

# ECE440 Digital Systems Engineering

## Spring 2005 Class Project

### Report Guidelines Supplement

---

Objective: Find and interpret the FPGA resource utilization report generated by the Xilinx synthesizer. Gain insight into decision the synthesizer made in order to implement your design, and determine whether this particular chip is appropriate is an appropriate choice for your burglar alarm project.

Note: One report per team.

This document describes the FPGA resource utilization information that you are to provide with you final class project report.

#### FPGA Resources

You will have synthesized and implemented your design on actual hardware. By doing so, you should have got a sense of the difference between computer-based simulation and verification on the hardware itself.

As part of your report, you are to answer the following.

#### Synthesis Report

In the Xilinx ISE, select your source code, which will cause the Synthesis menu to be displayed. Under **Synthesis - XST**, select **View Synthesis Report**. Look for the **Final Report** section. Answer the following questions.

#### Design Statistics

How many of each kind of RTL module did the synthesizer use?

- IO Buffers (IOBs)
- ROMs
- Registers
- Multiplexers
- Tristate buffers
- Adders/Subtractors
- Comparators

#### Device Utilization Summary

- Number and percent of:

- Configurable logic slices
- Slice flip-flops
- 4-input lookup tables (LUTs)
- IO buffers
- Global clocks

### **Timing Summary**

According to the synthesizer, what is the minimum clock period and maximum clock frequency that the your design can support on this chip?

### **Interpretation**

Based on the synthesis report, do you think you have come close to using all the resources in this Spartan FPGA? Have you come close to exceeding the maximum operating frequency? Could you get by with a smaller and cheaper FPGA or CPLD?

### RTL Schematic

Under **Synthesis – XST** select **View RTL Schematic**.

Explore the RTL schematic by descending into the various blocks to see their logic circuit equivalents.

Do you see anything that tells you about how the synthesizer implemented your design? Can you tell what kind of state encoding the synthesizer selected?

Does this tell you anything about the advantages or disadvantages of designing with a hardware description languages vs graphical design entry?