Soft Core Based Model of a Microcomputer Family[1]

Nguyen Quang Trung and Krystyna Siekierska
Department of Integrated Circuit and System Design
Institute of Electron Technology, Poland
{nqt,siek}@ite.waw.pl

Abstract

A significant trend towards developing soft core based designs has been observed for the last few years. The emphasis is on reduction of time to market and bridging the growing gap between technology capacity and design productivity. The paper briefly presents the design of a configurable microcomputer family model based on a VHDL soft core. Modeling process with emphasis on coding style is mentioned. The paper also proposes a virtual component configuration tool that gives the core user an effective way to set up the model of the microcomputer family upon the need of particular end-users, and introduces the implementation of industrial microcomputer series based on the model. The work on the extension of the soft core with regard to its reusability, as well as on the development of a configurable model of microcontroller class based on the soft core is being done.

1. Introduction

The availability of sub-micron manufacturing processes allows the growth in gate density by 60% per year. It is expected that this exponential growth will continue for the next years that allows integrating tens of millions of gates in a single chip. This progress however increases the need for effective design methodologies that can be accomplished with systematic use of already designed and verified hard, firm and soft cores. Over the last few years, soft cores have been used quite widespread due to high degree of flexibility and portability that they provide.

The design methodology based on VHDL and RTL synthesis that has been being applied at the ITE for the last few years allows obtaining designs of a number of soft cores. Based on one of them, a synthesizable model of a microcomputer family has been defined and developed.

2. Structure of the microcomputer family

The microcomputer soft core has been designed following the Harvard architecture scheme. Model of the core was partitioned into two parts: a datapath and a sequencer core [6]. The sequencer core consists of a set of finite state machines that input external control signals and status signals from the datapath, and generates control signals for conditional assignments of data into its registers or buses. The datapath of the soft core contains an arithmetic-logic unit, common-use registers, memory pointers, stack registers, and an interrupt unit.

The VHDL entities of the soft core components have been described with generics that allow users to configure the components as required in the design. The structural description of the core has been constructed to make it easy to modify.

The coding style complies with RTL coding guidelines recommended in the reuse methodology manual that was jointly developed by Mentor Graphics and Synopsys [4]. These guidelines recommend some specific practices for coding on RT level to produce synthesizable and reusable designs, where timing problems are quickly identified and easily solved.

The soft core deliverables, besides the source files with synthesizable core model, include synthesis scripts, testbenches for the whole core, stimuli files and source files with models of connected to the soft core program memory and data memory.

The microcomputer family model contains the soft core model and the models of the peripheral units. Those units can be either standard (e.g. timer-counter units, serial I/Os) or developed by end-users (e.g. LCD control unit). All of the peripheral unit models are parameterized and configurable.

[1] This work was partly supported by the grant 8T11 B001 12 from the State Committee for Scientific Research, Poland.

3. Virtual component reuse with VirComp

In order to help the soft core user to set the suitable core configuration for his designs, a virtual component configuration tool named VirComp was developed. First, according to the configurability range of the developed soft core, VirComp lets the user choose appropriate parameters of the virtual components that come into his model. Then it checks the chosen parameters for possible errors and reports them to the user. If there is no error found, it translates the information about the parameters into a synthesis script that provides appropriate generics for synthesis tools (e.g. Synopsys Design Compiler) so that they can generate the model as required.

VirComp also gives the core user the possibility of defining configurable assembly languages based on machine instruction sets of his developed microcomputers. Test programs written by the core user or other assembly programs written by end-users can be then analyzed and translated into machine instructions using built in the VirComp assembler. The assembler was design with special emphasis on configurability. The user can modify grammars of assembly languages, for example remove some existing instructions any time as required. Furthermore, structural modification of the microcomputer model may cause the automatically modification of the assembly languages.

4. Hardware implementation

As it was mentioned earlier, the model of the microcomputer family can be set up upon request and easily modified to the need of particular end-users. As an example, the model has been configured in such a way that obtained microcomputers are compatible with the Mitsubishi 4-bit MELPS 720 microcomputer series [3]. Each microcomputer in the original industrial series can be then replaced by a configured model that offers several improvements and a high degree of portability and flexibility.

Testbenches for all models of the microcomputers belonging to the series were designed to generate waveforms for reset, clock, ports, and to model portions of their program memories. The models of the microcomputers have been verified by simulation. After that, we have used the Synopsys VHDL design analyzer and compiler [1] for performing logic synthesis on each model of the series. The target technology is AMS for ASIC implementation and XILINX [2] for FPGA implementation. After synthesis, the microcomputer models belong to the series have been validated by simulation. The obtained results were functionally consistent with the pre-synthesis simulation results. The data memories of all microcomputers were implemented in FPGAs with built-in hard macros.

The logic structure of each microcomputer has been trimmed, partitioned and placed in CLBs, then routed. The configuration data was created and downloaded into the XC4025 device. The correctness of the microcomputer models has been confirmed by using the Tektronix LV500 tester in connection with a developed application board containing the XC4025 device.

5. Conclusions

We have developed a configurable soft core based microcomputer family model using VHDL description at RT level. We have also proposed a configuration tool to help the user to choose appropriate parameters for generating of required microcomputer models, and mentioned about hardware implementation of industrial microcomputer series based on the model.

The microcomputer family model is useful particularly in these automatic control applications, where several input signals need to be handled. In those cases, using of an appropriately configured microcomputer model resulted in significant decrease in the scale of software design complexity.

Actually, the work on the extension of the soft core with regard to its configurability and reusability [9], on the extension of the microcomputer family in model terms of architecture, as well as on the development of a configurable model of microcontroller class is being done.

References

[1] Synopsys: *Online Documentation for ver. 1999.10*, 1999.

[2] Xilinx Inc.: Documentation Foundation Series 3.1i, 2000.

[3] Mitsubishi Semiconductors: *Single-chip 4-bit cmos microcomputers*, 1990.

[4] Bricaud, P. - Keating, M.: *Reuse Methodology Manual for System-on-a-chip Designs*, Kluwer Academic Publishers, 1999.

[5] Seepold, R. – Kunzmann, A.: *Reuse Techniques for VLSI Design*, Kluwer Academic Publishers, 1999.

[6] Nguyen, Q. T. - Rosiński, A. T.: "Modelling of a Sequencer Core - A VHDL Approach", Proc. of the 22nd EUROMICRO Conf. (short contribution), IEEE CS Press, 1997, pp.156-160.

[7] Janiszewski, I. - Baraniecki, R. - Siekierska, K.: "A Reusable Microcontroller Core's Design", Proc. of VIUF'99 Conference, IEEE CS Press, Calif., 1999, pp. 14-19.

[8] Nguyen, Q. T. – Siekierska, K.: "Soft Core Based Model of a 4-bit Microcomputer Family", Proc. of the 3rd DDECS, Smolenice Castle, April 5-7, 2000, pp. 19-25.

[9] Nguyen, Q. T.: "Hardware Design with Emphasis on Synthesis and Configurability", Proc. of the 12th ZMAiP Conf., Warsaw, Oct. 9-10, 2000, pp. 43-48.