Test Data Modeling and Implementation with Enhanced Accuracy for the Diagnosis of a DUT provided with Real-World Failure Models

Gürkan Uygur, Wolfgang Sörgel, Wolfgang Magerl, Sebastian M. Sattler

Chair of Reliable Circuits and Systems, LZS, Friedrich-Alexander-University Erlangen-Nuremberg,
Paul-Gordan-Str. 5, 91052 Erlangen, Germany
Email: {guerkan.uygur,wolfgang.soergel,wolfgang.magerl,sebastian.sattler}@fau.de

***Abstract*—We present our data modeling and implementation for testing a device of elementary digital as well as analog functions provided with real-world failure models as a jumper construction. A modeled open circuit failure is done by unplugging the (normally plugged) respective jumper, and analogously a modeled short circuit failure is done by plugging the (normally unplugged) respective jumper. The DUT is supplied by a current limited voltage power supply. The programmable current limitation activates the overcurrent relay to protect the device in case of short circuit failure. The ATE resources for stimuli, measure and test evaluation are provided by an adequate uC-Board with programmable analog-to-digital io-channels. The test automation is implemented by the firmware of the uC, which corresponds to the test program of a conventional ATE including the pin-mapping, the test function library and the test flow (schedule). With this in mind, we are speaking about the Micro-ATE (uATE).**

Within the use case above, we discuss the challenge for localization of the so called dominated (dominent) faults, which are not detectable uniquely due to the dominating faults within the scope of the stuck-at failure modeling. At this case the testing accuracy has to be enhanced. We solve this problem definition by deploying external current sources to provide additional stimuli at the given test points. Our approach results in the overall data modeling, which covers the test tree, the test flow, and the diagnose flow for fault localization.

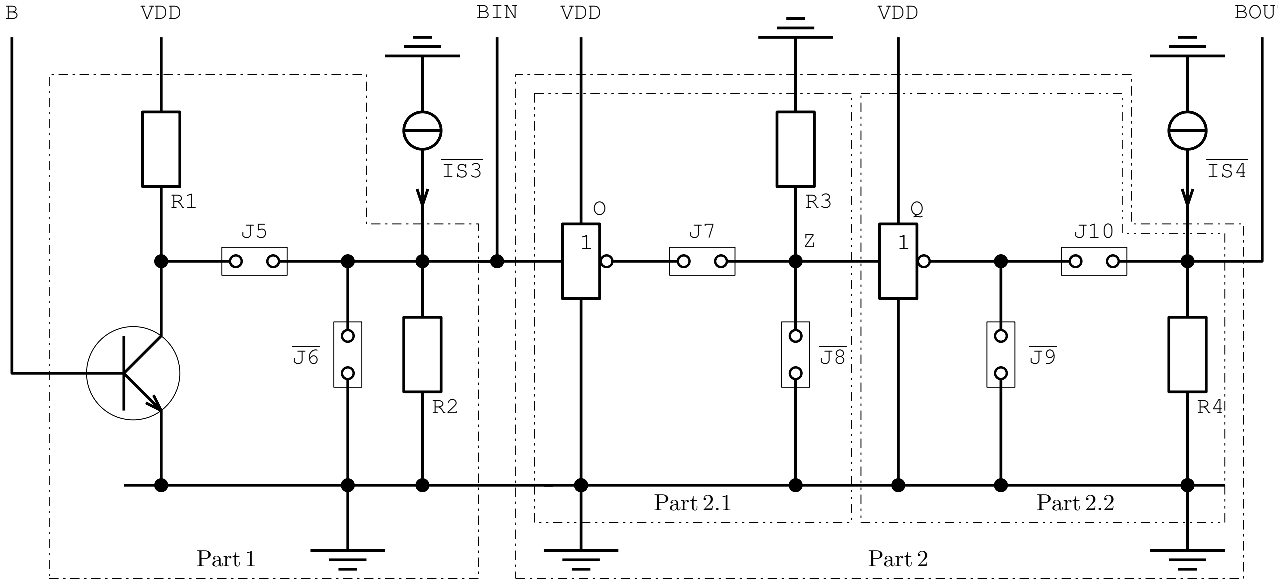
***Index Terms*—test, diagnose, accuracy, data modeling, stuck-at, dominating vs. dominated, adaptive fault localization**

I. Introduction

SAFETY-CRITICAL systems like avionics and automotive are increasingly developing into a modular, distributed, dynamic discrete event systems with complex requirements specification and high demand on safety. Further, they increasingly profit from the microelectromechanical systems (MEMS) technology as well as the Integrated Circuit (IC) technology, which bundle analog-to-digital functionality with high efficiency and quality at low cost [1]. On the other hand, design and technology are more and more reaching their limits, so that simulation based validation is incapable to cover all possible scenarios, which is an unacceptable foundation, especially for safety-critical systems [2]. Hence, it is well known that widely established simulation techniques are not by themselves adequate to ensure the correctness of complex systems [3]. The alternative is to employ theoretically sound formal verification and test [4]. Due to the increasing complexity of structure and functionality also the test of the real-world structure becomes more and more complex, hence the formal assignment between the (virtual) functionality and the (real-world) structure becomes more and more critical. This results in a high effort for design verification and test such that specification-oriented testing is getting more and more under pressure. This constellation leads to an ever-increasing challenge particularly regarding safety-critical circuits and systems. As a consequence, from the technical as well as economic point of view, a suitable fail-safe design with known and observable faults provided with enhanced Design-for-Test (DfT) measure seems to be more feasible than a safe-to-fail design to warrant normal operation without fail in each case of processing.

Within the laboratory conditions, the Test Environment (TE) could primarily be equipped with general purpose hand-held devices like voltmeters as well as oscilloscopes. For automated manufacturing testing of a Device Under Test (DUT) cost-prohibitive (monolithic) Automated Test Environments (ATEs) are deployed, which are specialized for and geared to fully automated high volume production test provided with powerful tools and universal Test Description Language (TDL) [5] for test development and debugging. Alternatively, for low volume DUT specific testing at low cost, the rack-and-stack (modular) ATE solutions are deployed while in case of a special approach the test development can be quite time consuming due to the lack of universal test development environment [6], [7]. Further, the build-in self-test is one other DfT approach, which provides the device with additional testability for verifying regarding to manufacturing defects. At this point, it should be noticed, that — as opposed to specification-oriented test — the “known” manufacturing defects are tested based on the underlying “failure model”, e.g. the well known stuck-at failure and bridging fault, respectively [8]. Due to the ever increasing size and complexity of Very-Large-Scale-Integration (VLSI) designs, there is also an increasing demand for investigation of new approaches for the automatic test pattern generation (ATPG) for both test and diagnosis of faults [9].

Organization of the paper: In Section II a fail-safe combinational DUT structure is introduced and the



(a) use case structure with primary input B, primary outputs (BIN, BOUT) and external current sources ($\overline{\mathtt{IS3}}, \overline{\mathtt{IS4}}$) for diagnosis

in	supply	fault-free								out	
B	VDD	J5	$\overline{\mathtt{J6}}$	J7	$\overline{\mathtt{J8}}$	O	$\overline{\mathtt{J9}}$	J10	Q	BIN	BOUT
0	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	0	0

(b) simple specification of the DUT in normal (fault-free) operation — the current sources ($\overline{\mathtt{IS3}}, \overline{\mathtt{IS4}}$) are not deployed

Fig. 1: Combinational fail-safe DUT — enhanced test accuracy is provided by external (low-active) current sources

corresponding data modeling is discussed. In Section III the challenge for localization of the so called dominated faults is discussed and the problem solution is presented. Further, the overall modeling of data, test and diagnose for single as well as multiple fault detection is presented. Additionally, the implementation of the uATE is described. Finally the paper closes with a conclusion in Section IV.

II. Case Study: A Fail-Safe Structure

Let the DUT structure in **Fig.** 1 be given. VDD denotes the pin of the voltage power supply. B denotes the gate pin of the high-active transistor and the primary input pin of the DUT, respectively. R1, R2, R3 and R4 denote the resistors with the resistance ratio

$$\frac{\mathtt{R1}}{\mathtt{R2}} = \frac{\mathtt{R1}}{\mathtt{R3}} = \frac{\mathtt{R1}}{\mathtt{R4}} = \frac{1}{10} \ .$$

A. Circuit Failure Modeling

In normal operation the serial jumpers J5, J7 and J10 are plugged, termed as $\mathtt{J5} = 1$, $\mathtt{J7} = 1$ and $\mathtt{J10} = 1$ or rather briefly J5, J7 and J10 using the high-active notation. Accordingly, a modeled open circuit failure is done by unplugging the (normally plugged) respective jumper termed as $\mathtt{J5} = 0$, $\mathtt{J7} = 0$ and $\mathtt{J10} = 0$ or rather briefly $\overline{\mathtt{J5}}$, $\overline{\mathtt{J7}}$ and $\overline{\mathtt{J10}}$. And analogously, in normal operation the parallel jumpers $\overline{\mathtt{J6}}$, $\overline{\mathtt{J8}}$ and $\overline{\mathtt{J9}}$ are unplugged, termed as $\overline{\mathtt{J6}} = 1$, $\overline{\mathtt{J8}} = 1$ and $\overline{\mathtt{J9}} = 1$ or rather briefly $\overline{\mathtt{J6}}$, $\overline{\mathtt{J8}}$ and $\overline{\mathtt{J9}}$ using the low-active notation. Accordingly, a modeled short circuit failure is done by plugging the (normally unplugged) respective jumper termed as $\overline{\mathtt{J6}} = 0$, $\overline{\mathtt{J8}} = 0$ and $\overline{\mathtt{J9}} = 0$ or rather briefly $\overline{\overline{\mathtt{J6}}}$, $\overline{\overline{\mathtt{J8}}}$ and $\overline{\overline{\mathtt{J9}}}$. In addition, O denotes that the first CMOS inverter exists and $\overline{\mathtt{O}}$ denotes that the respective CMOS inverter is missing, analogously the notations Q and $\overline{\mathtt{Q}}$ for the second CMOS inverter.

B. Specification

The DUT is provided with two capture pins, BIN and BOUT. In case of $\mathtt{B} = 0$ the transistor is not conductive, thus BIN is assigned with digital 1 ($\mathtt{BIN} = 1$) and — after double negation — BOUT is assigned with digital 1 ($\mathtt{BOUT} = 1$), too. And analogously, in case of $\mathtt{B} = 1$ the transistor is conductive, thus BIN is assigned with digital 0 ($\mathtt{BIN} = 0$) and — after double negation — BOUT is assigned with digital 0 ($\mathtt{BOUT} = 0$), too. This is the very simple specification of the DUT, see Fig. 1b.

C. Over-Current Protection

Obviously, in case of conductive input transistor ($\mathtt{B} = 1$) the resistor R1 limits the cross-current and protects against over-current, respectively. But it should be noticed, that in case of the short circuit defect $\overline{\overline{\mathtt{J9}}}$ and functional inverter Q, if $\mathtt{Z} = 0$ then the p-MOS transistor of the second CMOS inverter conducts leading to the short circuit current. Analogously, the short circuit current occurs in case of the short circuit defect $\overline{\overline{\mathtt{J8}}}$ with connection line (J7) to the first functional inverter O and $\mathtt{BIN} = 0$. To prevent any damage, we deploy a current limited voltage power supply: on exceeding the specified current limit the voltage power supply switches to the over-current protection mode. In that case, the ATE connects VDD of the DUT to the ground so that VDD becomes defined 0V.

D. Defects

In case of the open circuit defect $\overline{\texttt{J5}}$, the pull-down resistor R2 warrants that BIN is assigned with digital 0 ($\texttt{BIN} = 0$) even in the case of $\texttt{B} = 0$, see Fig. 1a Part 1: this failure is termed as stuck-at 0 denoted as s@0. Similarly, in case of the short circuit defect $\overline{\overline{\texttt{J6}}}$, BIN is connected to ground ($\texttt{BIN} = 0$), which results in s@0 failure, too. Analogously, in case of the defects $\overline{\overline{\texttt{J9}}}$, $\overline{\texttt{J10}}$ and $\overline{\texttt{Q}}$, respectively, 0V is captured at BOUT ($\texttt{BOUT} = 0$), which also results in s@0 failure, see Fig. 1a Part 2.2. Further, in case of functional Part 2.2, any defect in Part 2.1 results in $\texttt{Z} = 0$ and thus after negation $\texttt{BOUT} = 1$, observed as stuck-at 1 (s@1).

It is remarkable, that in any modeled defect, the capture pins BIN and BOUT are designed to be assigned with a fail-safe signal level.

III. Enhanced Test Data Modeling

Section II-D makes clear that multiple defects result in s@0 and s@1, respectively, so that they are not locatable separately. Hence, the question occurs, how to enhance the test accuracy. To surmount this challenge we deploy external current sources to provide additional stimuli at the given test points BIN and BOUT, see Fig. 1. $\overline{\texttt{IS3}}$ and $\overline{\texttt{IS4}}$ denote the first and the second current source, respective. Syntactically, $\overline{\texttt{IS3}}$ and $\overline{\texttt{IS4}}$ mean that both current sources are disabled, and $\overline{\overline{\texttt{IS3}}}$ and $\overline{\overline{\texttt{IS4}}}$ mean that both current sources are enabled.

A. Dominating vs. Dominated Faults

The use of external current sources leads to more varied test results, namely: In case of the single open circuit defect $\overline{\texttt{J5}}$, the signal at BIN changes from 0 to 1 ($\texttt{BIN} = 0 \rightarrow 1$) on enabling the first current source ($\overline{\texttt{IS3}} = 1 \rightarrow 0$). Analogously, in case of the single open circuit defect $\overline{\texttt{J10}}$, the signal at BOUT changes from 0 to 1 ($\texttt{BOUT} = 0 \rightarrow 1$) on enabling the second current source ($\overline{\texttt{IS4}} = 1 \rightarrow 0$).

Contrastingly, in case of the short circuit defect $\overline{\overline{\texttt{J6}}}$, the signal at BIN still remains low on enabling the first current source. Analogously, in case of the short circuit defect $\overline{\overline{\texttt{J9}}}$, the signal at BOUT still remains low on enabling the second current source.

Obviously, one can notice that the short circuit defect dominates the open circuit defect. Thus, we are speaking of dominated (dominent) faults $\overline{\texttt{J5}}$ and $\overline{\texttt{J10}}$ denoted as d@0; briefly, d@0 means "dominated by s@0".

B. Undefined Signal at BIN

The use of external current sources has its price also in data modeling: Let's say the over-current protection mode is activated ($\texttt{VDD} = 0$), the input pin B is assigned with 0 ($\texttt{B} = 0$) and the first current source is enabled ($\overline{\overline{\texttt{IS3}}}$). Then, due to the resistance ratio $\frac{\texttt{R1}}{\texttt{R2}} = \frac{1}{10}$, an analog voltage value is captured, which is neither a digital 1 nor a digital 0. In that case, BIN is modeled as undefined, symbolized with $*$: $\texttt{BIN} = *$.

C. Data Modeling

To create the overall data model, at first the encoding (header) is needed. It consists of the input vector x, the programing vector p and the output vector y. According to Fig. 1, the encoding of the Part 1 is given as follows

- $x = (\texttt{B}, \overline{\texttt{IS3}}, \texttt{VDD})$
- $p = (\texttt{J5}, \overline{\texttt{J6}})$
- $y = (\texttt{BIN})$

and the encoding of the Part 2 is given as follows

- $x = (\texttt{BIN}, \overline{\texttt{IS4}}, \texttt{VDD})$
- $p = (\texttt{J7}, \overline{\texttt{J8}}, \texttt{O}, \overline{\texttt{J9}}, \texttt{J10}, \texttt{Q})$
- $y = (\texttt{BOUT}, \texttt{VDD})$

Fig. 2 shows the corresponding block diagram. **Table** I shows the data model of Part 1, Part 2.1 and Part 2.2 — the abbreviated comment "a cbs" means "already covered by spec".

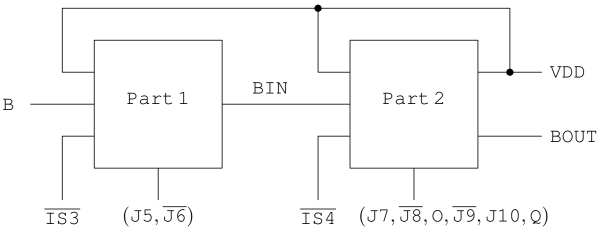

VDD
B
Part 1
BIN
Part 2
BOUT
$\overline{\texttt{IS3}}$
$(\texttt{J5}, \overline{\texttt{J6}})$
$\overline{\texttt{IS4}}$
$(\texttt{J7}, \overline{\texttt{J8}}, \texttt{O}, \overline{\texttt{J9}}, \texttt{J10}, \texttt{Q})$

Fig. 2: Block diagram

Each data model contains its overall modeling information about the test tree, the test flow and the diagnose flow for fault localization. For example, **Fig.** 3 shows the test tree of Part 2.2. **Fig.** 5, 6 and 7 show the test flow of Part 2.2, Part 1 and Part 2.1. **Fig.** 4, 8 and 9 show the diagnose flow of Part 2.2, Part 1 and Part 2.1.

D. uATE in a Nutshell

We deployed Arduino (Mega 2560) uC-Board to implement the uATE. Additionally, we used Raspberry Pi to implement the user interface for the test engineer as well as the client to program and control the uATE and for post processing, see **Fig.** 10. Further, we deployed a relay (changeover) to switch VDD of the DUT to the ground.

We used the digital output port of the Arduino to source current to the respective data output pin of the DUT. To not overload the digital output port we equipped it with a diode and resistor to only source a limited current.

The test and diagnose flow was implemented by defining the initial test function and further defining the next test function depending on the branch code returned from the current test function. The program flow was required to maximize the test coverage on minimal use of external current sources (Resources) and minimal test time. The test automation was implemented by the firmware of the uC. The test program is developed in terms of a conventional ATE including the pin-mapping, the test function library and the test flow (schedule).

Table I: Data model of the DUT

Part 1

B	$\overline{\mathtt{IS3}}$	VDD	J5	$\overline{\mathtt{J6}}$	BIN	Comment
0	–	1	1	1	1	spec
1	–	1	1	1	0	
–	–	1	–	0	0	s@0
–	1	1	0	1	0	
–	0	1	0	1	1	d@0
–	1	0	1	1	0	spec
1	0	0	1	1	0	
0	0	0	1	1	*	
–	1	0	–	0	0	s@0 a cbs
1	0	0	–	0	0	
0	0	0	–	0	*	
–	1	0	0	1	0	s@0a cbs
–	0	0	0	1	1	d@0

Part 2.1

BIN	VDD	J7	$\overline{\mathtt{J8}}$	O	Z	VDD	Comment
1	1	1	1	1	0	1	spec
0	1	1	1	1	1	1	
–	1	–	–	0	0	1	s@0
–	1	0	–	1	0	1	s@0
1	1	1	0	–	0	1	s@0 a cbs
0	1	1	0	1	0	0	s@0
–	0	1	1	1	0	0	spec
–	0	–	–	0	0	0	s@0 a csb
–	0	0	–	1	0	0	
–	0	1	0	–	0	0	

Part 2.2

Z	$\overline{\mathtt{IS4}}$	VDD	$\overline{\mathtt{J9}}$	J10	Q	BOUT	VDD	Comment
0	–	1	1	1	1	1	1	spec
1	–	1	1	1	1	0	1	
–	1	1	–	–	0	0	1	s@0
0	1	1	0	–	1	0	0	s@0, d@1
1	1	1	0	–	1	0	1	s@0 a cbs
–	1	1	1	0	–	0	1	s@0
1	0	1	–	0	–	1	1	s@1
0	0	1	1	0	–	1	1	s@1 a cbs, d@0
0	0	1	–	0	0	1	1	s@1 a cbs
0	0	1	0	0	1	1	0	s@1 a cbs, d@0, d@1
1	0	1	0	1	–	0	1	s@0 a cbs, d@1
0	0	1	0	1	0	0	1	s@0
0	0	1	0	1	1	0	0	s@0, d@1
1	0	1	1	–	0	1	1	s@1, d@0
0	0	1	1	–	0	1	1	s@1 a cbs
–	1	0	1	1	1	0	0	spec
1	0	0	1	1	1	0	0	
0	0	0	1	1	1	*	0	
–	1	0	–	–	0	0	0	s@0 a cbs
–	1	0	–	0	1	0	0	
–	–	0	0	1	–	0	0	
–	0	0	–	0	–	1	0	s@1, d@0
–	0	0	1	1	0	1	0	

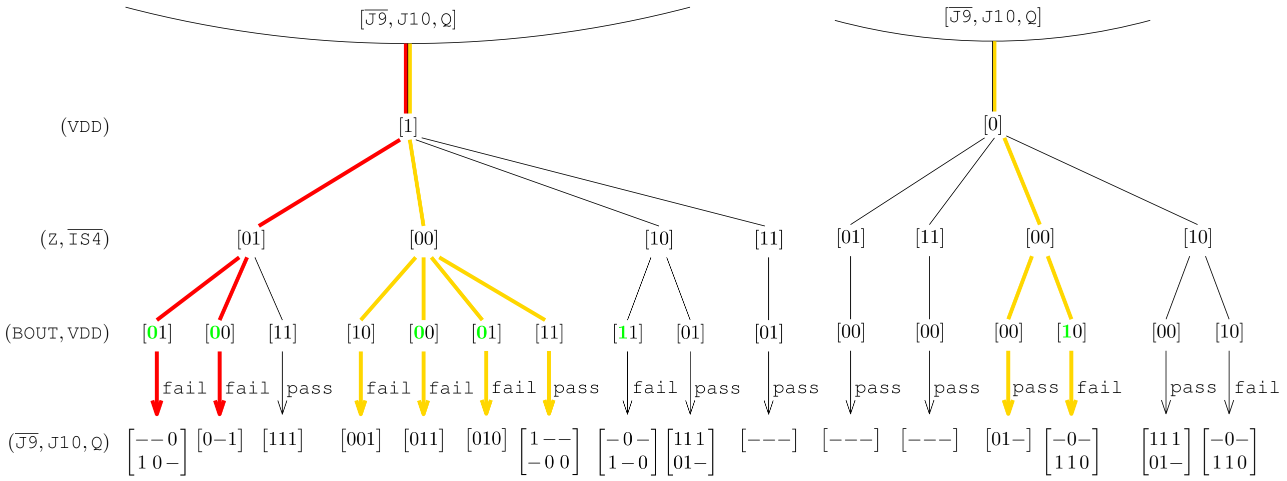


Fig. 3: Test tree of Part 2.2

IV. Conclusion

In this work, the overall event-based data modeling for test and diagnose was presented. For this purpose, the DUT structure was decomposed in sub-structures. For each substructure the corresponding encoding was formulated. The structure was modeled as a block diagram and the data model was created.

Proceeding from the data model the corresponding test tree was derived. The test tree already provides a suitable representation of paths to deduce the test flow and the diagnose flow for defect localization. To enhance the test accuracy external current sources were deployed and the data modeling was upgraded. Accordingly, the enhanced test and diagnose flow were created. In doing so the minimization of test time and resources was pursued. Finally, in accordance with the data modeling, the test and diagnose flow were implemented as a firmware and a suitable uC-Board was implemented as a uATE. Thus, the feasibility was shown.

It is remarkable that the presented data modeling is not limited to single faults, but also takes multiple faults into consideration. Furthermore, it is remarkable that enhancing the test accuracy using current sources results in an additional DfT criterion and establishes the concept of dominated failure in contrast to the stuck-at failure. Additionally, the presented data modeling exhibits adap-

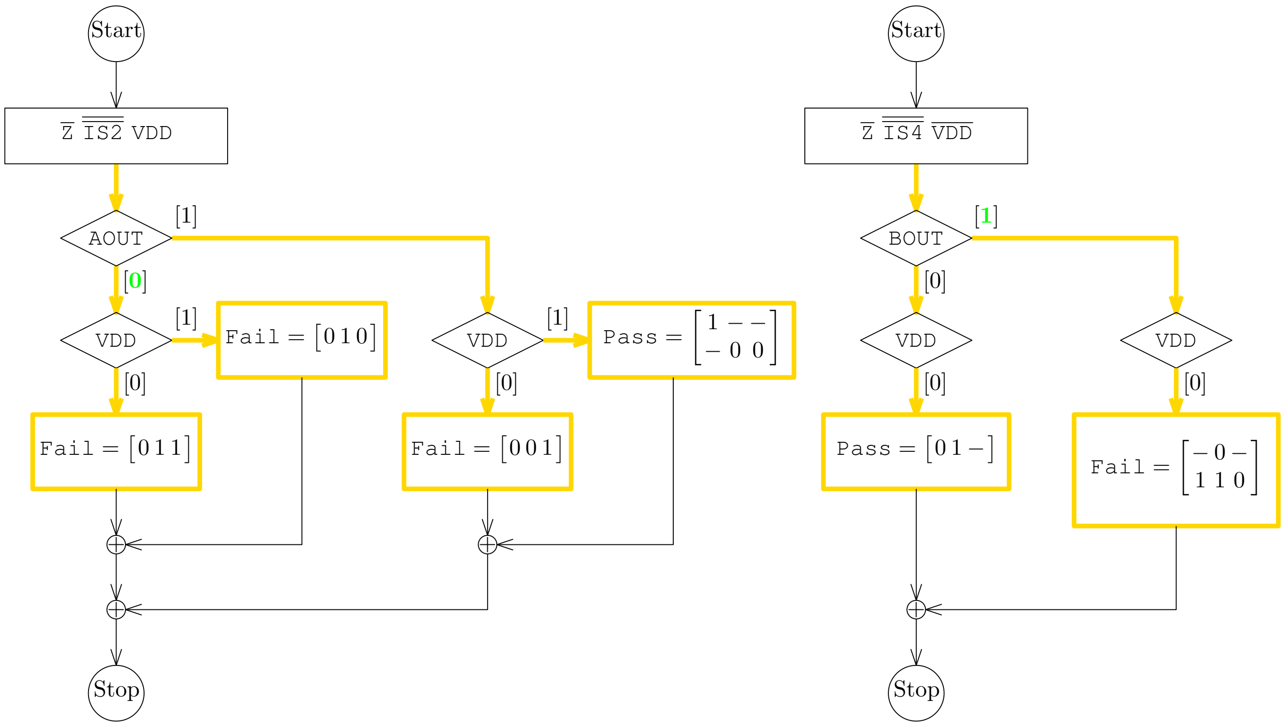

Start
$\overline{Z}$ $\overline{\overline{IS2}}$ VDD
AOUT
[1]
[0]
VDD
[1]
$\text{Fail} = [0\,1\,0]$
[0]
$\text{Fail} = [0\,1\,1]$
VDD
[1]
$\text{Pass} = \begin{bmatrix} 1 & - & - \\ - & 0 & 0 \end{bmatrix}$
[0]
$\text{Fail} = [0\,0\,1]$
Stop
Start
$\overline{Z}$ $\overline{\overline{IS4}}$ $\overline{VDD}$
BOUT
[1]
[0]
VDD
[0]
$\text{Pass} = [0\,1\,-]$
VDD
[0]
$\text{Fail} = \begin{bmatrix} - & 0 & - \\ 1 & 1 & 0 \end{bmatrix}$
Stop

Fig. 4: Diagnose flow of Part 2.2

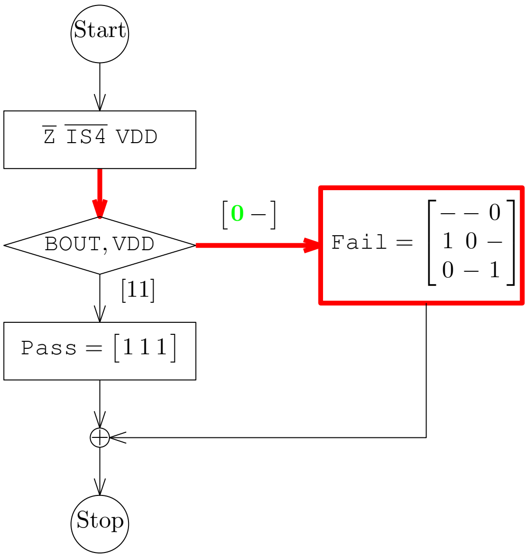

Start
$\overline{Z}$ $\overline{IS4}$ VDD
BOUT, VDD
[0 −]
$\text{Fail} = \begin{bmatrix} - & - & 0 \\ 1 & 0 & - \\ 0 & - & 1 \end{bmatrix}$
[11]
$\text{Pass} = [1\,1\,1]$
Stop

Fig. 5: Test flow of Part 2.2

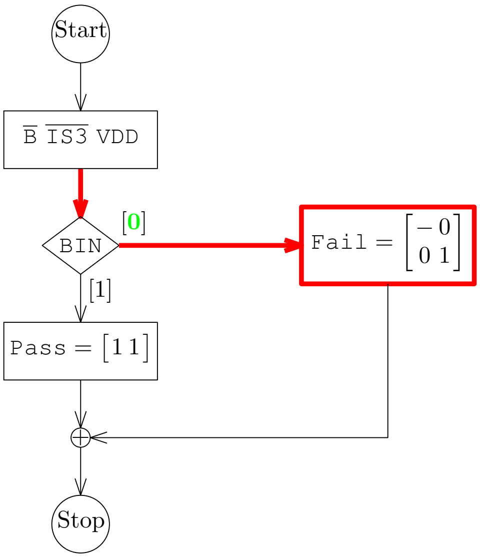

Start
$\overline{B}$ $\overline{IS3}$ VDD
BIN
[0]
$\text{Fail} = \begin{bmatrix} - & 0 \\ 0 & 1 \end{bmatrix}$
[1]
$\text{Pass} = [1\,1]$
Stop

Fig. 6: Test flow of Part 1

tive test and diagnose patterns suitable for both the operation mode (in-flight) and the laboratory conditions.

Working with the presented uATE approach can help to understand failure detection and fault localization in safety-critical circuits and systems, and serves as a prototype for processor based self testing and diagnosis.

Outlook: The presented modeling is suitable for generating of algorithms for enhanced test and diagnosis (defect localization). Furthermore, the used data format TVL (Ternary Vector List [10], [11]) is suitable to handle big data on low-level and hardware implementation, respectively, providing maximal processing power and minimal memory requirements. Hence, it is obvious to automatically generate high performance algorithms for adaptive test and diagnosis based on the presented data modeling. At this point, the invertible decomposition of the given DUT structure using the automata based parallel composition [12] can additionally provide “divide and conquer” and hence this can result in further reduction of complexity.

Acknowledgement: We express our thanks to Dr. Benjamin Cella, Dr. Michael Schmidt and Dr. Alfred Engler from Liebherr-Elektronik GmbH for inspiring and fruitful discussions and the provision of the PCB.

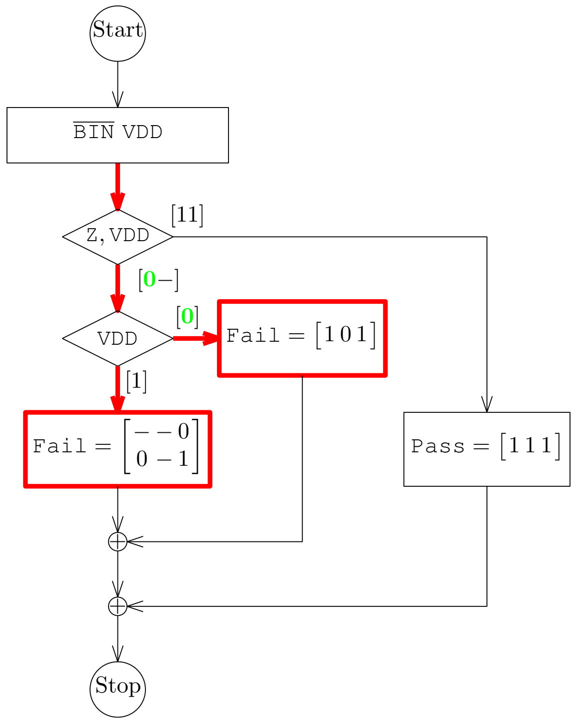


Fig. 7: Test flow of Part 2.1

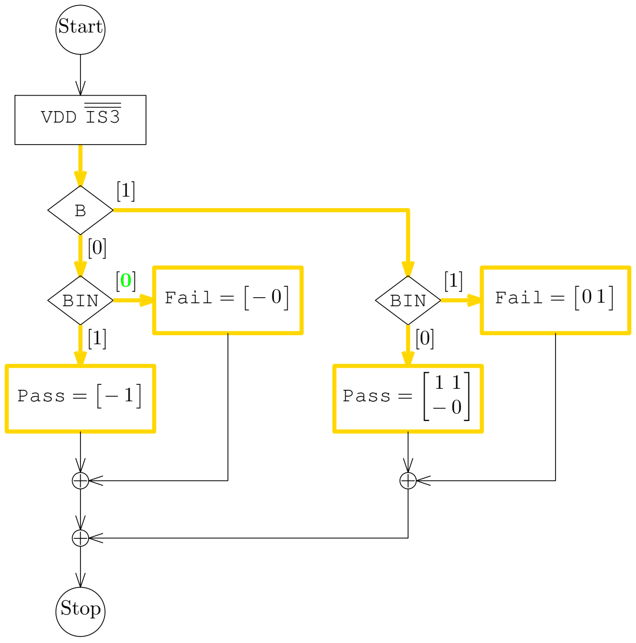


Fig. 8: Diagnose flow of Part 1

References

[1] D. Foty, “The future of “Moore’s Law“ - Does it have one?” in *Synthesis, Modeling, Analysis and Simulation Methods and Applications to Circuit Design (SMACD), 2015 International Conference on*, Sept 2015.

[2] Y. Xu, “Algorithms for automatic generation of relative timing constraints,” Ph.D. dissertation, Salt Lake City, UT, USA, 2011.

[3] R. P. Kurshan and K. L. McMillan, “Analysis of digital circuits through symbolic reduction,” *IEEE Trans. on CAD of Integrated Circuits and Systems*, vol. 10, no. 11, pp. 1356–1371, 1991.

[4] T. Yoneda, T. Kitai, and C. J. Myers, “Automatic derivation of timing constraints by failure analysis,” in *CAV*, ser. Lecture Notes in Computer Science, E. Brinksma and K. G. Larsen, Eds., vol. 2404. Springer, 2002, pp. 195–208.

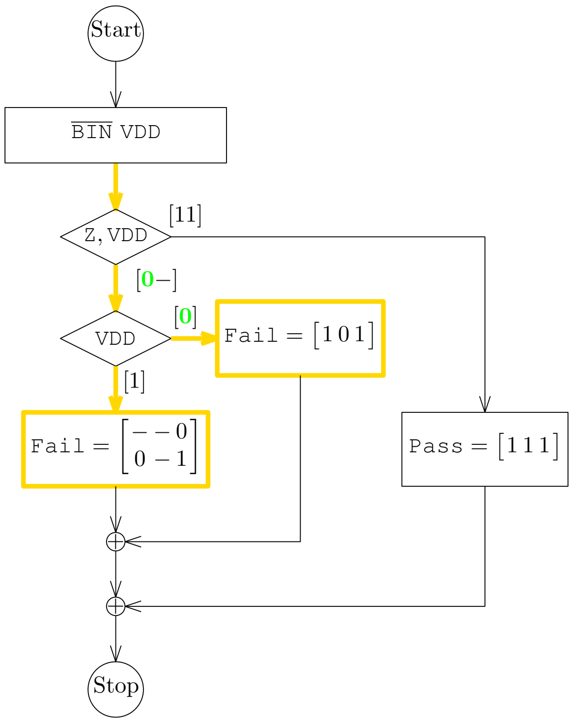


Fig. 9: Diagnose flow of Part 2.1

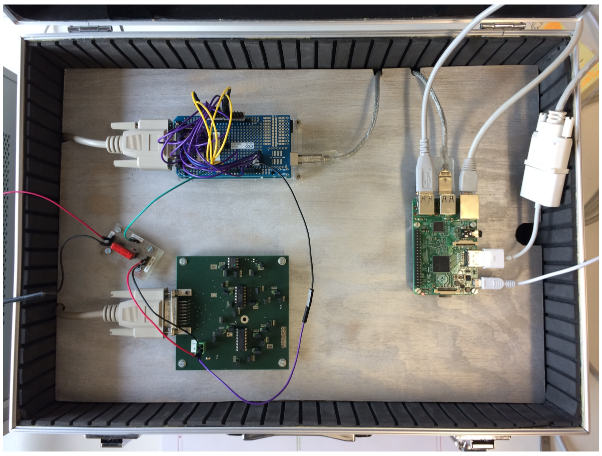


Fig. 10: uATE in a nutshell: Raspberry Pi, Arduino, DUT

[5] *Advantest Corporation, “Viewpoint TDL Programming Manual“, Manual No. 8262578-19 , Vol 1., November 2001.*

[6] A. Kafton, “Advanced Technology Speeds RF-Integrated-Circuit Testing,” *ARFTG Conference Digest-Fall, 46th*, vol. 28, pp. 104–113, Nov. 1995.

[7] R. Smith, D. Rosenthal, and S. Stevens, “Hybrid Modular Test System Architectures,” *AUTOTESTCON, 2006 IEEE*, pp. 658–658, Sept. 2006.

[8] Y.-R. Shieh and C.-W. Wu, “Design of cmos pscd circuits and checkers for stuck-at and stuck-on faults,” *VLSI Design*, vol. 5, no. 4, pp. 357–372, 1998.

[9] P. Raiola, J. Burchard, F. Neubauer, D. Erb, and B. Becker, “Evaluating the effectiveness of d-chains in sat-based atpg and diagnostic tpg,” *Journal of Electronic Testing*, vol. 33, no. 6, pp. 751–767, Dec 2017.

[10] Dieter Bochmann and Bernd Steinbach, *Logikentwurf mit XBOOLE*. Verlag Technik Berlin, 1991.

[11] C. Posthoff and B. Steinbach, *Logic Functions and Equations: Binary Models for Computer Science*. Boston, MA: Springer US, 2004, ch. Logic Functions and Equations, pp. 37–89.

[12] Gürkan Uygur and Sebastian Sattler, “A New Approach for Modeling Inconsistencies in Digital-Assisted Analog Design,” *Journal of Electronic Testing*, vol. 32, no. 4, pp. 491–503, 2016.