Teaching of System Level Design and Verification: Methodology & Tools

Benjamin Abramov, Ilya Levin
{avramov, ilia1}@post.tau.ac.il
Tel Aviv University
Ramat Aviv, Tel Aviv 69978, Israel

ABSTRACT

This paper discuses a new approach for teaching of system level digital design and verification by using Split Algorithmic State Machines. The proposed approach enables parallel representation of complex digital systems. At the same time, it provides verification of the system at the early stage of the design by using a spreadsheet-based simulation.
Keywords – Algorithmic State Machine, Split ASM, Verification, Spreadsheets Simulation.

1. INTRODUCTION

The growing complexity of systems to be designed leads to a conceptual gap between the system level of design and the implementation level of design. Namely, the human-machine interaction on the system level of design becomes more and more "human-oriented", while the implementation level of design becomes more and more remote from the system level and thus complex to be managed. In general, such a gap characterizes the technology progress in the modern society. However, the common sense and the basic technological orientation being successful in the every-day life, is not always applicable for developing digital systems and, especially, for teaching logic design. Distancing the user from a design automation system by means of an intelligent interface leads to a serious misunderstanding of the system's functioning and, as a result, may lead to negative consequences. Our experience in teaching digital design by using a number of design automation tools formed quite a negative opinion about efficiency of these tools in teaching. We have come to a conclusion that the majority of the modern powerful design automation tools are inefficient in teaching digital design due to their "human" orientation (and not "implementation" orientation).

The present paper describes a novel approach for teaching principles of digital design. The proposed approach is based on two main ideas: a) defining the system, by a newly introduced Split Algorithmic State Machine (ASM), and b) verification of logic design by a spreadsheet simulation learning environment.

Modern control-dominated designs include complex communication protocols described in a form of interacting Finite State Machines (FSMs). The most popular forms for FSM representations are: ASM chart and a State Transition Graph (STG). These two forms of FSM representation may be considered [1] as two different design paradigms: a programming (ASM chart) and a design (STG) paradigms. In this paper, each of the paradigms is considered on three conceptual levels: a system level, a formal model level, and an implementation level. Being very convenient form for system specification, the ASM [2], at the same time, has a significant cognitive contradiction. On the one hand, the ASM corresponds to a classical von Neumann's architecture that is based on the sequential, step-by-step performing of an algorithm. On the other hand, on the implementation level, the behavior of the ASM is concurrent. Furthermore, the ASM description includes a significant redundancy, associated with its tree-like nature. One of the main requirements for a modern systems' specification is its ability to support the concurrency. To overcome the above ASM disadvantages we introduce a novel type of a system specification – a so-called *Split ASM*. Being an alternative to the conventional "monolithic" ASM system description of the system, the Split ASM supports concurrency by describing the system as a set of component concurrent ASMs.

The proposed approach is intended to close the gap between the human-oriented system level specification and the level of hardware implementation of the system.

Another problem addressed in our work is a problem of the design verification. We propose using commercial spreadsheet software for simulation and further verification of the ASM specifications. Spreadsheets are available for any user; they are simple, flexible and highly efficient simulation tool. Construction of the spreadsheet simulations does not require any programming effort [3, 4]. The spreadsheet tool enables simulation of the system specification in its natural form. As a result, the initial phase of the design cycle becomes debugable and transparent. The proposed spreadsheet simulation provides rapid system prototyping in a very simple and widely available form. Actually, using the Split ASM specification together with the spreadsheet simulation of the specification opens a new way both for the system description and verification. We provide results of using the described approach in undergraduate classes. These results are based on our experience of teaching Introductory Logic Design in the School of Engineering of the Bar Ilan University.

2. CONTROL UNIT DESCRIPTION BY ASM

Control unit (CU) design is the main problem in the design of digital systems [2, 5]. This paper deals with designing of a control unit modeled as a sequential circuit. A design process transforms a set of specifications into an implementation of the specifications. Both the specification and the implementation are forms of description of the system functionality, but they have different levels of tangibility/abstraction. We use the ASM specification for description of the CU.

The ASM chart is a directed connected graph comprising an initial vertex, a final vertex, a finite set of operator vertices, and conditional vertices. Each conditional vertex contains a single logical condition from a set of input variables of the CU. Each operator vertex contains a specific output vector from the set of output signals of the CU. A simple procedure allows transforming any ASM into the corresponding FSM by associating ASM's paths between marks corresponding to the FSM states [2]. The ASM specification is quite logical and testable; however, at the same time, the designer has to keep in mind and to refer to all possible logical paths (situations) taking place within the system. On the one hand, this strict requirement renders the ASM specification logical and testable. On the other hand, it puts designers into a position where they have to define very sophisticated and even artificial/unexpected states of the system. When the number of variables grows, the process of defining the ASM specification becomes a difficult and unpractical task. In such cases, using our Split ASM's seems preferable.

We define the Split ASM by presenting it as several conventional ASMs as follows:

1. Parallel connection: connecting roots of two or more components ASMs.
2. Sequential connection: replacing one terminal node of an ASM with another ASM.

The Split ASM specification is therefore much more natural. It corresponds to human cognitive patterns to present a complex entity as an assembly of simpler components.

3. THE SPREADSHEET BASED VERIFICATION

The use of spreadsheets for control unit simulation can be best achieved by using a matrix model of a logical simulator proposed in [6] and developed in [7]. Description of the CU in the form of sum-of-products (SOP) may be obtained formally from the ASM chart [2]. The SOP may be presented and implemented in the form of a two level AND-OR structure. According to [6] the matrix spreadsheet model includes two interacting worksheets named SOP and PLA. The SOP worksheet represents a system of logic functions to be implemented, and the PLA worksheet implementing the functioning of the system. The SOP worksheet is flexible and can be changed by a student according to the specific system of logic functions. The PLA worksheet is a fixed homogeneous structure, which implements the system. Each cell of the worksheet is programmed as a specific conditional spreadsheet function. The whole SOP-PLA couple generates output values of the required system immediately after changing the input values or the content of the SOP worksheet [6].

The matrix spreadsheet environment is perfect for the Split ASM simulation. The students transform their ASMs into the SOP form and run the system within the spreadsheet. They are able to debug the ASM in a very simple style without using expensive tools.

4. EXPERIMENTS

We defined a number of CU examples the conventional ASM and the proposed Split ASM form; the resulting CU characteristics were compared. Since operators of the conventional ASM and the Split ASM are different, the common number of possible output vectors in the Split ASM specification is greater than in the conventional ASM. However, the Split ASM has a shorter Average Path Length than the corresponding conventional ASM. Also, in the terms of hardware overhead, our proposed method is preferable than the conventional. In most design cases, using the Split ASM approach provides about 30% reduction in terms of a number of gates of the resulting system.

Interviewing students after their experience in both techniques enables us to evaluate pedagogical effects of the proposed approach. We interviewed a group of 23 undergraduate EE students. The main conclusion is that the majority of students (14 versus 9) preferred the Split ASM both from the point of simplicity of the initial system representation and from the point of efficiency of the obtained hardware solutions.

5. CONCLUSION

Two key advantages of having the proposed unified model for describing control systems are: the ability to divide a system description into independent concurrent blocks, and to explore performance trade-offs and the ability to analyze and verify a complete system.

The paper proposed and studied a non-traditional approach for high-level specification of control unit. The proposed approach allows describing branches of the control unit independently by means of a newly introduced Split ASM data structure. The effectiveness of the approach was examined on a number of design tasks. Using a new data structure - the Split ASM provides a compact and readable representation of the control system specification. The Split ASM specification of a control unit is more compact than its conventional ASM equivalent in terms of the number of nodes and paths. The new approach gives a more adequate idea about the digital system's behavior. The Split ASM specification is preferable in teaching logic design than the conventional ASM since allows describing it separately and avoid necessity to list all possible situations corresponding to paths of the conventional ASM chart.

For the behavioral simulation and verification of the Split ASM, the spreadsheet-based environment was studied. By using the spreadsheet simulation, students were able to simulate and to debug the systems in a very effective manner.

By using the proposed unified method, both the educators and the students are exempted the necessity to build huge and complex conventional ASM specifications. In addition to that, the proposed spreadsheet-based verification successfully replacing very expansive specialized design automation tools.

REFERENCES

[1] Levin, I., Mioduser, D. (1996). A Multiple-Constructs Framework for Teaching Control Concepts, *IEEE Transactions of Education, 39(4), 488-496.*

[2] Baranov, S. (1994) *Synthesis for Control Automata*, Kluwer Academic Publisher.

[3] Levin, I. (1994). Behavioral Simulation of an Arithmetic Unit using the Spreadsheet. *International Journal of Electrical Engineering Education, 31, 334-341.*

[4] Smith, R. E. A spreadsheet-based simulation of CPU instruction execution. *Proceedings of the 2007 ASEE Annual Conference, Honolulu, HI, June 2007, 1699-1709.*

[5] Baranov, S. (2008). *Logic and System Design of Digital Systems*. TUT press.

[6] Levin, I. (1993). Matrix Model of Logical Simulator within Spreadsheet, *International Journal of Electrical Engineering Education, Vol.30, 3, pp.216-223.*

[7] Levin, I., Talis, V. (2004). Using Spreadsheets for Teaching Principles of On-line Checking Logic Circuits. *Spreadsheets in Education, 1 (3), 131-141.*