GUJARAT TECHNOLOGICAL UNIVERSITY

Program Name: Diploma Engineering
Level: Diploma
Branch: Electronics & Communication Engineering / Information & Communication Technology
Subject Code : DI04000111
Subject Name : Microprocessor & Microcontroller

w. e. f. Academic Year:	2025-26
Semester:	4th
Category of the Course:	PCC

Prerequisite:	Basics of digital electronics, computer organization, electronics fundamentals, and introductory programming concepts.
Rationale:	The course aims to provide students with a strong foundation in microprocessors and microcontrollers, focusing on the 8085 microprocessor and 8051 microcontroller. It emphasizes understanding architecture, instruction sets, programming concepts, and interfacing techniques. By combining theoretical knowledge with practical applications, this course prepares students to design and implement microcontroller-based systems for real-world problems across industrial, consumer, and automation fields.

Course Outcome:
After Completion of the Course, Student will be able to:

No	Course Outcomes	RBT Level
01	Demonstrate 8085 microprocessor architecture and operations.	R, U
02	Describe 8051 microcontroller architecture, features, and memory organization.	R, U
03	Develop assembly and C programs for 8051 microcontroller using instruction sets.	R, U, A
04	Implement timer/counter, serial communication, and interrupt-driven programs for embedded applications.	R, U, A
05	Interface various components and peripheral devices with the 8051 microcontroller.	R, U, A

**Revised Bloom's Taxonomy (RBT)*

Teaching and Examination Scheme:

Teaching Scheme (in Hours)			Total Credits L+T+ (PR/2)	Assessment Pattern and Marks				Total Marks
				Theory		Tutorial / Practical		
L	T	PR	C	ESE (E)	PA(M)	PA(I)	ESE (V)	
3	0	2	4	70	30	20	30	150

Course Content:

Unit No.	Content	No. of Hours	% of Weightage
1.	**Unit – I: Microprocessor Systems and 8085 Architecture** 1.1 Definition & History of Microprocessor 1.2 CPU, Control Unit, Arithmetic logic Unit (ALU), Memory Unit, Power Unit and Input-Output unit 1.3 Von neumann & Harvard architectures 1.4 RISC & CISC architectures 1.5 8085 microprocessor Pins, Address bus, Data bus, Control bus & Signals 1.6 8085 Block diagram and internal architecture 1.7 Comparison of Microprocessor & Microcontroller	6	15
2.	**Unit– II: Microcontroller 8051 Architecture and Features** 2.1 Common features of Microcontrollers: On-chip Oscillator, program and data memory, I/O Ports, Reset, SFRs, Timers, Counters, Interrupts 2.2 Blocks of Microcontroller 8051: ALU, PC, DPTR, PSW, Internal RAM, Internal ROM, SFRs, General purpose registers 2.3 Functions of each pin of 8051 microcontroller 2.4 Internal RAM and ROM organization 2.5 Stack, Stack Pointer and Stack operation concepts 2.6 Special Function Registers (SFRs) and their applications	9	20
3.	**Unit– III: 8051 Programming and Instruction Set** 3.1 Addressing Modes: Immediate, Register, Direct, Indirect, Indexed, Relative and bit addressing 3.2 Define Opcode, Operand, Instruction cycle, Machine cycle & T state 3.3 Instruction set: Data Transfer, Arithmetic, Logical, Branching, and Machine Control instructions 3.4 Data manipulation and transfer programs 3.5 Arithmetic and Logical operations programming 3.6 Masking, Stack operation, Conditional programming 3.7 Bit manipulation and control programs 3.8 Introduction to 8051 C programming: Structure of C program for microcontroller, Basic Example programs using C	12	25
4.	**Unit-IV: Timer/Counter, Serial Communication and Interrupts** 4.1 Timer/Counter logic diagram and architecture	7	15

	4.2 Timer/Counter operation in various modes (Mode 0, 1, 2, 3) 4.3 Serial communication modes and programming 4.4 Interrupt Vector Address, Priority & Operations 4.5 Interrupt service routine programming		
5.	**Unit-V: Interfacing and Applications of Microcontroller 8051** 5.1 Switches, Push buttons interfacing and programming 5.2 Relay, LED, 7 segment LED, LCD interfacing 5.3 DAC0808, ADC0804 interfacing and programming 5.4 DC Motor, Stepper motor interfacing and control 5.5 Application of microcontroller in various fields (Industrial automation, Consumer electronics, Automotive, etc.) 5.6 Applications: • Automatic Street Light Control using LDR sensor • Fire Alarm System using smoke sensor and buzzer • Water Level Controller with motor control and indicators • Digital Stopwatch with start/stop/reset functionality	11	25
	Total	**45**	**100**

Suggested Specification Table with Marks (Theory):

Distribution of Theory Marks (in %)					
R Level	**U Level**	**A Level**	**N Level**	**E Level**	**C Level**
20	**45**	**35**	**--**	**--**	**--**

Where R: Remember; U: Understanding; A: Application, N: Analyze and E: Evaluate C: Create (as per Revised Bloom's Taxonomy)

References/Suggested Learning Resources:
(a) Books:

S. No.	Title of Book	Author	Publication with place, year and ISBN
1	Microprocessor Architecture, Programming, and Applications with the 8085	Ramesh Gaonkar	Penram Publications
2	The 8051 Microcontroller and Embedded Systems: Using Assembly and C	Mazidi & Mazidi	Pearson Publication

3	The 8051 Microcontroller	Kenanth Ayala	Cengage Learning India

(b) Open-source Software, Websites, YouTube channels, Free MOOC courses:

Category	Name	What It Offers / Why Useful	Source / Link
Open-Source Tools / Simulators	MCU 8051 IDE	Free IDE + Simulator for 8051; supports C with SDCC & assembly; view registers, LCD/LED, interrupt simulation etc. (Wikipedia)	SourceForge project "MCU 8051 IDE" (Wikipedia
	SDCC (Small Device C Compiler)	Open-source C compiler + assembler/linker & simulator for 8-bit MCUs including 8051. Helpful for writing embedded C code. (Wikipedia)	SDCC official site / wiki (Wikipedia)
Free Websites / Tutorials / Lecture Collections	Free Video Lectures — Microprocessors & Microcontrollers course	A full lecture series covering introduction, 8085 architecture, etc., useful for theory foundation. (Free Video Lectures)	FreeVideoLectures site (Free Video Lectures)
	IIT Kharagpur — Video Lectures by Prof. Ajit Pal	Deep set of videos on 8085, instruction sets, memory interfacing, 8051, etc. Very good for detailed understanding. (satishkashyap.com)	via site SatishKashyap / IIT-KGP resource page (satishkashyap.com)
	NPTEL / DIGIMAT — Microprocessors & Microcontrollers	Video lectures on 8085 etc., good academic standard. (Digimat)	NPTEL / DIGIMAT Platform (Digimat)
Free Video Lectures /	"8051 Microcontroller" (Udemy, free)	Short course (1-2 hours); covers IO Ports, timers, counters, serial port, interrupts; mostly hands-on	Udemy (via ClassCentral listing) (Class Central)

YouTube / MOOC		via mikroC & simulation. (Class Central)	
	YouTube: *Introduction to 8085 Microprocessor* (Neso Academy)	Video giving an overview of 8085, pin-diagram etc.; good starter. (YouTube)	YouTube (Neso Academy) (YouTube)
	YouTube Playlist *Microprocessor & Interfacing 8085* (Engineering Funda)	Multiple videos walking through 8085 concepts and interfacing. (YouTube)	YouTube playlist – Engineering Funda (YouTube)
	YouTube: *Embedded Systems Full Course – The 8051 Microcontroller Using Assembly and Embedded C* (YT Training Institute)	Overview/introduction video, pins, first C program etc.; useful for mixed theory + demo. (YouTube)	YouTube (YT Training Institute) (YouTube)
	Coursera: *Microcontroller and Industrial Applications* (L&T EduTech)	MOOC with modules on 8051 architecture, I/O ports, interrupts etc.; free to audit. (MOOC List)	Coursera via MOOC-List (MOOC List)

Suggested Course Practical List: The Suggested list of PROGRAMS is based on 8051 Microcontroller.

Sr. No.	Practical Outcomes (PrOs)	Unit No.	Approx. Hrs required
1	Explore various blocks of Microprocessor System.		
2	Learn architecture and pin diagram of Microprocessor chip 8085.		
3	Learn architecture and pin diagram of Microcontroller chip 8051.	2	2
4	Use 8051 Simulation tool / Trainer kit for running ASM programs.	2	2
5	Write and execute assembly language programs based on Data transfer Instruction	3	2
6	Develop programs based on Arithmetic Instructions using assembly language and C.	3	2
7	Develop Programs based on Logical Instructions using assembly language and C.	3	2
8	Develop Complement and rotate data in accumulator using assembly language	3	2

9	Develop Programs based on Branch Instructions using Assembly Language and C.	3	2
10	Develop C program on 8051 microcontroller to blink an LED connected to Port 1 with a delay.	3	2
11	Develop Assembly Language Programs to introduce delay (e.g.1ms Delay) using Timer/Counter	4	2
12	Develop Assembly Language Programs to introduce serial communication	4	2
13	Develop a program to interface LED with 8051	5	2
14	Develop a program to interface Push Button switch with 8051	5	2
15	Develop a program to interface 7 segment Display with 8051	5	2
16	Develop a program to Interface 8 bit DAC and ADC with 8051	5	2
17	Develop a program to interface a DC Motor with 8051	5	2
18	Develop a program to interface a Stepper Motor with 8051	5	2
19	Develop a program to interface LCD Module with 8051	5	2
20	Develop a program to interface a LM35 with 8051	5	2
	Minimum 15 Practicals		**Min 30 Hrs**

List of Laboratory equipment /Learning Resources Required:

Laboratory Equipment

1. **8051 Microcontroller Trainer Kits** (with keypad, 7-segment, LCD, I/O ports, ADC/DAC, motor driver interface).
2. **8085 Microprocessor Trainer Kits** (for understanding basics of microprocessor architecture).
3. **Programmer / Development Board** (optional) – USB-ISP for 8051 based MCUs.
4. **LEDs, Switches, Push Buttons, Seven Segment Display** modules.
5. **LCD Display (16×2)** modules.
6. **Motors & Drivers:**
   - DC Motor with driver IC (L293D)
   - Stepper Motor with driver circuit
7. **Sensors:**
   - LDR (Light Dependent Resistor)
   - Temperature / Smoke sensor
   - Water level sensor
8. **ADC / DAC Converters** (ADC0804, DAC0808).
9. **Power Supply Units** (+5V regulated, dual supply if needed).
10. **Digital Multimeter & CRO (optional)** for observing signals.
11. **Breadboards, Resistors, Capacitors, ICs, Connecting Wires.**

Software / Learning Resources

1. **MCU 8051 IDE** (open-source simulator for Assembly + C).
2. **EdSim51 Simulator** (user-friendly for beginners).
3. **Keil µVision IDE** (widely used for Embedded C programming).
4. **Proteus Design Suite** (for circuit simulation & microcontroller interfacing).
5. **SDCC (Small Device C Compiler)** – free open-source C compiler for 8051.
6. **NPTEL Video Lectures** on Microprocessors & Microcontrollers.
7. **YouTube Playlists** (Neso Academy, Engineering Funda for 8085/8051).

Suggested Project List:

Sr. No.	Project Title	Objective / Description	Hardware / Software Required
1	Automatic Street Light Control using LDR	Switch ON streetlight at night, OFF during day	8051 kit, LDR sensor, LEDs
2	Digital Stopwatch	Implement Start/Stop/Reset with 7-segment display	8051 kit, 7-segment, push buttons
3	Water Level Controller	Control motor based on water level, show status on LEDs	8051 kit, water level sensor, motor driver
4	Temperature-Based Fan Control	Control fan speed using temperature sensor	8051 kit, LM35 sensor, DC fan, driver
5	Electronic Voting Machine	Record votes using push buttons, display results	8051 kit, push buttons, LCD
6	Password-Based Door Lock System	Unlock door using keypad password	8051 kit, keypad, LCD, relay
7	Traffic Light Controller	Simulate traffic signals using LEDs and timers	8051 kit, LEDs
8	Fire Alarm System	Activate buzzer when smoke is detected	8051 kit, smoke sensor, buzzer, LEDs
9	Digital Dice Using LEDs	Generate random number (1–6) using LEDs	8051 kit, LEDs, push button
10	Speed Control of DC Motor	Control motor speed via PWM	8051 kit, DC motor, driver IC
11	Automatic Room Light Controller	Auto ON/OFF lights based on entry/exit count	8051 kit, IR sensors, relay, lamp
12	Voting Machine with Result on LCD	Display candidate-wise results	8051 kit, push buttons, LCD
13	Smart Irrigation System	Auto motor ON/OFF based on soil moisture	8051 kit, soil moisture sensor, motor driver
14	Home Security System	Detect intruder and sound alarm	8051 kit, IR sensor, buzzer

15	Digital Alarm Clock	Display time and alarm using timer & interrupts	8051 kit, LCD, buzzer
16	Distance Measurement System	Measure distance using ultrasonic sensor	8051 kit, ultrasonic sensor, LCD
17	Automatic Fire Extinguisher Prototype	Detect fire, activate motor pump	8051 kit, smoke sensor, motor, pump

Suggested Activities for Students:

Sr. No.	Activity	Objective / Outcome
1	**Prepare a chart/poster** on 8085 vs 8051 features	Visual comparison to strengthen conceptual clarity
2	**Instruction Set Mnemonics Game** (flash cards/quiz)	Learn and recall 8051 instructions quickly
3	**Write small Assembly programs** (swap, sort, find largest, etc.) and share with peers	Practice coding and debugging
4	**Simulation challenge**: run a program in MCU8051 IDE/EdSim51 and explain results	Builds simulator usage skills
5	**Mini hackathon**: 2–3 hour group activity to design a simple interfacing project	Encourages teamwork & creativity
6	**Prepare flowcharts & algorithms** before coding 8051 programs	Improves problem-solving approach
7	**Case study discussion** on real-life microcontroller applications (cars, IoT, appliances)	Connects syllabus to industry
8	**Hands-on sensor demo**: students bring simple sensors (LDR, IR) and test with trainer kit	Encourages practical exploration
9	**Prepare a presentation** on "Evolution of Microcontrollers" or "RISC vs CISC"	Improves technical communication
10	**Quiz competition** on 8085/8051 architecture and programming	Reinforces theory knowledge
11	**Group assignment**: simulate a traffic light controller in software, then implement in hardware	Integrates simulation + hardware
12	**Documentation activity**: students write a short lab report or blog explaining a project	Enhances technical writing skills
13	**Prepare a glossary** of 50 key terms (CPU, opcode, ISR, stack pointer, etc.)	Builds technical vocabulary
14	**Peer teaching**: explain one assembly program to classmates	Develops confidence and mastery
15	**Explore open-source tools** (SDCC, Proteus, Keil) and prepare a demo video	Exposure to industry-relevant tools

Program Outcomes (POs):

1. **Basic & Discipline specific knowledge**: An apply knowledge of basic mathematics, science and engineering fundamentals and engineering specialization to solve the engineering problems.
2. **Problem Analysis:** Identify and analyze well defined engineering problems using codified standard methods.
3. **Design/ Development of Solution:** Design solutions for well-defined technical problems and assist with the design of systems, components or processes to meet specified needs.
4. **Engineering Tools, Experimentation and Testing:** Apply modern engineering tools and relevant technique to conduct standard tests and measurements.
5. **Engineering practices for Society, Environment and sustainability**: Apply relevant technology in context of Society, sustainability, environment and ethical practices.
6. **Project Management**: Use engineering management principles individually, as a team member or a leader to manage projects and effectively communicate about well-defined engineering activities.
7. **Life-long learning**: Ability to analyze individual needs and engage in updating in the context of context of technological changes.

Program Specific Outcomes (PSOs):

1. Develop proficiency in Installation, maintenance and troubleshooting of electronics and communication systems.
2. Create customized solution of real-life problems using hardware and software.

* * * * * * *