

# ARM Cortex A57 - Conception système

4 jours - 28 heures

## OBJECTIFS

- Cette formation traite en détails les particularités des cœurs ARM, aussi bien logiciel que matériel dans le but de faciliter la mise en oeuvre de cœurs Cortex-A57.
- Elle est destinée aux :
  - Ingénieurs logiciel qui veulent non seulement obtenir des détails sur la façon d'écrire un logiciel pour processeur ARM Cortex-A57, mais qui souhaitent également comprendre l'implémentation matérielle des cœurs au sein d'un microcontrôleur
  - Ingénieurs matériel qui ont besoin de comprendre comment concevoir des systèmes basés sur ARM Cortex-A57 mais également être capable de comprendre les bases de la programmation logicielle sur ces plates-formes

## FORMATIONS CONNEXES

- ARM Cortex A17/A15/A7 - Conception système
- ARM Cortex A53 - Conception système
- Langage C pour les applications temps-réel et embarquées

## PRÉREQUIS

- Une compréhension de base des microprocesseurs et microcontrôleurs est utile mais non indispensable
- Une compréhension de base de la logique numérique est utile mais non indispensable
- Une compréhension de base de la programmation en assembleur ou en langage C est utile mais non indispensable
- Des notions sur les cœurs ARM sont utiles mais non indispensables

## PARTENAIRES



## CHAPITRES

### CORTEX-A57 PROCESSOR OVERVIEW

- Hardware configuration options
- Software support

### INTRODUCTION TO ARMV8 ARCHITECTURE

- Architecture versions
- Privilege levels
- AArch64 registers
- A64 instruction set
- AArch64 exception model
- AArch64 memory model

### AARCH64 A64 ISA OVERVIEW

- Register set
- Load/Store instructions
- Data processing instructions
- Program flow instructions
- System control
- Advanced SIMD
- Cryptographic extensions

### AARCH64 EXCEPTION MODEL

- The AArch64 exception model
- Interrupts
- Synchronous exceptions
- SError exceptions
- SError exceptions in EL2 and EL3

### ARMV8-A MEMORY MANAGEMENT

- Memory management theory
- Stage 1 translations at EL 1/0
  - Kernel/application space translation tables
- translations at EL2/EL3
  - Stage 1 tables for hypervisor/secure exception levels
  - Stage 2 tables for virtualized systems

- Stage 2 tables for virtualized systems

- TLB maintenance

### ARMV8-A MEMORY MODEL

- Types
- Attributes
- Alignment & endianness

### CACHES AND BRANCH PREDICTION

- General cache information
- Cache attributes
- Cache maintenance operations
- Cache discovery

### BARRIERS

- Data barriers
- Instruction barriers

### SYNCHRONISATION

- Synchronisation in ARMv8-A
- Local and Global exclusive monitors

### CACHE COHERENCY

- Introduction to coherency
- Coherency details - multi-core processors
- Coherency details - multi-processor systems

### OS SUPPORT

- Context switching
- Modifying translation tables
- Privilege escalation protections
- Timers

### SOFTWARE ENGINEERING GUIDE TO THE CORTEX-A57

- Cache
- Interrupt configuration

### BOOTING

- Booting a Cortex-A5x processor in AArch64
- Processor setup

#### POWER MANAGEMENT FOR CORTEX-A

- Power overview
- Processor Power Modes
- Multiprocessor and System Power Mode
- Cortex-A5 and Cortex-A9 Power Mode

#### VIRTUALIZATION

- What is virtualization ?
- ARM virtualization support
  - Memory management
  - Exception handling
- Introduction to SMMU

#### SECURITY

- Software stack

- Memory system
- Debug
- TBSA (Trusted Base System Architecture)

#### GIC PROGRAMMING

- Distributor and CPU interfaces
- How to enable and configure interrupts
- How to handle interrupts
- How to send software interrupts
- Security extensions

#### DEBUG

- Debug infrastructure
- Invasive debug
- Non-invasive debug

## CONTACT

Tel : 05 62 13 52 32

Fax : 05 61 06 72 60

training@mvd-training.com