FEABHAS

Robust Software for Embedded Systems

Course category	Software engineering
Training area	Quality
Course code	RB-201
Duration	2 days
Additional information	For price: Contact Feabhas (available for on-site delivery only)

Across the spectrum of embedded systems there is a universal need for reliability and robustness beyond that required for desktop computer software. Failure detection and recovery is vital and the software must be designed so that it can run continuously without the need for a reboot. This course explores the accepted industry best practices for achieving that extra level of quality in your embedded software.

These techniques include the use of watchdog hardware to detect a deadlocked system. Error handling mechanisms should be capable of reporting the exact location of a software bug, even after the system has been deployed. Memory management techniques need to ensure that a memory leak or stack overflow will not lurk like a time bomb in a long-running system. Programmers need to be aware of real-time pitfalls when using interrupts and an RTOS.

This course is based on 10 years of experience writing embedded software for life-critical medical equipment. It has been developed by Niall Murphy, author of “Front Panel: Designing Software for Embedded User Interfaces” and regular contributor to Embedded Systems Programming magazine.

A two day course providing an introduction to the building blocks of a dependable embedded system.

Course objectives:

After completing the course, attendees will:

- Understand the processes that can be applied to ensure quality code, such as code inspections and Hazards Analysis
- Understand how to write defensive code and code which allows a device to test itself
- Understand how to make the best use of watchdog timers
- Understand how to organise non-volatile storage to avoid corruption and loss of data
- Understand the advantages and weaknesses of using features such as dynamic memory management and real time operating systems

Pre-requisites:

A working knowledge of C.

Who should attend:

Application programmers, software engineers or technical staff who need to address reliability or robustness issues at a code level. The course is ideal for engineers who can write software, but are new to embedded systems, especially if those embedded systems have safety concerns or require high reliability.

Duration:

- Two days

Course materials:

- Delegate handbook

Introduction

- Definitions for Safety, Availability, Reliability
- Graceful degradation
- Industry magic bullets

Data Integrity Checks

- Checksums
- CRCs

Persistent Storage

- Error Detection
- Double Buffering
- Versioning

Built-in Self-test

- RAM/ROM/address line tests
- Loopback tests
- Cable and subassembly tests

Interlocks

- Hardware limitations on software actions
- Software limitations on user actions

Exception Handling and Asserts

- Language Support
- Assert Macro
- Debug code issues

Memory Management

- Static Allocation
- Stack Measurement
- Malloc and fragmentation
- Pools
- Detecting Leaks

Interrupts

- Reentrancy
- Motivation
- Parallelism
- Timed Polling

Real Time Operating System

- Tasking
- Decomposition
- Preemptive Kernels
- Priority Inversion
- Queuing
- Timing Accuracy

Feabhas Ltd - UK Office 15-17 Lotmead Business Park, Wanborough, Swindon, SN4 0UY, UK

+44 (0) 1793 792909 • info@feabhas.com • www.feabhas.com