

Subject Code	Subject Name	Period per Week		Credit
28564	<b>MICROCONTROLLER BASED SYSTEM DESIGN &amp; DEVELOPMENT</b>	T	P	C
		2	6	4

<b>Rationale</b>	As electronic devices continue to play a vital role in our daily lives, the need for understanding Microcontroller and Embedded Systems becomes imperative. This course aims to equip Diploma in Engineering Level students with knowledge and skills related to Microcontroller Architecture, Programming, Interfacing within Embedded Systems, and practical applications in real-world projects.
<b>Learning Outcome (Theoretical)</b>	<b>After completing the subject, students will be able to:</b> <ol style="list-style-type: none"> <li>1. Interpret Microcontroller Architecture.</li> <li>2. Write Program Microcontrollers using suitable IDEs.</li> <li>3. State procedure to Interface the Microcontrollers with various sensors and actuators.</li> <li>4. Comprehend communication protocols in Embedded Systems.</li> <li>5. Explain Analog-to-Digital Conversion and PWM.</li> <li>6. Develop and troubleshoot projects involving Microcontrollers and Embedded Systems.</li> <li>7. Explain procedure of Interface with real world devices.</li> <li>8. Illustrate mini development kit.</li> </ol>
<b>Learning Outcome (Practical)</b>	<b>After undergoing the subject, students will be able to:</b> <ol style="list-style-type: none"> <li>1. Operate Microcontroller-based projects.</li> <li>2. Demonstrate effective debugging and troubleshooting skills.</li> <li>3. Interface Microcontrollers with sensors and actuators.</li> <li>4. Develop practical solutions using Microcontroller programming.</li> <li>5. Evaluate projects for functionality and reliability.</li> <li>6. Apply safety precautions in the development and testing of Microcontroller-based systems.</li> <li>7. Apply detailed procedure of microcontroller-based system development.</li> <li>8. Develop application in assembly &amp; C language.</li> <li>9. Originate and test a program for LEDs, 7-Segment display, LCD.</li> <li>10. Drive any AC load (light, fan etc.), DC motor.</li> <li>11. Design and test a program for using built-in timer with any MCU.</li> <li>12. Configure serial port communication by using any MCU.</li> <li>13. Interface temperature, light, sound, touch, infrared sensor by using any MCU.</li> </ol>

## Detailed Syllabus (Theory)

Unit	Topics with contents	Class (1 Period)	Final Marks
1	<b>Introduction to Microcontroller</b> 1.1 Define Microcontroller. 1.2 State the importance of Microcontroller. 1.3 Mention the types of Microcontrollers. 1.4 Define Embedded system. 1.5 Sketch the basic building blocks of a microcontroller. 1.6 Compare between Microprocessor and Microcontroller. 1.7 Distinguish between computer and embedded system. 1.8 List the applications of microcontroller. 1.9 Mention the criteria to choose a microcontroller. 1.10 List commonly used microcontrollers' series with targeted application.	2	3
2	<b>PIC Series Microcontroller</b> 2.1 Define PIC MCU. 2.2 State the features of PIC microcontroller. 2.3 Mention the purposes of different family/series of PIC MCU. 2.4 Describe basic functional blocks of PIC mid-range MCU. 2.5 State the types of Reset used in MCU. 2.6 Define Power Up Timer -PWRT and Oscillator Start Up Timer –OST. 2.7 Discuss the function of different interrupt sources. 2.8 Describe the operation of Watchdog Timer. 2.9 State the function of Power down/low power Mode (Sleep and Wake up from sleep).	2	3
3	<b>Microcontroller Programming and Interfacing</b> 3.1 Describe the development life cycle of embedded system. 3.2 List different types of software development tools for microcontroller programming. 3.3 Discuss the uses of IDE, linker, compiler, editor and assembler. 3.4 Mention commonly used IDE for microcontroller programming 3.5 Interpret MPLAB, MPLAB X and MikroC. 3.6 List different types of hardware development tools. 3.7 State the uses of flasher, debugger and emulator. 3.8 Mention commonly used debugger and flashing tools. 3.9 State the procedure to build embedded software. 3.10 Write a program for accessing GPIO port.	3	6
4	<b>Timer/counter Programming</b> 4.1 Define Timer in microcontroller. 4.2 List the functions of a timer in microcontroller. 4.3 Discuss the mode of operation of a timer in microcontroller. 4.4 Write a program to create delay for certain amount of time using Timer. 4.5 Describe the Timer as an even counter. 4.6 Develop program for generating square wave and PWM.	3	6

5	<b>Interrupts of Microcontroller</b> 5.1 List the interrupts of the mid-range PIC MCU. 5.2 State the interrupt priority and vector locations. 5.3 State the procedure of enabling and disabling interrupts. 5.4 Mention the steps to execute an interrupt service routine. 5.5 Describe External Hardware, Timer, and Serial communication Interrupt. 5.6 Write a program for interrupts of Microcontroller.	3	5
6	<b>Assembly Language Programming of PIC Mid-range Microcontroller</b> 6.1 Describe Assembly language. 6.2 Define Instruction and Instruction set. 6.3 Classify instructions. 6.4 Mention the fields of assembly language instruction. 6.5 Describe the function of assembly language instruction. 6.6 State CPU Instruction & Assembler directives. 6.7 Describe the commonly used Assembler directives. 6.8 Mention steps to create and execute assembly language program. 6.9 Develop a program in assembly language to send data to the output port.	3	6
7	<b>Interfacing Sensors and Actuators</b> 7.1 Define sensors and actuators. 7.2 Differentiate between Sensors and Actuators. 7.3 State the types of actuators with application. 7.4 Describe the interfacing process of sensors and actuators in an embedded system. 7.5 Write a program for Controlling motors and LEDs using microcontrollers. 7.6 Write a program to get temperature, humidity and light using MCU. 7.7 Develop an embedded system block diagram that uses different sensors and actuators.	4	8
8	<b>Communication Protocols in Embedded Systems</b> 8.1 Define Communication protocol. 8.2 State the uses of UART, I2C and SPI communication protocols. 8.3 Define wireless communication. 8.4 Discuss the process of Bluetooth and RF modules. 8.5 Mention the interfacing process of multiple microcontrollers. 8.6 Write a program for serial communication. 8.7 Develop an embedded system block diagram that uses different communication systems	3	6
9	<b>Arduino</b> 9.1 State Arduino, Arduino board and Arduino shield. 9.2 State the features of different Arduino boards. 9.3 State different Arduino Shields. 9.4 Describe the uses of Arduino Board and Sensor. 9.5 Explain the basic block diagram of Arduino board. 9.6 Describe the Interfacing with Arduino components. 9.7 Develop program for any Arduino device with basic operation. 9.8 Write a program for temperature sensor data or light sensor data using Arduino. 9.9 Develop a program to read analog signal from a sensor. 9.10 Develop a program to communicate with computer system with using any communication protocol.	2	3

10	<b>IoT device, Raspberry Pi and other development kit</b> 10.1 Mention the uses of ARM architecture. 10.2 Discuss the basic building blocks of an IoT devices 10.3 State the uses of GSM module like device to connect the internet. 10.4 State the uses of Wi-Fi module to connect the internet. 10.5 Develop an embedded system block diagram for internet connectivity module. 10.6 State the features and uses of Raspberry Pi board. 10.7 List commonly used mini development kit based on MCU. 10.8 Outline commonly used robotics development board.	3	6
	<b>TOTAL</b>	<b>32</b>	<b>60</b>

### **Detailed Syllabus (Practical)**

**NB:**

- For the experiments of microcontroller (MCU) students can use any MCU i.e., PIC, AVR, 8051 core, Fujitsu, Lapis, Renesas and ST etc.
- For developing programs, students can use any development tools i.e., MPLAB X, Atmel studio etc.

<b>SL. No.</b>	<b>Experiment name with procedure</b>	<b>Class (3 Period)</b>	<b>Continuous Marks</b>
1	<b>Design Schematic Diagram for LED blinking system.</b> 1.1 Select a simple PIC series mid-range MCU for LED flashing project. 1.2 Identify required input-output devices for building LED blinking system. 1.3 Interpret operation procedure of selected I/O devices. 1.4 Interpret the interface system of selected I/O devices with selected MCU. 1.5 Interpret the interface of different signals and pin of the selected MCU. 1.6 Sketch the connection diagram of the power pins of MCU with power supply in schematic design 1.7 Sketch the connection diagram of the crystal with clock signal related pin in schematic design. Internal clock source can be used as well. 1.8 Sketch the connection diagram of other system pin of MCU, for example, like reset etc. with necessary components or power supply if necessary. 1.9 Sketch the connection diagram of the I/O device and other parts, if necessary, with MCU to complete the schematic design.	1	1
2	<b>Develop a program for LED blinking system using MCU/Arduino.</b> 2.1 Design the circuit diagram. 2.2 Draw the process flow chart. 2.3 Write the code. 2.4 Compile/build the program. 2.5 Simulate the program and the circuit if necessary. 2.6 Flash/Download the Hex file/program to the MCU using flashing device.	1	1

	2.7 Construct the circuit. 2.8 Power the circuit and observe the output. 2.9 Debug program/system using debugger device if necessary. 2.10 Test the program. 2.11 Prepare a report. 2.12 Maintain the record of performed task.		
3	<b>Develop a program for Multiple LED blinking system using MCU/Arduino.</b> 3.1 Design the circuit diagram. 3.2 Draw the process flow chart. 3.3 Write the code. 3.4 Compile/build the program. 3.5 Simulate the program and the circuit if necessary. 3.6 Flash/Download the Hex file/program to the MCU using flashing device. 3.7 Construct the circuit. 3.8 Power the circuit and observe the output. 3.9 Debug program/system using debugger device if necessary. 3.10 Test the program. 3.11 Prepare a report. 3.12 Maintain the record of performed task.	1	1
4	<b>Develop a program for LED Control by Switch using MCU/Arduino.</b> 4.1 Design the circuit diagram. 4.2 Draw the process flow chart. 4.3 Write the code. 4.4 Compile/build the program. 4.5 Simulate the program and the circuit if necessary. 4.6 Flash/Download the Hex file/program to the MCU using flashing device. 4.7 Construct the circuit. 4.8 Power the circuit and observe the output. 4.9 Debug program/system using debugger device if necessary. 4.10 Test the program. 4.11 Prepare a report. 4.12 Maintain the record of performed task.	1	1
5	<b>Develop a program for LED Brightness Control using MCU/Arduino.</b> 5.1 Design the circuit diagram. 5.2 Draw the process flow chart. 5.3 Write the code. 5.4 Compile/build the program. 5.5 Simulate the program and the circuit if necessary. 5.6 Flash/Download the Hex file/program to the MCU using flashing device. 5.7 Construct the circuit. 5.8 Power the circuit and observe the output. 5.9 Debug program/system using debugger device if necessary. 5.10 Test the program. 5.11 Prepare a report. 5.12 Maintain the record of performed task.	2	1

6	<b>Develop a program for Buzzer On/Off using MCU/Arduino.</b> 6.1 Design the circuit diagram. 6.2 Draw the process flow chart. 6.3 Write the code. 6.4 Compile/build the program. 6.5 Simulate the program and the circuit if necessary. 6.6 Flash/Download the Hex file/program to the MCU using flashing device. 6.7 Construct the circuit. 6.8 Power the circuit and observe the output. 6.9 Debug program/system using debugger device if necessary. 6.10 Test the program. 6.11 Prepare a report. 6.12 Maintain the record of performed task.	1	1
7	<b>Develop a program for Buzzer Melody using MCU/Arduino.</b> 7.1 Design the circuit diagram. 7.2 Draw the process flow chart. 7.3 Write the code. 7.4 Compile/build the program. 7.5 Simulate the program and the circuit if necessary. 7.6 Flash/Download the Hex file/program to the MCU using flashing device. 7.7 Construct the circuit. 7.8 Power the circuit and observe the output. 7.9 Debug program/system using debugger device if necessary. 7.10 Test the program. 7.11 Prepare a report. 7.12 Maintain the record of performed task.	2	1
8	<b>Develop a program for displaying decimal digit(0 to 9) or hexadecimal digit on 7- Segment display by using MCU/Arduino.</b> 8.1 Design the circuit diagram. 8.2 Draw the process flow chart. 8.3 Write the code. 8.4 Compile/build the program. 8.5 Simulate the program and the circuit if necessary. 8.6 Flash/Download the Hex file/program to the MCU using flashing device. 8.7 Construct the circuit. 8.8 Power the circuit and observe the output. 8.9 Debug program/system using debugger device if necessary. 8.10 Test the program. 8.11 Prepare a report. 8.12 Maintain the record of performed task.	1	1
9	<b>Develop a program for Interfacing LCD by using any MCU.</b> 9.1 Design the circuit diagram. 9.2 Draw the process flow chart. 9.3 Write the code. 9.4 Compile/build the program. 9.5 Simulate the program and the circuit if necessary. 9.6 Flash/Download the Hex file/program to the MCU using flashing device. 9.7 Construct the circuit. 9.8 Power the circuit and observe the output. 9.9 Debug program/system using debugger device if necessary.	2	1

	9.10 Test the program. 9.11 Prepare a report. 9.12 Maintain the record of performed task.		
10	<b>Perform the construction of a MCU based timing pulse generation system by using built-in timer.</b> 10.1 Design the circuit diagram. 10.2 Draw the process flow chart. 10.3 Write the code. 10.4 Compile/build the program. 10.5 Simulate the program and the circuit if necessary. 10.6 Flash/Download the Hex file/program to the MCU using flashing device. 10.7 Construct the circuit. 10.8 Power the circuit and observe the output. 10.9 Debug program/system using debugger device if necessary. 10.10 Test the program. 10.11 Prepare a report. 10.12 Maintain the record of performed task.	<b>1</b>	<b>1</b>
11	<b>Perform the construction of a MCU based system to count pulses.</b> 11.1 Design the circuit diagram. 11.2 Draw the process flow chart. 11.3 Write the code. 11.4 Compile/build the program. 11.5 Simulate the program and the circuit if necessary. 11.6 Flash/Download the Hex file/program to the MCU using flashing device. 11.7 Construct the circuit. 11.8 Power the circuit and observe the output. 11.9 Debug program/system using debugger device if necessary. 11.10 Test the program. 11.11 Prepare a report. 11.12 Maintain the record of performed task.	<b>1</b>	<b>1</b>
12	<b>Perform the construction of a MCU based ADC (Analog to digital converter) interface System.</b> 12.1 Design the circuit diagram. 12.2 Draw the process flow chart. 12.3 Write the code. 12.4 Compile/build the program. 12.5 Simulate the program and the circuit if necessary. 12.6 Flash/Download the Hex file/program to the MCU using flashing device. 12.7 Construct the circuit. 12.8 Power the circuit and observe the output. 12.9 Debug program/system using debugger device if necessary. 12.10 Test the program. 12.11 Prepare a report. 12.12 Maintain the record of performed task.	<b>2</b>	<b>1</b>

13	<b>Develop a program for Interfacing Keyboard using MCU/Arduino.</b> 13.1 Design the circuit diagram. 13.2 Draw the process flow chart. 13.3 Write the code. 13.4 Compile/build the program. 13.5 Simulate the program and the circuit if necessary. 13.6 Flash/Download the Hex file/program to the MCU using flashing device. 13.7 Construct the circuit. 13.8 Power the circuit and observe the output. 13.9 Debug program/system using debugger device if necessary. 13.10 Test the program. 13.11 Prepare a report. 13.12 Maintain the record of performed task.	1	1
14	<b>Develop a program for Interfacing 8×8 LED Dot Matrix using MCU/Arduino.</b> 14.1 Design the circuit diagram. 14.2 Draw the process flow chart. 14.3 Write the code. 14.4 Compile/build the program. 14.5 Simulate the program and the circuit if necessary. 14.6 Flash/Download the Hex file/program to the MCU using flashing device. 14.7 Construct the circuit. 14.8 Power the circuit and observe the output. 14.9 Debug program/system using debugger device if necessary. 14.10 Test the program. 14.11 Prepare a report. 14.12 Maintain the record of performed task.	1	1
15	<b>Perform the construction of a MCU based LDR (Light dependent resistor) interface system.</b> 15.1 Design the circuit diagram. 15.2 Draw the process flow chart. 15.3 Write the code. 15.4 Compile/build the program. 15.5 Simulate the program and the circuit if necessary. 15.6 Flash/Download the Hex file/program to the MCU using flashing device. 15.7 Construct the circuit. 15.8 Power the circuit and observe the output. 15.9 Debug program/system using debugger device if necessary. 15.10 Test the program. 15.11 Prepare a report. 15.12 Maintain the record of performed task.	2	2
16	<b>Develop a program for Interfacing DC motor by using any MCU/Arduino.</b> 16.1 Design the circuit diagram. 16.2 Draw the process flow chart. 16.3 Write the code. 16.4 Compile/build the program.	2	1



	16.5 Simulate the program and the circuit if necessary. 16.6 Flash/Download the Hex file/program to the MCU using flashing device. 16.7 Construct the circuit. 16.8 Power the circuit and observe the output. 16.9 Debug program/system using debugger device if necessary. 16.10 Test the program. 16.11 Prepare a report. 16.12 Maintain the record of performed task.		
17	<b>Develop program for Interfacing a Bluetooth Module by using any MCU/Arduino.</b> 17.1 Design the circuit diagram. 17.2 Draw the process flow chart. 17.3 Write the code. 17.4 Compile/build the program. 17.5 Simulate the program and the circuit if necessary. 17.6 Flash/Download the Hex file/program to the MCU using flashing device. 17.7 Construct the circuit. 17.8 Power the circuit and observe the output. 17.9 Debug program/system using debugger device if necessary. 17.10 Test the program. 17.11 Prepare a report. 17.12 Maintain the record of performed task.	<b>2</b>	<b>1</b>
18	<b>Develop a program for Interfacing a Wi-Fi Module by using any MCU/Arduino.</b> 18.1 Design the circuit diagram. 18.2 Draw the process flow chart. 18.3 Write the code. 18.4 Compile/build the program. 18.5 Simulate the program and the circuit if necessary. 18.6 Flash/Download the Hex file/program to the MCU using flashing device. 18.7 Construct the circuit. 18.8 Power the circuit and observe the output. 18.9 Debug program/system using debugger device if necessary. 18.10 Test the program. 18.11 Prepare a report. 18.12 Maintain the record of performed task.	<b>2</b>	<b>1</b>
19	<b>Develop a program for interfacing Temperature Sensor by using Arduino.</b> 19.1 Design the circuit diagram. 19.2 Draw the process flow chart. 19.3 Write the code. 19.4 Compile/build the program. 19.5 Simulate the program and the circuit if necessary. 19.6 Flash/Download the Hex file/program to the MCU using flashing device. 19.7 Construct the circuit. 19.8 Power the circuit and observe the output. 19.9 Debug program/system using debugger device if necessary.	<b>1</b>	<b>1</b>

	19.10 Test the program. 19.11 Prepare a report. 19.12 Maintain the record of performed task.		
20	<b>Develop a program for interfacing Light Sensor by using Arduino.</b> 20.1 Design the circuit diagram. 20.2 Draw the process flow chart. 20.3 Write the code. 20.4 Compile/build the program. 20.5 Simulate the program and the circuit if necessary. 20.6 Flash/Download the Hex file/program to the MCU using flashing device. 20.7 Construct the circuit. 20.8 Power the circuit and observe the output. 20.9 Debug program/system using debugger device if necessary. 20.10 Test the program. 20.11 Prepare a report. 20.12 Maintain the record of performed task.	<b>1</b>	<b>1</b>
21	<b>Develop a program for interfacing Sound Sensor by using Arduino.</b>	<b>1</b>	<b>1</b>
22	<b>Develop a program for interfacing Touch Sensor by using Arduino.</b> 22.1 Design the circuit diagram. 22.2 Draw the process flow chart.	<b>1</b>	<b>1</b>
23	<b>Develop a program for interfacing Infrared Sensor by using Arduino.</b> 23.1 Design the circuit diagram. 23.2 Draw the process flow chart. 23.3 Write the code. 23.4 Compile/build the program. 23.5 Simulate the program and the circuit if necessary. 23.6 Flash/Download the Hex file/program to the MCU using flashing device. 23.7 Construct the circuit. 23.8 Power the circuit and observe the output. 23.9 Debug program/system using debugger device if necessary. 23.10 Test the program. 23.11 Prepare a report. 23.12 Maintain the record of performed task.	<b>1</b>	<b>1</b>
24	<b>Develop a program for driving a Relay by using Arduino kit with any wireless shield.</b> 24.1 Design the circuit diagram. 24.2 Draw the process flow chart. 24.3 Write the code. 24.4 Compile/build the program. 24.5 Simulate the program and the circuit if necessary. 24.6 Flash/Download the Hex file/program to the MCU using flashing device. 24.7 Construct the circuit. 24.8 Power the circuit and observe the output.	<b>1</b>	<b>1</b>

	24.9 Debug program/system using debugger device if necessary. 24.10 Test the program. 24.11 Prepare a report. 24.12 Maintain the record of performed task.		
	<b>Total</b>	<b>32</b>	<b>50</b>

**Necessary Resources (Tools, Materials, equipment's and Machineries):**

SL.No.	Item Name	Quantity
01	MCU training kit	5
02	Program/software development tools for MCU	5
03	Computer	5
04	Arduino board, shield with sensor set	5

SL.No.	Book Name	Writer Name	Publisher Name & Edition
01	PIC Microcontroller and Embedded system	Muhammad Ali Mazidi	Pearson
02	C Programming for the PIC Microcontroller	Hubert Henry Ward	Apress

**Website References:**

SL. No.	Web Link	Remarks
01	<a href="http://www.microchip.com">www.microchip.com</a>	
02	<a href="http://www.arduino.cc">www.arduino.cc</a>	
03	<a href="http://www.raspberrypi.org">www.raspberrypi.org</a>	