# Lesson Plan

*Cover Page*: Course Overview

*Semester:* **III**  Year: **2017-18**

|  |  |
| --- | --- |
| *Course Title*:  **COMPUTER ORGANIZATION** | *Course Code*: **16CS35** |
| *Total Contact Hours*: 44 hrs | *Duration of SEE*: 3 **hrs** |
| *SEE Marks*: **100 + 50** | *CIE Marks*: **100** |
| *Lesson Plan Author*: Prof. Anala M R, , Prof. Deepika Dash, Prof. Veena , Prof. Mamatha T | *Date*: **Jun 06, 2017** |
| *Checked By:* | *Date*: |

## Course Overview:

This course provides students the ability to visualize the overall layout of various computing elements in a typical computer. This course also provides an overview on Assembly language instruction, Instruction Set Architectures, Memory Layout, Arithmetic computations.

## Course Learning Objectives-CLO

Computer Organization is the fundamental step in understanding the Organization of various processing elements within a computer and also an microscopic focus on understanding instruction execution with collaborated efforts of various computing elements (Memory, IO etc.,). This course lays down the following objectives -

1. Understand the fundamentals of computer System and its Organization.
2. Appreciate the functionalities of basic processing unit and its control system in processing the Instruction.
3. Understand the role of bus system and its interfaces.
4. Develop a clear understanding on the Memory System and its design.
5. Present an adequate Instruction Set Architectures for better understanding of the assembly level programming.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Course Outcomes: After completing the course, the students will be able to** | | | 1 | Understand and explore the basic operation and organization of computer system | | 2 | Identify the design requirements in organizing computer system components | | 3 | Develop assembly language program for different instruction set architecture and its data representation | | 4 | Examine the different interfaces of a computer system | |

**Course Content**

**Course Code: 16CS35**

**Hrs/Week L-T-P-S: 4-0-0-1 CIE: 100 marks**

**Teaching Hours: 44 Hrs SEE: 100 marks**

|  |  |  |
| --- | --- | --- |
|  | **UNIT-I** | **Hours** |
| **1** | **Basic Structures of Computers**  Functional units, Basic Operational Concepts, Bus Structures. Performance | **3 Hrs** |
| **2.** | **Machine Instructions and Programs**  Memory Locations and Addresses, Memory Operation, Instruction and Instruction Sequencing, Addressing Modes, implementation of Variables & Constants, Indirection & pointers, Indexing & Arrays, Relative Addressing, Additional Modes, Assembly Language, Numbers, Number Notation, Arithmetic operations and characters, Example Programs. | **5 Hrs** |
|  | **UNIT-II** |  |
| **3.** | **Machine Instructions and Programs**  Stacks & Queues, Subroutines, Subroutine Nesting & Processor Stack, Parameter passing, The stack frame. Additional Instructions, Example programs. | **4 Hrs** |
| **4.** | **ARM Instruction sets**  Register Memory access and data transfer, Arithmetic and logical instructions, Branch instructions, Assembly language programs. | **5 Hrs** |
|  | **UNIT-III** |  |
| **5.** | **Input / Output Organization**  Basic Input / Output Operations, Accessing I/O devices, Interrupts: Interrupt Hardware, Enabling & Disabling Interrupt, Handling Multiple Devices, Controlling Device Requests, Exceptions, Direct Memory Access. | **4 Hours** |
| **6.** | **Buses**  Bus Arbitration, Synchronous Bus, Asynchronous Bus, PCI bus, SCSI bus, USB(Objective & Architecture) | **5 Hours** |
|  | **UNIT-IV** |  |
| **7.** | **Memory Design**  Semiconductor RAM memories: Internal Organization of Memory Chips, Static Memories, Asynchronous Drams, Synchronous DRAMs, Structure of Larger Memories, Memory System Considerations, Read-only memory, Speed, Size and cost. | **5 Hours** |
| **8.** | **Memory Types**  Cache memories: Mapping functions, performance considerations, Hit rate and miss penalty, Caches on the processor chip, other enhancements and virtual memory. | **4 Hours** |
|  | **UNIT-V** |  |
| **9.** | **Control Unit Logic**  Fundamental Concepts: Register Transfers, Performing an Arithmetic or Logic operation, Fetching a Word from Memory, Storing a Word in Memory, Execution of a Complete Instruction, Branch instruction. Multiple Bus Organization, Hardwired control, Basics of Micro programmed control. | **5 Hrs** |
| **10.** | **Number Representation and Arithmetic Operations**  Booth Algorithm, Fast Multiplication: Bit-pair Recording of Multipliers; Integer division, Floating-point Numbers & Operations, IEEE Standard for Floating-point Numbers. | **4 Hrs** |

**REFERENCE BOOKS:**

1. Carl Hamacher, Z Vranesic & S Zaky, Computer Organization, Mc Graw Hill, 5th Edition, 2002 ISBN10: 0-07-112218-4.
2. William Stallings, Computer Architecture and Organization, PHI, 6th edition, 2003, ISBN10: 0-13-049307-4.
3. David A. Patterson and John L. Hennessy, Computer Organization and Design, Elsevier, 3rd Edition, 2005, ISBN10: 1-55860-604-1.
4. Mostafa Abd-El-Barr, Hesham El-Rewini, FUNDAMENTALS OF COMPUTER ORGANIZATION ANDARCHITECTURE, Wiley publisher, 2005, ISBN10: 0-471-46741-3.

**Unit and Chapter wise Plan**

**Unit 1**

|  |  |
| --- | --- |
| *Course Code and Title:* (**16CS35) Computer Organisation** | |
| *Chapter Number and Title*:1. Basic Structures of Computers | *Planned Hours:* ***3* hrs** |

## Learning Objectives:

|  |  |
| --- | --- |
| **Sl. No.** | **Objectives** |
| **1** | Identify the fundamental units of computing |
| **2** | Discuss the bus structure and Processor Clock |
| **3** | Understand Pipelining and Superscalar operation |
| **4** | Analyze the performance measure by SPEC benchmarks process for any computer under test |

|  |
| --- |
| ***Lesson Schedule***  *Class No. Portion covered per hour*   1. Fundamental units ,basic operational concepts and Bus structures 2. Basic Performance equation, Pipelining and Superscalar Operation 3. Processor Clocks, Clock rate and Performance Measurements |

## Model Questions

With Neat block diagram explain the various units of the Computer?

With simple instruction illustrate the basic operational concepts and describe the various register used in the process?

Explain role of bus in processing and instruction and also state various bus concepts.

* + - 1. With an example illustrate how pipelining and superscalar operations are carried out.
      2. Explain different types of hazards in pipelining with examples.
      3. Discuss the basic performance equations of a computer system.
      4. Discuss the various generations through which the computers have evolved to the present stage. Indicate the important technological features and devices that characterized each generation

|  |  |
| --- | --- |
| *Course Code and Title:* (**16CS35) Computer Organisation** | |
| *Chapter Number and Title*: *2****.***   **Machine Instructions and Programs** | *Planned Hours:* ***05*  hrs** |

## Learning Objectives

|  |  |
| --- | --- |
| **Sl. No.** | **Objectives** |
| **1** | Discuss key concepts like Byte addressability, Memory operations, Strings, Memory formats. |
| **2** | Analyze the Instruction and Instruction sequencing and branching methods. |
| **3** | Discuss various instruction notations like Register Transfer notation and Assembly language notation |

|  |
| --- |
| ***Lesson Schedule***  *Class No. Portion covered per hour*   1. Memory Locations and Addresses, Memory Operation, Instruction and Instruction Sequencing 2. Addressing Modes, implementation of Variables & Constants, Indirection & pointers, 3. Indexing & Arrays, Relative Addressing, Assembly Language,. 4. Numbers, Number Notation, Arithmetic operations and characters, 5. Example Programs. |

**Model Questions**

1. Registers R1, R2, R3 of a processor contain the decimal values 1250 and 5500, 25. Determine the values of the registers after each instruction is executed. a) Load R3, (R2), b) Move #0250, R5, c) Add (R2)+, R5, d) Add –(R2), R1 e) Store R5, 20(R1, R2).
2. List three important differences between how a stack and a queue are organized (5 marks)
3. Quoting an example for each explain i) Different types of instructions in a computer to perform various operations. ii) Different addressing modes used to specify the location of operand.
4. What is an addressing mode? Explain different addressing modes with examples.
5. Write an assembler language program to solve an expression ax2 + bx + c= 0 using two addressing modes.

**Unit II**

|  |  |
| --- | --- |
| *Course Code and Title:* (**16CS35) Computer Organisation and Architecture** | |
| *Chapter Number and Title*: *3*  **Machine Instructions and Programs** | *Planned Hours:* ***04* hrs** |

## Learning Objectives

|  |  |
| --- | --- |
| **Sl. No.** | **Objectives** |
| **1** | Discuss the key concepts on stacks, queues and subroutines. |
| **2** | Analyze parameter passing and subroutine nesting |
| **3** | Understand Stack frame and processor stack. |
| **4** | Explain Assembly language directives and implement assembly programs. |

|  |
| --- |
| ***Lesson Schedule***  *Class No. Portion covered per hour*   1. Introduction to Stacks & Queues, Subroutines. 2. Subroutine Nesting & Processor Stack, 3. Parameter passing, The stack frame. 4. Additional Instructions, Example programs. |

**Model Questions**

1. Registers R1, R2, R3 of a processor contain the decimal values 1250 and 5500, 25. Determine the values of the registers after each instruction is executed. a) Load R3, (R2), b) Move #0250, R5, c) Add (R2)+, R5, d) Add –(R2), R1 e) Store R5, 20(R1, R2).
2. List three important differences between how a stack and a queue are organized.
3. Quoting an example for each explain: i) Different types of instructions in a computer to perform various operations. ii) Different addressing modes used to specify the location of operand.
4. Write an assembler language program to solve an expression ax2 + bx + c= 0 using two addressing modes.
5. Consider the following :
6. A subroutine may required the parameters passed to it (from the main program),in a random order and more than once.
7. A stack is a data structure in memory, from which the data can be accessed in a IIFO order

and obviously ten stack is not a suitable data structure for handling subroutine parameters.

1. Yet, passing subroutine parameters through stack is perhaps the commonest way of handling subroutine parameters. What mechanism is employed so that the parameters are made randomly accessible to the subroutine from the stack? Explain with an example.

|  |  |
| --- | --- |
| *Course Code and Title:* (**16CS35) Computer Organisation** | |
| *Chapter Number and Title*:4.  **ARM Instruction sets** | *Planned Hours:* ***04* hrs** |

## Learning Objectives

|  |  |
| --- | --- |
| **Sl. No.** | **Objectives** |
| **1** | Discuss various instructions of ARM processor. |
| **2** | Write assembly programs using ARM instructions |

|  |
| --- |
| ***Lesson Schedule***  *Class No. Portion covered per hour*   1. Register Memory access and data transfer 2. Arithmetic and logical instructions 3. Branch instructions 4. Assembly language programs 5. Assembly language programs using arm instruction set. |

**Model Questions**

* + - 1. Discuss various instruction notations like Register Transfer notation and Assembly language notation in ARM processor
      2. What is an addressing mode? Explain different addressing modes of Arm processor with examples.
      3. Discuss arithmetic and logical instructions supported by ARM.
      4. Illustrate the use of branch instructions in ARM processor.

## UNIT- 3

|  |  |
| --- | --- |
| *Course Code and Title:* (**16CS35) Computer Organisation** | |
| *Chapter Number and Title*: *5****.***   Input/output Organization | *Planned Hours:* ***04* hrs** |

## Learning Objectives

|  |  |
| --- | --- |
| **Sl. No.** | **Objectives** |
| **1** | Discuss various IO devices |
| **2** | Explain Interrupts and identify various interrupt handling mechanisms |
| **3** | Discuss key topics like Adding multiple devices. Handling exceptions and Bus Arbitration schemes. |
| **4** | Explain exception handling and DMA |

|  |
| --- |
| ***Lesson Schedule***  *Class No. Portion covered per hour*   1. Accessing I/O devices, Interrupts : Interrupt Hardware, 2. Enabling & Disabling Interrupt, Handling Multiple Devices, 3. Controlling Device Requests, Exceptions 4. Direct Memory Access |

**Model Questions**

1. For a simple example of I/O operations involving a keyboard and a display device, Write a assembly language program that reads one line from the keyboard, stores it in memory buffer and echoes it back to the display.
2. Explain how interrupt request from several I/O devices can be communicated to a processor through a single INTR line.
3. Which type of I/O devices are interfaced through DMA? Explain the bus-arbitration process used for DMA.
4. Show a circuit arrangement, whereby several devices may interrupt a processor on a single interrupt request line. If it is required to handle the device interrupts on a fixed priority basis, indicate in details

i) A hardware based method

ii) A software based method for addressing this requirement.

1. Why bus arbitration is required? Explain with block diagram bus arbitration using daisy chain.
2. Explain the operation of a 2- channel DMA controller.
3. Differentiate between subroutine and interrupt service routine

|  |  |
| --- | --- |
| *Course Code and Title:* (**16CS35) Computer Organisation** | |
| *Chapter Number and Title*: 6.  **Buses** | *Planned Hours:* ***05* hrs** |

## Learning Objectives

|  |  |
| --- | --- |
| **Sl. No.** | **Objectives** |
| **1** | Discuss the roles of buses and overview on Bus structure in computational environment. |
| **2** | Understand the need for Interface Circuits : Parallel and Serial Ports |
| **3** | Discuss various types of buses – PCI, USB |

|  |
| --- |
| ***Lesson Schedule***  *Class No. Portion covered per hour*   1. Bus arbitration 2. Synchronous Bus, 3. Asynchronous Bus, 4. PCI 5. SCSI and USB |

**Model Questions**

1. Describe the operation of synchronous and asynchronous bus.
2. Explain the significant features of any ONE of the following bases;

i) PCI ii) SCSI iii)USB

1. List out the functions of an I/O interface.
2. Explain how arbitration and selection is done in the SCSI bus.

5. Explain how USB operates with split-traffic mode.

|  |  |
| --- | --- |
| *Course Code and Title:* (**16CS35) Computer Organisation** | |
| *Chapter Number and Title*: *7.*  **The Memory System** | *Planned Hours:* ***05*hrs** |

**UNIT- IV**

## Learning Objectives

|  |  |
| --- | --- |
| **Sl. No.** | **Objectives** |
| **1** | Explain the Concepts of Static Memories, Asynchronous DRAMs, Synchronous DRAMs, |
| **2** | Design Memory systems and Understand the metrics for evaluating a memory layout design by speed, size and cost |
| **3** | Discuss the basics of Caches and cache mapping functions. |
| **4** | Handling of Cache Misses and Handling Writes to caches |

|  |
| --- |
| ***Lesson Schedule***  *Class No. Portion covered per hour*   1. Static Memories, Asynchronous DRAMs, 2. Synchronous DRAMs 3. Structure of Larger Memories 4. Rambus memory, Read-only memory, 5. Speed, Size & cost.Large and Fast Exploding Memory Hierarchy Introduction. |

**Model Questions**

Describe SDRAM and DDR SD RAM operations for data transfer between main memory and cache memory systems.

With the block diagram explain the operation of a 16-megabit DRAM chip configured as 2M×8

* 1. Give the organization of a 2M×32 memory module using 512k×8 static memory chips.
  2. Write a note on i) flash memory. ii) Memory system concepts.
  3. What are the key factors that affect the performance and cost of a computer with respect to memory? Explain Briefly.

|  |  |
| --- | --- |
| *Course Code and Title:* (**16CS35) Computer Organisation** | |
| *Chapter Number and Title*: *8.*  **The Memory Types** | *Planned Hours:* ***04*hrs** |

## Learning Objectives

|  |  |
| --- | --- |
| **Sl. No.** | **Objectives** |
| **1** | Discuss the basics of Caches and cache mapping functions. |
| **2** | Handling of Cache Misses and Handling Writes to caches |
| **3** | Discuss the need for virtual memory. |

|  |
| --- |
| ***Lesson Schedule***  *Class No. Portion covered per hour*   1. Cache memories: Mapping functions, 2. Performance considerations, Hit rate and miss penalty, 3. Caches on the processor chip, other enhancements 4. Introduction to virtual memory. |

**Model Questions**

1. Consider a processor running a program 30% of the instructions of which require a memory read or write operation if the cache bit ratio is 0.95 for instructions and 0.9 for data. When a cache bit occurs for instruction or for data , only one clock is needed while the cache miss penalty is 17 clocks to read/write on the main memory. Work out the time saved by using the cache, given the total number of instructions executed is 1 million.

2. Consider a processor system with 32 bit address capability, using 64KB of cache, arranged to operate as a 4 way set associative cache. Work out the logic which determines cache hit or miss for this system. Assume you have 20-bit comparators available for the purpose.

Is the average access time experienced by the processor an excellent indicator of the effectiveness of a particular implementation of the memory hierarchy? Explain.

3. Explain any two cache mapping functions.

4. What are the key factors that affect the performance and cost of a computer with respect to memory? Explain briefly.

**UNIT- V**

|  |  |
| --- | --- |
| *Course Code and Title:* (**16CS35) Computer Organisation** | |
| *Chapter Number and Title*: *9.*   **Control Unit Logic** | *Planned Hours:* ***04* hrs** |

## Learning Objectives

|  |  |
| --- | --- |
| **Sl. No.** | **Objectives** |
| **1** | Understand Arithmetic and logic operations |
| **2** | Understand single bus, two bus and multi-bus organization |
| **3** | Explain Hardwired control |
| **4** | Discuss Microprogrammed control |

|  |
| --- |
| ***Lesson Schedule***  *Class No. Portion covered per hour*   1. Register Transfers, Performing an Arithmetic or Logic operation (Intro). 2. Performing an Arithmetic or Logic operation. 3. Word in Memory, Fetching a Word from Memory. 4. Execution of a Complete Instruction, Branch instruction. 5. Multiple Bus Organization. 6. Hardwired Control, Micro programmed Control. |

|  |  |
| --- | --- |
| *Course Code and Title:* (**16CS35) Computer Organisation** | |
| *Chapter Number and Title*:10. Number Representation and Arithmetic Operations | *Planned Hours:* ***04* hrs** |

## Learning Objectives

|  |  |
| --- | --- |
| **Sl. No.** | **Objectives** |
| **1** | Discuss various arithmetic operations and various number systems |
| **2** | Solve multiplication using Booth’s Algorithm |
| **3** | Solve for multiplication using Bit-pair Recording of Multipliers; Solve for Division using Integer division(restoring and mon-restoring) |
| **4** | Represent floating point number using IEEE standard |

|  |
| --- |
| ***Lesson Schedule***  *Class No. Portion covered per hour*   1. Numbers, Number Notation, Arithmetic operations and characters 2. Booth Algorithm 3. Fast Multiplication: Bit-pair Recording of Multipliers; Integer division 4. Floating-point Numbers & Operations, IEEE Standard for Floating-point Numbers |

**Model Questions**

1. Explain Big-Endian and Little Endian assignments with examples.
2. What is register transfer notation? Write a RTN to store the difference of two register contents in memory location.
3. Explain the Byte addressability.
4. Explain with illustration, various number representation system for binary. Also suggest the best method for representing in a digital computer.
5. Using register transfer notation, and concept of indirect addressing of memory, show how you can rearrange an ascending sorted data array to a descending sorted array in the same memory locations. Assume the array elements are of 16-bit size, and the processor system is also 16 bit size.

**LESSON PLAN /WEEK**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Week** | **Day** | **Unit** | **Main topic** | **Sub topic** | **No. Hr** | **CO** | **ICT tools Used** | **Book Referred** |
|  | 1 | I | Basic Structures of Computers | Functional units, Basic Operational Concepts, Bus Structures. | 01 | 1 | Power Point Presentation | Carl Hamacher, Z Vranesic& S Zaky, Computer Organization, Mc Graw Hill, 5th Edition, 2011 ISBN 10: 1259005275 / ISBN 13: 9781259005275. |
| 2 | Processor clock, Basic performance equation, Clock rate. | 01 | 2 | Power Point Presentation |
| 3 | Performance measurement- SPEC, CISC and RISC models, Pipelining and Superscalar operation | 01 | 1 | Power Point Presentation |
| 4 | Machine Instructions  and Programs | Memory Locations and Addresses, Memory Operation, Instruction and Instruction Sequencing | 01 | 3 | Power Point Presentation |
|  | 5 | Addressing Modes, implementation of Variables & Constants, Indirection & pointers, | 01 | 4 | Flipped Classroom with TPS |
| 6 | Indexing & Arrays, Relative Addressing, Assembly Language, | 01 | 4 | Flipped Classroom with TPS |
| 7 | Numbers, Number Notation, Arithmetic operations and characters, | 01 | 4 | Power Point Presentation |
| 8 | Example Programs. | 01 | 4 | Flipped Classroom with TPS |
|  | 9 | II | **Machine Instructions**  **and Programs** | Stacks & Queues, Subroutines | 01 | 1 | Power Point Presentation |
| 10 | Subroutine Nesting & Processor Stack, | 01 | 2 | Power Point Presentation |
| 11 | Parameter passing, The stack frame. | 01 | 4 | Power Point Presentation |
| 12 | Additional Instructions, Example programs. | 01 | 4 | Flipped Classroom with TPS |
|  | 13 | **ARM Instruction sets** | Register Memory access and data transfer | 01 | 4 | Power Point Presentation |
| 14 | Arithmetic and logical instructions | 01 | 2 | Power Point Presentation |
| 15 | Branch instructions | 01 | 2 | Power Point Presentation |
| 16 | Assembly language programs. | 01 | 4 | Flipped Classroom with TPS |
|  | 17 |  | Assembly language programs. | 01 | 2 | Flipped Classroom with TPS |
| 18 | III | **Input / Output**  **Organization** | Basic Input / Output Operations, Accessing I/O devices, | 01 | 1 | Power Point Presentation |
| 19 | Interrupts: Interrupt Hardware, Enabling & Disabling Interrupt, | 01 | 3 | Power Point Presentation |
| 20 | Handling Multiple Devices, , Controlling Device Requests. | 01 | 3 | Power Point Presentation |
|  | 21 | Exceptions, Direct Memory Access. | 01 | 2 | Power Point Presentation |
| 22 | **Buses** | Bus Arbitration, | 01 | 1 | Power Point Presentation |
| 23 | Synchronous Bus, | 01 | 1 | Power Point Presentation | Carl Hamacher, Z Vranesic& S Zaky, Computer Organization, Mc Graw Hill, 5th Edition, 2011 ISBN 10: 1259005275 / ISBN 13: 9781259005275. |
| 24 | Asynchronous Bus, | 01 | 2 | Power Point Presentation |
|  | 25 | PCI bus, | 01 | 2 | Power Point Presentation |
| 26 | SCSI bus, USB(Objective & Architecture) | 01 | 3 | Power Point Presentation |
| 27 | IV | **Memory Design** | Semiconductor RAM memories: Internal Organization of Memory Chips, | 01 | 1 | Power Point Presentation |
| 28 | Static Memories, Asynchronous Drams | 01 | 1 | Power Point Presentation |
|  | 29 | Synchronous DRAMs, Structure of Larger Memories, | 01 | 2 | Power Point Presentation |
| 30 | Memory System Considerations, Read-only memory, | 01 | 2 | Power Point Presentation |
| 31 | Speed, Size and cost.  Problems |  |  | Power Point Presentation |
| 32 | **Memory Types** | Cache memories: Mapping functions, performance considerations | 01 | 1 | Power Point Presentation |
|  | 33 | Hit rate and miss penalty, | 01 | 2 | Power Point Presentation |
| 34 | Caches on the processor chip, other enhancements | 01 | 3 | Power Point Presentation |  |
| 35 | Virtual memory. | 01 | 4 | Power Point Presentation |  |
| 36 | V | **Control Unit Logic** | Fundamental Concepts: Register Transfers, Performing an Arithmetic or Logic operation, | 01 | 1 | Power Point Presentation | Carl Hamacher, Z Vranesic& S Zaky, Computer Organization, Mc Graw Hill, 5th Edition, 2011 ISBN 10: 1259005275 / ISBN 13: 9781259005275. |
| 10 | 37 | Fetching a Word from Memory, Storing a Word in Memory, Execution of a Complete Instruction, | 01 | 4 | Power Point Presentation |
| 38 | Branch instruction.  Multiple Bus Organization, | 01 | 4 | Power Point Presentation |
| 39 | Hardwired control,  Basics of Micro programmed control. | 01 | 2 | Power Point Presentation |
| 40 | **Number Representation and Arithmetic Operations** | Booth Algorithm, Fast Multiplication: | 01 | 1 | Flipped Classroom with TPS |
| 11 | 41 | Bit-pair Recording of Multipliers; | 01 | 3 | Flipped Classroom with TPS |
| 42 | Integer division, restoring | 01 | 3 | Flipped Classroom with TPS |
| 43 | Integer division, non restoring | 01 | 3 | Flipped Classroom with TPS |
| 44 | Floating-point Numbers & Operations, IEEE Standard for Floating-point Numbers. | 01 |  | Flipped Classroom with TPS |

# Evaluation Scheme

|  |  |
| --- | --- |
| **Continuous Internal Evaluation (CIE)**  **( Theory – 100 Marks)** | |
| Evaluation method | Course with Self-study |
| Quiz -1 | 10 |
| Test -1 | 25 |
| Quiz -2 | 10 |
| Quiz -3 | 10 |
| Test -2 | 25 |
| Self-study (EL) | 20 |
| **Total** | **100** |

**Experiential Learning**

The course has experiential learning as one of the assessment tool. In this students are assessed based on Two Components

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Sl.No | Component | Description | ICT/Assessment Tool | Max Marks |
| 1 | Workshop | Organization of computer system components. | MOODLE | 10 |
| 2 | Simulators | Executing and analyzing assembly programs | ARM and MIPS | 10 |

* 1. A Workshop on “Organization of computer system components” is organized- After the workshop quiz using MOODLE is conducted and evaluated for 10 marks.
  2. Use of simulators like ARM and MIPS for executing and analyzing assembly programs.

Students are assigned different assembly programs to execute on the simulators and are required to demonstrate and justify the updates in memory and registers after program execution.

**SAMPLE PROGRAMS:**

* + 1. Write an Assembly Language Program for computing the average scores on each test and store these sums in the memory word locations at address SUM, SUM+4, SUM+8.
    2. Register R1 and R2 of a computer contain the decimal values 1200 and 4600. Find the effective address of the memory operand in each of the following instruction and also indicate the contents of all the registers used

STORE R5,30(R1,R2)

ADD –(R2),R5

SUB (R1)+,R5

MOVE #3000, R5

LOAD 20(R1),R5

|  |  |  |  |
| --- | --- | --- | --- |
| **Components** | **Rubrics for program execution(SIMULATOR)** | | |
|  | **Excellent** | **Good** | **Not Satisfactory** |
| **Problem definition(2)** | Exhibit Clear understanding of the program with proper justification to the addressing modes used.  2marks | Exhibit understanding of the program without proper justification to the addressing modes used.  <=1 marks | Not Clear about the program- 0 Marks |
| **Implementation(4)** | Demonstrate the working program by satisfying all the inputs given – 4 marks | Implementation satisfying a few input condition specified < 3 marks | Not implemented- 0 marks |
| **Register/Memory mapping(4)** | Able to analyze and discuss both memory and register mapping effectively through use of simulators –4marks | Able to analyze and discuss either memory or register mapping effectively through use of simulators 2-3 marks | Not clear about register/memory mapping but demonstrates usage of simulator– 0-1marks |

|  |  |
| --- | --- |
| **Semester End Evaluation Theory (100)** | |
| **Part- –A**  **Objective type questions** | **20** |
| **Part –B**  There should be five questions from five units. Each question should be for maximum of 16 Marks.  The **UNIT-1**, **UNIT-4** and **UNIT-5** should not have any choice.  The **UNIT-2 and UNIT-3** should have an internal choice.  Both the questions should be of the same complexity in terms of COs and Bloom’s taxonomy level. | **80** |
|
|
| **Total** | **100** |

## Course Unitization for Internals and Semester End Examination

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Part** | **Chapter** | | **Teaching Hours** | **No. of Questions in** | | **No. of Questions in SEE** |
| **Internals I** | **Internals II** |
| **Unit 1** | 1 | **Basic Structures of Computers** | 3 | 1 | -- | 1 |
| 2 | **Machine Instructions and Programs** | 5 | 1 | -- |
| **Unit 2** | 3 | **Machine Instructions and Programs** | 4 | 1 | -- | 1 |
| 4 | **ARM Instruction sets** | 5 | 1 | -- | 1 |
| **Unit 3** | 5 | **Input / Output Organization** | 4 | 1 | -- | 1 |
| 6 | **Buses** | 5 | -- | 1 | 1 |
| **Unit 4** | 7 | **Memory Design** | 5 | -- | 1 | 1 |
| 8 | **Memory Types** | 4 | -- | 1 |
| **Unit 5** | 9 | **Control Unit Logic** | 5 | -- | 1 | 1 |
| 10 | **Number Representation and Arithmetic Operations** | 4 | -- | 1 |

**Faculties In-charge Head of Department**

**Course Articulation Matrix (16CS35)**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | **CO-PO Mapping** | | | | | | | | | | | | | | | **CO/PO** | **PO1** | **PO2** | **PO3** | **PO4** | **PO5** | **PO6** | **PO7** | **PO8** | **PO9** | **PO10** | **PO11** | **PO12** | | **CO1** | **L** | **L** | **-** | **-** | **L** | **-** | **-** | **-** | **-** | **-** | **-** | **L** | | **CO2** | **M** | **M** | **-** | **-** | **L** | **-** | **-** | **-** | **M** | **M** | **-** | **L** | | **CO3** | **M** | **-** | **L** | **-** | **-** | **-** | **-** | **-** | **M** | **M** | **-** | **L** | | **CO4** | **-** | **L** | **M** | **-** | **-** | **-** | **L** | **-** | **M** | **L** | **-** | **L** | |
| **Program Articulation Matrix**   |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | **Course - PO Mapping** | | | | | | | | | | | | | | |  | **PO1** | **PO2** | **PO3** | **PO4** | **PO5** | **PO6** | **PO7** | **PO8** | **PO9** | **PO10** | **PO11** | **PO12** | | **Course** | **L** | **L** | **L** | **-** | **L** | **-** | **L** | **-** | **M** | **L** | **-** | **L** | |
| |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **CO –PSO Mapping** | | |  | **Course – PSO Mapping** | | | | **CO/PSO** | **PSO1** | **PSO2** |  | **Course** | **PSO1** | **PSO2** | | **CO1** | **-** | **-** |  | **M** | **L** | | **CO2** | **L** | **-** |  |  |  |  | | **CO3** | **M** | **L** |  |  |  |  | | **CO4** | **H** | **L** |  |  |  |  |   **CO-PSO Mapping COURSE-PSO Mapping** |

**Flipped Class Room Activity**

**Out\_of\_class Activity Design-1:**

**After watching the video student should be able to know**

1. **Booth’s Algorithm**
2. **Multiply two signed integers using the above algorithm**
3. **Check the resultant with general multiplication.**

**Key Concept(s) to be covered:**

1. Signed Integer
2. Booth’s algorithm

**Out-of-class Activity Design – 2**

Uploaded Video URL: https://www.youtube.com/watch?v=MeD8iWb8wlo

License of Video: Creative common attribution License

Duration of video: V1-8.55 MIN

**Out-of-class Activity Design – 3**

**Aligning Assessment with Learning Objective**

|  |  |  |  |
| --- | --- | --- | --- |
| **Learning Objective** | **Assessment Strategy** | **Expected duration**  **(in min)** | **Additional Instructions (if any)** |
| **Booth’s Algorithm** | **1.Draw a flowchart for booth’s Algorithm** | **5 min** | **WATCH VIDEO V1** |
| **Multiply two integer using booth’s Algorithm** | **2.Multiply given integers**  **i). 3 X 5**  **ii).11 X 13** | **10 min** | **WATCH VIDEO V1** |

**In\_Class Activity: (Active Learning Strategy)**

**Think-Pair-Share (TPS) Activity:**

Domain: Computer Organization

Topic: Multiplication of signed integers using Booth’s Algorithm

Target Students: BE (CSE) 3rd Sem

Think Phase -              [3 minutes]

Question: **Multiply -6 X 7**

What Teacher does - Poses the question, asks students to think individually and write the binary equivalent of given decimal numbers.

What students do - **Thinks individually and Write down binary equivalent of the given numbers.**

Pair Phase -                [7 minutes]

Question: Discuss your answer with your neighbor and multiply the two binary numbers using booth’s algorithm.

What Teacher does - Poses the question, asks students to pair up and discuss, goes around the class to check whether students are discussing, and provides clues to pairs who are in doubt.

What students does - Pairs up with neighbor, Checks each other’s result by converting the multiplication value into 2’s complement form.

Share Phase -              [5 minutes]

What Student Does – Shares the result with whole class.

What Teacher Does - **Notes down the correct answer in the board, summarizes the key concepts involved in this problem.**

**Flipped Class Room Activity Incharges:**

|  |  |  |
| --- | --- | --- |
| **Sl No** | **Topic** | **Faculty** |
| 1 | Addressing Modes | https://drive.google.com/open?id=0B1En1h79yWkxWXd4S1JsUExBR3RUcGxwMVdoTnpHV1JGRHRJ |
| 2 | Integer Multiplication | https://drive.google.com/open?id=0B7OK4xl6r47daWxBcWIyN3UycU0 |
| 3 | Integer Division | https://drive.google.com/open?id=0B7OK4xl6r47deF81NHZpR2dZV3M |