

LANGAGE C POUR APPLICATION TEMPS REEL ET EMBARQUEE

Ref : 002603A

Durée : 5 jours

OBJECTIFS

- Cette formation traite de la configuration de l'éditeur de liens en fonction du mapping mémoire d'une cible
- L'accès aux registres des périphériques en utilisant des unions et des champs de bits est vue en détails
- Cette formation aborde également la re-direction des fonction `read()` et `write()` vers des drivers de bas niveaux
- Cette formation détaille la conception et le développement d'un gestionnaire de tâches propriétaire
- La communication inter-tâches est abordée
- La mise en œuvre des bibliothèques d'entrées/sorties de ST est expliquée

FORMATIONS CONNEXES

- Formations spécifiques à un composant (voir notre catalogue)
- Linux – Programmation Noyaux, Device Driver, BSP (Ref.003154A)
- Linux – Programmation Utilisateur (Ref.003346A)

PRE-REQUIS

- Des notions de base du langage C ou d'un autre langage de programmation est recommandé, même si la syntaxe du langage C est revue pendant la formation
- Ceci n'est pas une formation langage C pour débutant

TRAVAUX PRATIQUES

- Pour les formations sur site, les travaux pratiques peuvent être fait sous les environnements suivants :
 - GNU/Lauterbach
 - Keil μ Vision
 - IAR Workbench
- Pour les formations sur site, les travaux pratiques se font sous IAR Workbench

C

Contact

Tel : 05 62 13 52 32
Fax : 05 61 06 72 60
training@mvd-fpga.com

**Le contenu peut-être adapté
pour les formations intra-
entreprise**

Prochaines sessions, voir ici : <http://www.mvd-fpga.com/fr/formationsCalend.html>

PROGRAMME

CONFIGURATION DE LA CHAINE DE COMPILATION

- Prise en main d'un environnement de développement
- Structure d'un programme en C
- Configuration du compilateur
- Configuration de l'éditeur de liens
- Sections générées automatiquement par le compilateur
- Sections définies par l'utilisateur
- Méthodologie pour processeurs ayant une MMU ou un cache
- Fichier de configuration du linker
- Code de démarrage

LE PRE-PROCESSEUR

- Inclusion de fichiers d'en-tête
- Macros
- Macros assembleur
- Compilation conditionnelle

LES TYPES ET OPERATEURS

- Classe d'allocation : register, extern, static et auto
- Variables globales et variables locales
- Les types du langage C
- Constantes
- Casting implicite et explicite
- Operateurs, priorités et ordre d'évaluation
- L'attribut volatile
- L'attribut const

LES STRUCTURES DE CONTROLE

- Les instructions sélectives : if..else, opérateur ternaire, switch..case
- Les instructions itératives : while(), do..while(), for()
- Les instructions de contrôle : break, continue, return

LES POINTEURS ET TABLEAUX

- Modes d'adressage
- Pointeurs : déclaration, initialisation
- Tableaux unidimensionnelles
- Tableaux multidimensionnelles

LES STRUCTURES ET UNIONS

- Structures : déclaration, initialisation, accès à un champ, structures imbriquées
- Tableaux de structures
- Structures à champs de bits, sensibilité à l'endianness
- L'intérêt de typedef
- Le type enum
- Unions : déclaration, initialisation, accès à un champ
- Modélisation des registres d'entrées/sorties à l'aide de structures et d'unions
- Alignements implicites des champs et bourrage

LES FONCTIONS

- Déclaration en C ANSI et en C K&R
- Attributs static et extern
- Appels de fonctions, passage de paramètres
- Valeur de retour
- La fonction main(), séquence de démarrage
- Pointeur de fonction
- Interface C/assembleur, stack frame
- Assembleur embarqué

ALLOCATION DYNAMIQUE ET LISTES CHAINEES

- L'intérêt du tas par rapport aux sections de données statiques
- Fonctions associées : malloc(), calloc(), realloc() et free()
- Description de l'algorithme d'allocation 'first fit'
- Listes chaînées : simplement-chaînées, doublement-chaînées, arbre
- Opérations de bases : insertion et suppression d'un élément

PILOTES D'ENTREES / SORTIES

- Gestion d'exception en C, table de gestionnaire du cœur ARM
- Interruptions sur niveau ou sur front
- Introductions au timers du STR712
- Pilotes génériques
- Description des UARTs du STR712
- Etude d'un driver pour les ports UARTs
- Librairies et archives
- Mise en flash d'une application

FONCTIONS STANDARD D'ENTREE / SORTIE

- Connection à un périphérique
- Fonctions standards d'entrée / sortie, stdin, stdout, stderr
- Buffer d'entrée / sortie
- Fonctions de formatage d'entrée / sortie : printf() et scanf()

- Fonctions d'entrée / sortie fichiers : fopen() et fclose()

GESTIONNAIRE DE TACHES

- Définitions : tâche et processus, préemption
- Notion d'OS orienté temps réel (RTOS)
- Etat des tâches, transitions entre états
- Multi-tâches et temps réel
- Déterminisme
- Implémentation d'un gestionnaire de tâche : définition d'un algorithme de priorité
- Gestion d'une liste de descripteurs de tâches
- Mécanisme de suspension et reprise d'une tâche
- Communication et synchronisation inter-tâches

DOCUMENTATION

Les supports de cours seront fournis sur papier à chaque participant pendant la formation.