# ARM 7 / ARM 9 System Design

**Ref : 002879A** **Duration : 4 days**

## OBJECTIVES

- This course takes an **in depth** look at the considerations you will need to take into account when designing a system containing either an ARM7TDMI family or ARM9TDMI family processor core. Information on the latest generation of ARM processor cores, is also included.
- It is aimed at :
  - Software engineers who not only want to obtain details of how to write software to run on the ARM, but also wish to obtain an understanding of hardware design issues
  - Hardware engineers who need to understand how to design ARM based systems, but also wish to obtain an understanding of the issues of writing software to run on that system

## RELATED COURSES

- ARM7/9 Hardware design (Ref.002579A)
- ARM Software development using RealView (Ref.002580A)

## PARTNERS

- This training course is approved by ARM

## PREREQUISITES

- A basic understanding of microprocessors and microcontrollers is recommended
- A basic understanding of digital logic or hardware / ASIC design issues would be useful but not essential
- A basic understanding of assembler or C programming would be useful but not essential
- A basic awareness ARM cores is useful but not essential

## PRACTICAL LABS

- For on-site courses, labs can be run under the following environments : CodeWarrior/ADS/AXD, Eclipse/RVDS, Keil µVision, GNU/Lauterbach simulator, or IAR Workbench
- For open courses, labs are run under Eclipse/RVDS

APPROVED TRAINING CENTRE ARM

**Contact**

**Tel : +33 (0)5 62 13 52 32**
**Fax : +33 (0)5 61 06 72 60**
**training@mvd-fpga.com**

**Course also available customized**

**Next sessions, see : *http://www.mvd-fpga.com/en/formationsCalend.html***

## TOPICS

*First day*

### THE ARM ARCHITECTURE

- ARM operation modes
- The ARM registers set
- Program Status Registers
- Exception handling
- Instruction sets

### ARM PROCESSOR CORE

- ARM7TDMI core signals
- The ARM7TDMI instruction pipeline
- ARM7TDMI memory interface
- ARM9TDMI datapaths
- ARM9TDMI pipeline
- ARM11 Core
- ARM11 superscalar pipeline operation
- ARM11 dynamic vs static branch prediction
- ARM11 out of order execution
- ARM11 Return stack

### RealView DEVELOPPER SUITE (RVDS) OVERVIEW

- Using the core tools
- C/C++ compilers key features
- Supplied libraries
- Codewarrior introduction
- Debugging with multi-ICE

*Second day*

### ARM AND THUMB INSTRUCTION SETS

- Conditional execution and flags
- Branch instructions
- Single register data transfer
- Block data transfer
- Stack management
- Register access in Thumb
- ARM architecture V5TE new instructions

### ARM / THUMB INTERWORKING

- Branch exchange example
- Mixing ARM and Thumb subroutines
- ARM to thumb veneer
- Thumb-to-ARM veneer
- Interworking calls

### EXCEPTION HANDLING

- Exception priority
- Vector table instructions
- Chaining exception handlers
- Register usage in exception handlers
- Example C interrupt handler
- Software managed interrupt controller
- Issues when reenabling interrupts
- Invoking SWIs
- Data abort with memory management

*Third day*

### COMPILER HINTS AND TIPS

- Automatic optimization
- Instruction scheduling
- Tail-call optimization
- Loop termination
- Inline assembler
- Stack usage
- Global data layout

### INITIALIZING CACHED PROCESSORS

- Cache basics

- Cache flushing
- Write buffer
- Memory management
- TLB and translation tables
- Memory protection, MPU configuration steps
- System control coprocessor
- Tightly coupled memory

## EMBEDDED SOFWARE DEVELOPMENT

- ROM/RAM remapping
- Exception vector table
- Reset handler
- C library initialization
- Scatterloading
- Linker placement rules
- Long branch veneers
- C library functionality
- Placing the stack and heap

*Fourth day*

## AMBA3.0 INTERCONNECT SPECIFICATION

- Purpose of this specification
- Example of SoC based on AMBA specification
- Differences between AMBA2.0 and AMBA3.0

## AHB - ADVANCED HIGH PERFORMANCE BUS

- Centralized address decoding
- Address gating logic
- Arbitration, bus parking
- Single-data transactions
- Sequential transfers
- Retry response
- Split response
- AHB-lite specification

## APB - ADVANCED PERIPHERAL BUS

- Read timing diagram
- Write timing diagram
- Operation of the AHB-to-APB bridge
- APB3.0 new features

## ARM DEBUG SOLUTIONS

- Debugging with multiICE
- Watchpoints, hardware breakpoints, software breakpoints
- Debug communication channel
- Semihosting
- EmbeddedICE-RTT logic
- Real Time trace
- Instruction tracen, data trace
- Trace capture

## DOCUMENTATION

Training manuals will be given to attendees during training in print.