Q. Code:828876

Reg. No. | | | | | | | | | | | | | |

B.E. / B.TECH. DEGREE EXAMINATIONS, MAY 2024

Fourth Semester

EC18405–MICROPROCESSOR AND MICROCONTROLLER BASED SYSTEM DESIGN

(*Electronics and Communication Engineering)*

(Regulation2018/ 2018A)

TIME: 3 HOURS **MAX. MARKS: 100**

COURSE OUTCOMES	STATEMENT	RBT LEVEL
CO 1	Develop programs in 8086 microprocessors by understanding its architecture, instruction set and interrupt process.	**4**
CO 2	Sketch the system bus structure of 8086 and multiprocessor configurations.	**3**
CO 3	Design I/O and Memory interfacing units.	**4**
CO 4	Develop programs in 8051 microcontrollers by understanding its architecture and instruction set.	**4**
CO 5	Design various interfacing units with 8051 microcontroller-based systems.	**4**

PART- A (10 x 2 = 20 Marks)

(Answer all Questions)

		CO	RBT LEVEL
1.	Define Addressing Modes in microprocessors. Provide examples of different addressing modes.	**1**	**3**
2.	How do interrupts enhance the functionality of microprocessors?	**1**	**3**
3.	Define multiprogramming and discuss its importance in enhancing system efficiency.	**2**	**3**
4.	Explain the function and importance of a coprocessor in 8086 microprocessor-based systems.	**2**	**2**
5.	Compare and contrast Serial communication over Parallel communication Interface.	**3**	**3**
6.	List the significance of Interfacing an Analog to Digital Converter in a microprocessor system.	**3**	**2**
7.	On Power up of 8051 microcontrollers, Specify the location of the first stack and which of the register banks is used on power up?	**4**	**4**
8.	Discuss the priority levels of interrupts in the 8051 microcontroller.	**4**	**3**
9.	Compare LCD and LED in terms of its application in embedded system design.	**5**	**3**
10.	Justify the importance of I^2C protocol in multimaster bus communication.	**5**	**3**

PART- B (5 x 14 = 70 Marks)

			Marks	CO	RBT LEVEL
11. (a)		Discuss the architecture of the 8086 microprocessor in detail.	**(14)**	**1**	**3**
		(OR)			
(b)	**(i)**	Describe the following 8086 instructions with suitable examples. a, SUB b, DIV c, PUSH	**(7)**	**1**	**3**
	(ii)	Explain the various string manipulation instructions with suitable examples.	**(7)**	**1**	**3**
12. (a)		Tabulate the Minimum mode signals. Explain the functions of all the signals and how the signals are generated with the help of various functional units.	**(14)**	**2**	**3**
		(OR)			
(b)	**(i)**	Compare closely coupled and loosely coupled multiprocessor configurations.	**(7)**	**2**	**3**
	(ii)	Discuss in detail the three bus arbitration techniques (i)Daisy chaining (ii)Independent Priority (iii)Polling method.	**(7)**	**2**	**3**
13. (a)		Point out the features and explain the operation of 8255 Parallel Communication Interface and explain the various modes of operation.	**(14)**	**3**	**2**
		(OR)			
(b)		Point out the features and explain the operation of 8251 Serial Communication Interface and explain the various modes of operation.	**(14)**	**3**	**2**
14. (a)		Illustrate the architectural features of 8051 microcontroller with necessary diagram.	**(14)**	**4**	**3**
		(OR)			
(b)		Provide examples of instructions utilizing each addressing mode and explain how the addressing modes contribute to efficient memory access and data manipulation in assembly language programming.	**(14)**	**4**	**3**
15. (a)		Explain in detail how a stepper motor can be interfaced with 8051 and write a suitable assembly language program for the following condition. Assume a switch is connected at pin P1.5, rotate the stepper motor in counterclockwise direction when the switch is closed and in clockwise direction when the switch is open.	**(14)**	**5**	**4**
		(OR)			
(b)		Analyze in detail about RTC chip DS12877 interfacing with 8051 microcontroller, write a suitable program to set the Time for an Embedded application.	**(14)**	**5**	**4**

PART- C (1 x 10 = 10 Marks)

(Q.No.16 is compulsory)

			Marks	CO	RBT

			LEVEL	
16.	Construct with neat diagram, a 7x5 dot matrix LED to display "E" by interfacing it with 8051 microcontroller. Develop a suitable Assembly language code and comment on it.	**(10)**	**5**	**5**

****