problem 3

The following figure shows a 5-stage pipelined processor. We want this pipelined processor to run exactly the same instructions as the unpipelined processor (no branch delay slot). What kind of problems can the following instruction sequences cause? What hardware solutions are possible to solve the problems (if at all possible, propose the ways that do not impact performance)?



a)
```
BEQZ    r1, 200     (branch to 200 if r1 == 0)
ADD     r2, r3, r5
SUB     r4, r5, r6
…
```

b)
```
ADD     r1, r0, r2  (r1 ← r0 + r2)
SUB     r4, r1, r2  (r4 ← r1 - r2)
…
```

c)
```
J       200         (unconditional jump to 200)
ADD     r2, r3, r5
SUB     r4, r5, r6
…
```

d) LD r1, 0(r2) (r1 ← Mem[r2+0])
ADD r1, r0, r2
…

Problem 4

Describe the operation of a data cache. Your description should include discussion of the following:

a) Spatial and temporal locality.
b) Valid bits.
c) Direct mapped versus set-associative structures. Show how cache indexing and tag match works for both direct mapped and 2-way set-associative cache configurations assuming one word per cache line. What are the advantages and disadvantages of direct mapped versus set-associative structures?
d) Multiple-word cache lines. What are the advantages and disadvantages of multiple-word cache lines? Describe how they are implemented for a direct mapped cache.
e) LRU and random replacement policies. What are their relative advantages and disadvantages?

Problem Ratings

		subproblem						
		A	**B**	**C**	**D**	**E**	**F**	**G**
problem	**1**							
	2							
	3							
	4							

0	No idea
1	Used to know it
2	Know it