



## Cortex-R82 Software Development

### Course Description

Cortex-R82 is the highest performance real-time processor from Arm and the first to implement the Armv8-R AArch64 architecture.

The Cortex-R82 processor delivers the higher compute performance needed for complex data storage applications, including Computational Storage Drives (CSDs).

It's the first processor that enables real-time and Linux on the same core/cluster.

Cortex-R82 is ideally suited for 5G modems, where very high-performance and deterministic operations are the key components needed to meet the low-latency operations and deliver the high-throughput requirements in smartphones and laptops.

Cortex-R82 brings many new features such as virtualization, MMU, improved context switching, Security, safety, cache coherency, custom instructions, ML and DSP acceleration to name a few.

Cortex-R82 software development is a 4 days ARM official course.

The course goes into great depth and provides all necessary know-how to develop software for systems based on Cortex-R82 processor.

The course covers the Cortex-R82 architecture (Armv8-R AArch64), processor core, programmer's model, instruction set, exception handling, memory model, TCM, cache management, multi-core cache coherency, memory protection unit (MPU), memory management unit (MMU), NEON SIMD, virtualization, security, safety (RAS, ECC), debug & trace, custom instructions, Tightly Coupled I/O (TCIO), efficient C programming and compiler optimizations.

**At the end of the course the participant will receive a certificate from ARM.**

## **Course Duration**

4 days

## **Goals**

1. Become familiar with ARMv8-R architecture
2. Become familiar with Cortex-M82 architecture
3. Become familiar with ARMv8-R instruction set
4. Be able to handle interrupts and various exceptions
5. Be able to configure and use the MPU and MMU
6. Understand the memory model in v8-R architecture
7. Manage caches and TCM
8. Write an efficient C code for Cortex-M processor
9. Be able to debug your design
10. Become familiar with Arm custom instructions
11. Apply real-time virtualization and security
12. Accelerate ML and DSP algorithms with NEON and floating point
13. Programming the Generic Interrupt Controller (GIC625/700)

## Target Audience

Software engineers that would like developing software and Firmware for platforms based on Cortex-R82 processor.

## Prerequisites

- Computer architecture background
- C and Assembler
- Experience in developing embedded systems

## Course Material

- ARM official course book

## Agenda

### Main Topics:

- Cortex-R82 Overview
- ARMv8-R Programmer's Model
- Cortex-R82 Processor Core
- ARMv8-R Instruction Set Architecture (ISA)
- ARMv8-R Exception Handling
- ARMv8-R Memory Model
- Caches and TCMs
- Multi-core Cache coherency
- ARMv8-R Memory Protection Unit (EL1 and EL2)
- ARMv8-R Memory Management Unit
- ARMv8-R NEON and FPU Accelerators
- ARMv8-R Real-Time Virtualization & Security
- ARMv8-R Custom Instructions
- ARMv8-R Debug & Trace
- ARMv8-R Compiler Hints & Tips
- Programming the GIC