STR12 - STM32H5 programming

This course descirbe the STM32H5 architecture

Objectives

- Grasp STM32H5 fundamentals: Cortex-M33, memory map, dual-bank Flash, clock tree.
- Build performant drivers with GPDMA (linked-list) and measure impact.
- Use comms (USART, I²C, SPI), ADC, storage (optional SDMMC/FatFS), and low-power effectively.
- Introduce security: TrustZone-M, TF-M, and STM32Trust Secure Manager.

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.

Evaluation modalities

- The prerequisites indicated above are assessed before the training by the technical supervision of the traineein his company, or by the trainee himself in the exceptional case of an individual trainee.
- Trainee progress is assessed in two different ways, depending on the course:
  - For courses lending themselves to practical exercises, the results of the exercises are checked by the trainer while, if necessary, helping trainees to carry them out by providing additional details.
  - Quizzes are offered at the end of sections that do not include practical exercises to verifythat the trainees have assimilated the points presented
- At the end of the training, each trainee receives a certificate attesting that they have successfully completed the course.
  - In the event of a problem, discovered during the course, due to a lack of prerequisites by the trainee a different or additional training is offered to them, generally to reinforce their prerequisites,in agreement with their company manager if applicable.

Plan

Day 1

Cortex-M33 & memory map

- Programmer's model
- Exceptions/NVIC.
- SRAM regions; dual-bank Flash overview.
- MPU basics (no-exec, guards).

Exercise: *Cortex-M33 Exceptions/Interrupts*

Exercise: *ARMv8 MPU protection*

GPDMA

- Roles vs CPU; request routing; triggers.
- Linked-list descriptors; circular vs normal.
- Cache-safe buffers; invalidate/clean patterns.

Exercise: *ADC &rarr; GPDMA ring buffer with LL*

RCC & clock tree

- HSE/HSI/PLL; safe re-clocking.
- Domain clocks and prescalers.
- MCO output to verify SYSCLK.

Exercise: *Configure PLL*

GPIO & EXTI

- AF mapping; speeds/drive; input filtering.
- EXTI lines & priorities.
- Debounce strategies.
- Board bring-up checklist.

Exercise: *Button EXTI + LED*

Exercise: *timestamp ISR latency*

Timers (General Purpose)

- PWM modes & ARR/CCR.
- Input capture (period/frequency).
- One-pulse & basic chaining.

Exercise: *PWM dimmer + frequency meter (IC)*

Day 2

ADC

- Triggers & sampling; oversampling.
- DMA to ring buffer; window stats.
- Internal channels (Vref/Temp).

Exercise: *Timer-triggered ADC&rarr;GPDMA*

Communications

- USART

  - Modes & framing
  - DMA & flow control
  - Errors & diagnostics
- I²C
  - Master transfers
  - Bus management
  - Robustness
- SPI
  - Modes & timing
  - DMA & CS
  - Integrity & perf

SDMMC + FatFS (optional)

- Card detect/init; mount/format.
- Append patterns; buffering/latency.
- Wear & safe close on power loss.

PWR & low-power

- Run/Sleep/Stop/Standby overview.
- Wake sources (RTC/EXTI/LPTIM).
- Regulator notes (SMPS/LDO) & VCORE scaling.
- Measuring current: setup & pitfalls.

***Exercise:** Sleep vs Stop current table; Standby + RTC wake; print reset cause*

Day 3

Boot flow & Option Bytes

- Boot sources; vector relocation.
- Key OBs; read/verify safely.
- Dual-bank concepts for updates.
- RDP/WRP/PCROP overview.

TrustZone-M partitioning (intro)

- Secure vs Non-Secure images; SAU/IDAU; GTZC basics.
- NSC veneers; minimal secure API design.
- Peripheral/memory isolation patterns.
- Enabling/disabling TZ (safe pattern).

***Exercise:** Create S + NS projects and secure LED_toggle() veneer*

***Exercise:** Demonstrate NS access fault &rarr; secure wrapper*

TF-M & Secure Manager

- TF-M secure services: RNG, storage, attestation (concepts).
- STM32Trust Secure Manager: role and enablement path.
- Placeholders vs production keys; audit trail basics.
- Rollback/unlock procedures for labs

***Exercise:** Dry-run provisioning with placeholders; secure RNG service exposed to NS; restore training state*

Robustness & tracing

- IWDG vs WWDG; service windows.
- BOR levels; reset-cause logging at boot.
- ITM/SWO prints; markers around DMA/ISR.

Renseignements pratiques

Inquiry : 3 days