IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

DESIGN AND IMPLEMENTATION OF LOW-POWER ALU FOR BRISK ARCHITECTURE

O. VIVEDHINI[1], C. SENTHILKUMAR[2]
1 PG STUDENT 2 ASSISTANT PROFESSOR
ME-VLSI DESIGN
AVS ENGINEERING COLLEGE, SALEM, TAMILNADU, INDIA

ABSTRACT

A low-power (LP) programmable generator capable of producing pseudorandom test patterns with desired toggling levels and enhanced fault coverage gradient compared with the best-to-date binary robust invariant scalable key (BRISK)-based pseudorandom test pattern generators. It is comprised of a linear finite state machine (a linear feedback shift register or a ring generator) driving an appropriate phase shifter, and it comes with a number of features allowing this device to produce binary sequences with preselected toggling (PRESTO) activity. In this method to automatically select several controls of the generator offering easy and precise tuning. The same technique is subsequently employed to deterministically guide the generator toward test sequences with improved fault-coverage-to pattern- count ratios. In an LP test Compression method that allows shaping the test power envelope in a fully predictable, accurate, and flexible fashion by adapting the PRESTO-based logic BRISK (LBRISK) infrastructure. The proposed hybrid scheme efficiently combines test compression with LBRISK, where both techniques can work synergistically to deliver high quality tests. Experimental results obtained for industrial designs illustrate the feasibility of the proposed test schemes and are reported.

Keywords: Brisk, Presto, design-for-test and hybrid scheme.

1. INTRODUCTION
LOW POWER VLSI AND TESTING

Power and energy consumption of digital systems may increase significantly during testing. This extra power consumption due to test application may give rise to severe hazards to the circuit reliability. Moreover, it may be responsible for cost, performance verification as well as technology related problems and can dramatically shorten the battery life when on-line testing is considered. In this paper, we present a survey of the low power testing techniques that can be used to test VLSI systems. In the first part, the paper explains the problems induced by the increased power consumed during functional testing of a circuit, in either external testing or binary robust invariant scalable key (BRISK). Next, we survey state-of-the-art techniques that exist to reduce this power/energy consumption during test mode and allow non-destructive testing of the device under test. The consumed energy directly corresponds to the switching activity generated in the circuit during test application, and has impact on the battery lifetime during remote testing. The average power consumption is given by the ratio between the energy and the test time. This parameter is even more important than the energy as hot spots and reliability problems may be caused by constantly high power consumption. The peak power consumption corresponds to the highest switching activity generated in the circuit under test during one clock cycle. The peak power determines the thermal and electrical limits of components and the system packaging requirements. If the peak power exceeds certain limits, the correct functioning of the entire circuit is no longer guaranteed.

Although over the next years, the primary objective of manufacturing test will remain essentially the same to ensure reliable and high quality semiconductor products conditions and consequently also test solutions may undergo a significant evolution. The semiconductor technology, design characteristics, and the design process are among the key factors that will impact this evolution. With new types of defects that one will have to consider to provide the desired test quality for the next technology nodes such as 3-D, it is appropriate to pose the question of what matching design-for-test (DFT) methods will need to be deployed. Test compression, introduced a decade ago, has quickly become the main stream DFT methodology. However, it is unclear whether test compression will be capable of coping with the rapid rate of technological changes over the next decade. Interestingly, logic binary robust invariant scalable key (LBRISK), originally developed for board, system, and in-field test, is now gaining acceptance for production test as it provides very robust DFT and is used increasingly often with test compression. This hybrid approach seems to be the next logical evolutionary step in DFT. It has potential for improved test quality; it may augment the abilities to run at-speed power aware tests, and it can reduce the cost of manufacturing test while preserving all LBRISK and scan Compression advantages. Attempts to overcome the bottleneck of test data bandwidth between the tester and the chip have made the concept of combining LBRISK and test data compression a vital research and development area.

In particular, several hybrid BRISK schemes store deterministic top-up patterns (used to detect random pattern resistant faults) on the tester in a compressed form, and then use the existing BRISK hardware to decompress these test patterns Some solutions embed deterministic stimuli by using compressed weights or by perturbing pseudorandom vectors in various fashions. If BRISK logic is used to deliver compressed test data, then underlying encoding schemes typically take advantage of low fill rates, as originally proposed in LFSR coding ,which subsequently evolved first into static LFSR reseeding and then into dynamicLFSR reseeding .Thorough surveys of relevant test compression techniques can be found, As with conventional scan-based test, hybrid schemes, due to the high data activity associated with scan-based test operations, may consume much more power than a circuit under-test was designed to function under. With overstressing devices beyond the mission mode, reductions in the operating power of ICs in a test mode have been of concern for years. Full-toggle scan patterns may draw several times the typical functional mode power, and this trend continues to grow, particularly over the mission mode's peak power. This power induced over-test may result in thermal issues, voltage noise, power droop, or excessive peak power over multiple cycles which, in turn, cause a yield loss due to instant device damage, severe decrease in chip reliability, shorter product lifetime, or a device malfunction because of timing failures following a significant circuit delay increase, for example. Abnormal switching activity may also cause fully functional chips to fail during testing because of phenomena, such as IR-drop, crosstalk, or *di/dt* problem. As the BRISK power consumption can easily exceed the maximum ratings when testing at speed, scan patterns must be shifted at a programmable low speed, and only the last few cycles and the capture cycle are applied at the maximum frequency. In the burst-mode approach presented in typically five consecutive clock cycles are used. The first four cycles serve shifting purposes, whereas the last one is designated for capture. The objective is to stabilize the power supply before the last shift and capture pulses are applied, which are critical for at-speed tests. To reduce the voltage droop related to a higher circuit activity, a burst clock controller slows down some of the shift cycles. It allows a gradual increase of the circuit activity, thereby reducing the *di/dt* effect. The controller can gate the shift clocks, depending on the needs for gradually warming up of the circuit. Low power (LP) test compression schemes adapt again LFSR reseeding to reduce scan-in transitions as the low fill rates make it possible to deliver identical test data to scan chains for a number of shift cycles directly from the decompressor, thereby reducing the number of transitions. Propose a PRPG for LP BRISK applications. The generator primarily aims at reducing the switching activity during scan loading due to its preselected toggling (PRESTO)levels. It can assume a variety of configurations that allow a given scan chain to be driven either by a PRPG itself or by a constant value fixed for a given period of time. Not only the PRESTO generator allows loading scan chains with patterns having low transition counts, and thus significantly reduced power dissipation, but it also enables fully automated selection of its controls such that the resultant test patterns feature desired, user-defined toggling rates. It will demonstrate that this flexible programming can be further used to produce tests superior to conventional pseudorandom vectors with respect to a resultant fault-coverage-to-test-pattern-count ratio.

This paper culminates in showing that the PRESTO generator can also successfully act as a test data decompressor, thus allowing one to implement a hybrid test methodology that combines LBRISK and ATPG-based embedded test compression. Thesis the first LP test compression scheme that is integrated in every way with the BRISK environment and lets designers shape the power envelope in a fully predictable, accurate, and flexible fashion. As a result, it creates an environment that can be used to arrive at an efficient hybrid solution combining advantages of scan compression and logic BRISK. In addition, both techniques can complement each other to address, for example, a voltage drop caused by a high switching activity during scan testing, constraints of at-speed ATPG-produced test patterns, or new fault models. Data compression and finally, it wraps up with this paper is organized as follows. Introduces the basic operational principles of the PRESTO generator, while presents all architectural details of its structure with a brief discussion of the generator's abilities to produce patterns with various toggling rates. Demonstrates how the PRESTO generator can be programmed in order to yield pseudorandom test patterns of desired switching activity. Experiments validating this technique are discussed. In addition, a method to achieve higher BRISK fault coverage with shorter test application time by deploying the native PRESTO features is described. A PRESTO-based LP test data decompressors introduced I, which is followed by the presentation of the corresponding test data-encoding algorithmic. This paper concludes with a variety of comprehensive experimental results related to both performance of the low power PRPG and the power aware test data compression and finally, it wraps up with. The use of pseudorandom vectors; to test combinational circuits effectively avoids long and complex algorithmic test pattern generation procedures. Only a fault-free circuit simulation is required for the correct circuit output response. An important use of pseudorandom test patterns is in systems with BRISK (Built-in Self-Test) which internally generate test vectors. In such systems a custom test set may be placed in a look-up table, making it very costly to store and apply. Also, many test pattern generation programs produce pseudorandom test patterns initially, augmenting them with algorithmically generated test vectors when necessary.

2.LITERATURE SURVEY

P. GIRARD, L. GUILLER, C. LANDRAULT, "A MODIFIED CLOCK SCHEME FOR A LOW POWER BRISK TEST PATTERN GENERATOR" ,2001.

Low Power consumption has become increasingly important in hand-held communication systems and battery operated equipment, such as laptop computers, audio and video-based multimedia products, and cellular phones. For this new class of battery-powered devices, the energy consumption is a critical design concern since it determines the lifetime of the batteries. In addition, the capabilities presented by advanced submicron CMOS technology allowing to put millions on transistors on chip and clocking at hundreds of MHz have compounded the problem of power/energy consumption. A strong push towards reducing power consumption is also coming from producers of high-end systems. The cost associated with packaging and cooling of such devices is huge and technological constraints are severe: unless power consumption is reduced, the resulting heat limits systems performance. Intensive research efforts have been devoted to develop algorithmic and technological solutions for reducing power consumption during normal circuit operation; they are not concerned with power consumption minimization during test operation. The main motivation for considering power consumption during test operation is that power and energy of a digital system are considerably higher in test mode than in system mode. It has been shown in that the power consumption during test can be as high as 200% of the power consumed in the normal mode. The reason for this increased power consumption is that test patterns cause as many nodes switching as possible while a power saving system mode only activates a few modules at the same time. Another reason is that successive functional input vectors applied to a given circuit during system mode have a significant correlation, while the correlation between consecutive test patterns (produced by an LFSR for example) may be very low. This elevated switching activity during test may be responsible for cost, reliability, performance verification, autonomy and technology related problems. A survey of these problems is given. For example, in battery-powered devices, the power consumed during application of power-up or periodical on-line tests, which are often implemented resorting to the Built-In Self-Test (BRISK) approach, can dramatically shorten the battery lifetime.

C.V. KRISHNA AND NUR A. TOUBA,” HYBRID BRISK USING AN INCREMENTALLY GUIDED LFSR”.

The test data volume required for testing increasingly complex system-on-chip (SOC) designs continues to grow rapidly and is outstripping the capabilities of ATE (automated test equipment). One well-known solution to this problem is to use binary robust invariant scalable key (BRISK). BRISK involves performing test pattern generation and output response compaction on the chip. The most economical BRISK schemes are based on pseudo-random pattern testing. The problem with pseudo-random pattern testing, however, is that it generally does not provide high enough fault coverage due to the presence of random-pattern-resistant (r.p.r.) fault. There are two solutions to this problem. One is to modify the circuit to eliminate the random pattern resistance by inserting test points, and the other is to modify the test pattern generator by adding additional hardware to generate patterns that detect the hard faults. Both approaches have significant drawbacks. Test point insertion requires modifying the function logic which can degrade system performance, and modifying the test pattern generator can require large amounts of additional silicon area. In order to reduce the hardware overhead for BRISK, a hybrid approach between external testing and BRISK can be taken. A “hybrid BRISK” scheme is one in which some external data from the tester is combined with the BRISK hardware to achieve the desired fault coverage. A hybrid BRISK scheme reduces the test data stored on the tester compared with full external testing, but it does not require as much hardware overhead as full stand-alone BRISK. A simple hybrid BRISK scheme is to use STUMPS architecture to apply pseudo-random patterns to detect the random pattern testable faults, and then use deterministic vectors from the tester to detect the random pattern resistant (r.p.r.) faults. Results for using this scheme on large industrial designs have shown that it only achieves around a 30-50% reduction in tester storage requirements compared with conventional external testing. More sophisticated hybrid BRISK schemes have been developed including using hybrid patterns, folding counters, two dimensional compressions, weighted pattern testing, and RESPI.

JINKYU LEE AND NUR A. TOUBA,” LFSR-RESEEDING SCHEME ACHIEVING LOW-POWER DISSIPATION DURING TEST”, 2007.

Power dissipation during test is a significant problem as the size and complexity of systems-on-chip (SOCs) continue to grow. During scan shifting, more transitions occur in the flip-flops compared to what occurs during normal functional operation. This problem is further compounded when pseudorandom filling of the unassigned input values is employed. Excessive power dissipation during test can increase manufacturing costs by requiring the use of a more expensive chip packaging or causing unnecessary yield loss. In this paper, a new test-data-compression scheme based on linear feedback shift register (LFSR) reseeding that significantly reduces power consumption during test is proposed. Test-data volume has also increased dramatically as the size and the complexity of chips grow. Consequently, there has been a lot of work on test-data-compression schemes that can be used to reduce tester storage and bandwidth requirements. Commercial tools for test data compression, which are based on LFSR reseeding including Test compress by Mentor Graphics, SOC BRISK by Synopsys, and ELT-Comp by Logic Vision, have been introduced. The basic idea in LFSR reseeding is to generate deterministic test cubes by expanding seeds. A seed is an initial state of the LFSR that is expanded by running the LFSR. Given a deterministic test cube, a corresponding seed can be computed by solving a set of linear equations (one equation for each specified bit) based on the feedback polynomial of the LFSR. Since typically only 1%–5% of the bits in a test vector are specified, most bits in a test cube do not need to be considered when a seed is computed because they are don’t care bits. Therefore, the size of a seed is much smaller than the size of a test vector. Consequently, reseeding can significantly reduce test-data storage and bandwidth.

S. KUNDU AND S. CHATTOPADHYAY,” EMBEDDING A LOW POWER TEST SET FOR DETERMINISTIC BRISK USING A GRAY COUNTER”, 2011.

Suilt-in Self-Test (BRISK) is being increasingly used today for large and complex chips. Based on the test application method, BRISK can be classified into two types test-per-clock and test-per-scan. In test-per-clock method, a test pattern is applied to the CUT in every clock cycle and the test response is captured by a response analyzer. In test-per-scan scheme, a test pattern is serially loaded into the scan chain and then the pattern is applied to the CUT. Thus, for an n input circuit, n + 1 cycles are needed to apply a pattern. Based on the pattern generation scheme, BRISK can also be classified as pseudo exhaustive, pseudo-random and deterministic. Generally, in BRISK solution, pseudo-random patterns are generated using LFSR or CA for detecting the random pattern detectable faults. For the random pattern resistant faults, techniques based on test point insertion, reseeding, bit flipping or fixing, and weighted random pattern testing have been proposed in literature. Another technique to test random pattern resistant faults is based on deterministic BRISK. In a deterministic BRISK, it takes into account a precomputed set of patterns and generates the patterns among other patterns. The most trivial way to do that is to store the patterns in a ROM and apply one pattern in every clock cycle. However, the area overhead is extremely high. The solutions are: using a mapping logic between LFSR and CUT or test set embedding. In test set embedding, a large test set TS is produced in such a way that a pre-computed test set TD is produced within TS (|TD| ≤ |TS|). Several techniques have been proposed for test embedding. In use of binary counter as a pattern generator has been shown. They proposed two kinds of operations to minimize the cycle length. The first one is the preprocessing operation such as constant column elimination, identical column merging and complemented column merging. The second operation, known as basic operation, consists of column permutation and complementary column generation.

HANS-JOACHIM LDERLICH," MULTIPLE DISTRIBUTIONS FOR BIASED RANDOM TEST PATTERNS, HANS",1990.

Testing by random planners has many advantages compared to other test strategies, for instance the self-test capability, less computing time and the high coverage of parametric faults. An extensive literature has been published during the last few years concerning problems of random tests, as computing fault detection probabilities and test lengths. Unfortunately, most of these papers are only dealing with special views of the subject Therefore it is necessary to list some basic facts as prerequisites of the following investigations. Among these basic facts, there is a new Theorem establishing" hound of the probability that all faults of a given set are detected by a given amount of random patterns. Another theorem proves that a real random test and a pseudo-random test by shift-register sequences require the same length, if the number of primary inputs is sufficiently large. In section 3 the complexity of computing an optimized random test scheme is determined, and since this problem is at least upward, we avoid the exact calculation using an efficient heuristic in section 4. Some implementation details are given in section 5, and results are discussed in section 6. Finally, we present a system· generating weighted random test panamas according to multiple distributions, which is used for tube external test of circuits with an integrated scan-path.

3. EXISTING SYSTEM

3.1 GENERAL BRISK STRUCTURE

Many BRISK techniques have been developed. The vast majority of them use a pseudo-random pattern generator (PRPG) to produce test vectors that detect the easy-to-detect faults, which mostly represent more than 90% of the total faults. For the remaining faults, test vectors are either applied externally, or they are generated by the BRISK structure itself. Linear feedback shifts registers (LFSR) or cellular automata (CA) are mostly used as PRPGs, due to their simplicity and good properties concerning implementation space demands and the good fault coverage. The patterns are generated by a test pattern generator (TPG), then they are fed to the circuit-under-test (CUT) and the circuit's responses are evaluated.

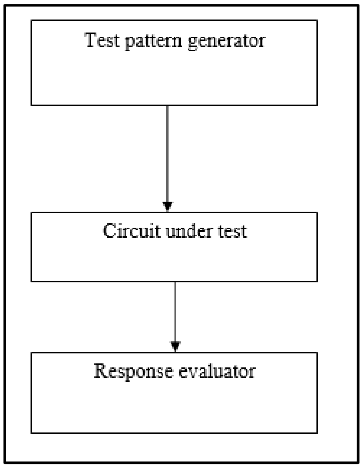


FIG 1: STRUCTURE OF BRISK

3.2 TEST-PER-CLOCK IMPLEMENTATION

BRISK Test-Per-Clock based upon same process of testing like automatic test equipment (ATE). In BRISK we are using circuitry instead of equipment so designers design a circuit for compressed implementation of test pattern generator. The implementation of test-per-clock BRISK. In normal mode multiplexer allows normal inputs to the circuit under test (CUT), in test mode operation**.** LFSR is used to generator test pattern for testing of CUT. Control inputs of multiplexer are generated by central test controller. Output response compacter (ORC) is used for compression of the outputs of CUT. There are some losses in the output of ORC. In testing we compare actual result with expected result so as in BRISK we compare actual result with expected result (called golden signature).

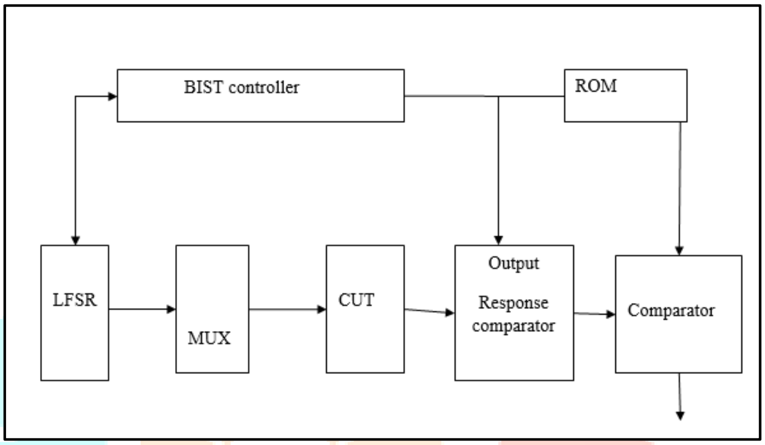


FIG 2: BASIC ARCHITECTURE

If actual result does not match as expected result so there is fault in circuit. Expected is stored in ROM for comparison. The basic architecture is shown with normal LFSR as a test pattern generator. LFSR generate pseudo-random pattern for testing.

3.3 TEST-PER-SCAN IMPLEMENTATION

In this paper we discussed test-per-scan based on low transition pattern generator (LT-PRG). LT-PRG reduces transition at scan inputs during scan shift operation of BRISK. The architecture of LT-PRG. This architecture is designed by using r stage LFSR, a K-input AND gate, and a T flip flop. Either normal or inverting outputs of LFSR stages are connected to each of K inputs of the AND gate. If we increase the no of K, then it will increase the length of test sequence and decrease the fault coverage so in this architecture we used the value of K= 2 or 3. T flip-flop remains in its state and hold the previous value as long as the input of the T flip-flop is set at 1. T flip-flop changes its state in every few clock cycles because in most of the cases output of the AND gate is zero (input of T flip-flop), so in scan chain have same value for most of the cases. In result of this transition probability in CUT will decrease.

3.4 DISTRIBUTED BRISK CONTROL SCHEMES

The goal in these approaches is to determine the BRISK blocks of a complex design to be activated in parallel at each stage of the test session in order to reduce the number of concurrently tested modules. The average power is reduced and consequently, the temperature related problems avoided by the increase of the test time duration. On the other hand, the total energy remains constant and the autonomy of the system is not increased.

3.5 VECTOR FILTERING ARCHITECTURES

As each vector applied to the CUT consumes power but not every vector generated by the pseudo-random TPG contributes to the final fault coverage, the vector filtering architectures consist in preventing application of non-detecting vectors to the CUT. This approach is very effective in reducing power without reducing fault coverage, but does not preserve the CUT from excessive peak power consumption and can lead to high area overhead.

3.6 LOW POWER TEST PATTERN GENERATORS

TPGs based either on LFSRs or Cellular Automata (CA) are carefully designed to reduce the activity at circuit inputs, thus reducing power consumption. These approaches effectively reduce power during test but sometimes at a cost of sub-optimal fault coverage and with no reduction of the peak power consumption.

3.7 CIRCUIT PARTITIONING FOR LOW POWER BRISK

This approach consists in partitioning the original circuit into structural sub-circuits so that each sub-circuit can be successively tested through different BRISK sessions. In partitioning the circuit and planning the test session, the average power, the peak power and the energy consumption during BRISK are minimized at a low expense in terms of area overhead and with no loss of fault coverage. The only drawback of this approach is that it requires circuit design modification.

3.8 THE LOW POWER TPG

The idea behind the use of such a low power TPG is to reduce the number of transitions on primary inputs at each clock cycle of the test session, hence reducing the overall switching activity generated in the CUT. Let us consider a CUT with n primary inputs. A n-stage primitive polynomial LFSR with a clock CLK would be used in a conventional pseudorandom BRISK scheme. Here, we use a modified LFSR composed of n D-type flip-flops and two clocks CLK/2 and CLK/2, and constructed as depicted. As one can observe, this modified LFSR is actually a combination of two n/2- stage primitive polynomial LFSRs, each of them being driven by a single clock CLK/2 or CLK/2. The D cells belonging to the first LFSR (referred to as LFSR-1 in the sequel) are interleaved with the cells of the second LFSR (referred to as LFSR-2 in the sequel), thus allowing to better distribute the signal activity at the inputs of the CUT. The advantage of the modified LFSR composed of two interleaved *n/2*-stage LFSRs (over a simpler structure composed of two separated *n/2*-stage LFSRs) is that it allows to better distribute the signal activity at the circuit inputs during the BRISK session. This is particularly important for circuits in which the input cones of the primary outputs are highly nonlover lapping. In this case, two separated LFSRs would activate only one part of the circuit in a given time interval, instead of the whole circuit with the proposed structure in which two LFSRs are interleaved. Shorter test lengths to reach a target fault coverage can hence be predicted with the proposed interleaved LFSR structure.

3.9 COMPLETE BRISK TPG STRUCTURE

This structure is first composed of a test clock module which provides test clock signals CLK/2 and CLK/2 from the master clock CLK used in the normal mode. The signal “Test” allows to switch from the test mode (=0) to the normal mode. As two different clock speeds are needed for the TPG, two clock trees are used in our proposed BRISK scheme rather than a single one. These clock trees are carefully designed so as to correctly balance the clock signals feeding each part of the modified LFSR. Finally, the main part of the TPG (*i.e.* the modified LFSR) is connected to the CUT.

3.10 MINIMIZING SWITCHING ACTIVITY DURING BRISK

The BRISK TPG proposed in this paper reduces switching activity in the CUT during BRISK by reducing the number of transitions at scan inputs during scan shift cycles. In this paper, we assume that the sequential CUT is implemented in CMOS, and employs full-scan. If scan input is assigned, where, at time and assigned the opposite value at time, then a transition occurs at time. The transition that occurs at scan input can propagate into internal circuit lines causing more transitions. During scan shift cycles, the response to the previous scan test pattern is also scanned out of the scan chain. Hence, transitions at scan inputs can be caused by both test patterns and responses. Since it is very difficult to generate test patterns by a random pattern generator that cause minimal number of transitions while they are scanned into the scan chain and whose responses also cause minimal number of transitions while they are scanned out of the scan chain, we focus on minimizing the number of transitions caused only by test patterns that are scanned in. Even though we focus on minimizing the number of transitions caused only by test patterns, our extensive experiments show that the proposed TPG can still reduce switching activity significantly during BRISK. Since circuit responses typically have higher correlation among neighborhood scan outputs than test patterns, responses cause fewer transitions than test patterns while being scanned out. A transition at the input of the scan chain at scan shift cycle, which is caused by scanning in a value that is opposite to the value that was scanned in at the previous scan shift cycle, continuously causes transitions at scan inputs while the value travels through the scan chain for the following scan shift cycles. Describes scanning a scan test pattern 01100 into a scan chain that has five scan flip-flops. Since a 0 is scanned into the scan chain at time =0, the 1 that is scanned into the scan chain at time =1 causes a transition at the input of the scan chain and continuously causes transitions at the scan flip-flops it passes through until it arrives at its final destination at time =5. In contrast, the 1 that is scanned into the scan chain at the next cycle =2 causes no transition at the input of the scan.

4. PROPOSED SYSTEM

4.1 FULLY OPERATIONAL GENERATOR

Much higher flexibility in forming low-toggling test patterns can be achieved by deploying a scheme presented. Essentially, while preserving the operational principles of the basic solution, this approach splits up a shifting period of every test pattern into a sequence of alternating hold and toggle intervals. To move the generator back and forth between these two states, we use a T-type flip-flop that switches whenever there is a 1 on its data input. If it is set to 0, the generator enters the hold period with all latches temporarily disabled regardless of the control register content. This is accomplished by placing AND gates on the control register outputs to allow freezing of all phase shifter inputs. This property can be crucial in SOC designs where only a single scan chain crosses a given core, and its abnormal toggling may cause locally unacceptable heat dissipation that can only be reduced due to temporary hold periods. If the T flip-flop is set to 1 (the toggle period), then the latches enabled through the control register can pass test data moving from the PRPG to the scan chains. Two additional parameters kept in 4-bit Hold and Toggle registers determine how long the entire generator remains either in the hold mode or in the toggle mode, respectively. To terminate either mode, a 1 must occur on the T flip-flop input. This weighted pseudorandom signal is produced in manner similar to that of weighted logic used to feed the shift register. The T flip-flop controls also four 2-input multiplexers routing data from the Toggle and Hold registers.

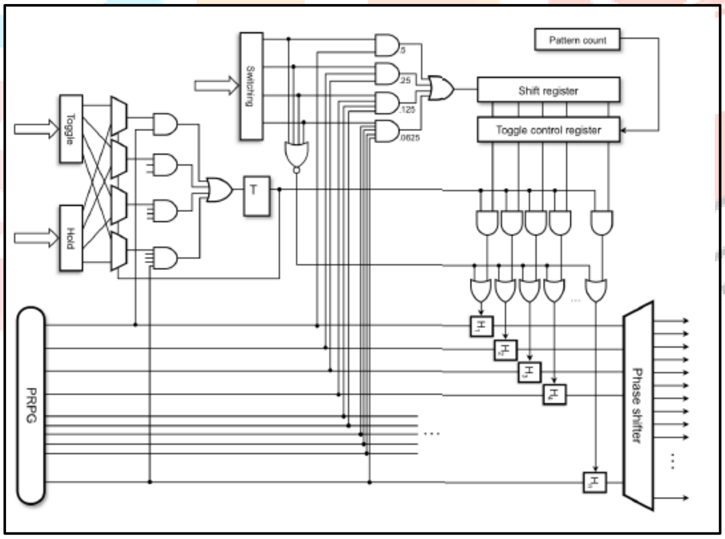


FIG. 3. FULLY OPERATIONAL VERSION OF PRESTO

It allows selecting a source of control data that will be used in the next cycle to possibly change the operational mode of the generator. For example, when in the toggle mode, the input multiplexers observe the Toggle register. Once the weighted logic outputs 1, the flip-flop toggles, and as a result all hold latches freeze in the last recorded state. They will remain in this state until another 1 occurs on the weighted logic output. The random Occurrence of this event is now related to the content of the Hold register, which determines when to terminate the hold mode.

Scan switching profile when deploying the PRESTO generator in a hypothetical environment with 15 scan chains is for two test patterns. Blue (0s) and red (1s) stripes make up the low power-toggling pattern, while gray areas correspond to periods of toggling. All-blue and all-rescan chains are fed by the constant values only. Note that their quantity does not change between patterns though they are not exactly the same in each case. As can be seen, test patterns are divided into hold and toggle intervals of random length, while LP scan chains remain still for the entire duration of single test pattern. When using the PRESTO generator with an existing DFT flow, all LP registers are either loaded once per test or every test pattern. The registers loaded only once act as test data registers or are parts of an IJTAG network, and are initialized by the test setup procedure. They are triggered using a slows can shift clock and operate at a very low speed thereby imposing no timing constraints. Although the remaining registers are loaded once per test pattern (also at the scan shift speed), timing is not compromised because of shallow logic generating bits to be loaded serially into the registers. With the help of shadow registers, values remain unchanged during capture. Clearly, it suits LBRISK applications, where the shift speeds are quite high. The LP registers are also added during embedded deterministic test (EDT) IP generation and insertion. The associated logic is integrated into the design along with the EDT logic. Since the EDT logic (including LP) is only added in the scan paths, there is no impact on the functional mode of operation.

4.2 PSEUDO-RANDOM TESTABILITY OF THE CIRCUITS

A low area overhead and good speed of the designed BRISK strictly depend on the nature of the circuit, for which the BRISK is being designed. Pseudo-random testability of a particular circuit strictly depends on the number of hard-to-detect faults. It is possible to apply an unmodified sequence of LFSR code words to fully test some circuits in a reasonable number of cycles, while some other circuits are particularly untreatable by this way. We have studied the pseudo-random testability of the ISCAS and ITC'99 benchmarks, using standard LFSRs. All the benchmarks were in their full-scan versions, thus turned into combinational. Each benchmark was tested 1000 times using different LFSR polynomials and seeds. Both the polynomials and the seeds were randomly generated, while a satisfactory period length was censured by a simulation of the PRPG run. The number of LFSR bits was set to be equal to the number of CUT inputs. The results of a simulation of a selected set of benchmarks. The "I" column shows the number of the benchmark inputs (including the scan path for sequential circuits), "range" indicates the range of the encountered number of test patterns to fully test the circuit (in those 1000 samples), while the statistical average value is shown in the last column. "K" stands for thousands of patterns, "M" for millions, "G" for billions. For some benchmarks the range has not been evaluated, for an extremely large number of patterns needed to fully test the circuit (more than 10 M). It can be seen that the number of pseudo-random patterns needed to fully test the circuits varies considerably. The distribution of the number of required patterns follows the curve. This particular curve corresponds to the ISCAS c1908 circuit.

4.3 INFLUENCE OF LFSR TYPE ON TEST LENGTH

An LFSR used as a pseudo-random pattern generator is mostly based on the primitive generating polynomial to provide the longest period of the code generated. In this Subsection we show that it is not necessary to use primitive polynomials. We investigated the influence of the number of LFSR taps on the testing capability. In particular, we studied the number of patterns needed to test all the faults in a circuit (like in Subsection 3.1), while varying the number and the position of the LFSR taps. A satisfactory period for each generated LFSR was ensured by simulating its run. Here the number of LFSR cycles needed to cover all the stuck-at faults in the c1355 circuit is shown. 100 different LFSRs were generated randomly for each LFSR size, differing both in the tap positions and the seed. Thus, for the circuit used (having 40 inputs) 3900 different LFSRs were produced. LFSRs 0-99 correspond to 1-tap LFSRs, 100-199 correspond to 2-tap LFSRs, and so on. It can be observed that the number of taps does not influence the fault coverage capability at all; the test lengths are steadily distributed. Thus, we can conclude that the most advantageous LFSR is one of the 1-tap LFSRs, since its area overhead is the smallest (only one feedback). A 1-tap LFSR having a satisfactory period can be found in most cases. The use of primitive polynomials thus becomes counterproductive, since the number of taps is mostly greater than one here, and they do not make any contribution while the number of covered faults was recorded. It can be observed that 90% of the faults were covered in the first 1000 cycles, while 60 000 cycles were needed to achieve a complete fault coverage. Thus, it is advantageous to apply a relatively small number of pseudo-random patterns to cover the easy-to-detect faults, and then produce several deterministic patterns to cover the rest. This approach is called a mixed-mode BRISK. It is necessary to find a trade-off between the number of pseudo-random and deterministic patterns. Moreover, the PRPG has to be properly chosen, since the fault coverage differs notably with PRPG type, as we have shown in Subsection 3.1. The probability of covering a given number of faults by a PRPG. Here, sets of 10, 50, 100, 500, 1000 and 5000 LFSR patterns were applied to the c3540 circuit, 10 000 samples for each test size. The seed and the tap position were selected randomly for each sample. The distribution of the number of faults that remained undetected is shown here. For a low number of patterns many faults are left undetected, and their number also varies considerably. As the number of test patterns increases, the number of undetected faults rapidly decreases, while the standard deviation of this number decreases as well.

4.4 COLUMN-MATCHING BRISK

` The column-matching BRISK method is based on a transformation of the PRPG code words into deterministic test patterns pre-computed by some ATPG tool. This transformation is being done by a combinational block, called Output Decoder. The method is designed for combinational or sequential full-scan circuits, thus the order of the test patterns applied to the CUT is insignificant. Moreover, not all the PRPG patterns have to be transformed into test patterns; the excessive ones just do not test any new faults. In our column-matching method we try to assign the PRPG code words to the deterministic patterns, so that some of the columns are equal. Then the decoding logic needed to implement the matched column will be reduced to a mere wire connecting the decoder output with its respective input. The unmatched outputs have to be synthesized by some Boolean minimize

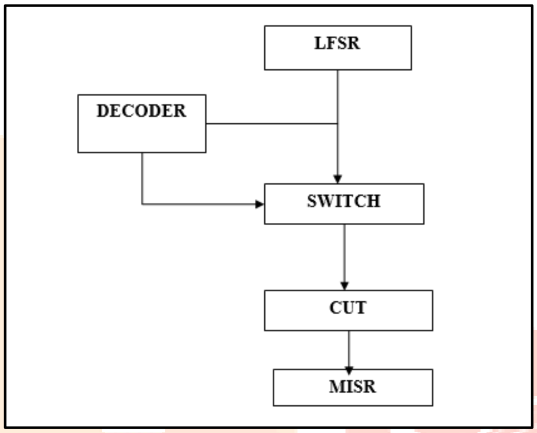


FIG .4. MIXED-MODE BRISK STRUCTURE

For a more detailed description. This principle has been further extended to support mixed-mode testing. The BRISK run is divided into two disjoint phases. First, the circuit is tested using an unmodified sequence of LFSR code words, detecting the easy-to-detect faults. The deterministic test patterns for the rest of the faults are computed by the Atlanta ATPG tool. These vectors are to be generated by several consecutive LFSR code words and modified by the Decoder to obtain deterministic vectors. There has to be some additional logic to control the switch between the two phases. The switch is implemented as an array of multiplexers, one for each CUT input. However, we attempt to eliminate the MUXas well, by introducing direct matches. The structure of a mixed-mode BRISK. The sequence of patterns is fed to the tested circuit and its response is then evaluated by a multi-input shift register MISR.

4.5 WEIGHTED PATTERN TESTING

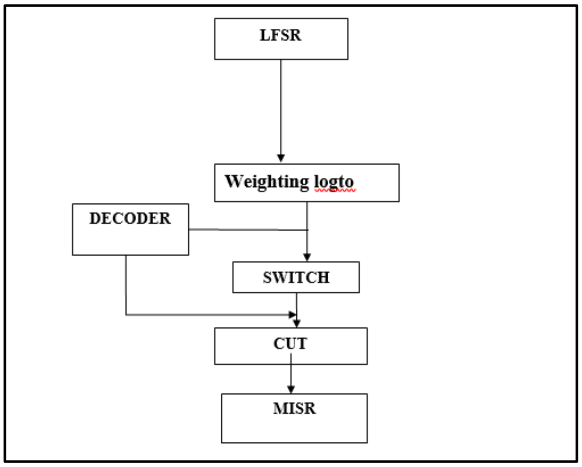


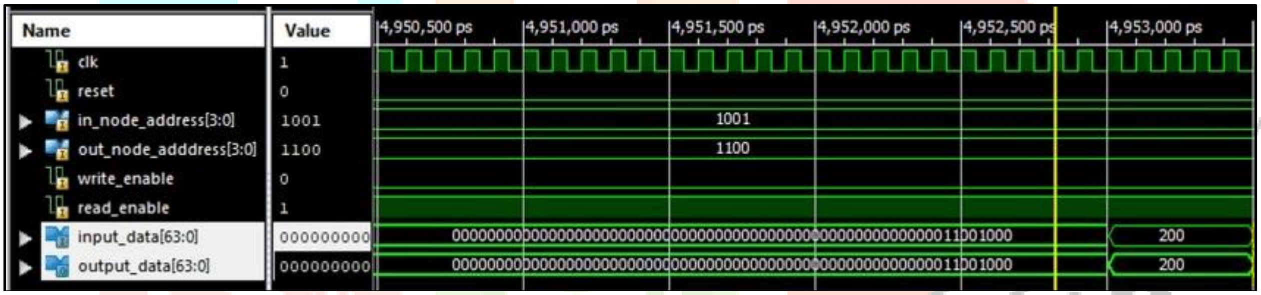
FIG 5: WEIGHTED COLUMN-MATCHING BRISK SCHEME

When the weighted pattern testing is used, a new block is introduced into the BRISK design the weighting logic block. The LFSR width (r) may be less than the number of CUT inputs (m), the number of TPG outputs is increased just by the weighting logic block. In practice, an LFSR with a random polynomial and random seed is used. This ensures (to some extent) a uniform distribution of '1's and '0's in code words produced by it. In other words, the weights of all the r LFSR outputs are approximately 0.5. To generate the weighted patterns, the outputs of the weighting logic are generated by AND-in or OR-in randomly selected LFSR outputs. Thus, m weighted PRPG outputs are generated by this way.

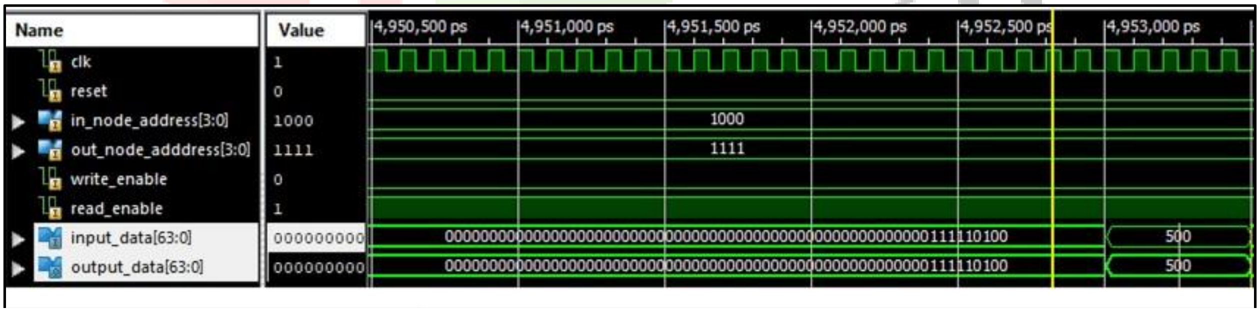
4.6 NON-WEIGHTED TESTING

Another possibility how to reduce the LFSR width is to use a splitter. No weighting logic is involved here, however we need to synthesize an m-output pseudorandom pattern generator, to feed the CUT in the pseudorandom phase. The simplest way how to do this is to split the LFSR outputs among the PRPG outputs, thus feeding several CUT inputs by one LFSR output. This, of course, reduces the pseudorandom testing capabilities. On the other hand, a significant area overhead reduction is obtained by eliminating the weighting logic (and, more importantly, the number of LFSR flip-flops). The LFSR width reduction has a significant influence on the deterministic phase as well: since there are less decoder inputs, the column-matching process is sped up (linearly with the number of inputs, see Section 2). The negative influence of a reduction of the LFSR input is an increase of the area overhead of the decoder. This is caused by a decrease of the number of possibilities for a column match (there are n m column matches in total). However, this disadvantage is fully compensated by the LFSR area reduction.

5. RESULT



Name	Value
clk	1
reset	0
in_node_address[3:0]	1001
out_node_adddress[3:0]	1100
write_enable	0
read_enable	1
input_data[63:0]	000000000
output_data[63:0]	000000000



Name	Value
clk	1
reset	0
in_node_address[3:0]	1000
out_node_adddress[3:0]	1111
write_enable	0
read_enable	1
input_data[63:0]	000000000
output_data[63:0]	000000000

6.CONCLUSION

Discussed the influence of the pseudo-random pattern generator type on its fault detection capability. Both LFSRs and CA are studied, with either a random or a "special" seed. The distribution of weights on the individual PRPG outputs is shown for all cases, together with the fault coverage curves obtained by the PRPGs. PRESTO the LP generator can produce pseudorandom test patterns with scan shift-in switching activity precisely selected through automated programming. The same features can be used to control the generator, so that the resultant test vectors can either yield a desired fault coverage faster than the conventional pseudorandom patterns while still reducing toggling rates down to desired levels, or they can offer visibly higher coverage numbers if run for comparable test times. This LP PRPG is also capable of acting as a fully functional test data decompressor with the ability to control scan shift-in switching activity through the process of encoding. The proposed hybrid solution allows one to efficiently combine test compression with logic BRISK, where both techniques can work synergistically to deliver high quality test. It is therefore a very attractive LP test scheme that allows for trading-off test coverage, pattern counts, and toggling rates in a very flex

7. REFERENCES

[1] A. S. Abu-Issa and S. F. Quigley, "Bit-swapping LFSR for low-power BRISK," *Electron. Lett.*, vol. 44, no. 6, pp. 401–402, Mar. 2008.

[2] C. Barnhart *et al.*, "Extending OPMISR beyond 10x scan test efficiency," *IEEE Design Test*, vol. 19, no. 5, pp. 65–73, Sep./Oct. 2002.

[3] S. Bhunia, H. Mahmoodi, D. Ghosh, S. Mukhopadhyay, and K. Roy, "Low-power scan design using first-level supply gating," *IEEE Trans.Very Large Scale Integr. (VLSI) Syst.*, vol. 13, no. 3, pp. 384–395, Mar. 2005.

[4] M. Chatterjee and D. K. Pradham, "A novel pattern generator for nearperfect fault-coverage," in *Proc. 13th IEEE Very Large Scale Integr. (VTSI) Test Symp.*, Apr./May 1995, pp. 417–425.

[5] F. Corno, M. Rebaudengo, M. S. Reorda, G. Squillero, and M. Violante, "Low power BRISK via non-linear hybrid cellular automata," in *Proc.18th IEEE Very Large Scale Integr. (VTSI) Test Symp.*, May 2000, pp. 29–34.

[6] D. Das and N. A. Touba, "Reducing test data volume using external/ LBRISK hybrid test patterns," in *Proc. Int. Test Conf. (ITC)*, 2000, pp. 115–122.

[7] R. Dorsch and H. Wunderlich, "Tailoring ATPG for embedded testing," in *Proc. Int. Test Conf. (ITC)*, 2001, pp. 530–537.

[8] M. Filipek *et al.*, "Low power decompressor and PRPG with constant value broadcast," in *Proc. 20th Asian Test Symp. (ATS)*, Nov. 2011, pp. 84–89.

[9] S. Gerstendorfer and H. Wunderlich, "Minimized power consumption for scan-based BRISK," in *Proc. Int. Test Conf. (ITC)*, 1999, pp. 77–84.

[10] V. Gherman, H. Wunderlich, H. Vranken, F. Hapke, M. Wittke, and M. Garbers, "Efficient pattern mapping for deterministic logic BRISK," in *Proc. Int. Test Conf. (ITC)*, Oct. 2004, pp. 48–56.