TEACHING UNDERGRADUATES TO DESIGN WITH VHDL

Edval J. P. Santos, Altamiro A. Suzim(*), Alberto C. Mesquita Jr.,
Departamento de Eletrônica e Sistemas, Universidade Federal de Pernambuco
Recife-PE, Brazil
(*)*Departamento de Engenharia Elétrica, Universidade Federal do Rio Grande do Sul*
Porto Alegre - RS, Brazil

Abstract

Traditional circuit designers think in terms of functional blocks, i.e., by dividing the digital system into interacting blocks. On the other hand, freshmen electrical engineering students who have taken an introductory programming language course will think algorithmically (sequentially) to create the design. Hardware description languages can use both formats to describe a circuit: concurrent and sequential. VHDL is an acronym for Very High Speed Integrated Circuit Hardware Description Language, it is an IEEE standard since 1987. Recently, we have introduced VHDL to undergraduate students who have only taken an introductory course on digital logic, and we now report on this experience.

Keywords: VHDL, education, design

1. INTRODUCTION

Electronics is the paradigm of the technological society. It has open up new horizons, from space exploration to new medical equipment, from factory to office automation, from computer networks to telecommunications. This development process has led to the need of complex integrated circuits. Complexity brings as consequence longer design time and a need for larger design staff. This only adds to final cost, and can make the difference between survival and failure in the marketplace. The only way to tackle the design of integrated circuits in a scenario of ever increasing complexity, is by dividing it into smaller blocks and assigning design teams for the development of each one. Of course, the design team must pay attention to appropriate documentation maintenance, task assignment, test generation and scheduling to be able to meet 'Time-To-Market' expectations.

The actual implementation of the design is technology dependent, and a well kept technology database is key to successful designs. Traditional design follows a complex path, as shown in Figure 1. One has to go one step at a time until a finished implementation can be delivered to the client.

Engineering is a mixture of science, technology, creativity, society demands and resources. When teaching young undergraduates to become engineers, a turning point in their education is reached when they finally develop the ability to combine their knowledge base into new designs. This ability is a consequence of being creative and is what makes them engineers. Therefore, design should be taught early in their education.

The design problem

A critical step in the design is the specification phase, once it depends on the client understanding of his needs and the engineer ability to extract all the needed information from the client. In any case, the specification is almost always incomplete and must be checked for inconsistency and for conflict with technological and scientific base. A key phase which needs to be stressed in the Brazilian EE departments, is the design phase itself. In this phase, the designer will combine his creativity with science and technology to come up with a (innovative) solution. A simple model for the design phase is the 'Top-Down' model. In this model the design is divided into smaller blocks, which are subdivided further and further until the basic

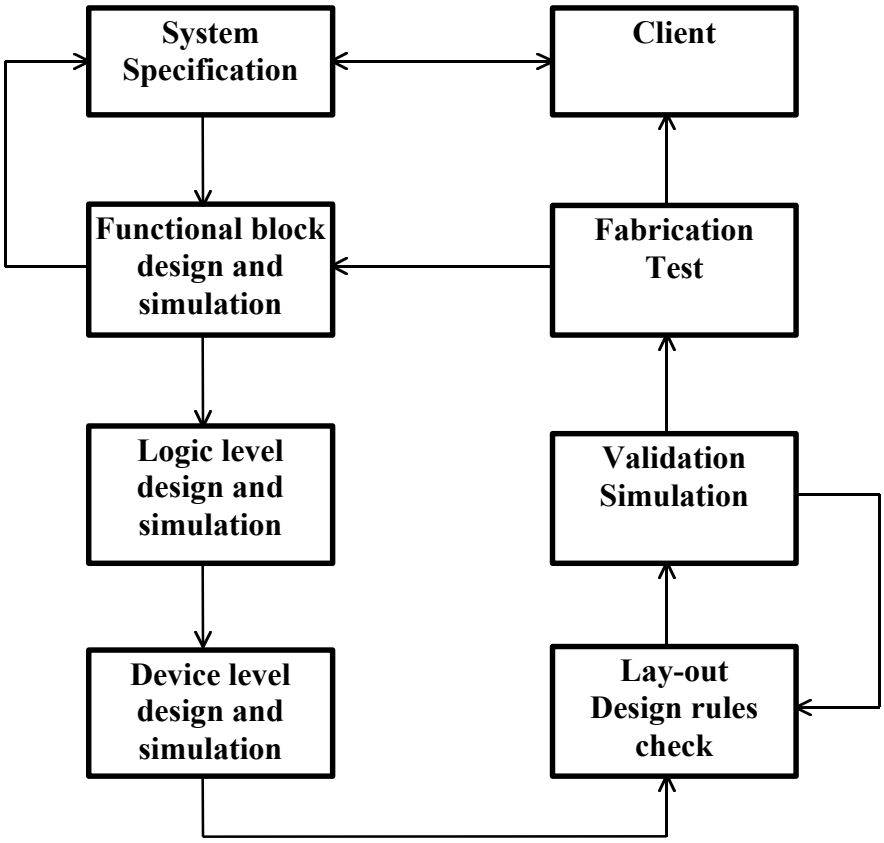

Fig. 1. Traditional design flow.

building blocks of the chosen technology is achieved. In reality a combination of 'Top-Down' and 'Bottom-Up' is used[1]. As microelectronics has evolved different building blocks have been available, such as: off-the-shelf TTL and CMOS circuits of the 70's, PLD's of the 80's, and ASIC's of the

The initial development of Computer Aided Design (CAD) tools were directed at design automation in each phase. The various tools developed would then help to model, verify and simulate the design at each phase. This has lead to a large number of tools, not necessarily being able to communicate among them. Ideally, the design flow should be as presented in Figure 2. Unfortunately, this is still not realized. Although, VHDL is a step in this direction. VHDL is an acronym for Very High Speed Integrated Circuits Hardware Description Language. It became an IEEE standard in 1987 (std 1076-1987[2]). The objectives of VHDL are: to be able to describe complex circuits and to be a popular standard. Its implementation belongs to a class of software known as Computer Aided Engineering (CAE) tools. CAE tools is a development induced by the need to manage the design of complex systems. Complexity requires specialized designers, which may translate into longer 'Time-To-Market', lower production volume, higher costs. Powerful CAE tools allows for the development of sophisticated integrated circuits, and this has had a positive feedback on further development of more powerful tools.

VHDL has become a language for modeling, design and simulation of hardware[3-4]. VHDL is quite wordy and this may lead to longer initial design times, on the other hand this helps to keep designs well documented, which is important to successful designs. It should be noted that debugging becomes time demanding as complexity increases. Beyond that, it is also costly when done later in the design cycle.

Although, VHDL descriptions may look like traditional computer languages, such as: FORTRAN, Pascal or C, it adds a new concept. Traditional languages uses sequential description to code an algorithm.

In VHDL, one may use the concept of concurrency, which VHDL inherited from ADA. The sequential description is quite useful to introduce VHDL to beginners, but the concurrent description is closer to the behaviour of a real circuit. The VHDL description can also use the 'Top-Down' strategy. In this strategy, one may divide the digital system in functional blocks not necessarily independent, and the design can be implemented using the library of cells method. In this method, the software has a database of well tested and documented cells, which is combined to build the design.

Abstraction levels

As shown in Figure 2, the goal of modern CAD tools is to provide the engineer with an interface having the highest level of abstraction. The abstraction levels are classified according to the proximity to the actual physical implementation. The VHDL description is said to have a high level of abstraction. At this level, the circuit is represented by a description of its behaviour. At the other extreme, one has the circuit/device layout with the specification of transistor areas, material choices, dopant concentration, etc. The layout representation has a low abstraction level (See Table 1).

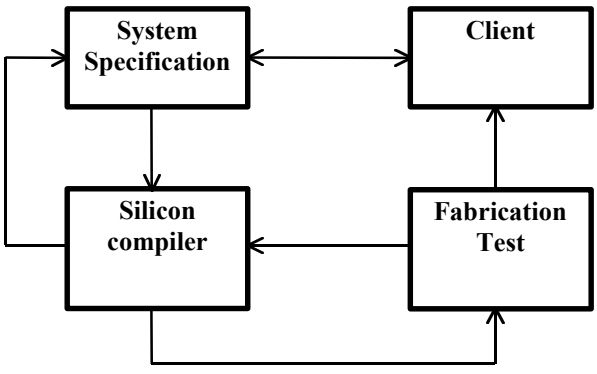

Fig. 2. Ideal design flow.

Table 1. Abstraction levels

Abstraction level	Representation	Primitives
High	Behaviour	Boolean equations, Finite State Machines (FSM)
	Logic	Switches, Inverters, basic gates, latches, etc
	Circuit	MOSFET's, Resistors, Capacitors, Wiring
Low	Layout	Geometry

2. COURSE STRUCTURE

The early introduction of the VHDL language gives the possibility to cover the aspects just mentioned and open up new horizons for the logic designer-to-be. Thus VHDL should be taught as soon as possible in the program. Some of its concepts may be introduced during the first course on logic design. To test this concept, a course syllabus was designed, as shown in Figure 3. It was presented during two weeks in the summer of 1997 and had forty hours of theory and thirty hours of computer exercises. To illustrate the use of VHDL, the students worked on the modeling of a clock/calculator. During the course various aspects about digital systems design are refreshed in parallel with the introduction of various VHDL concepts.

Twelve students were enrolled in the course and only one gave up. This drop out was expected because this student was the only one who had no prior class on digital logic and decided to take the course to check whether he could follow. Off the eleven students who completed the course three continued as junior researchers (scientific initiation) with REENGE (CNPq program) scholarships.

Module 1 - (6 hours)

Basic digital circuits - ideal logic;
Combinational circuits, sequential circuits (synchronous/asynchronous - clock/delay concept);
State machine (design with ROM, PLA and gates).

Module 2 - (4 hours)

Architecture; Operational part (basic elements) and control part (cells and structures).

Module 3 - (4 hours)

The VHDL language;
Alliance system and examples.

Module 4 - (6 hours)

Architectural design of an integrated controller.

Module 5 - (6 hours)

Introduction to CMOS technology;
The MOS transistor and the CMOS inverter;
Basic logic gates, delay time
Stick diagram;
Layout;
Simulation with SPICE.

Module 6 - (4 hours)

Delay time versus gate charge;
Cell design and characterization;
Software AGATA (UFRGS/CTI).

Module 7 - (4 hours)

Synthesis of sequential machines with VHDL;
Communication among sequential machines.

Module 8 - (6 hours)

Implementation/Description/Simulation of the microcontroller defined in module 4.

Course schedule

hour	daily activity									
	Day 1	Day 2	Day 3	Day 4	Day 5	Day 6	Day 7	Day 8	Day 9	Day 10
0-2	M1	M1	M1	M2	M2	M3	M3	M4	M4	M4
2-4	M5	M5	M5	M6	M6	M7	M7	M8	M8	M8
	Lunch									
5-7	Computer exercises, work groups.									

Fig. 3. Course program and schedule.

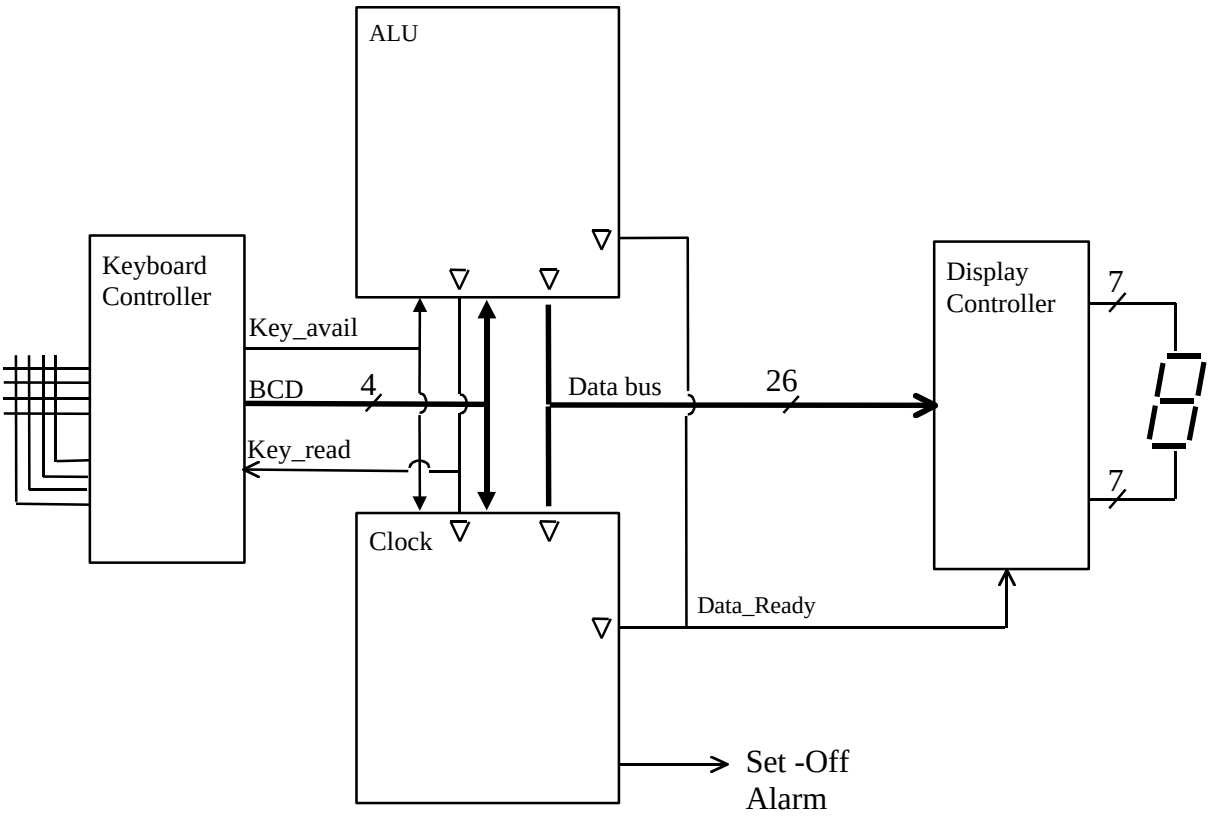

Fig. 4 Architecture of the clock/calculator with keyboard controller, ALU, display controller, and clock unit.

Example architecture

As mentioned above, the digital system used as a modeling example is a system which operates as a clock or a calculator, according to operator selection. The block diagram of the clock/calculator is shown in Figure 4. At turn on, the clock/calculator goes automatically into clock mode, if the calculator mode is desired, a Function key must be pressed. As a clock, it has the following functions: display current time, set alarm, alarm activation and time adjustment. As a calculator, it adds, subtracts, multiplies and divides.

The internal operation of the system is as follows: the clock is always updating the current time, in between time updates, it checks whether the alarm must be set off, whether a time adjustment is being performed or the alarm time is being set. The display shows the time if the system is in clock mode, or the last keyboard sequence, if in the calculator mode. In the calculator mode, the current time is kept in an internal register, but not displayed. The keyboard controller checks whether a key is pressed, decodes it and places the information in the bus. The display unit shows the output of the ALU module or the clock module depending on which mode the system is active. Further details about the design, including VHDL listings, can be found in [5].

4. DISCUSSION/CONCLUSIONS

Off the eleven students taking the summer course, four of them had only an introductory course on digital logic. At the end of the course, one can not see any significant difference between the two groups. After this experience, our proposal is that VHDL should be introduced, as early as possible in the course of study. The introduction of VHDL can be done jointly with the structured description traditionally

presented on introductory digital design courses. In this description, the design is described in terms of logical functions, such as: gates, full-adders, multiplexers, decoders, counters, and registers. Furthermore, other logical and electric aspects of the description, such as: bus, open collector, high impedance, and delay can be dealt with VHDL. The introduction of VHDL can be done in three steps.

In the first step, the various combinational functions would be described with an entity definition and the several forms of architecture descriptions: structured concurrent, behavioral concurrent and behavioral sequential. At this point, the concepts of concurrency, components instantiation, data and objects types, and architecture selection are introduced.

In the second step, the introduction of level synchronous sequential machine can be used to illustrate other VHDL concepts, such as: block statement, guarded block statement, clock, and sensitivity list.

On the last step, the student will learn to design a digital system described in an algorithmic form and specified in some language (even spoken language) directly in VHDL. In this step, new concepts are introduced, covering all modern logic design cycle.

Finally during other courses, such as: Digital Electronics, Introduction to VLSI design, Microcomputer, Computer Organization, Undergraduate Projects, a complex digital system may be considered and VHDL would be used to describe various other concepts, such as: microprogrammed control units, RISC architectures, dataflow architecture, etc.

The objective of this effort is beyond teaching modern design concepts. The idea is to induce creative design through VHDL. Creativity driven design and need driven design should be allowed to flourish early in the engineer education. By creativity driven design, we mean design which creates a market, and need driven design is the kind of design induced by the market. In the Brazilian case, telecommunications, electronic toys, factory and office automation, food industry, agriculture automation, metrology, medical equipment industry, and auto industry are some niches for the application of Brazilian created solutions. The electronics industry moves around the globe billions and billions of dollars, and this is good enough reason to educate electrical engineers. The electronics industry represents high level jobs. This explains how developed countries take care of their semiconductor industry, contrary to what we see in countries under development, namely Brazil. It should not be enough to become a supplier of unprocessed goods.

ACKNOWLEDGEMENTS

This work is supported by the REENGE program and UFPE. We would like to thank all students who participated in the I Internal Workshop of Electronics (I Oficina de Eletrônica) of the Department of Electronics and Systems at Universidade Federal de Pernambuco (UFPE). We also thank the Graduate and Undergraduate Office of the Department of Electronics and Systems for their support.

REFERENCES

[1] E. Hörbst, C. Müller-Schloer, and H. Schwärtzel "Design of VLSI Circuits", Springer-Verlag .

[2] IEEE Standard VHDL Language Reference Manual, IEEE New York NY (1988).

[3] Z. Navabi, "Using VHDL for modeling and design of processing units", proceedings of the fifth annual IEEE International ASIC Conference and Exhibit, pp315-326 (1992).

[4] J. S. Aude, "Design of the NCESPARC control unit using the Alliance system" proceedings of the I Ibero-American Microelectronics Conference, pp319-328 (1995).

[5] A. Suzim, Alberto C. Mesquita, Luiz G. C. Silva Jr., Eduardo H. Lima, Victor M. da Silva, Edval J. P. Santos, "Using VHDL to model concurrent FSMs" Paper submitted to SBCCI'97, August 25-27, 1997, Gramado, RS - Brazil