

Course Aims:

The Programming Microcontrollers in C course is designed to teach the C language from scratch. The aim of the course is to give the delegate the ability to write programs that are suitable for use in a variety of embedded environments.

Course Outline:

Introduction to C Programming: Structure of a C program The C compilation process Types and Operators: C base types Precedence & associativity Arithmetic operations General Purpose IO (GPIO) Control Flow: Logical expressions and operations Decision Making Loops Arrays, Structures and Unions: Arrays Arrays as circular buffers Structures Unions Bit-field structures Big & Little Endian Functions: The Function as a logical program unit How parameters are passed Memory segments Pointers: Pointer basics C string handling Relationship between pointers & arrays Pointer arithmetic Dynamic Memory Management: Malloc and free Linked list Issues (leaks, fragmentation, etc.) Program Structure: Definitions and declarations Header files Scope and lifetime The C Pre-Processor: Macros Conditional Compilation	The Standard C Library Introduction to Real-Time Systems: Characteristics and demands of realtime software development Bit Manipulation & Hardware Access: Accessing hardware with pointers Using structures to access I/O devices Manipulating information at the bit level Interrupts: Internal & External interrupt control Nested Interrupts Priorities & Masks Software Interrupts Device Interaction & Synchronization: Polling devices Serial peripheral programming Program Design Concepts: Why modular design? Coupling and cohesion Testing the program Concurrency: Scheduling strategies Context switching Function pointers Task synchronisation and communication MMU and memory protection Concurrency and libraries Target Specific Considerations: Language features affecting portability Non-standard C language features Assembly language interfacing What happens at power-on? Initialising the C runtime
---	--

Target Audience:

Systems and applications programmers who will be developing Embedded systems in C. Anyone who wants a practical understanding of C will benefit from this course. It is suitable for hardware and software engineers who want to expand their knowledge in a powerful all-purpose language, technical managers who want to manage Embedded C programming projects.

Assumed Knowledge:

Participants should have a good knowledge of programming techniques and at least one programming language.