

STR12 - STM32H5

This course describe 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.

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.

Modalités d'évaluation

- Les prérequis indiqués ci-dessus sont évalués avant la formation par l'encadrement technique du stagiaire dans son entreprise, ou par le stagiaire lui-même dans le cas exceptionnel d'un stagiaire individuel.
- Les progrès des stagiaires sont évalués de deux façons différentes, suivant le cours:
 - Pour les cours se prêtant à des exercices pratiques, les résultats des exercices sont vérifiés par le formateur, qui aide si nécessaire les stagiaires à les réaliser en apportant des précisions supplémentaires.
 - Des quizz sont proposés en fin des sections ne comportant pas d'exercices pratiques pour vérifier que les stagiaires ont assimilé les points présentés

- En fin de formation, chaque stagiaire reçoit une attestation et un certificat attestant qu'il a suivi le cours avec succès.
 - En cas de problème dû à un manque de prérequis de la part du stagiaire, constaté lors de la formation, une formation différente ou complémentaire lui est proposée, en général pour conforter ses prérequis, en accord avec son responsable en entreprise le cas échéant.

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 & 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 & 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 & 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

Renseignements : 3 jours