MINISTRY OF EDUCATION AND SCIENCE OF UKRAINE
KHARKOV NATIONAL UNIVERSITY OF RADIOELECTRONICS

ISBN 966-659-113-8

Proceedings of IEEE
East-West Design & Test Workshop
(EWDTW’06)

Copyright © 2006 by The Institute of Electrical and Electronics Engineers, Inc.

Sochi, Russia, September 15 – 19, 2006
CONTENTS

A Black-Box-Oriented Test Methodology
A. Benso, A. Bosio, P. Prinetto, A. Savino ................................................................. 11

Design and Optimization of Fault-Tolerant Distributed Real-Time Systems
Peng Z., Izosimov V., Eles P., Pop P ............................................................. 16

Interconnect Yield Improvement for Networks on Chip
Andre Ivanov .............................................................................................................. 22

The Scaling Semiconductor World and Test Technology
Yervant Zorian ........................................................................................................ 22

A Unified HW/SW Interface Model to Remove Discontinuities Between HW and SW Design
A. Jerraya ................................................................................................................. 23

Background Cache for Improving Memory Fault Tolerance
Michail F. Karavay, Vladimir V. Sinelnikov .............................................................. 24

Factors in High-Speed Wireless Data Networking – New Ideas and a New Perspective
Daniel Foty ........................................................................................................... 29

Hierarchical Silicon Aware Test and Repair IP: Development and Integration Flow Reducing Time to Market for Systems on Chip
Samvel Shoukourian, Yervant Zorian ............................................................... 39

The Pivotal Role of Performance Management in IC Design
Eyck Jentzsch ................................................................................................... 41

TEST METHODS AND TOOLS

Analysis of a Test Method for Delay Faults in NoC Interconnects
Tomas Bengtsson, Artur Jutman, Shashi Kumar, Raimund Ubar, Zebo Peng ...................... 42

Unified Framework for Logic Diagnosis
A. Rousset, P. Girard, S. Pravossoudovitch, C. Landrault, A. Virazel .................................. 47

Hierarchical Systems Testing based on Boundary Scan Technologies
Hahanov V., Yeliseev V., Hahanova A., Melnik D .......................................................... 53

Testing the Hardware Implementation of a Distributed Clock Generation Algorithm for SoCs
A. Steininger, T. Handl, G. Fuchs, F. Zangerl ............................................................... 59

Extended Boundary Scan Test Using Hybrid Test Vectors
Jan Heiber ............................................................................................................. 65

A March Test for Full Diagnosis of All Simple Static Faults in Random Access Memories
G. Harutunyan, Valery A. Vardanian ..................................................................... 68

Efficient Implementation of Physical Addressing for Testing and Diagnosis of Embedded SRAMs for Fault Coverage Improvement
K. Aleksanyan, Valery A. Vardanian ...................................................................... 72

High Level Models Based Functional Testing of Pipelined Processors
Victor Belkin, Sergey Sharshunov ...................................................................... 76

IEEE EWDTSW, Sochi, September 15-19, 2006 5
On Complexity of Checking of Cryptosystems
Volodymyr G. Skobelev ............................................................................................................................ 82

Distributed Fault Simulation and Genetic Test Generation of Digital Circuits
Skobtsov Y.A., El-Khatib A.I., Ivanov D.E ............................................................................................... 89

Hierarchical Evolutionary Approach to Test Generation
Skobtsov V.Y. Skobtsov Y.A .................................................................................................................... 95

VERIFICATION

Incremental ABV for TLtoRTL Design Refinement
Nicola Bombieri, Franco Fummi, Graziano Pravadelli ........................................................................ 100

RTL Compiler Templates Verification: Approach to Automation
Lev Danielyan, Sergey Hakobyan ........................................................................................................... 108

Verification of Implementation of Parallel Automata (Symbolic Approach)
Andrei Karatkevich .................................................................................................................................. 112

SystemCFL: An Infrastructure for a TLM Formal Verification Proposal (with an overview on a tool set for practical formal verification of SystemC descriptions)
K.L. Man, Andrea Fedeli, Michele Mercaldi, M.P. Schellekens ................................................................ 116

System Level Methodology for Functional Verification SoC
Alexander Adamov, Sergey Zaychenko, Yaroslav Miroshnychenko, Olga Lukashenko ................ 122

Path Sensitization at Functional Verification of HDL-Models
Alexandr Shkil, Yevgeniya Syrevitch, Andrey Karasyov, Denis Cheglikov ................................................. 126

Dynamic Register Transfer Level Queues Model for High-Performance Evaluation of the Linear Temporal Constraints
Vladimir Hahanov, Oleg Zaharchenko, Sergiy Zaychenko ......................................................................... 132

The Automation of Formal Verification of RTL Compilers Output
Pavlush Margarian .................................................................................................................................. 140

LOGIC, SYSTEM AND PHYSICAL SYNTHESIS

Congestion-Driven Analytical Placement
Andrey Ayupov, Alexander Marchenko ................................................................................................ 143

Estimation of Finite State Machine Realization Based on PLD
E. Lange, V. Chapenko, K. Boule ............................................................................................................. 149

Encoding of Collections of Fragment of Variables
Barkalov A.A., Ahmad Fuad Bader, Babakov R.M................................................................................... 153

An Algorithm of Circuit Clustering for Logic Synthesis
O. Venger, I. Afanasiev, Alexander Marchenko .......................................................................................... 156

CMOS Standard Cell Area Optimization by Transistors Resizing
Vladimir Rozenfeld, Iouri Smirnov, Alexander Zhuravlev ........................................................................ 163

Optimization of Address Circuit of Compositional Microprogram Unit
Wisniewski R., Alexander A. Barkalov, Larysa A. Titarenko ..................................................................... 167
Optimization of Circuit of Control Unit with Code Sharing
Alexander Barkalov, Larisa Titarenko, Malgorzata Kolopienczyk .................................................... 171

Routing a Multi-Terminal Nets with Multiple Hard Pins by Obstacle-Avoiding Group Steiner Tree Construction
J. D. Cho, A. I. Erzin, V. V. Zalyubovsky .................................................................................................. 175

Optimization for Electro- and Acousto-Optical Interactions in Low-Symmetric Anisotropic Materials
Kajdan Mykola, Laba Hanna, Ostrovskij Igor, Demyanyshyn Nataliya, Andrushchak Anatolij, Mytsyk Bohdan........................................ 179

Force-Position Control of the Electric Drive of the Manipulator
A.V. Zuev, V.F. Filaretov ......................................................................................................................... 184

FAULT TOLERANCE
K-out-of-n and K(m,n) Systems and their Models
Romankevych V., Potapova K., Hedayatollah Bakhtari ........................................................................ 189

Fault Tolerant Systems with FPGA-based Reconfiguration Devices
Vyacheslav S. Kharchenko, Julia M. Prokhorova.................................................................................... 190

Fault-Tolerant Infrastructure IP-cores for SoC: Basic Variants and Realizations
Ostroumov Sergii, Ushakov A. A., Vyacheslav S. Kharchenko............................................................... 194

Fault-tolerant PLD-based Systems on Partially Correct Automatons
Nataliya Yakymets, Vyacheslav Kharchenko.......................................................................................198

FME(C)A-Technique of Computer Network Reliability and Criticality Analysis
Elyasi Komari Iraj, Anatoliy Gorbenko.................................................................................................. 202

TEST GENERATION AND TESTABILITY
Scan Based Circuits with Low Power Consumption
Ondřej Novák, Zdeněk Plíva................................................................................................................... 206

Memory Address Generation for Multiple Run March Tests with Different Average Hemming Distance
S.V. Yarmolik, V.N. Yarmolik................................................................................................................ 212

Structural Method of Pseudorandom Fixed Weight Binary Pattern Sequences Generation
Romankevych A., Groi V., Fallahi Ali .................................................................................................. 217

Test Pattern Generation for Bridge Faults Based on Continuous Approach
N. Kascheev, F. Podyablonsky.............................................................................................................. 222

Hierarchical Analysis of Testability for SoCs
Maryna Kaminska, Vladimir Hahanov, Elvira Kulak, Olesya Guz.................................................... 226

Embedded Remote Wired or Wireless Communication to Boundary-Scan Architectures
Mick Austin, Ilkka Reis, Anthony Sparks.............................................................................................. 231

Economics Modeling the DFT of Mixed-Signal Circuits
Sergey G. Mosin...................................................................................................................................... 236

CAD TOOLS AND DEVICES
Optimal Electronic Circuits and Microsystems Designer
A.I. Petrenko ............................................................................................................................................ 239
Computer Aided Design Support of FSM Multiplicative Decomposition
Alexander Sudnitson, Sergei Devadze ................................................................. 241

Complex Process Engineering of Projection of Electronic Devices by Means of Automized System SATURN
D.V. Bagayev, A.C. Firuman .................................................................................. 247

Hand-Held Mobile Data Collecting Terminal
Armen Saatchyan, Oleg Chuvilo, Chaitanya Mehandru ..................................... 252

Logic and Fault Simulation Based on Multi-Core Processors
Volodymyr Obrizan, Valeriy Shipunov, Andiry Gavryushenko, Oleg Kashpur ........ 255

HES-MV – A Method for Hardware Embedded Simulation
Vladimir Hahanov, Anastasia Krasovskaya, Maryna Boichuk, Oleksandr Gorobets .... 257

Hierarchical Approach for Functional Verification of HW/SW System on Chip (SoC)
Oleksandr Yegorov, Podkolzin N., Yegor Denisov, Andrey Yazik ......................... 264

Output Buffer Reconfiguration in Case of Non Uniform Traffic
Vyacheslav Evgrafov ......................................................................................... 267

DESIGN METHODS AND MODELING

Time-Sensitive Control-Flow Checking Monitoring for Multitask SoCs
Fabian Vargas, Leonardo Picolli, Antonio A. de Alecrim Jr., Marlon Moraes, Márcio Gama 272

Development and Application of FSM-Models in Active-HDL Environment for Network Protocols Testing
Anna.V. Babich, Oleksandr Parfentiy, Eugene Kamenuka, Karina Mostovaya ........ 279

How to Emulate Network-on-Chip?
Peeter Ellervee, Gert Jervan ............................................................................. 282

Multistage Regular Structure of Binary Counter of ones Arbitrary Modulo
Saposhnikov V. V., Saposhnikov VL. V., Urganskov D. I. ..................................... 287

An Enhanced Analogue Current-Mode Structure of WP Control Circuit of Neural Networks
Hossein Aghababa, Leyla S.Ghazanfari, Behjat Forouzandeh ......................... 291

One-Parameter Dynamic Programming Algorithm for Optimal Wire Selection Under Elmore Delay Model
A.I. Erzin, V.V. Zalyubovsky ........................................................................... 296

Analytical Model of Clock Skew in Buffered H-Trees
Dominik Kasprowicz ..................................................................................... 301

High-Level Facilities for Modeling Wireless Sensor Networks
Anatoliy Doroshenko, Ruslan Shevchenko, Konstantin Zhereb ........................ 305

Class E Power Amplifier for Bluetooth Applications
Olga Antonova, George Angelov, Valentin Draganov ...................................... 311

An Automation Method for Gate-Count Characterization of RTL Compilers
Arik Ter-Galstyan ............................................................................................. 313

Algorithmic Method of The Tests Forming for Models Verification of Microcircuits Memory
M.K. Almaid, V.A. Andrienko, V.G. Ryabtsev ................................................ 317
SUM IP Core Generator – Means for Verification of Models–Formulas for Series Summation in RKHS
Vladimir Hahanov, Svetlana Chumachenko, Olga Skvortsova, Olga Melnikova.............................. 322

Design of Wavelet Filter Bank for JPEG 2000 Standard
Hahanova I.V., Hahanov V.I., Fomina E., Bykova V., Sorudeykin K……………………………………… 327

Design of Effective Digital Filters in FPGA
Pavel V. Plotnikov ................................................................................................................................... 332

POSTER SESSION

Applications of Combinatorial Cyclic Codes for Images Scan and Recognition
Vladimir Valkovskii, Dmitry Zerbin, Oleg Riznyk………………………………………………………………… 335

Architecture of Internet Access to Distributed Logic Simulation System
Ladyzhensky Y.V., Popoff Y.V. ................................................................................................................ 339

Computer System Efficient Diagnostics with the Usage of Real-Time Expert Systems
Gennady Krivoulya, Alexey Lipchansky, Olga Korobko………………………………………………………… 344

DASPUD: a Configurable Measurement Device
Nikolay P. Molkov, Maxim A. Sokolov, Alexey L. Umnov, Dmitri V. Ragozin ................................. 348

Design Methods of Self-Testing Checker for Arbitrary Number of Code Words of (m,n) Code

Dynamic Heat and Mass Transfer in Saline Water due to Natural Convection Flow over a Vertical Flat
Plate
Rebhi A. Damseh..................................................................................................................................... 361

Effect of Driving Forces On Cylindrical Viscoelastic Fluid Flow Problems
A. F. Khadrawi, Salwa Mrayyan, Sameh Abu-Dalo .............................................................................. 366

Evolutional Methods for Reduction of Diagnostic Information
D. Speranskiy ......................................................................................................................................... 371

Evolutionary Algorithms Design: State of the Art and Future Perspectives
Yuri R. Tsoly .................................................................................................................................. 375

Functional properties of faults on fault-secure FSM design with observing only FSM outputs
S. Ostanin................................................................................................................................................. 380

Hardware Methods to Increase Efficiency of Algorithms for Distributed Logic Simulation
Ladyzhensky Y.V., Teslenko G.A…………………………………………………………………………………….. 385

Information Embedding and Watermarking for Multimedia and Communication
Aleksandr V. Shishkin .......................................................................................................................... 386

Low Contrast Images Edge Detector
I.V. Ruban, K.S. Smelyakov, A.S. Smelyakova, A.I. Tymochko.............................................................. 390

Minimization of Communication Wires in FSM Composition
S.V. Zharikova, N.V. Yevtushenko........................................................................................................... 397

Neuro-Fuzzy Unit for Real-Time Signal Processing
Ye. Bodyanskiy, S. Popov......................................................................................................................... 403

IEEE EWDTW, Sochi, September 15-19, 2006 9
Hierarchical Analysis of Testability for SoCs

Maryna Kaminska, Vladimir Hahanov, Elvira Kulak, Olesya Guz
Kharkov National University of Radio Electronics, Ukraine
E-mail: {hahanov, maryna4329}@kture.kharkov.ua

Abstract

This paper presents the strategy of testable SoC design procedure. This approach based on the testability analysis on different levels of abstractions (gate level, register transfer level, system level). Analysis is based on structural analysis of SoC. Proposed methods give possibility to simplify the verification task and to generate test synthesis and and/or to improve faults covering for the given inputs. The main goal of the presented algorithms is to increase fault coverage before test generation and to decrease verification time. It could be reached by improving of testability and simplification of the verification task.

1. Introduction

As a complexity of today’s ASIC designs continues to increase, the challenge of verifying these designs intensifies at an even greater rate [1]. Testability is one of the most important factors that are considered at digital devices design along with reliability, speed and the cost. The low level of device testability leads to increasing of number of non-tested faults and verification time at design, production and operations stages. Therefore, the cost of diagnostic (a degree of faults concentration) decreases essentially during techniques of testability design.

The cost of a fault essentially increases in the process of ASIC crystal implementation (Fig. 1). Hence analysis of testability needs to be done at earlier level of device description. This is the main reason of development of the methods of testability analysis at the different levels of abstraction: system, RT, and gate levels.

Object under test – system on chip, which can be presented on different levels of abstraction.

Goal of work – maximal decreasing of test procedure cost; to provide digital circuit testability on all design levels of abstraction, till device manufacturing stage. To provide device testing possibility with minimal test by adding of scan cells on bottlenecks in circuit (circuit’s parts, which hard to test).

![Figure 1. Dependence between cost and verification Tasks: 1) to develop the method of testability analysis on different levels of abstraction. 2) Minimal increasing of observable lines in circuit or code lines, which will provide the best fault coverage. 3) To develop strategy of minimal additional lines selection. 4) To develop algorithm of circuit modification. 5) Verification and testing of developed methods on standard benchmarks.

It is need to provide testability analysis of developed system model on all of design stages. Here, the most adequate analysis is corresponding to most accurate model, represented on gate level. In this case, circuit is presented by the most detailed structure. Nevertheless, testability analysis on the more high levels of abstraction, where project model is represented as structure of interconnected components, laboriousness of analysis procedure is minimal, but testability values and further project modification based on boundary scan technology can essentially influence on diagnostic assurance and maintenance costs (time and economic expenses directed on test synthesis, fault simulation and diagnosis for each design stages). Using of IEEE DFT standards is presented on Fig. 2. Standard instructions are ensuring the scan and test modes of System on Chip (SoC) components.

Thus, for test procedure organization on the gate level is enough to use scan path, which consist of easy tested, modified boundary scan cells [2],[4],[7].

For testing procedure organization on register transfer level is practicable to use such DFT standards as IEEE 1149.1BS [4] or IEEE 1500SECT [5].

For device analysis and test procedure realization on system (algorithmic) level is proposed to present...
the program code as oriented marked Moore graph (FSM). To test such structure is supposed to use System JTAG standard [6].

<table>
<thead>
<tr>
<th>Description Levels</th>
<th>Structure for Testability Analysis</th>
<th>Boundary Scan Technologies</th>
</tr>
</thead>
<tbody>
<tr>
<td>System Level</td>
<td>Components</td>
<td>SystemJTAG IEEE 1149.1 BS</td>
</tr>
<tr>
<td>Processor Level</td>
<td>Architecture</td>
<td>IEEE 1500 Std IEEE 1149.1 BS</td>
</tr>
<tr>
<td>RT Level</td>
<td>Registers</td>
<td></td>
</tr>
<tr>
<td>Gate Level</td>
<td>Gates</td>
<td></td>
</tr>
</tbody>
</table>

Figure 2. Levels of abstraction

To succeed goal of work it is need to provide scan procedure of internal weakly observable and controllable lines in circuit.

Developed device, as a rule, can be presented as composition of control automata and operational automata. Structure of relations between control automata and operational automata could be specified by model on figure 3.

\[
M = \langle X, Y, Z, f, g \rangle, \]

where \( X = \{X_1, X_2, ..., X_i, ..., X_m\} \), \( Y = \{Y_1, Y_2, ..., Y_j, ..., Y_h\} \), \( Z = \{Z_1, Z_2, ..., Z_r, ..., Z_k\} \) — sets of inputs, internal and outputs variables. Relations between these variables could be described by generalized equations:

\[
Y(t) = f[X(t-1), X(t), Y(t-1), Z(t-1)];
Z(t) = g[X(t-1), X(t), Y(t-1), Y(t), Z(t-1)].
\]

(2)

Figure 3. Model of device

Structure of control automata could be presented as oriented marked Moore graph. Operational device, as usually, could be presented by regular structure, which easy to analyse on gate level or register transfer level. Such decomposition of control and operational automata is presented on figure 4.

Figure 4. Decomposition of CM and OM.

2. Testability analysis for FSM

In a FSM all hard reachable states have to be tested. The deadlock situation is also need to be tested. The places of branches localization and code feedbacks (if, case, loop, while operators) is also needed to be tested. The proposed method calculates the value of the vertex reachability (testability). The analysis is done on the marked Moore graph-scheme.

2.1. Calculation of reachability (controllability)

The reachability of the initial vertex is \( R(a_i) = 1 \). The controllability of each vertex depends of the value of reachability of the previous vertex and of the reachability index. The reachability can take the relative value from [0; 1]. The initial vertex has 100% of controllability. \( R(a_i) = 0 \) has the vertex that can not be accessible by any graph path

\[
R(a_j) = \sum_{i=1}^{m} R(a_{i-1}) \times \frac{1}{n} \times w_{ij}
\]

(3)

where \( n \) is the number of all possible set of graph path to the vertex \( a_i \); \( w_{ij} \) is edge weight: \( w_{ij} = 1/k_i; k \) is the number of outgoing edges, \( m \) is the number of incoming edges.

The Fig. 5 depicts the marked graph-scheme of Moore machine. The total controllability before modification is equal \( R_{total} = 0.333455 \).

Figure 5. Marked flow-chart of Moore FSM

\[
R(a_1) = 1; \quad R(a_2) = 1;
R(a_3) = 0.2863533467; \quad R(a_4) = 0.2863533467;
R(a_5) = 0.035794183; \quad R(a_6) = 0.008948545;
R(a_7) = 0.0363545; \quad R(a_8) = 0.045301.
\]
For our example the worst case of controllability is exists for the vertex $a_6$ (Fig. 6).

So the additional condition $x_4$ could be added to flowchart structure.

The total controllability before modification is equal $R_{\text{total}}=0.41725$ (Figure 5).

$$R(a_1) = 1; \quad R(a_2) = 1;$$

$$R(a_3) = 0.2863533467; \quad R(a_4) = 0.2863533467;$$

$$R(a_5) = 0.035794183; \quad R(a_6) = 0.05894;$$

$$R(a_7) = 0.039462; \quad R(a_8) = 0.0484105;$$

### 3. Testability analysis for RTL

Table presentation of digital devices is used to present models of functional elements and for their analysis. Table representation is the most practically feasible information perception form for user and computer, which based on cubic computation, mathematical tool for compact description of digital structures. Main disadvantage of table representation of finite state machine is dimension. It is could be decreased by using redundancy in alphabet for binary Boolean variables states coding. Thus, estimation of the testability is based on the topological analysis of the circuit. In probabilistic approaches values of controllability, observability, and testability are depend on truth table of primitive. In this case the extended probabilistic method for gate level could be used [3].

However, when device is presented by cubic coverage, problem of simulation of value ‘X’ is occurring. For obtaining of accurate analysis of cubic coverage, which stored in memory, it is proposed to redefine the symbol ‘X’. Testability analysis algorithm for RT level is following. Values of controllability for primary inputs are equal to $P(X_i) = 0.5$ (input of circuit could be set in logic one or logic zero as well). Values of controllability are calculated from primary inputs to primary outputs through circuit structure. Output values of block under test are dependent of values of controllability of inputs and logic zero (one) propagation ratio. Ratio could be calculated by formulas:

$$K(0) = \frac{n(0)}{n(0) + n(1)}; \quad K(1) = \frac{n(1)}{n(0) + n(1)}; \quad (4)$$

Thus, values of controllability of each node are calculated by following formulas:

$$P^0(X_{i+1}) = K(0) \cdot \prod_{i=1}^{m} P^0(X_i); \quad (5)$$

$$P^1(X_{i+1}) = K(1) \cdot \prod_{i=1}^{m} P^1(X_i); \quad (6)$$

where $n(1)$, $(n(0))$ is number of vectors, which gives logic one (zero) on the output of device; $m$ – number of inputs in logic block.

Values of the testability are calculated for each line (node). Indexes, that calculated are intended for comparative analysis of the testability of the nodes of the circuit. Analysis of the topology of the device is performed on the register transfer level description of the circuit. In case when convergent fan-outs are present in circuit, formulas have to be modified to following:

$$P^0(X_{i+1}) = K(0) \cdot \prod_{i=1}^{m} P^0(X_i) \cdot 2^{k+1}; \quad (7)$$

$$P^1(X_{i+1}) = K(1) \cdot \prod_{i=1}^{m} P^1(X_i) \cdot 2^{k+1}; \quad (8)$$

where $k$ – number of convergent fan-outs. This rule is valid for testability analysis on gate level as well.

### 4. Testability analysis for gate level

More detailed and adequate analysis is executed on the gate level. Here values of controllability, observability, and testability are calculated for each line in circuit. Such analysis is executed for circuits, which can not be tested by deterministic test.

Controllability $C_Y$ – the quantity of ability of the device to generate on a set line value 0 ($C_Y^0$) or 1($C_Y^1$) which depends on a logic function of the device. It decreases with the increasing of a distance of a line from external inputs of the circuit. Controllability can take the relative value, which belong to [0; 1] interval.
\( CY = 1 \) – has primary inputs of the device, where it is possible to set logic ‘0’ and ‘1’. \( CY = 0 \) – has line, that can not be set in any of the logic values.

Practically, most values of the controllability are situated between the limits of range \([0; 1]\). In general case, controllability of inputs of the gates is not equal 100%. Therefore controllability must consider ability to transmit logic values from gate and values of the controllability on its inputs:

\[
CN_Y^0(Y) = KC_Y^0 \cdot f^0, \quad CN_Y^1(Y) = KC_Y^1 \cdot f^1 \quad (9)
\]

where \( KC_Y \) – coefficient of the controllability transfer, that defined by the logic function of the gate (\( KC_Y^1 \) – for setting of logic one on the output of the gate, \( KC_Y^0 \) – for setting of logic zero on the output of the gate);

Coefficients of the controllability transfer are defined by these expressions:

\[
KC_Y^0 = \frac{N(0)}{N(1) + N(0)}, \quad KC_Y^1 = \frac{N(1)}{N(1) + N(0)} \quad (10)
\]

where \( N(0) \) (\( N(1) \)) – number of all methods of setting of logic zero (one) on the primitive output line.

\( f^0 \) – function, which defined by formula:

\[
f^0 = \left[ \sum_{\forall z^0} CY_i^0(X_1) + CY_j^0(X_2) + \ldots + CY_k^0(X_n) \right] / m \quad (11)
\]

where \( n \) – number of gate’s inputs; \( z^0 \) – input patterns \((X_1, X_2, \ldots, X_n)\), which allow to obtain logic ‘0’ on the output \( Y \); \( m \) – number of patterns \( z^0 \); \( i, j, \ldots, k \) \( \in \{0,1\} \) and equal to 0, if \( X_1, X_2, \ldots, X_n \) on \( z^0 \) are equal to zero value; and equal to 1, if \( X_1, X_2, \ldots, X_n \) on \( z^0 \) are equal to one value.

\( f^1 \) – function, which defined by formula:

\[
f^1 = \left[ \sum_{\forall z^1} CY_i^1(X_1) + CY_j^1(X_2) + \ldots + CY_k^1(X_n) \right] / p \quad (12)
\]

where \( n \) – number of gate inputs; \( z^1 \) – input patterns \((X_1, X_2, \ldots, X_n)\), which allow to obtain logic ‘1’ on the output \( Y \); \( p \) – number of patterns \( z^1 \); \( i, j, \ldots, k \) \( \in \{0,1\} \) and equal to 0, if \( X_1, X_2, \ldots, X_n \) on \( z^1 \) and take on ‘0’ values, and, equal to 1, if \( X_1, X_2, \ldots, X_n \) on \( z^1 \) take on ‘1’ values. Sum of \( z^0 \) and \( z^1 \) patterns is equal to \( 2^n \).

Calculation of controllability is beginning from primary inputs to primary outputs. It is not necessary to solve linear equations for sequential circuits, as in classical methods, because the fun-outs must be cut.

### 4.1. Calculation of the observability

Observability \( O_Y \) – the quantity of ability of the device to transport a condition of considered line on external outputs of the circuit which depends on logic functionality of the device.

Observability can take a relative value in \([0; 1]\). \( O_Y = 1 \) for primary output of the device. \( O_Y = 0 \), if it is impossible to change the logic value on the primary output by changing logic value in the node. Practically, most values of the observability are situated between the limits of range \([0; 1]\).

In general case (Fig. 7), transferring faults through primitive (logic gate) from inputs to output is depends on the ability to activate the appointed input. It is depends on the ability to set the fix values on the some/all inputs, which allows activating the path to appointed output of the device (the function of the controllability of these inputs).

\[
O_Y(X \rightarrow \text{primout}) = O_Y(Y \rightarrow \text{primout}) \cdot g, \quad (13)
\]

where \( \text{primout} \) – primary output of device; \( X \rightarrow Y \) – activation path; \( g \) – arithmetic mean of values of the controllability (on the inputs), which ensures activate of the input \( X \) to output \( Y \).

\[
g = \frac{CY_i^1(X_1) + CY_j^1(X_2) + \ldots + CY_k^1(X_{n-1})}{n-1} \quad (14)
\]

where \( n \) – number of inputs of device, \( (X_1, X_2, \ldots, X_{n-1}) \) – input patterns \( (z_a) \), which provides the activation of \( X_n \rightarrow Y \) path, \( i, j, \ldots, k \in \{0,1\} \) and equal to 0, if \( X_1, X_2, \ldots, X_{n-1} \) on \( z_a \) take on ‘0’ values, and, are equal to 1, if \( X_1, X_2, \ldots, X_{n-1} \) on \( z_a \) take on ‘1’ values. Values of observability are calculated from primary outputs to primary inputs.

### 4.2. Calculation of the testability

Testability of node can be calculated as multiplication of it controllability and observability.

\[
TY^0(Y) = CY^0(Y) \cdot O_Y(Y), \quad (15)
\]

\[
TY^1(Y) = CY^1(Y) \cdot O_Y(Y), \quad (16)
\]

\[
TY(Y) = (TY^0(Y) + TY^1(Y)) / 2, \quad (17)
\]

where \( TY^0(Y) (TY^1(Y)) \) – 0 – testability (1- testability) of node \( Y \); \( TY(Y) \) – testability of node \( Y \).
General value of circuit’s testability can be presented as measure of average laboriousness of test generation for circuit’s node; therefore, this measure can be presented as an arithmetic mean of testabilities of all nodes in circuit, i.e.

$$TY_{\text{circuit}} = \frac{\sum TY(Y_i)}{L},$$  \hspace{1cm} (18)

$TY_{\text{circuit}}$ – general testability of circuit, $L$ – number of nodes in circuit. For convenient interpretation of the results it is taken the 8-th root of the controllability, observability and testability values. Method complexity (performance) is linear.


Experimental Results

Strategy of point selection for the circuit modification on the gate level is: 3% of lines are selected with the minimal value of controllability and observability at the same time to the selected lines the lines with maximal value is added. Usually the number of such selected lines is small; this is a feature of the evaluation method. 3% of selected lines have been selected from the restrictions on the external additional contacts in the device – no more than 2-5%. Results of fault coverage for this case are presented in table 1. The proposed modification strategy consists of separation of test and functional operations. A conditional vertex is added on each selected vertices in FSM and scan cells are placed on selected lines in circuit for gate level and register transfer level. Such approach allows to obtain 100% controllability on selected bottleneck.

Table 1. Fault coverage for combinational and sequential circuits on the gate level

<table>
<thead>
<tr>
<th>Circuit</th>
<th>FC before modification</th>
<th>FC after modification</th>
<th>Test</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISCAS’85</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C432</td>
<td>87,173%</td>
<td>100%</td>
<td>Determ.</td>
</tr>
<tr>
<td>C499</td>
<td>99,763%</td>
<td>100%</td>
<td>Determ.</td>
</tr>
<tr>
<td>C880</td>
<td>93,563%</td>
<td>100%</td>
<td>Determ.</td>
</tr>
<tr>
<td>C3540</td>
<td>97,798%</td>
<td>98,53%</td>
<td>Random</td>
</tr>
<tr>
<td>C6288</td>
<td>99,653%</td>
<td>99,81%</td>
<td>Random</td>
</tr>
<tr>
<td>C20000</td>
<td>72,094%</td>
<td>99,71%</td>
<td>Random</td>
</tr>
<tr>
<td>ISCAS’99</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>sb01</td>
<td>96,667%</td>
<td>100%</td>
<td>Rand.</td>
</tr>
<tr>
<td>sb02</td>
<td>92,308%</td>
<td>100%</td>
<td>Rand.</td>
</tr>
<tr>
<td>sb03</td>
<td>98,077%</td>
<td>100%</td>
<td>Rand.</td>
</tr>
<tr>
<td>sb04</td>
<td>91,969%</td>
<td>100%</td>
<td>Rand.</td>
</tr>
<tr>
<td>ISCAS’89</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>s208</td>
<td>99,537%</td>
<td>100%</td>
<td>Rand.</td>
</tr>
<tr>
<td>s298</td>
<td>98,750%</td>
<td>100%</td>
<td>Rand.</td>
</tr>
<tr>
<td>s344</td>
<td>99,123%</td>
<td>100%</td>
<td>Rand.</td>
</tr>
</tbody>
</table>

The testability index calculation for FSM and application of the strategy of device modification are presented in the Table 2 (index of reachability of graph modification: $R_{BM}$ – before and $R_{AM}$ – after). Here the variable with the least index of reachability is the operated variable.

Table 2. Analysis of testability for FSM

<table>
<thead>
<tr>
<th>FSM</th>
<th>$R_{BM}$</th>
<th>$R_{AM}$</th>
<th>Differ</th>
<th>Increase</th>
</tr>
</thead>
<tbody>
<tr>
<td>Traf light</td>
<td>0.23306</td>
<td>0.3133</td>
<td>0.08</td>
<td>23.9%</td>
</tr>
<tr>
<td>Multiplier</td>
<td>0.44271</td>
<td>0.5508</td>
<td>0.11</td>
<td>25%</td>
</tr>
<tr>
<td>CSK M</td>
<td>0.38360</td>
<td>0.5232</td>
<td>0.14</td>
<td>36.8%</td>
</tr>
<tr>
<td>MPA M</td>
<td>0.24702</td>
<td>0.4001</td>
<td>0.16</td>
<td>64.7%</td>
</tr>
<tr>
<td>CA M</td>
<td>0.33346</td>
<td>0.4173</td>
<td>0.08</td>
<td>23.9%</td>
</tr>
</tbody>
</table>

A one additional variable in the graph structure increases quality of faults covering approximately on 15-30%. In the Table 3 the quality of the test for the sample Multiplier is absent because the quality of faults covering before modification is equal 100%.

Table 3. Faults coverage for FSM

<table>
<thead>
<tr>
<th>FSM</th>
<th>$Q_{BM}$</th>
<th>$Q_{AM}$</th>
<th>Increase</th>
</tr>
</thead>
<tbody>
<tr>
<td>Traf light</td>
<td>50.00</td>
<td>62.5</td>
<td>12.5</td>
</tr>
<tr>
<td>CSK M</td>
<td>71.43</td>
<td>85.71</td>
<td>14.28</td>
</tr>
<tr>
<td>MPA M</td>
<td>37.5</td>
<td>75.0</td>
<td>37.5</td>
</tr>
<tr>
<td>CA M</td>
<td>62.5</td>
<td>87.5</td>
<td>25</td>
</tr>
</tbody>
</table>

Practical importance and advantages: 1) methods simplicity; no needs to solve system of linear equations for sequential circuits; 2) simple strategy of bottlenecks selection; 3) simplicity and regularity of circuits modification; 4) scan cells could be completely tested independently of other circuit part; 5) possibility to provide high level of circuit testability (minimal or zero number of undetectable lines) before test generation; 6) spending of device analysis on the earliest design stages and increasing of Yield Ratio.

Disadvantages: 1) no absolutely guarantee to obtain 100% fault coverage; 2) increasing of testing time due to using of SP and F modes.

6. References
