Clean Code for Embedded

Duration: 3 Days
Available Languages: English German

Audience

Software Craftsmen: Embedded Software Developers, Embedded Architects, Project Managers, Scrum Masters.

Precondition

Good knowledge of C.

Goals

Learn how to develop embedded software using the Clean Code principles.

Contents

- Design Smells: Rigidity, Fragility, Immobility, Viscosity
- Core Architecture: Coupling and Cohesion
- The two values of Software
- Fundamentals: Source Code Structure, Naming
- TDD - Test-Driven Development: The Three Laws of TDD, Red-Green-Refactor Cycle, BDD, TPP
- The SOLID Principles: 11 Principles of Clean Architecture
- MISRA C
  - # MISRA C:2012
  - # Criticism and limitations of MISRA C
- CMU SEI CERT C Coding Standard
  - # Rules vs Recommendations
  - # Preprocessor
  - # Declarations and Initialization
  - # Expressions
  - # Integers
  - # Floating Point
  - # Arrays
  - # Characters and Strings
  - # Memory Management
  - # Input Output
  - # Environment
  - # Signals
  - # Error Handling
  - # Application Programming Interfaces
  - # Concurrency
  - # Miscellaneous

    - # POSIX
    - # Microsoft Windows
- Stack Measurement
- Tools
    - # Compiler Warnings
    - # PC-Lint
    - # splint
    - # AceUnit
- Specific Embedded Aspects
    - # Footprint
    - # Performance
    - # Safety
    - # Security
    - # Interrupts
    - # Dealing with NVM and its limitations

Examples will be performed on the CPU architectures AMD64 and ARM. This training can be customized for specific CPUs and microcontrollers, for example: 6502, 6800, 68k, 8051, 80251, 8080, 80x86, AMD64, AVR, CalmRISC, ColdFire, CPU32, H8, MIPS, PowerPC, SECUCALM, SPARC, TriCore.

Booking

Contact Siddhesh Nikude, +91-95-52572354, training@nelkinda.com