

Algebraic model for the JK flip-flop behaviour

Anca Vasilescu

Department of Computer Science
Transilvania University of Brasov, Romania
vasilex@unitbv.ro

Abstract. The aim of this paper is to use the algebraic theory of processes as a formal method applied for modelling the behaviour of a specific class of systems, namely *Sequential Logic Circuits* (SLC). This approach becomes a connection between the theory of processes and the computer architecture domain. We specify algebraic models, based on Milner's SCCS for the most representative SLC in the computer architecture, namely the JK flip-flop - the elementary structure of memory. Such algebraic models for the basic components of memory have to be regarded as a starting point both for algebraic modelling of other internal components of computers, and for modelling the behaviour of a computer system itself. The main advantage of having these algebra-based models is the capability of automatically verifying those models. For the interest of this paper, the CWB-NC platform is used for verifying the equivalences of models.

1 Introduction

In the large context of the theory of concurrent communicating processes, the global term *process algebra* refers to the study of concurrent communicating processes in an algebraic framework. In order to treat this concurrency theory in an algebraic way, we have to consider concrete structures, namely *process algebras*, equipped with several operators and axioms. For the scope of this paper, we will use the Milner's process algebra SCCS, a synchronous calculus derived from CCS (*Calculus of Communicating Systems*) [6], [7].

Process algebra can be seen as a worthy descendant of classical automata theory. The crucial difference is that in an algebraic-axiomatic approach like process algebra one is interested not merely in the execution traces or language of one automaton, but in the behaviour of systems of communicating automata. We will later combine these two approaches, process algebra and automata theory, when we use a CWB (*Concurrency Work-Bench*) platform [11] for automatic verification of models equivalences.

Using together SCCS and CWB we have many advantages, such as: CWB recognizes the SCCS specification files, CWB can simulate the behaviour of the system specified in SCCS and, moreover, the CWB platform can automatically verify many types of equivalences between models, including bisimilarity as the most appropriate equivalence between SCCS specifications of the target system behaviour.

