USEEV5 - FPGA Platforms: Programmable Embedded Systems

Mis à jour le 05-07-2024

Code : USEEV5

Unité spécifique de type cours

3 crédits

Responsabilité nationale : EPN05 - Informatique / Sami TAKTAK

Présentation

Prérequis

- Basic knowledge in Electronics and Digital Circuits.
- Programming and Algorithmics
- Boolean Algebra

Objectifs pédagogiques

This FPGA course's aim is to equip students with the knowledge to design and implement digital systems using Field Programmable Gate Arrays. Key focuses include understanding FPGA architecture mastering HDLs like VHDL/Verilog and practical skills in synthesis, simulation, and debugging. This prepares students for innovative application in technology.

Programme

Contenu

This course introduces the core concepts and applications of Field Programmable Gate Arrays (FPGA architecture, utility, and hardware description languages (HDLs). Students will learn the FPGA development process, including synthesis, simulation, timing constraints, and debugging.

- Introduction to FPGA (theoretical chapter)
  - Definition and utility of FPGA
  - FPGA Architecture
  - Hardware Description Languages (HDL)
- Overview of FPGA Development Process (theoretical/short exercises)
  - Synthesis and Simulation
  - Timing constraint and Floorplanning
  - Implementation
  - On-Board Testing and Debugging
- First Supervised lab (SL): 4x7-Segment Stopwatch
  - Objective: Design and implements a stopwatch using a 4x7-segment display.
  - Key Concepts: Timing mechanisms, multiplexing for display control.
- Second SL: Crossroad Management
  - Objective: Create o traffic light control system for managing crossroads.
  - Key Concepts: Finite State Machines (FSM).
- Third SL: Image Filtering and Display
  - Objective: Develop a system to apply a filter to an image and display the result on a screen.
  - Key Concepts: Basic image processing techniques, FPGA-based signal generation for video display (VGA or HDMI), and the implementation of digital filters in hardware description languages.
- Project: Pong game (or custom project).
  - Objective: Implement the classic Pong game, controlling paddles to bounce a ball back and forth on a screen.
  - Key Concepts: Real-time input processing for paddle movement, collision detection between the ball and paddles, generating graphics for game elements, and FSMs for game state management.

Complementary content:

Many more advanced applications can be considered after this introduction, including network packet processing, soft-core implementations, AI algorithm acceleration, etc.

Modalités de validation

- Contrôle continu
- Projet(s)

- Examen final

Description des modalités de validation

Continuous monitoring/lab reports, project.