ARMv8-A 64-bit Architecture

Training

Let MindShare Bring “ARMv8-A 64-bit Architecture” to Life for You
This course covers the 64-bit ARMv8-A architecture that follows on from and offers compatibility with the earlier ARMv7-A 32-bit architecture. Examples of processors that first implemented this 64-bit architecture are the ARM Cortex-A53 and Cortex A57 processors. Since the successful introduction of the ARMv8-A architecture and products, there have been several architectural enhancements. This course includes enhancements 8.1, 8.2, 8.3, and 8.4. This course covers ISA – Instruction Set Architecture, rather than the details of individual implementations.

You Will Learn:

- ARM architecture (ARMv8.x-A)
- Support for execution of 32-bit ARMv7-A code
- 64-bit ISA (registers, instruction set, system instructions, etc)
- Floating point and Neon
- Calling conventions
- Memory model and paging
- Exception and Interrupt handling, and the exception levels
- Virtualization (Exception Level 2)
- TrustZone overview (Secure monitor at Exception Level 3 plus secure world)
- Power management
- Debug

Note, v8A includes compatibility support for v7-A code execution. This support is covered, but not details of the v7-A architecture.

Course Length: 4-Days

Target Audience:
This course is aimed at software developers and system architects developing for systems powered by ARMv8-A processors such as Cortex-A53 and Cortex-A57 Processors. It is relevant for operating systems development, device drivers, low-level coding and for application software.

Course Outline:

- Introduction to ARM 64-bit Architecture
- ARM architecture profiles, what is v8-A
- v8-A introduction and rational
- Introduction of the newer point releases
- Support for v7 legacy code
  - AArch32 and AArch64 state
  - v7 instruction set changes
    - Deprecation
    - Additional features (some new 64-bit features have also been added as new features available to 32-bit execution)
- 64-bit platform architecture overview
  - Sample SoC
  - MP Core
  - Interconnect (ACE or CHI)
    - Coherency and the interconnect
  - Distributed interrupt controller
  - Role of firmware
- Booting
  - A64 ISA
  - Integer registers
  - Instruction set
    - Integer operations
    - Memory operations
    - Stack
    - System instructions
      - System control registers
      - Relationship to v7 support and co-processors
    - Calling conventions
    - Memory access (DRAM and device)
      - Ordering model
      - Barriers
        - dmb, dsb, isb
        - load-acquire and store-release
        - Domains
      - Semaphores
      - Cache management
    - Floating point, advanced SIMD, crypto
      - Registers and instructions
    - Exception levels
      - The 4 exception levels
      - Stack model, handler and thread
      - Vector table
      - Core implementation choices
      - Switching AArch32 and AArch64 state
    - Exception and interrupt handling
      - Control of delivery of exceptions and interrupts
      - Syndrome registers
      - Switching exception levels
      - Return from exception
    - Paging
      - Page tables
      - 4K, 16K and 64K granules
      - Page sizes
      - Features achieved with page tables, such as execute never
      - Address space trickery – fields in pointers that are not part of the address, such as tags and pointer authentication
      - TLB management
- Virtualization Overview
  - Processor virtualization features
  - Use of exception levels
  - Nested virtualization
  - Interrupt virtualization features
  - Memory management
    - Second level page tables
    - Memory partitioning
    - I/O MMU (SMMU)
- Caches
  - Hardware cache coherency
  - Software responsibilities
  - Cache control in software
- Security (TrustZone)
  - TrustZone functionality
  - Links to TrustZone in other architectures
  - 32-bit or 64-bit TrustZone
Implications on exception levels, and the addition of secure EL2
Switching bitness of TrustZone

Other topics
Core power management, external power controller
  Power modes (dormant, shutdown)
  WFI, WFE, SEV
Debug
RAS – Reliability, Availability, Serviceability

Recommended Prerequisites:
Knowledge of ARM 32-bit v7 Architecture

Course Materials:
Students will be provided with an electronic version of the slides used in class.