PROPIC: A HARDWARE/SOFTWARE PLATFORM FOR THE SYNTHESIS OF MICROCONTROLLER-BASED SYSTEMS IN FPGA

Higor Hostins *Júlio C. Fiamoncini* *Adhemar M. V. Filho* *Cesar A. Zeferino*

Universidade do Vale do Itajaí – Centro de Ciências Tecnológicas da Terra e do Mar
Rua Uruguai, 458 – 88302-202 – Itajaí – SC – BRAZIL
E-mail: {higor_hostins, juliocf, adhemar, zeferino}@inf.univali.br

ABSTRACT

Microcontrollers are largely used in the industry, being applied in automation and consumer electronics products as an embedded computer dedicated for control tasks. In this paper we present works being developed in ProPIC Project, which aims the developing of a hardware-software platform for the synthesis of embedded applications in FPGA. The goal is to integrate, into a single device, a microcontroller and all the digital glue logic needed for the target application. In this sense, we have developed a VHDL model of an application specific instruction-set processor (ASIP) compliant with the architecture of Microchip PIC16 microcontroller family; a tool to allow the automatic customization of the ASIP for the target application; and a set of software drivers for interfacing with peripheral devices. This paper presents details about this project and some current results of ongoing works.

1. INTRODUCTION

In the design of a microcontroller-based product, the designer must select the microcontroller model that best meets the requirements of the target application, which include: costs, performance, size, weight, power consumption, and easiness for maintenance and updating. However, in some projects, it is necessary to include additional logic devices to improve features like I/O capability and protocol adaptation, for instance. This increases the system size and reduces its reliability. Meanwhile, some microcontroller-based products must be as small and light as possible, and they present strong requirements in respect to their size and weight. In these cases, the designer needs to increase the integration level by reducing the number of devices.

One solution for this problem relies on the integration of all digital circuits (microcontroller and glue logic) into a single FPGA (*Field Programmable Gate Array*), which allows to reduces the system size and weight, and also increases the reliability and the flexibility for future updates. However, it is necessary to have a synthesizable model of the target microcontroller.

In this context, we are developing a hardware-software platform (named ProPIC) based on the integration of microcontroller-based systems in FPGA, supported by developing tools and a set of software drivers for interfacing with devices commonly used in electronic products, like displays, keyboards and memories. The microcontroller is implemented as a parameterizable VHDL model compliant with the architecture of Microchip PIC16 microcontroller family [1]. We use an ASIP (Application Specific Instruction-set Processor) [2] approach in order to ensure that only the resources needed by the target application are synthesized.

2. THE DESIGN FLOW

The design flow is shown in Fig. 1, where the models and tools being developed in this project are represented as gray filled objects. The developing of an application, begins with the writing of the application source code in C language by using a library of device drivers (*dds*). A commercial C compiler tool is then used to generate the machine-level code, available as a .hex file. After the source code compilation, the .hex file is submitted to the customization tool, which analyses this code and defines the parameters that it needs to customize the parameterizable VHDL code of the PIC16 microcontroller. Such customized VHDL model is then submitted to a commercial VHDL compiler tool in order to synthesize the system according to the application requirements.

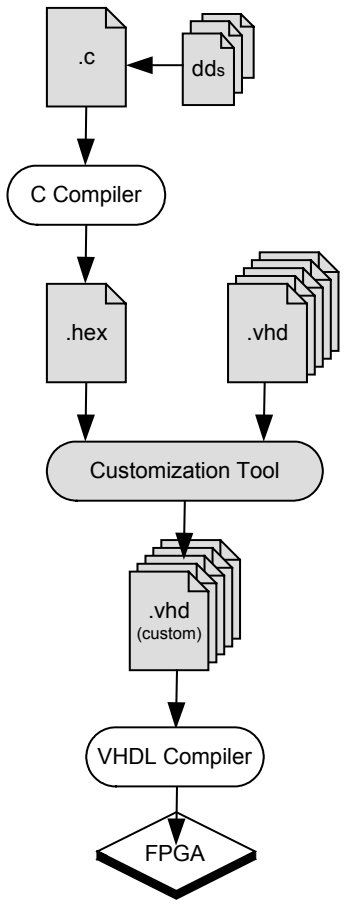

Fig. 1. Design flow.

2.1. Software

As it is shown in Fig. 1, the software part of ProPIC is composed by device drivers, application code, customization tool, and the C and VHDL commercial compilers.

The device drivers are coded in C language and include functions to drive a set of commonly used devices, like: I^2C devices (memory and real-time clock), phone keyboard, A/D converter, 7-segment display, step motor and LCD. It is used a layered structure composed by device-dependent and -independent layers. The application uses the same functions to drive any device: *dev_write* (to send a character), *dev_read* (to get a character), and *dev_execute* (for special commands).

The customization tool is also implemented in C language. It is responsible for the following tasks:

1. to identify the resources needed by the application: instructions, peripherals, stack and data memory sizes, etc;
2. to generate the program memory file in VHDL (rom.vhl); and
3. to configure the parameters of the VHDL model.

The commercial compilers used in this project include CCS C Compiler (for PIC microcontrollers) and Altera Quartus II VHDL compiler (for synthesis in FPGA). We also use other tools of Altera Quartus II for validation and analysis.

2.2. Hardware

The hardware part of the design flow relies on the VHDL models for the PIC microcontroller and on the additional logic needed for integration, beyond the FPGA.

The PIC microcontroller family chosen for this project is the PIC16, which is based on a RISC CPU core with 35 instructions, organized using a Harvard structure, with an 8-bit data bus and a 14-bit instruction bus.

The first version of the VHDL model is based on the PIC16F84 device and has the following features:

- Program memory: up to 2048 14-bit words;
- Data memory: up to 116 8-bit words;
- Stack: up to 8 11-bit words;
- Timer an 8-bit timer with prescaler;
- Interrupts: timer and externals;
- I/O ports: a 5-bit port and an 8-bit port.

A new version of the PIC model will be based on the features of other device of the same family (probably the PIC16F877 model), which use the same CPU core but includes more I/O pins and peripherals.

The organization of the VHDL model was defining by using the "data path – control path" model [3]. The data path includes the registers, operators and interconnections, while the control path includes the state machine which sequences the operations by selecting the interconnections, registers and operators which must act at each step of the implemented algorithm. Fig. 2 shows a simplified scheme of the VHDL model organization.

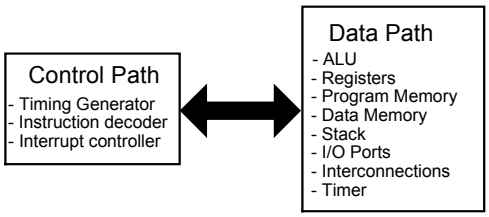

Fig. 2. Simplified organization of the PIC VHDL model.

3. IMPLEMENTATION AND VALIDATION

The device drivers was described in C language and validated by using the Microchip PICDEM2 development board, where a PIC16F84 microcontroller was attached and the devices were connected.

The customization tool is under development and it is not still ready for validation.

The first hardware model, which is compliant with PIC16F84 microcontroller, was described in VHDL and synthesized by using Altera Quartus II, targeting the EPF10K70RC240-4 FPGA, from Altera. It has been validated by using the simulator tool available in Quartus II, and also with its prototyping in the Altera UP2 development board.

In the validation of the VHDL model, we have verified the operation of each instruction and also the execution of simple applications. For instance, one of the used applications is based on an assembly source code available in [4], where the microcontroller has two inputs connected to on/off switches, and eight outputs connected to a 7-segment display (with a decimal dot). The application uses the timer to make a time counting, viewed on the display, which can be controlled (played or paused) by using the switches. This application uses 103 program words, 8 data memory positions, 3 levels on the stack and only 17 of the 35 instructions of the CPU core. Its synthesis in FPGA resulted in an ASIP which consumes 1019 logic cells and 223 flip-flops of the target FPGA. The operating frequency was limited to 12.6 MHz.

4. CONCLUSION

This paper presented the works being developed in the context of ProPIC Project, which aims the developing of a hardware-software platform for the synthesis of embedded applications in FPGA. Nowadays, they are ready the first versions of the device drivers and the first microcontroller model. Ongoing works includes the development of the customization tool and a new version of the VHDL model which will include more peripherals and I/O capabilities.

5. ACKNOWLEDGMENTS

This work is supported by FUNCITEC (*Fundação de Ciência e Tecnologia de Santa Catarina – Edital 002/003*) and by CNPq (*Conselho Nacional de Desenvolvimento Científico e Tecnológico – Programa PIBIC*).

6. REFERENCES

[1] Microchip. *PIC16F8X: 18-pin Flash/EEPROM 8-Bit Microcontrollers*. Chandler: Microchip Technology, 1998.

[2] S. A. Ito; L. Carro; R. P. Jacobi. *Making Java work for microcontroller applications*. **IEEE Design & Test of Computers**, Los Alamitos, v.18, n.5, p.100-110, Set-Out 2001.

[3] C. T. Clare, *Design logic Systems Using State Machines*. New York: McGraw-Hill, 1973.

[4] D. J. Souza. *Desbravando o PIC*. São Paulo: Érica, 2000.