CSEE W4840 Embedded System Design Lab 4

Stephen A. Edwards

Due February 24, 2005

Abstract

Design and implement, in synthesizable VHDL, a circuit that sums the contents of an on-chip memory. Validate it using the simulator, run it through logic synthesis, and verify it works on the FPGA.

1 Introduction

For labs 1 and 2, you treated the XSB-300E as mostly a target for C programs. If this were our only objective, we would have chosen a board with, say, a small processor and a few peripherals. Instead, the board has a very flexible FPGA and beginning with the lab, you will get a chance to take advantage of this flexibility by designing and implementing your own hardware. For this lab, you will only design hardware; your simple system will not include the Microblaze processor or any related components.

2 The Assignment

In `~sedwards/4840/lab4.tar.gz`, you will find a simple project that contains a simple ROM represented using the template discribed in class and displays "00" on the two LED displays using a pair of simple hex-to-seven-segment decoders.

Your job is to add a controller (i.e., some sort of state machine) that sums the contents of this ROM and displays the result on the LEDs. Add an accumulator, an adder, and a counter that generates addresses for the ROM. Draw a block diagram of these components and a timing diagram showing signals such as the ROM address, the data coming from the ROM, and the accumulated sum.

As usual, we have provided a skeleton of the code for this lab, located as usual at `~sedwards/4840/lab4.tar.gz` on the ilab machines. This tarball contains a Makefile, two VHDL files (suffix ".vhd"), and a few configuration files that together produce a bitstream for the FPGA that counts in hex on the LEDs on the XSB-300E board. "make download" will compile the VHDL source files, place and route them, produce a bitstream, and download it to the FPGA. "make sim" will run the simulation, allowing you to examine the detailed behavior of your system.

Show your working solution to the TA, have him sign off on it, and turn in a listing of your VHDL source files along with block and timing diagrams.

As usual, short, elegant solutions (this is more difficult in VHDL, but not impossible) will receive better grades than messy ones.