

## This training is intended to professional who want to use or maintain programmable components

### Objectives

- Knowing the programmable logic basics
- Knowing the general offer for CPLDs and FPGAs
- Understand application description in HDL
- Understand the logical synthesis notions and process flow
- Discover FPGA programming in VHDL and Verilog
- Understand how to elaborate and simulate a design

### Course environment

- A PC in pairs
- Xilinx ISE Design Suite 14.4 Webpack Edition
- A Nexys-3 (Xilinx Spartan6-based) board

### Prerequisites

- Knowledge of digital technology
- Concepts of Boolean algebra
- Some programming concepts are desirable (whatever language)

### 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
- Activités pratiques
  - Les activités pratiques représentent de 40% à 50% de la durée du cours
  - Elles permettent de valider ou compléter les connaissances acquises pendant le cours théorique.
  - Exemples de code, exercices et solutions
  - Pour les formations à distance:
    - ▶ Un PC Linux en ligne par stagiaire pour les activités pratiques, avec tous les logiciels nécessaires préinstallés.
    - ▶ Le formateur a accès aux PC en ligne des stagiaires pour l'assistance technique et pédagogique
    - ▶ Certains travaux pratiques peuvent être réalisés entre les sessions et sont vérifiés par le formateur lors de la session suivante.
  - Pour les formations en présentiel:
    - ▶ Un PC (Linux ou Windows) pour les activités pratiques avec, si approprié, une carte cible embarquée.
    - ▶ Un PC par binôme de stagiaires s'il y a plus de 6 stagiaires.
  - Pour les formations sur site:
    - ▶ Un manuel d'installation est fourni pour permettre de préinstaller les logiciels nécessaires.
    - ▶ Le formateur vient avec les cartes cible nécessaires (et les remporte à la fin de la formation).
- Une machine virtuelle préconfigurée téléchargeable pour refaire les activités pratiques après le cours

- Au début de chaque session (demi-journée en présentiel) 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 de deux façons différentes, suivant le cours:
  - Pour les cours se prêtant à des exercices pratiques, les résultats des exercices sont vérifiés par le formateur, qui aide si nécessaire les stagiaires à les réaliser en apportant des précisions supplémentaires.
  - Des quizz sont proposés en fin des sections ne comportant pas d'exercices pratiques pour vérifier que les stagiaires ont assimilé les points présentés
- En fin de formation, chaque stagiaire reçoit une attestation et un certificat attestant qu'il 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**

## **From the logic gate to the CPLDs and FPGAs**

- Reminder on digital electronic
- Structure of an Integrated Circuit
- SSI (small scale integration), TTL
- MSI (medium scale integration), PALs, GALs, PLDs
- LSI (large scale integration), CPLDs
- VLSI (very large scale integration), ASICs, ASSPs, FPGAs
- Logical architectures evolution
- The various components
- Technologies available on the market
- Technology constraints
- Interconnection methods (SRAM, Fuse, AntiFuse, Flash)
- Clock distribution
- Logic element types
- Timing issues

## **HDL Contribution**

- Interest of HDL programming
  - VHDL
  - Verilog
- Different steps of the design
  - Programming
  - Simulation
  - Synthesis
  - Mapping
  - Place and Route
  - Timing Analysis
  - Bitstream generation
- Definition of a project
- Structure of a program

- Allocation of PIN-OUT
- Programming

*Exercise : Understanding the steps of design and programming:*

- *Getting started with the ISE IDE*
- *Creating a project from scratch*
- *Synthesis, Translate*
- *Map*
- *Place and Route (PAR)*
- *BitGen*
- *Report Analysis*
- *Assigning I/O locations using PlanAhead (editing constraint file)*
- *Schematics*
- *Analyzing the placement*
- *Flashing with Impact*

## **Second Day**

### **Schematic Editor**

- The schematic capture
- Primitives and symbols definition
- Resources definition
- Compilation

*Exercise : Developing a new IP with the Schematic Editor, Designing a Bound Detector*

### **HDL Basic Concepts (VHDL and Verilog)**

- Entity/ Architecture and Module
- Signals and wires
- Processes and Always/Initial statements
- Connecting existing IPs together

*Exercise : Adding a 7-Segment Display to your design*

### **Test benches and simulation**

- HDL instructions specific to simulation
- Functional and behavioral simulation (with delays)
- Test vector generation

*Exercise : Getting started with the ISIM simulator, developing a tesbench and simulating the previous designs*

## **Renseignements pratiques**

**Renseignements : 2 jours**