

Diploma in Embedded Systems

Duration: 5 Months[5 days a week,3 hours a day, Total 300 hours]

❖ **Module 1: 8051 Microcontroller in Assemble Language**

- ✓ Characteristics of Embedded System
- ✓ Overview of 8051 Family
- ✓ Features of 89V51
- ✓ Block diagram and pin configuration of 89V51
- ✓ Oscillator circuitry, reset circuitry
- ✓ Memory mapping in 89V51
- ✓ Accessing banks & special function registers
- ✓ Flag & PSW register
- ✓ Timer, counter, external interrupt registers
- ✓ I / O registers
- ✓ UART (RS232 Communication)
- ✓ Introduction to Assembly Language
 - Addressing modes for memory access
 - Instructions set
 - Data move
 - Arithmetic instruction
 - Logical instruction
 - CALL and JUMP instruction
 - Boolean instruction
- ✓ Programming in assembly language using cross compiler put into practice different instruction set
- ✓ Introduction to tool kit mostly used in industry

- ✓ **MCU 8051 Simulator IDE Assignments.**
 - MOV Commands
 - Rotate Right/Left
 - Bit Operations
 - Arithmetic Instructions
 - Logic Operations
 - INC,DEC, Compliment, NOP
 - Call & JMP
 - CJNE & DJNZ
 - DPTR

 - Clear and structured program writing skills with the use of markers, labels and comments.
 - Interfacing LED and driver development.
 - Interfacing 7segment display and driver development.
 - Interfacing Hex Keypad and driver development.
 - To toggle the port using interrupt.
 - Interfacing LCD and driver development.
 - Interfacing serial port and driver development (serial communication).

❖ **Module 2: 8051 Microcontroller in C Language**

- ✓ Introduction to C programming
 - Introduction to different data types
 - Introduction to operators
 - Decision control structure
 - Loop control structure
 - Hashing & Bitwise operation
- ✓ Advance C programming
 - Function calls, passing / returning values
 - Pointers & Arrays
 - Call by value / Call by reference
 - Embedded systems programming in C
 - ISP programming
 - SFR bits & variables declaration
- ✓ Embedded C application development
 - Device driver for LED, Seven segment, LCD & Hex key pad.
 - Device driver for serial port communication.
 - Practice programs in embedded C
- ✓ Programming in C language using cross compiler put into practice different instruction set
- ✓ I2C Bus
 - Concept of serial & Parallel communication
 - Synchronous & Asynchronous communication
 - Classification system bus
 - Introduction to I2C bus
 - Definition of basic I2C terminology
 - I2c pins
 - Master and slave concept
 - Pull up resistor concept
 - Slave addressing
 - Explanation of different I2C condition using timing diagram
 - I2C Driver development using embedded C

List of Practicals:

- Interfacing Serial ADC/DAC with 89V51RD2 microcontroller using I2C
- Interfacing RTC & EEPROM with 8051 microcontroller using I2C
- Interfacing Traffic Light Controller with Microcontroller
- Interfacing DC Motor, Stepper Motor, Traffic Light Controller
- Interfacing Parallel ADC & DAC

Module 3: Arduino

Arduino Uno Syllabus

Introduction to Arduino:

The Arduino Platform, Block diagram, Architecture, Pin functions, overview of main features such as I/O Ports, Timers, interrupts serial port, PWM, ADC, etc.

Introduction to Arduino IDE, writing, saving, compiling and uploading sketches.

Interfacing discrete LEDs, Binary counter, Seven Segment LEDs.

Interfacing LCD, switch Interface.

Interfacing with different type of sensors and communication modules.

(Note-Since Arduino is released under an open source license, it already has many built in libraries, hence in this module a project based approach will be followed rather than a conventional one.)

❖ **Module 4: AVR Microcontroller (Basic)**

- Introduction to AVR architecture.
- Comparison of AVR with other CISC & RISC based systems and Microprocessors.
- AVR family Categories and importance (AT tiny/ ATmega/ Xmega).
- Atmega 16 pin details and specifications (with package detail).
- Instruction set / Bus architecture.
- RAM, FLASH, UART and other peripherals.
- Interrupts, timer, Counters.

Software: Introduction to AVR studio, Writing C programs in AVR studio, Compiling, Linking and simulating these programs, AVR boot loader, burning the hex file in to the flash memory of AVR Microcontroller.

List of Programs:

1. Interfacing discrete LED,LED Pattern,Binary Counter
2. Interfacing seven segment led, Decimal Counter.
3. Interfacing LCD.
4. Hex keypad Interfacing.
5. Timer & Interrupt.
6. Serial Communication.

❖ **Module 5: AVR Microcontroller (Advanced)**

- PWM
- SPI Bus
 - Introduction to SPI
 - SPI master and SPI slave concept
 - SPI as data exchange protocol
- SPI Registers
 - SPCR
 - SPDR
 - SPSR
 -
- ADC Registers
 - ADCSRA
 - ADMUX
 - ACSR
- I2C Registers
 - TWCR
 - TWBR
 - TWSR
 - TWDR
- EEPROM
 - Read/write Access
 - EEPROM address register

List of Programs:

1. Stepper Motor Control.
2. DC Motor Control
3. Traffic Light Controller
4. SPI Communication
5. Inbuilt ADC
6. Inbuilt EEPROM
7. I2C based RTC

❖ **Module 6: Introduction to ARM 7 Microcontroller (Basic)**

- ✓ Introduction of ARM as RISC machine
- ✓ Overview of ARM Family
- ✓ Special features of ARM LPC2148
- ✓ Memory organization
- ✓ Processor operating states
 - Architectural overview ARM- 32 bit
 - Instruction Length, data types, operating modes
 - ARM State General Registers and Program Counter
 - 32-bit ARM High performance instruction set
 - Thumb Architecture- 16 bit
 - Block diagram of ARM TDMI
 - Thumb State General Registers and Program Counter
 - 16-bit THUMB high code density instruction set
- ✓ 3-stage ARM Pipeline
- ✓ load/store architecture
- ✓ ARM operating modes
- ✓ Vector Interrupt controller
 - FIQ Interrupt
 - Vectored IRQ Interrupt
 - Non-Vectored Interrupt
- ✓ Peripherals
 - UART
 - Timer & counters
 - UART

List of Practicals:

1. Familiarization with Keil software for ARM 7. Writing a simple program in Assembly / C Language, compiling, simulating & debugging the program. Burning the hex code in flash memory and testing it. Interfacing discrete LEDs, Binary counter.
2. Seven Segment LEDs, Decimal counter.
3. Interfacing LCD
4. Keyboard Interface
5. Using RS-232 Serial port
6. Timer
7. Interrupt

❖ **Module 7: ARM 7 Microcontroller (Advanced)**

- ✓ ADC Features
 - PIN Description
 - ADC Registers
 - Hardware Triggered Conversion
 - DAC Pin Description
- ✓ RTC Description
 - RTC Block Diagram
 - Register description
 - RTC Interrupts
 - Leap year calculation
 - RTC Prescaler Block Diagram
- ✓ USB
 - USB Device Controller
 - Device Controller Block Diagram
 - Data flow
 - Slave Mode Transfer
 - DMA Mode Transfer
 - USB device Register MAP

List of Practicals:

1. DC Motor Interfacing
2. Stepper Motor Interfacing
3. Traffic Light Controller Interfacing
4. Using Built in ADC / DAC
5. Inbuilt RTC
6. Parallel ADC

ARM Cortex M3 Syllabus

Theory Syllabus:

- Introduction to ARM CORTEX M3
- Overview & Architecture of ARM CORTEX
- Pin connect block
- Iterative Design workflow for Communication Systems
- Peripheral Interface
- 32 bit & 16 bit timer accessing
- Serial Communication
- I2C & SPI communication
- CAN Module
- LPC 1786 Ethernet
- Nested vectored Interrupt controller
- Sticky Bits

Practical Syllabus

- Familiarization with Keil 4 software for CORTEX. Writing Simple programs in Embedded C language. Compiling, Simulation & debugging the program. Burning the hex code in flash memory & testing it. Interfacing discrete LED's, Seven segment LED's, Binary Counter & Decimal counter
- Interfacing LCD
- Keyboard Interface
- ADC/DAC Programming
- UART Communication
- Timers Programming
- Interrupt Handling
- I2C & SPI
- RTC & EEPROM