

## **Embedded C Training: Programming Methods and Tools for Embedded Applications - Face-to-Face Training**

This training focuses on the hardware-near C-programming of 8, 16 or 32 bit microcontroller architectures. It shows you how to identify and avoid the pitfalls of C programming .You learn how to program a HW abstraction layer according to an architecture model. Operating system mechanisms and services are explained by programming a scheduler. You get an overview of the whole lifecycle of a product - from the idea to project planning, software development process, test planning, quality planning, acceptance, commissioning, operation and decommissioning.

### **Objectives**

This training focuses on the hardware-near C-programming of 8, 16 or 32 bit microcontroller architectures.

You learn how to program a HW abstraction layer according to an architecture model. OS mechanisms and services are explained by programming a scheduler.

You get an overview of the entire lifecycle of a product - from the idea to project planning, SW development process, test planning, quality planning, acceptance, commissioning, operation and decommissioning.

You are able to efficiently develop programs for an embedded system in "C" according to the guidelines of modern software engineering.

You are familiar with using pointers, function pointers and structures.

Based on your knowledge of programming/coding guidelines and software quality features, functional and non-functional requirements as well as internal quality and generate software that is reusable, extendable and easily tested.

In addition, you know all stages of a software development process, from the idea to system acceptance.

### **Participants**

Software developers, software architects

### **Requirements**

A good understanding of ANSI-C and microcontroller architectures.

## **Embedded C Training: Programming Methods and Tools for Embedded Applications - Face-to-Face Training**

### **Content**

#### **Introduction**

- ANSI-C
- Embedded systems and their specifics
- Software toolchain
- Software architecture
- Debug features and bugs

#### **Programming Language C for Embedded**

- Hardware-near programming
- Data types
- Pointers, function pointers

- Structures, linked lists
- Circular buffer, queue, FIFO, LIFO
- Programming rules and guidelines
- Pitfalls and stumbling blocks in C

**Driver Programming**

- Selecting a suitable SW architecture
- HW abstraction, object-based programming
- Access to HW registers from "C"
- Interfaces, callback interfaces, queues
- Interrupt handling /service routines, callback function
- Exercises: Timer hardware abstraction plus callback

**Using Pointers, Function Pointers and Linked Lists**

- Programming example - scheduler
- Task management with linked lists
- Exercises: Programming a task management

**Real-Time Operating Systems (RTOS) - Overview**

- Types, services, selection criteria
- Function and programming of a scheduler
- Exercises: Task switch

**Library Management**

- Adapting standard library functions to hardware
- Generating and managing user libraries
- Exercise: Generating and integrating a library

**Locating Code and Files in the ( $\mu$ C) Memory (Flash, RAM Address Space)**

- Logical sections (.text, .data, .bss) in the build process
- Load and run addresses
- Controlling the linker through command files

**Finite State Machines, FSM**

- Descriptions and representation variants
- Philosophy and implementation of an FSM in C
- Exercises: Programming a traffic light control

**Aspects of Embedded Software Engineering**

- Software quality criteria
- Software development process models (Waterfall, V, agile)
- Functional safety
- Requirements engineering
- Verification and test
- Capability maturity models

**Outlook: OOP Techniques**

- Advantages and challenges of object oriented programming
- UML diagrams

**Coding Guidelines**

- Purpose
- MISRA-C directives and rules

**MicroConsult Plus: Extensive Exercises on a Target Hardware**

- The hands-on exercises are performed and tested using the Keil  $\mu$ Vision IDE and Arm compiler on an M0-based 32-bit hardware platform.

**FACE-TO-FACE TRAINING**

Date	Price *	Duration
15.09.2025 – 18.09.2025	2.400,00 €	4 days
19.01.2026 – 22.01.2026	2.400,00 €	4 days

\* Price per attendee, in Euro plus VAT

Training code: E-EMB-C

### **Live Online - English**

<b>Date</b>	<b>Duration</b>
08.12. – 11.12.2025	4 days
16.03. – 19.03.2026	4 days

### **Face-To-Face - German**

<b>Date</b>	<b>Duration</b>
15.09. – 18.09.2025	4 days
19.01. – 22.01.2026	4 days

### **Live Online - German**

<b>Date</b>	<b>Duration</b>
23.06. – 26.06.2025	4 days
08.12. – 11.12.2025	4 days
16.03. – 19.03.2026	4 days

### **Coaching**

Our coaching services offer a major advantage: our specialists introduce their expertise and experience directly in your solution process, thus contributing to the success of your projects.

We will be happy to provide you with further information or submit a quotation tailored to your requirements.