



ATELIER

Programmation temps réel avec FreeRTOS pour Coeur ARM Cortex-M

Les 10 -11 et 12 mars 2015
A Gardanne
De 9h00 à 12h30 et de 13h30 à 17h00

L'architecture ARM est connue pour offrir un des meilleurs ratios performance/efficacité du marché, tout en consommant très peu d'énergie. Ces architectures ARM ont d'ailleurs pris naturellement possession du marché de la mobilité et équipent l'essentiel des smartphones du marché et de nombreux produits embarqués industriels. Le noyau temps réel FreeRTOS est particulièrement bien adapté à ces architectures par sa faible empreinte et ses performances.

Objectifs de cet atelier :

- Acquérir une compréhension globale de l'architecture Cortex-M
- Comprendre la mise en œuvre logicielle du Cortex- M et déboguer
- Gérer les interruptions
- Comprendre l'architecture STM32F2
- Découvrir les notions de temps réel multitâches
- Comprendre les vraies contraintes de temps sur les ressources mutuelles
- Comprendre l'architecture de l'OS FreeRTOS
- Découvrir les différents services et API de FreeRTOS
- Apprendre à développer et déboguer les applications FreeRTOS

PROGRAMME DE L'ATELIER

Jour 1

Cortex- M : Présentation de l'architecture (6h30)

- V7-M Présentation de l'architecture
- Core Architecture
 - o Harvard Architecture, I- Code, D-Code et le système de bus
 - o Buffer d'écriture
 - o Bit-Banding
 - o Registres (Deux pointeurs de piles)
 - o Etats
 - o Mode de marche et niveaux de privilèges
 - o Block Control System
 - o SysTick minuterie
 - o MPU Présentation

(*) Critères d'éligibilité : Toute PME de droit français, de taille inférieure à 2000 personnes n'étant pas détenue à plus de 50% par un groupe de plus de 2000 personnes

- Programmation
 - o Alignement et Endianness
 - o CMSIS Bibliothèque
- Présentation des mécanismes d'Exception / Interruption
 - o Table des vecteurs
 - o Entrée et sortie d'interruptions
 - ♣ Queue-chaînage
 - ♣ préemption (Nesting)
 - o NVIC : contrôleur d'interruption intégré
 - o Gestion des exceptions et des priorités
 - o Escalade des défauts
- Interface de débogage
 - o Cortex- M3 JTAG et unités connexes de débogage : SWJ -DP, AHB- AP
 - o Avantages de Debug Wire Serial
 - o Mécanisme pour sélectionner le JTAG -DP ou le SW- DP
 - o Flash Patch et points d'arrêt
 - o Les données Watchpoint et Trace
 - o Instrumentation Trace Macrocell , Trace Interface Unit Port , trace entonnoir

Démo: Se familiariser avec l' IDE et clarifier la séquence de démarrage

Atelier : Gestion des interruptions sur Cortex- M3 (mis en tail-chaining/nesting)

Conseils du compilateur et astuces pour Cortex-M (30min)

- Optimisations du compilateur ARM
- Mélanger C et Assembleur
- Codage avec compilateur ARM
- L'attribut volatile
- Endianness
- Les accès non alignés, padding
- Les problèmes de données locales et globales, l'alignement, la structure

Jour 2

STM32F2 MCU : Présentation de l'architecture (1h30)

- Architecture ARM à base de noyau
- Description de l'architecture SoC STM32F20X
- Clarifier les données internes et les chemins d'instruction : Matrice Bus, AHB- Lite interconnexion , bus périphériques , AHB- à -APB ponts, DMA
- Organisation de la mémoire
 - o Interface de lecture de mémoire flash
 - o Adaptive Real-Time memory accelerator, instruction prefetch, queue and branch cache
 - o Effacement de secteur et de masse
 - o Interne SRAM
 - ♣ L'accès simultané à 112 Ko et 16 blocs KB
- SoC mapping
- Les méthodes de programmation en flash
- Configuration du Boot

Démo: copie de données en utilisant le DMA ou non

RTOS Introduction (30min)

() Critères d'éligibilité : Toute PME de droit français, de taille inférieure à 2000 personnes n'étant pas détenue à plus de 50% par un groupe de plus de 2000 personnes*

Le code source de FreeRTOS (15min)

- Introduction à FreeRTOS
 - o L'architecture et les caractéristiques FreeRTOS
- Obtenir les sources de FreeRTOS
 - o Fichiers et structure des répertoires
- Les types de données et le style de codage
 - o Les conventions de nommage
- FreeRTOS sur Cortex -M

Introduction au temps réel (15min)

- Base des concepts du temps réel
- Les contraintes temps réel
- Temps réel et Multi- tâche

Structures de données des Threads (1h00)

- Nécessité de structures de données spécifiques
- Structure de données sur pour implémentation des threads
 - o Linked list
 - o Listes circulaires
 - o FIFO
 - o Stacks
- Structures de données et preuves d'intégrité
 - o Assertions
 - o Post et pré conditions

Atelier : Construire un thread générique avec des linked-list

Élément d'un système temps réel (1h00)

- Les tâches et les descripteurs de tâches
 - o Contenu du descripteur de tâche
 - o Liste des descripteurs de tâches
- Changement de contexte
- Ordonnancement des tâches et préemption
 - o Ordonnancement « tick based »
- Les systèmes d'ordonnancement et la preuve d'ordonnançabilité
 - o Ordonnancement en priorités fixes
 - o Ordonnancement RMA et EDF
- Ordonnancement avec FreeRTOS
 - o Ordonnancement préemptif déterministe
 - ♣ Stratégies d'ordonnancement
 - ♣ Programmation cyclique (RMA)
 - ♣ Ordonnancement de limite
 - o Ordonnancement Coopératif
 - ♣ Ordonnancement hybride

Démo: Analyse d'un changement de contexte

Gestion des tâches (2h30)

- cycle de vie des tâches
 - o Création de tâches
 - o Suppression des tâches
 - o Le modèle de boucle sans fin

- Les priorités
 - o Affectation des priorités de tâches
 - o Changement de priorités de tâches
- La tâche Idle
 - o Hook de la tâche Idle
- Calendrier
- fonctions de débogage FreeRTOS (Hook, Trace)

Atelier: Gestion des tâches , se tracer informations

Jour 3

Gestion de la mémoire (1h00)

- Algorithmes de gestion de mémoire
 - o Système amis
 - o Best Fit /First Fit
 - o Gestion des pools
- Les systèmes d'allocation mémoire fournis par FreeRTOS
 - o Buddy system
 - o Best Fit /First Fit
 - o Thread-safe malloc
- Vérification de la mémoire libre restante
- Ajout d'un allocateur de mémoire spécifique à l'application
- Les erreurs de gestion de la mémoire
- Surveillance de la pile
- Dimensionnement Stack et Heap
- Les erreurs de gestion de la mémoire
- Surveillance de la pile

Démo: Ecrire un système de gestion de mémoire thread-safe

Démo: Détecter un débordement de pile

Primitives de synchronisation (1h30)

- Présentation
 - o Attente et réveil des tâches
 - o Sémaphores
 - o Événements
 - o Boîtes aux lettres
- Sémaphores binaires de FreeRTOS
 - o « Set » de sémaphore binaire
 - o « get » de sémaphore binaire
- Gestion des files d'attente de FreeRTOS
 - o Utilisation de base
 - ♣ Création
 - ♣ Envoi dans une file d'attente
 - ♣ Réception à partir d'une file d'attente
 - o La gestion des données
 - ♣ Envoi types de composés
 - ♣ Transfert de données volumineuses

Atelier : Synchronisation de tâches à travers des sémaphores binaires

Atelier : Synchronisation de tâches à travers les files d'attente



Gestion des ressources (1h00)

- Exclusion mutuelle
 - o Spinlocks et masquage d'interruption
 - o Mutex ou Semaphore
 - o Mutex récursifs ou non
 - o Problème d'inversion de priorité
 - o Héritage de priorité
- L'exclusion mutuelle avec FreeRTOS
 - o Les mutex et les sémaphores binaires
 - o Blocages
 - o Inversion de priorité
 - o L'héritage de priorité
- Les sections critiques
 - o Les sections critiques
 - o Suspension (blocage) de l'ordonnanceur
- Tâches Gatekeeper

Atelier : Mettre en œuvre l'exclusion mutuelle entre deux tâches

Parallélisme : Problèmes et Solution (1h00)

- Les problèmes de programmation parallèle
 - o Accès parallèle incontrôlé

Démonstration: Le modèle producteur-consommateur, illustrant les problèmes d'accès concurrents

- o Blocages
- o Cycle non productif
- o Starvation

Démonstration: Le problème des 'Philosophers dinner', illustrant deadlock, livelock et starvation

Management des interruptions (1h30)

- Traitement d'interruption différée par FreeRTOS
 - o Les tâches avec synchronisation d'interruption
 - o Utilisation des sémaphores dans une ISR
 - o Sémaphores de comptage
 - o Utilisation des files d'attente dans une ISR
- Traitement d'interruption par FreeRTOS
 - o ISR d'écriture en C
 - o Fonctions de sécurité d'interruption
 - o Interruption de nidification

Atelier : Synchroniser des interruptions avec des tâches

Timer logiciel (1h00)

- La Task Timer daemon
- Configuration du Timer
- One-shot / Auto -reload Timer
- API du Timer logiciel

Atelier : Mise en œuvre de Timers logiciel



INFORMATIONS PRATIQUES

Intervenant : Société AC6

Environnement de cours

- Manuel de cours
- des exemples de code, des TP et des solutions
- Un STM32F2 (Cortex- M3) avec GCC / Lauterbach pour Labs / Demos

Les prérequis minimum pour la participation à l'atelier sont les suivants :

- Connaissance des concepts et de la programmation C
- Connaissance de base des processeurs embarqués

Lieu :

Centre de microélectronique de Provence
880 route de Mimet
13120 GARDANNE

Participation aux frais :

Pour les PME adhérentes CAP'TRONIC : prise en charge totale du coût de l'inscription dans la limite de 10 hommes/jour de formation dans l'année d'adhésion, pour les ateliers traitant de logiciel embarqué.

L'inscription sera validée à réception d'un chèque de caution de **100 € TTC** par personne, qui sera renvoyé au participant après l'atelier. En cas d'absence non remplacée à l'atelier, la caution sera encaissée et une facture sera établie.

Chèque de caution à retourner par courrier à :

JESSICA France – CEA Grenoble - Bât. 51C – 17 rue des Martyrs – 38054 GRENOBLE CEDEX 09.

Si vous êtes une PME non adhérente :

Vous pouvez adhérer à l'association JESSICA France pour un montant de 500 € HT soit 600 € TTC. Pour cela contacter Janique Pernoud : pernoud@captronic.fr

Pour les entreprises non éligibles (*) et les PME qui ne souhaitent pas adhérer :

1 200 € HT soit 1 440 € TTC pour les trois journées et par personne.

Remarque : Cet atelier est éligible au financement par votre Organisme Paritaire Collecteur Agréé (OPCA) ou au DIF

Contacts :

Alain BRITON – briton@captronic.fr

Jean-Luc BAUDOIN – baudouin@captronic.fr

Contact inscriptions : Janique PERNOUD : pernoud@captronic.fr

() Critères d'éligibilité* : Toute PME de droit français, de taille inférieure à 2000 personnes n'étant pas détenue à plus de 50% par un groupe de plus de 2000 personnes