**ARM CORTEX-A9MP SYSTEM DESIGN**

**Course Description**
This course takes an in depth look at the considerations you will need to take into account when designing a system containing a Cortex-A9 core

**It is aimed at:**
Software engineers who not only want to obtain details of how to write software to run on the Cortex-A9, but also wish to obtain an understanding of hardware design issues

Hardware engineers who need to understand how to design Cortex-A9 based systems, but also wish to obtain an understanding of the issues of writing software to run on that system.

**Level:** Intermediate

**Training Duration:** 4 days

**Prerequisites:**
- Knowledge of ARM7/9.
- This course does not include chapters on low level programming.

**Course objectives:**
- This course is split into 3 important parts:
  - Cortex-A9(MP) architecture
  - Cortex-A9(MP) software implementation and debug
  - Cortex-A9(MP) hardware implementation
- MMU operation under Linux is described.
- Spin-lock implementation in a multicore system is also detailed
- Interaction between level 1 caches, level 2 cache and main memory is studied through sequences.
- The exception mechanism is explained, indicating how virtualization enables the support of several operating systems.
- An overview of the Coresight specification is provided prior to describing the debug related units.
- The operation of the Snoop Control Unit when supporting SMP is fully explained, particularly the utilization of cache tag mirrors, the advantage of connecting DMA channels to ACP and the sequences that have to be used to modify a page descriptor.

**Practical Labs**
- For on-site courses, labs can be run under the following environments : Eclipse/RVDS, GNU/Lauterbach simulator
- For open courses, labs are run under Eclipse/RVDS

**First Day**

1. **INTRODUCTION TO CORTEX-A9 [1-hour]**
   - Block diagram, 1 or 2 AXI master interfaces
   - Cortex-A9 variants: single core vs multicore
   - New memory-mapped registers in MPCore
   - The 3 instruction sets
   - Configurable options: cache size, Jazelle, NEON, FPU, PTM and IEM

2. **ARM BASICS [1-hour]**
   - States and modes
   - Benefit of register banking
   - Exception mechanism
   - Instruction sets
   - Purpose of CP15

3. **INSTRUCTION PIPELINE [2-hour]**
   - Superscalar pipeline operation, out-of-order operation
   - Instruction cycle timing
   - Branch prediction mechanism, BTAC and GHB usage
   - Guidelines for optimal performance
   - Return stack
   - Predicted and non-predicted instructions
   - Prefetch queue flush
   - PMU related events

4. **TRUSTZONE [2-hour]**
   - TrustZone
   - Secure to non secure permitted transitions
   - Related CP15 registers
   - L1 and L2 secure state indicators, memory partitioning
   - Interrupt management when there is a mix of secure and non-secure interrupt sources
   - Boot sequence
   - Entering / exiting dormant mode

5. **INTRODUCTION TO MULTI-CORE SYSTEMS [2-hour]**
   - AMP vs SMP
   - Inter-Processor Interrupts
   - Boot sequence
   - Cluster ID
   - Exclusive access monitor, implementing Boolean semaphores
   - Global monitor
   - Spin-lock implementation
   - Using events
   - Indicating the effect of Multi Core on debug interfaces
   - Basic concepts of RTOS supporting A9 SMP architecture

*Continued...*
CORTEX-A9MP SYSTEM DESIGN

... Continued

Second Day

6. THUMB-2 INSTRUCTION SET (V7-A) [2-hour]
   • Introduction
   • General points on syntax
   • Data processing instructions
   • Branch and control flow instructions
   • Memory access instructions
   • Exception generating instructions
   • If...then conditional blocks
   • Stack in operation
   • Exclusive load and store instructions
   • Accessing special registers
   • Coprocessor instructions
   • Memory barriers and synchronization
   • Interworking ARM and Thumb states
   • Demonstration of assembly sequences aimed to understand this new instruction set

7. MEMORY MANAGEMENT UNIT [3-hour]
   • MMU objectives
   • Page sizes
   • Address translation
   • Page access permission, domain and page protection
   • Page attributes, memory types
   • Utilization of memory barrier instructions
   • Format of the external page descriptor table
   • Tablewalk
   • TLB organization
   • TLB lockout
   • Utilization of microTLBs
   • Abort exception, on-demand page mechanism
   • MMU maintenance operations
   • Using a common page descriptor table in an SMP platform, maintaining coherency of multiple TLBs
   • PMU related events
   • Related CP15 registers

8. LEVEL 1 MEMORY SYSTEM [2-hour]
   • Cache organization
   • Virtual indexing, physical tagging for instruction cache; physical indexing and tagging for data cache
   • Supported maintenance operations
   • Write-back write allocate cache allocation
   • Memory hint instructions PLD, PLI, PLDW, data prefetching
   • Describing transient cache related transactions: line fills and line eviction
   • No lockdown support
   • 4-entry 64-bit merging store buffer
   • PMU related events

9. HARDWARE COHERENCY [1-hour]
   • Snooping basics: CLEAN, CLEAN & INVALIDATE and INVALIDATE snoop requests
   • Snoop Control Unit: cache-to-cache transfers
   • MOESI state machine
   • Address filtering
   • Understanding through sequences how data coherency is maintained between L2 memory and L1 caches
   • Accelerator Coherency Port: connecting a DMA channel that uses this port to enforce coherency of data it is transmitting
   • Enabling coherency mode

Third Day

10. AMBA 3 [2-hour]
   • AXI
     - Topology: direct connection, multi-master, multi-layer
     - PL301 AXI interconnect
     - Separate address/control and data phases
     - AXI channels, channel handshake
     - Support for unaligned data transfers
     - Transaction ordering, out of order transaction completion
     - Read and write burst timing diagrams
     - Cortex-A9 external memory interface, ID encoding
   • APB 3

11. HARDWARE IMPLEMENTATION [2-hour]
   • Clock domains
   • Reset domains, power-on reset, debug and Data Engine logic reset
   • Power control, dynamic power management
   • Wait For Interrupt architecture
   • AXI master interface attributes
   • Level 2 memory interface: AXI read & write issuing capability
   • Exclusive L2 cache
   • AXI sideband information

Continued...
12. PL310 LEVEL 2 CACHE [2-hour]
   - Cache configurability
   - AXI interface characteristics
   - Exclusive mode operation when connected to Cortex-A9
   - Understanding through sequences how cacheable information is copied from memory to level 1 and level 2 caches
   - Transient operations, utilization of line buffers LFBs, LRBs, EBs and STBs
   - Discarding a level 3 memory line load through merging writes into STBs
   - TrustZone support
   - Power management
   - Cache event monitoring
   - Memory mapped registers included in the cache controller
   - Describing each maintenance operation
   - Cache lock down, implementation of a small memory by a boot program
   - Initialization sequence
   - Interrupt management

13. PERFORMANCE MONITOR [1-hour]
   - Event counting
   - Selecting the event to be counted for the 6 counters
   - Related interrupts
   - Debugging a multi-core system with the assistance of the PMU

Fourth day

14. INTERRUPT CONTROLLER [3-hour]
   - Cortex-A9 exception management: enforcing a particular endian mode on exception entry, configuring FIQ to be non maskable, configuring the default exception handling state: ARM vs Thumb
   - Interrupt virtualization
   - Integrated timer and watchdog unit in MPCore
   - Interrupt groups: STI, PPI, SPI, LSPI
   - Legacy mode: direct IRQ and FIQ
   - Assigning a security level to each interrupt source (Secure or Non Secure)
   - Prioritization of the interrupt sources
   - Distribution of the interrupts to the Cortex-A9 cores
   - Generation of interrupts by software
   - Detailing the interrupt sequence, purpose of Interrupt Acknowledge register and End-Of-Interrupt register
   - Clarifying which registers have a single instance and which registers are replicated to support MC
   - Spurious interrupt

15. LOW POWER MODES [1-hour]
   - Voltage domains
   - Cortex-A9 power control
   - Communication to the power management controller
   - Fast loop mode
   - Standby and wait for event signals, implementation in a multi-core system
   - SCU power status register

16. CORESIGHT DEBUG UNITS [3-hour]
   - Benefits of CoreSight
   - Invasive debug, non-invasive debug, taking into account the secure attribute
   - APBv3 debug interface
   - Connection to the Debug Access Port
   - Debug facilities offered by Cortex-A9
   - Process related breakpoint and watchpoint
   - Program counter sampling
   - Event catching
   - Debug Communication Channel
   - PTM interface, connection to funnel
   - Debugging while the processor is in shutdown or dormant mode
   - Debug registers description
   - Miscellaneous debug signals
   - Cross-Trigger Interface, debugging a multi-core SoC

17. Summary