PG Diploma in Embedded Systems & Design

Embedded C Programming - 120 Hours

Basics of Program Writing & Coding Practices, Overview of C Programming language, Introduction to GNU Toolchain and GNU Make utility, Linux environment and vi editor, Tokens of C - Keywords, Data-Types, Variables, Constants, Operators, Identifiers, Storage Class Specifiers, Control Flow Statements, Arrays, Multidimensional arrays, Data Input & Output, Strings, Loops, Functions and Recursion, Pointers - Introduction, Pointer Arithmetic, Pointers and Arrays, Pointers and Functions, Pointers and Strings, Structures, Unions, Enum, Typedef, Bit field operators and pointers with structures, Preprocessors, C and Assembly, Files, I/O, Variable number of arguments, Command Line arguments, Error handling, Debugging and Optimization of C programs, Bit operations, Handling portability issues in C, Hardware, Time, Space and Power aware Programming.

Data Structures and Algorithms - 90 Hours

Introduction to Data Structures, Algorithms and Abstract Data Types, Complexity of Algorithms, Linked Lists, Stacks, Queues, Searching and Sorting Algorithms, Hashing, Trees.

Microcontroller Programming and Interfacing - 120 Hours

Overview of Microcontrollers, Microprocessors and SoC, RISC vs CISC, Harvard vs Princeton Ar- chitectures, Overview of Computer Architecture, Embedded Memories, Timers/Counters, UART, SPI, PWM, WDT, Input Capture, Output Compare Modes, I2C,
CAN, LED, Switches, ADC, DAC, LCD, RTC, Bus Standards (USB, PCI), Programming in Assembly and Embedded C.

ARM: Overview of ARM Architecture and Organization, Introduction to Cortex-M Architecture, Programming Model and Instruction Set Architecture, Alignment and Endianness, Register access, State, Privilege, Stack, System Control Block, Power Modes, Memory Model, NVIC, Exception Handling, Bit- Banding, Peripheral Programming, SVCall, SysTick, PendSv, MPU, DMA, Mixing Assembly and C programs, Introduction to CMSIS & CMSIS Components, Overview of Cortex A & R architectures.

RISC V: Why RISC-V processor, RISC-V processor overview, ARM vs RISC-V, Modes in RISC-V, Setting up of necessary tools, RISC-V register set and calling convention, Instruction formats and type, Build Process, Practical examples of instructions, Detail description on Control and Status Registers, Exception handling, Examples in assembly

for exception handling, Interrupts, Interrupt Entry and Exit procedure.

Introduction to C-DAC VEGA processors

Embedded Operating Systems - 90 Hours

Introduction to Embedded Operating Systems, Anatomy of an Embedded Linux System - Boot- loader, Kernel, Root File System, Application -, Process Management, Interprocess Communi- cation & Synchronization, Memory Management, I/O subsystem & Embedded File Systems, POSIX Thread Programming, POSIX Semaphores, Mutexes, Conditional Variables, Barriers, Mes- sage Queues, Shared Memory, Debugging

and Testing of Multithreaded Applications, Socket Programming, Customizing Embedded Linux based on Yocto, Virtualization: Dockers & Containers.

Embedded Device Driver - 90 Hours

The Embedded Linux Software Eco-System, Linux Kernel Modules and Module Programming, Char Device Drivers, Kernel Internals: Dynamic memory allocations, Handling Delays, Timers, Synchronization, Locking, I/O Memory and Ports, Interrupts, Deferred Executions, Driver De- bugging Techniques, Drivers for GPIO, I2C, and SPI, Pseudo Filesystems (procfs, sysfs).

Real-time Operating Systems - 60 Hours

Introduction to Real-Time Concepts, RTOS Internals & Real Time Scheduling, Performance Met- rics of RTOS, Task Specifications, Schedulability Analysis, Application Programming on RTOS, Porting of RTOS, Configuring RTOS, Building RTOS Image for Target platforms.

Internet of Things (IoT) - 90 Hours

IoT: IoT Trends, IoT Architecture, IoT Applications, IoT Standards and Protocols, Wireless LAN: IEEE 802.11, Wireless PAN: IEEE 802.15.1 & 802.15.4, Zigbee, Bluetooth, BTLE, LPWAN (LoRa, NB- IoT), 6LowPAN, REST, CoAP, MQTT, Basics of Cryptography, Overview of IoT and Embedded security, Overview of 5G technologies.

Embedded AI: AI Fundamentals, Supervised Learning, Unsupervised Learning, Ensemble Techniques, Time Series Forecasting, Neural Networks and Deep Learning, Embedded AI applications, Embedded AI frameworks (CMSIS-NN, AIfES, TensorFlow- Lite, TensorFlow-Lite Micro etc), Feature Engineering, Model Selection & Tuning, Development and deployment of embedded ML models, Case Study on Embedded AI.

Aptitude & Effective Communication - 90 Hours

Percentage, Profit & Loss, Ratio & Proportion, Average, Mixture & Allegation, Simple Interest & Compound Interest, Number Systems , Series, Cyclicity & Remainders, Data Interpretation, Syllogism, Coding & Decoding, Blood Relations, Seating Arrangements (Linear & Circular), Ages, Puzzles, Time, Speed & Distance, Trains, Boats & Streams, Time & Work, Wages (Man days), Pipes & Cisterns, Clocks, Permutations & Combinations, Probability, Calendar Fundamentals of Communication, The Art of Communication, Personality Development, English Grammar, Correct Usage of English, Common Mistakes in English Communication, Listening Skills, Reading Skills, Writing Skills, Public Speaking, Presentation Skills, Group Discussions, Interpersonal Skills, Personal Interviews

Project and Seminar - 150 Hours

Students are required to execute project work for the duration of four weeks (after the completion of all modules) as a part of this course. For seminar, students need to choose the topic themselves and give the seminar on the respective dates allocated by the concerned faculty members. The topic chosen by the students should be relevant to the Embedded Systems Design. Project work is distributed in the following phases:

1. Study and Requirements Elicitation
2. Design
3. Implementation
4. Testing
5. Project report
6. Viva Voce and Presentation

Students need to submit a project report at the conclusion of the project. Mentors should be allocated within 3 weeks of the course commencement and should be executed throughout the course duration. The students should maintain a logbook, which contains their day-to-day activities during the project phases. The mentor allocated for that project should sign this logbook regularly. The allocated 4 weeks should be focused on implementation, testing and consolidating the documentation.