EMBEDDED SYSTEM COURSE CONTENT

C Language

- Why C in Embedded
- ANSI Standard
- Fundamentals of C
- Datatypes and Constants
- Simple & Formatted I/O
- Memory Usage
- Operators & Expressions
- Flow Control
- Loops

Functions

- Role of Functions
- Pass by value / reference
- Returning values from Functions
- Recursive Functions
- Call Back Functions
- Implications on Stack
- Library Vs User defined function
- Passing variable number of arguments

Arrays

- Defining, initializing and using arrays
- Multi Dimensional Arrays
- Arrays of Characters and Strings
- Arrays and Pointers
- Passing arrays to functions
- String handling with and without library functions

Storage Classes

- Scope and Life
- Automatic, Static, External, Register
- Memory(CPU / RAM)

Structures & Unions

- What structures are for
- Declaration, initialization
- Accessing like objects
- Nested Structures
- Array of Structures
- Passing structures through functions
- Allocation of memory and holes
- Structure Comparison
- Structure bit operation
- Typedef for portability
- Unions
- Overlapping members

Enumerated data types

- Enum, Indexing, enum Vs #define

Bit Operations

- AND (&), OR (|), XOR (^)
- Compliment (~)
- Left-Shift (<<), Right Shift (>>)
- Masking, Setting, Clearing and Testing of Bit / Bits

Pointers

- The purpose of pointers
- Defining pointers
- The & and * operators
- Pointer Assignment
- Pointer Arithmetic
- Multiple indirections
- Advanced pointer types
- Generic and Null Pointer
- Function Pointers
- Pointers to Arrays and Strings
- Array of Pointers
- Pointers to Structure and Union
- Pointers to Dynamic memory
- Far, Near and Huge Pointers
- Pointer Type Casting

Dynamic Memory Allocation

- Malloc(), Calloc(), Realloc(), Free()
- Farmalloc(), Farcalloc()

File Handling Concepts

- Concept of a FILE data type
- Inode, FILE structure
- File pointer
- Character handling routines
- Formatted Data Routines
- Raw data Routines
- Random Access to FILE

Command line Arguments

- Argc, argv
- Variable Inputs to the main

Compiler in Practical

- Preprocessor Directives
- Compiler, Assembler, Linker
- Conditional Compilation
- Multiple File Compilation
- Code Optimization techniques
- Volatile, #pragma

Data Structures

- Linear & non-linear
- Homogeneous & non-homogeneous
- Static & Dynamic
- Single, Double & Circular Linked Lists
- Stacks & Queues
- Binary Trees

Sorting and Searching Techniques

- Insertion, Selection, Bubble, Merge, Quick, Heap

Concepts and Real-time Exposure

- Development Tools and Environment
- Make Utility
- Industry Coding Standards
- Object / Executable File Format
- Debugger

MICROCONTROLLER AT89C51

Introduction

- Microprocessor vs Microcontroller
- CISC vs RISC

Overview of Architecture of 8051

- Processor Core and Functional Block Diagram
- Description of memory organization
- Overview of ALL SFR’s and their basic functionality

Low-level Programming Concepts

- Addressing Modes
- Instruction Set and Assembly Language (ALP)
- Developing, Building and Debugging ALP’s

Middle Level Programming Concepts

- Cross Compiler
- Embedded C Implementation, prog. * Debugging
- Differences from ASNSI-C
- Memory Models
- Library reference
- Use of #pragma directive
- Functions, Parameter passing and return types

On-Chip Peripherals

- Ports: Input/output
- Timers & Counters
- Interrupts, UART

External Interfaces

- LEDS
- Switches (Momentary type, Toggle type)
- Seven Segment Display: (Normal mode, BCD mode,
- Internal Multiplexing & External Multiplexing)
- LCD (4bit, 8bit, Busy Flag, Custom Character Generation)
- Keypad Matrix

Protocols

- I2C (EEPROM), SPI (EEPROM)

Keil's RTX51 Tiny / Pumpkin's Salvo

- Overview
- Specifications
- Single-Tasking Programs
- Multi-Tasking Programs
- RTX51 Tiny Programs
- Theory of Operation
- Timer Tick Interrupt
- Task Management & Scheduler Events
- Round-Robin & Co-operative Task Switching
- Idle Task
- Stack Management
- Function Reference
- Porting on to H/W
- Implementation Examples

Selective Discussion during Project Development

- A/D & D/A Converter
- Stepper motor, DC Motor
- I2C Protocol (RTC:800583,DS1307 ADC:PCF8591, DS1621)
- SPI Protocols (ADC:MCP3001)
- IR Communications (Phillips RC5 Protocol)
- ZIGBEE, GSM, GPS, USB, MMC & SD
- Ethernet MAC, CAN Protocol

Real Time Operating System

RTOS RT-LINUX

- RT- Linux
- Different types of Operating systems
- RTOS basics – Linux as Real Time
- RTOS Introduction (Hard Real Time, Soft Real time)
- Latency in Linux, Priority Inheritance
- Linux 2.6 features for realtime
- 2.6 Kernel Compilation
- RT LINUX patching
- Linux RTPREEMPT Patches
- Configuring the Kernel with RT-PATCH
- Implemantation of Real Time application
- Linux real-time API
- Measuring and camparing scheduling latency in standard Linux and in RT-Linux with the latest RT patches
- Porting RT-Linux on ARM and application development

Object Oriented Programming with C++

- Overview
- Characteristics
- Function Overloading
- Scope Resolution Operator
- Classes in C++
- Access Specifiers
- Constructor, Destructor
- Static members, Functions
- Friend Classes, Friend Functions
- Operator Overloading
- Data Conversions
- Inheritance, Polymorphism
- Exception Handling, Templates
- Input and Output Streams

IoT on Raspberry Pi

Introduction to the “Internet of Things”

Introduction to Raspberry Pi

- Tools : Win32ImageWrite & SDFormatter
- Loading Raspbian OS image on SD CARD
- Demo programs using C
- Backing Up Updated SD CARD & OS image
- Installing Wiring Pi package
- Controlling the Raspberry Pi 2 GPIO by command line

Networking with Pi

- Client – Server programming for Automating Device / sniffing Device State

IoT -Cloud

- Installing and configuring PubNub SDK for C language
- Creating credentials with PubNub Cloud Server
- End to End IoT program demonstrations

Project on Embedded IoT

ARM 7

- Introduction
- Core Features
- Version History
- Data Flow Model
- Registers
- CPU Modes
- Memory Organization
- Interrupts
- Pipelining
- ARM Assembly Language Programming
- Addressing Modes
- ARM 7 Instruction Set (20/80% -rule of assembly language)
- Usage of Keil IDE

- Demonstrating ARM ISA
- Demonstrating THUMB ISA
- ARM Embedded C language Implementation
- Exposure to an ARM7 CPU Core Based Microcontroller
- LPC2114-ARM7 Based Microcontroller from Philips Semiconductors
- On-Chip System Peripherals
- Bus Structure (AMBA)
- Memory Map
- Phase Locked Loop
- VPB Driver
- Pin Connect Block
- On-ChipUser Peripherals
- General Purpose I/0 : Demo using switch & LED
- Vectored Interrupt Controller (VIC)
- External Interrupts : Demos