JHADES: OPEN DESIGN ENVIRONMENT ON JAVA

Ulisses Chieppe, Giliardo Freitas, Cristiano Biancardi
Ricardo Santos Ferreira (Advisor)
Universidade Federal de Viçosa, Departamento de Ciência da Computação
Viçosa, MG Brasil CEP 36570-000
(uchieppe, gfreitas, cbianc, cacau)@dpi.ufv.br

ABSTRACT

The main goal of this work is to improve the Hades simulator environment, which is an open source logic circuit design tool written in Java, by adding others tools: (a) Misim, microcontroller simulator; (b) external hardware (parallel port); (c) Sis, a logical synthesis tool; (d) and Sashimul, a simulator of on-the-fly microcontroller embedded systems for synthesis in FPGA/VLSI.

1. INTRODUCTION

There are some CAD tools for design digital circuits, however the major part of CAD are commercial and complex tools for a beginner student [1] (Altera [2], Xilinx [3], Cadence [4]) and others tools are quite simple like Diglog [5], TKGate [6], Workbench [7], Digital Works [8], which the capacity to improve is limited.

The Hades (Hamburg Design System) is an open source CAD tool and it can be used such as for introduction or advanced circuit design [9]. The main goal of work is improve Hades adding new resources and by attaching others tools. Figure 1 shows our proposed Hades new environment.
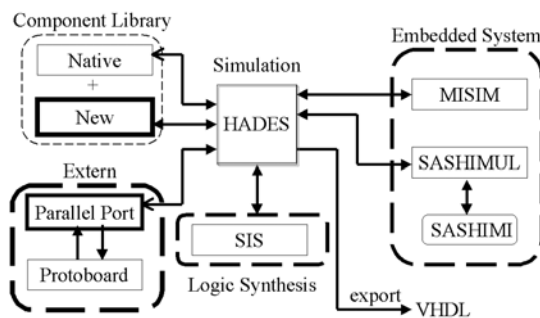

Figure 1: Hades Environment Block Diagram.

This paper is organized as follow: Section 2 introduces Hades which is our main environment. Section 3 gives an overview of new didactics components that improve Hades library. Section 3 also presents the procedure to develop a graphical interface for a new component. Section 4 focus on Sashimi, embedded system tool. Section 5 shows the parallel port component. The logical synthesis features are discussed on Section 6.

2. HADES

Hades [9] is a simulator tool developed in Java. The Hades graphical interface is interactive and includes animation resources. The component definition in most of simulation tools is restricted, because the designer should use a specific internal language to create new components. On the other hand, Hades is open-source and by using Java, a popular high level language, a new component can be developed. A computer science student can quickly write a component behavior in Java than learn VHDL at the beginner level. The main features in Hades are: simulation and design of digital logic circuits, hierarchical design, VHDL export, hardware and software codesign, a large and open source component library, batch processing controlled by Java source or scripts, an object oriented component configuration, structural or behavioral description. Furthermore, Hades have been developed to provide connectivity with others programs. Figure 2 shows the Hades simulation environment.
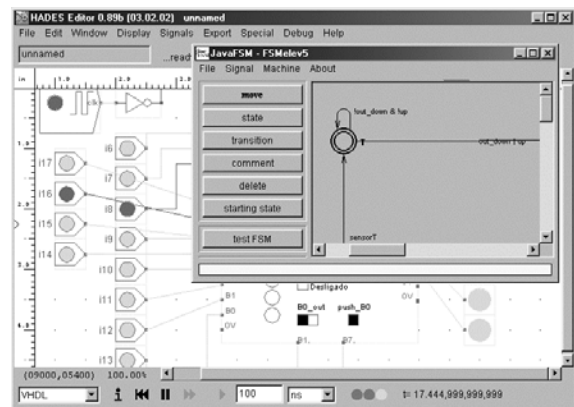

Figure 2: The Hades Graphical Interface

3. NEW COMPONENTS

Hades have a set of pre-defined components and it is also possible to add new components. We have developed some samples: an elevator and a traffic light controller, illustrated in Figure 3. Our main goal is to show how developed a component with an animated graphical interface. These examples are typical in Finite State Machine Teaching. In addition, we have also created components to provide message exchange with others tools, like SIS, Misim and Sashimul. In addition, a special component was developed to

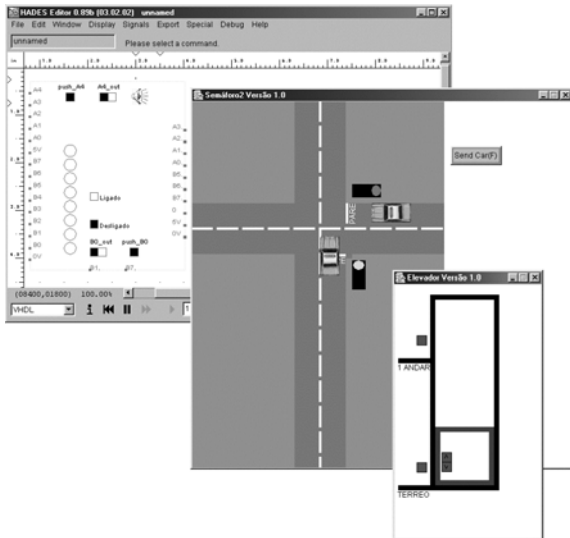connect to the external environment by using the parallel port.

Figure 3: The new components of Hades: an elevator and a traffic light controller.

Although Hades support hierarchical designs, the simulation could be very slow. A component called Modules Generator has created to solve this problem. A beginner student can develop new components quickly and easily. The student should only answer some questions as: bus size and number, number of inputs etc, i.e., the student defines only the external interface. Then, the student have to write a Java function to describe the component behavior.

4. EMBEDDED SYSTEMS

Although microcontrollers (like PIC16F84) is a native library component in Hades, even a small assembler source is very slow to simulate. We implement a communication between Hades and Misim to enhance both tools. Misim is an efficient PIC simulator [10], written in Java. Therefore, Hades was improve by using an attach code simulator and Misim was able to use all Hades component library.

To improve even more the embedded system design in Hades, we have added the Sashimul simulator, which is a simulator of Sashimi [11], developed in Java. Sashimi is synthesis tool for on-the-fly microcontroller embedded system in FPGA/VLSI. Sashimul library is restricted, can not export VHDL code. But Hades library is bigger and can export VHDL.

Sashimi can generate an design from a source code specified in Java [12], and can also compute some performance statistic like: total area in FPGA , required RAM space, etc. The Sashimi major feature is ASIP generation.

5. PARALLEL PORT

We propose to use the parallel port to connect our simulation environment to the real world. We have implement a component parallel port (like another Hades component), to enable the communication between any component and a external circuit build on a protoboard, for instance. Thus, an external led or a display or a complex board can exchange signals with the simulator.

6. LOGIC SYNTHESIS

Hades does not offer any support to logic synthesis. Thus, a new component to allow to use Sis inside Hades environment, has been implemented. Sis is a open source CAD tool, an standard in logic synthesis [13]. The main features are: state assign and minimization, two-level or multi-levels optimization, technologic mapping for standard cell or FPGA, etc.

7. REFERENCES

[1] Giorgio Da Bormida, Domenico Ponta, and Giuliano Donzellini. Methodologies and tools for learning digital electronics. *IEEE Transaction in Education,* 40 (4), 1997.

[2] Altera. Available in http://www.altera.com

[3] Xilinx. Available in http://www.xilinx.com

[4] Cadence. Available in http://www.cadence.com

[5] Gillespie and Lazzaro. Diglog and Analog. Technical Report, Caltech VLSI CAD Tools, California Institute of Technology, 1996.

[6] TKGate. Available in http://www-2.cs.cmu.edu/~hansen/tkgate/index.html

[7] Workbench. Available in http://www.interactiv.com/

[8] Digital Works. Available in http://www.mecanique.co.uk/

[9] Norman Hendrich. Hades – The "Hamburg Design System". In *European Academic Software Award,* 1998.

[10] Feersum Technology. Misim – microchip simulator. Available in http://www.feertech.com/misim

[11] ITO, Sérgio A.; CARRO, Luigi; JACOBI, Ricardo P. System Design Based on Single Language and Single-Chip Java ASIP Microcontroller. In: Design Automation and Test in Europe. Proceedings... Paris, France: IEEE Computer Society Press, 2000. P. 703-707

[12] ITO, Sérgio A.; CARRO, Luigi; JACOBI, Ricardo P. Making Java Work for Microcontroller Applications. IEEE Design & Test of Computer, Los Alamitos, Set-Out 2001.

[13] E. M. Sentovich, K. J. Singh, L. Lavagno, C. Moon, R. Murgai, A. Saldanha, H. Savoj, P. R. Stephan, R. K. Brayton and A. Sangiovanni-Vincentelli. SIS: A system for sequential circuit synthesis. Technical report, U.C. Berkeley, May 1992.