

B.E.

Seventh Semester Examination, 2009-2010

Advanced Computer Architecture (CSE-401-E)

Note : Attempt any *five* questions. All questions carry equal marks.

Q. 1. (a) What do you mean by micro programming? Give its advantages and disadvantages?

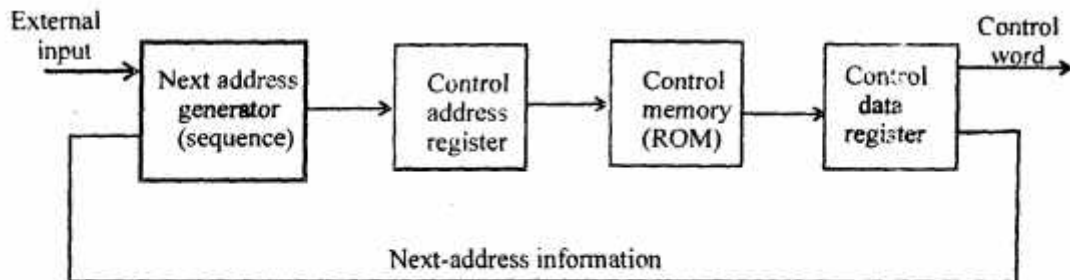
Ans. The control function that specifies a micro-operation is a binary variable. When it is in one binary state, the corresponding microoperation is executed. In a bus-organized system, the control signals that specify microoperations are groups of bits that select the paths in multiplexers, decoders, and arithmetic logic units.

A sequence of microinstructions constitutes a program called as microprogram and constitution is called microprogramming. Control words can be programmed to perform various operations on the components of the system. A control unit whose binary control variables are stored in memory is called a microprogrammed control unit.

A computer that employs a microprogramming control unit will have two separate memories;

- (i) A main memory
- (ii) A control memory.

Microprogramming consists of microinstructions that specifies various internal control signals for execution of register microoperations.



Microprogrammed Control Organization

The main advantage of the microprogrammed control unit is the fact that once the hardware configuration is established, there should be no need for further hardware or wiring changes.

This can lead to a disadvantage that hardware configuration should not be changed for different operations; the only thing that must be changed is the microprogram residing in control memory.

Q. 1. (b) What are basic data types? Explain.

Ans. The C language contains four basic data types; int, float, char and double. These four types are native to the machine's hardware. A double variable is a double precision floating point number. There are three qualifiers that can be applied for int : short, long and unsigned. A variable declaration in C specifies two things. First, it specifies the amount of storage that must be set aside for objects declared with data types.

A variable declaration specifies that storage be set aside for an object of the specified type and that the object at that storage can be referenced with the specified variable identifier.

Second, it specifies how data represented by strings of bits are to be interpreted. The same bits at a

specific storage location can be implemented as an integer of a floating point number, yielding two completely different numeric values.

A short or long integer variable refers to the maximum size of the variable's value. The actual maximum sizes implied by short int, long int or int vary from machine to machine.

Abstract Data Types : A useful tool for specifying the logical properties of a data type is the abstract data types or ADT. In defining abstract data types we are not concerned with time or space complexity. Those are implementation issues.

A data type is a collection of values and a set of operations on those values. That collection and operations form a mathematical construct that may be implemented using a particular hardware software data structure.

The term "abstract data types" refers to basic mathematical concept that defines data types.

By specifying mathematical and logical properties of data type or structure, ADT is a useful tool to programmers who wish to use data type correctly.

Q. 2. (a) Explain instruction set of a processor.

Ans. Instruction Set of a Processor : The set of an instruction is usually depicted in a rectangular box symbolizing the bits of the instruction as they appear in memory words or in a control register. The bits of the instruction are divided into groups called fields. The most common field found in instruction formats are :

- (i) "An operation code field that specifies the operation to be performed.
- (ii) An address field that designates a memory address or a processor register.
- (iii) A mode field that specifies the way the operand or the effective address is determined.

An instruction that specifies an arithmetic solution is defined by an assembly language instruction as
ADD X

ADD R₁, R₂, R₃ denotes the operation $R_1 \leftarrow R_2 + R_3$

MOV R₁, R₂ denotes the operation $R_1 \leftarrow R_2$.

The influence of number of addresses on computer programs, we will evaluate arithmetic statement.

$$X = (A + B) * (C + D)$$

Using zero, one, two or three address informations instructions. We will use the symbols ADD, SUB, MUL and DIV for the four arithmetic operations; MOV for the transfer type operation; and LOAD and STORE for transfer to and from memory and AC register. We will assume that the operands are in memory address A, B, C and D and the result must be stored in memory location X.

Three Address Instructions : $X = (A + B) * (C + D)$

ADD	R ₁ , A, B	$R_1 \leftarrow M[A] + M[B]$
ADD	R ₂ , C, D	$R_2 \leftarrow M[C] + M[D]$
MUL	X, R ₁ , R ₂	$M[X] \leftarrow R_1 * R_2$

Q. 2. (b) Explain processor evaluation matrix.

Ans. Processor Evaluation Matrix : Optimal benchmarking of processors does not require any detailed knowledge of processor or system architecture.

Knowledge of benefits of processor based systems will help enabling benchmarks to show the different ways of how this processor performs relative to its competition.

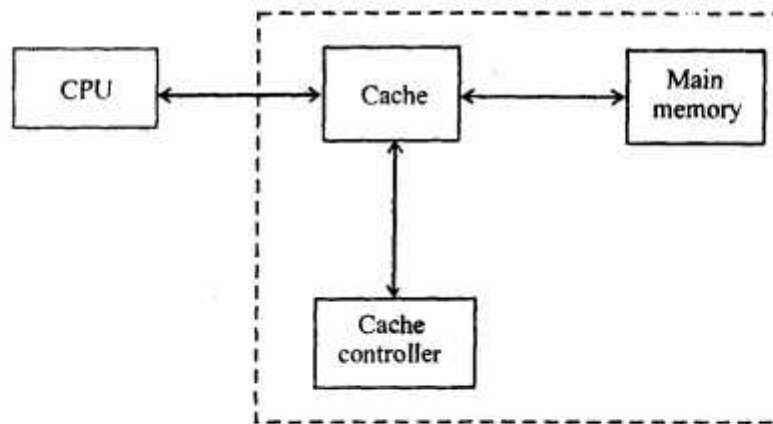
AMD designed a 64-bit PC processor that offers industry leading performance and native **compatibility** with current 32-bit applications.

Architectural improvements specifically designed to increase instructions per clock (IPC) includes :

- (i) AMD 64
- (ii) Integrated DDR memory controller
- (iii) An advanced hyper transport link
- (iv) Very very level one (L_1) and level very (L_2) on die-cache
- (v) Processor core clock for clock instruction set
- (vi) Instruction of SSE2 instruction set
- (vii) 64 bit processing.

Q. 3. (a) What do you mean by two level cache? Explain.

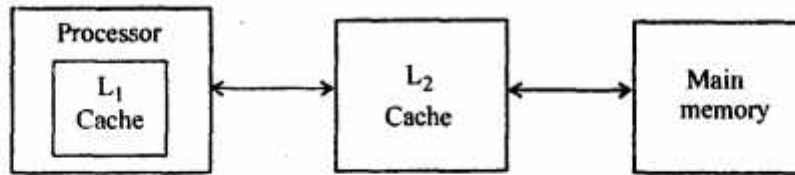
Ans. Two Level Cache : A cache memory system includes a small amount of fast memory (SRAM) and a large amount of slow memory (DRAM). This system is configured to simulate a large amount of fast memory.



Cache Memory System

It is also possible to place an even smaller cache between the cache and the processor and the system created is called as two level cache.

The need of using two level cache is to reduce the time of access to any record. There is some time occupied in travelling from processor to cache that is due to integrated circuits embedded in central processing unit. To make a solution for this time acquiring and slow processing is to embed a very small sized cache into the processor which will be known as one level cache and the regular cache placed between the CPU and main memory known to be at second level cache.

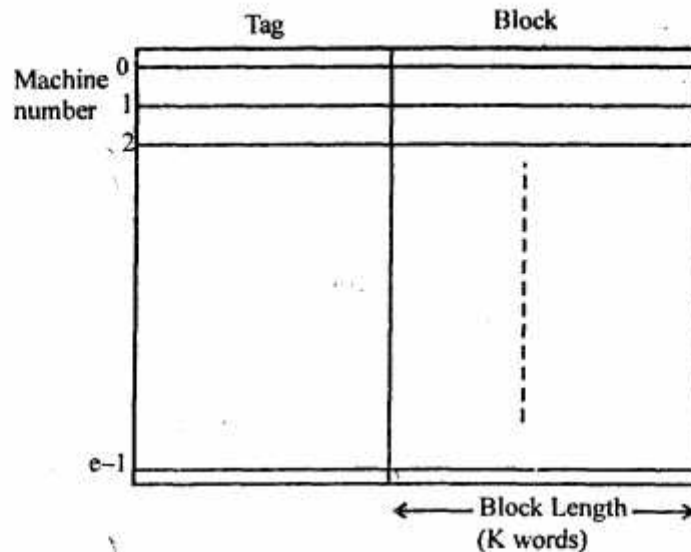


L₁ and L₂ cache placement

Q. 3. (b) Explain cache organization in detail.

Ans. Memory words in cache are combined to form a small group, known as cache blocks or lines or frames. An address is assigned to each block, referred as tag. The collection of tag addresses currently assigned to the cache, which can be non-contiguous, is stored as directory.

Figure shows the organization of a cache/main memory system. Consider a main memory consists of 2^n addressable locations, each having a unique n-bit address.



(a) Cache

For mapping purposes, let this memory consist of K number of fixed length blocks. So, there are $M = 2^n / K$ blocks. Cache consists of C lines and K words each and the number of lines is always less than the number of main memory blocks ($C < M$). At any time some subset of the memory blocks resides in the cache lines. Each line includes a tag that identifies which particular memory block is currently being stored. Usually the tag is the portion of memory address. To improve the performance, the time required to check tag addresses and access the data from cache memory must be less than the time required to access cache memory.

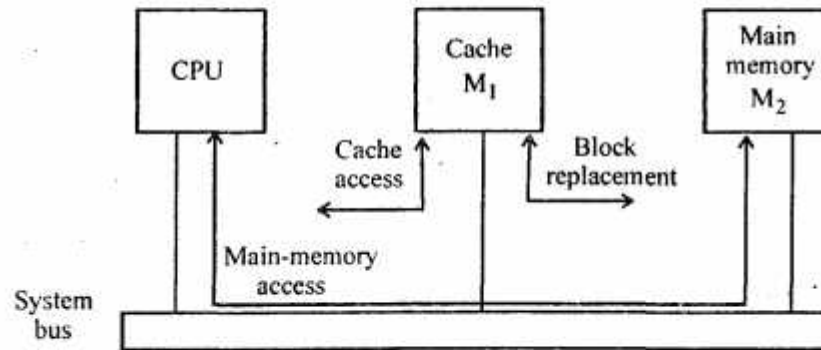
A cache is arranged within a computer in two general ways :

- (i) Look-aside and
- (ii) Look-through.

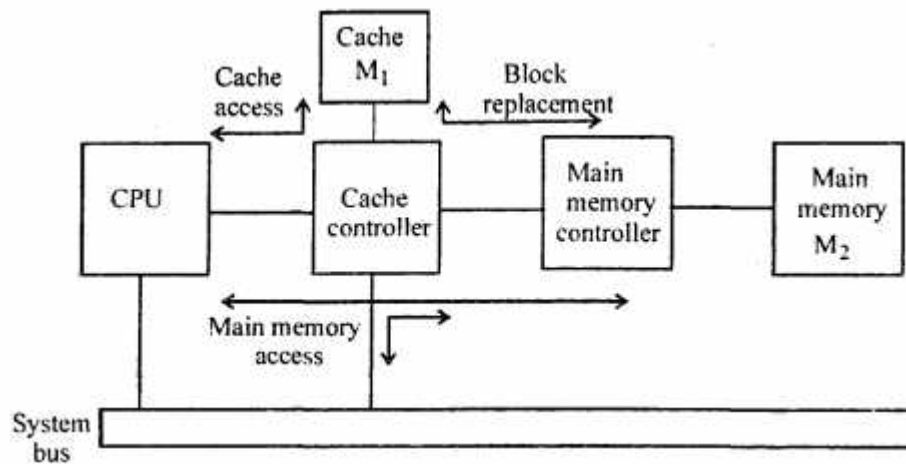
In the look aside design, the cache and the main memory are directly connected to the system bus. In this

design, the processor communicates with the cache via the system bus.

In the look through design the processor communicates via a separate local bus that is isolated from the main system bus. Thus, the system bus is available for use by other units to communicate with the memory. A look through cache allows the local bus to be wider than the system bus that speeds up the cache main memory transfers. The disadvantage of this design is higher complexity, higher cost and takes longer for M_2 to respond to the processor when a cache miss occurs.



(a) Look aside



(b) Look through

Q. 4. (a) Explain the memory modelling using queuing theory.

Ans. Memory Modelling : Pricing queuing theory is used to effectively allocate system resource to process the client by gateway servers. It is an improved scheme for autonomous performance of gateway servers under highly dynamic traffic loads.

This methodology calculates queue length and waiting time utilizing gateway server information to reduce response time variance in presence of bursty traffic.

The layout of gateway server is shown below in which data arriving and data departing at a mode has been recorded.

All jobs of a single class have the same service demands and transition probabilities. There is each class consists of jobs which are indistinguishable.

Q. 5. (a) Explain vector processor.

Ans. Vector Processing : A vector operand contains an ordered set of n elements where n is called the length of the vector. Each element in a vector is a scalar quantity, which may be a floating point number, an integer, a logical value, or a character. Vector instructions can be classified into four primitive types :

$$f_1 ; V \rightarrow V$$

$$f_2 ; V \rightarrow S$$

$$f_3 ; V \times V \rightarrow V$$

$$f_4 ; V \times S \rightarrow V$$

Where V and S denotes a vector operand and a scalar operand. The mappings f_1 and f_2 are unary operations and f_3 and f_4 are binary operations.

Most vector processors have pipeline structures. Vector instructions need to perform the same operation on different data sets.

Some special instructions may be used to facilitate the manipulation of vector data.

A boolean vector can be generated as a result of comparing two vectors and can be used as a masking vector for enabling or disabling components operations in a vector instruction. A compress instruction will shorten a vector under the control of a masking vector. A merge instruction combines two vectors.

Q. 5. (b) Compare vector and issue processors.

Ans. Most vector processors have pipeline structure. Vector instruction needs to perform the same operations on data sets. This is not two for scalar processing over a single pair of operands. One obvious advantage of vector processing over scalar processing is the elimination of the overhead processing is the elimination of the overhead called and caused by loop control mechanism. Because of startup delay in a pipeline, a vector processor should perform better with longer vectors.

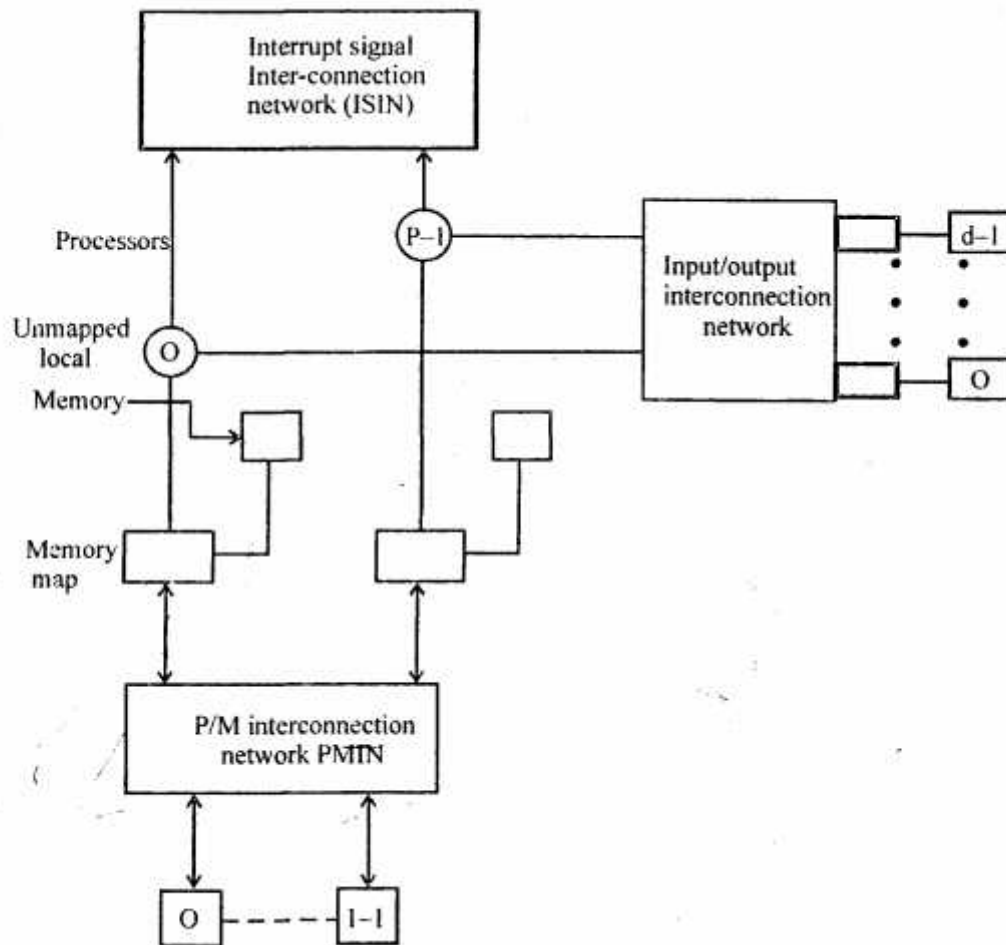
Q. 6. (a) What are basic issues in multi processors?

Ans. A multiprocessor system is an inter-connection of two or more CPU's with memory and input output equipment. A multiprocessor system implies the existence of multiple CPUs although there will be one or more IOPs as well.

A multiprocessor system is controlled by one operating system that provides interaction between processors and all the components of the system cooperate in the solution of a problem.

Multiprocessing can improve performance by decomposing a program into parallel executable tasks.

Most multiprocessors manufactures provide an operating system with programming language constructs suitable for specifying parallel processing. To make it more efficient way is to provide a compiler with multiprocessor software that can automatically detect parallelism in a user's program. The compiler checks for data dependency in the program. If a program depends on data generated in another part, the part yielding the needed data must be executed first. Two parts of a program that do not use data generated by each can run currently. The parallelizing compiler checks the entire program to detect any possible data dependencies. These that have no data dependency are then considered for concurrent scheduling on different processors.



Without Private Cache

Q. 7. (a) How to overlap the T cycle in V.R. translation?

Ans. T-Cycles : It is a machine cycle. A machine cycle is defined as the time required to complete the one operation. Its necessary steps are to fetch, a read or a write operation constitutes a machine cycle.

It is combined with an instruction cycle. The necessary steps that a CPU carries out to fetch an instruction and necessary state data from the memory, and to execute it constitute an instruction cycle.

T-State : One sub-division of an operation performed in one clock is called a T-State.

VR cycle or variable resistor cycle can be translated from transistor cycle.

Q. 7. (b) Explain virtual to real mapping.

Ans. Real memory refers to the actual memory chips that are installed in the computer. All programs actually runs in physical memory. It is often useful to allow the computer to think that it has memory that exist actually there, in order to permit the use of programs which are larger than will physically fit in the memory or to allow multitasking. This concept is called virtual memory.

The way virtual memory works is relatively simple, Let's suppose the operating system needs 80 MB of

memory to hold all programs that are running, but there are only 32 MB of RAM chips installed in the system. The operating system sets up 80 MB of virtual memory and employs a virtual memory manager, a program design to control virtual memory.

The virtual memory manager sets up a file on the hard disk that is 48 MB in size (80–32). The operating system then proceeds to use 80 MB worth of memory addresses. To the operating system it appears as if 80 MB of memory exists. It lets the virtual memory manager worry how to handle the fact that we only have 32 MB of real memory.

Virtual memory can also hamper performance. The larger the virtual memory is compared to real memory, the more swapping is done to the hard disk. The hard disk is, much slower than system memory. Trying to use too many programs using too little memory will result into thrashing.

Q. 8. Write short notes on :

- (i) **Instruction timing**
- (ii) **Cache considerations**
- (iii) **Processor with cache**

Ans. (i) Instruction Timing : Generally instructions take 1 cycle per word of memory accessed. Thus, start with 1 cycle for the instruction itself. Then add 1 cycle for a memory source, 2 cycles for a memory destination and one additional cycle per offset word.

Note : In two operand instructions, memory destinations requires an offset word, so they cost a total of three cycles. This holds even for instructions (MOV, CMP and BIT) that only access the destination once.

Short immediate constants (using r_2 and r_3) count as register operands for instruction timing purpose.

(ii) Cache Considerations : There are several considerations to complete the design of a set associative cache located between the processor and memory :

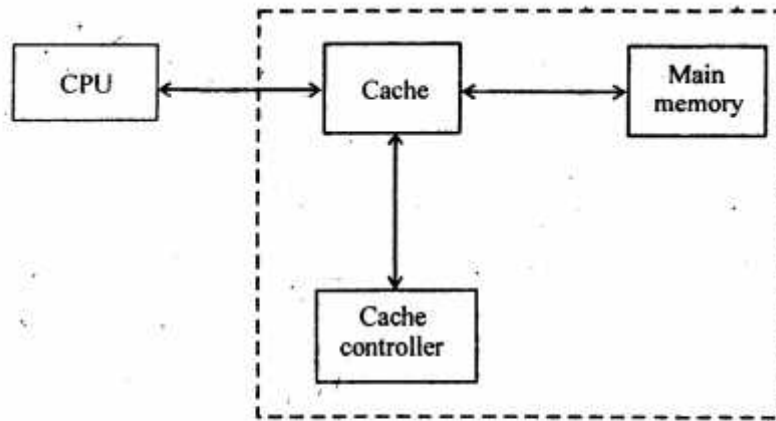
- (i) Cache block size
- (ii) When to clean the cache block
- (iii) When to load a cache block
- (iv) When to dump a cache block
- (v) Cache position.

Size of cache including the cache cost, the cache amount of information retrieved in a single memory access and the amount of information accessed to republish the instruction processing pipeline. A cache block is clean when its components matches the corresponding block of main memory. Cache loading policy decides that if there is but a single program in the system, it always pays to load a block is first read, for locality predicts that the program is likely to read other information from that same block.

To check cache position alternatives for decision making are :

- (i) Request competition
 - (ii) Multiple copies of data
 - (iii) Speed.
- (iii) Processor with Cache :**

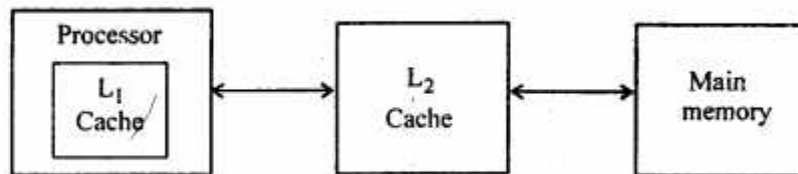
Two Level Cache : A cache memory system includes a small amount of fast memory (SRAM) and a large amount of slow memory (DRAM). This system is configured to simulate a large amount of fast memory.



Cache Memory System

It is also possible to place an even smaller cache between the cache and the processor and the system created is called as two level cache.

The need of using two level cache is to reduce the time of access to any record. There is some time occupied in travelling from processes to cache that is due to integrated circuits embedded in central processing unit. To make a solution for this time acquiring and slow processing is to embed a very small sized cache into the processor which will be known as one level cache and the regular cache placed between the CPU and main memory known to be at second level cache.



L₁ and L₂ cache placement