CND111: Introduction to Digital Design

Course Description:

In this course, we will present a review of digital logic design. We will explore various digital design concepts such as: implementation technologies, timing in combinational and sequential circuits, digital design EDA tools, arithmetic units (adders, multipliers, …), introduction to simulation and synthesis using Verilog, implementation Using FPGA development board and design flow.

Learning Outcomes

After successful completion of this course, the student will be able to:

- Acquire logic design knowledge for advanced study topics related to computer architecture, microprocessor systems, VLSI design, design automation, etc.
- Perform basic logic circuit design and minimization including both combinational and sequential circuits.
- Perform design of finite state machine and arithmetic circuit design.
- Understand the timing constraints of digital circuits.
- Use hardware description language Verilog and EDA design tools to perform modern digital circuit design.
- Use the FPGA development board and design flow to implement digital systems.

Lecture Schedule

- Course overview, review of logic design and design methodologies of digital systems.
- Review of Boolean logic and minimization techniques: logic circuit design, K-maps, Two Level Logic Minimization Algorithms (Quine McCluskey Method, branch and bound approach, …)
- Combinational circuit design. Introduction to Verilog HDL
- Modern Digital Design Tools – Verilog in depth.
- Sequential Circuit Design and Timing Analysis
- Programmable Logic Technologies
- Arithmetic Circuits Design – Adders, Multipliers
- Finite State Machine Design and Optimization
- Introduction Advanced Logic Design: Digital System Design, Power Analysis and Technology Integration

Lab Sessions

The student will work towards FPGA Design Flow, from high-level functional specifications, creating source files (Verilog(R) files, I/O pin constraints, timing constraints, …, etc.) design translation (Synthesis), mapping, placement, and routing for use with Intel's Quartus (R) software for the FPGA development board. TCL scripting might be used to integrate all design assignments in the compilation and optimization flows. The lab will cover extensive use of Verilog for describing and implementing digital logic designs. Throughout each session, lab assignments are designed to prepare the student for the course project.

- **Lab 1**: Introduction to FPGA Design Flow
  The main objective of this lab is to introduce the students to the fundamentals of Intel's Quartus design flow and its tool set such as the synthesis tool, the Questa simulator, and the FPGA implementation.

- **Lab 2**: Implementing Multi-function Gate
  Design, model, and simulate a Multi-Function Gate using both behavioral and structural description methodologies.

- **Lab 3**: Binary Adders
  Design, model, and simulate binary adders and subtractors and compare the synthesis results using behavioral and structural methodologies.

- **Lab 4**: Combinational Logic Blocks
  Design, model, and simulate the procedural descriptions for combinational blocks such as a multiplexer, decoder, and comparator. At the end of this lab, the trainee will build and implement a complete arithmetic logic unit on the FPGA development board.

- **Lab 5**: Sequential Circuits: Flip-Flops & Registers
  Build behavioral and structural models of different types of flip-flops, latches, and registers. The timing simulation is carried out to investigate the functionality. At the end of this lab, the trainee will build and implement a register bank on the FPGA development board.

- **Lab 6**: Sequential Circuits: Counters and Shift Registers
  Build behavioral and structural models of different types of counters and shift registers. The timing simulation is carried out to investigate the functionality.

- **Lab 7**: Arithmetic Circuits
  Build behavioral and structural models of different types of adders and multiplication algorithms. The synthesis results of each structure are investigated and compared. The timing simulation is carried out to investigate the functionality.

- **Lab 8**: Controllers and Finite State Machines
  This lab exercise concerns the design and implementation of digital system controllers using finite-state machines. The simulation is carried out to investigate the functionality and to test its completeness.