BIRZEIT UNIVERSITY

ENCS336 – 1st Exam

Dr. Emad Hamadeh

1st Semester 08/09
Date: 28/10/2008

Student ID :_____

Student Name :_____SOLUTION_____

☐ **Section 1 (1:00PM SMW)**

☐ **Section 2 (8:00AM SMW)**

Instructions:

- **You have 90 minutes, so budget your time carefully!**
- **Turn OFF your mobile.**
- **To make sure you receive credit, please write clearly and show your work.**

Question	Maximum	Your Score
1	20	
2	25	
3	15	
4	15	
5	10	
6	15	
Total	100%	

Question 1 (20 marks**) :**

a) TRUE or FALSE :

TRUE	FALSE	
○	💣	PCI Express consists of lanes such that each lane is 32-bit wide
💣	○	Performance of x machine is better than y machine if y has more execution time than x
○	💣	Extended PCI bus has 32-bit data/address lines and twice the speed of standard PCI
○	💣	Number of address lines for a 256K x 16 memory organization is 16
○	💣	Moore's Law states that the number of transistors on a chip will double every two years
○	💣	ENIAC was a binary machine programmed manually by switches
💣	○	In Booth's Algorithm, the number of Shift Arithmetic Right operations is equal to the number of bits in M or Q
💣	○	A 4-bit binary decrementer can be implemented by adding **1111b** to the desired register
○	💣	All Intel x86 family share the same basic organization in order to maintain code backwards compatibility
○	💣	AR is a special register used to store the address of next instruction to be read from memory

b) What are the main types of interrupts? And what are some of the approaches for dealing with multiple interrupts?

Main Types:

1) Hardware, 2) software, 3) special events (div by 0…)

Dealing with Multiple INTs:

1) first come first served (ignore others),

2) set priority

Question 2 (25 marks**) :**

a) An instruction encoder uses 5 bits for encoding the opcode. One of the 5-bit patterns is used as a special case, in which 3 other bits decide the actual opcode. How many unique opcodes can this computer have? (3pt)

$2^5 - 1 + 2^3 = 39$

b) A user code consists of the following four sequenced RTL lines:

T_0: R2 ← NOT[R1]
T_1: R1 ← NOT[R1 + R2]

T_2: R2 ← ASR[R1] , R1 ← R1 U R2
T_3: R1 ← SHL[R1]

Assuming 8-bit registers, with **R1 = AAh**, write out the values for R1 and R2 after the execution of each RTL line? **NOT** is bit negation operation, **SHL** is Shift Left, and **ASR** is Arithmetic Shift Right operation. **All values should be in hexadecimal**. (10pt)

After T0: R1 = ____AA_____, R2 = _____55_____

After T1: R1 = ____00_____, R2 = _____55_____

After T2: R1 = ____55_____, R2 = _____00_____

After T3: R1 = ____AA_____, R2 = _____00_____

c) Calculate the CPI and then find the CPU execution time of a computer running at 500 MHz, and the following Instruction Class distribution for a program with 100 instructions:

Class 1:	**3 CPI**	**40%**
Class 2:	**5 CPI**	**20%**
Class 3:	**3 CPI**	**30%**
Class 4:	**1 CPI**	**10%**

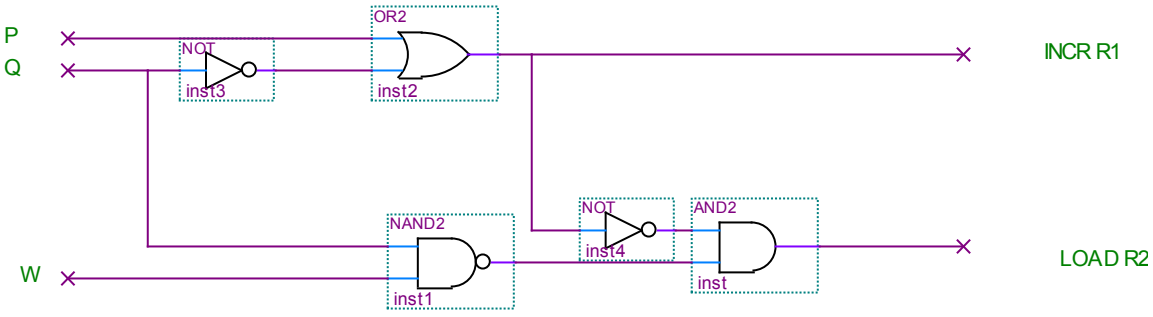
CPI = 3*0.4 + 5*0.2 + 3*0.3 + 1*0.1 = 3.2

E.T. = 100 * 3.2 / (500*10^6) = 640 ns

d) Draw the control logic (only) for the following RTL pseudo code:

If (P = 1 OR Q = 0) then R1 gets (R1 + 1)
Elseif (W = 1 NAND Q = 1) then R2 gets (R1 + R2)

Where P, Q, and W are the input signals to the control unit. (6pt)



Question 3 (15 marks):

a) Show the steps for doing **(- 8 x - 5)** using Booth's algorithm? Write final result. (10pts)

A	Q	Q_{-1}	M	Comments
00000	11011	0	11000	Initial Values

Result = AQ = 00001 01000 = 40

b) How many addition and how many subtraction operations are needed when using Booth's algorithm to multiply a number X by the multiplier **"11101001110b"**

Number of Addition Operations: _____2_____

Number of Subtraction Operations: _____3_____

Just count the pairs of "01" for addition and "10" for subtraction

Question 4: (15 marks)

a) Given the 32-bit pattern

1011	1111	1100	1000	0000	0000	0000	0000

What is the value in **decimal** if the above pattern is for:

I) Unsigned integer? 2pts

$2^{31} + (2^{30} - 2^{22}) + 2^{19} = 3{,}217{,}555{,}456$

II) Two's complement integer? 3pts
Sign bit is 1, negative

0100	0000	0011	1000	0000	0000	0000	0000

$= - (2^{30} + (2^{22} - 2^{19}) = - 1{,}077{,}411{,}840$

III) Single format floating point number? 5pts

Sign = 1, i.e. negative
Biased Exponent = $(2^7 - 2^0) = 127$
True exponent = 127 – 127 = 0
Fraction = $2^{-1} + 2^{-4} = 0.5625$

Value = - 1.5625 x 2^0 = -1.5625

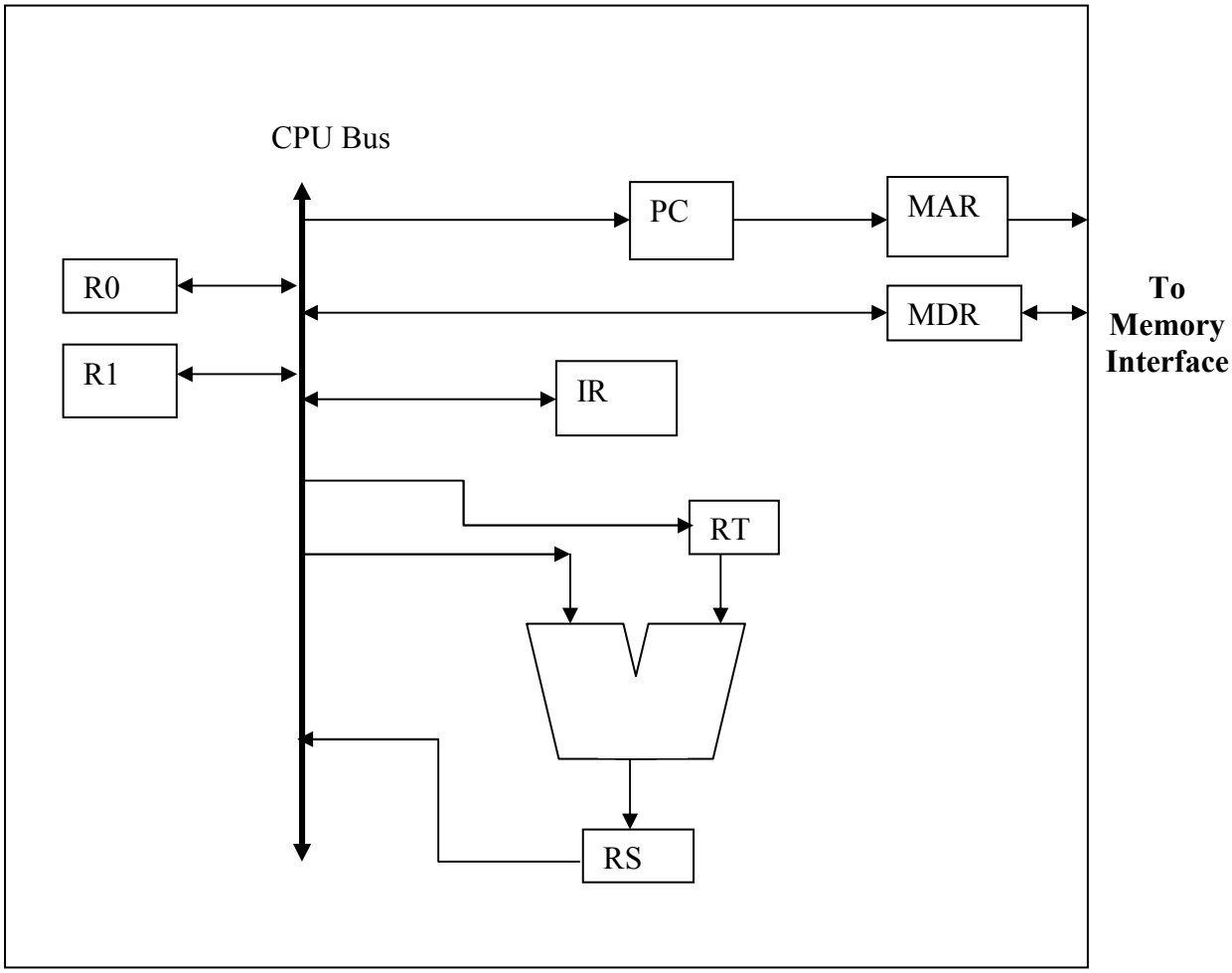
b) Which is more difficult to implement; floating point addition or floating point division operations? Why?

Floating point Addition
Need alignment of exponents

Question 5: (10 marks)

Draw a block diagram of the data path for a CPU that has the following hardware components: 2 general purpose registers **(R0, R1)**, Program Counter **(PC)**, Instruction Register **(IR)**, an **ALU** that can perform logical and arithmetic functions, Memory Address Register **(MAR)**, Memory Data Register **(MDR)**, a register that temporarily holds one operand for the ALU **(RT)**, and a register to temporarily hold the result from the ALU **(RS)**.

Use a single CPU bus, drawn below, to interconnect the hardware resources.



Question 6 (15 marks):

A **4K x 16** memory, shown below, is used to store instructions set and data for a basic computer architecture based machine. The first instruction is stored at memory address location **FF0h**;

a) fill in the table below with the values for **PC**, **IR**, and **AC**? Initial values mean the values of the registers prior to fetching the first instruction from memory.

Opcode **1h** is to load AC from memory
Opcode **3h** is to store AC to memory, while
Opcode **5h** is to add to AC from memory.
(All numbers are in hex representation)

Instruction Phase	PC	IR	AC
Initial Values	FF0	--	--
After 1st Instruction Fetch	FF1	1FF3	--
After 1st Instruction Execution	FF1	1FF3	3402
After 2nd Instruction Fetch	FF2	5401	3402
After 2nd Instruction Execution	FF2	5401	8402

Memory Address	Instructions & Data
:	:
400	0 A C E
401	5 0 0 0
402	3 1 0 0
403	0 F F 1
:	:
FF0	1 F F 3
FF1	5 4 0 1
FF2	3 F F 1
FF3	3 4 0 2
:	:

b) Using the above memory organization, what is the value of the operand stored in memory if the instruction format indicated an **indirect memory access** with an **address field** value of **403h**?

5401