Using an FPGA Processor Core and Embedded Linux for Senior Design Projects

Tyson S. Hall
School of Computing
Southern Adventist University
Collegedale, TN 37315
tyson@southern.edu

James O. Hamblen
School of Electrical and Computer Engineering
Georgia Institute of Technology
Atlanta, Georgia 30332-0250
hamblen@ece.gatech.edu

Abstract

This paper describes our experiences using a low-cost SoPC FPGA board and an open source RTOS for senior design projects. The Altera DE2 is a small low-cost FPGA-based SoPC system designed for educational use. It has many of the I/O features typically found in a standard PC and can support a Nios soft processor core. Students developed a custom Nios processor design for the board and customized a µClinux OS to support their senior design projects.

1. Introduction

The market for embedded systems has rapidly grown such that embedded processors alone account for over 98 percent of the $50 billion processor market [1], [2]. In addition, the market for ASICs and FPGAs, currently a $2 billion industry, is expected to grow to more than $5 billion by 2010 [3], [4]. This explosive growth in the embedded systems market has been fueled by rapid prototyping technologies. The ability to quickly program microprocessor memories in-circuit and reconfigure field-programmable digital hardware is critical to embedded systems engineers who must meet ever shortening development cycles.

Fast development times are also important in educational environments where student design projects must be completed within a 10-15 week term, and individual projects are often confined to a three-hour laboratory period. As the size of a single FPGA has increased to several million gates, it has now become practical to consider adding a processor cores to the FPGA chip for System-on-Chip (SoC) applications. Soft IP cores such as Altera's Nios and Xilinx's MicroBlaze use existing programmable logic elements from the FPGA to build the processor. Soft cores are more flexible than fixed VLSI layouts, but they have slower clock rates and use more power. ASICs still offer higher performance, but for student projects requiring an actual implementation, the FPGA-based system-on-a programmable chip (SoPC) approach is faster and more economical.

FPGA processor cores are now supported by a number of embedded real-time operating systems (RTOS). By incorporating an RTOS into their projects, students can explore custom driver development and operating system enhancement (for open-source operating systems). This also makes FPGA-based boards more appropriate for both computer engineering and computer science projects.

Open source RTOS's are particularly attractive for educational use because of the availability of source code and the low/no cost. One open source operating system that has been successfully ported to FPGA processor cores is µClinux, which is a version of the open-source Linux kernel that does not require a memory management unit (MMU). This is an important feature of µClinux because FPGA processor cores typically do not include a memory management unit (MMU) due to the large gate count requirement.

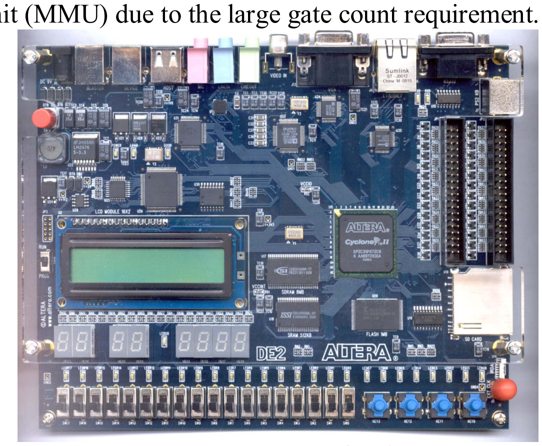

Fig. 1. Altera's DE2 Educational SoPC board with a Cyclone II FPGA and external memory that can run the Nios II soft processor core.

2. Hardware Platform

The Altera DE2 FPGA SoPC educational board [5] seen in Figure 1 uses a Cyclone II FPGA with 512 KB of SRAM, 8MB SDRAM and 4MB of Flash and a full range of I/O interfaces. The large Cyclone II FPGA is capable of hosting a Nios 32-bit RISC soft processor core. FGPA CAD Tools to develop the Nios processor hardware design and a customized C compiler are available free to schools from Altera. An overview of the SoPC design process can be found in [6]. Several introductory tutorials that work though the tool flow to develop a Nios design can be found in [7].

3. Senior Design Projects

Combining FPGA processor cores and RTOS's in senior design course projects provides a unique system design experience for students. Projects that combine hardware and software design offer a wide array of real-world problems that must be solved. Students can experiment with partitioning a computational module between software and hardware and compare the trade-offs in performance, cost, and power between different solutions.

Using FPGAs for the hardware development platform also gives students the opportunity to incorporate sensor interfaces, memory interfaces, and custom hardware modules with the soft-core processor to expand the functionality of the processor and operating system. In addition to exercising their hardware design skills, interfacing unique devices to the processor often requires students to write or customize device drivers in the RTOS. Adding this component to design projects provides explicit ties back to students' operating systems and programming courses.

A recent student design project used the DE2 board to implement a custom Nios II FPGA processor core running µClinux [8,9]. The students created the hardware and software necessary to extend the µClinux operating system to output color graphics to a VGA monitor attached to the DE2 board as shown in Fig. 2. On the hardware side, the students designed a VHDL hardware module that implemented a rudimentary graphics card. External SRAM was used to store the image pixel data, which was then accessed via a memory interface block in the VHDL module. A custom VHDL module also implemented a color palette look-up table and an interface to the external video DAC chip that controls the physical VGA output. On the software side, the students modified the MicroGUI graphics subsystem to be compatible with their custom graphics hardware.

This project required the students to partition the graphics functionality they desired between its hardware and software components and then implement both modules. The students were exposed to hardware/software co-design, operating systems, reconfigurable hardware, and system design concepts.

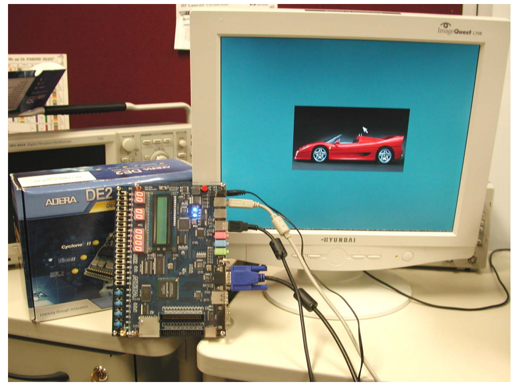

Fig. 2. The Altera DE2 board's Nios processor running the µClinux OS using a custom color VGA controller.

4. Conclusions

Overall, using FPGA-based SoPC boards along with an embedded OS for student design projects has enabled the development of more involved senior design projects. The design projects have increased in complexity and sophistication with the introduction of SoPC technology along with an embedded OS. Using a general purpose SoPC board saves time and money since the boards can be reused several semesters for a wide variety of projects.

5. References

[1] D. J. Jackson and P. Caspi, "Embedded systems education: future directions, initiatives, and cooperation," SIGBED Rev., vol. 2, no. 4, pp. 1–4, 2005.

[2] J. Turley, The Two Percent Solution, Embedded Systems Programming, Dec. 2002.

[3] In-Stat, FPGA Market Will Reach $2.75 Billion by Decade's End, May 2006.

[4] In-Stat, Structured ASICs: Designing For The Future, 2005.

[5] Altera Corporation, Embedded Processor Web Site, http://www.altera.com

[6] T. S. Hall and J. O. Hamblen, "System-on-a-Programmable-Chip Development Platforms in the Classroom," IEEE Transactions on Education, vol. 47, no. 4, pp. 502-507, Nov. 2004.

[7] J. O. Hamblen, T. S. Hall, and M. Furman, Rapid Prototyping of Digital Systems Quartus II Edition, Springer, August 2005.

[8] µClinux distribution, http://www.uclinux.org

[9] µClinux patches for Linux and Nios, http://nioswiki.jot.com/WikiHome/OperatingSystems/TryOutuClinux