

GANPAT UNIVERSITY				
FACULTY OF DIPLOMA ENGINEERING				
Programme	Diploma in Computer Engineering/ Information Technology			
Semester	III	Version	1.0.0.0	
Effective from Academic Year	2025-26	Effective for the batch Admitted in	JULY 2025	
Course Code	1CEIT3105	Course Name	Computer Architecture and Microprocessor Systems	

I. TEACHING-LEARNING AND ASSESSMENT SCHEME

Course Type	Course code	Course Title	Teaching & Learning Scheme									Examination Scheme							
			Credit				Actual Contact Hrs/week			SLH	Total Learning Hrs/Week	TH			PR			SLA	Total
			CL	TL	LL	Total	CL	TL	LL			CE	SEE	Total	CE	SEE	Total		
DSC	1CEIT3105	COMPUTER ARCHITECTURE & MICROPROCESSOR	4	-	1	5	4	-	2	2	8	40	60	100	30	20	50	20	170

Abbreviation:	CL- Classroom Learning	TL - Tutorial Learning	CL- Classroom Learning
	SLH - Self Learning Hours	SLA - Self Learning Assessment	SLH - Self Learning Hours
	SEE – Semester End Examination		

II. PRE-REQUISITES

Basic knowledge of digital electronics and computer fundamentals is required.

III. INDUSTRY /EMPLOYER EXPECTED OUTCOMES

- Understand the organization and operation of computer systems at the architectural level.
- Interpret instruction formats, instruction cycles, and control mechanisms.
- Analyze the architecture and interrupt system of microprocessors.
- Develop and execute assembly language programs for basic data processing tasks.
- Apply microprocessor concepts in simple embedded and control applications.
- Use trainer kits and simulators effectively for testing and debugging.

IV. COURSE LEARNING OUTCOMES

At the end of the course, students will be able to achieve the following course learning outcomes:

CO1. Understand the basic organization of a computer system and analyze register transfer operations and micro-operations involved in data processing.

CO2. Interpret instruction formats and illustrate the instruction cycle, including memory, I/O, and interrupt operations in a computer system.

CO3. Describe the internal architecture, bus organization, and interrupt structure of the 8085 microprocessor with proper signal and pin-level understanding.

CO4. Classify and analyze the 8085-instruction set, addressing modes, machine cycles, and timing behaviour during instruction execution.

CO5. Develop and test assembly language programs using the 8085-instruction set to perform data transfer, arithmetic, logical, branching, and delay operations.

V. THEORY LEARNING OUTCOMES AND ALIGNED COURSE CONTENT:

Name of Unit	Theory Learning outcomes (TLO's) aligned to CO's	Learning Content mapped with Theory Learning outcomes (TLO's) & CO's	Marks	Hours
Unit-1 Fundamentals of Computer Architecture	TLO 1.1 Identify the organization and functional units of a computer system	1.1 The organization and functional units of a computer system	10	10

<p>and Micro-Operations</p>	<p>TLO 1.2 Describe data flow between input unit, output unit, memory, and CPU TLO 1.3 Define register transfer concepts and symbolic notation TLO 1.4 Interpret Register Transfer Language (RTL) representation TLO 1.5 Justify the need for common bus systems TLO 1.6 Describe memory read and memory write transfer operations TLO 1.7 Perform arithmetic micro-operations like addition, subtraction TLO 1.8 Perform logic micro-operations: AND, OR, XOR, NOT TLO 1.9 Perform shift micro-operations: logical, arithmetic, circular TLO 1.10 Describe the function of the Arithmetic Logic Shift Unit</p>	<p>1.2 Data flow between CPU, memory, and I/O units 1.3 Register transfer operations and notation 1.4 Register Transfer Language (RTL) statements 1.5 Common bus transfer systems 1.6 Memory read and write operations 1.7 Arithmetic micro-operations 1.8 Logic micro-operations 1.9 Shift micro-operations 1.10 Arithmetic Logic Shift Unit (ALSU)</p>		
<p>Unit-2 Computer Instruction Format and Instruction Cycle</p>	<p>TLO 2.1 Identify computer registers and describe their role in instruction execution TLO 2.2 Describe instruction format and instruction fields. TLO 2.3 Classify computer instructions based on operation and function TLO 2.4 Describe the role of the timing and control unit TLO 2.5 Describe generation of control signals for instruction execution TLO 2.6 Illustrate the phases of the instruction cycle TLO 2.7 Describe memory reference instructions and their operation TLO 2.8 Interpret Input–Output reference instructions TLO 2.9 Describe the interrupt cycle and sequence of operations TLO 2.10 Draw and illustrate flowcharts of instruction cycle and interrupt cycle</p>	<p>2.1 Computer registers and their role in instruction execution 2.2 Describe instruction format and instruction fields such as opcode, operand, and address. 2.3 Types of computer instructions 2.4 Timing and control unit 2.5 Control signal generation 2.6 Instruction cycle phases: fetch, decode, and execute 2.7 Memory reference instructions 2.8 Input–Output reference instructions 2.9 Interrupt cycle and sequence of operations 2.10 Flowchart representation of instruction and interrupt cycle</p>	<p>10</p>	<p>10</p>
<p>Unit-3 8085 Micro-processor Architecture and Interrupt System</p>	<p>TLO 3.1 Describe the evolution and key features of microprocessors with emphasis on 8085 TLO 3.2 Describe the bus organization of 8085 microprocessor and functions of address, data, and control buses TLO 3.3 Illustrate de-multiplexing of address/data bus using ALE pin</p>	<p>3.1 Overview and features of microprocessors with reference to 8085 3.2 8085 bus organization and signal overview 3.3 De-multiplexing of address and data bus 3.4 Control signal generation in 8085</p>	<p>14</p>	<p>14</p>

	<p>TLO 3.4 Describe the generation of read, write, and control signals in 8085</p> <p>TLO 3.5 Describe the internal block diagram and functional units of the 8085 microprocessor</p> <p>TLO 3.6 Describe register organization and the function of the flag register</p> <p>TLO 3.7 Describe the timing and control unit, serial input/output, and interrupt control</p> <p>TLO 3.8 Identify and classify 8085 pins based on their functions</p> <p>TLO 3.9 Distinguish between hardware and software interrupts</p> <p>TLO 3.10 Describe interrupt priority levels and vector addresses</p>	<p>3.5 Internal architecture and functional units of the 8085 microprocessor</p> <p>3.6 Register organization and flag register</p> <p>3.7 Timing and control, serial I/O, and interrupt control</p> <p>3.8 Pin configuration and functional classification of 8085 pins</p> <p>3.9 Hardware and software interrupts in 8085</p> <p>3.10 Interrupt priority structure and vector locations</p>		
<p>Unit-4</p> <p>8085 Instruction Set Architecture</p>	<p>TLO 4.1 Describe the instruction format and various addressing modes used in 8085</p> <p>TLO 4.2 Classify the 8085 instruction set based on functionality</p> <p>TLO 4.3 Describe data transfer instructions and their operation</p> <p>TLO 4.4 Analyze arithmetic instructions and their effect on flags</p> <p>TLO 4.5 Describe logical instructions and rotate operations</p> <p>TLO 4.6 Illustrate branching instructions such as jump, call, and return</p> <p>TLO 4.7 Describe stack organization and subroutine call/return mechanism</p> <p>TLO 4.8 Interpret program control operations in the 8085</p> <p>TLO 4.9 Describe machine control instructions and their purpose</p> <p>TLO 4.10 Analyze the role of machine cycles and T-states</p>	<p>4.1 The instruction format and addressing modes of 8085</p> <p>4.2 Classification of the 8085 instruction set</p> <p>4.3 Data transfer instructions</p> <p>4.4 Arithmetic instructions</p> <p>4.5 Logical instructions and rotate operations</p> <p>4.6 Branch instructions: jump, call, and return</p> <p>4.7 Stack organization and subroutine handling</p> <p>4.8 Program control operations</p> <p>4.9 Machine control instructions</p> <p>4.10 Machine cycles and T-states</p>	12	12
<p>Unit-5</p> <p>Assembly Language Programming Techniques</p>	<p>TLO 5.1 Describe the format and structure of an 8085 assembly language program</p> <p>TLO 5.2 Identify commonly used assembler directives and comments</p> <p>TLO 5.3 Develop data transfer programs using registers and memory</p> <p>TLO 5.4 Develop arithmetic programs such as addition and subtraction</p> <p>TLO 5.5 Develop logical programs using AND, OR, XOR, and complement</p>	<p>5.1 The structure of assembly language program</p> <p>5.2 Assembler directives and their functions</p> <p>5.3 Data transfer programs</p> <p>5.4 Arithmetic operation programs</p> <p>5.5 Logical operation programs</p> <p>5.6 Rotate instruction programs</p> <p>5.7 Branching and looping programs</p>	14	14

	<p>TLO 5.6 Develop rotate instruction programs</p> <p>TLO 5.7 Develop branching and looping programs using jump instructions</p> <p>TLO 5.8 Develop subroutine programs using stack operations</p> <p>TLO 5.9 Design delay programs using loops and counters</p> <p>TLO 5.10 Identify the role of microprocessors in embedded and control applications</p>	<p>5.8 Stack-based subroutine programs</p> <p>5.9 Delay subroutine programs</p> <p>5.10 Applications of microprocessors in embedded and control systems</p>		
--	---	--	--	--

VI. LABORATORY LEARNING OUTCOME AND ALIGNED PRACTICAL			
SR. NO	PRACTICAL/LABORATORY LEARNING OUTCOME (LLO)	PRACTICAL TITLES	RELEVANT COs
1	LLO 1.1 Identify the functional units of a computer system and interpret data flow between CPU, memory, and I/O.	Understand the basic organization and functional units of a computer system and data flow between input, output, memory, and CPU	CO1
2	LLO 1.2 Demonstrate arithmetic, logic, and shift micro-operations using appropriate examples.	Demonstration of arithmetic, logic, and shift micro-operations using examples	CO1
3	LLO 2.1 Identify instruction format, instruction fields, and functions of the timing and control unit.	Study of instruction format, instruction fields, and functions of the timing and control unit	CO2
4	LLO 2.2 Illustrate instruction cycle phases using flowchart representation.	Study and flowchart representation of instruction cycle phases: fetch, decode, and execute	CO2
5	LLO 3.1 Identify and describe the internal architectural blocks of the 8085 microprocessor.	Study of internal architecture of the 8085-microprocessor using block diagram or simulator	CO3
6	LLO 3.2 Interpret register organization and flag register operations of the 8085 microprocessor.	Study of register organization and flag register of the 8085 microprocessors	CO3
7	LLO 3.3 Distinguish between hardware and software interrupts and analyze interrupt priority and vector addresses.	Study of hardware and software interrupts in 8085, including interrupt priority and vector addresses	CO3
8	LLO 4.1 Classification of 8085 microprocessor instruction set.	Understand the classification of 8085 microprocessor instruction set	CO4
9	LLO 5.1 Develop and execute data transfer programs using 8085 registers and memory locations.	Write and execute an 8085 program to perform data transfer operations using registers and memory	CO5
10	LLO 5.2 Develop and execute arithmetic programs using addition and subtraction instructions of 8085.	Write and execute an 8085 program to perform arithmetic operations (addition and subtraction)	CO5
11	LLO 5.3 Develop and execute logical operation programs using AND, OR, XOR, and complement instructions.	Write and execute an 8085 program to perform logical operations (AND, OR, XOR, complement)	CO5

12	LLO 5.4 Develop and execute programs using rotate instructions of the 8085 microprocessors.	Write and execute 8085 programs using rotate instructions	CO5
13	LLO 5.5 Develop and execute programs using branching and looping instructions.	Write and execute 8085 programs using branching and looping instructions	CO5
14	LLO 5.6 Develop and execute programs using stack operations and subroutine call and return mechanism.	Write and execute an 8085 program to implement stack operations and subroutine call and return	CO5
15	LLO 5.7 Design and execute delay generation programs using delay subroutines.	Write and execute an 8085 program to generate a time delay using a delay subroutine	CO5

VII. SUGGESTED MICRO PROJECT/ASSIGNMENTS/ACTIVITIES FOR SELF LEARNING/SKILL DEVELOPMENT (SELF LEARNING)

Micro Projects (Mini Applications / Use Cases)

- Develop a program to transfer data between registers and memory locations
Skills: Data transfer instructions, addressing modes, memory access
- Develop a program that performs arithmetic and logical operations
Skills: Instruction set usage, branching, looping, program control
- Design a delay subroutine program and analyze the time delay using machine cycles and T-states
Skills: Loop control, timing analysis, delay subroutine design
- Develop a program using subroutines to perform addition, subtraction, and logical operations
Skills: Stack operations, subroutine call and return, modular programming
- Study a real-world embedded application and present its block diagram and working
Skills: System-level thinking, application analysis, technical communication

Self-Learning / Skill Building Activities

1. Practice 8085 assembly programs using online simulators or software tools.
Goal: Improve programming accuracy and execution understanding.
2. Prepare a categorized list of 8085 instructions with examples and flag effects.
Goal: Strengthen instruction-level understanding.
3. Calculate execution time for given 8085 programs using machine cycles and T-states.
Goal: Develop analytical skills related to processor timing.
4. Draw flowcharts for arithmetic, logical, and branching programs before coding.
Goal: Enhance logical thinking and structured problem solving.
5. Observe and document changes in registers and flags during program execution.
Goal: Understand processor behavior at register level.
6. Compare features of 8085 with a modern microcontroller (e.g., 8051 or Arduino).
Goal: Build perspective on processor evolution.
7. Identify and correct common errors in assembly language programs.
Goal: Improve debugging and troubleshooting skills.
8. Prepare a short report on interrupt-based processing with examples.
Goal: Understand real-time event handling.
9. Watch lectures on computer architecture or microprocessors and submit a summary.
Goal: Reinforce theoretical concepts through visual learning.
10. Create a poster or presentation on topics such as microprocessor architecture.
Goal: Improve technical communication and conceptual clarity.

VIII. LIST OF INSTRUMENTS / EQUIPMENT / TRAINER BOARD	
1	8085 Microprocessor Trainer Kit: Intel 8085 Microprocessor, 3 MHz Clock Frequency, On-board RAM and EPROM, Hex keypad and LED/LCD display, Support for assembly program execution, Provision for interfacing experiments.
2	Desktop Computer System: Intel Core i3 Processor, 4 GB RAM, 256 GB SSD / HDD Storage, Windows 10 OS
3	8085 Microprocessor Simulator Software: <ul style="list-style-type: none"> • https://www.sim8085.com • https://www.oshonsoft.com/8085.html
4	Assembler and Debugging Tools: <ul style="list-style-type: none"> • GNUSim8085: https://gnusim8085.sourceforge.net/ • Online 8085 Simulator: https://www.tutorialspoint.com/compile_assembly_online.php

IX. LIST OF REFERENCE BOOKS			
Sr. No	Title	Author	Publication
1	Computer System Architecture	M. Morris Mano	Pearson Education
2	Computer Organization and Architecture	William Stallings	Pearson Education
3	Computer Organization and Architecture	H. A. Momaya	Atul Prakashan
4	Microprocessor: Architecture, Programming, and Applications with the 8085	Ramesh Gaonkar	Penram International
5	The 8085 Microprocessor: Architecture, Programming, and Interfacing	K. Udaya Kumar, B. S. Umashankar	Pearson Education
6	8085 Microprocessor Programming and Interfacing	N. K. Srinath	Prentice Hall of India (PHI)

X. LINK OF LEARNING WEB RESOURCE	
1	NPTEL – Computer Organization and Architecture: https://nptel.ac.in/courses/106103068
2	NPTEL – Microprocessors and Interfacing: https://nptel.ac.in/courses/117106092
3	Computer Architecture: https://www.youtube.com/watch?v=DsK35f8wyUw
4	Microprocessor: https://www.youtube.com/playlist?list=PLxCzCOWd7aiHL7mF_dRsj4Q9x1NNaZqkh

XI. SUGGESTED WEIGHTAGE TO LEARNING EFFORTS & ASSESSMENT PURPOSE							
Unit	Unit Title	Aligned COs	Learning Hours	R-Level	U-Level	A-Level	Total Marks
1	Fundamentals of Computer Architecture and Micro-Operations	CO1	10	4	4	2	10
2	Computer Instruction Format and Instruction Cycle	CO2	10	3	5	2	10
3	8085 Microprocessor Architecture and Interrupt System	CO3	14	4	6	4	14
4	8085 Instruction Set Architecture	CO4	12	3	5	4	12

5	Assembly Language Programming Techniques	CO5	14	2	4	8	14
Grand Total			60	16	24	20	60

XIII. COs AND POs AND PSOs MAPPING										
Course Outcome (Cos)	Programme Outcomes (POs)							Programme Specific Outcomes (PSOs)		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PSO1	PSO2	PSO3
CO1	3	2	1	1	0	1	0	1	1	0
CO2	3	2	1	1	0	1	0	1	1	0
CO3	3	2	2	2	0	1	0	1	2	0
CO4	3	3	2	1	0	1	0	2	1	0
CO5	2	3	3	2	1	1	1	3	1	2

Legends: - 3-High; 2-Moderate/Medium; 1-Slight/Low; 0-None