

CSE352 Autumn 2013 Homework #5

Instructor: Mark Oskin

TAs: Vincent Lee, Mark Wyse

Due In Class 11/15/2013

Version 1.2

Please write your name and student ID at the top right corner of each page, and staple or paperclip your work together. We are NOT responsible for losing papers that were not stapled or paperclipped together.

Complete the following questions. Please write legibly and try to draw clean diagrams. Spaghetti wiring in circuit diagrams is difficult to grade. We will not grade work that is too heavily encrypted for us to read (i.e. we can't read it, we can't grade it). Please consider typesetting your work if you think that it may not be legible to the grader. You are encouraged to collaborate with your peers but you must turn in your own work. Justice will be enforced if you are caught cheating.

For this problem set, the instruction set you should use is the MIPS instruction set in your textbook, NOT x86. Chapter 6 in your textbook gives an overview and formulation of the MIPS instruction set. We recommend you read through that section and familiarize yourself with the MIPS instruction set before attempt some of the problems. Remember, in MIPS the zero register \$0 always contains zero.

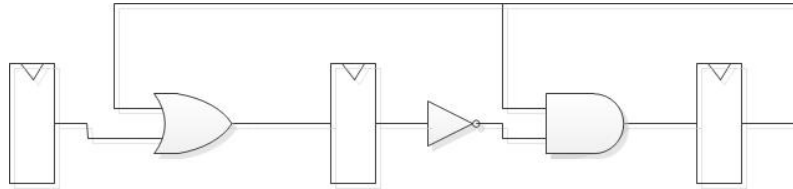
Problem 1 *Pipelining Warm Up*

Using only full adders and registers design a 4-bit adder circuit that can run at 100MHz to compute the 4-bit output `sum` and carry out `cout` given two 4-bit inputs `A` and `B`. Assume the worst case propagation delay through a full adder is $7ns$. The hold time and setup time for registers is $1ns$ and the clock-to-q delay for registers is $2ns$. It is okay for there to be several cycles of latency before the first result appears at the output.

Problem 2 *Pipeline Placement*

Ben Bitdiddle and Alyssa P. Hacker are arguing over whether the performance of the following circuit can be improved with additional pipelining. Ben argues that it is possible to improve the performance by adding another register between the AND

gate and inverter. Alyssa argues otherwise and claims that the structure of the circuit does not allow further pipelining. Who is correct? Briefly justify your answer.



Problem 3 *The 5 Stage Pipeline*

Ben Bitdiddle is compiling his program with NCC (Not Very Good Compiler Collection) for a 5 stage MIPS processor which **does not** use forwarding and gets the following sequence of assembly instructions:

```
addi $t0, $t0, $zero
addi $t0, $t0, 1
and $t1, $t1, $t0
addi $t2, $t2, $zero
addi $t2, $t2, 1
sll $t2, $t2, $t2
lui $t3, 0x1337
ori $t3, $t3, 0xCAFE
```

When he benchmarks his program he finds that his program is running poorly. How can he optimize this snippet of assembly instructions to improve the performance of his code? Write the new and improved sequence of instructions that should get him better performance.

Problem 4 *Long Latency Pipeline Stage*

Consider the following sequence of assembly code which is run on a slightly different 5 stage MIPS pipeline processor. In this processor, the memory stage now takes 5 cycles to complete. Assume forwarding paths between stages remains the same.

```
    add $t0, $0, 0x0100
    add $t2, $s0, $zero
    add $t3, $t3, $zero
loop: lw $t4, 0($t2)
      add $t3, $t3, $t4
      addi $t2, $t2, 4
      addi $t0, $t0, -1
      bgez $t0, loop
      jr $ra
```

1. Part A. Are there any hazards in this sequence of code? If so when do they occur, what types of hazards are they, and how long do they stall the pipeline for?
2. Part B. Rewrite the code such that all stall cycles are eliminated. Try to minimize the length of the code. (Hint: consider loop unrolling)

Problem 5 *Lightning Review Round*

1. For the following storage elements, order them in terms of increasing area: flip-flop, SRAM, DRAM
2. For the storage elements in part a, order them in terms of increasing storage density: flip-flop, SRAM, DRAM
3. Suppose you have a graphics processing algorithm implemented as a C program, Java program, ASIC solution, and FPGA solution. Order these solutions in terms of which completes the computation fastest.
4. What is a glitch? Is it good, bad, or indifferent? Why?
5. What is one of the primary reasons why FPGA solutions are slower than ASIC solutions?
6. For the following cases, state whether an FPGA or ASIC solution is more suitable and why:
 - (a) You are a large company designing ultralow power, high volume, and high performance processors.
 - (b) You are a government contractor manufacturing a highly specialized processor for the next generation of fighter jets to shoot down the fighter jets that shoot down the fighter jets that are currently used today.
 - (c) You are a small start up company developing a novel architecture that you need to show off at Techcrunch Seattle next month.
 - (d) You are a research group which gets donations from Altera and Xilinx.

Problem 6 *Infamous Digital Design Question of the Century (Optional)*

You are stuck on an island with a bucket of NAND gates. Using hierarchy and abstraction, design a 5 stage MIPS processor.

Problem 7 *Bonus Question: Combo Breaker (Optional)*

Write a dramatic monologue lamenting how terrible you're least favorite operating system is:

http://www.youtube.com/watch?v=nb_ogCmWjmo