Computer Architecture

Two Marks Questions with Answers

CS6303 COMPUTER ARCHITECTURE

II YEAR/ 3RD SEMESTER CSE

UNIT-I	OVERVIEW & INSTRUCTIONS

1. What are the eight great ideas in computer architecture?

The eight great ideas in computer architecture are:

1. Design for Moore's Law
2. Use Abstraction to Simplify Design
3. Make the Common Case Fast
4. Performance via Parallelism
5. Performance via Pipelining
6. Performance via Prediction
7. Hierarchy of Memories
8. Dependability via Redundancy

2. What are the five classic components of a computer?

The five classic components of a computer are input, output, memory, datapath, and control, with the last two sometimes combined and called the processor.

3. Define – ISA

The instruction set architecture, or simply architecture of a computer is the interface between the hardware and the lowest-level software. It includes anything programmers need to know to make a binary machine language program work correctly, including instructions, I/O devices, and so on.

4. Define – ABI

Typically, the operating system will encapsulate the details of doing I/O, allocating memory, and other low-level system functions so that application programmers do not need to worry about such details. The combination of the basic instruction set and the operating system interface provided for application programmers is called the application binary interface (ABI).

5. What are the advantages of network computers?

Networked computers have several major advantages:

- Communication: Information is exchanged between computers at high speeds.
- Resource sharing: Rather than each computer having its own I/O devices, computers on the network can share I/O devices.

- Nonlocal access: By connecting computers over long distances, users need not be near the computer they are using.

6. Define – Response Time

Response time is also called execution time. The total time required for the computer to complete a task, including disk accesses, memory accesses, I/O activities, operating system overhead, CPU execution time, and so on is called response time.

7. Define – Throughput

Throughput or bandwidth is the total amount of work done in a given time.

8. Write the CPU performance equation.

The Classic CPU Performance Equation in terms of instruction count (the number of instructions executed by the program), CPI, and clock cycle time:

$$\text{CPU time} = \text{Instruction count} \times \text{CPI} \times \text{Clock cycle time}$$

or, since the clock rate is the inverse of clock cycle time:

$$\text{CPU time} = \frac{\text{Instruction count} \times \text{CPI}}{\text{Clock rate}}$$

9. If computer A runs a program in 10 seconds, and computer B runs the same program in 15 seconds, how much faster is A over B.

We know that A is n times as fast as B if

$$\frac{\text{Performance}_A}{\text{Performance}_B} = \frac{\text{Execution time}_B}{\text{Execution time}_A} = n$$

Thus the performance ratio is

$$\frac{15}{10} = 1.5$$

and A is therefore 1.5 times as fast as B.

10. What are the basic components of performance?

The basic components of performance and how each is measured are:

Components of Performance	Units of measure
CPU execution time for a program	Seconds for the program
Instruction count	Instruction executed for the program
Clock cycles per instruction(CPI)	Average number of clock cycles per instruction
Clock cycle time	Seconds per clock cycle

11. Write the formula for CPU execution time for a program.

$$\text{CPU execution time for a program} = \text{CPU clock cycles for a program} \times \text{Clock cycle time}$$

Alternatively, because clock rate and clock cycle time are inverses,

$$\text{CPU execution time for a program} = \frac{\text{CPU clock cycles for a program}}{\text{Clock rate}}$$

12. Write the formula for CPU clock cycles required for a program.

$$\text{CPU clock cycles} = \text{Instructions for a program} \times \text{Average clock cycles per instruction}$$

13. Define – MIPS

Million Instructions Per Second (MIPS) is a measurement of program execution speed based on the number of millions of instructions.

MIPS is computed as:

$$\text{MIPS} = \frac{\text{Instruction count}}{\text{Execution time} \times 10^6}$$

14. What are the fields in an MIPS instruction?

MIPS fields are

op	rs	rt	rd	shamt	funct
6 bits	5bits	5 bits	5 bits	5 bits	6 bits

Where,

op: Basic operation of the instruction, traditionally called the opcode.
rs: The first register source operand.
rt: The second register source operand.
rd: The register destination operand. It gets the result of the operation. shamt: Shift amount.
funct: Function.

15. Write an example for immediate operand.

The quick add instruction with one constant operand is called add immediate or addi. To add 4 to register $s3, we just write

```
addi    $s3,$s3,4       # $s3 = $s3 + 4
```

16. Define – Stored Program Concepts

Today's computers are built on two key principles:

1. Instructions are represented as numbers.
2. Programs are stored in memory to be read or written, just like data.

These principles lead to the stored-program concept. Treating instructions in the same way as data greatly simplifies both the memory hardware and the software of computer systems.

17. Define – Addressing Modes

The different ways in which the operands of an instruction are specified are called as addressing modes.

The MIPS addressing modes are the following:

1. Immediate addressing
2. Register addressing
3. Base or displacement addressing
4. PC-relative addressing
5. Pseudo direct addressing

UNIT-II	ARITHMETIC OPERATIONS

1. Add 6_{10} to 7_{10} in binary and Subtract 6_{10} from 7_{10} in binary.

Addition,

$$\begin{array}{ll} & 0000\ 0000\ 0000\ 0000\ 0000\ 0000\ 0000\ 0111_{two} = 7_{ten} \\ + & 0000\ 0000\ 0000\ 0000\ 0000\ 0000\ 0000\ 0110_{two} = 6_{ten} \\ \hline = & 0000\ 0000\ 0000\ 0000\ 0000\ 0000\ 0000\ 1101_{two} = 13_{ten} \end{array}$$

Subtraction directly,

$$\begin{array}{ll} & 0000\ 0000\ 0000\ 0000\ 0000\ 0000\ 0000\ 0111_{two} = 7_{ten} \\ - & 0000\ 0000\ 0000\ 0000\ 0000\ 0000\ 0000\ 0110_{two} = 6_{ten} \\ \hline = & 0000\ 0000\ 0000\ 0000\ 0000\ 0000\ 0000\ 0001_{two} = 1_{ten} \end{array}$$

Or via two's complement of -6,

$$\begin{array}{ll} & 0000\ 0000\ 0000\ 0000\ 0000\ 0000\ 0000\ 0111_{two} = 7_{ten} \\ + & 1111\ 1111\ 1111\ 1111\ 1111\ 1111\ 1111\ 1010_{two} = -6_{ten} \\ \hline = & 0000\ 0000\ 0000\ 0000\ 0000\ 0000\ 0000\ 0001_{two} = 1_{ten} \end{array}$$

2. Write the overflow conditions for addition and subtraction.

Operation	Operand A	Operand B	Result Indicating overflow
A+B	≥0	≥0	<0
A+B	<0	<0	≥0
A-B	≥0	<0	<0
A-B	<0	≥0	≥0

3. Define – Moore's Law

Moore's Law has provided so much more in resources that hardware designers can now build much faster multiplication and division hardware. Whether the multiplicand is to be added or not is known at the beginning of the multiplication by looking at each of the 32 multiplier bits.

4. **What are the floating point instructions in MIPS?**

MIPS supports the IEEE 754 single precision and double precision formats with these instructions:

- Floating-point addition
- Floating-point subtraction
- Floating-point multiplication
- Floating-point division
- Floating-point comparison
- Floating-point branch

5. **Define – Guard and Round**

Guard is the first of two extra bits kept on the right during intermediate calculations of floating point numbers. It is used to improve rounding accuracy.

Round is a method to make the intermediate floating-point result fit the floating-point format; the goal is typically to find the nearest number that can be represented in the format. IEEE 754, therefore, always keeps two extra bits on the right during intermediate additions, called guard and round, respectively.

6. **Define – ULP**

Units in the Last Place is defined as the number of bits in error in the least significant bits of the significant between the actual number and the number that can be represented.

7. **What is meant by sticky bit?**

Sticky bit is a bit used in rounding in addition to guard and round that is set whenever there are nonzero bits to the right of the round bit. This sticky bit allows the computer to see the difference between 0.50 … 00 ten and 01 ten when rounding.

8. **Write the IEEE 754 floating point format.**

The IEEE 754 standard floating point representation is almost always an approximation of the real number.

$$(-1)^S \times (1 + \text{Fraction}) \times 2^{(\text{Exponent} - \text{Bias})}$$

9. **What is meant by sub-word parallelism?**

Given that the parallelism occurs within a wide word, the extensions are classified as sub-word parallelism. It is also classified under the more general name of data level parallelism. They have been also called vector or SIMD, for single instruction, multiple data . The rising popularity of multimedia applications led to arithmetic instructions that support narrower operations that can easily operate in parallel.

For example, ARM added more than 100 instructions in the NEON multimedia instruction extension to support sub-word parallelism, which can be used either with ARMv7 or ARMv8.

10. Multiply $1000_{10} * 1001_{10}$.

Multiplicand		1000_{ten}
Multiplier	×	1001_{ten}
		1000
		0000
		0000
		1000
Product		1001000_{ten}

11. Divide $1,001,010_{ten}$ by 1000_{ten}.

	1001_{ten}	Quotient
Divisor 1000_{ten}	1001010_{ten}	Dividend
	−1000	
	10	
	101	
	1010	
	−1000	
	10_{ten}	Remainder

12. What are the steps in the floating-point addition?

The steps in the floating-point addition are

1. Align the decimal point of the number that has the smaller exponent.
2. Addition of the significands
3. Normalize the sum.
4. Round the result.

UNIT-III	PROCESSOR AND CONTROL UNIT

1. **What is meant by data path element?**

   A data path element is a unit used to operate on or hold data within a processor. In the MIPS implementation, the data path elements include the instruction and data memories, the register file, the ALU, and adders.

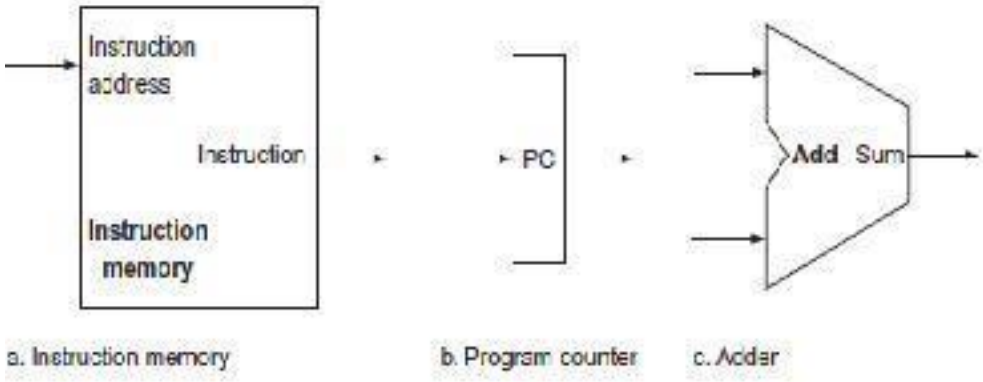
2. **What is the use of PC register?**

   Program Counter (PC) is the register containing the address of the instruction in the program being executed.

3. **What is meant by register file?**

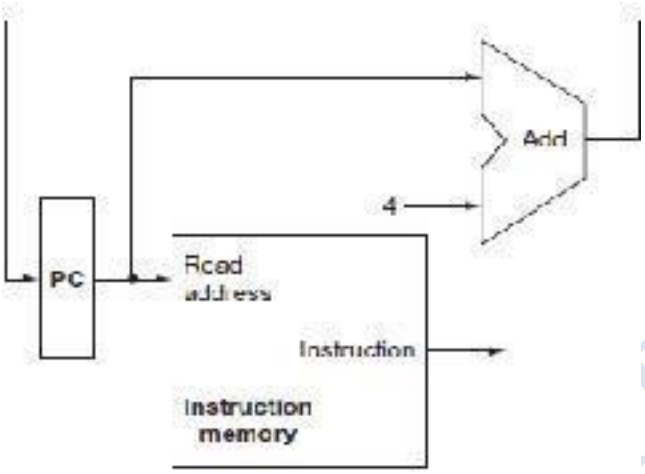
   The processor's 32 general-purpose registers are stored in a structure called a register file. A register file is a collection of registers in which any register can be read or written by specifying the number of the register in the file. The register file contains the register state of the computer.

4. **What are the two state elements needed to store and access an instruction?**



   Two state elements are needed to store and access instructions, and an adder is needed to compute the next instruction address. The state elements are the instruction memory and the program counter.

5. Draw the diagram of portion of datapath used for fetching instruction.



A portion of the data path is used for fetching instructions and incrementing the program counter. The fetched instruction is used by other parts of the data path.

6. Define – Sign Extend

Sign-extend is used to increase the size of a data item by replicating the high-order sign bit of the original data item in the high order bits of the larger, destination data item.

7. What is meant by branch target address?

Branch target address is the address specified in a branch, which becomes the new program counter (PC) if the branch is taken. In the MIPS architecture the branch target is given by the sum of the off set field of the instruction and the address of the instruction following the branch.

8. Differentiate branch taken from branch not taken.

Branch taken is a branch where the branch condition is satisfied and the program counter (PC) becomes the branch target. All unconditional jumps are taken branches.

Branch not taken or (untaken branch) is a branch where the branch condition is false and the program counter (PC) becomes the address of the instruction that sequentially follows the branch.

9. What is meant by delayed branch?

Delayed branch is a type of branch where the instruction immediately following the branch is always executed, independent of whether the branch condition is true or false.

10. What are the three instruction classes and their instruction formats?

Field	0	rs	rt	rd	shamt	funct
Bit positions	31:26	25:21	20:16	15:11	10:6	5:0

a. R-type instruction

Field	35 or 43	rs	rt	address
Bit positions	31:26	25:21	20:16	15:0

b. Load or store instruction

Field	4	rs	rt	address
Bit positions	31:26	25:21	20:16	15:0

c. Branch instruction

The three instruction classes (R-type, load and store, and branch) use two different instruction formats.

11. Write the instruction format for the jump instruction.

The destination address for a jump instruction is formed by concatenating the upper 4 bits of the current PC + 4 to the 26-bit address field in the jump instruction and adding 00 as the 2 low-order bits.

Field	000010	address
Bit positions	31:26	25:0

12. What is meant by pipelining?

Pipelining is an implementation technique in which multiple instructions are overlapped in execution. Pipelining improves performance by increasing instruction throughput, as opposed to decreasing the execution time of an individual instruction.

13. What are the five steps in MIPS instruction execution?

1. Fetch instruction from memory.
2. Read registers while decoding the instruction. The regular format of MIPS instructions allows reading and decoding to occur simultaneously.
3. Execute the operation or calculate an address.
4. Access an operand in data memory.
5. Write the result into a register.

14. Write the formula for calculating time between instructions in a pipelined processor.

$$\text{Time between instructions}_{\text{pipelined}} = \frac{\text{Time between instruction}_{\text{nonpipelined}}}{\text{Number of pipe stages}}$$

15. What are hazards? Write its types.

There are situations in pipelining when the next instruction cannot be executed in the following clock cycle. These events are called hazards, and there are three different types.

1. Structural Hazards
2. Data Hazards
3. Control Hazards

16. What is meant by forwarding?

Forwarding, also called bypassing, is a method of resolving a data hazard by retrieving the missing data element from internal buffers rather than waiting for it to arrive from programmer visible registers or memory.

17. What is pipeline stall?

Pipeline stall, also called bubble, is a stall initiated in order to resolve a hazard. They can be seen elsewhere in the pipeline.

18. What is meant by branch prediction?

Branch prediction is a method of resolving a branch hazard that assumes a given outcome for the branch and proceeds from that assumption rather than waiting to ascertain the actual outcome.

19. What are the 5 pipeline stages?

The 5 stages of instruction execution in a pipelined processor are:

1. IF: Instruction fetch
2. ID: Instruction decode and register file read
3. EX: Execution or address calculation
4. MEM: Data memory access
5. WB: Write back

20. What are exceptions and interrupts?

Exception, also called interrupt, is an unscheduled event that disrupts program execution used to detect overflow. Eg. Arithmetic overflow, using an undefined instruction.

Interrupt is an exception that comes from outside of the processor. Eg. I/O device request

21. Define – Vectored Interrupts

Vectored interrupt is an interrupt in that the address to which the control is transferred is determined by the cause of the exception.

UNIT-IV	PARALLELISM

1. What is meant by ILP?

Pipelining exploits the potential parallelism among instructions. This parallelism is called instruction-level parallelism (ILP). There are two primary methods for increasing the potential amount of instruction-level parallelism.

1. Increasing the depth of the pipeline to overlap more instructions.
2. Multiple issue.

2. What is multiple issue? Write any two approaches.

Multiple issue is a scheme whereby multiple instructions are launched in one clock cycle. It is a method for increasing the potential amount of instruction-level parallelism. It is done by replicating the internal components of the computer so that it can launch multiple instructions in every pipeline stage. The two approaches are:

1. Static multiple issue (at compile time)
2. Dynamic multiple issue (at run time)

3. What is meant by speculation?

One of the most important methods for finding and exploiting more ILP is speculation. It is an approach whereby the compiler or processor guesses the outcome of an instruction to remove it as dependence in executing other instructions.

For example, we might speculate on the outcome of a branch, so that instructions after the branch could be executed earlier.

4. Define – Static Multiple Issue

Static multiple issue is an approach to implement a multiple-issue processor where many decisions are made by the compiler before execution.

5. Define – Issue Slots and Issue Packet

Issue slots are the positions from which instructions could be issued in a given clock cycle. By analogy, these correspond to positions at the starting blocks for a sprint.

Issue packet is the set of instructions that issues together in one clock cycle; the packet may be determined statically by the compiler or dynamically by the processor.

6. Define – VLIW

Very Long Instruction Word (VLIW) is a style of instruction set architecture that launches many operations that are defined to be independent in a single wide instruction, typically with many separate opcode fields.

7. Define – Superscalar Processor

Superscalar is an advanced pipelining technique that enables the processor to execute more than one instruction per clock cycle by selecting them during execution. Dynamic multiple-issue processors are also known as superscalar processors, or simply superscalars.

8. What is meant by loop unrolling?

An important compiler technique to get more performance from loops is loop unrolling, where multiple copies of the loop body are made. After unrolling, there is more ILP available by overlapping instructions from different iterations.

9. What is meant by anti-dependence? How is it removed?

Anti-dependence is an ordering forced by the reuse of a name, typically a register, rather than by a true dependence that carries a value between two instructions. It is also called as name dependence.

Register renaming is the technique used to remove anti-dependence in which the registers are renamed by the compiler or hardware.

10. What is the use of reservation station and reorder buffer?

Reservation station is a buffer within a functional unit that holds the operands and the operation.

Reorder buffer is the buffer that holds results in a dynamically scheduled processor until it is safe to store the results to memory or a register.

11. Differentiate in-order execution from out-of-order execution.

Out-of-order execution is a situation in pipelined execution when an instruction is blocked from executing does not cause the following instructions to wait. It preserves the data flow order of the program.

In-order execution requires the instruction fetch and decode unit to issue instructions in order, which allows dependences to be tracked, and requires the commit unit to write results to registers and memory in program fetch order. This conservative mode is called in-order commit.

12. What is meant by hardware multithreading?

Hardware multithreading allows multiple threads to share the functional units of a single processor in an overlapping fashion to try to utilize the hardware resources efficiently. To permit this sharing, the processor must duplicate the independent state of each thread. It Increases the utilization of a processor.

13. What are the two main approaches to hardware multithreading?

There are two main approaches to hardware multithreading. Fine-grained multithreading switches between threads on each instruction, resulting in interleaved execution of multiple threads. This interleaving is often done in a round-robin fashion, skipping any threads that are stalled at that clock cycle.

Coarse-grained multithreading is an alternative to fine-grained multithreading. It switches threads only on costly stalls, such as last-level cache misses.

14. What is SMT?

Simultaneous Multithreading (SMT) is a variation on hardware multithreading that uses the resources of a multiple-issue, dynamically scheduled pipelined processor to exploit thread-level parallelism. It also exploits instruction level parallelism.

15. Differentiate SMT from hardware multithreading.

Since SMT relies on the existing dynamic mechanisms, it does not switch resources every cycle. Instead, SMT is always executing instructions from multiple threads, leaving it up to the hardware to associate instruction slots and renamed registers with their proper threads.

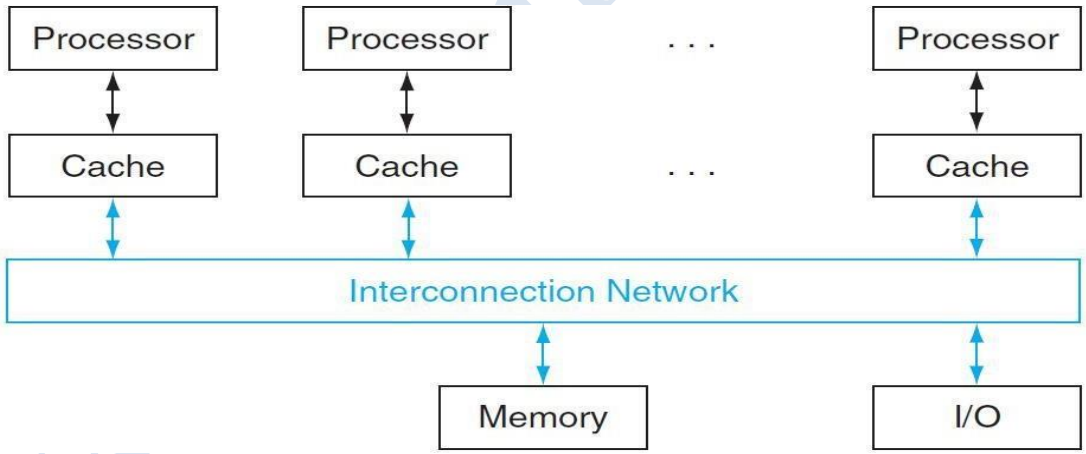
16. What are the three multithreading options?

The three multithreading options are:

1. A superscalar with coarse-grained multithreading
2. A superscalar with fine-grained multithreading
3. A superscalar with simultaneous multithreading

17. Define – SMP

Shared memory multiprocessor (SMP) is one that offers the programmer a single physical address space across all processors - which is nearly always the case for multicore chips. Processors communicate through shared variables in memory, with all processors capable of accessing any memory location via loads and stores.



18. Differentiate UMA from NUMA.

Uniform memory access (UMA) is a multiprocessor in which latency to any word in main memory is about the same no matter which processor requests the access.

Non uniform memory access (NUMA) is a type of single address space multiprocessor in which some memory accesses are much faster than others depending on which processor asks for which word.

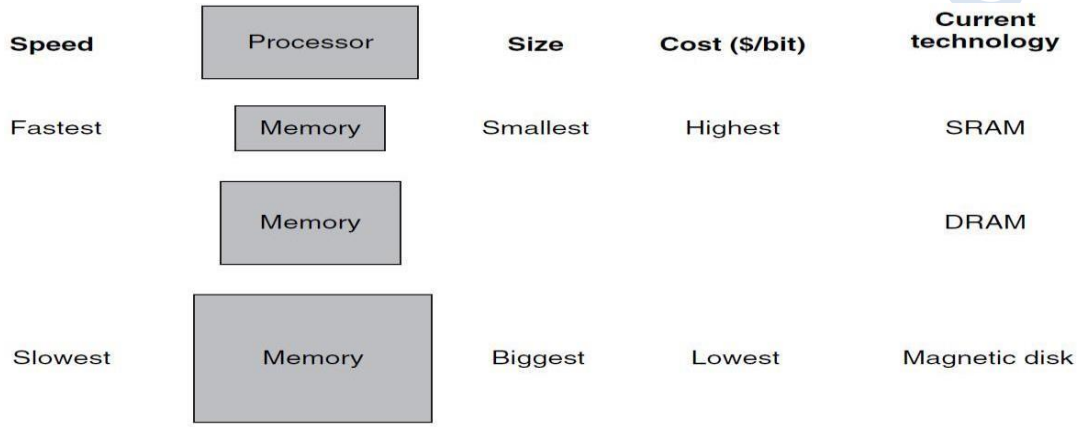
UNIT-V	MEMORY AND I/O SYSTEMS

1. What are the temporal and spatial localities of references?

Temporal locality (locality in time): if an item is referenced, it will tend to be referenced again soon.

Spatial locality (locality in space): if an item is referenced, items whose addresses are close by will tend to be referenced soon.

2. Write the structure of memory hierarchy.



The basic structure of a memory hierarchy.

3. What are the various memory technologies?

The various memory technologies are:

1. SRAM semiconductor memory
2. DRAM semiconductor memory
3. Flash semiconductor memory
4. Magnetic disk

4. Differentiate SRAM from DRAM.

SRAMs are simply integrated circuits that are memory arrays with a single access port that can provide either a read or a write. SRAMs have a fixed access time to any datum. SRAMs don't need to refresh and so the access time is very close to the cycle time. SRAMs typically use six to eight transistors per bit to prevent the information from being disturbed when read. SRAM needs only minimal power to retain the charge in standby mode.

In a dynamic RAM (DRAM), the value kept in a cell is stored as a charge in a capacitor. A single transistor is then used to access this stored charge, either to read the value or to overwrite the charge stored there. Because DRAMs use only a single transistor per bit of storage, they are much denser and cheaper per bit than SRAM. As

DRAMs store the charge on a capacitor, it cannot be kept indefinitely and must periodically be refreshed.

5. What is flash memory?

Flash memory is a type of electrically erasable programmable read-only memory (EEPROM). Unlike disks and DRAM, EEPROM technologies can wear out flash memory bits. To cope with such limits, most flash products include a controller to spread the writes by remapping blocks that have been written many times to less trodden blocks. This technique is called wear levelling.

6. Define − Rotational Latency

Rotational latency, also called rotational delay, is the time required for the desired sector of a disk to rotate under the read/write head, usually assumed to be half the rotation time.

7. What is direct-mapped cache?

Direct-mapped cache is a cache structure in which each memory location is mapped to exactly one location in the cache. For example, almost all direct-mapped caches use this mapping to find a block,

(Block address) modulo (Number of blocks in the cache)

8. Consider a cache with 64 blocks and a block size of 16 bytes. To what block number does byte address 1200 map?

The block is given by,

(Block address) modulo (Number of blocks in the cache)

where the address of the block is

$$\frac{\text{Byte address}}{\text{Bytes per block}}$$

Notice that this block address is the block containing all addresses between

$$\left\lfloor \frac{\text{Byte address}}{\text{Bytes per block}} \right\rfloor \times \text{Bytes per block}$$

and

$$\left\lfloor \frac{\text{Byte address}}{\text{Bytes per block}} \right\rfloor \times \text{Bytes per block} + (\text{Bytes per block} - 1)$$

Thus, with 16 bytes per block, byte address 1200 is block address

$$\left\lfloor \frac{1200}{6} \right\rfloor = 75$$

which maps to cache block number (75 modulo 64) = 11. In fact, this block maps all addresses between 1200 and 1215.

9. How many total bits are required for a direct-mapped cache with 16 KiB of data and 4-word blocks, assuming a 32-bit address?

We know that 16 KiB is 4096 (2^{12}) words. With a block size of 4 words (2^2), there are 1024 (2^{10}) blocks. Each block has 4×32 or 128 bits of data plus a tag, which is $32 - 10 - 2 - 2$ bits, plus a valid bit. Thus, the total cache size is

$$2^{10} \times (4 \times 32 + (32 - 10 - 2 - 2) + 1) = 2^{10} \times 147 = 147 \text{ Kibibits}$$

or 18.4 KiB for a 16 KiB cache. For this cache, the total number of bits in the cache is about 1.15 times as many as needed just for the storage of the data.

10. What are the writing strategies in cache memory?

Write-through is a scheme in which writes always update both the cache and the next lower level of the memory hierarchy, ensuring that data is always consistent between the two.

Write-back is a scheme that handles writes by updating values only to the block in the cache, then writing the modified block to the lower level of the hierarchy when the block is replaced.

11. What are the steps to be taken in an instruction cache miss?

The steps to be taken on an instruction cache miss are

1. Send the original PC value (current PC – 4) to the memory.
2. Instruct main memory to perform a read and wait for the memory to complete its access.
3. Write the cache entry, putting the data from memory in the data portion of

the entry, writing the upper bits of the address (from the ALU) into the tag field, and turning the valid bit on.

4. Restart the instruction execution at the first step, which will refetch the instruction, this time finding it in the cache.

12. Define – AMAT

Average memory access time is the average time to access memory considering both hits and misses and the frequency of different accesses.It is equal to the following:

$$\text{AMAT} = \text{Time for a hit} + \text{Miss rate} \times \text{Miss penalty}$$

13. What are the various block placement schemes in cache memory?

Direct-mapped cache is a cache structure in which each memory location is mapped to exactly one location in the cache.

Fully associative cache is a cache structure in which a block can be placed in any location in the cache.

Set-associative cache is a cache that has a fixed number of locations (at least two) where each block can be placed.

14. Define – MTTF and AFR

Reliability is a measure of the continuous service accomplishment or, equivalently, of the time to failure from a reference point. Hence, mean time to failure (MTTF) is a reliability measure. A related term is annual failure rate (AFR), which is just the percentage of devices that would be expected to fail in a year for a given MTTF.

15. Define – Availability

Availability is then a measure of service accomplishment with respect to the alternation between the two states of accomplishment and interruption. Availability is statistically quantified as

$$\text{Availability} = \frac{\text{MTTF}}{(\text{MTTF} + \text{MTTR})}$$

16. What are the three ways to improve MTTF?

The three ways to improve MTTF are:

1. Fault avoidance: Preventing fault occurrence by construction.
2. Fault tolerance: Using redundancy to allow the service to comply with the service specification despite faults occurring.
3. Fault forecasting: Predicting the presence and creation of faults, allowing the component to be replaced before it fails.

17. Define – TLB

Translation-Lookaside Buffer (TLB) is a cache that keeps track of recently used address mappings to try to avoid an access to the page table.

18. What is meant by virtual memory?

Virtual memory is a technique that uses main memory as a "cache" for secondary storage. Two major motivations for virtual memory: to allow efficient and safe sharing of memory among multiple programs, and to remove the programming burdens of a small, limited amount of main memory.

19. Differentiate physical address from logical address.

Physical address is an address in main memory.

Logical address (or) virtual address is the CPU generated addresses that corresponds to a location in virtual space and is translated by address mapping to a physical address when memory is accessed.

20. Define – Page Fault

Page fault is an event that occurs when an accessed page is not present in main memory.

21. What is meant by address mapping?

Address translation also called address mapping is the process by which a virtual address is mapped to an address used to access memory.