CS 152
Computer Systems Architecture
Winter 2015

Homework 1

Due Tuesday, January 20th, 2015, EEE 11:59PM PDF

Name	Student ID
Ford Tang	46564602

IMPORTANT NOTES:

1. **No late submissions.**
2. **Please show your work. Remember that bottom line answers without proper explanations are worth ZERO points.**
3. **Remember that you are solely responsible for the answers to the questions, therefore, please refrain from consulting with your class peers.**

For Grading Purposes Only:

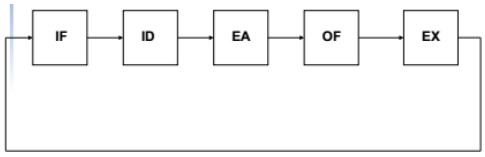
Q1	Q2	Q3	Q4	Q5	Q6
10	10	10	10	10	10

Q7	Q8	Q9	Q10	BONUS
10	10	10	10	10

Total Score
100

Problem 1 (10 points)

Draw the Von Neumann Cycle and explain the basic function carried out in each state. Explain clearly how a RISC cycle is obtained from the Von Neumann cycle.



IF: Instruction Fetch - Get instruction from memory
ID: Instruction Decode - Decodes instruction
EA: Evaluate Address - Gets address of instruction operation
OF: Operand Fetch - Loads operands into memory
EX: Execute - Execute Code

RISC:
Instructions are fully decoded in memory, so Instruction Decode is unnecessary.
Only Store/Load instructions can access memory, so no need to Evaluate Address and Operand Fetch cycles.
Only Instruction Fetch and Execute Cycles for RISC

Problem 2 (10 points)

Explain what is done in each of the RISC states for the 5 types of MIPS instructions represented with R, I, and J instruction formats.

R-Type - R Format: Performs a functions on first given register and second given register, and storing in destination register
I-Type - Conditional Branch: If first given register data and second given register data are different, then move current program pointer address by the offset
I-Type - Memory Access: Loads or stores data from the given register and the offset
I-Type - Immediate: Loads a constant into the register.
J-Type - Unconditional Jump: Moves the program counter by given offset.

Problem 3 (10 points)

In MIPS, the structure of its instructions is simplified. The way we implement complex instructions through the use of MIPS simplified instructions is to decompose complex instructions into multiple simpler MIPS ones. Show how MIPS can implement the instruction swap $rs, $rt, which swaps the contents of the registers $rs and $rt. Consider the case in which there is an available register that may be destroyed as well as the case in which no register exists.

If the implementation of this instruction in hardware will increase the clock period of a single instruction by 10%, what percentage of swap operations in the instruction mix would justify implementing it in hardware?

Swap: (with register)
Load r1, $rs
Load $rs, $rt
Load $rs, r1
Swap: (without register)
Store $rs, address
Load $rs, $rt
Load $rt, address

1.1t * 100 instructions = 110 Time to execute
(100 + 2k) * t = 100 + 2k Time to execute

If more than 5% of the instructions are swap instructions, then the function should be implemented in hardware.

Problem 4 (10 points)

a. Given two n-bit two's complement numbers, prove that an addition overflow occurs if and only if the carry into the most significant bit position and the carry out of the most significant bit position are different.
b. Consider now the sign-magnitude representation. Define the meaning of overflow in this representation and suggest an overflow detection mechanism.

Sign of A	Sign of B	Carry In	Carry out	Overflow
0	0	0	0	0
0	0	1	0	1
0	1	0	0	0
0	1	1	1	0
1	0	0	0	0
1	0	1	1	0
1	1	0	1	1
1	1	1	1	0

For sign-magnitude, overflow occurs when adding two integers of the same sign that will result in an opposite sign. To detect this error, change the sign-magnitude to two's complement and check for overflow.

Problem 5 (10 points)

With x = $(0101\ 0101)_2$ and y = $(1110\ 1111)_2$ representing two's complement signed integers, perform the following operations showing all work:

a. x + y
b. x – y
c. x * y
d. x / y

X + Y:

```
c 1 1 1 1 1 1 1 0
  0 1 0 1 0 1 0 1
+ 1 1 1 0 1 1 1 1
= 0 1 0 0 0 1 0 0
```

X - Y:

```
c 0 0 1 0 0 0 1 0
  0 1 0 1 0 1 0 1
+ 0 0 0 1 0 0 0 1
= 0 1 1 0 0 1 1 0
```

X * Y:

```
                0 1 0 1 0 1 0 1
              * 0 0 0 1 0 0 0 1
                0 1 0 1 0 1 0 1
+         0 1 0 1 0 1 0 1
          0 1 0 1 1 0 1 0 0 0 1 0 1
convert:  1 0 1 0 0 1 0 1 1 0 1 1
```

X / Y:

```
              _______0 1 0 1
0 0 0 1 0 0 0 1)0 1 0 1 0 1 0 1
             - 0 1 0 0 0 1
               0 0 0 1 0 0 0 1
              - 0 0 0 1 0 0 0 1
               0 0 0 0 0 0 0 0
convert: 1011
```

Problem 6 (10 points)

Given the bit pattern:

1010 1111 1010 1000 0000 0000 0000 1000

what does it represent, assuming that it is:

a. a two's complement integer?
b. an unsigned integer?
c. a single precision IEEE standard floating point number?
d. a MIPS instruction?

two's complement integer:
-1347944440
unsigned integer:
2947022856
floating point:
-2621449 x 2^63
MIPS:
op = 1010 11 = Store Word
rs = 11 101 = rs = first register
address = 0 1000 0000 0000 0000 1000

Problem 7 (10 points)

a. Convert the number 752 in base-9 representation into base 3 representation

$$752_{nine} = (\ 211202\ )_{three}$$

b. Represent the decimal number -1/3 in IEEE 754 floating-point binary representation (single and double precision).

_ 0 0 1 0 1 0 1 ...___ ; repeating
1 1) 1 0 0
- 1 1
1 0 0
- 1 1
1 0 0

Single:
10111110101010101010101010101010
Double:
1011111111010101010101010101010101010101010101010101010101010101

Problem 8 (10 points)

Consider 2 machines, A and B on which the following measurements are made for a certain program P.

A: Execution time of P: 5 sec
Instructions Executed: 5.4×10^8
CPI: 1.8

B: Execution time of P: 6.4 sec
Instructions executed: 115×10^6
CPI: 2.0

a. Find the execution rate (in Millions of Instructions per Second (MIPS)) for each machine.
b. Find the clock cycle for each machine.
c. Using the book's definition of performance measure, which machine is faster and by how much.

A:
Execution rate = (5.4 x 10^8)/5 = 1.08 x 10^8 / sec; 108 MIPS
Clock Cycle = 5.4 x 10^8 x 1.8 = 9.72 x 10^8
B:
Execution Rate = (115 x 10^6)/6.4 = 17.96875 x 10^6 / sec; 17.96875 MIPS
Clock Cycle = 115 x 10^6 x 2 = 230 x 10^6
Machine A is 6.010434782608695652173913043478 3 times faster.

Problem 9 (10 points)

Assume that a multiply instruction takes 12 cycles and accounts for 15% of the instructions in a typical program. Assume that 85% of the instructions require an average of 4 cycles for each instruction. What percentage of time does the CPU spend doing multiplication?

from 100 instructions, 15 multiplication instructions at 12 cycles = 180 cycles
85 instructions at 4 cycles = 340 cycles
percentage of multiplication = 180/340 = 0.529411764705882352941176470588 24
~52% of the time, CPU is doing multiplication.

Problem 10 (10 points)

Software optimization can dramatically improve the performance of a computer system. Assume that a CPU can perform a multiplication operation in 9ns and an addition or a subtraction in 1ns.

a. How long will it take for the CPU to calculate the result of following equation assuming that we only have 1 multiplier and 1 adder?

$$x = a * b * c * d + a * e$$

b. If possible, optimize the equation so that it takes less time. What is the best-case running time for the calculation of the equation? What if we have 2 multipliers and 1 adder?

x = 9ns + 9ns + 9ns + 9ns + 1ns = 37ns
x = a(b * c * d + e) = 9ns + 9ns + 1ns + 9ns = 28ns
2 multipliers and 1 adder:
x = a * b *c *d + a * e = 9ns + 9ns + 1ns = 19ns
x = a(b * c * d + e) = 9ns + 9ns + 1ns + 9ns = 28ns

BONUS QUESTION (10 points)

Provide the pseudocode for the addition of two natural numbers using only increment, decrement, and conditional looping instructions.

```
int addtwo(int a, int b)
{
        while (a > 0)
        {
                --a;
                ++b;
        }
        return b;
}
```