TI1 - TI STELLARIS CORTEX-M3 BASED MCUs IMPLEMENTATION

This course covers all MCUs belonging to the Stellaris Cortex-M3 family, X00, 1000, 2000, 3000, 5000, 8000 & 9000 SERIES

Objectives

- This course has 5 main objectives:
  - Describing the hardware implementation and highlighting the pitfalls
  - Describing the ARM Cortex-M3 core architecture
  - Becoming familiar with the CCS or Keil IDE and low level programming.
  - Describing the units which are interconnected to other modules, such as clocking, interrupt controller and DMA controller, because the boot program generally has to modify the setting of these units
  - Describing independent I/O modules and their drivers developed by TI (DriverLib).
- Note that this course has been designed from the architecture of the most complex STELLARIS device, the LM3S9B96
  - Consequently, a chapter has been designed by Acsys for each possible integrated IP
  - According to the actual reference chosen by the customer, some chapters may be removed.
- Products and services offered by ACSYS:
- ACSYS is able to assist the customer by providing consultancies
  - Typical expertises are done during board bringup, hardware schematics review, software debugging, performance tuning.
  - ACSYS has also an expertise in FreeRTOS porting and uIP /LWIP or Interniche stack integration.

This document is necessary to tailor the course to specific customer needs and to define the exact schedule.

Prerequisites and related courses

- This course provides an overview of the ARM Cortex-M3 core. Our course reference RM2 - Cortex-M3 implementation course details the operation of this core.
- The following courses could be of interest:
  - USB Full Speed High Speed and USB On-The-Go, reference IP2 - USB 2.0course
  - Ethernet and switching, reference N1 - Ethernet and switchingcourse
  - IEEE1588, reference N2 - IEEE1588 - Precise Time Protocolcourse
  - CAN bus, reference IA1 - CAN buscourse

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.

Course Outline

ARCHITECTURE OF STELLARIS MCUs

- ARM core based architecture
- Description of Series 1000, 2000, 3000, 5000, 8000 and 9000 SoC architecture
- Clarifying the internal data and instruction paths
- Highlighting possible concurrent transactions
- Integrated memories
- SoC mapping

THE ARM CORTEX-M3 CORE

- V7-M core family
- Core architecture
- Programming
- Exception behavior, exception return
- Basic interrupt operation, micro-coded interrupt mechanism
- Memory Protection Unit

BECOMING FAMILIAR WITH CODE COMPOSER STUDIO

- Getting started with the IDE
- Parameterizing the compiler / linker
- Creating a project from scratch
- C start program

PROGRAMMING AND DEBUGGING

- IEEE 1149.1-1990 compatible Test Access Port (TAP) controller
- Integrated ARM Serial Wire Debug

RESET, POWER AND CLOCKING

- Reset
- Clocking
- Power control

INTERNAL INTERCONNECT

- Bus matrix
- µDMA

HARDWARE IMPLEMENTATION

- Power pins
- Pinout
- GPIO module

INTEGRATED MEMORIES

- Flash memory, this module is not implemented in all STELLARIS devices
- Internal SRAM
- Internal ROM

EXTERNAL PERIPHERAL INTERFACE

- Host bus
- General purpose interface
- DRAM controller

TIMERS

- General Purpose Timer Module block
- Capture Compare PWM pins
- Watchdog timers
- Advanced Motion Control

ANALOG MODULES

- 10-bit Analog-to-Digital Converter and Programmable Gain Amplifier
- Analog comparators

CONNECTIVITY AND COMMUNICATION

- SSI
- UART
- I2C
- CAN modules
- USB
- Fast ethernet with IEEE1588
- ISO7816 smartcard interface
- I2S audio interface
- Using the drivers developed by TI to implement these IO ports (I2C, SPI, UART, USB)