

VHDL - Initiation au langage

Ref : 003342A

Durée : 3 jours

OBJECTIFS

- Cette formation s'adresse aux ingénieurs et concepteur en électronique numérique désireux d'acquérir les bases du langage VHDL, afin de mieux connaître les possibilités dans le cadre de la synthèse logique et celui de la simulation.
- De nombreux travaux pratiques accompagnés de démonstrations permettent d'assurer la bonne compréhension des principes de base, par l'écriture d'un code générique synthétisable et de son testbench de simulation associé, permettant d'en vérifier le bon fonctionnement.

FORMATIONS CONNEXES

- VHDL - Méthodologie de conception avancée (003343A)
- Synthèse logique et simulation VHDL pour Conception de FPGA Xilinx (002572A)

PRE-REQUIS

- Connaissances intermédiaires en conception de circuit d'électronique numérique.

MATERIEL DE FORMATION

Configuration logicielle :

- Xilinx ISE
- Outil de simulation ISIM ou Modelsim

Configuration matérielle recommandée :

- Intel Core 2 ou équivalent
- Windows XP
- 1 Go d'espace disque disponible après installation des logiciels
- Au minimum 1Go de mémoire vive
- Résolution d'affichage : au moins 1024 x 768
- Pour les formations sur site, prévoir un vidéo projecteur

Contact

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

Le contenu peut-être adapté sur site

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

PROGRAMME

- Le couple entité - architecture
 - Règles principales à observer pour les déclarations de ports
 - Déclaration de signaux internes
 - Exemples
- Les objets fréquemment manipulés en synthèse logique
 - Signaux, variables, constantes
- Les types prédéfinis
 - Bit, bit_vector - limitations de ces types
 - Booléens
 - Entiers
 - Types d'objets " user defined "
- Les types STD_LOGIC et STD_LOGIC_VECTOR et package STD_LOGIC_1164
 - Avantages par rapport aux types prédéfinis pour la synthèse logique et la simulation
 - Interprétation par les synthétiseurs et les simulateurs
- Les opérateurs prédéfinis et leur utilisation étendue au STD_LOGIC_VECTOR
 - Opérateurs logiques
 - Opérateurs relationnels - pièges à connaître
 - Opérateurs arithmétiques
- Règles à observer pour l'affectation des vecteurs de données
 - Exemples typiques
- Instructions concurrentes et règles d'utilisation
 - When ... else
 - With ... select
 - For ... generate
 - Implémentation d'un design d'exemple incluant
 - Multiplexeurs
 - Compérateurs
 - Décodeurs 7 segments
- Instructions séquentielles et règles d'utilisation
 - If ... else
 - Case
 - For ... loop
 - Designs d'exemples, Compteurs synchrones
 - Initialisables
 - Chargeables
 - Up-down
 - Cascadables
 - Implémentation d'une machine d'états et simulation de celle-ci
 - Implémentation de plusieurs designs d'exemple, afin de bien mettre en évidence les caractéristiques fondamentales de ces instructions
- Initiation à la simulation en langage VHDL
 - Quelques nouvelles instructions exclusivement liées à la simulation
 - After
 - Wait for
 - Simulation des designs d'exemples développés précédemment
- Gestion de la hiérarchie et VHDL structurel
 - Comment découper le design en modules interconnectés
 - Utilisation de cores et IP
 - Utilisation de primitives spécifiques de la technologie ciblée et simulation de celles-ci
- Attributs prédéfinis et notions de généricité
 - Range, reverse_range, length, left, right...
 - Exemple d'utilisation
 - Introduction aux sous-programmes (fonctions et procédures) et mise en évidence de l'utilisations des génériques
 - Implémentation d'un design hiérarchique d'interface micro-processeur et timer programmable avec ligne d'interruption

DOCUMENTATION

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