RR1 - Cortex-R5 implementation

This course covers the Cortex-R5 / Cortex-R5F ARM cores

Objectives

- This course is split into 3 important parts:
  - Cortex-R5 architecture
  - Cortex-R5 software implementation and debug
  - Cortex-R5 hardware implementation.
- Interaction between level 1 caches, TCM and main memory is studied through sequences.
- The course explains how to assign access permissions and attributes to regions by using the MPU.
- The exception mechanism is detailed, indicating how the VIC port can contribute to reduce interrupt latency.
- Sequences involving memory, cache and external maste are used to explain the benefits of the ACP port.
- The course also details the hardware implementation and provides some guidelines to design a SoC based on Cortex-R5.
- An overview of the Coresight specification is provided prior to describing the debug related units.

Labs are run under RVDS

A more detailed course description is available on request at formation@ac6-formation.com

Prerequisites and related courses

- Basic knowledge of the ARM architecture.
- Assembly-level programming notions

Course Environment

- Theoretical course
  - PDF course material (in English) supplemented by a printed version for face-to-face courses.
  - Online courses are dispensed using the Teams video-conferencing system.
  - The trainer answers trainees' questions during the training and provide technical and pedagogical assistance.
- At the start of each session the trainer will interact with the trainees to ensure the course fits their expectations and correct if needed

Target Audience

- Any embedded systems engineer or technician with the above prerequisites.

Evaluation modalities

- The prerequisites indicated above are assessed before the training by the technical supervision of the traineein his company, or by the trainee himself in the exceptional case of an individual trainee.
- Trainee progress is assessed by quizzes offered at the end of various sections to verify that the trainees have assimilated the points presented
- At the end of the training, each trainee receives a certificate attesting that they have successfully completed the course.
  - In the event of a problem, discovered during the course, due to a lack of prerequisites by the trainee a different or additional training is offered to them, generally to reinforce their prerequisites,in agreement with their company manager if applicable.

Plan

First day

ARM BASICS

- States and modes
- Benefit of register banking
- Exception mechanism
- Instruction sets

INTRODUCTION TO CORTEX-R5

- Slave and master AXI ports
- Highlighting the new features with regard to Cortex-R4
- ARMv7-R architecture
- Exceptions
- System control coprocessor
- Configurable options
- Redundant CPU vs Twin-CPU

INSTRUCTION PIPELINE

- Prefetch unit
- Instruction cycle timing and interlock behavior
- Dynamic branch prediction mechanism: global history buffer
- Data Processing Unit
- Limited dual-issuing
- Global History Buffer
- Return stack

EXCLUSIVE RESOURCE MANAGEMENT

- Explaining issues when several processors share an exclusive resource
- Software aspects, load / store exclusive instructions
- Integrated local monitor
- Hardware aspects
- Using events to avoid to consume power while waiting for resource release

MEMORY TYPES

- Device and normal memory ordering
- Memory type access restrictions
- Access order
- Memory barriers

MEMORY PROTECTION UNIT

- ARM v7 PMSA
- Default memory map
- Cortex-R5 MPU and bus faults
- Region overview
- Setting up the MPU

EXCEPTION MANAGEMENT

- Low Interrupt Latency
- Primecell VIC PL192
- VIC basic signal timing
- Connectivity: daisy-chained VIC
- Interrupt priority and masking
- Determining the cause of the fault through CP15 status registers
- Precise vs imprecise faults

Second day

LEVEL 1 MEMORY SYSTEM

- Cache basics
- Tag RAM and Data RAM organization
- Handling cache parity / ECC errors
- Cache maintenance operations
- Tightly Coupled Memories
- ECC/parity protection
- Preloading TCMs with ECC
- Using TCMs from reset
- Store buffer, merging data

CACHE COHERENCY

- Hardware coherency vs software coherency
- ACP pass through interface,
- Virtual AXI peripheral interface region
- DMA into TCM
- Highlighting the difference between the µSCU and the Cortex-A SCU

AXI PROTOCOL

- PL301 AXI interconnect
- AXI channels, channel handshake
- Transaction ordering
- Read and write burst timing diagrams
- AXI master interface attributes
- Write merging example
- AXI slave interfaces attributes
- Peripheral interfaces port attributes
- Accelerator Coherency Port interface
- Controlling an external cache

HARDWARE IMPLEMENTATION

- Clock domains
- Reset domains
- Power control
- Maintaining caches and TCM powered while turning off the pipeline: dormant mode
- Power mode interaction with ACP
- Debugging the processor while powered down

Third day

APB - ADVANCED PERIPHERAL BUS

- Second-level address decoding
- Pinout

- APB3.0 new features

PERFORMANCE MONITOR

- Event counting
- Related interrupts
- Debugging a multi-core system with the assistance of the PMU

LOW POWER MODES

- Voltage domains
- Run mode, standby mode, dormant mode
- Communication to the power management controller

CORESIGHT DEBUG UNITS

- Benefits of CoreSight
- Invasive debug, non-invasive debug
- APBv3 debug interface
- Connection to the Debug Access Port
- Process related breakpoint and watchpoint
- Debug Communication Channel
- ETM interface
- Cross-Trigger Interface
- Debugging systems with energy management capabilities

THUMB-2 INSTRUCTION SET (V7-A)

- Introduction
- General points on syntax
- Data processing instructions
- Branch and control flow instructions
- Memory access instructions
- If&then conditional blocks
- Stack in operation
- Exclusive load and store instructions
- Memory barriers and synchronization
- Interworking ARM and Thumb states

Renseignements pratiques

Duration : 3 days
Cost : 2290 € HT