VV COLLEGE OF ENGINEERING

VV NAGAR ,TISAIYANVILAI

DEPARTMENT OF ECE

EC 8552 COMPUTER ARCHITECTURE AND ORGANIZATION

PART-A-TWO MARK QUESTIONS AND ANSWERS

UNIT I

COMPUTER ORGANIZATION & INSTRUCTIONS

1. Define Computer Architecture.

- It is concerned with the structure and behavior of the computer.
- It includes the information formats, the instruction set and techniques for addressing memory.

2. Define Computer Organization.

- It describes the function and design of the various units of digital computer that store and process information.
- It refers to the operational units and their interconnections that realize the architectural specifications**.**

3. What are the components of a computer.(Nov/Dec 2012)

- Input unit
- Memory unit
- Arithmetic and Logic Unit
- Output unit
- Control unit

4. List the eight ideas invented by computer architecture(April/May 2015)

1. Design for moore's law
2. Use abstraction to simplify design
3. Make the common case fast
4. Performance via parallelism
5. Performancevis pipelining
6. Performance via prediction

7.Hierarchy of memory

8.Dependability via redundancy

5. What is CPU execution time, user CPU time and system CPU time?(Nov/Dec 2015)

CPU time : The actual time the CPU spends computing for a specific task.

user CPU time : The CPU time spent in a program itself.

system CPU time : The CPU time spent in the operating system performing tasks on behalf the program.

6. State and explain the performance equation?(May/June 2014)

$$T = (N \times S) / R$$

- N denotes number of machine Instructions,
- Suppose that the average number of basic steps needed to execute one machine instruction is S, where each basic step is completed in one clock cycle. If the clock cycle rate is R cycles per second, the processor time is given by
- This is often referred to as the basic performance equation.

7. What is instruction set architecture(Nov/Dec 2015)

- Instruction set is a significant aspect of computer architecture.
- The instruction set selected for a particular computer determines the way,machine language programs are developed and executed
- Its design includes opcode and operand specification and the design of instruction types,to be included in processor's instruction set.
- Two major instructions set design approaches

  1)Reduced Instruction set Computer(RSIC)

  2)Complex Instruction set computer(CISC)

8. Name two special purpose registers. (Nov/Dec 2014)

1)Index register

2)Stack pointer

9. What is AddressingModes?(May/June 2013)

- Thedifferent ways in which thelocation of an operand is specifiedinan instruction is calledas Addressingmode.
- It specifies how to calculate the effectiveness of an operand by using elements of effective information held in registers.

10. What are the different types of address instruction?(May/June 2013)

• Three-address instruction-it can be represented as

ADD A, B, C

Operands a,b are called source operand and c is called destination operand.

• Two-address instruction-it can be represented as

ADD A, B

• One address instruction-it can be represented as

LOAD A
ADD B
STORE C

11. What are the different types of addressing Modes?(May/June 2012)(April/May 2015)

Immediatemode
Register mode
Absolutemode
Indirect mode
Indexmode
Basewith index
Basewith indexand offset
Relative mode
Auto-increment mode
Auto-decrement mode

12. Define Register mode andAbsoluteMode with examples.(Nov/Dec 2013)

Register mode

- Theoperand is thecontents of the processorregister.
- Thename (address) of the register is given in the instruction.

AbsoluteMode(DirectMode):

- Theoperand is in new location.
- The address of this location is given explicitlyin theinstruction.

Eg: MOVE LOC,R2

- The aboveinstruction uses the registerand absolute mode.
- Theprocessor register is the temporarystoragewherethe data in the register are accessed usingregister mode.
- The absolute modecan representglobal variablesin theprogram.

Mode	AssemblerSyntax	Addressing Function
Register mode	Ri	EA=Ri
Absolutemode	LOC	EA=LO

Where**EA**-EffectiveAddress

13. Define Indirect addressingMode.(Nov/Dec 2008)

- The effective address of an operand is the contents of a register or main memory location whose address is given explicitly in the instruction.
- The indirection is denoted by name of the register or new address given in the instruction.
- Address of an operand(B) is stored into R1 register.Ifwewant this operand,we canget it through register R1(indirection).
- The register or new location that contains the address of an operand is calledthe **pointer.**

Mode	AssemblerSyntax	Addressing
Indirect	Ri,LOC	EA=[Ri] orEA=[LOC

14. What is a RelativeAddressing mode?(May/June 2012)(Nov/Dec 2014)

Relative Mode:

- TheEffectiveAddress is determined bytheIndexmode usingthe PCin placeof the general purposeregister (gpr).
- This mode can beusedtoaccess thedata operand.But itsmost common useis to specifythe target addressin branch instruction.Eg.Branch>0Loop
- It causes theprogram execution to gotothebranchtarget location.It is identified bythe name loop if the branch condition is satisfied.

Mode	AssemblerSyntax	Addressing Function
Relative	X(PC)	EA=[PC]+X

15. Define Auto-increment addressing mode.(April/May 2010)

- TheEffectiveAddress ofthe operand is thecontents of a register in the instruction.
- Afteraccessingthe operand,the contents of this register is automatically incremented to point to the next item in thelist.

Mode	Assemblersyntax	Addressing Function
Auto-increment	(Ri)+	EA=[Ri]; Increment Ri

16. Differentiate direct and indirect addressing mode(Nov/Dec 2012)

- In direct addressing mode, the address of the location of the operand is given explicitly as a part of the instruction.
- In indirect addressing mode, the effective address of the operand is the contents of a register or main memory location whose address is given explicitly in the instruction.

17. Define Auto-decrement addressing mode.(April/May 2010)

- TheEffectiveAddress ofthe operand is thecontents of a register in the instruction.
- Afteraccessingthe operand,the contents of this register is automatically decremented to pointto thenext item in thelist.

Mode	AssemblerSyntax	Addressing Function
Auto-decrement	-(Ri)	EA=[Ri]; Decrement Ri

18. Draw the block diagram of computer.

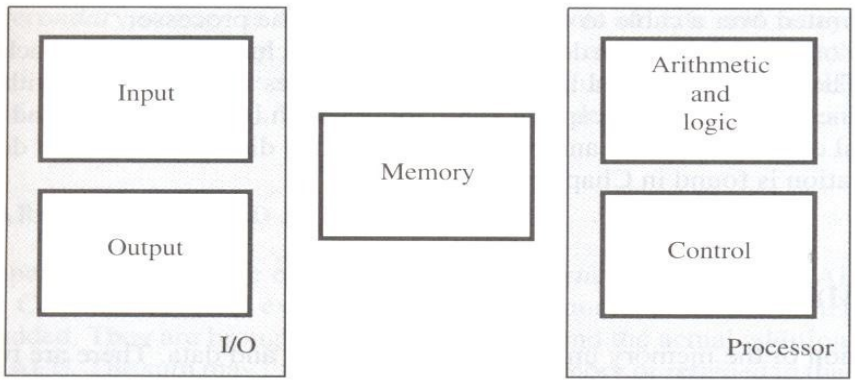


Figure 1.1 Basic functional units of a computer.

19. What is Execution time/Response time?

Response time 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.

20. What are the types of instructions According to number of addresses

- Three address instruction
- Two address instruction
- one address instruction
- Zero address instruction

21. What is clock cycleand clock period?

clock cycle : The time for one clock period, usually of the processor clock, which runs at a
constant rate.
clock period :The length of each clock cycle is called clock period.

22. Define CPI

The term Clock Cycles Per Instruction which is the average number of clock cycles each instruction takes to execute, is often abbreviated as CPI.

CPI=CPU clock cycles/Instruction count

23. Define MIPS and its rate

- MIPS:One alternative to time as the metric is MIPS(Million Instruction Per Second)

- MIPS=Instruction count/(Execution time x1000000).
- This MIPS measurement is also called Native MIPS to distinguish it from some alternative definitions of MIPS.
- The rate at which the instructions are executed at a given time.

24. Define Throughput and Throughput rate.

- Throughput -The total amount of work done in a given time.
- Throughput rate-The rate at which the total amount of work done at a given time.

25. What are the various types of operations required for instructions?

• Data transfers between the main memory and the CPU registers

• Arithmetic and logic operation on data

• Program sequencing and control

• I/O transfers

26. What is meant by Program and Computer Instruction?

- A program is a set of instructions that specify the operations, operands and the sequence by which processing has to occur.
- A Computer instruction is a binary code that specifies a sequence of micro operations for the computer.

27. Define Instruction Code and opcode?

- An instruction code is a group of bits that instruct the computer to perform a specific operation.
- The operation code of an instruction is a group of bits that define operations as add, subtract, multiply, shift and complement etc.

28. Define Instruction Format.

- Instructions are represented as numbers .
- Entire programs can be stored in memory to be read or written just like numbers(data).
- It simplifies software/Hardware of computer systems.
- Each instruction is encoded in binary called machine code.

29. What are the Most Common Fields Of An Instruction Format?

- An operation code field that specifies the operation to be performed.
- An address field that designates, a memory address or register.

- A mode field that specifies the way the operand or the effective address is determined

30. What is the straight-line sequencing?

The CPU control circuitry automatically proceed to fetch and execute instruction, one at a time in the order of the increasing addresses. This is called straight line sequencing.

31. Write down the MIPS Assembly language notation for arithmetic operations.

Category	Instruction	Example	Meaning	Comment
Arithmetic	add	add	$s1=$s2+$s3	Threeregister operands
	subtract	sub	$s1=$s2–$s3	Threeregister operands
	addimmediate	addi$s1,$	$s1=$s2+20	Usedtoaddconstants

32. Write down the MIPS Assembly language notation for Logical operations.

Category	Instruction	Example	Meaning	Comments
Logical	and	and	$s1=$s2&$s3	Threereg.operands;bit-by-
	or	or	$s1=$s2|$s3	Threereg.operands;bit-by-
	nor	nor	$s1=~($s2|$s3)	Threereg.operands;bit-by-
	andimmediate	andi$s1,$s2	$s1=$s2&20	Bit-by-bit
	orimmediate	ori$s1,$s2,	$s1=$s2|20	Bit-by-bit
	shiftleftlogica	sll$s1,$s2	$s1=$s2<<10	Shiftleftbyconstant
	shiftrightlogic	srl$s1,$s2,	$s1=$s2>>10	Shiftrightbyconstant

33. Write down the MIPS Assembly language notation for conditional branch operations.

Category	Instruction	Example	Meaning	Comments
Conditional branch	branchonequal	beq$s1,$s2,25	if($s1==$s2)goto PC+4+100	Equaltest; PC-relativebranch
	branchonnotequal	bne$s1,$s2,25	if($s1!=$s2)goto PC+4+100	Notequaltest; PC-relative
	setonlessthan	slt$s1,$s2,$s3	if($s2<$s3)$s1=1; else$s1=0	Comparelessthan;forbeq, bne
	setonlessthan	sltu$s1,$s2,$s3	if($s2<$s3)$s1=1; else$s1=0	Comparelessthanunsigned
	setlessthan	slti$s1,$s2,20	if($s2<20)$s1=1; else$s1=0	Comparelessthanconstant
	setlessthan immediateun	sltiu$s1,$s2,20	if($s2<20)$s1=1; else$s1=0	Comparelessthanconstant unsigned

34. What is a Immediateaddressing Mode?

- Theoperand is givenexplicitlyin theinstruction.

  Eg:Mov200 immediate,R0

- It places the value 200 inthe register R0.Theimmediatemode used to specifythe value ofsource operand.

- Inassemblylanguage, theimmediate subscript isnot appropriate so # symbol is used.

- It can bere-written as

  MOV #200,R0

Assembly Syntax: Operand=value

Addressing Function Immediate#value

35. Define Index addressing Mode.

- The effective address of an operand is generated by adding a constant value to the contents of a register.
- The constant value uses either special purpose or general purpose register.
- Index mode is represent symbolically as

  X(Ri)

  Where**X** –denotes the constant value contained in the instruction
  Ri–It is the nameof the register involved.

  TheEffectiveAddress ofthe operand is,

  EA=X+ [Ri]

- Theindexregister R1 contains the address ofanew location and the valueofX defines an offset(also calledadisplacement).

Index Mode	**AssemblerSyntax**	**Addressing Function**
Index	X(Ri)	EA=[Ri]+X
BasewithIndex	(Ri,Rj)	EA=[Ri]+[Rj]
BasewithIndexand offset	X(Ri,Rj)	EA=[Ri]+[Rj]+X

UNIT II

ARITHMETIC

1. State the principle of operation of a carry look-ahead adder.(April/May 2011,2013)

The input carry needed by a stage is directly computed from carry signals obtained from all the preceding stages i-1,i-2,…..0, rather than waiting for normal carries to supply slowly from stage to stage. An adder that uses this principle is called carry look-ahead adder.

2. What are the main features of Booth's algorithm?(Nov/Dec 2014)

1) It handles both positive and negative multipliers uniformly.

2) It achieves some efficiency in the number of addition required when the multiplier has a few large blocks of 1s.

3. How can we speed up the multiplication process?(Nov/Dec 2012)

There are two techniques to speed up the multiplication process:

1) The first technique guarantees that the maximum number of summands that must be added is n/2 for n-bit operands.
2) The second technique reduces the time needed to add the summands.

4. What is the advantage of using Booth algorithm?(April/May 2008)

1) It handles both positive and negative multiplier uniformly.
2) It achieves efficiency in the number of additions required when the multiplier has a few large blocks of 1's.
3) The speed gained by skipping 1's depends on the data.

5. Discuss the principle behind Booth's multiplier(April/May 2008)

- Booth's algorithm generates a 2n bit product and treats both positive and negative numbers uniformly.
- This algorithm suggests that we can reduce the number of operations required for multiplication by representing multiplier as a difference between 2 numbers.

6. Discuss the role of Booth's algorithm in design of fast multipliers.(April/May 2009)

- To speedup the multiplication process in Booth's algorithm a techniques called bit pair recoding is used.
- It is also called modified Booth's algorithm.
- It halves the maximum number of summands.
- Booth-recoded multiplier bits are grouped in pairs..Then each pair is represented by its equivalent single bit multiplier reducing total number of multiplier bits to half.

7. Write the algorithm for restoring division.(Nov/dec 2013)

Do the following for n times:

1) Shift A and Q left one binary position.

2) Subtract M and A and place the answer back in A.

3) If the sign of A is 1, set q0 to 0 and add M back to A.

Where A- Accumulator, M- Divisor, Q- Dividend.

8. Write the algorithm for non restoring division.(Nov/dec 2013)

Do the following for n times:

Step 1: Do the following for n times:

1) If the sign of A is 0 , shift A and Q left one bit position and subtract M from A; otherwise , shift A and Q left and add M to A.

2) Now, if the sign of A is 0,set q0 to 1;otherwise , set q0 to0.

Step 2: if the sign of A is 1, add M to A.

9. Write the Add/subtract rule for floating point numbers.(Nov/Dec 2008)

1) Choose the number with the smaller exponent and shift its mantissa right a number of steps equal to the difference in exponents.
2) Set the exponent of the result equal to the larger exponent.
3) Perform addition/subtraction on the mantissa and determine the sign of the result
4) Normalize the resulting value, if necessary.

10. What are the ways to truncate the guard bits? (May/June 2014)

There are several ways to truncate the guard bits:

1) Chooping
2) Von Neumann rounding
3) Rounding

11. Define Little endian arrangement(Nov/Dec 2014)

Little-endian is used for the opposite ordering, where the lower byte addresses are used for the less significant bytes (the rightmost bytes) of the word, addressing is called little-endian

12. What are the difficulties faced when we use floating point arithmetic?(Nov/Dec 2013)

- **Mantissa overflow**: The addition of two mantissas of the same sign may result in a carryout of the most significant bit
- **Mantissa underflow**: In the process of aligning mantissas ,digits may flow off the right end of the mantissa.
- **Exponent overflow**: Exponent overflow occurs when a positive exponent exceeds the maximum possible value.
- **Exponent underflow**: It occurs when a negative exponent exceeds the maximum possible exponent value.

13. In conforming to the IEEE standard mention any four situations under which a processor sets exception flag.(April/May 2011)

- **Underflow:** If the number requires an exponent less than -126 or in a double precision, if the number requires an exponent less than -1022 to represent its normalized form the underflow occurs.
- **Overflow:** In a single precision, if the number requires an exponent greater than -127 or in a double precision, if the number requires an exponent greater than +1023 to represent its normalized form the underflow occurs.
- **Divide by zero**: It occurs when any number is divided by zero.
- **Invalid**: It occurs if operations such as 0/0 are attempted.

14. Give the booth's recoding and bit-pair recoding of the computer. 1000111101000101 (May/June 2006)

Booth's recoding

1	0	0	0	1	1	1	1	0	1	0	0	0	1	0	1	0
-1	0	0	+1	0	0	0	-1	+1	-1	0	0	+1	-1	+1	-1	

Bit-Pair recoding:

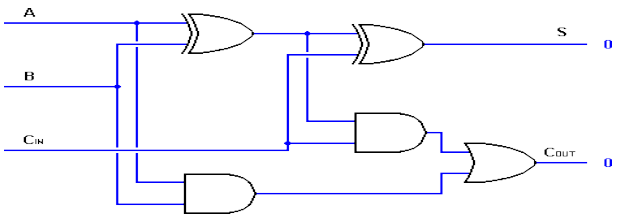
1	0	0	0	1	1	1	1	0	1	0	0	0	1	0	1	0
	-2		+1		0		-1		+1		0		+1		1	

15. Why floating point number is more difficult to represent and process than integer?(CSE May/June 2007)

An integer value requires only half the memory space as an equivalent. IEEE double-precision floating point value. Applications that use only integer based arithmetic will therefore also have significantly smaller memory requirement.

A floating-point operation usually runs hundreds of times slower than an equivalent integer based arithmetic operation.

16. Draw the full adder circuit and give the truth table (CSE May/June 2009)



Inputs			Outputs	
A	B	C	Carry	Sum
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	*1*	0
1	1	0	1	0
1	1	1	1	1

17. What is subword parallelism (April/May 2015)

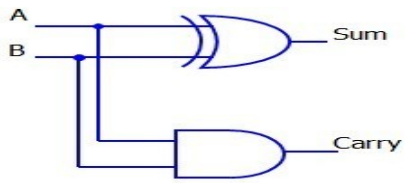
- Partioning the carry chains for the size of the data item within the vector is 128 bits ,a processor can use parallelism to perform simultaneous operation on short vectors of sixteen 8 bit operands ,eight 16-bit operands,four 32 bit operands or two 64 bit operands.
- Such parallelism is known as subword parallelism.

18. How overflow is detected in fixed point arithmetic?(Nov/dec 2013)

- Overflow can occur only when adding two numbers that have same sign.
- When both operands a and b have the same sign,an overflow occurs when the sign of result does not agree with signs of **a** and b.

19. What is half adder and draw the half adder circuit?(April/May 2009)

- A half adder is a logic circuit with two inputs and two outputs, which adds two bits at a time, producing a sum and a carry.



20. State the representation of double precision floating point number(Nov/Dec 2015)

- The 64-bit standard representation is called double precision representation because it occupies two 32 bit words.
- The 64 bits are divided into three fields as shown below
- (Field 1) Sign- 1 bit
- (Field 2)Exponent- 1 bit
- (Field 3)Mantissa -52 bits

21. What is bit pair recoding? Give an example.

Bit pair recoding halves the maximum number of summands. Group the Booth-recoded multiplier bits in pairs and observe the following: The pair (+1 -1) is equivalent to the pair (0 +1). That is instead of adding -1 times the multiplicand m at shift position i to +1 (M at position i+1, the same result is obtained by adding +1 (M at position i).

Eg: 11010 – Bit Pair recoding value is 0 -1 -2

22. Define carry save addition(CSA) process.

Instead of letting the carries ripple along the rows, they can be saved and introduced into the next row at the correct weighted position. Delay in CSA is less than delay through the ripple carry adder.

23. Write the multiply rule for floating point numbers.

1) Add the exponent and subtract 127.

2) Multiply the mantissa and determine the sign of the result .

3) Normalize the resulting value , if necessary.

24. What is the purpose of guard bits used in floating point arithmetic

Although the mantissa of initial operands are limited to 24 bits, it is important to retain extra bits, called as guard bits.

25. Explain about the special values in floating point numbers.

The end values 0 to 255 of the excess-127 exponent E(are used to represent special values such as:

- When E(= 0 and the mantissa fraction M is zero the value exact 0 is represented.
- When E(= 255 and M=0, the value (is represented.
- When E(= 0 and M (0 , denormal values are represented.
- When E(= 2555 and M(0, the value represented is called Not a number.

26. What are generate and propagate function?

The generate function is given by **Gi=xiyi**

The propagate function is given as **Pi=xi+yi**.

27. What is floating point numbers?

In some cases, the binary point is variable and is automatically adjusted as computation proceeds. In such case, the binary point is said to float and the numbers are called floating point numbers.

28. In floating point numbers when so you say that an underflow or overflow has occurred?

In single precision numbers when an exponent is less than -126 then we say that an underflow has occurred. In single precision numbers when an exponent is less than +127 then we say that an overflow has occurred.

29. What is full adder ?

A full adder is logic circuit with three inputs and two outputs, which adds three bits at a time giving a sum and a carry.

30. When can you say that a number is normalized?

When the decimal point is placed to the right of the first (nonzero) significant digit, the number is said to be normalized.

UNIT III THE PROCESSOR

1. Define pipelining.(May /June 2015)

Pipelining is a technique of decomposing a sequential process into sub operations with each sub process being executed in a special dedicated segment that operates concurrently with all other segments.

2. Definedatapath.(May/June 2014)

- **Datapath** is a collection of functional units, such as arithmetic logic units or multipliers, that perform data processing operations, registers, and buses.
- It composes the central processing unit (CPU).

3. Define instruction pipeline.(May/June 2014)

- The transfer of instructions through various stages of the CPU instruction cycle., including fetch opcode, decode opcode, compute operand addresses. Fetch operands, execute Instructions and store results.
- This amounts to realizing most (or) all of the CPU in the form of multifunction pipeline called an instruction pipelining.

4. What are Hazards?(May/June 2013)

A hazard is also called as hurdle .The situation that prevents the next instruction in the instruction stream from executing during its designated Clock cycle. Stall is introduced by hazard. (Ideal stage)

5. State different types of hazards that can occur in pipeline.(May/June 2013) (Nov/Dec 2015)

The types of hazards that can occur in the pipelining were,

1. Data hazards.

2. Instruction hazards.

3. Structural hazards.

6. Define Data hazards(May/June 2013)

A data hazard is any condition in which either the source or the destination operands of

an instruction are not available at the time expected in pipeline. As a result some operation has

to be delayed, and the pipeline stalls.

7. Define Instruction hazards(May/June 2012)

The pipeline may be stalled because of a delay in the availability of an instruction. For

example, this may be a result of miss in cache, requiring the instruction to be fetched from the

main memory. Such hazards are called as Instruction hazards or Control hazards.

8. Define Structural hazards?(MAY/June 2014)

The structural hazards is the situation when two instructions require the use of a given

hardware resource at the same time. The most common case in which this hazard may arise is

access to memory.

9. How data hazard can be prevented in pipelining?(MAY/June 2012)

Data hazards in the instruction pipelining can prevented by the following techniques.

a) Operand Forwarding

b) Software Approach

10. How addressing modes affect the instruction pipelining?(May/June 2014)

- Degradation of performance is an instruction pipeline may be due to address dependency where operand address cannot be calculated without available informatition needed by addressing mode
- for e.g. An instructions with register indirect mode cannot proceed to fetch the operand if the previous instructions is loading the address into the register.
- Hence operand access is delayed degrading the performance of pipeline.

11. What is locality of reference?(May/June 2014)

Many instruction in localized area of the program are executed repeatedly during some time period and the remainder of the program is accessed relatively infrequently .this is referred as locality of reference.

12. What would be the effect ,if we increase the number of pipelining stages? (Nov/Dec 2011)

- As the number of pipeline stages increases, the probability of pipeline being stalled also increases because more instructions are being executed concurrently.
- Thus the dependancies between instructions that are far apart may still cause the pipeline to stall.
- Increase in pipeline stage may result in significant branch prediction.

13. Define memory cycle time.(Nov/Dec 2014)

The minimum time delay required between the initiations of two successive memory operations, for example, the time between two successive READ operations.

14. What is exception? (Nov/Dec 2014)

- An *exception* is an internally generated unscheduled event, which occurs during the execution of a program, that disrupts the normal flow of the program's instructions.
- They are used to detect overflow.

15. What is meant by branch prediction?(Nov/Dec 2015)

- Branch prediction is a technique used to check whether the branch will be valid or not valid.
- This technique reduces the branch penalty.

16. What is meant by pipelining control?(April/May 2015)

- Pipeline control implements decode portion of fetch/execute loop.
- Part of decode involves translating opcode into control signals.

17. What is meant by speculative execution?(May/June 2012)

- Specualative execution means that instructions are executed before the processor is certain that they are in correct execution sequence.

- Modern pipelinedmicroprocessors use speculative execution to reduce the cost of conditional branch instructions using schemes that predict the execution path of a program based on the history of branch executions.

18. **What is the role of cache in pipelining?(Nov/Dec 2011)**

    - If each instruction is fetched from main memory ,pipeline is useless because main memory is slow compared to execution.
    - The cache memory reduces the memory access time and makes pipelining useful.

19. **Define parallel processing.**

- Parallel processing is a term used to denote a large class of techniques that are used to provide simultaneous data-processing tasks for the purpose of increasing the computational speed of a computer system.
- Instead of processing each instruction sequentially as in a conventional computer, a parallel processing system is able to perform concurrent data processing to achieve faster execution time

20. **Define memory access time**

The time that elapses between the initiation of an operation and completion of that operation ,for example ,the time between the READ and the MFC signals .This is Referred to as memory access time.

21. **What are the steps required for a pipelinened processor to process the instruction?**

- • F Fetch: read the instruction from the memory
- • D Decode: decode the instruction and fetch the source operand(s).
- • E Execute: perform the operation specified by the instruction.
- • W Write: store the result in the destination location

22. **What are the classification of data hazards?**

Classification of data hazard: A pair of instructions can produce data hazard by referring

reading or writing the same memory location. Assume that i is executed before J. So, the hazards

can be classified as,

1. RAW hazard

2. WAW hazard

3. WAR hazard

23. Define RAW hazard : (read after write)and WAR hazard :(write after read)

- Instruction ‘j’ tries to read a source operand before instruction ‘i’ writes it.
- Instruction ‘j’ tries to write a source operand before instruction ‘i’ reads it.

24. Define WAW hazard :(write after write)

Instruction ‘j’ tries to write a source operand before instruction ‘i’ writes it.

25. How Compiler is used in Pipelining?

- A compiler translates a high level language program into a sequence of machine instructions.
- To reduce N, we need to have suitable machine instruction set and a compiler that makes good use of it.
- An optimizing compiler takes advantages of various features of the target processor to reduce the product N*S, which is the total number of clock cycles needed to execute a program.
- The number of cycles is dependent not only on the choice of instruction, but also on the order in which they appear in the program.
- The compiler may rearrange program instruction to achieve better performance of course, such changes must not affect of the result of the computation.

26. What is the need for reduced instruction chip?

• Relatively few instruction types and addressing modes.

• Fixed and easily decoded instruction formats.

• Fast single-cycle instruction execution.

• Hardwired rather than micro programmed control

27. Define Static Memories.

Memories that consist of circuits capable of retaining the state as long as power is applied are known as static memories.

28. List out various branching technique used in micro program control unit?

a) Bit-Oring

b) Using Conditional Variable

c) Wide Branch Addressing

29. How the interrupt is handled during exception?

* CPU identifies source of interrupt

* CPU obtains memory address of interrupt handles

* pc and other CPU status information are saved

* Pc is loaded with address of interrupt handler and handling program to handle it.

30. List out the methods used to improve system performance.

The methods used to improve system performance are

1. Processor clock

2.Basic Performance Equation

3.Pipelining

4.Clock rate

5.Instruction set

6.Compiler

31. What is called static and dynamic branch prediction?

- Static branching prediction-Branch prediction decision is always same at every time for a given instruction is executed.
- Dynamic branching prediction-- Branch prediction decision may change depending on execution history

32. What is Instruction Level Parallelism(ILP)? (NOV/DEC 2011)(Nov/Dec 2015)

Pipelining is used to overlap the execution of instructions and improve performance. This potential overlap among instructions is called instruction level parallelism (ILP).

33. State the need for speculation?(Nov/Dec 2014)

- It is the most important method for finding and exploiting more ILP.
- Compiler can use speculation to reorder instructions.
- Processor hardware can reorder instructions at runtime.

34. Explain various types of Dependences in ILP.

- Data Dependences
- Name Dependences

35. State the Amdahl's law?(Nov/Dec 2014)

- It calculates the performance gain that can be obtained by improving some portion of a computer.

$$\text{Speed up} = \frac{\text{Performance of entire task using improved machine}}{\text{Performance of entire task using old machine}}$$

or

$$\text{Speed up} = \frac{\text{Execution time for entire task using improved machine}}{\text{Execution time for entire task using original machine}}$$

Unit-IV

MEMORY AND I/O ORGANIZATION

1. What are the various memory technologies?(Nov/Dec 2015)

1) Random Access Memory(RAM)

- Static RAM
- Dynamic RAM(DRAM)

2) Read Only Memory(ROM)

- Programmable Read Only Memory(PROM)
- Erasable Programmable Read Only Memory(EPROM)
- Electrically Erasable Programmable Read OnlyMemory(EEPROM)

3)Flash Memory

4)Synchronous DRAM (SDRAM)

2. Define Hit ratio(Nov/Dec 2015)

- The performance of memory is frequently measured in terms of quantity is called hit ratio.
- The ratio of number of hits is divided by the total cpu reference of memory is called hit ratio.

Hit ratio=number of hits/(number of hits+ number of miss)

3. Distinguish Between Static RAM and Dynamic RAM?(May/June 2008)

- Static RAM are fast, but they come at high cost because their cells require several transistors.
- Less expensive RAM can be implemented if simpler cells are used.
- However suchcells do not retain their state indefinitely;
- Hence they are called Dynamic RAM.

4. Distinguish between asynchronous DRAM and synchronous RAM.(Nov/Dec 2008)

- The specialized memory controller circuit provides the necessary control signals, RAS

  And CAS ,that govern the timing.

- The processor must take into account the delay in the response of the memory.
- Such memories are referred to as asynchronous DRAMS.The DRAM whoseoperations is directly synchronized with a clock signal.
- Such Memories are known as synchronous DRAM.

5. What is meant by dirty bit or modified bit(Nov/Dec 2014)

- The cache location is updated with an associated flag bit is called dirty bit.
- It is a bit that is associated with a block of computer memory and indicates whether or not the corresponding block of memory has been modified

6. How the interrupt is handled during exception?(May/June 2012)

* cpu identifies source of interrupt

* cpu obtains memory address of interrupt handles

* pc and other cpu status information are saved

* Pc is loaded with address of interrupt handler and handling program to handle it

7. What is an interrupt and give its uses ?(May/June 2012)

An interrupt is an event that causes the execution of one program to be suspended and another program to be executed.

*Recovery from errors

*Debugging

*Communication between programs

*Use of interrupts in operating system

8.Define vectored interrupts.(Nov/Dec 2013)

- In order to reduce the overhead involved in the polling process, a device requesting an interrupt may identify itself directly to the CPU.
- Then, the CPU can immediately start executing the corresponding interrupt-service routine.
- The term vectored interrupts refers to all interrupt handling schemes base on this approach.

9. What is DMA ? State its advantages(May/June 2014)

- Transfer a block of data directly between an external device and main memory without contiguous intervention by CPU.This approach is called DMA.
- Main advantages are
- Data transfer is very fast.
- Processor is not involved in data transfer operation and hence it is free to execute other tasks.

10. Define word length(Nov/Dec 2014)

- The size of a word is reflected in many aspects of a computer's structure and operation
- Each group of n bits is referred to as a word of information and n is called the word length.

11. Name some of the IO devices.(Nov/Dec 2014)

*Video terminals

*Video displays

*Alphanumeric displays

*Graphics displays

* Flat panel displays

*Printers

*Plotters

12. What is programmed I/O?(Nov/Dec 2014)

- Data transfer to and from peripherals may be handled using this mode.
- If I/O operations in computer system are completely controlled by the processor ,then the system is said to be using programmed I/O

13. Give the features of ROM cell(May/June 2008)

- It can hold one bit of data.
- It can hold data even if power is turned off.

14. Define spatial locality and temporal locality(April/May 2008)

- Spatial locality means that the instructions stored near by the to the recently executed instruction are also likely to be executed soon.
- Temporal locality means that recently executed instruction is likely to be executed again very soon.

15. What is mapping function and list out its types?(June 2009)

- The correspondence between main memory blocks and those in cache is specified by mapping function.
- Types of mapping procedures
- Associative mapping.
- Direct mapping.
- Set-associative mapping

16. What is memory mapped I/O/(May/June 2014)

The technique in which the total memory address space is partitioned and part of this space is devoted to I/O addressing is called memory mapped I/O.

17. What is meant by interleaved memory?(April/May 2013)

- Interleaved memory is a design made to compensate for the relatively slow speed of dynamic random-access memory (DRAM) or core memory, by spreading memory addresses evenly across memory banks.
- This results in contiguous reads (which are common both in multimedia and execution of programs) and contiguous writes.

18. What is TLB?(Nov/Dec 2011)

- Translation look-aside buffer (TLB) is a memory cache that stores recent translations ofvirtual memory to physical addresses for faster retrieval.
- Most processors include TLBs to increase the speed of virtual memory operations through the inherent latency-reducing proximity as well as the high-running frequencies of current CPU's.

19. An Address space is specified by 24 bits and corresponding memory space by 16 bits.How many words are there in 1)virtual memory 2)Main memory (April/May 2012)

Words in virtual memory=2^{24}

=16 M words

Words in main memory=2^{16} =64 K words

20.Differentiate Programmed I/O and interrupt I/O (May/June 2014)

Programmed I/O	Interrupt I/O
It is implemented without interrupt hardware support	It is implemented using interrupt hardware support
It does not depend on interrupt status	Interrupt must be enabled to process interrupt driven I/O
It does not need initialization of stack	It need initialization of stack
System Throughput decreases as number of I/O devices connected in the system increases.	System Throughput does not depend on number of I/O devices connected in the system

21. Explain virtual memory technique.

- Techniques that automatically move program and data blocks into the physical memory
  when they are required for execution are called virtual memory technique
- It extends the apparent size of the physical memory

22. What are virtual and logical addresses?

The binary addresses that the processor issues for either instruction or data are called

virtual or logical addresses.

23. Define translation buffer.

- Most commercial virtual memory systems incorporate a mechanism that can avoid the bulk of the main memory access called for by the virtual to physical addresses translation buffer.
- This may be done with a cache memory called a translation buffer.

24. What is branch delay slot?

The location containing an instruction that may be fetched and then discarded because of

the branch is called branch delay slot.

25. What is the role of MAR and MDR?

The MAR (memory address register) is used to hold the address of the location to or from

which data are to be transferred and the MDR(memory data register) contains the data to be

written into or read out of the addressed location.

26. What do you mean associative mapping technique?

The tag of an address received from the CPU is compared to the tag bits of each block of the cache to see if the desired block is present. This is called associative mapping technique.

27. What is SCSI?

Small computer system interface can be used for all kinds of devices including RAID

storage subsystems and optical disks for large- volume storage applications.

28. What is IO mapped input output?

A memory reference instruction activated the READ M (or)WRITE M control line and does not affect the IO device. Separate IO instruction are required to activate the READ IO and WRITE IO lines ,which cause a word to be transferred between the address aio port and the CPU. The memory and IO address space are kept separate.

29. What are the multimedia applications which use caches?

Some Multimedia application areas where cache is extensively used are

*Multimedia Entertainment

*Education

*Office Systems

*Audio and video Mail

*Computer Architecture - Set 6

30.Specify the three types of the DMA transfer techniques?

- Single transfer mode(cyclestealing mode)
- Block Transfer Mode(Brust Mode)
- Demand Transfer Mode
- Cascade Mode

31. Name any three of the standard I/O interface.

*SCSI (small computer system interface),bus standards

*Back plane bus standards

*IEEE 796 bus (multibus signals)

*NUBUS & IEEE 488 bus standard

32. What is an I/O channel?

An i/o channel is actually a special purpose processor, also called peripheral processor.The main processor initiates a transfer by passing the required information in the input output channel. the channel then takes over and controls the actual transfer of data.

33.What is a bus?

- A collection of wires that connects several devices is called a bus.
- It consist of two parts -- an address busand a data bus.
- The data bus transfers actual data whereas the address bus transfers information about where the data should go.

34. Why program controlled I/O is unsuitable for high-speed data transfer?

In program controlled i/o considerable overhead is incurred..because several program instruction have to be executed for each data word transferred between the external devices and MM.Many high speed peripheral; devices have a synchronous modes of operation.that is data transfer are controlled by a clock of fixed frequency, independent of the cpu.

35.what is the function of i/o interface?

- The function is to coordinate the transfer of data between the cpu and external devices.
- An I/O interface is required whenever the I/O device is driven by the processor.
- The interface must have necessary logic to interpret the device address generated by the processor.

36.what is NUBUS?

A NUBUS is a processor independent, synchronous bus standard intended for use in 32 bitmicro processor system. It defines a backplane into which upto 16 devices may be plugged each in the form of circuit board of standard dimensions.

37. What are the steps taken when an interrupt occurs?

*Source of the interrupt

*The memory address of the required ISP

* The program counter &cpu information saved in subroutine

*Transfer control back to the interrupted program

38.Define interface and list out its components

- The word interface refers to the boundary between two circuits or devices
- Components of I/O interface are
  - Data register
  - Status/control register
  - Address decoder
  - External devices interface logic

39.Define Synchronous bus.

- Synchronous bus on other hand contains synchronous clock that is used to validate each and every signal.

- Synchronous buses are affected noise only when clock signal occurs.

- Synchronous bus designers must control with meta stability when attempting different clock signal Frequencies

- Synchronous bus of meta stability arises in any flip flop. when time will be violated.

40. Define Asynchronous bus.

- Asynchronous buses can mistake noise pulses at any time for valid handshake signal.

- Asynchronous bus designer must deal with events that like synchronously.

- It must contend with meta stability when events that drive bus transaction.

UNIT-5

ADVANCED COMPUTER ARCHITECTURE

1. **Define parallel processing.**

   Processing data concurrently is known as parallel processing. There are two ways by which we can achieve parallelism. These are:

   Multiple functional units

   Multiple processors

2. **Define multiprocessor system.**

   A computer system with at least two processors is called multiprocessor system

3. **Define task-level or process level parallelism.**

   Utilizing multiple processors for executing independent programs simultaneously is known as task-level parallelism.

4. **What is cluster?**

   A set of computers connected over a load area network that function as a single large multiprocessor is called cluster.

5. **What is mean by CMPs?**

   The multicore architecture designs that allow single chip multiprocessing are known as chip multiprocessors

6. **What is Instruction Level Parallelism (ILP)? (NOV/DEC 2011)(Nov/Dec 2015)**

   Pipelining is used to overlap the execution of instructions and improve performance. This potential overlap among instructions is called instruction level parallelism (ILP).

7. **What is Multithreading? (Nov/Dec 2014)**

   Multithreading allows multiple threads to share the functional units of a single processor in anoverlapping fashion. To permit this sharing, the processor must duplicate the independent state ofeach thread.

8. **Write the advantages of Multithreading.(Nov/Dec 2011)**

- If a thread gets a lot of cache misses, the other thread(s) can continue, taking advantage of the unused computing resources, which thus can lead to faster overall execution, as these resources would have been idle if only a single thread was executed.
- If a thread cannot use all the computing resources of the CPU (because instructions depend on each other's result), running another thread permits to not leave these idle.
- If several threads work on the same set of data, they can actually share their cache, leading tobetter cache usage or synchronization on its values.
- Improves pipeline utilization by taking advantage of multiple threads.

9. What are the Disadvantages of SMT?(May/June 2008)

- Simultaneous multithreading cannot improve performance if any of the shared resources are limiting bottlenecks for the performance.
- It is a considerable burden to put on software developers that they have to test whether simultaneous multithreading is good or bad for their application in various situations and insert extra logic to turn it off if it decreases performance.

10. State the need for speculation?(Nov/Dec 2014)

- It is the most important method for finding and exploiting more ILP.
- Compiler can use speculation to reorder instructions.
- Processor hardware can reorder instructions at runtime.

11. What is coarse grained multithreading?(May/June 2014)

- It switches threads only on costly stalls. Thus it is much less likely to slow down the execution an individual thread.
- It needs hardware support
- It requires PC and register file for each thread.
- It does not cover short stalls

12. What are superscalar processors? (Nov/Dec 2015)

- Several instructions start execution in the same clock cycle, and the processor is said to use multiple issue.
- Such processors are capable of achieving an instruction execution throughput of more than one instruction per cycle. They are known as superscalar processors.

13. What is meant by flynn's classification?(Nov/dec 2014)

- In 1966, Michael Flynn proposed a classification for computer architectures
- It was based on the number of instruction streams and data streams (Flynn's Taxonomy).
- SISD:Single Instruction Single Data
- SIMD: Single Instruction Multiple Data
- MISD: Multiple Instructions Single Data
- MIMD: Multiple Instructions Multiple Data

14. State the Amdahl's law?(Nov/Dec 2014)

- It calculates the performance gain that can be obtained by improving some portion of a computer.

$$\text{Speed up} = \frac{\text{Performance of entire task using improved machine}}{\text{Performance of entire task using old machine}}$$

or

$$\text{Speed up} = \frac{\text{Execution time for entire task using improved machine}}{\text{Execution time for entire task using original machine}}$$

15. Define Strong scaling and week scaling(April/May 2015)

1)Strong scaling

Speedup achieved on multiprocessor without the size of the problem.

2)Weak scaling

Increasing the size of the problem proportionally to the increase in number of processors.

16. Explain various types of Dependences in ILP.

- Data Dependences
- Name Dependences
- Control Dependences

17. What are multiprocessors? Mention the categories of multiprocessors?

- Multiprocessor are used to increase performance and improve availability.
- The different categories are SISD, SIMD, MIMD.

18. What are two main approaches to multithreading?

- Fine-grained multithreading

- Coarse-grained multithreading

19. What is the need to use multiprocessors?

- Multiprocessor is contain two or more processing units (multiple processors) each sharing main memory and peripherals, in order to simultaneously process programs.
- This processors may share "some or all of the system's memory and I/O facilities"; it also gave tightly coupled system.

20. Write the disadvantages of Multithreading.

- Multiple threads can interfere with each other when sharing hardware resources such as caches or translation lookaside buffers (TLBs).
- Execution times of a single-thread are not improved but can be degraded, even when only one thread is executing.
- This is due to slower frequencies and/or additional pipeline stages that arc necessary to accommodate thread-switching hardware.
- Hardware support for Multithreading is more visible to software, thus requiring more changes to both application programs and operating systems than Multi processing.

21. What is CMT?

- Chip multiprocessors - also called multi-core microprocessors or CMPs are way to build high-performance microprocessors,
- Large uniprocessors are no longer scaling in performance, because it is only possible to extract a limited amount of parallelism from a typical instruction stream using conventional superscalar instruction issue techniques.

22. What is SMT?

- Simultaneous multithreading, often abbreviated as SMT, is a technique for improving the overall efficiency of superscalar CPUs with hardware multithreading.
- SMT permits multiple independent threads of execution to better utilize the resources provided by modern processor architectures.

23. Write the advantages of CMP?

CMPs have several advantages over single processor solutions energy and silicon
area efficiency

i. By Incorporating smaller less complex cores onto a single chip
ii. Dynamically switching between cores and powering down unused cores
iii. Increased throughput performance by exploiting parallelism

iv.Multiple computing resources can take better advantage of instruction, thread,
and process level

24. What is meant by Thread-level parallelism (TLP)?

- Explicit parallel programs already have TLP (inherent)
- Sequential programs that are hard to parallelize or ILP-limited can be speculatively parallelized in
  hardware.

25. List the major MIMD Styles

- Centralized shared memory ("Uniform Memory Access" time or "Shared Memory Processor")
- Decentralized memory (memory module CPU) get more memory bandwidth, lower memory
  Drawback: Longer communication latency
  Software model more complex

26. Distinguish between shared memory multiprocessor and message-passing multiprocessor.

- A multiprocessor with a shared address space, that address space can be used to communicatedata implicitly via load and store operations is shared memory multiprocessor.
- A multiprocessor with a multiple address space, communication of data is done by explicitlypassing message among processor is message-passing multiprocessor.

27. What is multicore'?

- Multi-core is a design in which a single physical processor contains the core logic of more than one processor.
- Intel Xeon processor were opened up and inside were packaged all the circuitry and logic for two (or more) Intel Xeon processors.
- The multi-core design takes several such processor "cores" and packages them as a single physical processor.
- The goal of this design is to enable a system to run more tasks simultaneously and thereby achieve greater overall system performance

.

28. Write the software implications of a multicore processor?

- Multi-core systems will deliver benefits to all software, but especially multi-threaded programs.

- All code that supports HT Technology or multiple processors, for example, will benefit automatically from multi-core processors, without need for modification.
- Most server-side enterprise packages and many desktop productivity tools fall into this category.

Unit-1

PART – B

1. Discuss about the various components of a computer system with neat diagram.
2. What is need for addressing mode in a computer system? Discuss different types of addressing modes with a suitable example.
3. Explain in brief the evolution of computer systems.
4. Explain the eight ideas invented for computer design.
5. Explain the features of various generation (Various Technology) and computers.
6. Explain how performance is calculated in a computer system and derive the necessary performance equation.
7. Explain with an example about the operations and operands of the computer hardware?
8. Explain how instructions that involve decision making are executed with an example.
9. Discuss various techniques to represent instructions in a computer system.
10. State the CPU performance equation and discuss the factors that affect performance.
11. Explain the addressing modes used and the operation done in every instruction.
    (1) Move (R5)+, R0 (2) Add(R5)+, R0 (3) Move 16(R5), R3 (4) Move R0, (R5) (5) Add #40,R5

Unit-2

Part-B

1. Explain in detail about carry look ahead adder (high speed adder).
2. Explain Booth multiplication algorithm with suitable algorithm.
3. Explain floating point addition and subtraction algorithm with a neat diagram.
4. Discuss in detail about restoring division algorithm in detail with diagram and example.
5. Perform X+Y and X-Y using 2's compliment for given two binary

numbers 6. X = 0000 1011 1110 1111 and Y = 1111 0010 1001 1101 (5 marks)

7. Multiply following signed 2's compliment numbers using the booth algorithm.
8. A= 001110 and B – 111001. Where A is multiplicand and B is Multiplier.
9. Explain in detail about multiplication algorithm with suitable example and diagram.
10. Explain the sequential version of multiplication algorithm and its hardware.
11. Multiply the following pair of signed numbers using Booth bit pair coding of the multiplier. A = +13 (multiplicand) and B = -6 (multiplier).

Unit-3

Part-B

1. Explain the basic MIPS implementation with necessary multiplexer and control lines.
2. Explain the different type of pipeline hazard with suitable examples
3. What is Hazard? Explain its types with suitable examples.
4. Explain in detail the operation of the data path and its control.
5. Explain in detail how exceptions are handled in MIPS architecture.

Unit-4

Part-B

1. Elaborate on the various memory technologies (memory chip organization)and its relevance.
2. Explain mapping function in catch memory to determine how memory blocks are placed in cache.
3. What is virtual memory? Explain the steps involved in virtual memory address translation.
4. Explain in detail about DMA controller with neat block diagram.
5. Explain in detail about any two standard input and output interfaces required to connect the I/O device to the bus.
6. Describe the principle approaches of Serial Bus Architectures with necessary diagrams.
7. Discuss the following in detail i). Input Devices.
   ii). Output Devices.
8. Explain in detail about the Parallel Bus Architectures.

Unit-5

Part-B

1. Explain instruction level parallel processing. State the challenges of parallel processing.
2. Explain the terms:
   (I) Hardware Multi-threading
   (II) Multicore processors.
3. Explain the detail Flynn's classification of parallel hardware.

   or

   Discuss about SISD, MIMD, SIMD, SPMD and VECTOR system.
4. Classify shared memory multiprocessor based on memory access.
5. List the limitations of instruction level parallelism.
6. Explain in detail about graphics processing units.
7. Illustrate the following in detail i). Clusters
   ii). Wharehouse scale computers