STR11 - STM32H7

This course describe the STM32H7 architecture and practical examples

Objectives

- Understand STM32H7 (Cortex-M7 @ up to 480 MHz), AXI/TCM memory architecture, caches, and power domains.
- Configure clocks, Flash/Option Bytes (dual-bank, RDP/WRP/PCROP), and boot flow safely.
- Drive key peripherals (DMA/MDMA, timers, ADC, comms) with performance in mind.
- Apply low-power modes across D1/D2/D3 domains; measure impact.

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.
- Practical activities
  - Practical activities represent from 40% to 50% of course duration.
  - Code examples, exercises and solutions
  - For remote trainings:
    - One Online Linux PC per trainee for the practical activities.
    - The trainer has access to trainees' Online PCs for technical and pedagogical assistance.
    - QEMU Emulated board or physical board connected to the online PC (depending on the course).
    - Some Labs may be completed between sessions and are checked by the trainer on the next session.
  - For face-to-face trainings:
    - One PC (Linux ou Windows) for the practical activities with, if appropriate, a target board.
    - One PC for two trainees when there are more than 6 trainees.
  - For onsite trainings:
    - An installation and test manual is provided to allow preinstallation of the needed software.
    - The trainer come with target boards if needed during the practical activities (and bring them back at the end of the course).
- Downloadable preconfigured virtual machine for post-course practical activities
- 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

Day 1

Cortex-M7 & memory map

- Programmer’s model, exceptions/NVIC, FPU/DP.
- ITCM/DTCM vs AXI-SRAM: when to place code/data.
- I-Cache/D-Cache: coherency rules; DMA implications.
- MPU basics for safety.

Exercise: Enable caches
Exercise: Place a hot loop in ITCM
Exercise: MPU guard

AXI & DMA family (DMA1/2, BDMA, MDMA)

- AXI matrix overview; masters/targets.
- DMA vs BDMA vs MDMA roles; scatter-gather, linked lists.
- Throughput and arbitration basics.
- Cache maintenance around DMA (invalidate/clean).

Exercise: ADC→DMA stream with cache-safe buffers; MDMA move/format frames

RCC & clock tree

- HSE/HSI/PLL1..3; domain clocks (D1/D2/D3).
- Safe re-clocking; MCO for verification.
- Timer clocks vs core; prescaler pitfalls.
- Debug clock freeze effects.

GPIO & EXTI

- Speed, drive, AF mapping; EXTI lines.
- Debounce strategies; input filtering.
- Interrupt latency tips on M7.
- Simple board bring-up checklist.

Exercise: Button EXTI + LED

TIM (general-purpose/advanced)

- PWM modes, dead-time (brief), one-pulse.
- Input capture/measure; trigger chaining.
- LPTIM vs TIM for low-power.

Exercise: Timer example

Day 2

ADC

- Trigger sources; sampling time; oversampling.
- DMA to ring buffer; window stats in main.
- Internal channels (Vref, temperature).

Exercise: Timer-triggered ADC→DMA

Communications

- USART
  - Modes & framing: word length, parity, oversampling; baud tolerance.
  - DMA RX/TX (idle-line, half/full callbacks); ring buffers.
  - Flow control (RTS/CTS) and latency/throughput trade-offs.
- I²C
  - Master transfers; repeated-START; timing vs bus speed.
  - Clock stretching; timeouts; “bus busy”.
  - Bus recovery for stuck SDA/SCL
- SPI
  - CPOL/CPHA, word sizes, simplex/half/full-duplex.
  - HW NSS vs GPIO CS; inter-frame delays.
  - DMA streaming; FIFO usage; dummy bytes.

SDMMC + FatFS (optional)

- Card detect, init/clocking.
- Mount/format; file append patterns.
- Buffering/latency; wear; safe close on power loss.
- Simple log rotation.

Exercise: Log “timestamp, ADC” to CSV

PWR & low-power (D1/D2/D3)

- Run/Stop/Standby; what’s retained per domain.
- Wake sources (RTC/EXTI/LPTIM) across domains.
- Regulator choices (LDO/SMPS) basics; VCORE scaling.
- Measurement setup.

Exercise: Sleep vs Stop current table; Standby + RTC wake; log reset cause.

Day 3

Boot & Option Bytes (dual-bank aware)

- Boot sources (Flash, system memory, SRAM); vector relocation.
- Key OBs: dual-bank/boot swap (BFB2), WRP/PCROP, RDP levels.
- Safe read/modify/verify with CubeProgrammer.
- Bank-swap update concept (overview).

Exercise: Read OBs; toggle a user OB; verify after reset

Robustness: faults, MPU, watchdogs

- HardFault decoding; capture LR/PC/CFSR.
- MPU regioning: stack guards, no-exec, peripheral windows.
- IWDG vs WWDG; service windows and recovery.
- BOR levels; reset-cause logging at boot.

Tracing & logging

- ITM/SWO quick setup; timestamped printf.
- Event markers around DMA/ISR.
- Buffering vs blocking; minimal asserts.
- Measuring UART vs ITM overhead.

Exercise: ITM printf + markers; compare overhead to UART

External memory (option): FMC/OCTOSPI

- FMC vs OCTOSPI use-cases; mapping to AXI.
- Command, dummy cycles, and memory-mapped mode.
- Cache/line-fill effects on XIP; prefetch tips.
- Basic integrity/perf test templates.