*ISSN 0005-1179, Automation and Remote Control, 2013, Vol. 74, No. 1, pp. 113–122. © Pleiades Publishing, Ltd., 2013.*
*Original Russian Text © L.A. Zolotorevich, 2013, published in Avtomatika i Telemekhanika, 2013, No. 1, pp. 146–158.*

# Project Verification and Construction of Superchip Tests at the RTL Level

**L. A. Zolotorevich**

*Belarus State University, Minsk, Belarus*

Received January 24, 2012

**Abstract**—Methods were proposed for project verification and directed design of the superchip tests represented in VHDL at the RTL level. The problem of test design and project verification was solved on the basis of the CNF-satisfiability of some system of Boolean functions.

**DOI:** 10.1134/S0005117913010104

## 1. THE PROBLEM OF PROJECT VERIFICATION AND TEST DESIGN: STATE-OF-THE-ART

Within the entire spectrum of the superchip design problems, those of project verification and design of tests for complex electronic systems remain the most science-intensive ones [1]. The modern integrated circuits contain about billions of transistors on a chip, and the design of tests for the objects of such size at the level of structural representation proved to be an intractable problem. At the same time, an acute need for tests arises at all stages of the life cycle beginning from the start of design because the formal verification methods are developed insufficiently, and in practice the project verification is based on modeling. The tests are also required at the stages of manufacture to screen the ready products and operation to estimate the object operability.

It deserves noting that the rapidly progressing electronic industry presents new requirements on and conditions for the test design. If at the end of the last century consideration was given to the problem of efficient construction of tests at the level of the given structure of an object, today it is complemented by a formulation of the problem of test design for the systems represented in various identification systems whose solution is sought.

An approach to the problem of design of superchip tests was enabled by the object identification at the initial stages of design where there exists some behavioral description or an object description at the level of inter-register transfers which has much smaller number of primitives than at the structural level. The questions of superchip test design in the absence of data about the structural realization of the object are answered mostly through malfunction modeling and random construction [2–4]. A procedure for construction of functional faults reasonably corresponding to the faults of the structural realization of the corresponding mechanism is proposed in [5]. There are studies considering the problems of directed test design at the upper levels of design [3, 4].

A general approach to the hierarchical generation of the superchip tests at the RTL level is presented in [5]. It is based on the fact that each operation of the program code describing the object at the RTL level is realized at the structural level by some set of hardware facilities. Their test can be designed by certain methods and means at the level of structural representation of the device. The test is integrated in the object description, and the constraints on object operation are set. The problem of designing the test for the entire object comes to solving a system of arithmetic equations with the introduced constraints.

The present paper describes a mechanism to solve the problem of directed test design on the basis on a system of arithmetic equations and iterative solution of the problem of CNF-satisfiability of some system of Boolean functions. The object structure in VHDL is presented in Section 2. The next section considers the problem of test design on the basis of a system of arithmetic equations, and Section 4 extends this method to the problem of verification. The problem of test design is related directly with that of verification of the projects of complex-function digital systems. For the time being, the problem of project verification at different stages of design is solved mostly by the object modeling because the methods of formal verification are oriented to some particular problems. Completeness of project verification is ensured by the completeness of the tests used for modeling. The method proposed below is oriented both to the solution of the problem of directed test design and to project verification by modeling at the given test.

## 2. STRUCTURE OF THE OBJECT REPRESENTATION IN VHDL

The digital system is traditionally considered at the RTL level as two subsystems, the operational subsystem transforming data in compliance with the given algorithms and the control subsystem controlling the operational one. Therefore, we make use of the data flow and control flow graphs as the mathematical platform for description of the digital system. The control flow graph is a directed graph with nodes corresponding to the operators of the program code and edges showing the order of operator execution.

The control flow graph is a directed graph with nodes corresponding to the executed operations and edges indicating the sequence of operations for transformation of some variable or signal. The control flow graph and the data flow graphs are designed in the course of static analysis of the program code. The representation of the object description as the graphs of data and control flows is commonly used in the compiler design. These graphs represent in essence the internal structure of the compiler data carrying the results of the program text analysis. In the existing commercial VHDL compilers, the internal representation of the object is, unfortunately, inaccessible to the external users. The majority of works on the design of the data and control flow graphs relies on the manually constructed models. A realization of the idea of formal graph construction as applied to the C-specification was proposed in [6]. In what follows, we present the structure of internal representation of the RTL description for realization of the method of directed test design.

Figure 1 depicts a general structure of the decision diagram (DD) of the control flow which is a directed acyclic graph with the VHDL operators as vertices. The directed edges correspond to the transfer of control from one vertex to another. Each program code operator is repre-

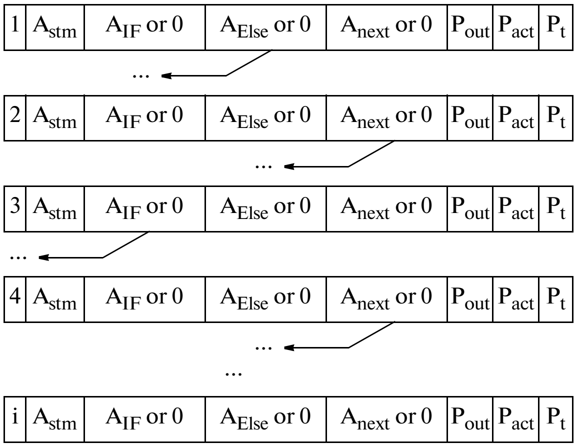
1 $A_{stm}$ $A_{IF}$ or 0 $A_{Else}$ or 0 $A_{next}$ or 0 $P_{out}$ $P_{act}$ $P_t$
...
2 $A_{stm}$ $A_{IF}$ or 0 $A_{Else}$ or 0 $A_{next}$ or 0 $P_{out}$ $P_{act}$ $P_t$
...
3 $A_{stm}$ $A_{IF}$ or 0 $A_{Else}$ or 0 $A_{next}$ or 0 $P_{out}$ $P_{act}$ $P_t$
...
4 $A_{stm}$ $A_{IF}$ or 0 $A_{Else}$ or 0 $A_{next}$ or 0 $P_{out}$ $P_{act}$ $P_t$
...
...
i $A_{stm}$ $A_{IF}$ or 0 $A_{Else}$ or 0 $A_{next}$ or 0 $P_{out}$ $P_{act}$ $P_t$

**Fig. 1.** Structure of the control flow DD.