

Efficient Production of Computer-Based Training in Chip Design

Tamer Çatalkaya, Ulrich Golze

Dept. of Integrated Circuit Design, Technical University of Braunschweig
Gaußstr. 11, D-38106 Braunschweig, Germany
catalkay@eis.cs.tu-bs.de, <http://www.cs.tu-bs.de/eis/english>

Abstract

Chip and system design can be made more efficient and attractive by using multimedia training techniques. In our case, these techniques are employed as a supplement of classical teaching. We point out the need for the development of own computer based training material (CBT). To reduce the high expenses of multimedia productions, we propose a universal Reference Model as a framework for CBTs in many technical areas. We demonstrate the applicability of the Reference Model to a CBT system "VeriBox", which introduces the hardware description language VERILOG and the associated logic synthesis.

1 Introduction

CAD tools for designing integrated circuits and systems are quite complex. Their introduction requires a significant training effort. Due to tool complexity and licence restrictions, training is often limited to the design lab. Access time to these tools may also be restricted. Therefore, additional learning programs are desirable which can also be used at home or at other places.

We examined several commercial CBTs on chip design, e.g., Doulos [1], Xilinx [2], and others. Unfortunately, they often do not cover all of our topics or do not go into the details necessary. Moreover, they cannot be adapted easily, when the design software changes. As with the CAD design tools, license restrictions may prevent their use outside the lab.

Experiences with self-developed CBTs have shown, that multimedia techniques can often better explain complex or dynamic situations. A major advantage is the presentation of a problem by visualization, motion, interactivity, sound, and spoken explanations. An example is given in Section 3. Furthermore we observed, that our students were more motivated by multimedia education.

It is well known that the production of high-quality CBTs is quite expensive and time consuming. As an order of magnitude, \$50,000 per multimedia hour are a realistic estimate [7]. This amount increases rapidly, when additional functionality such as user-adaptation or

complicated interactions are added. Moreover, management, support, and updating of CBTs require extensive work. Finally, not only technical know-how is required in CBT production, but also an understanding of creative design and didactical aspects.

To reduce and solve these problems, and to make CBT production affordable by ordinary technical institutions, we have developed the Reference Model of the subsequent section.

2 A Universal Reference Model for CBTs

An average university budget can often not afford the costs mentioned above. We therefore introduce the principle of *Low-Cost/Low-Time* production. By this we mean that the development time of a CBT is only exceeded by a factor of two to four as compared to the effort of producing conventional material.

In developing several CBTs since 1996 we observed that many components and features can be reused such as navigation, user interface design, user management, and a common look and feel. This resulted in a general Reference Model.

This Reference Model [5] is given as a source code application of the authoring system Authorware of Macromedia. It is a sample structure providing a base for the look and feel of the CBTs deduced. In addition to navigation, user interface and user management, it provides functionality such as automatically generated lists of contents, management of bookmarks, and a history list. By using setup options, any CBT based on the Reference Model can be adapted to special user needs. For every particular CBT to be developed, the Reference Model is copied and extended by the application contents. Thus the CBT designer saves time and is relieved from tedious work.

CBT applications developed with our reference model (called *modules*) can be composed to form larger systems. Inter-module navigation is automatically provided in the tables of contents.

The Reference Model also includes quite a few guidelines on visual and audio presentation such as style guides, recommendations on fonts, font size, contrast,

audio recording, and much more. This ensures a unique and efficient look and feel of all CBTs produced.

In addition, there is also a Reference Model for Dictionaries. We recognized that there is often a need to supply index like dictionary information. Of course, common functions such as searching are provided. With the tool Knowledge-Display for Authorware, a dictionary can be generated as easily as with a common editor.

An important factor in the Low-Cost/Low-Time concept is an appropriate authoring tool for developing CBTs. We analyzed several high level authoring tools, and Macromedia Authorware 5.2 [6] turned out to be best suited for our needs.

By using license-free sound and picture libraries, we save time and money for developing extra CBT assets.

The Reference Model based CBTs are developed on standard PC systems and do not have special requirements except for the sound hardware. To get an acceptable audio quality, we use a high-standard sound card.

3 Example Application VeriBox

The use of a hardware description language (HDL) in chip and system design needs as little explanation as the importance of logic synthesis tools. At our institute, we use one of the two major HDLs, namely VERILOG [3,4] and the synthesis system of Synopsys. HDL and synthesis are taught more theoretically in lectures and practically in lab exercises. In both cases, the VeriBox acts as a support, which makes learning more attractive and more efficient as well as more motivating. Figure 1 shows a page where delays in VERILOG are explained. At the same time, the VeriBox serves as a test vehicle to evaluate the feasibility of the Reference Model.

The VeriBox consists of four Modules and a dictionary: *Introduction to Verilog*, *Fundamental Concepts*, *Examples and Tasks*, *Logic Synthesis*, and the *Dictionary*.

As in real hardware design, logic synthesis is a main part of the VeriBox. Subtleties of the synthesis input may highly influence the resulting design quality. The designer should learn that in spite of automatic tools, he still has the responsibility for producing high quality results. This is also achieved by the synthesis input style.

By using multimedia techniques, we can explain complex behavior in logic synthesis. As just one example, we mention the concept of *temporal locality* of VERILOG registers. This determines whether synthesis produces a flip-flop, a latch, or combinational logic out of a VERILOG register. We first define a *potential register* as a VERILOG register, which gets an assignment within an always-block. A potential register is defined as

spatially local, if it is only used within a single always-block. Finally, a spatially local register is *temporally local*, if its present value is not referenced in following passes of the always-block. While the definition of potential and spatially local register is relatively easy to illustrate, temporally local registers require dynamic demonstration which are best presented using motion.

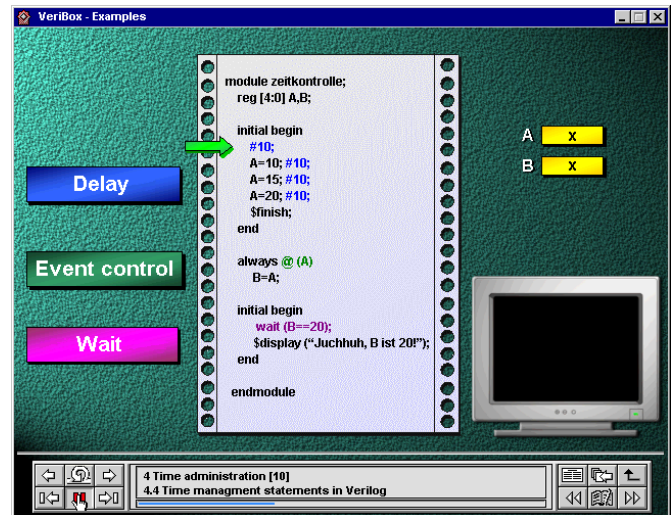


Figure 1: Screen shot of the VeriBox

The module *Logic Synthesis* of the VeriBox is divided into four sections. In the first section, the relevance of automatic logic synthesis is indicated for motivation. In the second section, VERILOG keywords being recognized by most synthesis tools are introduced in conjunction with VERILOG register transfer models. The third section explains and demonstrates the hardware resulting from VERILOG models. All examples are generated by the real synthesis tool, and the user can modify them for evaluation. The fourth and last section describes the subtleties of always-blocks w.r.t. synthesis of latches, flip flops, and combinational logic.

References

- [1] CD-ROM: HDL Pacemaker, Hampshire UK, 1997.
- [2] CD-ROM: Xilinx Verilog CBT, San Jose USA, 1998.
- [3] Coffman, K., Real-World FPGA Design with Verilog, Prentice Hall, 2000.
- [4] Golze, U., VLSI Chip Design with the Hardware Description Language VERILOG, Springer, 1996.
- [5] Reference Model, Abt. E.I.S., TU Braunschweig, 2000.
- [6] Schiffman, R., Authorware Attain Praxis, Springer, 1999.
- [7] Schreiber, A., CBT-Anwendungen professionell entwickeln, Springer, 1998.
- [8] Steinmetz, R., Multimedia-Technologie, Springer, 1999.