DEGREE EXAMINATIONS — 2000   COMP203

END-YEAR

COMP 203

Computer Organisation
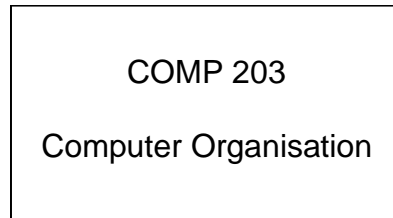
Time Allowed: 3 Hours (180 minutes)

Instructions: Answer all questions.
Make sure your answers are clear and to the point.
Calculators and foreign language dictionaries are allowed.
No reference material is allowed.
There are 180 possible marks on the exam.

NOTE:

There are two parts to this exam, Part I and Part II.
Please use separate answer booklets for each of the two parts, and
clearly label answer booklets either Part I or Part II.

	Topic	Marks
PART I		
1	General	*20 marks*
2	Machine Language	*20 marks*
3	Arithmetic	*20 marks*
PART II		
4	Logic Design Basics	*20 marks*
5	Processor Data Path and Control	*20 marks*
6	Pipelined Data Path	*30 marks*
7	Memory Hierarchy	*20 marks*
8	Input/ Output	*30 marks*

PART I

Question 1: General   [20 marks]

A. Give *brief* (one sentence) definitions for the following terms:
1. Arithmetic and logic unit (ALU)
2. Central processing unit (CPU)
3. Operating system

[9 marks]

B. A chip manufacturer is advertising a brand new design of microprocessor. The manufacturer's advertisement claims that its processor runs at 600 MIPS (million instructions per second), which is more than twice the speed of the comparable Intel processor. The advertisement also says that the new processor runs an unspecified benchmark program three times faster than the Intel processor.

Assume that the claims are not lies. Why should you still be sceptical about the validity of the comparison?

[6 marks]

C. The first stage in the process of fabricating an integrated circuit (IC) is growing the silicon crystal. What other stages are there in the process?

[5 marks]

Question 2: Machine Language   [20 marks]

A. The following three MIPS instructions can all be used to transfer control to a label Dest in another part of the program:

beq $zero, $zero, Dest	j Dest	jr $t0

Assume that the address of **Dest** is already loaded in register $t0.

Compare the control transfer instructions in these fragments, discussing:
- Instruction format
- Addressing mode
- How far the control can be transferred.

[6 marks]

B. How does the MIPS convention for procedure call specify how the responsibility for saving registers is divided between the caller and the callee? Why is it specified this way?

[6 marks]

C. Consider the following MIPS program:

```
        addi $t1, $zero, 1
loop:   add $t1,$t1,$t1
        addi $t0, $t0, -1
        bne $t0, $zero, loop
```

1. Suppose $t0 contains a positive integer input and $t1 is the output. What function does the program compute?
2. How many instructions will be executed, if $t0 contains the value 4 initially?
3. Rewrite the line of the program labelled **loop** so that it uses a shift instruction in place of **add**.

[8 marks]

Question 3: Computer Arithmetic   [20 marks]

A. Perform the following multiplications using IEEE binary floating point, with one sign bit, four exponent bits (bias 7), and three bits for the significand. In each case, express both operands in normalized binary floating point, perform the multiplication, and then renormalize the result, remembering to check the sign.
   1. 2 x 3
   2. 1.5 x 4
   3. 3 x 6

                                                              [9 marks]

B. A one-bit adder may be drawn as a rectangle with an appropriate number of inputs and outputs. Draw a diagram showing how three 1-bit adders may be connected to produce a 3-bit ripple-carry adder. Be sure to label every input and output line.

                                                              [5 marks]

C. Suppose we need to add two signed 32-bit numbers, both of which are in MIPS registers, and we need to detect whether an overflow has happened. How can we do this
   1. Using the **add** instruction?
   2. Using the **addu** instruction?

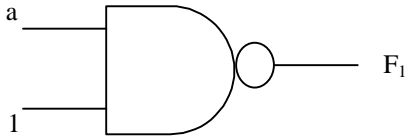   **[6 marks]**
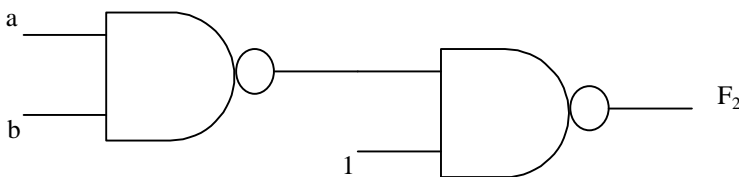
PART II

Question 4. Logic Design Basics　　　　[20 marks]

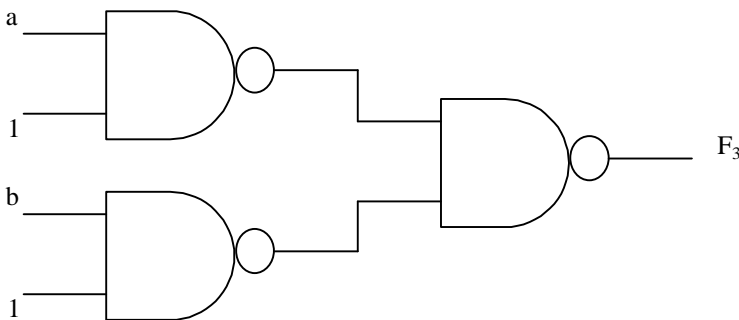**A.** Which logic functions are implemented by following NAND logic gates:

A1.

a ———|
　　　 | NAND ○——— F_1
1 ———|

　　　　　　　　　　　　　　　　　　　　　　　[2 marks]

A2.

a ———|
　　　 | ○——|
b ———|　　 | ○——— F_2
　　　 1 ——|

　　　　　　　　　　　　　　　　　　　　　　　[4 marks]

A3.

a ———|
　　　 | ○——|
1 ———|　　 |
　　　　　　 | ○——— F_3
b ———|　　 |
　　　 | ○——|
1 ———|

　　　　　　　　　　　　　　　　　　　　　　　[4 marks]

Express answer in simplest form!

Hint: Evaluate logic functions by applying gate operation on inputs and by "barring" the gate outputs, then apply DeMorgan's Theorems and Identity Law to achieve the final result.

B. Build an OR logic function by using two-input NOR gates.

　　　　　　　　　　　　　　　　　　　　　　　[10 marks]

Question 5. Processor Data Path and Control   [20 marks]

A. Consider the following single cycle data path (this is a copy of a figure from your Comp203 Study Guide). The figure is reproduced on a full page at the back of the examination paper, as well.

(Have a look at Figure 5.19 on page 360 of the textbook.)

Suppose that one of the processor control signals (RegDst, ALUSrc, MemtoReg, RegWrite, Zero) is stuck at 1. This will cause some instructions to fail, while the others will still work. Copy the table Q5A from below into your answer booklet, and for each of the four instructions, and five control signals, place a + (plus) in the corresponding table box, if the instruction will still work with that signal stuck at 1. (Note that RegWrite = 1 allows writing into register file.)

[10 marks]

Table Q5A

	RegDst	ALUSrc	MemtoReg	RegWrite	Zero
add $2, $2, $3					
sw $10, 0($2)					
lw $10, 0($2)					
beq $10, $0, L					

B. The microinstructions of the multicycle processor data path are given in the table Q5B. Write the micro code sequence for the MIPS sw instruction.

[10 marks]

Table Q5B

Label	ALU	SRC1	SRC2	Reg. Con.	Memory	PC Write	Sequencing
Fetch	Add	PC	4		Read PC	ALU	Seq
	Add	PC	Extshft	Read			Dispatch1
Mem1	Add	Rs	Extend				Dispatch2
LW2					Read ALU		Seq
				Write Rt			Fetch
SW2					Write ALU		Fetch
Rfrmt1	Fun-code	Rs	Rt				Seq
				Write ALU			Fetch
BEQ1	Subt	Rs	Rt			Zero	Fetch
JUMP1						Jumpaddr	Fetch

Question 6. Pipelined Data Path   [30 marks]

The pipelined data path stages when a source register is needed and when the result is delivered (or the instruction is finished) for some MIPS instructions are given in the table Q6A.

Table Q6A

	Stage when source needed	Stage when result delivered (or instruction finished)
Add	2	5
Sw	2	4

Consider the following code:

add $2, $5, $4
add $4, $2, $5
sw $5, 100($4)
add $3, $2, $5

A. Identify all of the data dependencies in the code given, and indicate how many nops are required to clear them.

[5 marks]

B. Assuming that register $2 had the value 0 before the first instruction and the value 10 after, describe what would happen if the data path could not detect hazards.

[5 marks]

C. How does a data path detect a hazard, and what techniques can be used to overcome them?

[5 marks]

D. What kinds of hazards could be resolved via forwarding (as you have seen in lab3)?

[5 marks]

Question 7. Memory Hierarchy   [20 marks]

The eight slot, 4 words per block direct mapped cache is presented in the following figure.

Slot	V	Tag	Word0	Word1	Word2	Word3
000						
001						
010						
011						
100						
101						
110						
111						

Suppose the cache is initially empty, and then the processor issues the following sequence of byte addresses:

111110001111
111110000000
111110000101
110110001111
110110000000

A. Show the contents of V and Tag columns after all these 5 processor requests.

[5 marks]

B. Calculate the miss rate after these 5 processor requests.

[5 marks]

C. How big is the first miss penalty, expressed in the number of processor cycles, if DRAM delivers a word in 5 processor cycles, and there is no bursting?

[5 marks]

D. How big is the first miss penalty, expressed in the number of processor cycles, if DRAM delivers a word in 5 processor cycles, and the DRAM burst rate is 5/1/1/1?

[5 marks]

Question 8. Input/Output [30 marks]

A. A program repeatedly performs a three-step process:
1. It reads in a 4-KB block from disk,
2. Does some processing on that data, and then
3. Writes out the result as another 4-KB block elsewhere on the disk.

Each block is contiguous and randomly located on a single track on the disk. The disk drive rotates at 7200RPM, has an average seek time of 8 ms, and has a transfer rate of 4 MB/sec.

The controller overhead is 1.55 ms.

No other program is using the disk or processor, and there is no overlapping of disk operation with processing.

The processing step takes 200,000 clock cycles, and the processor clock rate is 400MHz.

What is the overall speed of the system in blocks processed per second?
[10 marks]

B. Controlling data transfer between disk and RAM is implemented:
- in one case by the processor, using a polling algorithm, and
- in the other case by using a direct memory access (DMA) controller.

The disk controller represents the only difference between the two cases.

Suppose:
- The disk transfers data in 4 word (16 byte) chunks.
- The disk transfer rate is 4 MB/s.
- There is one page of 4KB transferred in every disk I/O operation.
- The disk drive rotates at 7200RPM.
- The disk has an average seek time of 8 ms.
- The disk controller overhead is 1.55 ms.
- Processor clock frequency is 400MHz.
- All not mentioned delays and overheads are negligible.
- Note that disk and disk controller performance figures are the same as in question 5A. of this Final Examination.

B1. Compute the processor overhead during transfer of 1 page between disk and RAM that is controlled by polling, if:
- One polling operation (including control of data transfer) needs 400 processor cycles, and
- No data transfers can be missed.

[10 marks]

B2. Compute the processor overhead during the transfer of 1 page between disk and RAM that is controlled by the DMA controller, if:
- DMA controller set up for the transfer of one page takes 1,500 processor clock cycles, and
- Handling the interrupt at the completion of the page transfer takes 500 processor clock cycles.

[10 marks]

(Express the processor overhead as the ratio between number of clocks needed to execute all necessary operations related to disk I/O control in a unit of time (e.g. the time of one disk I/O), and the number of clocks available in the same unit of time)

<p style="text-align:center">******************************</p>

COMP 203   **end**