The University of Alabama in Huntsville
Electrical & Computer Engineering Department
CPE 431 01
Test 1 Solution
Fall 2008

1. (1 point) An _operating system_ is a program that manages the resources of a computer for the benefit of the programs that run on that machine.
2. (1 point) A _stack_ is a data structure for spilling registers organized as a last-in-first-out queue.
3. (1 point) When I say sources, you say _mux_ .
4. (1 point) A number in floating-point notation that has no leading 0s is said to be _normalized_.
5. (1 point) Performance is _inversely_ related to execution time.
6. (10 points) Show the single MIPS instruction or minimal sequence of instructions for this C statement:

```
x[4] = x[5] + a;
```

Assume that a corresponds to register $t3 and the array x has a base address of $64{,}000_{10}$ and that x has been declared with the following statement:

```
char x[100];

addi  $t0, $zero, 64000
lbu   $t1, 5($t0)
add   $t1, $t1, $t3
sb    $t1, 4($t0)
```

7. (10 points) In a magnetic disk, the disks containing the data are constantly rotating. On average it should take half a revolution for the desired data on the disk to spin under the read/write head. Assuming that the disk is rotating at 9600 revolutions per minute (RPM), what is the average time for the data to rotate under the disk head?

$$0.5rev \times \frac{1\min}{9600rev} \times \frac{60s}{1\min} = 3.125ms$$

8. (20 points) Consider two different implementations, I1 and I2, of the same instruction set. There are three classes of instructions (A, B, and C) in the instruction set. I1 has a clock rate of 6 GHz, and I2 has a clock rate of 3 GHz. The average number of cycles for each instruction class on I1 and I2 is given in the following table:

Class	CPI on I1	CPI on I2	C1 usage	C2 usage	Third-party usage
A	2	3	25%	30%	50%
B	5	1	30%	20%	25%
C	4	2	45%	50%	25%

The table also contains a summary of how three different compilers use the instruction set. C1 is a compiler produced by the makers of I1, C2 is a compiler produced by the makers of I2, and the other compiler is a third-party product. Assume that each compiler uses the same number of instructions for a given program but that the instruction mix is as described in the table. Using C1

on both I1 and I2, how much faster can the makers of I1 claim that I1 is compared with I2? If you purchase I1, which compiler would you use?

$CPI_{C1\text{-}I1} = 0.25*2 + 0.3*5 + 0.45*4 = 0.5 + 1.5 + 1.8 = 3.8$
$CPI_{C1\text{-}I2} = 0.25*3 + 0.3*1 + 0.45*2 = 0.75 + 0.3 + 0.9 = 1.95$

$$\frac{P_{C1-I1}}{P_{C1-I2}} = \frac{ET_{C1-I2}}{ET_{C1-I1}} = \frac{IC_{C1-I2} * CPI_{C1-I2} * CT_2}{IC_{C1-I1} * CPI_{C1-I1} * CT_1} = \frac{1.95 * \frac{1}{3\times10^9}}{3.8 * \frac{1}{6\times10^9}} = \frac{1.95*6}{3.8*3} = 1.03$$

So, for C1, I1 is 1.03 times as fast as I2.

For I1, all compilers use the same number of instructions and the clock rate is the same, so the defining factor is the CPI. Whichever compiler has the smallest CPI is the fastest.

$CPI_{C2\text{-}I1} = 0.3*2 + 0.2*5 + 0.5*4 = 0.6 + 1.0 + 2.0 = 3.6$
$CPI_{3P\text{-}I1} = 0.5*2 + 0.25*5 + 0.25*4 = 1.0 + 1.25 + 1.0 = 2.25$

Since the CPI for the third party compiler is the smallest, I would use it with I1.

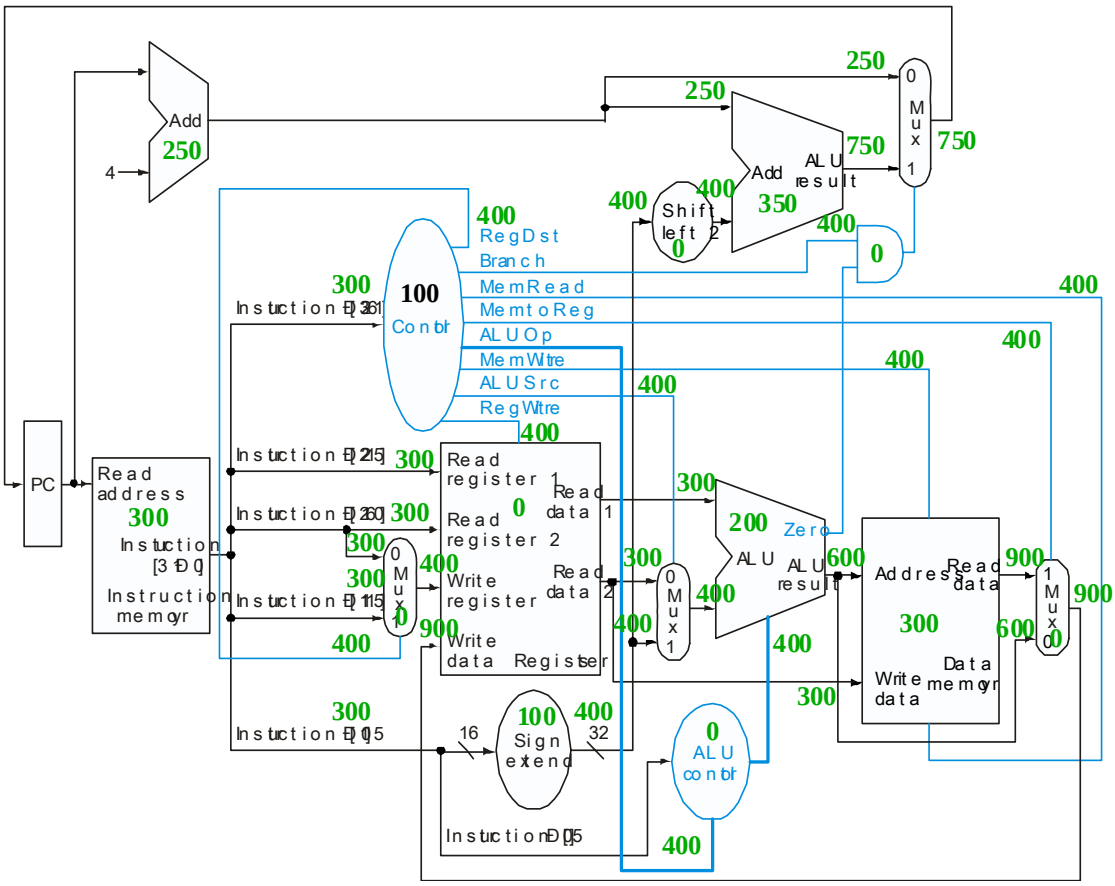
10. (15 points) In estimating the performance of the single-cycle implementation, assume the following delays:
    - ALU: 200 ps
    - adder for PC + 4: X ps
    - adder for branch address computation: Y ps
    - multiplexors, wires, and PC access: 0 ps
    - control unit and sign extension unit: 100 ps.
    - instruction and data memory: 300 ps

What would the cycle time be if X = 250 and Y = 350?

Using delay information shown on figure below and considering all instructions

beq	750 ps
R-type	600 ps
sw	900 ps
lw	900 ps

So, the cycle time must be as long as the longest instruction, so 900 ps.



11. (10 points) Consider program P, which runs on a 2 GHz machine M in 0.12 seconds. An optimization is made to P, replacing all instances of multiplying a value by 8 (mult X, X, 8) with three instructions that set X to X + X thrice (add X, X; add X, X; add X, X;) Call this new optimized program P'. The CPI of a multiply instruction is 12, and the CPI of an add is 2. After recompiling, the program now runs in 0.105 seconds on machine M. How many multiplies were replaced by the new compiler?

$IC_{multiply} = x$, $IC_{add} = 3x$. We can write two equations for execution time for the two cases.

(1) $0.12\ s = (IC_{other}*CPI_{other} + IC_{multiply}*CPI_{multiply})*0.5\ ns$
(2) $0.105\ s = (IC_{other}*CPI_{other} + IC_{add}*CPI_{add})*0.5\ ns$

Substituting,

(1) $0.12\ s = (IC_{other}*CPI_{other} + x*12)*0.5\ ns$
(2) $0.105\ s = (IC_{other}*CPI_{other} + 3x*2)*0.5\ ns$

Multiplying (2) by –1 and adding it to (1) yields

$0.015\ s = (12x - 6x)*0.5\ ns$

Then, $0.03*10^9 = 6x$ and $x = 5*10^6$

9. (15 points) Show the IEEE 754 binary representation for the floating-point number -47.75_{ten} in single and double precision.

2	47	R 1
2	23	R 1
2	11	R 1
2	5	R 1
2	2	R 0
2	1	R 1
	0	

0.75	x 2	1.50	1
0.50	x 2	1.0	1

$47.75_{10} = 101111.11_2 = 1.0111111 \times 2^5$

Single Precision:

Sign = 1, Fraction = 011 1111 0000 0000 0000 0000

Exponent = $5 + 127 = 132_{10} = 10000100_2$

1100 0010 0011 1111 0000 0000 0000 0000 = 0xC23F0000

Double Precision:

Sign = 1, Fraction = 0111 0100 0110 0110 0110 0110 0110 0110 0110 … 0110

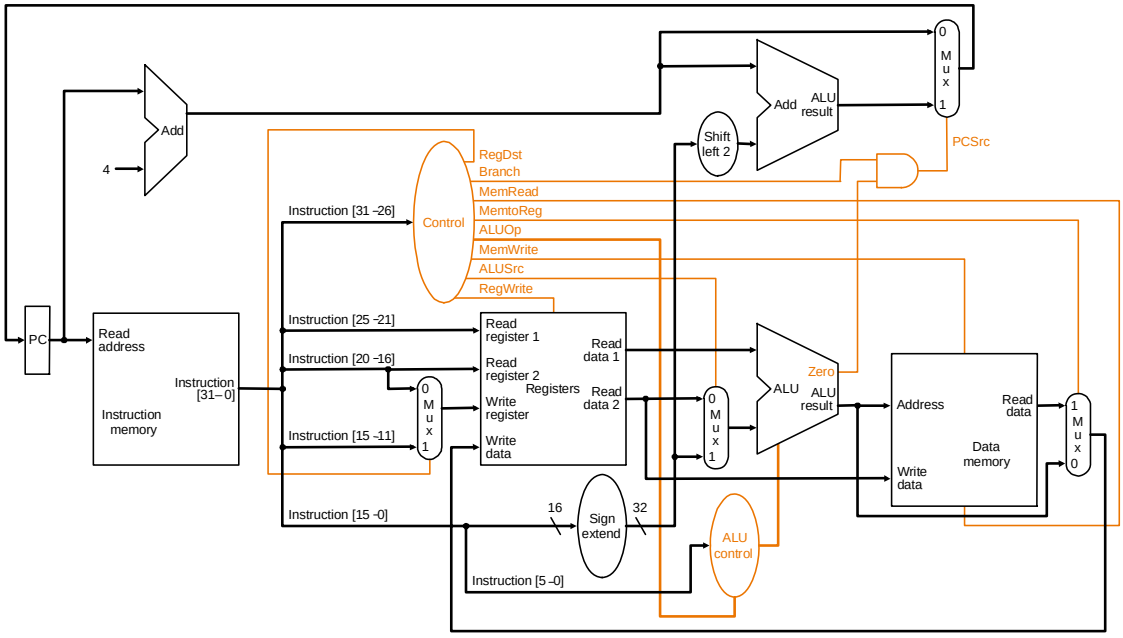
Exponent = $5 + 1023 = 1028_{10} = 10000000100_2$

1100 0000 0100 0111 1110 0000 0000 0000 … 0000 = 0xC047E00000000000

12. (15 points) Add a variant of the lw instruction which sums the contents of two registers to obtain the address of the data and which uses the R format to the single-cycle datapath shown in the figure below. Add any necessary datapaths and control signals and show the necessary additions to the table of control signals given.

```
lw   $9, $6, $22
```
$9 ← MEM[$6+$22]

36	6	22	9	0	32



Instruction	RegDst	ALUSrc	Memto-Reg	Reg Write	Mem Read	Mem Write	Branch	ALU Op1	ALU Op0	
R-format	1	0	0	1	0	0	0	1	0	
lw	0	1	1	1	1	0	0	0	0	
sw	d	1	d	0	0	1	0	0	0	
beq	d	0	d	0	0	0	1	0	1	
lw_mod	1	0	1	1	1	0	0	0	0	

d – don't care

No hardware modification is necessary.