This course covers the Cortex-M3 ARM core

Objectives

- This course is split into 3 important parts:
  - Cortex-M3 architecture
  - Cortex-M3 software implementation and debug
  - Cortex-M3 hardware implementation.
- Although the Cortex-M3 seems to 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-M3 low level programming is explained, particularly the ARM linker parameterizing and some tricky assembly instructions.
- 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-M3, 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.

Modalités d'évaluation

- Les prérequis indiqués ci-dessus sont évalués avant la formation par l'encadrement technique du stagiaire dans son entreprise, ou par le stagiaire lui-même dans le cas exceptionnel d'un stagiaire individuel.
- Les progrès des stagiaires sont évalués par des quizz proposés en fin des sections pour vérifier que les stagiaires ont assimilé les points présentés
- En fin de formation, une attestation et un certificat attestant que le stagiaire a suivi le cours avec succès.
  - En cas de problème dû à un manque de prérequis de la part du stagiaire, constaté lors de la formation, une formation différente ou complémentaire lui est proposée, en général pour conforter ses prérequis, en accord avec son responsable en entreprise le cas échéant.

Plan

ARM Cortex-M3 CORPORATE INTRODUCTION

- ARM architectural summary
- Meeting the challenge with profiles
- ARM instruction set evolution

ARM Cortex-M3 INTRODUCTION

- ARM Cortex-M3 processor macrocell
- Programmer s model
- Program status registers
- Instruction pipeline
- Fixed memory map
- Memory Protection Unit
- Interrupt handling
- Power management

ARM Cortex-M3 CORE

- Block diagram
- Datapath and pipeline
- Write buffer
- Bit-banding
- State, privilege and stacks
- Alignment and endianness
- System control block

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
- Accessing special registers
- Tutorial: Becoming familiar with Keil IDE
  - How to design a new project
  - Parameterizing the IDE
  - Executing simple labs to understand the operation of assembly complex instructions, such as table branch and it

INTERRUPTS

- Basic interrupt operation
- Interrupt entry / exit, timing diagrams
- Tail chaining
- Interrupt response, pre-emption
- NVIC registers
- Interrupt prioritization
- Interrupt implementation configurability, impact on core size

EXCEPTIONS

- Exception behavior, exception return
- Non-maskable exceptions

- Privilege, modes and stacks
- Priority boosting
- Vector table

MEMORY TYPES

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

MEMORY PROTECTION UNIT

- Memory protection overview
- Fault status and address registers
- Region overview, memory type and access control, sub-regions
- Setting up the MPU

EMBEDDED SOFTWARE DEVELOPMENT WITH Cortex-M3

- Embedded development process
- Application startup
- Placing code, data, stack and heap in the memory map, scatterloading
- Reset and initialisation
- Placing a minimal vector table
- Building and debugging your image
- Long branch veneers
- Tutorial: Becoming familiar with Keil IDE
  - Scatterloading
  - Retargeting the C library
  - Handling interrupts in C language
  - Using SVC

INVASIVE DEBUG

- Coresight debug infrastructure
- Halt mode
- Monitor mode
- Debug event sources
- Flash patch and breakpoint features
- FPB remapping
- Data watchpoint and trace
- DWT registers
- ARM debug interface specification
- AHB-Access Port
- Possible DP implementations

NON-INVASIVE DEBUG

- Basic ETM operation
- Instruction trace principles
- ITM packets
- DWT trace packets
- Time-stamping packets
- Instruction tracing
- TPIU components
- TPIU pinout
- Software interface

C/C++ COMPILER HINTS AND TIPS FOR Cortex-M3

- ARM compiler optimisations
- Mixing C/C++ and assembly
- Coding with ARM compiler
- Measuring stack usage
- Unaligned accesses
- Local and global data issues, alignment of structures
  - Tutorial: Implementing these optimizations by using ARM/Keil compiler

AMBA3.0 INTERCONNECT SPECIFICATION

- Purpose of this specification
- 2-bus organization
- Example of SoC based on AMBA specification

AHB - ADVANCED HIGH PERFORMANCE BUS

- Centralized address decoding
- Address gating logic
- Arbitration, bus parking
- Single-data transactions
- Address pipelining
- Sequential transfers
- AHB-lite specification

APB - ADVANCED PERIPHERAL BUS

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

AHB CORTEX-M3 HARDWARE IMPLEMENTATION

- Clocking and reset, power management
- Bus interfaces
- AMBA-3 compliance
- Unifying the code buses
- Branch Status signal
- Unaligned access management
- Connection to the TPIU

Renseignements pratiques

Renseignements : 4 jours