MSP - Microchip SmartFusion2 Programming

This course describe the Microcontroller Subsystem (MSS) of SmartFusion2 Microchip FPGAs

Objective

- Get an Overview on the Cortex-M Architecture
- Understand the Cortex-M Software implementation and debug
- The Microchip Implementation and Embedded Development Kit (EDK) with Libero SoC and software Integrated Development environment (IDE) tools are described to create a hardware platform and the software to execute to programme it
- Describe SmartFusion MSS Architecture, I/Os and understand the SmartFusion2 FPGA Fabric Interface
- Become familiar with the MSS Peripherals

Prerequisites

- Basic knowledge of processor and FPGA technology
- Knowledge of VHDL and C languages

Course environment

- Microchip LiberoSoC v12.0 and SoftConsole v6.0
- SmartFusion2 based board

Environnement du cours

- Cours théorique
  - Support de cours imprimé et au format PDF (en anglais).
  - Le formateur répond aux questions des stagiaires en direct pendant la formation et fournit une assistance technique et pédagogique.
- Activités pratiques
  - Les activités pratiques représentent de 40% à 50% de la durée du cours.
  - Elles permettent de valider ou compléter les connaissances acquises pendant le cours théorique.
  - Exemples de code, exercices et solutions
  - Un PC (Linux ou Windows) par binôme de stagiaires (si plus de 6 stagiaires) pour les activités pratiques avec, si approprié, une carte cible embarquée.
  - Le formateur accède aux PC des stagiaires pour l'assistance technique et pédagogique.
- Une machine virtuelle préconfigurée téléchargeable pour refaire les activités pratiques après le cours
- 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.

Course Outline

First Day

Libero SoC

- Microchip FPGA & SoC overview
- Libero SoC overview
- Create and Design
- Constraint management
- TestBench and Simulations
- Programme and Debug
- Microchip (Microsemi) tool’s

Exercise : Creating a Libero Project for Firmware
Exercise : Libero Project and SoftConsole

Cortex-M3 Architecture Overview

- V7-M Architecture Overview
- Core Architecture
  - Harvard Architecture, I-Code, D-Code and System Bus
  - Write Buffer
  - Registers (Two stack pointers)
  - States
  - Different Running-mode and Privileged Levels
  - System Control Block
  - Systick Timer
  - MPU Overview
- Programming
  - Alignment and Endianness
  - CMSIS Library
- Exception/ Interrupts Mechanism Overview
  - Vector Table
  - Interrupt entry and return Overview
  - Tail-Chaining
  - Pre-emption (Nesting)
  - NVIC Integrated Interrupt Controller
  - Exception Priority Management
  - Fault escalation
  - Debug Interface

Second Day

SmartFusion2 Cortex-M3 Processor

- System Level Interface
- Integrated Configurable Debug
- Cortex-M3 Processor Core Peripherals
  - Nested Vectored Interrupt Controller
  - System Control Block
  - System Timer
  - Memory Protection Unit
- Cortex-M3 Processor Description

    - Programmers Model
    - Memory Model
    - Exception Model
    - Fault Handling
    - Power Management
- Cortex-M3 Processor Instruction Set
    - CMSIS Functions
    - Memory Access Instructions
    - General Data Processing Instructions
    - Saturating Instructions
    - Branch and Control Instructions
- Cortex-M3 Processor Peripherals
    - System Control Block
    - System Timer (Systick)
    - Memory Protection Unit

Exercise : Cortex-M3 Mode Privilege
Exercise : Cortex-M3 Exception Management
Exercise : Cortex-M3 MPU
Exercise : Cortex-M3 Real-Time Operating System (FreeRTOS)

MSS Cache Controller

- Cache Matrix
- Memory Mapping
- Memory Maps and Transaction Mapping
- Cache Locked Mode
- How to use cache Controller

Exercise : Cache Controller Configuration

Embedded NVM (eNVM) and SRAM (eSRAM) Controllers

- Functional Description
- Security
- How to use eNVM
- SYSREG Control Registers

High Performance DMA Peripheral and Controller

- DMA Controller Initialization
- DMA Controller Operations
- How to use HPDMA
- HDMA Controller Register Map
- Peripheral DMA Architecture Overview
- How to use PDMA
- PDMA Register Map

Fabric Interface Controller

- Architecture Overview
- Functional Description
- AHB-Lite Options
- Implementation Considerations
- How to Use FIC

Third Day

Reset Controller

- Power-On Reset Generation Sequence
- Power-Up to functional Time Data
- CoreResetP Soft Reset Controller
  - Reset Topology
  - Implementation
- How to use the Reset Controller

AHB - Advanced High Performance Bus

- Centralized address decoding
- Address gating logic
- Address pipelining
- Sequential transfers
- AHB-Lite Specification

Connectivity and Communication

- Universal Serial Bus OTG Controller
- Ethernet MAC
- CAN Controller
- MMUART Peripherals
- Serial Peripheral Interface Controller (SPI)
- Inter-Integrated Circuit Peripherals (I^2C)
- MSS GPIO
- Communication Block
  - Architecture Overview
  - CoreSysServices Soft IP
  - How to use the communication block
- Real-Time Controller
- System Timer
- Watchdog Timer
- ECC System Service

Exercise : Programming using UART interface
Exercise : MSS CAN drivers and APIs
Exercise : Programming using USB OTG Controller Interface
Exercise : Using ECC System Service
Exercise : FreeRTOS and LWIP Project