

# **Advanced Embedded System Design and Application**

## **Part – 1: Programming Languages**

### **Module 1 Introduction to Embedded Systems**

What are and Why Embedded Systems?, Types of Embedded System, Classifications of Embedded Systems, Characteristics of an Embedded System, Applications of an Embedded Systems, Overview of Embedded Industry, Comparing Layers of GPC & Embedded System

### **Module 2 C Programming**

What are Micro Processors?, What are Micro Controllers?, Difference between Micro controller and Microprocessor, System Design using Microcontroller and Microprocessor, Difference between Von Newman and Harvard Architectures, Difference between CISC & RISC Architectures, Overview of 8 bit, 16bit, 32bit & 64bit microcontrollers

### **Module 3 Advanced C Programming**

Function, Storage Classes, Scope and Lifetime of a variable, Volatile, Recursive Functions, Stack Frame Analysis, Drawbacks of Functions, GDB, Basic Debugging Commands, Debugging a sample C Program, Arrays, Arrays and Functions, Pointer, Pointer Arithmetic, Pointers and Arrays, Pointers and Functions, String Handling Function like strcpy, strcat, strcmp, strlen, Pointers and Strings, Function Pointers, Variable Augmented Function,

### **Module 4 Data Structure Programming**

Dynamic Memory Allocation, Memory Leaks and Dangling Pointers, Structure, Structures and Array, Structures and Functions, Structure Padding and Alignment, Union, Bit fields, Using Typedefs, Enumerations, Macros

### **Module 5 Embedded C Programming**

Introduction to Data Structures, Types of Arrays, Coding Standard, Linked Lists, Singly Linked List, Doubly Linked Lists, Stacks and Queues.

## **Part – 2: Micro-Controllers**

### **Module 6 Basics of Micro controllers**

System Programming Vs Application Programming, Why C for embedded programming, Review of C language with embedded perspective, Bitwise operators programming, Bit field programming, Number System conversion program, Pointer Arithmetic programming, Register manipulation, Mixing Assembly and C.

### **Module 7 8051 Microcontroller MCS-51 Programming**

8051 Architecture, AT89C51 specifications, Register architecture, Special function registers, KEIL IDE, ProgISP, Flash Magic, LED programs, switch, 7 segment, LCD and scrolling LCD, Keypad, DC and Steeper motor, UART Programming, Timer and Counters, Interrupt programming, Relay, PWM, MOSFET, Opt couplers, Parallel ADC ADC0804, ADC0809, Serial ADC, I2C RTC DS1307.

### **Module 8 Sensor Programming with 8051 Microcontroller**

Temperature Sensor LM35, LDR Light detection Sensor, Ultrasonic distance sensor, PIR Motion detection sensor, humidity sensor, Level Sensor, MEMS Sensor, Gas Sensor,

---

## **Module 9 ARM 7 Microcontroller LPC2148 Programming**

ARM Architecture, LPC2148 specifications, Register architecture, Special function registers, KEIL IDE, Flash Magic, GPIO Programming, LED programs, switch, 7 segment, LCD and scrolling LCD , Keypad , DC and Steeper motor , UART Programming, Timer and Counters, Interrupt programming, Relay, PWM, MOSFET, Opt couplers, Parallel ADC ADC0804, ADC0809 , Serial ADC, Temperature Sensor LM35, LDR Sensor, Ultrasonic sensor, I2C RTC DS1307, SPI, USB.

## **Module 10 ARM 9 Board Bring up & Porting Embedded Linux**

Understanding ARM development Board, Porting of Bootloader, Porting of Kernel , Porting of File System on Development Boards, Accessing Development board using UART cable, SSH and Ethernet cable.

## **Module 11 ARM 9 Architecture**

ARM 9 Architecture, ARM 9 Instruction Set ( ISA-v5TEJ ), DNW tool, USB Push, Arm-linux-gcc, setting Qt library, Setting Opencv library ,Understanding GPIO pins and ADC pins. GUI Programming using QT, Image and video processing using Opencv.

## **Module 12 Peripherals Programming on ARM 9 Board**

Programming I/O Ports, Interfacing and Programming LEDs, Interfacing and Programming LCD, Interfacing and Programming KEYPAD, Interfacing and Programming ADC, Interfacing and Programming STEPPER MOTOR, Interfacing and Programming DC MOTOR.

## **Module 13 QT framework GUI Application Development on ARM 9**

Qt creator installation, Qt creator configuration, Setting cross compiler environment in Qt, qmake role in programming, writing GUI application programming for RFID reader, GPS module.

## **Module 14 ARM Cortex Microcontroller LPC1768 Programming**

Cortex Architecture, LPC1768 specifications, Register architecture, Special function registers, KEIL IDE, Flash Magic, GPIO Programming, LED programs, switch, 7 segment, LCD and scrolling LCD , Keypad , DC and Steeper motor , UART Programming, Timer and Counters, Interrupt programming, Relay, PWM, MOSFET, Opt couplers, Parallel ADC ADC0804, ADC0809 , Serial ADC, Temperature Sensor LM35, LDR Sensor, Ultrasonic sensor, I2C RTC DS1307, SPI, USB, CAN.

## **Part – 3: Wireless Modules**

### **Module 15 GPS Wireless Module Programming**

Architecture | NEMA Standards | GPS Sentences | GPGGA Sentence | UTC Time | Latitude, Longitudes and Altitude extraction | Digitalized Map

### **Module 16 GSM Wireless Module Programming**

Architecture | Frequency Bands | International Standards | AT Commands | SIM card Role | SMS / MMS commands | GSM Module Interfaces | GUI Programming

---

## **Module 17 RF Wireless Module Programming**

Architecture | Frequency Bands | Range calculation | Data Encryption | Data Decryption | Send and Received Program

## **Module 18 Infrared (IR) Wireless Module Programming**

Frequency bands | wavelength | Active Infrared | Passive Infrared | Modes | Obstacle and Reflection | RC5 Protocol

## **Module 19 Bluetooth Wireless Module Programming**

Architecture | Commands | IEEE standards | Radio interference | L2CAP profile | Pairing | Security | Data transfer programming

## **Module 20 RFID Wireless Module Programming**

Radio Frequency | Tag / Transponders | Readers | Data Processing subsystem | Types of Tag | Near Field Communication | Active and Passive Tag | Interfacing and programming with microcontroller

## **Module 21 WiFi / WLAN Wireless Module Programming**

Radio Architecture | IEEE standards | Frequency bands | Bandwidth | TCP / IP Layers Accessing in programming | Receive and Transmit Programming

## **Module 22 Zigbee Wireless Module Programming**

Communication Interfaces | Protocol Layers | Radio Architecture | IEEE standards | Frequency bands and range | Bandwidth | Programming with microcontrollers.

## **Module 23 Touch Screen Panel Module Programming**

Touch Screen Panel Architecture | Resistive and capacitive nature | 4 wire resistive touch panel construction and working | Programming the Touch Screen Panel | Application of Touch Screen Panel in Projects.

## **Part – 4: Real –Time OS**

### **Module 24 RTOS Concepts**

Introduction to Real Time Operating System, Classifications of RTOS, Misconception of RTOS, Features of RTOS, RTOS Architecture, Kernel, Task Management, Task Synchronization & Inter task Communication, Memory Management, Timer Management, Interrupt & Event handling, Device i/o Management, Selection of RTOS, Technical Consideration, Commercial Considerations.

### **Module 25 FreeRTOS**

Board support package of FreeRTOS for LPC2148, Compiling FreeRTOS projects in Keil IDE, Understanding the functions related to FreeRTOS, Led programming with Free RTOS, Serial port programming with FreeRTOS, Priority and preemptive programming with Free RTOS, Message queue, semaphore and timedelay programming with Free RTOS.

### **Module 26 UCOS-II RTOS**

---

Understanding Board support packages of ucous RTOs for LPC2148 board, Compiling and Configuring ucous environment in keil IDE, Understanding the functions defined in ucous RTOs for priority, task handling and preemptive programming, led and UART programming with ucous RTOS. Message queue, semaphore and timedelay programming with ucous RTOs.

### **Module 27 RT Linux**

Installation of RT Patches on Linux Kernel, Configuration and compilation of RTLinux Kernel, Booting of RTKernel. Led programming in RTLinux, Understanding the API of RTLinux.

### **Module 28 CAN Protocol Programming**

CAN History, CAN protocol standard, Understanding CAN protocol Architecture, Layers in CAN protocols, CAN Kingdom, CAN open, CAN Programming with Cortex Board.

### **Module 29 Project Development with 8051 /ARM7**

### **Module 30 Project Development with ARM9 / ARM Cortex**

## **Part – 5: Linux System Programming & Device Drivers**

### **Module 31 Linux Basic and Utilities**

Unix / Linux / Windows History, Free Software Foundation, GNU Project, Booting Process of OS ,Advantages and disadvantages of Linux, Features of Linux, Linux root File System, Virtual File system, Linux Kernel version, Shell and Shell Commands , system calls, file handling in Linux, Environmental variables, inodes and file permissions, error handling in linux, data types in linux.

### **Module 32 Programming Language Libraries in Linux**

Shell Programming, Glibc, gcc compiler, gdb debugger, Makefile Utility, Compilation steps, Memory allocation in programming, C++ Library and manpages, Java Library support in Linux, Python and perl Langaguge programming in Linux.

### **Module 33 Linux Kernel Programming**

Linux Kernel Architecture, User & System Mode, Process Management, pid, fork , vfork, exec, orphan process, zombie process, wait, exit, POSIX standards, pthreads, pthread attributes.

### **Module 34 POSIX Thread Programming**

POSIX standards, POSIX Threads, pthreads, pthread attributes, POSIX Message Queues, POSIX Semaphores, POSIX Shared memory , Asynchronous input and output, Threads in real-time, Multithread Vs Multiprocessing.

### **Module 35 Linux Synchronization Programming**

Need of Synchronization, Deadlock problem ,race condition, critical section problem, Atomic operators, Deadlock, conditional variables, Mutex, Semaphores, System v API, POSIX API.

---

## **Module 36 Linux IPC (Inter Process Communication) Programming**

Need of IPCs, pipe, FIFO, Shared Memory, message queue, Signals

## **Module 37 Linux Network Programming OSI Reference Model**

Introduction to OSI Ref Model, Physical Layer Protocol Functions , Data Link Layer Protocol Functions , Network Layer Protocol Functions , Transport Layer Protocol Functions , Session Layer Protocol Functions , Presentation Layer Protocol Functions, Application Layer Protocol Functions

## **Module 38 Understanding TCP / IP**

Origin of TCP/IP, TCP/IP Model, IP Addressing, Subnet masking, Assigning IP, Overview of Routing process, TCP/IP Stack, TCP/IP Characteristics, TCP/IP Utilities

## **Module 39 Linux Socket Programming**

Socket API, Client Server Architecture, TCP Socket, UDP Socket, RAW Socket, Unix Socket, FTP Programming, Chat Programming, Concurrent Server Programming, Multiple client Programming, Arithmetic server.

## **Module 40 Fundamentals of Device Drivers**

Linux Driver Module ,Role of Driver,Inbuild and Modular Drivers, Types of Device Driver , Relationship between Kernel and Driver, Major and Minor Number, Modular Program Vs. Application Program , Compile, run a sample module, Registering & Un-registering a sample character device , Creating a device file and test character device driver, Hands on Sample module creation, passing arguments to module.

## **Module 41 Developing Character Device Drivers in Linux**

Memory Allocation and Freeing with in modules, IO Port and IOMEM allocation, Registering and Freeing Interrupt Handling , Random Access to the character device, Using Capabilities , Implementing IOCTL Commands , Implementing Kernel Locking Mechanism , Overview of Network , Device Drivers, Overview of Block Device Driver

## **Module 42 Developing Block Device Drivers in Linux**

Driver Registration, Block Device Operation, gendisk structure, Block Device Operation, Request queue, bio structure, request structure, sample driver

## **Module 43 Understanding & Driving UART / RS-232**

Difference between Serial and Parallel Communication, What are DTE and DCE, What are DB9 and DB25 Connectors, Describe the pins of DB9 Connector, What are loop back and Null modem connection, Serial Port Architecture under X86 Architecture, Description of UART Register in X86 Architecture, RS232 Frame Format, Max232/233, a voltage level converter, Programming and Driving RS-232 serial port

## **Module 44 Parallel Port Device Driver**

Difference between Serial and Parallel Communication, DB25 connector and Centronix connector, Data Register, Control Register, Status Register, Data Input / Output using parallel port, Base Address Accessing , Practical device driver for parport.

## **Module 45 USB Keyboard Device Driver**

USB specifications, USB standards, USB Layers and Linux USB subsystems, Topologies used in USB,OHCI, UHCI and EHCI type of USB controller ICs. Commands related to USB drivers, Registering the USB driver, USB Urbs, Practical USB device driver.

---

### **Module 46 USB Mouse Device Driver**

USB mouse specifications, 2 button and 3button mouse, scroll button architecture, PS2 and USB subsystems, USB wrapper driver programming on PS2 mouse programming, Inserting USB Mouse driver in running kernel, hot swapping architecture, Testing working of our driver with practical connection.

### **Module 47 USB Pen Drive Device Driver**

Pendriver Architecture, Flash memory commands, Storage architecture of data on Flash memory, Accessing and writing data on flash ICs, USB wrapper driver on MMC driver, Auto displaying drive name in file system, Practical Pen drive driver.

### **Module 48 Developing Network / Stream Device Drivers in Linux**

Comparison of char and Network driver, Driver Registration, net\_device structure, net\_device\_status structure, skbuff structure, network device methods, sample driver

### **Module 49 Embedded Linux based Project on ARM9**

### **Module 50 Embedded Linux based Project on ARM Cortex**