INDIAN INSTITUTE OF TECHNOLOGY, KHARAGPUR

End Autumn Semester Examination 2000-2001

Subject No. 173001 **Subject Name : Computer Org. & Arch - I**

Time : 3 Hrs. **Full Marks: 100**

Date of Examination : 28th Nov. 2000 (*FN*)

No. of Students :50 3rd Yr *CSE Dept.*

This paper contains five(5) *pages* and a TOTAL of seven(7) questions.

Instructions: *Answer* Question 1. & *any* four (4) questions from the rest i.e. a total of five (5) questions. Marks *are indicated* at the *right hand bottom corner* of each question.Make your own assumptions wherever necessary *stating them* at relevant places.

1. (compulsory)

(a) Justify or **contradict** *each* of the following statements with **proper reasoning** and/or *examples* in each case. Merely writing **TRUE/FALSE** otherwise YES/NO **will not fetch** any credit. **[10 × 1 = 10]**

i. CACHE memories *must possess* dedicated bus connectivity to the CPU.

ii. More GPRs inside a CPU speeds up the data processing activity.

iii. Associative access happens to be the costliest form of data search in memory.

iv. Machine OP Code Length & Data Bus width cannot be related.

v. A hardwired control unit occupies more space compared to a microprogrammed control unit.

vi. A stack reference instruction at the assembly language level need not possess any memory operand.

vii. TLB Cache is not relevant in segmented memory management.

viii. Machine Code version of the same high level program in RISC machine will be lengthy than its CISC counterpart.

ix. Virtual memory management is needed to accomodate the CPU physical address space in the existing RAM space which always happens to be smaller.

x. All the OS modules are loaded on system demand from secondary store to main memory due to internally/externally generated interrupt.

(b) Each statement below has got **blank space(s)**. From the *list of phrases* given **below**, choose the **most appropriate** one and to fill up *each* **blank space.** You must **rewrite** the *complete statement* **clearly indicating the filled up blank space(s)** in *each* in order to obtain **full credit.**

(a) 2-D **(b)** UV erasable **(c)** microinstructions **(d)** not **(e)** Addition
(f) hardware **(g)** high level **(h)** firmware **(i)** software
(j) main memory **(k)** 2 port **(l)** contribute **(m)** on chip
(n) indirect addressing **(o)** electronic **(p)** directly accessible
(q) register transfer **(r)** microprogram **(s)** display refreshing **(t)** reloading

[20 × 0.5 = 10]

i. ... Subtraction operations are ... sign sensitive.

ii. A ... memory organization saves both ... and off chip ....

iii. EPROM happens to be ....

iv. Each machine instruction can be implemented by a ... composed of one or many ...

v. A pointer in a ... language can be implemented using an ... mode at the assembly level.

Contd..2

Q. 1(b) (Contd.)

vi. All ... memories happen to be randomly accessible while Disks are ....

vii. Cache memory does not ... to ... space.

viii. Video RAM module is made into a ... one in order to do ... and ... data from main memory together.

ix. A microoperation is actually a ... operation.

x. Instruction register is a piece of hardware holding a ... activating a ... in the control unit.

(c) Specify possible data transfer modes along with proper justifications in each case, to interface the following devices to a CPU.

i. A Dynamic *RAM* which is *twice as slow as* the CPU.

ii. A **disk drive** whose data transfer rate is **10** *times faster* than the CPU.

[4 + 4 = 8]

2. Consider a simple 16 bit word CPU of a 12 bit *Main Memory address* machine having the following assembly language instruction set and their corresponding *interpretations.*

1.	ADD X	ACC ← ACC + M [X] M [X] content of the memory location X
2.	AND X	ACC ← ACC ↑ M [X]
3.	CMA	ACC ← Complement of ACC
4.	LOAD X	ACC ← M [X]
5.	STORE X	M [X] ← ACC
6.	JMP X	Jump to the location X in Memory
7.	JZ X	if Result =0 then jump to X
8.	RSHIFT	Right Shift ACC

Assume the following :

- Each memory location is 16 Bit wide.
- Each *OP-Code* occupies 4 Bits.
- Each Address Operand takes up 12 Bits.
- There exists only one *General Purpose Register (GPR)* **ACC** inside CPU.
- Only **one internal Bus** of 16 Bit wide exists inside CPU.
- It employs a **microprogrammed control unit.**

(a) Specify the suitable **Data Path** for the above CPU alongwith the Main Memory interface. [4]

(b) Specify the *Micro-operations*, as well as the corresponding micro instructions OR (**control words**) associated with each of the following assembly instructions:

i. *ADD X.*

ii. *JZ X.*

[4 + 4 = 8]

(c) Identify the possible type of exceptions that can occur in executing any instruction of this CPU. How are these exceptions handled in any **generalised CPU**? Discuss briefly. [1 + 3 = 4]

(d) Without making any changes in the CPU architecture, mention *at least* **one more instruction** that can be included in the INSTRUCTION SET of the above-mentioned CPU alongwith *brief* **justification.** [2]

Contd..3

4. You are to implement the following high level programming language *features* in an ASSEMBLY INSTRUCTION SET in the most efficient manner.

(a) A *2-Dimensional*, static sized signed Integer *Array*.

(b) A *linear*, singly *linked list* of real numbers.

(c) A post increment operator.

Pickout the *most appropriate* ADDRESSING MODES that should be supported by the underlying CPU in order to *facilitate* your implementation of each of the above high level language features. You must elaborate the following things too :

(a) Representation of the *associated* data types.

(b) Justification of your choice.

(c) Block schematic of the address generation scheme in *each* of your chosen ADDRESSING MODES, clearly *highlighting* the hardware supports that are needed in *each* case.

[(4 + 3) + (3 + 4) + 4 = 18]

5. A computer employs 8bit data Bus, *RAM* chips of 256 bytes and ROM chips of 1K bytes. The computer system needs 2K bytes of *RAM*, 4K bytes of ROM and four (4) interface units, *each* with four(4) 8 bit registers. A memory mapped I/O configuration is used.

(a) How many address lines should be there in the CPU assuming that the following selection mechanism is adopted in the address?

[3]

00	*RAM* bank
01	ROM bank
10	*Interface Registers*
11	Not used

(b) How many *RAM* and ROM chips are needed ? *Clearly depict* all the calculation steps.

[3]

(c) Show by *neat* schematic diagramsin *each case*, the interface details between the CPU and the following:

i. A *RAM* chip.

ii. A ROM chip.

iii. One interface unit.

[3 + 3 + 2 = 8]

(d) Specify the address range in Hex for *RAM*, ROM and interface units. **[4]**

Contd..4

5. (a) A cache memory organization using one level of cache has the following characteristics:

   i. The cache access time is 10 *nsec.*

   ii. Main Memory access time is 200 *nsec.*

   iii. **Write through** technique is employed.

   iv. 70% of memory references are **read** rest 30% are **write.**

   v. The **hit percentage** is **80%** for **read access** and **70%** for **write access.**

   Compute the **average memory access time.** *Clearly show* all **calculation steps** with *explanation.* [6]

   (b) From the specifications given later, explain with **schematic diagrams** alongwith the **clear depiction** of the various fields in the **memory address**, how the cache memory can be *implemented* by using the following schemes. $[(3 + 1) \times 3 = 12]$

   i. Associative Mapping.

   ii. Direct Mapping.

   iii. 8 way set associative mapping.

   Memory Specifications:

   - Size = **4 M Byte**
   - Word Size = **4 byte** but **individual bytes** *can be accessed.*
   - Cache memory size = **64k bytes.**
   - Cache Line Size = **16 bytes** each.

6. (a) i. Suppose you are provided with a data path having the following features:

   - A **4 bit** ADDER/SUBTRACTOR that employs **2's complement scheme.**
   - *Relevant* **flag F/Fs** .
   - A REGISTER BANK *composed* of **8 (Eight)** *user accessible* **4 bit** *REGIST* $\$R_{i(i=1..8)}$
   - *Additional components as necessary.*

   Design in a *stepwise fashion* (**clearly depicting the different design steps**), a **Booth Multiplier** that uses the *above mentioned* DATAPATH together with a **hardwired sequencer** to *multiply* two **2's complement 4 bit** *Binary numbers A & B* and produce a **2's complement 8 Bit** *RESULT C*

   C = AxB employing *BOOTH's* algorithm.

   Assume that both the operands *A*, *B* as well as the result *C will be available* in some DATAPATH registers $\$R_{i(i=1..8)}$ after completion of the Multiplication process.

   [8]

Contd..5

Q.6.(a) (Contd.)

ii. Clearly specify the *snapshots* of your **Datapath Components when it is multiplying** $A = 7_{10}$ with $B = 5_{10}$. [6]

(b) For a hard disk drive the following informations are given :

- Sector size = **512 bytes**.
- SPEED = **7200 rpm**.
- Time to *move* the **Read Write head** from one cylinder to the *adjacent* one = **0.1 msec**.
- **Data transfer rate = 16M byte** *per second.*
- Disk Controller overhead = **2 msec.**
- PRESENT POSITION of **Read Write head** = at *cylinder number* **1500.**

Find the **maximum time** needed to *transfer* **4 Kbytes** of data *starting* from **cylinder number 2000**. The data is in **consecutive sectors. Clearly** show all calculation steps. [4]

7 (a) A computer has **4K byte pages**, 30 bit *physical address*, 32 bit virtual addresses, The system *main memory capacity* happens to be **64K bytes** .It also possesses a 12 entry *fully associative* **TLB Cache.**

i. Specify a Virtual Address schematic.

ii. Specify a Physical Address schematic.

iii. How many **bits** will be there in each **TLB** line assuming **6 bit PID** has to be included for memory protection?

[2 × 3 = 6]

(b) Consider a *paging system* in which the main memory has a capacity of **three (3)** pages. The execution of a *program* **Q** requires reference to **five** *distinct* pages **p1, p2, p3, p4, &p5**. The page address stream formed by *executing* **Q** happens to be the following:

p2, p3, p2, p1, p5, p2, p4, p5, p3, p2, p5, p2

Depict the manner in which the **pages** are **brought in main memory** using *each* of the following algorithms, clearly **marking out** *in each step* the **page** *selected* for **replacement**.

i. FIFO

ii. LRU

[2 × 6 = 12]

******