



This course covers both Cortex-M0 and Cortex-M0+ ARM CPUs

Objectives

- This course is split into 3 important parts:
 - Processor architecture
 - Software implementation
 - Hardware implementation.
 - A tutorial has been developed by ACSYS to facilitate the understanding of Cortex-M0 low level programming, therefore labs can be replayed after the course.
 - The course explains how to design a SoC based on Cortex-M0 / Cortex-M0+, clarifying the operation of the interconnect and the debug facilities integrated in the CPU.
 - This training has been delivered several times to companies developing SoCs for wireless / consumer market.
- A more detailed course description is available on request at formation@ac6-formation.com*

Prerequisites

- Basic knowledge of processor or DSP.

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
- Au début de chaque demi-journée 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 par des quizz proposés en fin des sections pour vérifier que les stagiaires ont assimilé les points présentés
- En fin de formation, une attestation et un certificat attestant que le stagiaire 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

First day

CORTEX-M0/M0+ ARCHITECTURE

- Instruction pipeline
- Internal bus matrix, fixed memory map
- Highlighting the differences between Cortex-M0 and Cortex-M3
- Implementation options
- Cortex-M0+ additional features, dual privilege levels, dual stack

ARM V6-M PROGRAMMING

- Program registers, xPSR format
- Thumb 16-bit instruction set
- Keil library functions, divide
- Barrier instruction, use cases

DEBUG

- Coresight overview
- CPU-dependent coresight units, breakpoints, watchpoints
- Vector catch
- Serial Wire Debug
- Optional Micro Trace Buffer (Cortex-M0+)

MEMORY PROTECTION UNIT - CORTEX-M0+

- Memory protection overview, ARM v7 PMSA
- Cortex-M0 MPU and bus faults
- Region overview, memory type and access control, sub-regions
- Setting up the MPU

Second day

EXCEPTION MECHANISM AND LOW POWER MODES

- Exception vs interrupt
- Automatic state saving on exception entry and exit, CISC approach
- Interrupt priority levels, nesting
- Tail-chaining and late arriving interrupts
- Fault management
- OS system call and task switching

LOW POWER MODES

- Standby and deep sleep with state retention
- Event vs interrupt
- Optional wake-up interrupt controller
- SysTick hardware timer
- Requirements for the Power Management Unit

EMBEDDED SOFTWARE DESIGN

- Application startup
- Placing code, data, stack and heap in the memory map, scatterloading
- Reset and initialisation
- Placing a minimal vector table
- Further memory map considerations, 8-byte stack alignment in handlers
- Long branch veneers
- CMSIS library

HARDWARE IMPLEMENTATION

- Bus architecture, von Neuman operation
- Single-cycle I/O port (Cortex-M0+)
- Address pipelining
- Sequential transfers
- AHB-lite specification

Renseignements pratiques

Renseignements : 2 jours