9 PeNLogic – System for Concurrent Logic Controllers Design

Marek Węgrzyn and Agnieszka Węgrzyn

University of Zielona Góra, Institute of Computer Engineering and Electronics, ul. Podgórna 50, 65-246 Zielona Góra, Poland

Abstract. In the paper the CAD system dedicated for modeling, verification, and synthesis of concurrent controllers modeled by interpreted Petri net is presented. Petri net model can be prepared as graph or as textual form. Controllers specified by Petri nets can be analyzed and implemented using method suitable for such model. For verification of Petri net another part of system is used. Moreover, the results of verification are decomposition of net into several communicating state machines (as finite state machines, FSMs). After verification it is possible to transform Petri net model into HDLs model (VHDL and Verilog) and alternatively into EDIF or XNF netlist format. Such prepared models are also simulated and synthesized using other academic or commercial CAD systems. The system has been developing at University of Zielona Góra. Development of new methods of modeling, verification and synthesis has been contributed to make an attempt the new integrated version of the system. In addition, using of Java environment gives opportunity to work out the system that is platform independent.

9.1 Introduction

The specific application often dictates the system design requirements, such as modularity and flexibility. In general, the design procedure involves the integration of analytical and graphical descriptions. Graphical descriptions, such as control-interpreted Petri nets [11], SFC (Sequential Function Chart) [8] and Grafcet [7], provide established techniques for proper system designs. They have helped industrial engineers to understand the system behavior and performance over many years.

A behavioral representation describes the system’s functionality independently of its implementation. It treats a system as a black box, and defines how the black box responds to any combination of input values. The process of designing a system proceeds from a behavioral specification (SFC diagram or Petri net) to a programmable logic implementation (FPGA, Field Programmable Gate Array). A design in purely behavioral form, like a Petri net, is converted into an intermediate rule-based description, void of any technology-specific implementation details. The final FPGA implementation is generated by automatic synthesis using CAD

tools, instead of manual, tedious design process. The textual format serves as a bridge with some related university tools.

In the paper the integrated environment for design of concurrent logic controllers, called PeNLogic, is presented.

9.2 PeNLogic System

The PeNLogic system has been developing for many years. The first version as a set of different application was presented in [4,15]. However, development of new methods of modeling, verification and synthesis has been contributed [2] to make an attempt the new integrated version of the system. In addition, using of Java environment gives opportunity to work out the system that is platform independent.

The core of the PeNLogic system is Petri net models of concurrent logic controllers. Petri net can be prepared as graph or as textual form. Controllers specified by Petri nets can be analyzed and implemented using method suitable for Petri nets [1]. For verification of Petri net another part of system is used. This part bases on formal method of Petri net analysis, which is widely described in [13]. Moreover, the results of verification are decomposition of net into several communicating state machines (as finite state machines, FSMs) [19,21]. After verification it is possible to transform Petri net model into HDLs model (VHDL [12,22] and Verilog [10,18]) and alternatively into EDIF or XNF netlist format [17]. Such prepared models are also simulated and synthesized using other academic [3,4,20] or commercial CAD systems.

Fig. 9.1 shows a general schema of the PeNLogic system.

9.2.1 Petri Net Modeling of Concurrent Controllers

As opposed to sequential automaton, concurrent automaton can be in one or more internal state at the same time. Maximal sets of concurrently occurring local states are defined by global state of automaton. Any subset of concurrent local states is called partial state. In concurrent automata local relation are considered that relates internal partial states (current and next) and suitable input and output states of automata. Interpreted Petri net is one of the forms for representing of concurrent automaton.

On the other hand, Petri nets as a graphical tool provides a unified design methodology for specifying discrete-event systems. They can be applied in various stages of the design implementation from hierarchical system description to its physical realization. A Petri net is used as a tool for the modeling and analysis of digital circuits, especially concurrent controllers [1,3,5,12,13].

In PeNLogic system it is possible to specify Petri net in textual formats [16]: PNSF2, PNSF3 and CCPNML format and as a graph (figure). For preparing of graphical form of hierarchical, colored, interpreted Petri net, Petri net Editor was implemented. Using this part of the system there is generated from a net graph into PNSF2, PNSF3 and CCPNML formats. Moreover, a net graph and each type of format is stored in database using relational model.