

NIRMA UNIVERSITY

Institute:	Institute of Technology
Name of Programme:	BTech CSE, Integrated BTech (CSE)-MBA
Course Code:	XXXX
Course Title:	Microprocessor and Interfacing
Course Type:	Core
Year of Introduction:	2025-26

L	T	Practical Component				C
		LPW	PW	W	S	
3	0	2	-	-	-	4

Course Learning Outcomes (CLO):

At the end of the course, the students will be able to:

1. explain the basic architecture of microprocessors (BL2)
2. apply the fundamental knowledge of assembly language programming to implement the basic functions (BL3)
3. analyse modular programs for efficient and maintainable code development. (BL4)
4. design microprocessor-based systems for interfacing peripherals. (BL6)

Unit	Contents	Teaching Hours (Total 45)
Unit-I	Microprocessor Architecture: architectural details of 8085, 8086, and 80x86 processors, pin functions, read/write machine cycles, memory organization (linear, segmentation, paging), interrupts.	12
Unit-II	Microprocessor Programming: Assembler instruction format, data transfer, Arithmetic, Branch, loop, NOP, Call, Flag manipulations, logical shifts, rotate instructions, directives, and operators, assembly process, translation of assembly instructions	10
Unit-III	Modular Programming: Linking and relocation, stacks, procedures, interrupt routines, macros, program design examples	12
Unit-IV	Interfacing: Architecture and interfacing of Intel 8255 Programmable Peripheral Interface, Intel 8259 Programmable Interrupt Controller, Intel 8254 programmable timer, 8237 DMA Controller, Analog to Digital Converter, and Digital to Analog Converter. Microcontrollers, interfacing, and industrial applications in process control	11

Self-Study:

The self-study contents will be declared at the commencement of the semester. Around 10% of the questions will be asked from self-study content

Suggested Readings/ References:

1. Yu-Cheng Liu, 8086/8088 family, PHI
2. Barry B. Brey, The Intel microprocessors, Prentice Hall



3. Douglas V Hall, Microprocessors and Interfacing: Programming and Hardware, McGraw-Hill
4. Ramesh S. Gaonkar, Microprocessor Architecture, Programming, and Applications with the 8085, Prentice Hall
5. Peter Abel, IBM PC Assembly language and programming, Pearson

Laboratory Work:

Laboratory work will be based on the above syllabus with a minimum of 10 experiments to be incorporated. The students in a suitable group size will design and perform one experiment as a part of Laboratory work.

Sr. No.	List of Experiments/Exercises	Hours
1	a) Introduction to debugging. (Prepare a report on debugging with all its commands)	02
2	b) Introduction to programming with assembler, with an example of the addition of two numbers. Exploring the usage of debug commands. Perform binary arithmetic operations on two 16-bit numbers and two 32-bit numbers. Write a program using C' to perform inline assembly	02
3	Find the maximum and minimum of numbers in an array.	02
4	a) Convert an octal number to a binary number. b) Convert a lowercase string to uppercase and vice versa i/p AMP Program o/p AMP Program	04
5	Arrange numbers in ascending order.	02
6	a) Add ten numbers and find their average. b) Find the number of 1's in the given 8-bit number and determine its parity. c) Convert string to integer and vice versa using the far procedure with stack parameter passing.	04
7	Reverse a string entered by the user using the near procedure.	02
8	a) Find the occurrences of a character from the entered string. b) Write a program to find a substring from a given string using the near procedure. Take the string and the substring from the user.	04
9	Generate the Fibonacci series using the far procedure and display the series.	04
10	Check whether the number is prime, odd, even, etc, using far procedures in the same assembly module and in different assembly modules c) Implement the copy command for copying one file to another. d) Study of the 8254 timer chip and its interfacing	04

