# GUJARAT TECHNOLOGICAL UNIVERSITY

**Program Name: Diploma Engineering**
**Level: Diploma**
**Branch: Electronics & Communication Engineering/ Information & Communication Technology**
**Subject Code: DI04000121**
**Subject Name: Embedded System**

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

| | |
|---|---|
| **Prerequisite:** | Basic Electronics, Basic Programming Knowledge, Digital Electronics |
| **Rationale:** | The knowledge of embedded system and microcontrollers is essential in the field of electronics as the world is migrating towards automation rapidly in every field. By learning this course students can develop their own embedded system using microcontrollers which is application specific to solve given real time problems. Thus, this course is an important course for students who want to apply the skills and knowledge of automation using embedded technology in various applications of the industries. |

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

| No | Course Outcomes | RBT Level |
|---|---|---|
| 01 | Select appropriate microcontroller for given embedded system. | R, U, A |
| 02 | Demonstrate architecture and working of AVR microcontroller. | R, U, A |
| 03 | Write and execute embedded C program for given application. | R, U, A |
| 04 | Interface AVR microcontroller with hardware for given embedded system | R, U, A |
| 05 | Develop small embedded system Applications using AVR 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. | Overview of Embedded System | 6 | 10 |
| 2. | AVR Microcontroller Architecture and Pin diagram | 11 | 25 |
| 3. | AVR Programing in C | 11 | 25 |
| 4. | AVR Interfacing | 10 | 25 |
| 5. | Embedded System Applications | 7 | 15 |
| | **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** | **40** | **40** | **-** | **-** | **-** |

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

**UNDERPINNING THEORY**

Only the major Underpinning Theory is formulated as higher level UOs of *Revised Bloom's taxonomy* in order development of the COs and competency is not missed out by the students and teachers. If required, more such higher level UOs could be included by the course teacher to focus on attainment of COs and competency.

| Unit | Unit Outcomes (UOs) | Topics and Sub-topics |
|---|---|---|
| **Unit – I Overview of Embedded System** | 1.a Define basic concept of embedded system.<br>1.b Explain Characteristics of embedded system.<br>1.c Explain Characteristics of real time operating system.<br>1.d Compare different AVR microcontrollers. | 1.1 Embedded system: Definition, General block diagram, working and characteristics.<br>1.2 Real Time Operating System: Definition, Characteristics.<br>1.3 Microcontrollers for embedded system: Criteria for choosing microcontroller.<br>1.4 History of AVR microcontroller.<br>1.5 AVR family overview. |

| Unit | Unit Outcomes (UOs) | Topics and Sub-topics |
|---|---|---|
| **Unit – II AVR Microcontroller Architecture and Pin diagram** | 2.a Explain general block diagram of AVR microcontroller.<br>2.b Explain data memory organization of ATmega32.<br>2.c Differentiate between SRAM and EEPROM.<br>2.d Explain purpose of Status Register.<br>2.e Describe how code is fetched from program memory.<br>2.f With a sketch, identify pin of ATmega32.<br>2.g Describe configuration of each port.<br>2.h Describe different ways of Power-On Reset.<br>2.i Describe different oscillator clock source.<br>2.j Describe mode of operation of Timers/Counters (Timer0).<br>2.k Describe features and hardware consideration of on-chip ADC. | 2.1 AVR Microcontroller architecture: (Simplified/general block diagram)<br>2.2 Data memory: General Purpose Registers, I/O Memory, Internal SRAM<br>2.3 EEPROM Memory<br>2.4 Status Register<br>2.5 Program Memory and Program Counter<br>2.6 ATmega32 pin configuration<br>2.7 I/O port configuration<br>2.8 Clock and Reset Circuits<br>2.9 Timers/Counters and its operation in various modes<br>2.10 On-chip ADC in ATmega32: Features, Hardware considerations |
| **Unit– III AVR Programing in C** | 3.a Distinguishes different data types for programming AVR in C.<br>3.b Write C program to configure and access I/O ports of ATmega32.<br>3.c Use bit-wise logic operations for bit manipulation.<br>3.e Write C programs to access EEPROM.<br>3.f Write C programs to generate delays using timers.<br>3.g Explain function of MAX232.<br>3.h Write C programs for serial data transmission. | 3.1 Data types and time delays<br>3.2 I/O port programing in C: Byte size and bit size I/O<br>3.3 Bit-wise Logic operation in C: AND, OR, EX-OR, Invert and Shift operation<br>3.5 Memory Allocation in C<br>3.6 Timer programing in C<br>3.7 Serial Communication: RS232 standard, MAX232 |
| **Unit– IV AVR Interfacing** | 4.a Read ADC using polling method.<br>4.b Interface Sensors with ATmega32.<br>4.c Interface Relay with ATmega32.<br>4.d Describe SPI working.<br>4.e Interface multiple 7-segment displays using MAX7221. | 4.1 On-chip ADC programing: Polling Method<br>4.2 Interfacing of LM35, MQ-8, Pressure and Flow Sensor.<br>4.3 Interfacing Relay using ULN2803<br>4.4 SPI programing in C |

| Unit | Unit Outcomes (UOs) | Topics and Sub-topics |
|---|---|---|
| | 4.f Explain functions of I2C(TWI) registers in AVR. | 4.5 Interfacing MAX7221<br>4.6 I2C-Two Wire Serial Interface (TWI). |
| **Unit-V Embedded System Applications** | 5.a Describe function of L293D.<br>5.b Control DC motor using PWM modes in 8-bit timer.<br>5.c Explain basic block diagram of Smart Irrigation System.<br>5.d Explain basic block diagram of IoT based Home Automation System.<br>5.e Explain basic block diagram of Motorized Control Robotics System.<br>5.f Embedded System applications for Green Hydrogen | 5.1 Motor Driver L293D<br>5.2 Speed control of DC motor using 8-bit timer in AVR.<br>5.3 Smart Irrigation System<br>5.4 IoT based Home Automation<br>5.5 Motorized Control Robotics System<br>5.6. Hydrogen gas monitoring system using Atmega32 |

**References/Suggested Learning Resources:**

**(a) Books:**

| Sr. No. | Title of Book | Author | Publication with place, year and ISBN |
|---|---|---|---|
| 1 | The AVR microcontroller and Embedded System. | Muhammad Ali Mazidi, Sarmad Naimi, Sepehr Naimi | Pearson Publication |
| 2 | Embedded C Programming and the Atmel AVR | Richard Barnett, Larry O'cull, Sarah Cox | Cengage Learning India |
| 3 | Programming and Interfacing ATMEL AVR Microcontrollers | Thomas Grace | Cengage Learning India |
| 4. | Hydrogen Fuel: Production, Transport, and Storage | Gupta, R. B. | CRC Press, Taylor & Francis Group |

**(b) Open-source software and website:**

a) **www.nptel.ac.in**
b) **www.electronicshub.org**
c) **www.circuitdigest.com**
d) **www.microchip.com/en-us/product/atmega32**

**Suggested Course Practical List:**

| S. No. | Practical Outcomes (PrOs) | Unit No. | Approx. Hrs. Required |
|---|---|---|---|
| 1 | Explore various blocks of Embedded System. | I | 02 |
| 2 | Learn architecture of ATMega32 Microcontroller. | II | 02 |
| 3 | Learn pin diagram of ATMega32 Microcontroller. | II | 02 |
| 4 | Write and execute a C program to configure and access I/O ports of ATmega32. | III | 02 |
| 5 | Write and execute a C program to perform bit-wise logic operations for bit manipulation. | III | 02 |
| 6 | Write and execute a C program to access EEPROM. | III | 02 |
| 7 | Write and execute a C program to generate delays using timers. | III | 02 |
| 8 | Write and execute a C program for serial data transmission. | III | 02 |
| 9 | Write and execute a C program to read data from ADC channel using polling method. | IV | 02 |
| 10 | Write and execute a C program to interface LM35 with ATMega32 | IV | 02 |
| 11 | Write and execute a C program to configure SPI. | IV | 02 |
| 12 | Write and execute a C Program to interface 7 segment display using MAX7221 with ATMega32. | IV | 02 |
| 13 | Write and execute a C program to configure Two wire serial interface (I2C) for sending and receiving data. | IV | 02 |
| 14 | Write and execute a C program to control speed of DC motor using PWM mode in 8-bit timer. | V | 02 |
| 15 | Write and execute a C program to control Servo Motor. | V | 02 |
| | **Total** | | **30** |

**Note**

i. More numbers of **Practical Exercises** can be designed and offered by the respective course teacher to develop the industry relevant skills/outcomes to match the COs. The above table is only a suggestive list**.**

ii. The following are some **sample** 'Process' and 'Product' related skills (more may be added/deleted depending on the course) that occur in the above listed **Practical Exercises** of this course required which are embedded in the COs and ultimately the competency.

| S. No. | Sample Performance Indicators for the PrOs | Weightage in % |
|---|---|---|
| 1 | Lab Records | 05 |
| 2 | Question answer or Writing steps exercise | 20 |

| S. No. | Sample Performance Indicators for the PrOs | Weightage in % |
|---|---|---|
| 3 | Executing of exercise | 40 |
| 4 | Printout/ Result | 20 |
| 5 | Viva voice | 15 |
| **Total** | | **100** |

**List of Laboratory/Learning Resources Required:**

| Sr. No | Equipment Name |
|---|---|
| 1. | AVR Atmega32 Trainer Kit/ AVR Atmega32 Development Board |
| 2. | DC Power Supply |
| 3. | Various Interfacing boards or components |

**Suggested Project List:**

1. Digital Voltmeter using ATmega32
   - Uses ADC to measure analog voltage (0–5V) and displays it on an LCD.
2. Traffic Light Controller
   - Simulates a 4-way traffic light system using LEDs and timers.
3. Password Protected Door Lock
   - Uses a keypad and LCD to enter a password and control a relay or servo motor.
4. Temperature Monitoring System
   - Interface LM35 temperature sensor with ADC, display temperature on LCD.
5. Digital Clock using RTC (DS1307)
   - Real-time clock interfaced via I2C; display time on a 16x2 LCD.
6. RFID Based Access System
   - Read RFID tags via UART; match ID and control access mechanism.
7. Scrolling Message Display (LED Matrix or LCD)
   - Display custom messages scrolling on an LCD or dot-matrix display.
8. Battery Level Indicator
   - Use ADC to measure battery voltage and display percentage using LEDs or LCD.
9. Automatic Water Level Controller
   - Use float sensors or ultrasonic sensor to detect water levels and control a pump.
10. Light-Dependent Street Light
    - Use LDR to turn ON street lights in dark using ATmega32.

**10. Hydrogen gas monitoring system using MQ-8.**

**11. Case Study on use of Embedded Technology in Green Hydrogen System**

**Suggested Activities for Students:**

Other than the laboratory learning, following are the suggested student-related ***co-curricular*** activities which can be undertaken to accelerate the attainment of the various outcomes in this course: Students should conduct following activities in group and prepare reports of each activity.

i. Prepare chart to represent the block diagram of different interfacing chips. Develop a practical application using ATMega32 Microcontroller

ii. Prepare General purpose board with all ports available as connector

iii. Prepare/Download a dynamic animation to illustrate the following

- Timer operation
- Two Wire serial Interface (I2C)
- MAX 7221 Interfacing.
- DC Motor Interfacing

* * * * * * *