COURSE TITLE	**MICROCONTROLLER PROGRAMMING AND INTERFACING**
COURSE CODE	**01EC0308**
COURSE CREDITS	**4**

Objective:

1 To develop a strong foundation in microcontroller architecture, instruction sets, and embedded programming using AVR and PIC families. The course enables students to design, program, and interface on-chip peripherals for real-world embedded applications while understanding processor selection criteria and architectural trade-offs between RISC and CISC systems.

Course Outcomes: After completion of this course, student will be able to:

1 Explain differences between microprocessors and microcontrollers, including RISC vs CISC architectures and selection criteria.

2 Analyze AVR architecture and develop programs using its instruction set.

3 Interface and program AVR on-chip peripherals such as timers, ADC, UART, and GPIO.

4 Program PIC microcontrollers by configuring its SFRs and peripherals.

Pre-requisite of course: Knowledge of Electronic Devices and Circuits, Fundamental knowledge of Digital Circuits.

Teaching and Examination Scheme

Theory Hours	Tutorial Hours	Practical Hours	ESE	IA	CSE	Viva	Term Work
3	0	2	50	30	20	25	25

Contents : Unit	Topics	Contact Hours
1	**Introduction to Microprocessors and Microcontrollers** Introduction to Microprocessors and Microcontrollers – Comparison (Microprocessor vs Microcontroller), Embedded Systems overview, and Applications of Microcontrollers, Architecture of Microcontrollers – Block Diagram and Basic Organization	2
2	**Architecture and Design Considerations** Processor Architectures – RISC vs CISC (features, comparison, examples) and Harvard vs Von Neumann architecture, Microcontroller Selection Criteria – Cost, power consumption, performance, peripherals, and ecosystem, Overview of Popular Microcontroller Families – AVR, PIC, ARM and their key features	3

DR. RAJENDRAKUMAR DAHYALAL PATEL
Digitally signed by (Name of HOD)

DR. RAJENDRASINH BAHADURSINH JADEJA
Digitally signed by (Name of Dean/ Principal)

Contents : Unit	Topics	Contact Hours
3	**AVR Architecture and Instruction Set** Introduction to AVR Architecture, AVR Register File Organization, Arithmetic Logic Unit (ALU) and Status Register, Internal Memory Organization – Overview of Flash, SRAM, and EEPROM, Flash Memory and Program Storage Details, SRAM and EEPROM Organization and Usage, Addressing Modes in AVR Microcontrollers, Instruction Set Overview – Data transfer, arithmetic, logical, and control instructions, Assembly Language Programming Basics, Assembly Programming Examples and Applications	10
4	**AVR On-Chip Peripherals and Programming** Introduction to GPIO Programming, GPIO Configuration and Interfacing Techniques, Timers and Counters – Basic Concepts, Timer/Counter Modes and Applications, Interrupts in Microcontrollers, Watchdog Timer and Its Applications, Power Down Modes and Power Management, ADC Interfacing and Data Acquisition, UART Communication and Serial Interface, PWM Generation and Applications	10
5	**PIC Architecture and Instruction Set** Introduction to PIC Microcontroller Architecture, Pipeline Concept in PIC, Memory Organization in PIC Microcontrollers, Memory Banking and Its Operation, Working Register (WREG) and Special Function Registers (SFRs), Addressing Modes in PIC, Instruction Set Overview, Programming Model of PIC Microcontrollers, Basic Programming Examples and Applications	9
6	**PIC Peripheral Programming and Interfacing** GPIO Configuration and Digital I/O Programming, Timers – Basic Concepts and Configuration, Capture/Compare/PWM (CCP) Module Overview, Timer and CCP Applications, Interrupt System in Microcontrollers, UART Communication and Serial Interfacing, ADC Programming and Analog Interfacing, PWM Generation and Control Applications	8
Total Hours		**42**

Suggested List of Experiments:

Contents : Unit	Topics	Contact Hours
1	**Experiment No. 1** GPIO LED Blinking - Write an assembly language program to configure a GPIO pin of an AVR microcontroller as output and generate a continuous LED blinking pattern with a fixed delay.	2
2	**Experiment No. 2** Switch Interfacing with LED Control - Write an assembly language program to interface a push button switch and an LED with an AVR microcontroller such that the LED turns ON when the switch is pressed and OFF when released.	2

DR. RAJENDRAKUMAR DAHYALAL PATEL
Digitally signed by (Name of HOD)

DR. RAJENDRASINH BAHADURSINH JADEJA
Digitally signed by (Name of Dean/ Principal)

Suggested List of Experiments:

Contents : Unit	Topics	Contact Hours
3	**Experiment No. 3** Timer-Based Delay Generation - Write an assembly/C program to generate precise time delays using an AVR timer (Timer0/Timer1) instead of software delay loops, and toggle an LED periodically.	2
4	**Experiment No. 4** PWM-Based LED Dimming - Write an assembly/C program to configure PWM mode in an AVR microcontroller and control the brightness of an LED by varying the duty cycle.	2
5	**Experiment No. 5** UART Communication (Transmit/Receive) - Write an assembly/C program to initialize UART in an AVR microcontroller and transmit a string to a serial terminal, then receive data and display it on LEDs or echo back.	2
6	**Experiment No. 6** ADC Interfacing with Sensor (LM35) - Write an assembly/C program to interface an LM35 temperature sensor with the ADC of an AVR microcontroller and display thej temperature value via UART.	2
7	**Experiment No. 7** Interrupt-Based Button Handling - Write an assembly/C program to configure an external interrupt in an AVR microcontroller such that pressing a push button triggers an interrupt service routine to toggle an LED.	2
8	**Experiment No. 8** Traffic Light Controller - Write an assembly/C program to simulate a traffic light system using three LEDs (Red, Yellow, Green) with appropriate timing delays using timers.	2
9	**Experiment No. 9** OLED Interface - Write an assembly/C program to interface an OLED display with AVR Microcontroller.	2
10	**Experiment No. 10** GPIO LED Blinking (PIC) - Write an assembly/C program to configure a GPIO pin of a PIC microcontroller as output and blink an LED with a specific delay.	2
11	**Experiment No. 11** Switch Interfacing with Pull-up/Pull-down - Write an assembly/C program to interface a switch with a PIC microcontroller using internal/external pull-up resistors and control an LED based on switch status.	2
12	**Experiment No. 12** Timer Interrupt Application - Write an assembly/C program to configure a timer interrupt in a PIC microcontroller to toggle an LED at regular intervals.	2

DR. RAJENDRAKUMAR DAHYALAL PATEL
Digitally signed by (Name of HOD)

DR. RAJENDRASINH BAHADURSINH JADEJA
Digitally signed by (Name of Dean/ Principal)

Suggested List of Experiments:

Contents : Unit	Topics	Contact Hours
13	**Experiment No. 13** UART Communication (PIC) - Write an assembly/C program to initialize UART on a PIC microcontroller and transmit/receive data to/from a PC terminal.	2
14	**Experiment No. 14** ADC-Based Temperature Measurement - Write an assembly/C program to interface a temperature sensor (LM35) with the PIC ADC module and display temperature readings via UART or LCD.	2
15	**Experiment No. 15** PWM-Based Motor Speed Control - Write an assembly/C program to configure PWM using the CCP module of a PIC microcontroller and control the speed of a DC motor.	2
16	**Experiment No. 16** Serial Communication Between AVR and PIC - Write C programs for both AVR and PIC microcontrollers to establish UART-based serial communication where one device transmits sensor data and the other receives and displays it.	2
17	**Capstone Project** Students will complete a comprehensive project that integrates microcontroller programming with hardware interfacing, demonstrating system design, implementation, and testing within the stipulated semester duration	0
Total Hours		**32**

Textbook :

1 The AVR Microcontroller and Embedded Systems: Using Assembly and C, Muhammad Ali Mazidi, Sarmad Naimi & Sepehr Naimi, Pearson Education, 2011

2 PIC Microcontroller and Embedded Systems: Using Assembly and C for PIC18, Muhammad Ali Mazidi, Rolin D. McKinlay & Danny Causey, Pearson Education, 2008

References:

1 Designing with PIC Microcontrollers, Designing with PIC Microcontrollers, John B. Peatman, Pearson Education, 1997

2 Programming and Customizing the AVR Microcontroller, Programming and Customizing the AVR Microcontroller, Dhananjay V. Gadre, McGraw-Hill Education, 2001

Suggested Theory Distribution:

The suggested theory distribution as per Bloom's taxonomy is as follows. This distribution serves as guidelines for teachers and students to achieve effective teaching-learning process

Distribution of Theory for course delivery

Remember / Knowledge	Understand	Apply	Analyze	Evaluate	Higher order Thinking / Creative
5.00	10.00	40.00	20.00	15.00	10.00

Instructional Method:

1. The internal evaluation will be done based on the continuous evaluation of students in the laboratory and class-room.
2. A practical examination will be conducted at the end of the semester for evaluation of practical performance.
3. Students may use supplementary resources such as online videos, NPTEL videos, e-courses, Virtual Laboratory, etc
4. The course delivery method will depend upon the requirements of content and need of the students. The teacher in addition to conventional teaching methods (Chalk and Talk) may use any of the tools/techniques such as demonstration, role play, Quiz, brainstorming, Flipped class, Project based learning, Collaborative learning, MOOCs etc. for effective teaching.

Supplementary Resources:

1. https://nptel.ac.in/courses/117104072
2. https://www.microchip.com/en-us/application-notes
3. https://www.microchip.com/en-us/tools-resources/develop/microchip-studio
4. https://www.avrfreaks.net/s/forums
5. https://ww1.microchip.com/downloads/aemDocuments/documents/DEV/ProductDocuments/UserGuides/MPLAB_X_IDE_Users_Guide_50002027.pdf
6. https://simulide.com/p/

DR. RAJENDRAKUMAR DAHYALAL PATEL
Digitally signed by (Name of HOD)

DR. RAJENDRASINH BAHADURSINH JADEJA
Digitally signed by (Name of Dean/ Principal)