UNIVERSITY OF COLOMBO, SRI LANKA

UNIVERSITY OF COLOMBO SCHOOL OF COMPUTING

**DEGREE OF BACHELOR OF INFORMATION TECHNOLOGY (EXTERNAL)**
***Academic Year 2008/09 – 3rd Year Examination – Semester 6***

# *IT6503: Computer Systems II*

***Structured Question Paper***
**13th September, 2009**

***TWO HOURS***

**To be completed by the candidate**

BIT Examination Index No: _____

**Important Instructions:**

- The duration of the paper is 2 **(Two) hours**.
- The medium of instruction and questions is English.
- This paper has **4 questions** and **pages**.
- **Answer all 4 questions: .**
- **Write your answers** in English using the space provided **in this question paper**.
- Do not tear off any part of this answer book. Under no circumstances may this book, used or unused, be removed from the Examination Hall by a candidate. Note that questions appear on both sides of the paper. If a page is not printed, please inform the supervisor immediately.
- **Non-programmable calculators may be used.**

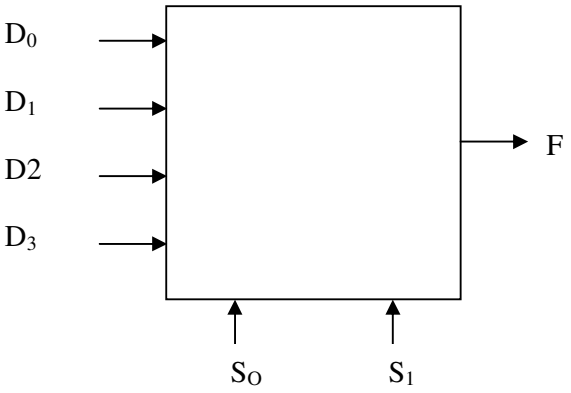
**Questions Answered**
Indicate by a cross (×), (e.g [X] ) the numbers of the questions answered.

| **To be completed by the candidate by marking a cross (×).** | 1 | 2 | 3 | 4 | |
|---|---|---|---|---|---|
| To be completed by the examiners: | | | | | |
| | | | | | |
| | | | | | |

1) (a) Draw the block diagram of a 4:1 multiplexer. Clearly identify the inputs $D_0$, $D_1$,$D_2$,$D_3$, select lines $S_0$ and $S_1$ and the output F.

**(Marks 4)**

**ANSWER IN THIS BOX**




(b) Write down the truth table for the multiplexer.

**(Marks 3)**

**ANSWER IN THIS BOX**

| $S_0$ | $S_1$ | F |
|---|---|---|
| 0 | 0 | $D_0$ |
| 0 | 1 | $D_1$ |
| 1 | 0 | $D_2$ |
| 1 | 1 | $D_3$ |

(c) A three variable 'majority function' logic circuit is to be designed where the output will be logic 1 if the majority of inputs are logic 1, or the output will be logic 0, if the majority of inputs are logic 0. Write down the truth table for the majority function logic circuit.

**(Marks 4)**

**ANSWER IN THIS BOX**

| a | b | c | f |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 0 |
| 0 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 |
| 1 | 0 | 1 | 1 |
| 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 |

(d) Obtain the simplified Boolean expression for the majority function logic circuit.

**(Marks 3)**

**ANSWER IN THIS BOX**

**F = $\bar{a}$.b. c + $a.\overline{b}.c$ + $a.b.\overline{c}$+ a.b.c**

**= b.c + a($\overline{b.c}$ +$b.\overline{c}$)**

**= b.c + a (b $\oplus$ c)**

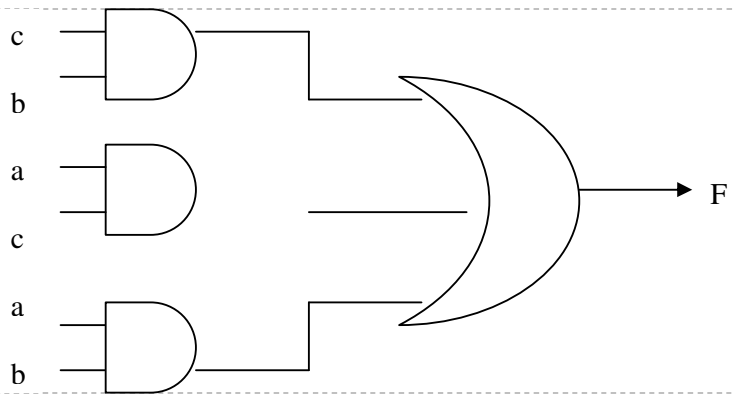
**Or**

**F = c.b + a.c + a.b**

(e) Draw the combinational logic diagram that corresponds to the expression in (d).

**(Marks 3)**

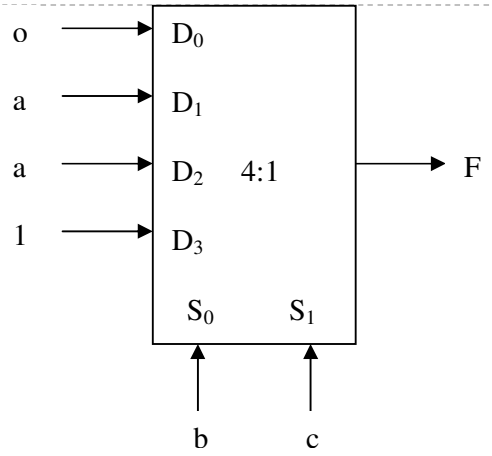
**ANSWER IN THIS BOX**



(f) A multiplexer may be effectively used to replace the combinational logic in many instances. Implement the majority function logic circuit of (c) using a 4: 1 multiplexer and a minimum of extra logic.

**(Marks 8)**

**ANSWER IN THIS BOX**



2 (a) Fill in the blanks in the table with the most suitable option given within brackets.

**(Marks 6)**

**ANSWER IN THIS BOX**

| CPU architecture / property | Load/Store (risc) | Register/Memory (cisc) | stack |
|---|---|---|---|
| Size of general purpose register set (small, large, none) | **Large** | **Small** | **None** |
| Instruction length (constant, variable) | **Constant** | **Variable** | **Variable** |
| Compiler complexity (low, high) | **High** | **Low** | **Low** |
| Performance of general purpose applications (low, high) | **High** | **Low** | **low** |

(b) (i) The cisc (register-memory) machine instruction `SWAP [M1],[M2]` exchanges the contents of memory locations $M_1$ and $M_2$. Write down the equivalent risc (Load/Store) machine instructions to this cisc instruction.

**(Marks 3)**

**ANSWER IN THIS BOX**

**LD $R_1$,$M_1[R_0]$**

**LD $R_2$ , $M_2[R_0]$**

**ST $M_1[R_0]$ , $R_2$**

**ST $M_2[R_0]$, $R_1$**

**Where $R_0$ = 0 assumed**

(ii) Write down the sequence of equivalent risc machine instruction to the following cisc instruction.

```
ADD R2, 100[R1++]
```

**(Marks 3)**

**ANSWER IN THIS BOX**

**LD $R_3$ , 100[ R1]**

**ADD $R_1$, $R_1$, # 1**

**ADD $R_2$,$R_2$,$R_3$**

(iii) When a certain high level language code fragment is compiled it produces the following set of risc machine instructions.

```
loop:     SUB R1, R1, #1
          BEQZ R1, label
          ADD R2, R2, #1
          JMP loop
          ……..
label:    ……..
```

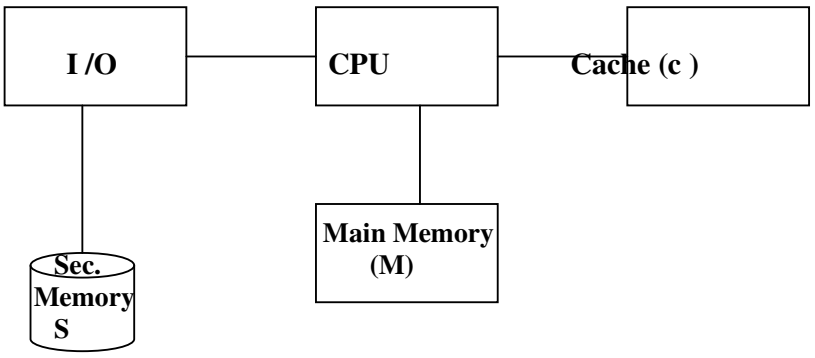
Assuming R1 and R2 have been assigned for variables p and q, write down the high level language code that corresponds to the above machine code.

**(Marks 4)**

**ANSWER IN THIS BOX**

**While (-- p>0)**
**q ++;**

(c) The block diagram of a typical computer system is shown in Fig. 1.




**( Fig. 1)**

(i) With reference to Fig.1, and the two level memory consisting of a cache with latency 1.5nsec and a hit ratio of 0.95, and a main memory with an access time of 1.5 μsec, calculate the average memory access time of the combined memory.

**(Marks 4)**

**ANSWER IN THIS BOX**

**t avg = hc tc + (1 – hc)(tc +tm)**

**= 0.95 x 1.5 ns + 0.05 x ( 1.5 ns + 1.5 μ s)**

(ii) As shown in Fig. 1,a data item can reside in one or more of three places:cache(C ), main memory(M) and the secondary memory(S). State two CPU actions that can cause memory inconsistency.

**(Marks 3)**

**ANSWER IN THIS BOX**

1. **CPU update only cache via write update mode**
2. **via DMA, main memory update S or, vice versa**

(iii) Compare the effects of a cache miss against a page fault on an executing process.

**(Marks 2)**

**ANSWER IN THIS BOX**

1. **cache miss does not stop executing, process; CPU recovers cache blocks from M to C**
2. **page fault causes executing, process to stop under OS control; new process starts executing**

3 (a) A programmer writes a parallel algorithm for a certain computational task and runs it on a 4, 8 and 16 node multiprocessor respectively. He does not observe a x4, x8 and x16 fold performance gain compared to the serial code as he expected, in these configurations. Give two reasons for this observation.

**(Marks 6)**

**ANSWER IN THIS BOX**

1**. According to Amdahl's law, the fraction of inherent parallelism in a task affects the speedup. If it is 100% then linear spreading**

**2 In a multiprocessor, the communication over head between processors lower the speed up**

(b) (i) By the use of instruction pipelining a processor aims to achieve a (cycles per instruction) CPI of 1. To improve the CPU performance further, should one decrease or increase the CPI?

**(Marks 2)**

**ANSWER IN THIS BOX**

**Decrease**

(ii) What architectural feature change will enable the CPU performance to be enhanced?

**(Marks 3)**

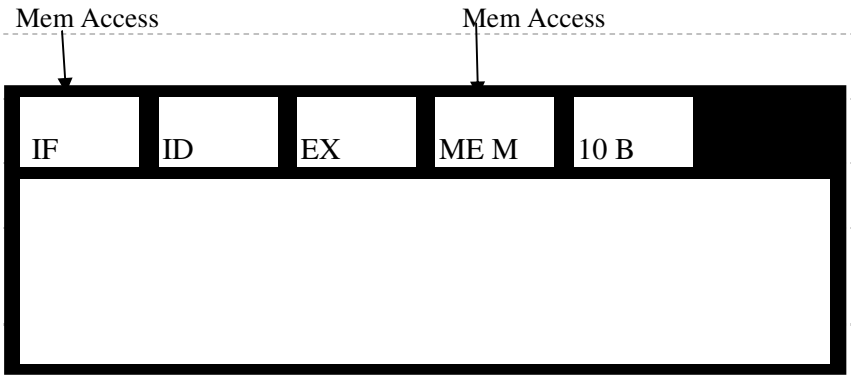
**ANSWER IN THIS BOX**

**multiple issue pipelines ( super scaling)**

(c)(i) Draw a sketch of a typical 5-stage risc instruction pipeline and identify (by marking) the stage(s) where memory access is made.

**(Marks 5)**

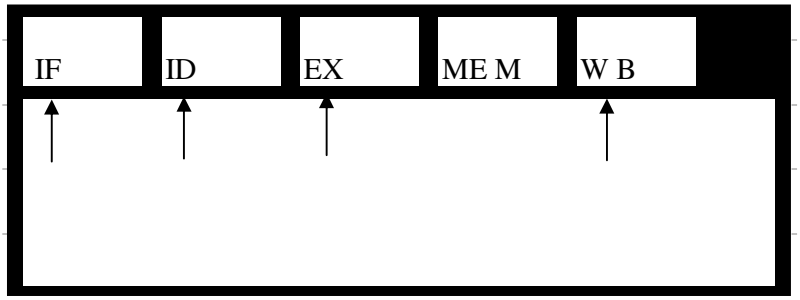
**ANSWER IN THIS BOX**



(ii) Identify and mark in a typical 5-stage risc pipeline, the stages that are active during the execution of an risc arithmetic/logic instruction.

**(Marks 3)**

**ANSWER IN THIS BOX**




(iii) The following Load/Store (risc) code executes in a 5-stage pipeline. Draw a space-time diagram and show the instances where there can be data hazards.

```
loop:   LD R1, 100[R2]
        ADD R1, R1, R3
        ST 100[R2], R1
        ADDI R2, R2, #4
        SUB R4, R5, R2
        BNEZ R4, loop
```

**(Marks 6)**

**ANSWER IN THIS BOX**

LDR1, . .: IF ID EX MEM 10 B

ADD R : IF ID EX MEM 10 B

ST : IF ID EX MEM 10 B

ADD I : IF ID EX MEM 10 B

SUB : IF ID EX MEM 10 B

4 (a) Using a suitable diagram show the process states and their transitions, that exisits in a muti – tasking kernel.

**(6 marks)**

**ANSWER IN THIS BOX**

**Process states:**

**Running: Instructions of process are being executed.**

**Waiting: The process is waiting for some event to occur (such as an I/O completion or reception of a signal)**

**Ready: The process is waiting to be assigned to a processor.**

**Terminated: The process has finished execution.**

**Diagram:**

(b) Explain the result of executing a fork() statement in a running process.

**(2 marks)**

**ANSWER IN THIS BOX**

**It will create a new process that will be added to the existing process list in the computer. That process will start execution the rest of the code in the program after the fork(statement).**

(c) (i) Explain what is meant by a "deadlock" in a multitasking operating system.

**(3 marks)**

**ANSWER IN THIS BOX**

**When a process requests a resource, if that resource is not available at that time, the process enters a "wait" state. It may happen that a waiting process will never change that wait state, because the resource that it requested is held by another waiting process that is waiting for a resource that is presently held by the first process.**

(ii) Describe the four necessary conditions that must hold if a deadlock situation is to occur.

**(5 marks)**

**ANSWER IN THIS BOX**

Mutual exclusion

Hold and wait

No preemption

Circular wait

(iii) With respect to the above conditions (that you listed in ii above) how can an operating system prevent the occurrence of a deadlock?

**(1 mark)**

**ANSWER IN THIS BOX**

**By ensuring that at least one of the conditions in ii cannot hold, the OS can prevent deadlocks from occurring.**

(d) Describe "round robin" process scheduling.

**(3 marks)**

**ANSWER IN THIS BOX**

**Here each process is given a time quantum to run on the computer. After a process has run its quantum, another READY process is chosen to run and is given its quantum. Similarly all the READY processes are given a quantum each to run on the computer. However, the processes are free to release the CPU voluntarily before its full quantum is run.**

(e) Including the terms "pages", "frames" and "page tables", explain the concept of "virtual memory".

**(4 marks)**

**ANSWER IN THIS BOX**

**The concept of virtual memory let programs that are bigger than the physical memory of a computer run on it.**

**The physical memory of a computer is divided into equal fixed-size chunks called "frames" and each process is also divided into small fixed-size chunks of the same size called "pages". Then the pages are assigned to available frames in memory. A list of free frames is maintained by the OS. OS also maintains page tables for each process. The "page table" shows the frame location for each page of the process.**

f. Is there a limit to the size of a virtual memory of a computer? Explain.

**(1 mark)**

**ANSWER IN THIS BOX**

**Yes. It will be limited by the maximum virtual address that the computer can use.**

**********