As of 19.05.2024

Armv8-R Workshop: 32-Bit Cortex® Microcontroller Core for Real-Time Applications in Automobiles, the Industry and Embedded Systems - Live Online Training

Objectives

You know the architecture, features and advantages of the Armv8-R (real-time) core architecture.

You learn how to choose and make efficient use of the right microcontrollers with the respective cores.

You can write software in C and Assembler and utilize safety and security mechanisms - the perfect start for designing Cortex® v8-R based systems.

YOUR BENEFIT:

Efficient and compact jump-start into the overall topic

Practical tips on multicore, safety and security

Exercises on USB stick or as download

Extensive training documentation helps you apply and reproduce what you learned.

Participants

Hardware and software architects, hardware and software developers, test engineers

Requirements

ANSI-C knowledge; experience in microcontroller/microprocessor system programming and architecture

Live-Online-Training

* Price per attendee, in Euro plus VAT

Training code: LE-ARMV8R

Face-To-Face - English

Duration
4 days

Live Online - German

Duration
4 days

Face-To-Face - German

Date	Duration
02.07. – 05.07.2024	4 days

As of 19.05.2024

17.12. – 20.12.2024 4 days

Armv8-R Workshop: 32-Bit Cortex® Microcontroller Core for Real-Time Applications in Automobiles, the Industry and Embedded Systems - Live Online Training

Content

Overview Arm®v8 Real Time Profile

Cortex® R52 Processor Architecture

Instruction Sets

Synchronization and Barriers
- Exclusive monitors

Writing C for Arm

Exceptions and Handling
- Exception levels
- Interrupts and aborts
- Interrupt controller architectures
- Routing and trapping
- Nesting and returning

Timer
- Generic timer
- Performance monitor unit

Internal Connectivity and Memory
- Bus interfaces
- Tightly coupled memory and caches
- Performance aspects

Memory Protection Unit
- Purpose and use cases
- Regions and attributes

Safety Aspects
- Register protection
- Indirect memory access
- Dual core lockstep
- ECC
- Error reporting

Debug and Tracing

Virtualization
- Use cases
- Hypervisor
- Temporal and spatial separation
- Identification
- Asynchronous stimuli

Exercises