



STR8 - STM32MP15 Implementation

This course describes the STM32MP15x SoC

Objectifs

- This course details the hardware implementation of the STM32MP15x SoC
- The course focuses on the boot sequence, the clocking and the power management strategies
- The course explains all parameters that affect the performance of the system in order to easily perform the final tuning
- An overview of the Cortex-A7MP core helps to understand issues caused by MMU, cache and snooping
- An overview of the Cortex-M4F with MPU is included to understand the microcontroller side of the STM32MP15 implementation
- To become more familiar with the synchronization features of the STM32MP15x implementation labs are proposed
- Note that this course has been designed from the architecture of STM32MP15x-series devices, the STM32MP157C
- The peripherals overview presented in this course can be detailed upon request (cours [STR9 - STM32 Peripherals](#))

Prerequisites and Related Courses

- Familiarity with C concepts and programming targeting the embedded world
- Basic knowledge of embedded processors
- The following courses could be of interest:
 - cours [RA4 - Cortex-A7 implementation](#)
 - cours [RM3 - Cortex-M4 / Cortex-M4F implementation](#)
 - cours [STR9 - STM32 Peripherals](#)
 - cours [OS3 - FreeRTOS Programming](#)
 - cours [D1S - Embedded Linux with Ac6 System Workbench](#)
 - cours [D1Y - Linux embarqué avec Yocto](#)

Course environment

- Convenient course material
- Example code, labs and solutions

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
- 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
 - Pour les formations à distance:
 - ▶ Un PC Linux en ligne par stagiaire pour les activités pratiques, avec tous les logiciels nécessaires préinstallés.
 - ▶ Le formateur a accès aux PC en ligne des stagiaires pour l'assistance technique et pédagogique
 - ▶ Certains travaux pratiques peuvent être réalisés entre les sessions et sont vérifiés par le formateur lors de la session suivante.
 - Pour les formations en présentiel:
 - ▶ Un PC (Linux ou Windows) pour les activités pratiques avec, si approprié, une carte cible embarquée.
 - ▶ Un PC par binôme de stagiaires s'il y a plus de 6 stagiaires.
 - Pour les formations sur site:
 - ▶ Un manuel d'installation est fourni pour permettre de préinstaller les logiciels nécessaires.

- ▶ Le formateur vient avec les cartes cible nécessaires (et les remporte à la fin de la formation).
- Une machine virtuelle préconfigurée téléchargeable pour refaire les activités pratiques après le cours
- Au début de chaque session (demi-journée en présentiel) 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

Cortex M4 Architecture Overview

- V7-M Architecture Overview
- Core Architecture
 - Harvard Architecture, I-Code, D-Code and System Bus
 - Write Buffer
 - Registers (Two stacks pointers)
 - States
 - Different Running-modes and Privileged Levels
 - System Control Block
 - Systick Timer
 - MPU Overview
- Programming
 - Alignment and Endianness
 - CMSIS Library
- Exception/ Interrupt 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

Exercise : Becoming familiar with the IDE and Clarifying the boot sequence

Exercise : Cortex-M4 Mode Privilege

Exercise : Cortex-M4 Exception Management

Exercise : Cortex-M4 MPU

The ARM Cortex-A7MP Architecture Overview

- V7-A Architecture Overview
- Cortex A7 Overview
 - Cortex-A7 Architecture
 - Hardware Cache Coherency
 - Cortex-A7 Main Features
- System Features
 - Multi-processing
 - Cache Maintenance
 - Cache Coherency Hardware
 - Interrupt Distribution
 - Power saving Modes

- Memory System Hierarchy
- Software Storage and Upload
- Memory Management Unit
- Generic Interrupt Controller
- Multicore operation
- TrustZone
- Virtualization Extension

Second Day

STM32MP15 Architecture Overview

- ARM Core Based Architecture
- Description of STM32MP15 SoC architecture
- Clarifying the internal data and instruction paths:
 - Bus Architecture
 - NIC-400 Network Interconnect AXI-based
 - Multi-Layer AHB Interconnect
- Memory Organization
 - Embedded Memories: (ROM, SYSRAM, MCU SRAM, Retention RAM)
 - External Memories: (DDR3/LPDDR2, FMC, QUADSPI, SDMMCx)
- SoC mapping
- Boot Configuration

Boot, Security and One time programmable (OTP) control (BSEC)

- Introduction
- BSE Block diagram
- Interface to OTP
- OTP security Mode
- OTP operations
 - OTP read
 - OTP programming
 - OTP permanent write lock
- Scratch registers and Transport key (TK) access
- OTP Mapping

Reset, Power and Clocking

- Power Control
 - Power control overview
 - Power supplies
 - Power supply supervision
 - Power management
 - Low-Power Modes
 - Power control interrupts
 - Power Control and TrustZone capability
- RCC Reset and Clock
 - RCC overview
 - RCC Block diagram
 - RCC Reset
 - RCC Clock
 - RCC interrupts
 - Handling dynamic Clock switching
 - PLL Programming
 - Configuring the sub-system clock
 - Clock calibrations using timers

Exercise : Configure the system to measure the current consumption in different low-power modes (Cortex-M4)

Exercise : RTC wakeup timer event / interrupt (Cortex-M4)

Exercise : Configure the system Clock (SYSCLK) and modify the clock settings in Run Mode (Cortex-M4)

Synchronizations mechanisms

- Hardware Semaphore (HSEM)
 - HSEM Overview
 - Lock procedures
 - Clear Procedures
 - Interrupts
- Inter-Processor Communication Controller (IPCC)
 - IPCC Overview
 - Simplex Channel and Half-Duplex channel modes
 - IPCC interrupts

Third Day

Hardware Implementation

- Power pins
- Pinout
 - Pin Muxing, alternate functions
- GPIO Module
 - Configuring a GPIO
 - Speed selection
 - Locking mechanism
 - Analog function
 - Integrated pull-up / pull-down
 - I/O pin multiplexer and mapping
 - TrustZone security
- System Configuration Controller
 - I/O compensation cell
 - Ethernet Clock source
- Interrupts
 - Nested Vectored Interrupt Controllers (NVIC)
 - Global interrupt Controller (GIC)
 - Extended Interrupt and event controller (EXTI)

Exercise : How to configure the external interrupt lines

Internal Interconnect

- Bus Matrix
- DMA

Exercise : DMA FIFO mode

Security and Integrity

- Extended TrustZone Protection Controller
 - Extended Trustzone architecture and ETZPC
 - STM32P15x Security Architecture
 - STM32P15x MCU resource isolation
- True Random number generator (RNG)
- Hash processor (HASH)
- Cryptographic Processor (CRYP)

Exercise : CRC User Defined Polynomial

Exercise : How to use ASH peripheral to hash data with SHA-1 and MD5 algorithms

Exercise : How to use the Cryptographic Processor

Memory Features

- DDR3/LPDDR2/LPDDR3 Controller (DDRCTRL)
 - DDRCTRL Architecture overview
 - Transaction Service Control and QoS
 - Power saving
 - Address mapper
 - DRAM timing parameters
 - SDRAM initialization sequence
 - Refresh controls
 - DDRCTRL Configuration
 - TrustZone Address Space Controller for DDR
- DDR physical Interface Control (DDRPHYC)
- DDR performance monitor (DDRPERFM)
- Master Direct Memory Access (MDMA) Controller
- Flexible Memory Controller (FMC)
- Quad-SPI interface
- Delay Block

Exercise : Configuring the FSMC controller to access the SRAM memory

Exercise : QSPI Read Write IT

Fourth Day

Asymmetric Multiprocessing Communication

- Open AMP-Overview
- Components in OpenAMP
- Connection between OpenAMP and Libmetal
- How to write a simple OpenAMP application
- CoProSync APIs

Exercise : Creating rpmsg channel between Cortex-M4 and Cortex-A7MP (Cortex-M4 side)

Analog modules

- Analog-to-Digital (ADC)
- Digital-to-Analog Converter (DAC)
- Digital filter for sigma delta modulators (DFSDM)
- Temperature sensor (TDS)

Exercise : ADC Single Conversion Trigger Timer DMA

Fifth Day

Watchdogs and Real-time clock (RTC)

- Independent Watchdogs (IWDG)
- Window Watchdog
- RTC
 - Overview
 - Functional Description
 - RTC low power-modes

Exercise : MCU WWDG reset

Exercise : RTC Alarm

Connectivity and communication

- SPI

- USART/UART
- I2C
- Serial audio interface & SPDIF receiver interface
- Management data input/output (MDIOS)
- Secure Digital input/output MultiMediaCard Interface
- FDCAN
- USB OTG
- USB HS PHY controller
- USB EHCI/USB_OHCI
- HDMI-CEC
- Ethernet

Exercise : How to handle I2C data buffer Tx/Rx between two boards via DMA

Debug Support

- Debug block diagram
- Debug power and clocking
- Security
- Chip Level TAP controller
- Serial wire and JTAG debug port (SWJ-DP)
- Access port
- Cortex-A7 debug
- Cortex-M4 debug