



<http://www.alse-fr.com>

# Nouvelle Formation Pratique SoPC Altera NIOS II

V2011.04a



- Contenu** Cette formation est désormais proposée en **deux parties** pour un total de cinq jours.
- La **première partie** de trois jours concentre tout le savoir-faire **fondamental** et **essentiel** pour démarrer la conception de Systèmes avec Processeur Embarqué : construction de la plateforme hardware, choix du processeur, choix et intégration des périphériques matériels, des interconnexions et des flots de communication etc... ainsi que les bases de l'utilisation de l'environnement de développement logiciel Nios II EDS / Eclipse.
- La *deuxième partie*, destinée aux spécialistes, va plus loin en exposant les concepts hardware détaillés (système d'interconnexion et différents bus), la création de périphériques custom complexes, jusqu'à l'écriture de drivers système dans la couche d'abstraction matérielle (HAL).  
*Cette deuxième partie sera disponible en Septembre 2011.*
- Cette formation est **indispensable** pour construire un Système et des Applications performants et fiables en tirant bénéfice des riches possibilités de l'architecture SoPC (construction de flots de données et d'une *fabric* efficaces, paramétrage des interconnexions et de flux, ajout de nouvelles instructions personnalisées au processeur ou encore mise en œuvre de l'accélération matérielle).
- Théorie et Pratique** alternent à travers de très nombreux exercices décrits pas-à-pas, implémentés et testés sur un Kit Altera facilement disponible.
- Les aspects Matériel et Logiciel sont abordés dans ce séminaire pour permettre aux ingénieurs hardware et software d'avoir une vue complète des deux mondes, ce qui est *indispensable* pour concevoir un système embarqué efficace et optimisé.
- Pré-requis** Connaissances de base en langage C et de préférence en Conception FPGA Altera (VHDL).
- Durée** **Trois (3) jours soit 21 heures**, ou deux (2) jours soit 14 heures, ou cinq (5) jours soit 35 heures.
- Prix** **1 690 € HT** (partie I), 1 190 € HT (partie II) ou 2 690 € HT (Parties I + II, 5 jours)
- Lieu** Sur notre site de Montparnasse, Paris 14ème.
- Horaires** 9h30 → 18h, 9h → 18h, 9h → 17h30, soit 21 heures effectives (pour les trois premiers jours)  
9h → 18h, 9h → 17h00, soit 14 heures effectives (pour les deux derniers jours).

## Objectifs

### Partie I

- Acquérir le savoir-faire essentiel pour la Conception et la Programmation de Systèmes avec Processeur(s) embarqué(s) sur FPGAs Altera.
- Apprendre à concevoir la plate-forme hardware avec l'outil SOPC Builder
- Maîtriser l'environnement de développement IDE Nios II EDS / Eclipse.

### Partie II « Expert » – sera disponible en Septembre 2011

- Programme indicatif : Maîtriser les concepts avancés dont **QSys**, la nouvelle architecture d'interconnexion (System Interconnect Fabric) « **Network On Chip** ». Les modes complexes des différents bus, les nouveautés de la version 11, la compréhension des protocoles hardware et timings, l'optimisation de la bande passante, la création de composants custom complexes et leur intégration automatique dans la librairie Altera, la création de drivers au formalisme de la Hardware Abstraction Layer, etc...

## Pré-requis

- Avoir besoin d'implémenter ou de programmer un SOPC à base de NIOS II sur FPGA Altera.
- Connaissances de base en Électronique Digitale et en Conception de Logique Programmable.
- Connaissances de base (rudiments) en Langage C.

N.B. : notre autre formation « *Conception avec Quartus II* » n'est pas un pré-requis et elle peut être suivie séparément (avant ou après).

N.B. : nous proposons désormais des formations complémentaires « *Programmation C embarqué et OS Temps Réel* » pour acquérir les connaissances en programmation et en *Operating Systems* embarqués. Contactez-nous si ces formations vous intéressent.

## Programme\*

Remarques:

- ✓ La deuxième partie « Expert » de la formation contient des informations préliminaires et ne figure qu'à titre indicatif. Elle sera disponible en Septembre 2011.
- ✓ Noter que les Exercices de la formation utilisent un Kit Altera facilement disponible, ce qui permet de pouvoir reproduire ultérieurement les exercices après la formation.

## SOPC – Hardware : Construire la Plateforme Matérielle

### Journée 1

- **Les bases de la Conception d'un « System on Programmable Chip » (SoPC Builder)**  
Avantages et raison d'être des architectures SoPC. Introduction aux processeurs NIOS II dans ses différentes variantes. Configuration du processeur, les IPs, les périphériques standard, introduction au flot de conception SOPC avec SOPC Builder.  
Exercice : Création d'un système SOPC complet, et test sur maquette.
- **Validation du système matériel par (co)simulation HDL.**  
Création des modèles de simulation, modélisation des composants périphériques et UART, utilisation du Simulateur ModelSim.  
Exercice : Simulation RTL par ModelSim du système SoPC de l'exercice précédent.
- **Avalon System Interconnect Fabric**  
Le système d'interconnexions Avalon, les différents types de ports et les modes de transfert : Maître avec et sans wait state, Esclave et wait states, le mode Streaming, systèmes multi-maîtres et/ou multi-CPU, alignement des données et adaptation des datapaths, génération de la matrice d'interconnexion (System Interconnect Fabric)...
- **Introduction à Qsys (Beta).**  
Exercice optionnel
- **Les Périphériques Utilisateur usuels**  
SystemId, Jtag UART, On-chip memory, PIO, Timer, Nios, Jtag debug, custom.  
Exercice : Conception et Mise en Œuvre d'un périphérique utilisateur simple (PWM en VHDL).

## Journée 2

- **Accélération Matérielle et Instructions « Custom »**  
Présentation du concept. Implémentation dans NIOS II et dans l'environnement de développement. Différents types d'instructions Custom, utilisation du pipeline, exemples d'applications.  
*Exercice :* Accélération d'algorithme logiciel par Conception et Mise en Œuvre d'une instruction Custom.
- **Systèmes Multi-Maîtres et Accès Direct Mémoire (DMA)**  
Systèmes traditionnels, architecture optimisée, périphériques et transferts DMA, concepts et applications, mise en œuvre dans SOPC Builder, transferts R/W en mode Maîtres, arbitrage, mode streaming.  
*Exercice :* Mise en œuvre d'un périphérique Custom avec transferts DMA.
- **Utilisation des Cartes de développement Altera**  
Configuration des E/S, domaines d'horloge, configuration, la Mémoire Flash, le Flash Programmer, interface Flash dans système hôte quelconque.

## SOPC - Software :

### Développement Logiciel sur la plateforme SOPC

- **Le processeur NIOS II - Flot de Conception Logiciel.**  
Concepts fondamentaux, aspects logiciels, **l'environnement NIOS II EDS / Eclipse SBT**, Création d'un projet logiciel, présentation des outils de base pour compilation et mise au point, Gestion des projets, Application C/C++ et Board Support Package, Présentation de l'environnement de debug hardware.  
*Exercice :* Création d'un premier projet logiciel, compilation et test + debug sur maquette.
- **Développer des Programmes pour Nios II : la Couche d'Abstraction Matérielle HAL.**  
Introduction à la couche d'abstraction matérielle (Hardware Abstraction Layer) et aux périphériques standard : horloge système, alarmes, timers haute résolution. Accès aux périphériques, interaction avec la mémoire cache, introduction aux périphériques sur mesure (custom).  
*Exercice :* Utilisation de l'API HAL, accès aux périphériques PIOs & Timers.

## Journée 3

- **Introduction aux Architectures Logicielles des Systèmes Embarqués**  
System.h, mise à jour de la configuration matérielle, le mapping mémoire, la pile et le tas, le linker, connaître la séquence de boot (alt\_sys\_init et alt\_main), approche *Hosted*.  
*Exercice :* Observation de la phase d'initialisation de la HAL, mise en œuvre de la protection de la pile, déplacement d'une zone de code dans une mémoire physique par le linker.
- **Les Interruptions.**  
Les avantages et les risques à utiliser des traitements sous interruption. Présentation des deux modes disponibles. Avantages et implications du nouveau contrôleur cascade d'Interruption vectorisées. Mise en œuvre du contrôleur natif. L'API logicielle. Mesure de la taille du code généré.  
*Exercice :* Modification du projet PWM pour traitement par interruptions, mesure de la taille du code.
- **Custom Instructions, Accélération matérielle, le Compilateur Hardware C2H.**  
Identification des goulots d'étranglement des performances. Création et mise en œuvre d'instructions spécifiques. Accélération hardware par bloc matériel collaboratif.  
Présentation de **C2H** : générateur de modules HDL d'accélération de routines C.  
*Exercice :* Utilisation de l'accélération matérielle par instruction custom, par périphérique hardware via DMA, puis au travers de l'utilisation du **compilateur hardware C2H**.  
Mesure et comparaison des gains respectifs.