Using the Xilinx Verilog Flow for Efficient High-Speed Design

A real-life example of a Verilog design that runs as fast as a schematic-based design; a testimonial to an excellent tool flow and to the capability of XC4000XL FPGAs.

by Rob Weinstein, Senior Member Technical Staff, and Timothy Smith, Managing Director, Memec Design Services, rob_weinstein@memecdesign.com, tim_smith@memecdesign.com

For many years we have heard that to get real speed and performance out of FPGAs you must use a schematic design flow. However, this is no longer true. The following is an example of a Xilinx design that had to work at 100MHz, and we completed it using the Xilinx Alliance Series 1.5 tools, Synplicity, and the Silos III simulator. This design flow was significantly faster than schematic entry (we estimate a 40% time savings).

Design Overview

This design is used to characterize high-speed analog to digital converters. It accumulates histogram data on 14-bit data words, at 100M words per second. Each incoming 14-bit data value is presented to the address bus of an external SRAM. The data value stored at that address is read into the FPGA, incremented by one, and written back to the same location. It basically sorts the input samples into bins corresponding to the value of the sample. For instance, each time an input sample of 0 is detected, the "0" bin is incremented, likewise, each time an input sample of 42 is detected, the "42" bin is incremented. There is a bin for each possible input value.

The input samples are directed sequentially to four separate SRAMs. This allows 40ns for each read-increment-write cycle while maintaining a 10ns sample period. When the acquisition is complete, the SRAM values can be read out of the FPGA for analysis.

The FPGA also contains logic to keep track of the maximum deviation between adjacent samples. The maximum deviation value is computed every 10ns.

Device utilization is shown in **Figure 1** and a block diagram is shown in **Figure 2**.

Device Utilization

Device	CLBs Used	Flip Flops	4 Input LUTs	3 Input LUTs	IOBs Used	Max Clock Speed
XC4013XL-09 PQ240C	303 (52%)	307	329	69	160 (83%)	100MHz

Figure 1

Block Diagram

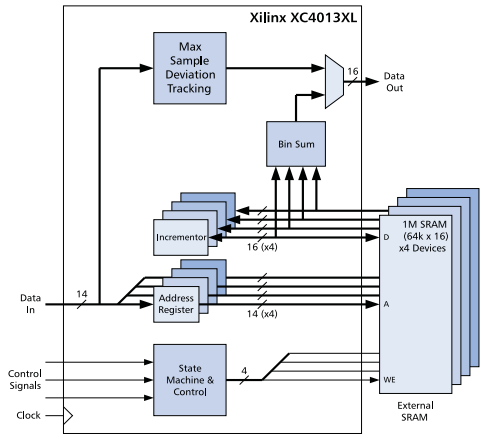

Figure 2

Design Flow

We have been using schematic entry for Xilinx designs since 1992; this year we switched to Verilog. We were somewhat apprehensive about using an HDL. Using Synplicity, however, with its HDL Analyst, let us see the synthesized implementation, in schematic form, without performing a place and route. This let us hone our coding style to achieve one level of logic between registers.

In addition, Synplicity does an excellent job of using the Xilinx carry logic. It builds optimized arithmetic functions to whatever word width I am using. Using schematics, it is a tedious process to modify "standard" width arithmetic functions to the width I need.

Functional simulation is also much faster with an HDL simulator than with the gate-level simulator used with schematic designs. Also, using Verilog as a simulation language is easier for developing complete test benches than the command language of schematic based gate-level simulators. For example, it was trivial to model the external SRAMs along with the Xilinx device. You basically have a complete programming language at your disposal.

Conclusion

This example demonstrates the power of Xilinx FPGAs for implementing high-speed designs as well as the strength of using a High-level Design Language such as Verilog. Using an HDL, you can achieve design performance that equals schematic designs.

For more information on Memec Design Services, contact: Maria Aguilar, Design Coordinator, Memec Design Services, (888) 360-9044, (602) 491-4311, maria_aguilar@memecdesign.com, http://www.memecdesign.com, or contact your local Insight Electronics office: (800) 677-7716, http://www.insight-electronics.com