RM3 - Cortex-M4 / Cortex-M4F implementation

This course covers both Cortex-M4 and Cortex-M4F (with FPU) ARM core

Objectives

- This course is split into 3 important parts:
  - Cortex-M4 architecture
  - Cortex-M4 software implementation and debug
  - Cortex-M4 hardware implementation.
- Although the Cortex-M4 seems to be a simple 32-bit core, it supports sophisticated mechanisms, such as exception pre-emption, internal bus matrix and debug units.
- Through a tutorial, the Cortex-M4 low level programming is explained, particularly the ARM linker parameterizing and some tricky assembly instructions.
- The course also indicates how to use new DSP and FPU instructions to boost DSP algorithm implementation.
- Note that attendees can replay these labs after the training.
- The course also details the hardware implementation and provides some guidelines to design a SoC based on Cortex-M4, taking benefit of concurrent AHB transactions.
- An overview of the Coresight specification is provided prior to describing the debug related units.

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

Prerequisites

- A basic understanding of microprocessors and microcontrollers.

Environnement du cours

- Cours théorique
  - Support de cours au format PDF (en anglais) et une version imprimée lors des sessions en présentiel
  - Cours dispensé via le système de visioconférence Teams (si à distance)
  - Le formateur répond aux questions des stagiaires en direct pendant la formation et fournit une assistance technique et pédagogique
- Au début de chaque demi-journée une période est réservée à une interaction avec les stagiaires pour s'assurer que le cours répond à leurs attentes et l'adapter si nécessaire

Audience visée

- Tout ingénieur ou technicien en systèmes embarqués possédant les prérequis ci-dessus.

Plan du cours

FIRST DAY - ARCHITECTURE

INTRODUCTION TO ARM CORTEX-M4

- ARM Cortex-M4 processor macrocell
- Programmer s model
- Instruction pipeline
- Fixed memory map
- Privilege, modes and stacks
- Memory Protection Unit

- Interrupt handling
- Nested Vectored Interrupt Controller [NVIC]
- Power management
- Debug

ARM CORTEX-M4 CORE

- Special purpose registers
- Datapath and pipeline
- Write buffer
- Bit-banding
- System timer
- State, privilege and stacks
- System control block

ARCHITECTURE OF A SOC BASED ON CORTEX-M4

- Internal bus matrix
- External bus matrix to support DMA masters
- Connecting peripherals
- Sharing resources between Cortex-M4 and other CPUs
- Connection to Power Manager Controller

SECOND DAY - PROGRAMMING

EMBEDDED SOFTWARE DEVELOPMENT WITH CORTEX-M4

- Application startup
- Placing code, data, stack and heap in the memory map, scatterloading
- Reset and initialisation
- Placing a minimal vector table
- Further memory map considerations, 8-byte stack alignment in handlers

THUMB-2 INSTRUCTION SET

- 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, implementing atomic sequences
- Memory barriers and synchronization

CORTEX-M4 DSP INSTRUCTION SET

- Multiply instructions
- Packing / unpacking instructions
- V6 ARM SIMD packed add / sub instructions
- SIMD combined add/sub instructions, implementing canonical complex operations
- Multiply and multiply accumulate instructions
- SIMD sum absolute difference instructions
- SIMD select instruction
- Saturation instructions

FLOATING POINT UNIT

- Introduction to IEEE754
- Floating point arithmetic
- Cortex-M4F single precision FPU
- Register bank
- Enabling the FPU
- FPU performance, fused MAC
- Improving the performance by selection flush-to-zero mode and default NaN mode
- Extension of AAPCS to include FP registers

C/C++ COMPILER HINTS AND TIPS FOR Cortex-M4

- Mixing C/C++ and assembly
- Coding with ARM compiler
- Measuring stack usage
- Unaligned accesses
- Local and global data issues, alignment of structures
- Further optimisations, linker feedback

THIRD DAY - EXCEPTIONS, DEBUG

INTERRUPTS

- Basic interrupt operation, micro-coded interrupt mechanism
- Interrupt entry / exit, timing diagrams
- Interrupt stack
- Tail chaining
- Interrupt response, pre-emption
- Interrupt prioritization
- Interrupt handlers

EXCEPTIONS

- Exception behavior, exception return
- Non-maskable exceptions
- Privilege, modes and stacks
- Fault escalation
- Priority boosting
- Vector table

MEMORY PROTECTION UNIT

- Memory types
- Access order
- Memory barriers, self-modifying code
- Memory protection overview, ARM v7 PMSA
- Cortex-M4 MPU and bus faults
- Fault status and address registers
- Region overview, memory type and access control, sub-regions
- Region overlapping

INVASIVE DEBUG

- Coresight debug infrastructure
- Halt mode

- Vector catching
- Debug event sources
- Flash patch and breakpoint features
- Data watchpoint and trace
- ARM debug interface specification
- Coresight components
- AHB-Access Port
- Possible DP implementations: Serial Wire JTAG Debug Port [SWJ-DP] or SW-DP

NON-INVASIVE DEBUG

- Basic ETM operation
- Instruction trace principles
- Instrumentation trace macrocell
- ITM stimulus port registers
- DWT trace packets
- Hardware event types
- Instruction tracing
- Synchronization packets
- Interface between on-chip trace data from ETM and Instrumentation Trace Macrocell [ITM]
- TPIU components
- Serial Wire connection

FOURTH DAY HARDWARE IMPLEMENTATION

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
- Indivisible transactions
- Single-data transactions
- Address pipelining
- Sequential transfers
- AHB-lite specification
- Parameterizing the AHB core provided by ARM

APB - ADVANCED PERIPHERAL BUS

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

AHB CORTEX-M4 HARDWARE IMPLEMENTATION

- Clocking and reset, power management
- Using an external Wake-up Interrupt Controller (WIC)
- Bus interfaces: Icode memory interface, Dcode memory interface, System interface and External Private Peripheral Bus interface

- AMBA-3 compliance
- Unifying the code buses
- Unaligned access management
- Debug interface
- Connection to the TPIU
- AHB Trace Macrocell (HTM)