CS8491 COMPUTER ARCHITECTURE L T P C

3 0 0 3

- To learn the basic structure and operations of a computer.
- To learn the arithmetic and logic unit and implementation of fixed-point and floating point arithmetic unit.
- To learn the basics of pipelined execution.
- To understand parallelism and multi-core processors.
- To understand the memory hierarchies, cache memories and virtual memories.
- To learn the different ways of communication with I/O devices.

UNIT I BASIC STRUCTURE OF A COMPUTER SYSTEM 9

Functional Units – Basic Operational Concepts – Performance – Instructions: Language of the Computer – Operations, Operands – Instruction representation – Logical operations – decision making – MIPS Addressing.

UNIT II ARITHMETIC FOR COMPUTERS 9

Addition and Subtraction – Multiplication – Division – Floating Point Representation – Floating Point Operations – Subword Parallelism

UNIT III PROCESSOR AND CONTROL UNIT 9

A Basic MIPS implementation – Building a Datapath – Control Implementation Scheme – Pipelining – Pipelined datapath and control – Handling Data Hazards & Control Hazards – Exceptions.

UNIT IV PARALLELISIM 9

Parallel processing challenges – Flynn‘s classification – SISD, MIMD, SIMD, SPMD, and Vector Architectures – Hardware multithreading – Multi-core processors and other Shared Memory Multiprocessors – Introduction to Graphics Processing Units, Clusters, Warehouse Scale Computers and other Message-Passing Multiprocessors.

UNIT V MEMORY & I/O SYSTEMS 9

Memory Hierarchy – memory technologies – cache memory – measuring and improving cache performance – virtual memory, TLB‘s – Accessing I/O Devices – Interrupts – Direct Memory Access – Bus structure – Bus operation – Arbitration – Interface circuits – USB.

TOTAL : 45 PERIODS

OUTCOMES:

On Completion of the course, the students should be able to:

- Understand the basics structure of computers, operations and instructions.
- Design arithmetic and logic unit.
- Understand pipelined execution and design control unit.
- Understand parallel processing architectures.
- Understand the various memory systems and I/O communication.

TEXT BOOKS:

1. David A. Patterson and John L. Hennessy, Computer Organization and Design: The Hardware/Software Interface, Fifth Edition, Morgan Kaufmann / Elsevier, 2014.
2. Carl Hamacher, ZvonkoVranesic, SafwatZaky and NaraigManjikian, Computer Organization and Embedded Systems, Sixth Edition, Tata McGraw Hill, 2012.

REFERENCES:

1. William Stallings, Computer Organization and Architecture – Designing for Performance, Eighth Edition, Pearson Education, 2010.
2. John P. Hayes, Computer Architecture and Organization, Third Edition, Tata McGraw Hill, 2012.
3. John L. Hennessey and David A. Patterson, Computer Architecture – A Quantitative Approach‖, Morgan Kaufmann / Elsevier Publishers, Fifth Edition, 2012.

UNIT I BASIC STRUCTURE OF A COMPUTER SYSTEM

COURSE OBJECTIVE:

To learn the basic structure and operations of a computer.

SYLLABUS:

Functional Units – Basic operational concepts – Performance – Instructions: Language of the computer – operations, operands – Instruction representation – Logical operations – decision making – MIPS Addressing.

PART-A

1. What is the function of ALU? (R)

Most of the computer operations (arithmetic and logic) are performed in ALU.Thedata required for the operation is brought by the processor and the operation is performed by the ALU.

2. What is the function of CU? (R)

The control unit acts as the nerve center that coordinates all the computer operations. It issues timing signals that governs the data transfer.

3. What are basic operations of a computer? (R)

The basic operations are READ and WRITE.

4. Distinguish between auto increment and auto decrement addressing mode. (Apr/May 2010) (An)

It is generally used to increment or decrement the array pointer. For example while executing a loop the processor may require to increment or decrement the pointer to the adjacent address at each iteration. So it can be used to increment or decrement file pointers, or it can be used to implement stack in which the top can be incremented (TOP++) or decremented (TOP--).

5. What are the steps in executing a program? (R)

1. Fetch
2. Decode
3.Execute
4.Store

6. What is System Software? Give an example? (R)

It is a collection of programs that are executed as needed to perform functions such as

- Receiving and interpreting user commands
- Entering and editing application programs and storing them as files in secondary storage devices. **Ex:** Assembler, Linker, Compiler etc\

7. What is Application Software? Give an example? (R)

Application programs are usually written in a high-level programming language, in which the programmer specifies mathematical or text-processing operations. These operations are described in a format that is independent of the particular computer used to execute the program. **Ex:** C, C++, JAVA

8. What is a compiler? (R)
A system software program called a compiler translates the high-level language program into a suitable machine language program containing instructions such as the Add and Load instructions.

9. What is text editor? (R)
It is used for entering and editing application programs. The user of this program interactively executes command that allow statements of a source program entered at a keyboard to be accumulated in a file.

10. What is an opcode? (R) (Apr/May 2011)
Opcode(Operation Code) is the portion of a machine languageinstruction that specifies the operation to be performed. Their specification and format are laid out in the instruction set architecture of the processor.

11. What is elapsed time of computer system?(U)
The total time to execute the total program is called elapsed time. It is affected by the speed of the processor, the disk and the printer.

12. What is processor time of a program? (U)
The period during which the processor is active is called processor time of a program. It depends on the hardware involved in the execution of individual machine instructions.

13. Define clock rate? (R)
The clock rate is given by,
R=1/P,where P is the length of one clock cycle.

14. Write down the basic performance equation? (R)
T=N*S/R
T=processor time
N=no of instructions
S=no of steps
R=clock rate

15. What is pipelining? (R)
The overlapping of execution of successive instructions is called pipelining.

16. What is byte addressable memory? (R)
The assignment of successive addresses to successive byte locations in the memory is called byte addressable memory.

17. What is big endian and little endian format? (U)
The name big endian is used when lower byte addresses are used for the more significant of the word. The name little endian is used for the less significant bytes of the word.

18. What is a branch instruction? (R)
Branch instruction is a type of instruction which loads a new value into the program counter. As a result of branch instructions, the processor fetches and executes the instruction at a new address called

branch target, instead of the instruction at the location that follows the branch instruction in sequential address order.

19. Define addressing mode. (R)(APRIL/MAY 2009)(May/ June 2013)(Nov/Dec 2017)
The different ways in which the location of an operand is specified in an instruction are referred to as addressing modes.

20. What is Relative addressing mode? When it is used? (U)(May/ June 2012)(Dec-2014)
Relative addressing mode is used by branch instructions (e.g. BEQ, BNE, etc.) which contain a signed 8 bit relative offset (e.g. -128 to +127) which is added to program counter if the condition is true. As the program counter itself is incremented during instruction execution by two the effective address range for the target instruction must be with -126 to +129 bytes of the branch.

21. Define various addressing modes. (R)(May/ June 2013)
The various addressing modes are
1. Absolute addressing mode
2. Register addressing mode
3. Indirect addressing mode
4. Index addressing mode
5. Immediate addressing mode
6. Relative addressing mode
7.Autoincrement addressing mode
8. Auto decrement addressing mode

22. What is index register? (U)
In index mode the effective address of the operand is generated by adding a constant value to the contents of a register. The register used may be either a special register or may be any one of a set of general purpose registers in the processor. This register is referred to as an index register.

23. What is assembly language? (R)
A complete set of symbolic names and rules for the use of machines constitute a programming language, generally referred to as an assembly language.

24. What are the basic functional units of a computer? (R)
Input,memory,arithmetic and logic unit, output and control units are the basic functional units of a computer

25. Define Response time and Throughput. (R)
Response time is the time between the start and the completion of the event. Also referred to as execution time or latency. Throughput is the total amount of work done in a given amount of time

26. Suppose that we are considering an enhancement to the processor of a server system used for web serving. The new CPU is 10 times faster on computation in the web serving application than the original processor. Assuming that the original CPU is busy with computation 40% of the time and is waiting for I/O 60% of the time. What is the overall speedup gained by incorporating the enhancement? (Ap)
Fractionenhanced = 0.4

Speedupenhanced = 10
Speedupoverall = 1/(0.6+0.4/10) =1/0.64 = 1.56

27. How will you compute the SPEC rating? (R) (May/June 2012)

SPEC stands for system performance Evaluation Corporation

$$\text{SPEC rating} = \frac{\text{Running time on the reference computer}}{\text{Running time on the computer under test}}$$

28. Define power wall(R)

A powerwall is a large high-resolutiondisplay wall used for projecting large computer generated images. The size of a wall display can range in size from six to over forty feet in width and potentially even larger. For CMOS, the primary source of power dissipation is so called dynamic power that is, power that is consumed during switching. The dynamic power dissipation depends on the capacitive loading of each transistor, the voltage applied, and the frequency that the transistor is switched
Power=capacitive load x voltage^2 x frequency switched

29.What are fields available in MIPS (U)

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

- 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. This field selects the specific variant of the operation in the op field and is sometimes called the function code.

30. Compare RISC with CISC Architecture.(An) (Apr/May 2010) (Nov/Dec 2013)

RISC

- Simple instructions, few in number
- Fixed length instructions
- Complexity in compiler
- Only **LOAD/STORE** instructions access memory
- Few addressing modes

CISC

- Many complex instructions
- Variable length instructions
- Complexity in microcode
- Many instructions can access memory
- Many addressing modes

31. Describe the two control signals used for register transfer. (U)(May/June 2013)

For each register, two control signals are used to place the contents of that register on the bus or to load data on the bus into the register. The input and output of register Ri are connected to the bus via switches controlled by the signals Rin and iout, respectively. When Rin is set to 1, the data on the

bus are loaded into Ri. Similarly, when Rout is set to 1, the contents of register Ri are placed on the bus. While Rout is equal to 0, the bus can be used for transferring data from other registers.

32. State Amdahl's Law(R)(Dec-2014 / Apr/May 2019)

Amdahl's law, is used to find the maximum expected improvement to an overall system when only part of the system is improved. It is often used in parallel computing to predict the theoretical maximum speed up using multiple processors.

33. List the eight great ideas by computer architects. (R)(APR/MAY 2015)

- Design for Moore's law
- Use abstraction to simplify design
- Make the common case fast
- Performance via parallelism
- Performance via pipelining
- Performance via prediction
- Hierarchy of memories
- Dependability via redundancy

33. Distinguish Pipelining from Parallelism. (An)(APR/MAY 2015)

Parallel computing is the simultaneous execution of the same task (split up and specially adapted) on multiple processors in order to obtain results faster. The idea is based on the fact that the process of solving a problem usually can be divided into smaller tasks, which may be carried out simultaneously with some coordination.

Pipelining is a method of increasing system performance and throughput. It takes advantage of the inherent parallelism in instructions. Instructions are divided into 5 stages: IF, ID, EX, MEM, WB. In pipelining, we try to execute 2 or more instructions at the same time thereby increasing thethroughput.

34. What is instruction set architecture (R)(NOV/DEC 2015)

An instruction set, or instruction set architecture (**ISA**), is the part of the computer architecture related to programming, including the native data types, instructions, registers, addressing modes, memory architecture, interrupt and exception handling, and external I/O.

35. How CPU execution time for program is calculated (U)(NOV/DEC 2015)

- CPU time (or CPU Execution time)is the time between the start and the end of execution of a given program. This time
- Accounts for the time CPU is computing the given program, including operating system routines executed on the program's behalf, and it does not include the time waiting for I/O and running other programs.
- CPU time is a true measure of processor/memory performance
- Performance of processor/memory = 1 / CPU_time

36. What is an instruction register?(R)(NOV/DEC 2016)

An instruction register (IR) is the part of a CPU's control unit that holds the instruction currently being executed or decoded. In the instruction cycle, the instruction is loaded into the Instruction register after the processor fetches it from the memory location pointed by the program counter.

37. Give the formula for CPU execution time for a program(R)(NOV/DEC 2016)

- **CPU Time = I * CPI * T**

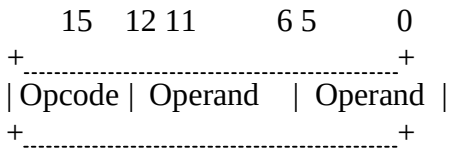
I = number of instructions in program
CPI = average cycles per instruction
T = clock cycle time

- **CPU Time = I * CPI / R**

R = 1/T the clock rate
T or R are usually published as performance measures for a processor
I requires special profiling software
CPI depends on many factors (including memory).

38. How to represent instruction in a Computer System? (U)(APR/MAY 2016)

Instructions are encoded as binary instruction codes. Each instruction code contains of aoperation code, or opcode, which designates the overall purpose of the instruction (e.g. add, subtract, move, input, etc.). The number of bits allocated for the opcode determined how many different instructions the architecture supports.



39. Distinguish between auto increment and auto decrement addressing mode?(An)(APR/MAY 2016)

Auto increment mode: The effective address of the operand is the contents of a register specified in the instruction. After accessing the operand, the contents of this register are automatically incremented to the next value. This increment is 1 for byte sized operands, 2 for 16 bit operands and so on.

E.g. Add (R2) +, R0

Here are the contents of R2 are first used as an E.A. then they are incremented.

Auto decrement mode: The effective address of the operand is the contents of a register specified in the instruction. Before accessing the operand, the contents of this register are automatically decremented and then the value is accessed.

E.g. Add - (R2), R0

Here are the contents of R2 are first decremented and then used as an E.A. for the operand which is added to the contents of R0. The auto increment addressing mode and the auto decrement addressing mode are widely used for the implementation of data structures like Stack. There may be other addressing modes that are unique to some processors. However the addressing modes mentioned above are common to many of the popular processors out there.

40. What are components of a computer system?(R) (Nov/Dec 2017)

- Motherboard.
- Processor.
- Memory (RAM)
- Case/chassis.
- Power supply.
- Floppy drive.

- Hard disk.
- CD-ROM, CD-RW, or DVD-ROM drive.

41. Write the equation for the dynamic power required per transistor. (R) (Apr/May 2018)
Subtract static power from total power in order to compute dynamic power. i.e Dynamic Power = Total Power - Static Power.

42. Classify the instructions based on the operations they perform and give one example to each category.(U) (Apr/May 2018)

Stack	**Accumulator**	**Register**
PUSH A	Load A	Load R1,A
PUSH B	ADD B	ADD R1,B
ADD	Store C	Store C,R1
POP C		

43. Consider three processors P1, P2, and P3 executing the same instruction set. They have clock rates of 3 GHz, 2.5 GHZ and 4.0 GHz respectively and CPI of 1.5, 1.0 and 2.2 respectively. Which processor has the highest performance expressed in instructions persecond?(Nov/Dec 2018)-Ap

Instructions per second = frequency/CPI

For P1 : 3×10^9Hz / 1.5 = 2×10^9instructions per second

For P2 : 2.5×10^9Hz / 1.0 = 2.5×10^9instructions per second

For P3 : 4×10^9Hz / 2.2 = 1.8×10^9instructions per second

P2 has the highest performance

44.Classify the instructions based on the operations they perform and give one example to each category. (Nov/Dec 2018)-R

Arithmetic Instructions- ADD A, Rn -Adds the register to the accumulator

Branch Instructions- ACALL addr11-Absolute subroutine call

Data Transfer Instructions-MOV A, Rn-Moves the register to the accumulator

Logic Instructions-ANL A,Rn -AND register to accumulator

Bit-oriented Instructions-CLR C-Clears the carry flag

45. Suppose that we are considering an enhancement to the processor of a server system used for Web Serving. The new CPU is 10 times faster on computation in the Web servingapplication than the original processor. Assuming that the original CPU is busy with computation 40% of the time and is waiting for I/O 60% of the time, what is the Overall speedup gained by incorporating the enhancement? (An) (Apr/May2019)
Fraction enhanced = 0.4
Speedup enhanced = 10

Speedup overall = 1/ ((1 – Fraction enhanced) + (Fraction enhanced)/ Speedup enhanced)
= 1 / (0.6 + 0.4/10)
= 1 /0.64
= 1.56

46. What are the various units in the computer ? (U) (Nov/Dec2020 / April/May2021)

- Input Unit
- Central Processing Unit (CPU)
- Arithmetic and Logic Unit (ALU)
- Control Unit

47. What is the absolute addressing mode ? (U) (Nov/Dec2020 / April/May2021)

An absolute address is represented by the contents of a register. This addressing mode is absolute in the sense that it is not specified relative to the current instruction address. Both the BranchConditional to Link Register instructions and the Branch Conditional to Count Register instructions use an absolute addressing mode.

PART B

1. Explain about functional units of computer? Discuss each with neat diagram?**(U) (APRIL/MAY 2009)**

2. Write notes on Instruction formats.**(U)(NOV/DEC 2007) & (NOV/DEC 2006)**

3. Explain about Instruction & Instruction Sequencing? **(U) (May/June 2013) (Nov / Dec 2013)(NOV/DEC 2006)(May/June2012)**

4. Explain in detail the different Instruction types and Instruction Sequencing**.(U)**

5. Explain the different types of Addressing modes with suitable examples. **(An) (APRIL/MAY 2008&2012)**

6. With examples explain the data transfer, logic and Program control instructions? **(Ap)(Apr/May 2011)**

7. List and explain the different types of instructions. Explain the types of instruction sequencing with an example for each. **(Ap)(NOV/DEC 2012)**

8. Why do we use addressing modes? Explain the different types of addressing modes with example. **(U) (NOV/DEC 2012)**

9. Explain different types of instructions with example. Compare their relative merits and demerits. Explain with an example how to multiply two unsigned binary numbers. **(An) (MAY/JUNE 2013)**

10. Explain the design of ALU in detail. **(U)(MAY/JUNE 2013) (Nov / Dec 2013)**

11. Explain the components of computer system**(U)(Dec-2014) (MAY/JUNE 2016)**

12. State the CPU performance equation and discuss the factors that affect performance**(An) (Dec-2014)**

13. Assume a 2 address format specified as source, destination. Examine the following sequence of instruction and explain the addressing modes used and the operation done in every instruction.**(10)(Ap) (Dec-2014)**

1) Move (R5)+R0
2) Add(R5)+R0
3) Move R0,(R5)
4) Move 16(R5),R3
5) Add #40,R5

14. Consider the computer with three instruction classes and CPI measurements as given below and instruction counts for each instruction class for the same program from two different compilers are

given. Assume that the computer's clock rate is 4 GHZ. Which code sequence will execute faster according to execution time? **(Ap)(6) (Dec-2014)**

Code from		CPI for this Instruction Class	
	A	B	C
CPI	1	2	3
Code from		Instruction Count for each Class	
	A	B	C
Compiler 1	2	1	2
Compiler 2	4	1	1

15. Discuss about the various techniques to represent instructions in a computer system. **(U)(APR/MAY 2015) (16)**

17. Explain in detail the various components of computer system with neat diagram**(U) (16) (NOV/DEC 2015)**

19. Explain the different types of addressing modes with suitable examples**(An)(APR/MAY 2015) (NOV/DEC 2015) (NOV/DEC 2016)(MAY/JUNE 2016) (Apr/May 2019)**

20. Explain various instruction formats and illustrate the same with an example. **(Nov/Dec 2017)**

23. Explain with an example about the operations and operands of the computer hardware. **(R)(Nov/Dec 2017)**

24. (i) Consider three different processors P1,P2 and P3 executing the same instruction set. P1 has a 3 GHz clock rate and a CPI of 1.5. P2 has a 2.5GHz clock rate and a CPI of 1.0. P3 has a 4 GHz clock rate and a CPI of 2.2.

a) Which processor has the highest performance expressed in instructions per second?

b) If the processors each execute a program in 10 seconds, find the number of cycles and the number of instructions in each processor. (5)

(ii) Explain in detail the components of a computer system. (8)**(An)(Apr/May 2018)**

25. (i) Translate the following C code to MIPS assembly code. Use a minimum number of instructions. Assume that I and k correspond to registers $s3 and $s5 and the base of the array save is in $s6. What is the MIPS assembly code corresponding to this C segment?

while(save[i]==k)

i+=1; **(An)** (5)

(ii) What is an addressing mode in a computer? Classify MIPS addressing modes and give one example instruction to each category.**(An)**(8)**(Apr/May 2018)**

26. Consider two different implementations of the same instruction set architecture. The Instructions can be divided into four classes according to their CPI (class A, B, C, and D) P1 With a clock rate of 2.5 GHZ and CPIs of 1, 2, 3, and 3 respectively, and P2 with a clock rate of 3 GHz and CPIs of 2, 2, 2, and 2 respectively. Given a program with a dynamic instruction count of 1.0x106 instructions divided into classes as follows:10% class A, 20% class B, 50% class C, and 20% class D, which implementation is faster? What is the global CPI for each implementation? Find the clock cycles required in both cases(7) **(Nov/Dec 2018)-An**

27. Explain the three broad classes of applications of computers. **(Nov/Dec 2018)-U**

28. Assume that the variables f and g are assigned to registers $S0 and $Sl respectively Assume that the base address of the array A is in register $S2 Assume f is zero initially

f = -g - A[4]

A[5] = r + 100 ,

Translate the above C statements into MIPS code . How many MIPS assembly instructions are needed to perform the C statements and how many different registers are needed to carry out the C statements?(5) **(Nov/Dec 2018)-Ap**

29. Define addressing mode in a computer. What are the different MIPS addressing modes? Give one example instruction to each category.(8) **(Nov/Dec 2018)-U**

30. Explain the eight ideas of the Computer architecture which empowered the computer design over the past decades**(U)**(7). **(Apr/May 2019)**

31. Tabulate the difference between the RISC and CISC processor(6) **(An) (Apr/May 2019)**

32. Explain the components of a computer with the block diagram in detail. **(U) (Nov/Dec2020 / April/May2021)**

33. What do you mean by addressing modes ? Explain various addressing modes with the help of examples. **(U) (Nov/Dec2020 / April/May2021)**

COURSE OUTCOME:

Students can able to understand the basic structure of computers, operations and instructions.

UNIT II ARITHMETIC

OPERATIONS

COURSE OBJECTIVE: To learn the arithmetic and logic unit and implementation of fixed-point and floating point arithmetic unit.

SYLLABUS:

Addition and subtraction– Multiplication–Division– FloatingPointRepresentation– Floating Point operations - Subword parallelism.

PART-A

1. What is skipping over of one's in Booth decoding? (U)

The Transformation 011… 110= +100…0 – 10 is called skipping over one's. In his case multiplier has its ones grouped into a few contiguous blocks.

2. What are the two attractive features of Booth Algorithm?(R)

- It handles both positive and negative multipliers uniformly
- It achieves some efficiency in the number of additions required when the multiplier has a few large blocks of ones

3. Give an example for the worst case of Booth algorithm (U)

In the worst case each bit of the multiplier selects the summands. This results in more number of summands.

4. What are the two techniques for speeding up the multiplication operation?(R)

- Bit Pair recoding
- CSA

5. How bit pair recoding of multiplier speeds up the multiplication process? (U)
It guarantees that the maximum number of summands that must be added is n/2 for nbit operands.

6. How CSA speeds up multiplication? (U)
It reduces the time needed to add the summands. Instead of letting the carries ripple along the rows, they can be saved and introduced into the next row, at the correct waited position.

7. Write down the levels of CSA steps needed to reduce k summands to two vectorsin CSA (U)
The number of levels can be shown by 1.7log2k-1.7

8. Write down the steps for restoring division and non-restoring division (R)
Non Restoring:
Step1: Do the following 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 to 0
Step 2: If the sign of A is 1, add M to A
Restoring:
§ Shift A and Q left one binary position
§ Subtract M from A
§ If the sign of A is one, set Q0 to 0, add M back to A otherwise set Q0 to 1

9. What is the advantage of non restoring over restoring division? (U)
Non restoring division avoids the need for restoring the contents of register afteran successful subtraction.

10. What is the need for adding binary 8 values to the true exponential in floating point numbers? (U)
This solves the problem of negative exponent. Due to this the magnitude of the numbers can be compared. The excess-x representation for exponents enables efficient comparison of the relative sizes of the two floating point numbers.

11. Briefly explain the floating point representation with an example? (U)
The floating point representation has 3 fields
1. Sign bit
2. Significant bits
3. Exponent
For example consider 1.11101100110 x 10^5,
Mantissa=11101100110
Sign=0
Exponent=5

12. What are the 2 IEEE standards for floating point numbers?(R)
1. Single
2. Double

13. What is overflow, underflow case in single precision(sp)?(R)

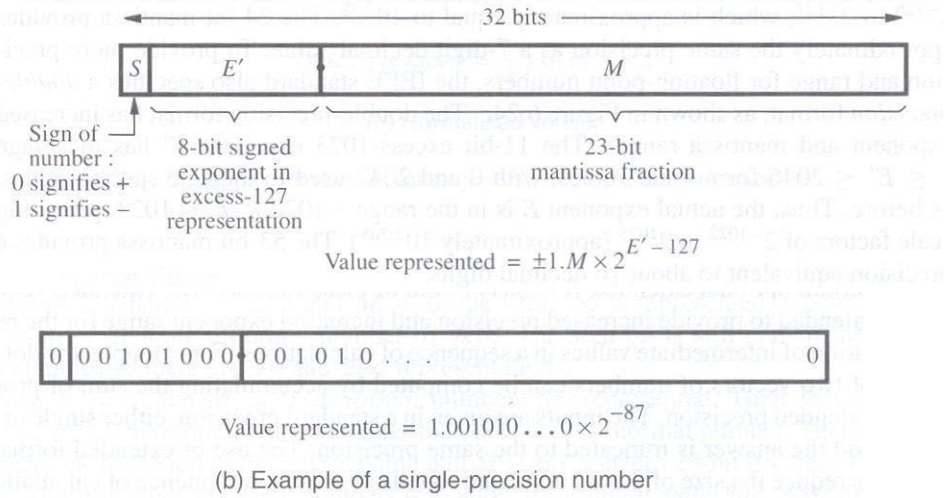
Underflow-In SP it means that the normalized representation requires an exponent less than -126.

Overflow- In SP it means that the normalized representation requires an exponent greater than +127.

14. Draw the format of floating point number. (R)(NOV/DEC 2012)

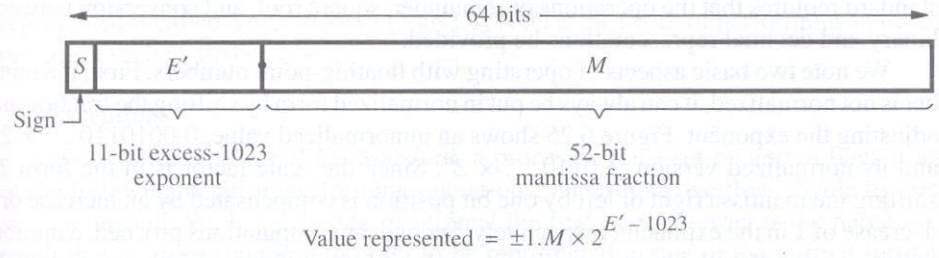
$$1.2345 = \underbrace{12345}_{\text{mantissa}} \times \overbrace{10^{-4}}^{\text{exponent}}$$

15. Draw the IEEE Standard Single Precision Floating-Point Format (R)



(b) Example of a single-precision number

16. Draw the IEEE Standard Double Precision Floating-Point Format (R)



17. Define Little Endian Arrangement(R)(Dec-2014)

Little-endian systems, store the least significant byte in the smallest address.

18. What is DMA (R)(Dec-2014)

Direct memory access (DMA) is a feature of computerized systems that allows certain hardware subsystems to access main system memory independently of the central processing unit (CPU).

19. How overflow occur in subtraction? (U)(APR/MAY 2015)

Overflow in two's complement may occur, not when a bit is carried out of the left column, but when one is carried into it and no matching carry out occurs. That is, overflow happens when there is a carry into the sign bit but no carry out of the sign bit.

20. What is meant by sub word parallelism? (R)(APR/MAY 2015)

It's another name for SIMD-Within-A-Register (SWAR), or register-sized vector operations. The idea is that if you have registers which can hold machine words of multiple times your data typesize, we can pack several data elements into them, and make single instructions affect all of those simultaneously. A 128-bit register, for instance, can hold two 64-bit floating point values; as long as your 'multiply' instruction is aware that the register is split in the middle, you can get 2 multiplications out of 1 operation.

21. What are the overflow/underflow conditions for addition and subtraction (U)(NOV/DEC2015)

If you have a carry out of 1 on addition with unsigned numbers, you have overflowed, and if you have a carry out of 0 with subtraction, you have underflowed. Does this work in every case, though? If you do 5-0: 0101 -0000

= 0101 +(1111 + 1)

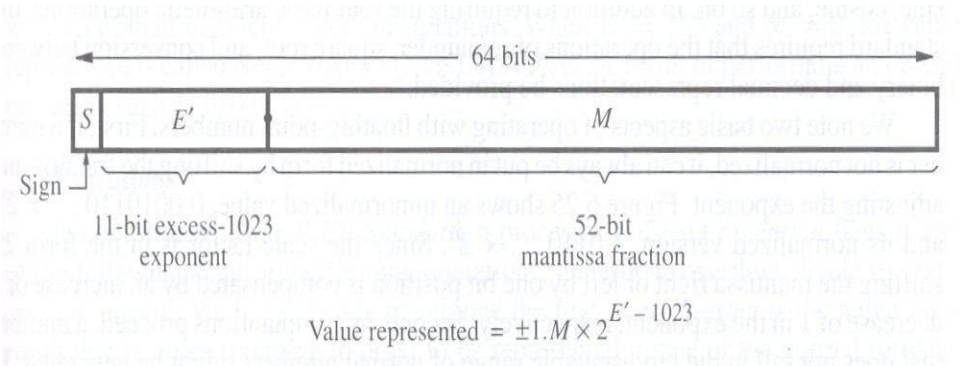
= 0101 +0000

= 0101... there is a carry out of zero here, instead of 1

22. State the representation of double precision floating point number(R) (NOV/DEC 2015)

The IEEE 754 standard specifies a **binary64** as having:

- Sign bit: 1 bit
- Exponent width: 11 bits
- Significantprecision: 53 bits (52 explicitly stored)



23. Define ALU?(R)(MAY/JUNE2016)

An arithmetic logic unit (ALU) is a digital circuit used to perform arithmetic and logic operations. It represents the fundamental building block of the central processing unit (CPU) of a computer. Modern CPUs contain very powerful and complex ALUs. In addition to ALUs, modern CPUs contain a control unit (CU).

25.What is Sub word Parallelism? (R)(MAY/JUNE2016)

Sub word Parallelism is a technique that enables the full use of word-oriented data paths when dealing with lower-precision data. It is a form of low-cost, small-scale SIMD parallelism

24. What is a guard bit and what are the ways to truncate the guard bits?(U) (NOV/DEC 2016)

- **Guard digits can be used to reduce the amount of round off error.**

Take 2 numbers:

$2.56 * 10^0$ and $2.34 * 10^2$

we bring the first number to the same power of 10 as the second one:

$0.0256 * 10^2$

The addition of the 2 numbers is:

```
0.0256*10^2
2.3400*10^2 +

_________
2.3656*10^2
```

After padding the second number (i.e., $2.34 * 10^2$) with two 0s, the bit after 4 is the guard digit, and the bit after is the round digit. The result after rounding is 2.37 as opposed to 2.36, without the extra bits (guard and round bits), i.e., by considering only $0.02 + 2.34 = 2.36$. The error therefore is 0.01.

26. Subtract (110110)2 – (10110)2 using 2's complement. (Ap)(Nov/Dec 2017)

The numbers of bits in the subtrahend is 5 while that of minuend is 6. We make the number of bits in the subtrahend equal to that of minuend by taking a `0' in the sixth place of the subtrahend. Now, 2's complement of 010110 is (101101 + 1) i.e.101010. Adding this with the minuend.

	1	1 0 1 1 0	Minuend
	1	0 1 0 1 0	2's complement of subtrahend
Carry over 1	1	0 0 0 0 0	Result of addition

After dropping the carry over we get the result of subtraction to be 100000.

27. Divide (1001010)2 / (1000)2 (Ap) (Nov/Dec 2017)

Ans: 1001

28. Show the IEEE 754 binary representation of the number (-0.75)10 in single precision. (Ap)(Apr/May 2018)

1 - 0111 1110 - 100 0000 0000 0000 0000 0000

29. Define a datapath in a CPU. (R) (Apr/May 2018)

A data path (also written as datapath) is a set of functional units that carry out data processing operations. Datapaths, along with a control unit, make up the central processing unit (CPU) of a computer system.

30. Perform X-Y using 2's complement arithmetic for the given two 16-bit binary numbers

X = 0000 1011 1110 1111 and Y= 1111 0010 1001 1101. **(Nov/Dec 2018)-Ap**

-1110011010101110

31. Define sub-word parallelism. (Nov/Dec 2018)-U

A subword is a lower precision unit of data contained within a word. In subword parallelism, multiple subwords are packed into a word and then process whole words. SUB WORD PARALLELISM. A subword is a lower precision unit of data contained within a word.

32. Convert $(1.00101)_2$ to decimal (Ap)(April/May 2019)

1.15625

33. Perform subtraction by two's complement method : 100 -110000.(Ap)(Apr/May 2019)

In a 2's complement subtraction, negative number is represented in the 2's complement form and actual addition is performed to get the desired result.

For Eg: A – B

a. Take 2's Complement of B
b. Result = A+ 2's Complement of B
c. If carry exist then the result is positive and in the true form. In this case, carry is ignored.
d. If carry does not exist then the result is negative and in the 2's complement form.

Here A = 000100, B = 110000.

Find A - B = complement of B = 110000

1's complement?using 2's complement

First find 2's

```
  1  1  1  1  1  1
- 1  1  0  0  0  0
-----------------
  0  0  1  1  1  1
```

Now add 1 : 001111 + 1 = 010000

Now Add this 2's complement of B to A

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

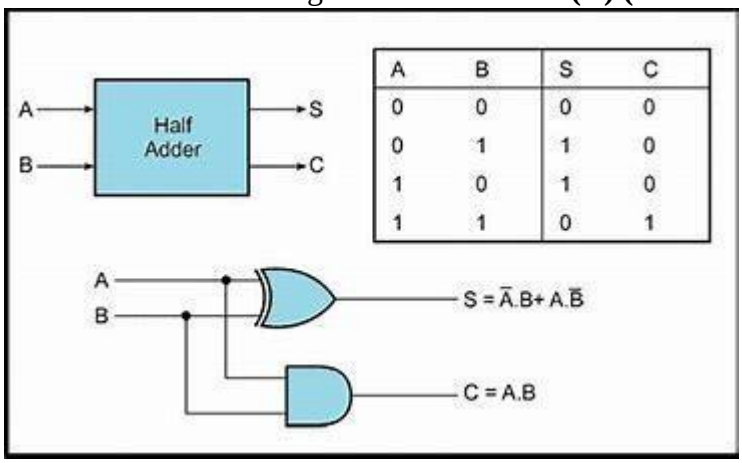
1's complement of 010100 is

```
  1  1  1  1  1  1
- 0  1  0  1  0  0
```

1 0 1 0 1 1

Now add 1 : 101011 + 1 = 101100
So answer is -101100

34. Draw the circuit diagram for half adder. **(U) (Nov/Dec2020 / April/May2021)**



35. Show that the logic expression Cn ⊕ Cn – 1 is a correct indicator of overflow in the addition of 2's complement integers, by using an appropriate truth table. **(An) (Nov/Dec2020 / April/May2021)**

Overflow – Overflow occurs in signed numbers having same signs, and sign of the result is different, and also it is shown that carry bits C_n and C_{n-1} are different. A circuit is added to detect overflow, eg. $C_{n-1} \oplus C_n$

In order to perform the subtract operation X-Y on 2's complement numbers X and Y, we form the 2s-complement of Y and add it to X. The logic circuit network shown in figure (5) can be used to perform either addition or subtraction based on the value applied to the Add/Sub input control line. This line is set to 0 for addition, applying the Y vector un changed to one of the adder inputs along with a carry-in signal,C0 of 0 . When Add/Sub control line is set to 1, the Y vector is 1's complemented by the XOR gates and C0 is set to 1 to complete the 2's complementation of Y. Remember that 2's complementing a negative number is done exactly same manner as for positive number. An XOR gate can be added to Figure(5) to detect the overflow condition $C_{n-1} \oplus C_n$

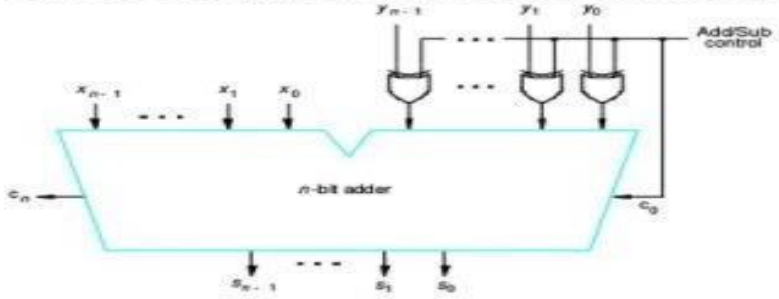


Figure 6.3. Binary addition-subtraction logic netw ork.

PART-B

1. Give the algorithm for multiplication of signed 2's complement numbers and illustrate with an example. **(U)(APRIL/MAY 2008)**

2. Write the algorithm for division of floating point numbers and illustrate with an example. **(U)(APRIL/MAY 2008)**

3. Write about the CSA method of fast multiplication. Prove how it is faster with an example. **(An) (NOV/DEC 2007) & (MAY/JUNE 2006)**

4. Draw the circuit for integer division and explain. **(R)(NOV/DEC 2007)**

5. Perform the division on the following 5-bit unsigned integer using non-restoring division: 10101 / 00101**(Ap)(MAY/JUNE 2006)**

6. Explain the working of a floating point adder/ subtractor. With a detailed flow chart explain how floating point additional/ subtraction is performed.**(C)**(**MAY/JUNE 2006) (NOV/DEC 2006) (APRIL/MAY 2009)(May/June 2012)**

7. Multiply the following pair of signed 2's complements numbers using bit-pair-recoding of the multipliers: A= 010111, B=101100. **(Ap)(MAY/JUNE 2006)**

8. Give the IEEE standard double precision floating point format.**(U) (NOV/DEC 2006)**

9. Explain the represent.tation of floating point numbers in detail. **(U)(MAY/JUNE 2007)**

10. Design a multiplier that multiplies two 4-bit numbers.**(U) (MAY/JUNE 2007)**

11. Explain the algorithm for integer division with suitable example.**(Ap) (MAY/JUNE 2007**)

12. Give the block diagram of the hardware implementation of addition and subtraction of signed number and explain the operations with flowchart.**(An) (MAY/JUNE 2007)**

13. Multiply the following pair of signed nos using Booth's bit-pair recoding of the multiplier. A=+13 (Multiplicand) and B= -6 (Multiplier)**(Ap)** (10**)(Dec-2014)**

14. Briefly explain carry look ahead adder**(U)(Dec-2014)**

15. Divide $(12)_{10}$ by $(3)_{10}$ using the Restoring and Non restoring division algorithm with step by step intermediate results and explain**(Ap)(16)(Dec-2014)**

16. Explain the sequential version of multiplication algorithm and its hardware**.(U) (APR /MAY 2015) (16)**

17. Explain how floating point addition is carried out in a computer system. Give an example for a binary floating point addition.**(U) (APR /MAY 2015) (16)**

18. Explain in detail about multiplication algorithm with suitable example and diagram**(An)(16)(NOV/DEC 2015)**

19. Explain in detail about division algorithmwith suitable example and diagram **(An)(16)(NOV/DEC 2015)**

20. Explain Booth's Algorithm for the multiplication of signed two's complement numbers**(An)(16)(NOV/DEC 2016)**

21. Discuss in detail about division algorithm in detail with diagram and examples **(An)** (16) **(NOV/DEC 2016) (NOV/DEC 2017)**

22. Explain briefly about floating point addition and Subtraction algorithms **(An)((MAY/JUNE2016)**

22. Define Booth Multiplication algorithm with suitable example **(An)(MAY/JUNE2016)**

23. (i)Add the numbers (0.5)10 and (0.4375)10 using the floating point addition.

(ii)Multiply the numbers (0.5)10 and (0.4375)10 using the floating point multiplication. **(Ap)(NOV/DEC 2017)**

24. (i) Perform X+Y and Y-X using 2's complement for given the two binary numbers X=0000 1011 1110 1111 and Y=1111 0010 1001 1101. (5)

(ii) Multiply the following signed 2's compliment numbers using the Booth algorithm. A=001110 and B=111001 where A is multiplicand and B is multiplier. (8) **(Ap)(Apr/May 2018)**

25. (i) Draw the block diagram of integer divider and explain the division algorithm. (5)

(ii) Add the numbers (0.75)10 and (-1.275)10 in binary using the Floating point addition algorithm. (8) **(Ap)(Apr/May 2018)**

26. Multiply the following signed numbers using Booth algorithm A=(-34)10=(1011110)2 and B =(22)10 =(0010110)2 where B is a multiplicand and A is multiplier.(6) **(Nov/Dec 2018)-Ap**
27.Draw the block diagram of integer divider and explain the division algorithm.(7) **(Nov/Dec 2018)-C**
28. How IEEE 752 32-bit single precision floating point numbers represented? Example. How are print numbers represented?(3) **(Nov/Dec 2018)-U**
29. Explain floating point addition algorithm with a neat block diagram?(10) (Nov/Dec 2018)-U
30. Calculate the following problems using BOOTH'S ALGORITHM (13)
(Apr/May 2019)(Ap)
(i) (+13) x (-6)
(ii) (+13) x(+6)
(iii) (-13) x (-6)
(iv) (-13) x (+6)
31. Calculate 10011 (-13) x 01011 (+11) using Signed-Operand Multiplication (13)
(Ap) (Apr/May 2019)
32. Describe in detail booth's multiplication algorithm and perform the booth's operation for the 5-bit signed operand, +23 is the multiplicand, and its multiplied by –10, the multiplier to get the 10-bit product –230. Similarly find the remaining three combinations numbers. (+23 × + 10, –23 × – 10, –23 × –10).(E) **(Nov/Dec2020 / April/May2021)**
33. Write a non-restoring and restoring algorithm then perform the number 8/3 integer division using non-restoring division.(U) **(Nov/Dec2020 / April/May2021)**

COURSE OUTCOME:
Students can able to design arithmetic and logic unit.

UNIT III

PROCESSOR AND CONTROL UNIT

COURSE OBJECTIVE: To learn the basics of pipelined execution.

SYLLABUS:
A BasicMIPSimplementation–Buildinga datapath–ControlImplementationscheme–Pipelining–Pipelined datapathand control– HandlingData hazards&Control hazards–Exceptions.

PART-A

1.Define pipelining. Mention its advantages.(R) (Apr/May 2010)
Pipelining is an effective way of organizing concurrent activity in a computer system. The processor executes the program by fetching and executing instructions, one after the other. The main advantage of pipeline is that it saves time in the execution ofinstruction because in pipeline overlapping of instruction takes place.

2. List the key aspects in gaining the performance in pipelined systems.(U) (Apr/May 2010)
One key aspect of pipeline design is balancing pipeline stages. Another design consideration is the provision of adequate buffering between the pipeline stages — especially when the processing times are irregular, or when data items may be created or destroyed along the pipeline.

3. Name the four steps in pipelining.(R)

Fetch: read the instruction from the memory.
Decode: decode the instruction and fetch the source operand.
Execute: perform the operation specified by the instruction.
Write: store the result in the destination location.

4. What are the disadvantages of increasing the number of stages in pipelined processing? (U)(Apr/May 2011)

It is not possible to feed a large number of pipelines with data. The memory system is just not powerful enough. Even with the existing pipelines, a fairly large number of clock ticks are wasted. The processor core is simply not utilised efficiently enough, because data cannot be brought to it quickly enough.

Another problem of having several pipelines arises when the processor can decode several instructions in parallel – each in its own pipeline. It is impossible to avoid the wrong instruction occasionally being read in (out of sequence).

5. What is the use of cache memory?(U)

The use of cache memories solves the memory access problem. In particular, when a cache is included on the same chip as the processor, access time to the cache is usually the same as the time needed to perform other basic operations inside the processor. This makes it possible to divide instruction fetching and processing into steps that are more or less equal in duration. Each of these steps is performed by a different pipeline stages, and the clock period is chosen to correspond to the longest one.

6. What is data hazard?(R)(NOV/DEC 2007)

Any condition that causes the pipeline to stall is called a hazard. 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 the pipeline. As a result some operation has to be delayed, and the pipeline stalls.

7. What are instruction hazards?(R) (May/June 2012)

The pipeline may also be stalled because of a delay in the availability of an instruction. For example, this may be a result of a miss in the cache, requiring the instruction to e fetched from the main memory. Such hazards are often called control hazards or instruction hazards.

8. What are called stalls?(R)

An alternative representation of the operation of a pipeline in the case of a cache miss gives the function performed by each pipeline stage in each clock cycle. The periods in which the decode unit, execute unit, and the write unit are idle are called stalls. They are also referred to as bubbles in the pipeline.

9. What is structural hazard?(R)

Structural hazard 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 in access to memory.

10. What is said to be side effect? (U)

When a location other than one explicitly named in an instruction as a destination operand is affected, the instruction is said to have a side effect.

11. What is dispatch unit?(R)
A separate unit which we call the dispatch unit takes instructions from the front of the queue and sends them to the execution unit. The dispatch unit also performs the decoding function.

12. What is branch folding?(U)
The instruction fetch unit has executed the branch instruction concurrently with the execution of other instructions. This technique is referred to as branch folding.

13. What is delayed branching?(U)
A technique called delayed branching can minimize the penalty incurred as a result of conditional branch instructions. The idea is simple. The instructions in the delay slots are always fetched. Therefore, we would like to arrange for them to be fully executed whether or not the branch is taken. The objective is to be able to place useful instructions in these slots. If no useful instructions can be placed in the delay slots, these slots must be filled with NOP instructions.

14. Define speculative execution.(U) (May/June 2012)(Dec-2014)
Speculative execution means that instructions are executed before the processor is certain that they are in the correct execution sequence. Hence, care must be taken that no processor registers or memory locations are updated until it is confirmed that these instructions should indeed be executed. If the branch decision indicates otherwise, the instructions and all their associated data in the execution units must be purged, and the correct instructions fetched and executed.

15. What is called static and dynamic branch prediction?(U)
The branch prediction decision is always the same every time a given instruction is executed. Any approach that has this characteristic is called static branch prediction. Another approach in which the prediction decision may change depending on execution history is called dynamic branch prediction.

16. What are condition codes?(U)
In many processors, the condition code flags are stored in the processor status register. They are either set or cleared by many instructions, so that they can be tested by subsequent conditional branch instructions to change the flow of program execution.

17. What are superscalar processors?(U)(NOV/DEC 2006)
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.

18. What is imprecise and precise exception?(U)
Situation in which one or more of the succeeding instructions have been executed to completion is called imprecise exception. Situation in which all subsequent instructions that may have been partially executed are discarded. This is called a precise exception.

19. Why are interrupt masks provided in any processor? (U)(APRIL/ MAY 2009)
A technique of suppressing certain interrupts and allowing the control program to handle these masked interrupts at a later time.

20. What is a deadlock? (U)

A deadlock is a situation that can arise when two units, A and B, use a shared resource. Suppose that unit B cannot complete its task until unit A completes its task. At the same time, unit B has been assigned a resource that unit A needs. If this happens, neither unit can complete its task. Unit A is waiting for the resource it needs, which is being held by unit b. at the same time, unit B is waiting for unit A to finish before it can release that resource.

21. What is meant by hazard in pipelining? Define data and control hazards.(R)(MAY/JUNE 2013)

Eliminating a hazard often requires that some instructions in the pipeline to be allowed to proceed while others are delayed. When the instruction is stalled, all the instructions issued later than the stalled instruction are also stalled. Instructions issued earlier than the stalled instruction must continue, since otherwise the hazard will never clear.
Data hazards: They arise when an instruction depends on the result of a previous instruction in a way that is exposed by the overlapping of instructions in the pipeline. Control hazards: They arise from the pipelining of branches and other instructions that change the PC.

22. Why is branch prediction algorithm needed? Differentiate between the static and dynamic techniques.(R)(MAY/JUNE 2013)

- Increases the number of instructions available for the scheduler to issue Increases instruction level parallelism (ILP)
- Allows useful work to be completed while waiting for the branch to resolve

Static:

– Decided before runtime
– Examples:
– Always-Not Taken
– Always-Taken
– Backwards Taken, Forward Not Taken (BTFNT)
– Profile-driven prediction

Dynamic:

– Prediction decisions may change during the execution of the program

23. What is a hazard? (R)(NOV/DEC 2012)

In CPU design, Hazards are problems with the instruction pipeline in central processing unit (CPU) microarchitectures when the next instruction cannot execute in the following clock cycle and can potentially lead to incorrect computation results. There are typically three types of hazards:

- data hazards
- structural hazards
- control hazards (branching hazards)

24. Define exception. (R) (NOV/DEC 2012)(Dec-2014)

Exceptions and interrupts are unexpected events that disrupt the normal flow of instruction execution. An exception is an unexpected event from within the processor. An interrupt is an unexpected event from outside the processor.

25. Define Data path(R)
The registers, the ALU, and the interconnecting bus are collectively referred to as the data path.

26. What do we need to add/modify in our MIPS datapath to implement pipelining?(U)
In pipeline execution, all units are operating in every cycle; we have to duplicate hardwarewhere needed State registers are added between stages to preserve intermediate data and control for each instruction.

27. Define microroutine and microinstruction. (R) (May/June 2013)
A sequence of control words corresponding to the control sequence of a machine instruction constitutes the microroutine for that instruction, and the individual control words in this microroutine are referred to as microinstructions.

28. What is Nano programming? (R) (Nov / Dec 2013)
An alternative strategy to generate control signals
- Having the concept of a secondary control memory
- A microinstruction is in primary control-store memory; it then has the control signals generated for each microinstruction using a secondary control store memory
- The output word from the secondary memory is called nanoinstruction

29. What are R-Type instructions? (R)(APRIL/MAY 2015)
R-type instructions refer to register type instructions. Of the three formats, the R-type is the most complex.
This is the format of the R-type instruction, when it is encoded in machine code.

B_{31-26}	B_{25-21}	B_{20-16}	B_{15-11}	B_{10-6}	B_{5-0}
opcode	**register s**	**register t**	**register d**	**shift amount**	**function**

The prototypical R-type instruction is:
add $rd, $rs, $rt

31. What is hazard? What are its types? (R) (NOV/DEC 2015)
Hazards are problems with the instruction pipeline in CPU microarchitectures when the next instruction cannot execute in the following clock cycle,[1] and can potentially lead to incorrect computation results. Three common types of hazards are data hazards, structural hazards, and control hazards (branching hazards).

32. What is meant by branch predication?(U)(APRIL/MAY 2015) &(NOV/DEC 2015)
Abranch predictor is a digital circuit that tries to guess which way a branch (e.g. an if-then-else structure) will go before this is known for sure. The purpose of the branch predictor is to improve the flow in the instruction pipeline.

33. What is meant by pipeline bubble?(U) (NOV/DEC 2016)
In computing, a bubble or pipeline stall is a delay in execution of an instruction in an instruction pipeline in order to resolve a hazard. The values are preserved until the bubble has passed through the execution stage.

34. What is a data path? (R) (NOV/DEC 2016)

A data path is a collection of functional units (such as arithmetic logic units or multipliers that perform data processing operations), registers, and buses. Along with the control unit it composes the central processing unit (CPU).

35. What are the advantages of pipelining?(U)(MAY /JUNE2016)

The cycle time of the processor is reduced; increasing the instruction throughput. If pipelining is used, the CPU Arithmetic logic unit can be designed faster, but more complex.

36. What is Exception? (R)(MAY /JUNE2016)

Exceptions and interrupts are unexpected events that disrupt the normal flow of instruction execution. An exception is an unexpected event from within the processor. An interrupt is an unexpected event from outside the processor. You are to implement exception and interrupt handling in your multicycle CPU design.

37. Mention the various types of pipelining. (R)(Nov/Dec 2017)

Types of Pipeline

It is divided into 2 categories:

1. Arithmetic Pipeline
2. Instruction Pipeline

38. Mention the various phase in executing an instruction. (R) (Nov/Dec 2017)

1. The Instruction Cycle. Instructions are processed under direction of the control unit in step-by-step manner. ...
2. Fetch Instruction Phase. Obtain next instructionfrom memory. ...
3. Decode Instruction Phase. ...
4. Evaluate Operand Address Phase. ...
5. Fetch Operands Phase. ...
6. Steps in a Typical Read Cycle. ...
7. Execute phase. ...
8. Store Result Phase.

39. What is the ideal CPI of a pipelined processor? (U)(Apr/May 2018)

Pipeline stalls are introduced in the pipelines architecture, so CPI will increase
Stalls per instruction =0.2×1=0.2×1 cycle +0.05×2+0.05×2 cycle = 0.3
New CPI = old CPI + stalls per cycle = 1.4 + 0.3 = 1.7
so, Pipelined architecture will have CPI of 1.7

40. What is meant by exception? Give one example of MIPS exception. (R)(Apr/May 2018)

The MIPS convention calls an exception any unexpected change in control flow regardless of its source (i.e. without distinguishing between a within the processor source and an external source). ... Arithmetic overflows, undefined instructions, page faults are some examples of synchronous exceptions.

41. Write the two steps that are common to implement any type of instruction (Nov/Dec 2018)-R

1. fetch **instruction** (aka pre-fetch)

2. decode **instruction**

42. What is an exception? Give one example for MIPS exception(Nov/Dec 2018)-R

An exception is said to be synchronous if it occurs at the same place every time a program is executed with the same data and the same memory allocation. Arithmetic overflows, undefined instructions, page faults are some examples of synchronous exceptions. Asynchronous exceptions, on the other hand, happen with no temporal relation to the program being executed. I/O requests, memory errors, power supply failure are examples of asynchronous events.

Example :the status register is referred to as $12.

43. Convert the following code segment in C.to MIPS instructions, assuming all variables are in memory and are addressable as offsets from $t0:(Ap) (Apr/May 2019)

a=b+e; c=b+f

Here is the generated MIPS code for this segment, assuming all variables are in memory and are addressable as offsets from $t0:

```
lw $t1, 0($t0)# b is stored at 0 offset from $t0
lw $t2, 4($t0)# Word size is 4 bytes in MIPS
add $t3, $t1, $t2
sw $t3, 12($t0)
lw $t4, 8($t0)
add $t5, $t1, $t4
sw $t5, 16($t0)
```

44. Write down the five stages of instruction executions (R) (Apr/May 2019)

Stage 1 (Instruction Fetch):
In this stage the CPU reads instructions from the address in the memory whose value is present in the program counter.

Stage 2 (Instruction Decode):
In this stage, instruction is decoded and the register file is accessed to get the values from the registers used in the instruction.

Stage 3 (Instruction Execute):
In this stage, ALU operations are performed.

Stage 4 (Memory Access):
In this stage, memory operands are read and written from/to the memory that is present in the instruction.

Stage 5 (Write Back):
In this stage, computed/fetched value is written back to the register present in the instructions.

45. What are the steps required for a pipelined processor to process the instruction ? **(U) (Nov/Dec2020 / April/May2021)**

In a pipelined computer, instructions flow through the central processing unit (CPU) in stages. For example, it might have one stage for each step of the von Neumann cycle: Fetch the instruction, fetch the operands, do the instruction, write the results

46. What is locality of reference ? **(U) (Nov/Dec2020 / April/May2021)**
Locality of reference, also known as the principle of locality, is a term for the phenomenon in which the same values, or related storage locations, are frequently accessed, depending on the memory access pattern.

PART-B

1. Define pipeline. Explain its performance with an example **(APRIL/MAY 2009) (May/June 2012) (R&U)**

2. Design a 4-stage instruction pipeline and show how its performance is improved over sequential execution. **(NOV/DEC 2007) & (NOV/DEC 2006) (C)**

3. Explain the function of a six segment pipelines and draw a space diagram for a six segment pipeline showing the time it takes to process eight tasks. **(MAY/JUNE 2007)(Ap)**

4. Explain the performance of the instruction pipeline can be improved. **(MAY/JUNE 2007)(U)**

5. Explain about Data Hazards with its representation**.(APRIL/MAY 2008) (May/June 2012)(Dec-2014)(U)**

6. Explain the execution of an instruction with diagram. **(MAY/JUNE 2007)(U)**

7. Explain about instruction Hazards. **(May/June 2012)(U)**

8. Highlight the solutions of instruction Hazards. What are branch hazards? Describe the methods for dealing with the branch hazards. **(NOV/DEC 2007)(U)**

9. Explain how pipelining helps to speed-up the processor. Discuss the hazards that have to be taken care of in a pipelined processor**. (MAY/JUNE 2006) (U)**

10. Explain about Data path & Control Consideration. **(May/June 2012)(Nov / Dec 2013)(Dec-2014)(U)**

11. What are the hazards of conditional branches in pipelines? How it can be resolved? **(Apr/May 2011)(Ap)**

12. Describe the role of cache memory in pipelined system.**(Apr/May 2010)(U)**

13. Discuss the influence of pipelining on instruction set design. **(Apr/May 2010)(U)**

14. Describe the procedure to fetch a word from the memory and store a word into the memory **(APRIL/MAY 2009)(An)**

15. Discuss the role of cache in pipelining and Discuss about the pipeline performance considerations. **(NOV/DEC 2012)(U)**

16. What is data hazard? Explain the ways and means of handling it. **(NOV/DEC 2012)(U)**

17. Explain a 4-stage instruction pipeline. Also explain the issues affecting instruction pipeline. Explain dynamic branch prediction technique. **(MAY/JUNE 2013)(U)**

18. Explain the relation between pipelined execution and instruction feature and Describe the techniques for handling control hazards in pipelining **(MAY/JUNE 2013)(An)**

19. Explain about multiple-bus organization in detail. **(MAY/JUNE 2007&may/June 2012) (May/June 2013) (Nov / Dec 2013)(U)**

20. Explain the organization of Micro programmed control unit in detail. **(NOV/DEC 2007) (& (MAY/JUNE 2006) & (NOV/DEC 2006) (Nov/Dec 2012) (May/June 2013)(U)**

21. Explain the following:
(i)Address Sequencing in control memory
(ii)Micro program sequencer **(Nov / Dec 2013)(U)**
22. Explain the design of hardwired control unit? **(Nov / Dec 2013)(U)**
23. What is instruction hazard? Explain in detail how to handle the instruction hazards in pipelining with relevant examples? **(Nov / Dec 2013)(An)**
24. Write short notes on exception handling?**(Nov / Dec 2013)(U)**
25. Explain different types of pipeline hazards with suitable examples.**(Apr/May 2015) (16) (U)**
26. Explain in detail how exceptions are handled in MIPS architecture.**(Apr/May 2015) (16)(U)**
27. Explain the basic MIPS implementation with necessary multiplexers and control lines **(16)(NOV/DEC 2015)(U)**
28. Explain how the instruction line work? What are the various situations where the instruction pipeline can stall? Illustrate with an example? **(16)(NOV/DEC 2015) (U)**
29. Why is branch prediction algorithm needed? Differentiate between the static and dynamic techniques **(16) (NOV/DEC 2016) (An)**
30. Explain how the instruction pipeline works. What are the various situations where an instruction pipeline can stall? **(16) (NOV/DEC 2016) (An)**
31. What is pipelining? Discuss about pipelined data path and control **(16) (MAY/JUNE 2016) (U)**
32. Briefly explain about various categories of hazards with examples **(16) (MAY/JUNE 2016) (U)**
33. Explain in detail the operations of the datapath.(U)**(Nov/Dec 2017)**
34. Explain the pipeline hazard in detail. (U)**(Nov/Dec 2017)**
35. Design a simple datapath with the control unit and explain in detail.(U)**(13) (Apr/May 2018)**
36. Discuss the limitations of pipelining a processor's datapath. Suggest the methods to overcome them. **(13) (Apr/May 2018)**

37. Draw a simple MIPS datapath with the control unit and explain the execution of ALU

instructions(13) **(Nov/Dec-2018)-C**

38. A processor has five individual stages, namely, IF, ID EX, MEM, and WB and their latencies are 250ps, 350ps, 150ps , 300ps, and 200ps respectively. The frequency of the instructions executed by the processor are as follows , ALU:40%, Branch: 25%, load: 20% and store:15% What is the clock cycle time in a pipelined and non pipelined processor? If you can split one stage of the pipelined datapath into two new stages, each with half the latency of the original stage, which stage would you split and what is the new clock cycle time of the processor? Assuming there are no stalls or hazards, What is the utilization of the data memory? Assuming there are no stalls or hazards, what is the utilization of the write –register port of the "Registers" unit?(6) **(Nov/Dec-2018)-An**

39. List the hazards in pipelining a processor and give one example for each.(7)

(Nov/Dec-2018)-R 40.
Explain the basic MIPS implementation with necessary multiplexers and control lines. **(13)(U)(Apr/May 2019)**

41. Explain the basic MIPS implementation with necessary multiplexers and control lines. **(U)(13)(Apr/May 2019)**

42. Explain how the instruction pipeline works? What are the various situation where an instruction pipeline stalls? Illustrate with an example. **(U)(13) (Apr/May 2019)**

43. Describe in detail pipelined implementation of data path and control with diagrams. **(U) (Nov/Dec2020 / April/May2021)**

44. Find out the hazards in the following instructions and eliminate them by using stalls : LW R1 , 0(R2) SUB R4 , R1 , R5 AND R6 , R1 , R7 OR R8 , R1 , R9.(**U) (Nov/Dec2020 / April/May2021)**

COURSE OUTCOME:
Students can able to Understand pipelined execution and design control unit.

UNIT IV
PARALLELISM

COURSE OBJECTIVE: To understand parallelism and multi-core processors.

SYLLABUS:
Instruction-level-parallelism–Parallel processingchallenges–Flynn'sclassification–Hardware multithreading–Multicoreprocessors

PART-A

1. What is ILP?(U)
Pipelining is used to overlap the execution of instructions and improve performance. This potential overlap among instructions is called instruction level parallelism (ILP) since the instruction can be evaluated in parallel.

2. What are approaches used to exploit ILP?(U)
- Dynamic approach
- Static Approach

3. What is an Ideal Pipeline?(U)
Pipeline CPI = Ideal pipeline CPI + Structural stalls + Data hazard stalls + Control stalls

4. What is multithreading?(U)(Dec-2014)
The process of executing the multiple thread by common memory or common processor in which the execution is done is overlapping fashion.

5. What is Hardware Multithreading (U)
- Hardware multithreading is one that allows multiple threads to share the functional units of a single processor in an overlapped manner.

- Processor resources are fully duplicated to support the thread execution. Ex: Register file, PC(program counter) etc.
- Memory can be shared through virtual memory management mechanism.
- Hardware supports fast thread switching.

6. What are the two main approaches in hardware multithreading (U)

- Fine Grain Multithreading:

Switch threads after each cycle. Interleaved thread execution using round robin fashion If one thread stalls other threads are executed.

- Coarse Grain Multithreading

Switch only on costly / long stalls (ex: L2 cache misses). Simplifies hardware but does not hide short stalls.(Ex: data hazards)

7. What is a Multi-Core processor? (U)

Multi-core is a design in which a single physical processor contains the core logic of more than one processor. The multi-core design puts several such processor "cores" and packages them as a single physical processor.

8. What is loop level analysis and loop carried dependence?(U)

- Loop level analysis involves determining what depends exist among the operands in a loop across the iterations of a loop are data dependent on data values produced in earlier iterations.
- Data dependence between different loop iterations (data produced in earlier iterations used in a later one) is called loop carried dependence.

9. What is data hazard? (R)(May/June 2013) (Nov / Dec 2013)

Any condition that causes the pipeline to stall is called a hazard. 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 the pipeline. As a result some operation has to be delayed, and the pipeline stalls.

10. What are instruction hazards? (R) (May/June 2013)

The pipeline may also be stalled because of a delay in the availability of an instruction. For example, this may be a result of a miss in the cache, requiring the instruction to e fetched from the main memory. Such hazards are often called control hazards or instruction hazards.

11. What is called static and dynamic branch prediction?(U)(May/June 2013)

The branch prediction decision is always the same every time a given instruction is executed. Any approach that has this characteristic is called static branch prediction. Another approach in which the prediction decision may change depending on execution history is called dynamic branch prediction.

12. What is Hazard? (R) (Nov/Dec 2012)(May/June 2013)

Any condition that causes the pipeline to stall is called a hazard.

13. Define Pipeline Speedup? (U)(Nov / Dec 2013)

Pipeline speedup**:**

The ideal speedupfrom a pipeline is equal to the number of stages in the pipeline.

$$\frac{\text{Time per instruction on unpipelined machine}}{\text{Number of pipe stages}}$$

15. What is Flynn's Classification(U) (Dec- 2014)

Flynn's taxonomy is a classification of computer architectures, The classification system has stuck, and has been used as a tool in design of modern processors and their functionalities. Since the rise of multiprocessing CPUs, a multiprogramming context has evolved as an extension of the classification system.

16. Difference between strong scaling and weak scaling. (U)(Apr/May 2015) (Nov/Dec 2017)

Strong scaling means, that the workload on a processors is different, the number of processors is increased (thus the total problem size as well).

Weak scaling means, that the workload on a processors is the same, the number of processors is increased (thus the total problem size as well).

17. Compare UMA and NUMA multiprocessors (U)(Apr/May 2015)

UMA (Uniform Memory Access)

Most x86 multiprocessor systems are UMA (Uniform Memory Access) machines. That is, all of the processors in the system send their requests to a single memory controller (generally in the Northbridge) which, in turn, retrieves the information for them. In these systems, all of the processors have equal access to all of the memory in the computer. Systems based on Intel's Xeon chips.AMD's older Athlon MPProcessors and Apple's G5 towers are examples of this type of architecture.

NUMA(Non -Uniform Memory Access)

AMD's Opteron chips, on the other hand, use a different type of architecture called NUMA (Non-Uniform Memory Access). In these systems, each physical processor has its own memory controller and, in turn, its own bank of memory. While NUMA has been around for a while, the Opteron is currently the only x86 implementation at this stage so it's not terribly common.

18. What is ILP?(U) (NOV/DEC 2015)

The Individual Learning Plan (ILP) is a personalized document that students develop with their counselors and other supportive adults to chart their progress toward their goals during each school year, to ensure they are on the path to graduation, and to determine what resources and tools they need to be prepared for a successful pathway in life after graduation.

19. Define a super scalar processor(U) (NOV/DEC 2015)

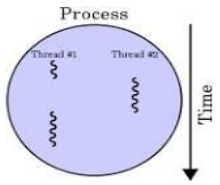
A **superscalar processor** is a **CPU** that implements a form of parallelism called instruction-level parallelism within a single **processor**. It therefore allows faster **CPU** throughput (the number of instructions that can be executed in a unit of time) than would otherwise be possible at a given clock rate.

20. What is instruction level parallelism? (U)(MAY/JUNE 2016) &(NOV/DEC 2016)

Instruction-level parallelism (ILP) is a measure of how many of the instructions in a computer program can be executed simultaneously. There are two approaches to instruction level parallelism: Hardware, Software.

21. What is multithreading? (U)(NOV/DEC 2016)

In computer architecture, multithreading is the ability of a central processing unit (CPU) or a single core in a multi-core processor to execute multiple processes or threads concurrently, appropriately supported by the operating system.



22. What is Fine grained Multithreading?(U) (MAY/JUNE 2016)

Interleaved multithreading: Interleaved issue of multiple instructions from different threads, also referred to as temporal multithreading. It can be further divided into fine-grained multithreading or coarse-grained multithreading depending on the frequency of interleaved issues.

23. Difference between Fine grained Multithreading and Coarse grained Multithreading (U)(Nov/Dec 2017)

1. Fine-grain multithreading issues instructions for different threads after every cycle.
2. Coarse-grain multithreading only switches to issue instructions from another thread when the current executing thread causes some long latency events (like page fault etc.)

24. Give example for each class in Flynn's classification. (U)(Apr/May 2018)

Dominant representative SISD systems are IBM PC, workstations.
Dominant representative SIMD systems is Cray's vector processing machine.

25. Web server is to be enhanced with a new CPU which is 10 times faster on computation than old CPU. The original CPU spent 40% of its time processing and 60% of its time waiting for I/O. What will be the overall speedup? **(Nov/Dec 2018)-Ap**

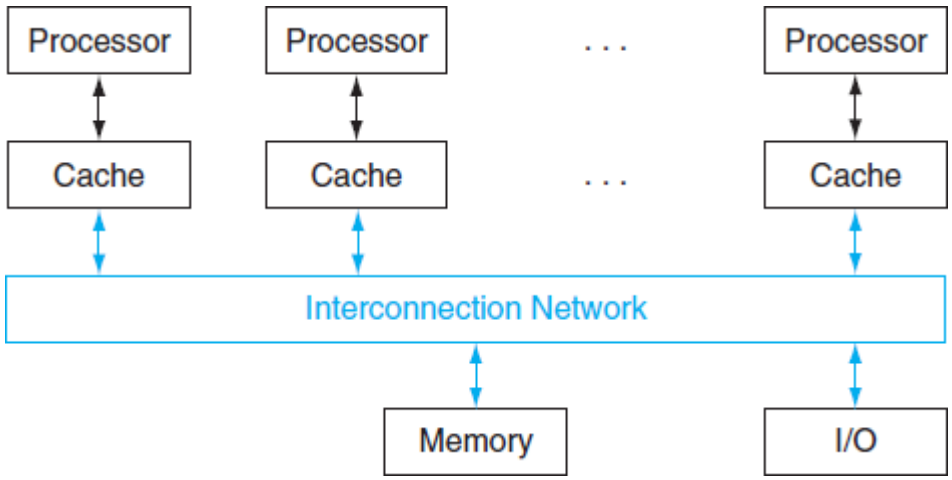
26. Classify shared memory multiprocessor based on the memory access latency. **(Nov/Dec 2018)-R**

Uniform Memory Access (UMA)
Non Uniform Memory Access (NUMA)
Cache Only Memory Access (COMA)

27. List the four multicore systems(R) (Apr/May 2019)

SISD or Single Instruction stream, Single Data stream.A uniprocessor.
MIMD or Multiple Instruction streams, Multiple Data streams. A multiprocessor.
SPMD Single Program, Multiple Data streams.
SIMD or Single Instruction stream, Multiple Data streams.

28. What is shared memory multiprocessor? (U)(Apr/May 2019)



29. What are multiprocessors ? Mention the categories of multiprocessors. **(Nov/Dec2020 / April/May2021)**

A multiprocessor is a computer system with two or more central processing units (CPUs), with each one sharing the common main memory as well as the peripherals. This helps in simultaneous processing of programs.

The key objective of using a multiprocessor is to boost the system's execution speed, with other objectives being fault tolerance and application matching.

Categories of multiprocessors:

- Symmetric Multiprocessor
- Asymmetric Multiprocessor
- Shared Memory Multiprocessor
- Distributed Memory Multiprocessor
- UMA Multiprocessor
- NUMA Multiprocessor

30. What is NUMA processor ? **(Nov/Dec2020 / April/May2021)**

NUMA (non-uniform memory access) is a method of configuring a cluster of microprocessor in a multiprocessing system so that they can share memory locally, improving performance and the ability of the system to be expanded. NUMA is used in a symmetric multiprocessing (SMP) system.

Non-uniform memory access (NUMA) is a computer memory design used in multiprocessing, where the memory access time depends on the memory location relative to the processor. NUMA architectures logically follow in scaling from symmetric multiprocessing (SMP) architectures.

PART-B

1. What are the challenges faced by ILP?**(R)(Apr/May 2018)**
2. What are the basic compiler techniques for exposing ILP? Explain them.**(U)**
3. What are the limitations of ILP?(R)**(Apr/May 2018)**
4. Explain Intel multi-core Architecture in detail.**(Dec-2014) (U)**
5. Discuss about heterogeneous multi-core processors in detail.**(U)**
6. Explain in detail about Flynn's classification**. (Nov/Dec 2017) (U)**

7. Design a 4-stage instruction pipeline and show how its performance is improved over sequential execution. **(NOV/DEC 2007) & (NOV/DEC 2006) (May/June 2013) (An)**
8. What are branch hazards? Describe the methods for dealing with the branch hazards. **(May/June 2013) (U)**
9. Explain in detail Multithreading **(Dec-2014)(U)**
10. Explain Instruction level Parallel Processing. State the challenges of parallel Processing**(Dec-2014) (U)**
11. Discuss about SISD,MIMD,SIMD,SPMD and VECTOR systems (**Apr/May 2015) (16) (An)**
12. What is hardware multithreading ?Compare and Contrast Fine grained Multi-Threading and coarse grained Multi-Threading **(Apr/May 2015) (16) (An)**
13. Explain in detail Flynn's classification of parallel hardware **(16) (NOV/DEC 2015)(U)**
14. Explain in detail about hardware multithreading **(16) (NOV/DEC 2015)(U)**
15. Explain in detail about Flynn's classification of parallel hardware **(16) (NOV/DEC 2016)(U)**
16. Discuss Shared memory multiprocessor with a neat diagram **(16) (NOV/DEC 2016) (U)**
17. Explain in detail about Flynn's classification **(MAY/JUNE 2016)(U)**
18. Write short notes on: **(MAY/JUNE 2016)**
(i) Hardware multithreading**(U)**
(ii) Multicore processors**(U)**
19. Describe simultaneous multithreading with an example. **(Nov/Dec 2017)**
20. (i) Compare and contrast fine grained multi-threading, coarse grained multithreading and simultaneous multithreading. (9)
(ii)Classify shared memory multiprocessor based on memory access latency. (4) **(Apr/May 2018)**
21. List the software and hardware techniques to achieve instruction Level Parallelism (ILP)(**4) (Nov/Dec 2018)-R**
22. Discuss the challenges in parallel processing in enhancing computer architecture**(9) (Nov/Dec 2018)-U**
23. Explain any three types of hardware multithreading(9) **(Nov/Dec 2018)-U**
24. Define the classes in Flynn's Taxonomy of computer architectures. Give one example for each class(**4) (Nov/Dec 2018)-R**
22. Explain in detail Flynn's classification of parallel hardware. **(13) (Apr/May 2019)(U)**
23. Discuss the principle of hardware multithreading and elaborate its types**.(13)(Apr/May 2019)(An)**
24. Discuss the principle of hardware multithreading and elaborate its types. **(Nov/Dec2020 / April/May2021)(U)**
25. Explain about the multicore processors. **(Nov/Dec2020 / April/May2021)(U)**

COURSE OUTCOME:
Students can able to Understand parallel processing architectures.

UNIT – V
MEMORY AND I/O SYSTEMS

COURSE OBJECTIVES:
- To understand the memory hierarchies, cache memories and virtual memories.
- To learn the different ways of communication with I/O devices.

SYLLABUS:

Memory hierarchy-Memory technologies–Cachememory–Measuringandimprovingcache performance-Virtual memory, TLBs– Accessing I/O devices – Interrupts – Direct Memory Access – Bus Structure – Bus operation – Arbitration – Interface Circuits – USB.

PART-A

1. What is the maximum size of the memory that can be used in a 16-bit computer and 32 bit computer?(An)

The maximum size of the memory that can be used in a 16-bit computer is 2^{16}=64K memory locations.

The maximum size of the memory that can be used in a 32-bit computer is 2^{32}=4G memory locations.

2. Define memory access time? (U)

The time required to access one word is called the memory access time. Or It is the time that elapses between the initiation of an operation and the completion of that operation.

3. Define memory cycle time?(U)

It is the minimum time delay required between the initiations of two successive memory operations. Eg.The time between two successive read operations.

4. When is a memory unit called as RAM?(U)

A memory unit is called as RAM if any location can be accessed for a read or writes operation in some fixed amount of time that is independent of the location's address.

5. What is MMU?(U)

MMU is the Memory Management Unit. It is a special memory control circuit used for implementing the mapping of the virtual address space onto the physical memory.

6. Define memory cell? (R)

A memory cell is capable of storing one bit of information. It is usually organized in the form of an array.

7. What is a word line? (R)

In a memory cell, all the cells of a row are connected to a common line called as word line.

8. Define static memories? (R)

A memory that consists of circuits capable of retaining their state as long as power is applied is called Static memories.

9. What are the Characteristics of semiconductor RAM memories? (U)

- They are available in a wide range of speeds.
- Their cycle time range from 100ns to less than 10ns.
- They replaced the expensive magnetic core memories.
- They are used for implementing memories.

10. Why SRAMs are said to be volatile?(U)

SRAMs are said to be volatile, because their contents are lost when power is interrupted.

11. What are the Characteristics of SRAMs?(R)

- SRAMs are fast.
- They are volatile.
- They are of high cost.
- Less density.

12. What are the Characteristics of DRAMs? (R)

- Low cost.
- High density.
- Refresh circuitry is needed.

13. Define Refresh Circuit? (R)

It is a circuit which ensures that the contents of a DRAM are maintained when each row of cells are accessed periodically.

14. Define Memory Latency?(U)

It is used to refer to the amount of time it takes to transfer a word of data to or from the memory.

15. What are asynchronous DRAMs?(U)

In asynchronous DRAMs, the timing of the memory device is controlled asynchronously. A 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 asynchronous DRAMs.

16. What are synchronous DRAMs?(U)

Synchronous DRAMs are those whose operation is directly synchronized with a clock signal.

17. Define Bandwidth?(U)

When transferring blocks of data, it is of interest to know how much time is needed to transfer an entire block. since blocks can be variable in size it is useful to define a performance measure in terms of number of bits or bytes that can be transferred in one second. This measure is often referred to as the memory bandwidth.

18. What is double data rate SDRAMs?(U)

Double data rates SDRAMs are those which can transfer data on both edges of the clock and their bandwidth is essentially doubled for long burst transfers.

19. What is mother board? (R)

Mother Board is a main system printed circuit board which contains the processor. It will occupy an unacceptably large amount of space on the board.

20. What are SIMMs and DIMMs?(U)

SIMMs are Single In-line Memory Modules. DIMMs are Dual In-line Memory Modules. Such modules are an assembly of several memory chips on a separate small board that plugs vertically into a single socket on the motherboard.

21. What is memory Controller?(U)

A memory controller is a circuit which is interposed between the processor and the dynamic memory. It is used for performing multiplexing of address bits. It provides RAS-CAS timing. It also sends R/W and CS signals to the memory. When used with DRAM chips, which do not have self-refreshingcapability, the memory controller has to provide all the information needed to control the refreshing process.

23. What is Ram Bus technology?(U)

Rambus Dynamic Random Access Memory (RDRAM) is a memory subsystem that promises to transfer up to 1.6 billion bytes per second. The subsystem consists of the random access memory, the RAM controller, and the bus (path) connecting RAM to the microprocessor and devices in the computer that use it.

22. Differentiate static RAM and dynamic RAM?(U)(APRIL/MAY 2008)(Nov / Dec 2013)(Apr/May 2018)

Static RAM	Dynamic RAM
They are fast	They are slow
They are very expensive	They are less expensive
They retain their state indefinitely	They do not retain their state indefinitely
They require several transistors	They require less no transistors.
Low density	High density

24. What are RDRAMs?(U)

RDRAMs are Ram bus DRAMs. Ram bus requires specially designed memory chips. These chips use cell arrays based on the standard DRAM technology. Multiple banks of cell arrays are used to access more than one word at a time. Circuitry needed to interface to the Ram bus channel is included on the chip. Such chips are known as RDRAMs.

25. What are the special features of Direct RDRAMs? (U)

- It is a two channel Rambus..
- It has 18 data lines intended to transfer two bytes of data at a time.
- There are no separate address lines.

26. What are RIMMs? (U)

RDRAM chips can be assembled in to larger modules called RIMMs. It can hold upto 16 RDRAMs.

27. What is cache memory? (R)

It is a small, fast memory that is inserted between the larger, slower main memory and the processor. It reduces the memory access time.

28. Define cache line. (R)

Cache block is used to refer to a set of contiguous address locations of some size. Cache block is also referred to as cache line.

29. What are the two ways in which the system using cache can proceed for a write operation?(U)

- Write through protocol technique.
- Write-back or copy back protocol technique.

30. What is write through protocol?(U)

For a write operation using write through protocol during write hit: the cache location and the main memory location are updated simultaneously. For a write miss: For a write miss, the information is written directly to the main memory.

31. What is write-back or copy back protocol?(U)

For a write operation using this protocol during write hit: the technique is to update only the cache location and to mark it as updated with an associated flag bit, often called the dirty or modified bit. The main memory location of the word is updated later, when the block containing this marked word is to be removed from the cache to make room for a new block. For a write miss: the block containing the addressed word is first brought into the cache, and then the desired word in the cache is overwritten with the new information.

32. When does a read miss occur?(U)

When the addressed word in a read operation is not in the cache, a read miss occur.

33. What is load-through or early restart?(U)

When a read miss occurs for a system with cache the required word may be sent to the processor as soon as it is read from the main memory instead of loading in to the cache. This approach is called load through or early restart and it reduces the processor's waiting period.

34. What is a hit? (R)

A successful access to data in cache memory is called hit.

35. Define hit rate? (R)(NOV/DEC 2006) (APRIL/MAY 2009)(Nov / Dec 2013)

The number of hits stated as a fraction of all attempted access.

36. Define miss rate? (R)(Nov / Dec 2013)

It is the number of misses stated as a fraction of attempted accesses.

37. Define miss penalty? (R) (APRIL/MAY 2009)

The extra time needed to bring the desired information into the cache.

38. What is dirty or modified bit?(U)

The cache location is updated with an associated flag bit called dirty bit.

39. What is write miss? (R)

During the write operation if the addressed word is not in cache then said to be write miss.

40. What is virtual memory and what are the benefits of virtual memory?(U)(NOV/DEC 2007) (APRIL/MAY 2010)

Techniques that automatically move program and data blocks into the physical main memory when they are required for execution are called as virtual memory. The virtual memory concept also frees the programmer. The programmer no longer needs to worry about the size constraints of the physical memory on every computer his or her program is going to be used.

41. What is virtual address?(U)

The binary address that the processor used for either instruction or data called as virtual address.

42. What is virtual page number?(U)

Each virtual address generated by the processor whether it is for an instruction fetch is interpreted as a virtual page.

43. What is page frame?(U)

An area in the main memory that can hold one page is called as page frame.

44. What is meant by interleaved memory?(U) (May/June 2012)(May/June 2013)

Interleaved memory is a technique for compensating the relatively slow speed of DRAM. The CPU can access alternative sections immediately without waiting for memory to be cached. Multiple memory banks take turns supplying data.

45. What is Translation Look aside Buffer? (U)(MAY/JUNE 2006)

A translation look aside buffer (TLB) is a cache that memory management hardware uses to improve virtual address translation speed. All current desktop, notebook, and server processors use a TLB to map virtual and physical address spaces, and it is nearly always present in any hardware which utilizes virtual memory.

46. What are called memory-mapped I/O devices?(U)

When I/O devices and the memory share the same address space the arrangement is called memory-mapped I/O devices.

47. What are the two important mechanisms for implementing I/O operations?(U)

There are two commonly used mechanisms for implementing I/O operations .They are interrupts and direct memory access.

48. What are known as interrupts?(R)(MAY/JUNE 2006)

In the case of interrupts, the synchronization is achieved by having the I/O device send a special signal over the bus whenever it is ready for a data transfer operation.

49. What do you mean by direct memory access? (R) (NOV/DEC 2007)

Direct memory access is a technique used for high speed I/O devices. It involves having the device interface transfer data directly to or from the memory.

50. What do you mean by an interrupt- request line?(U)

The bus control line is also known as an interrupt-request line.

51. What do you mean by an interrupt acknowledge signal?(U)

The processor must inform the device that its request has been recognized so that it may remove its interrupt-request signal .This may be accomplished by an interrupt acknowledge signal.

52. What is a subroutine?(U)(NOV/DEC 2007)
A subroutine performs a function required by the program from which it is called.

53. What is interrupt latency?(U)
Saving registers also increases the delay between the time an interrupt request is received and the start of execution of the interrupt-service routine .This delay is called interrupt latency.

54. What is known as real-time processing?(U)
The concept of interrupts is used in operating systems and in many control applications where processing of certain routines must be accurately timed relative to external events .The latter type of application is referred to as real-time processing.

55. What is known as a edge triggered line?(U)
The processor has a special interrupt-request line for which the interrupt handling circuit responds only to the leading edge of a signal .such a line is called a edge-triggered line.

56. What is known as an interrupt vector?(U)
The location pointed to by the interrupting device is used to store the starting address of the interrupt-service routine .The processor reads this address ,called the interrupt vector.

57. What is known as a debugger?(U)
System software usually includes a program called a debugger, which helps the programmer find errors in a program.

58. What is an exception? (R)
The term exception is often used to refer to any event that causes an interruption.

59. What are known as privileged instructions?(U)
To protect the operating system of a computer from being corrupted by user programs, certain instructions can be executed only while the processor is in the supervisor mode. These are called privileged instructions.

60. What is known as multitasking? (R)
Multitasking is a mode of operation in which a processor executes several user programs at the same time.

61. What is known as direct memory access?(R)NOV/DEC 2006)
A special control unit may be provided to allow transfer of a block of data directly between an external device and the main memory, without continuous intervention by the processor. This approach is called direct memory access, or DMA.

62. What is known as a DMA controller?(U)
DMA transfers are performed by a control circuit that is part of the I/O device interface. This circuit is known as DMA controller.

63. What is known as cycle-stealing?(U)
The processor originates most memory access cycles, the DMA controller can be said to "steal" memory cycles from the processor. Hence ,this interweaving technique is usually called cycle stealing.

64. What is known as block/burst mode?(U)
The DMA controller may be given exclusive access to the main memory to transfer a block of data without interrupt. This is known as block/burst mode.

65. What is called a bus master?
The device that is allowed to initiate data transfers on the bus at any given time is called the bus master.

66. What is meant by bus arbitration? (R) (Apr/May 2010)
Bus arbitration is a way of sharing the computer's data transferring channels (buses) in an optimal way so the faster devices won't have to wait to be able to transfer and the slower devices (like peripherals) will have a chance to transfer as well. Different methods exist but two main types are the serial and parallel arbitrations

67. Name and give the purpose of widely used bus standard. (U)(Apr/May 2010)
The GPIB or General Purpose Interface Bus or IEEE 488 bus is still one of the more popular and versatile interface standards available today. GPIB is widely used for enabling electronics test equipment to be controlled remotely, although it us also used in a many other applications including data acquisition. Today most bench electronics test equipment has either a GPIB option or are fitted with it as standard.

68. What is known as distributed arbitration?(U)(Apr/May 2011)
Distributed arbitration means that all devices waiting to use the bus have equal responsibility in carrying out the arbitration process, without using a central arbiter

69. What is a strobe?(U)
Strobe captures the values of the data given instant and stores them into a buffer.

70. What is meant by handshake?(U)
Handshake is used between the master and the slave for controlling data transfers on the bus.

71. What is known as full handshake? (U)
A change of state in one signal is followed by a change in the other signal. This is known as a full handshake.

72. What is a bit rate? (R)
The speed of transmission is known as a bit rate.

73. What is known as plug- and- play? (U)
The plug-and –play feature means that a new device, such as an additional speaker, can be connected at any time while the system is operating.

74. What is called a hub? (R)
Each node of the tree has a device called a hub which acts as an intermediate control point between the host and the I/O devices.

75. What is a root hub? (U)
At the root of a tree, a root hub connects the entire tree to the host computer.

76. Specify the different I/O transfer mechanisms available(U) (May/June 2012)
The mechanisms for data transfer differ greatly based on the transfer speed of the device. Therefore it has three different busses: the address bus, the data bus, and the control bus.

77. What is priority interrupt? (U) (APRIL/MAY 2008)
An interrupt procedure in which control is passed to the monitor, the required operation is initiated, and then control returns to the running program, which never knows that it has been interrupted.

78. Distinguish between isolated and memory mapped I/O.(An)(MAY/JUNE 2013)
Isolated I/O, have special instructions for I/O operations. For example, you might have an assembly languages instructions input or output, or various versions of these instructions for different devices. With memory-mapped I/O, the addresses of the registers and/or memory in each I/O device are in a dedicated region of the kernel's virtual address space. This allows the same instructions to be used for I/O as are used for reading from and writing to memory, e.g. in MIPS you use lw and sw.

79. What is the use of DMA?(U) (NOV/DEC 2012) (Apr/May 2018)
A DMA controller can generate addresses and initiate memory read or write cycles. It contains several registers that can be written and read by the CPU. These include a memory address register, a byte count register, and one or more control registers. To carry out an input, output or memory-to-memory operation, the host processor initializes the DMA controller with a count of the number of words to transfer, and the memory address to use. DMA transfers can either occur one word at a time or all at once in burst mode.

80. Differentiate programmed I/O and interrupt I/O (U) (Dec-2014)
Programmed I/O

- I/O module does not inform CPU directly
- CPU may wait or do something and come back later
- Wastes CPU time because typically processor is much faster than I/O
- CPU acts as a bridge for moving data between I/O module and main memory, i.e., every piece of data goes through CPU
- CPU waits for I/O module to complete operation

Interrupt I/O

- Overcomes CPU waiting
- Requires setup code and interrupt service routine
- No repeated CPU checking of device
- I/O module interrupts when ready
- Still requires CPU to be go between for moving data between I/O module and main memory

81. What is the purpose of Dirty/Modified bit in cache memory(U) (Dec-2014)

A dirty bit or modified bit 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. The dirty bit is set when the processor modifies this memory. The bit indicates that its associated block of memory has been modified and has not yet been saved to storage.
Dirty bits are used by the CPU cache and in the page replacement algorithms of an operating system

82. What is the need to implement memory as a hierarchy (U) (Apr/May 2015) (16)

- To provide CPU with necessary data (and instructions) as quickly as possible
- To reduce traffic on memory bus

83. Point out how DMA can improve I/O speed(U) (Apr/May 2015) (16)

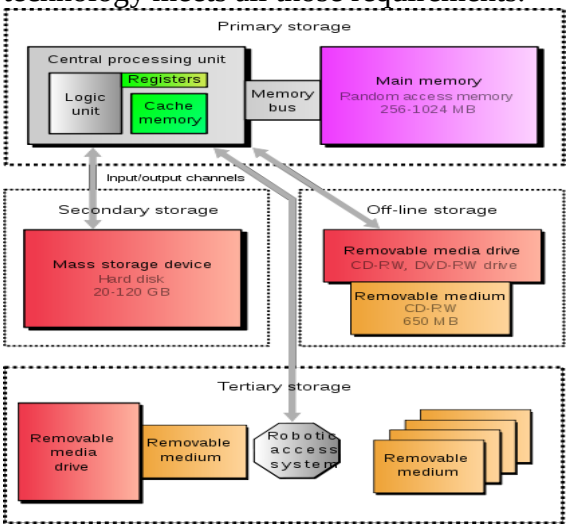
- DMA module controls exchange of data between memory and an I/O module
- CPU sends a request to transfer a block of data to the DMA module and is interrupted only after the entire block has been transferred
- DMA module takes over control of system bus to perform the transfer
- CPU initiates the I/O by sending the following information to DMA module

84. Define Hit Ratio(U) (NOV/DEC 2015)

The hit ratio is the fraction of accesses which are a hit. The miss ratio is the fraction of accesses which are a miss. It holds that. Miss rate = 1 − hitrate . The (hit/miss) latency (AKA access time) is the time it takes to fetch the data in case of a hit/miss.

85. What are the various memory technologies (R)(NOV/DEC 2015?)

Computers have employed various technologies to preserve information. Most fall into two broad categories: memory and storage.
Memory holds running programs and information the processor is currently using. Storage preserves data and programs for future use.
Memory must be fast and flexible. Storage has to be big, permanent, and affordable. No single technology meets all those requirements.



86. What is meant by address mapping? (U) (NOV/DEC 2016)

A related choice is the granularity of address mapping, which is defined as the smallest unit of addressed data (from the persistent store) that can be mapped independently to an area of the virtual address space.

87. What is cache memory? (R) (NOV/DEC 2016)

Cache memory, also called CPU memory, is random access memory (RAM) that a computer microprocessor can access more quickly than it can access regular RAM. This memory is typically integrated directly with the CPU chip or placed on a separate chip that has a separate bus interconnect with the CPU.

88. Define Memory Hierarchy (U) (MAY/JUNE 2016)

In computer architecture the memory hierarchy is a concept used to discuss performance issues in computer architectural design, algorithm predictions, and lower level programming constructs involving locality of reference. The memory hierarchy in computer storage separates each of its levels based on response time.

89. State the advantages of virtual memory (U) (MAY/JUNE 2016)

1. Main memory is used more efficiently
2. Programs that are bigger than
3. Main memory can still be executed.

90. What is virtual memory. (U) (Nov/Dec 2017)

Virtual memory is a memory management capability of an OS that uses hardware and software to allow a computer to compensate for physical memory shortages by temporarily transferring data from random access memory (RAM) to disk storage. Virtual address space is increased using active memory in RAM and inactive memory in hard disk drives (HDDs) to form contiguous addresses that hold both the application and its data.

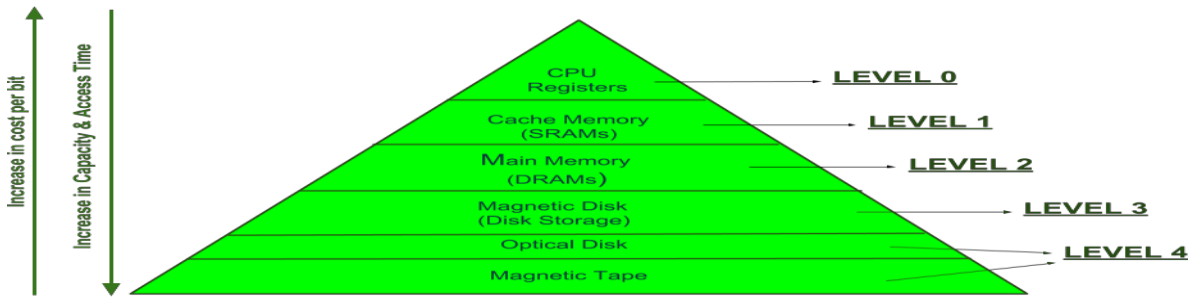
91. How many total bits are required for a direct mapped cache with 16KB of data and 4-word blocks, assuming a 32-bit address?(An) (Nov/Dec 2017)

Each block hs 32 bits of data plus a tag, which is 32-14-2 bits, plus a valid bit. Thus, the total cache size is (2 to the power 14)*49= 784 bits or 98KB for a 64-KB cache. For this cache, the total number of bits in the cache is over 1.5 times as many as needed just for the storage of the data.

92. What is meant by memory –mapped I/O?(Nov/Dec 2018)-R

Memory mapped I/O is a way to exchange data and instructions between a CPU and peripheral devices attached to it. Memory mapped IO is one where the processor and the IO device share the same memory location(memory),i.e. the processor and IO devices are mapped using the memory address.

93. Draw the basic structure of a memory hierarchy. (R)(Nov/Dec 2018)(April/May 2019)



MEMORY HIERARCHY DESIGN

94. How many total bits are required.for a direct-mapped cache with 16KB of data and 4-word blocks, assuming a 32-bit address?(Ap) (April/May 2019)
$2^n x (2^m x 32 + 31 – n - m)$

95. When is a memory unit called as RAM ? (Nov/Dec2020 / April/May2021)(U)
The term RAM refers solely to solid-state memory devices (either DRAM or SRAM), and more specifically the main memory in most computers.

96. Define vectored interrupts. (Nov/Dec2020 / April/May2021)(U)
A vectored interrupt is an I/O interrupt that tells the part of the computer that handles I/O interrupts at the hardware level that a request for attention from an I/O device has been received and and also identifies the device that sent the request.

A vectored interrupt is an alternative to a polled interrupt , which requires that the interrupt handler poll or send a signal to each device in turn in order to find out which one sent the interrupt request.

PART-B

1. Describe the organization of a typical RAM chip.**(MAY/JUNE 2007) (U)**
2. Explain about Synchronous DRAMS. **(May/June 2012)(May/June 2013) (U)**
3. Explain about Static & Dynamic memory systems. **(NOV/DEC 2007) (U)**
4. Write note on:
   i ROM technologies.
   ii Memory Interleaving
   iii Set associative mapping of cache.
   iv RAID Disk arrays. (**NOV/DEC 2007) (U)**
5. Explain various mechanisms of mapping main memory address into cache memory addresses.(**APRIL/MAY 2008) (May/June 2012**) **(May/June 2013) (Nov / Dec 2013)(Dec-2014)(An)**
6. Explain about Cache memory in detail **(NOV/DEC 2006)(Nov/Dec 2017) (U)**
7. Explain how the virtual address is converted into real address in a paged virtual memory system.(**APRIL/MAY 2008**)**(May/June 2013) (Nov / Dec 2013)(An)**
8. Discuss the address translation mechanism and the different page replacement policies used in a virtual memory system. **(MAY/JUNE 2006)(An)**
9. Explain the concept of memory hierarchy. **(NOV/DEC 2007) (U)**
10. Explain the performance factors in memory. (**MAY/JUNE 2006)(U)**

11. Explain how the virtual address is converted into real address in a paged virtual memory system.**(Apr/May 2010)(May/ June 2012)(An)**

12. Explain the need for cache memory and discuss the different types of mapping functions with necessary block diagram. (**NOV/DEC 2012) (An)**

13. Discuss the steps involved in address translation of virtual memory with necessary block diagram. (**NOV/DEC 2012)(C)**

14. Draw the block diagrams of two types of DRAMS and explain address translation method in virtual memory. **(MAY/JUNE 2013) (C)**

15. Explain interrupt priority schemes? **(May/June 2012)(U)**

16. Explain how I/O devices can be interfaced with a block diagram. (NOV/DEC 2007)(U)

17. Explain DMA and the different types of bus arbitration mechanisms. **(May/June 2012)(Dec-2014)(U)**

18. Design parallel priority interrupt hardware for a system with eight interrupt source. **(MAY/JUNE 2007)(U)**

19. Explain how DMA transfer is accomplished with a neat diagram**. (NOV/DEC 2006)(U)**

20. Draw the typical block diagram of a DMA controller and explain how it is used for direct data transfer between memory and peripherals.**(APRIL/MAY 2008) (MAY/JUNE 2007)(C)**

21. Explain the use of vectored interrupts in processors. Why is priority handling desired in interrupt controllers? How do the different priority schemes work?**(MAY/JUNE 2006)(An)**

22. How do you connect multiple I/O devices to a processor using interrupts? Explain with suitable diagrams **(NOV/DEC 2007)(An)**

23. Explain the use of DMA Controllers in a computer system with a neat diagram.(U)

24. Describe the hardware mechanism for handling multiple interrupt requests. **(APRIL/MAY2010)(U)**

25. What are handshaking signals? Explain the handshake control of datatransfer during input and output operation. **(APRIL/MAY 2010)(Apr/May 2011)(U)**

26. What are the needs for input- output interface? Explain the functions of a typical 8- bit parallel interface in detail. **(APRIL/MAY 2010) (Apr/May 2011)(Dec-2014)(U)**

27. Explain the need for memory hierarchy technology, with a four-level memory?
(Nov / Dec 2013)(U)

28. What is an interrupt? Explain the different types of interrupts and the different ways of handling interrupts. **(NOV/DEC 2012)(U)**

29. Draw different memory address layouts and brief about the technique used to increase the average rate of fetching words from the main memory **(8) (Dec-2014)(Ap)**

30. Elaborate on the various memorytechnologies andits relevance **(Apr/May 2015) (Nov/Dec 2017) (16)(U)**

31. What is virtual memory? Explain the steps involved in virtual memory address translation **(Apr/May 2015) (16)(U)**

32. What is virtual memory? Explain in detail how the virtual memory is implemented with neat diagram **(16) (NOV/DEC 2015) (Apr/May 2018) (Ap)**

33. Draw a typical block diagram of DMA controller and explain how it is used for direct data transfer between memory and peripherals **(16) (NOV/DEC 2015)(Ap)**

34. Discuss DMA controller with block diagram **(16) (NOV/DEC 2016)(Ap)**

35. Discuss the steps involved in the address translation of virtual memory with necessary block diagram **(16) (NOV/DEC 2016)(An)**

36. Define cache Memory? Explain the various mapping techniques associates with cache memories?**(16) (MAY/JUNE 2016)(Apr/May 2018)(An)**

37. Explain about DMA controller, with the help of a block diagram**(16) (MAY/JUNE 2016)(Ap)**
38. (i) Discuss about Programmed I/Os associated with computers. (6)**(U)**
(ii) Write the sequence of operations carried out by a processor when interrupted by a peripheral device connected to it. (7) **(An)(Apr/May 2018)**
39.Discuss the three mapping techniques in memory hierarchy. Explain with examples(10)

(Nov/Dec 2018)-U

40. Define Translation Lookaside Buffer (TLB).What is its use?(3) **(Nov/Dec 2018)-R**

41.Explain mechanisms Direct Memory Access and Interrupt handling(6+7)

(Nov/Dec 2018)-U

42. Explain the various mapping functions that can be applied on cache memories in detail.**(13) (April/May 2019)(U)**
43. With a neat sketch explain the working principle of DMA. **(8) (April/May 2019) (U)**
44. Explain about input-output processor (IOP) **(5) (April/May 2019) (U)**
45. i) With neat sketch explain about Synchronous DRAMS. (6)
(Nov/Dec2020 / April/May2021)(U)
ii) With neat sketch explain about Asynchronous DRAMS. (7) **(Nov/Dec2020 / April/May2021)(U)**
46. Explain briefly about direct memory access.(13) **(Nov/Dec2020 / April/May2021)(U)**

COURSE OUTCOME:
Students can able to Understand the various memory systems and I/O communication.

PART-C

1. What is the disadvantage of Ripple carry addition and how it is overcome in carry look ahead adder and draws the logic circuit CLA? **(16) (An) (NOV/DEC 2016)**
2. Design and explain parallel priority interrupt hardware for a system with eight interrupt sources. **(16) (C) (NOV/DEC 2016)**
3. (i) Suppose you want to achieve a speed up of 90 times faster with 100 processors. What percentage of the original computation can be sequential? (8)
(ii)Suppose you want to perform two sums: one is a sum of 10 scalar variables and one is a matrix sum of a pair of two dimensional arrays, with dimensions 10 by 10. For now let's assume only the matrix sum is parallelizable; what speed up do you get with 10 versus 40 processors? Next, calculate the speed-ups assuming the matrices grow to 20 by 20. **(7) (An) (Nov/Dec 2017)**
4. Assume the miss rate of an instruction cache is 2% and the miss rate of the data cache is 4%. If a processor has a CPI of 2 without any memory stalls and the miss penalty is 100 cycles for all misses, determine how much faster a processor would run with a perfect cache that never missed. Assume the frequency of all loads and stores is 36% **(15) (An) (Nov/Dec 2017)**
5. The following sequence of instructions are executed in the basic 5-stage pipelined processor:
Or r1,r2,r3
Or r2,r1,r4
Or r1,r1,r2
a) Indicate dependencies and their type.
b) Assume there is no forwarding in this pipelined processor. Indicate hazards and add NOP instructions to eliminate them.

c) Assume there is full forwarding. Indicate hazards and add NOP instructions to eliminate them. **(15) (An) (Apr/May 2018)**

6. Explain the detail of DMA control with suitable diagrams. Discuss how it improve the overall performance of the system. **(15) (U) (Apr/May 2018)**

7. (i) Consider web browsing application. Assuming both client and server are involved in the process of web browsing application, where can caches be placed to speed up the process? Design a memory hierarchy for the system. Show the typical size and latency at various levels of the hierarchy. What is the relationship between cache size and its access latency? What are the units of data transfers between hierarchies? What is the relationship between the data location, data size and transfer latency?(An)

(ii)The following sequence of instructions are executed in the basic 5-stage pipelined processor:

```
1w $1, 40($6)
add $6, $2, $2
sw $6, 50($1)
```

Indicate dependences and their type. Assuming there is no forwarding in this pipelined processor, indicate hazards and add NOP instructions to eliminate them**. (An)(NOV/DEC 2018)(15)**

8. Compare hardwired and microprogrammed control unit designs in terms of their mechanism of generating control signals with diagram.. (An)(NOV/DEC 2018)(15)

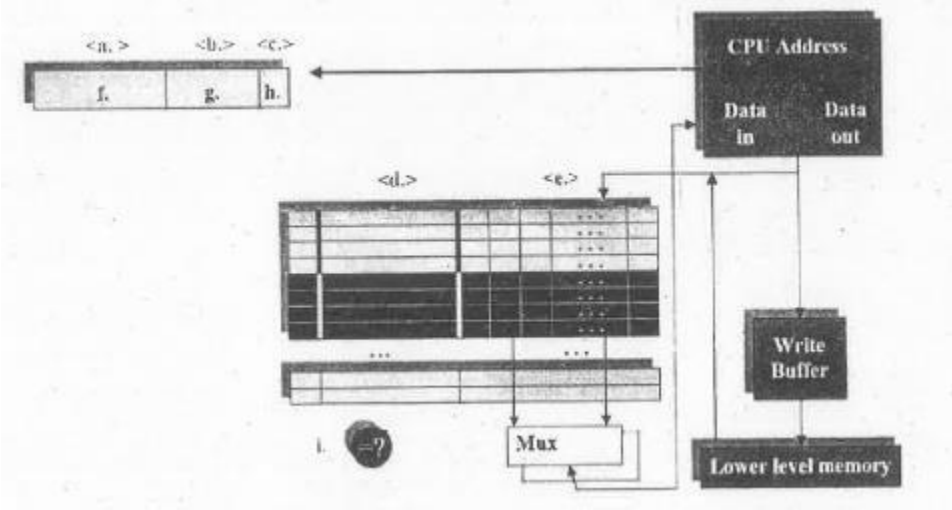
9. In a small town, there are three temples in a row and a well in front of each temple. A pilgrim came to the town with certain number of flowers. Before entering the first temple, he washed all the flowers ho had with the water of well. To his surprise, flowers doubled. He offered few flowers to the God in the first temple and moved to the second temple. Here also, before entering the temple he washedthe remaining flowers with the water of well. And again his flowers doubled. He offered few flowersto the God in second temple and moved to the third temple. Here also his flowers doubled after washing them with water. He offered few flowers to the God in third temple.

There were no flowers left when pilgrim came out of third temple and he offered same number of flowers to the God in all three temples. What is the minimum number of flowers the pilgrim had initially (X)? And find the value of (X/3) using Restoring Division method? How many flower did he offer to each God (Y)? And find the value of (Y/3) using Non-Restoring Division method? **(15) (Apr/May 2019) (Ap)**

10. You have been asked to design a cache with the following properties : (8) **(Apr/May 2019)U**

(1) Data words are 32 bits each
(2) A cache block will contain 2048 bits of data
(3) The cache is direct mapped
(4) The address supplied from the CPU is 32 bits long
(5) There are 2048 blocks in the cache
(6) Addresses are to the word.

11. In the below picture, there are 8 fields (labeled a, b, c, d, e, f, g, and h), you will need to indicate the proper name or number of bits for a particular portion of this cache configuration. Explain the process of accessing data using this design (7)
(Apr/May 2019) (8)-U



12. Discuss the different mapping techniques used in cache memories and their relative merits and demerits. **(Nov/Dec2020 / April/May2021)(U)**

13. b) A computer system uses 16-bit memory addresses. It has a 2K-byte cache organized in a direct-mapped manner with 64 bytes per cache block. Assume that the size of each memory word is 1 byte.

i) Calculate the number of bits in each of the Tag, Block and Word fields of the memory address. (7) **(Nov/Dec2020 / April/May2021)(U)**

ii) If the cache is organized as a 2-way set associative cache that uses the LRU

replacement algorithm. (8) **(Nov/Dec2020 / April/May2021)(U)**

Course Name : CS8491 /COMPUTER ARCHITECTURE
Year/Semester : II/ IV
Year of Study : 2021 –2022 (R – 2017)

On Completion of this course student will be able to

COURSE OUTCOMES

CO	DESCRIPTION
CO1	Understand the basics structure of computers, operations and instructions.
CO2	Design arithmetic and logic unit.
CO3	Understand pipelined execution and design control unit.
CO4	Understand parallel processing architectures.
CO5	Understand the various memory systems and I/O communication

CO-PO MATRIX:

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO.1	3	2	3	-	-	-	-	-	-	-	-	-
CO.2	3	2	3	-	-	-	-	-	-	-	-	-
CO.3	3	2	3	-	-	-	-	-	-	-	-	-
CO.4	3	2	3	-	-	-	-	-	-	-	-	-
CO.5	3	2	3	-	-	-	-	-	-	-	-	-
CO	3	2	3	-	-	-	-	-	-	-	-	-

CO – PSO MATRIX:

CO	PSO1	PSO2	PSO3
CO.1	3	3	-
CO.2	3	3	-
CO.3	3	3	-
CO.4	3	3	-
CO.5	3	3	-
CO	3	3	-