Fault-Tolerance: New Trends for Digital Circuits

José Miguel Vieira dos Santos
FEUP/ ISEP- I. S. Engenharia do Porto
R. de S. Tomé - 4200 Porto - PORTUGAL
jmvs@dee.isep.ipp.pt

José Manuel Martins Ferreira
FEUP- Universidade do Porto
Rua dos Bragas 4050 Porto - PORTUGAL
jmf@fe.up.pt

Abstract:
Fault-Tolerance usually means expensive designs. The solution proposed allows fault detection in cost effective designs, with a small overhead, and may also correct some output patterns providing a Discrete Fault-Tolerance.

Keywords: *Failure Detection, Concurrent test, Fault-Tolerance, BST.*

1- Introduction

The objective of Dependable systems [1] is to avoid failures, mainly through Design Prevention and Fault-Tolerance (**FT**). Design Prevention includes all the measures to avoid bugs during the project phase and circuit implementation. VLSI technology has improved this feature by reducing IC number and pin counting, and earlier debug of delay problems. But people involved with FT usually believes that "Murphy was an optimist" and even the best design may fail, which requires additional measures.

Fault-Tolerance aims to prevent the error to become critical, but the hardware overhead required (and associated cost) is high, restricting its use. The low cost of today's digital technology has allowed circuits to spread into a wide range of applications, some of them *critical-but-not-life-critical*, and in many cases subject to aggressive environments increasing failure rates. Then low cost FT solutions may have a wide range of interest.

The Boundary Scan Test (BST) infrastructure, defined in the IEEE 1149.1 std [2], places cells in all signal pins and required internal nodes, allowing to observe them off-line. The enhancement now developed allows BST self-sycronization to circuit logic and an effective way to verify concurrently expected patterns, under the control of a small BST controller. The ability to inject output signals allows to bypass a defective internal logic, and provides a discrete FT with low overhead to the mission circuit, low cost and flexibility.

The paper follows with a review of FT methodologies, BST overview and previous work references, and the new solution.

2- Background

The concept of FT in digital systems, introduced by von Newman "to build reliable circuits from unreliable components", is based on redundancy to *detect* or to *correct* errors [3]. FT circuits are also able to hide many failure modes hardening their own test and debug process, which may lead to fault accumulation [4,5]. These extra costs have confined FT to life critical systems, where the consequences of a failure are important enough to justify them.

Failure causes may be classified according to *physical*, *logical* and *time* manifestation [6]. A *failure* occurs when the system deviates from its specification, as a result of an *error* in the finite state machine (**FSM**) data, induced by *faults* in the FSM hardware . Faults are a consequence of *defects*, which may have an internal or external origin. Errors may also be a result of man-made software faults. Concerning the time aspect, a fault may be *permanent* or *temporary*, in which case it may still be *intermittent* or *transient*. Temporary faults are accepted to represent 90-99% (application dependent) of all faults. In sequential circuits, errors become *latent* until they have effect on the system outputs or are overwritten.

Most FT systems are designed to face *single fault* models (which is acceptable in some cases but not in others, as design errors and production defects, usually multiple), assuming that faults occur one at a time, and a fault is detected and corrected before the occurrence of a second one [7]. Preventive measures may give an extra level of confidence, helping the above assumptions. Typically they include careful logic design and simulation, electrical and time safety margins and *Design Diversity* to reduce the probability of undetected bugs. Being of passive nature, they have a limited cost and little or no impact on system performance.
If the fault Model only considers temporary faults, the detection process may be merged into the software (sw redundancy) but the recovery time is higher.

0-7803-5008-1/98/$10.00©1998 IEEE.

By definition a FT system has the ability to survive to faults, according to 2 figures, *Availability* and *Dependability,* usually attained through 2 error treatment levels:
Error detection: the low cost solution, usually based on Error Detecting Codes (**EDC**), may only provide a Fail-Stop system. Self-Checking (**SC**) circuits, where outputs are coded and their properties verified concurrently by Checkers, are the most representative. The checkers must satisfy some properties to detect their own faults, to allow for the Totally Self-Checking (**TSC**) goal.
Correct errors: an error may be *corrected after detection* or *masked* through massive redundancy. The ultimate goal is masking, but the redundancy required (replication and decision) makes these architectures very expensive. With no error detection (it is not necessary here), masking designs become vulnerable to fault accumulation. The usual solutions are:
- Duplex system: build up with two base (simplex) circuits, output disagreement implies to disable the system until the fault is located through error detection mechanisms.
- Triple Modular Redundant (**TMR**) systems with a majority voter and output in accordance with at least two modules, provide immediate decisions for single fault correction and belong to the generic group of NMR architectures.

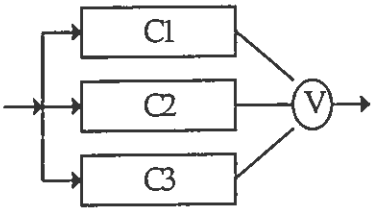

Fig.01- TMR architecture

- Error Correcting Codes (**ECC**) which may be seen as a special kind of TMR designs.
- Fail-Safe (**FS**) circuits with outputs classified in two groups, the safe and the non-safe code-words. In FS systems the idea is to provide the correct output or a safe output. The FS architecture, with no intrinsic error detection, has been used with good results in circuits with standard components but is not reliable enough for VLSI common mode faults.

3- BST and Previous work

The BST infrastructure is a standard for structural and functional off-line test, with many compliant ICs provided by major manufacturers, and ASIC libraries give to design engineers an easy access to this infrastructure. The BST Test Access Port (TAP) state machine, controlled by TMS and TCK signals, enables the selection of the Instruction (IR) or Data Registers (DR).

DR's may be mandatory (Bypass and Boundary-Scan) or optional (IDentification and Design Specific). Relevant test points and usually all primary I/O pins, with Boundary Scan cells (BSc) become accessible through TDI/ TDO (Test Data In/Out) lines, in a shifting process.

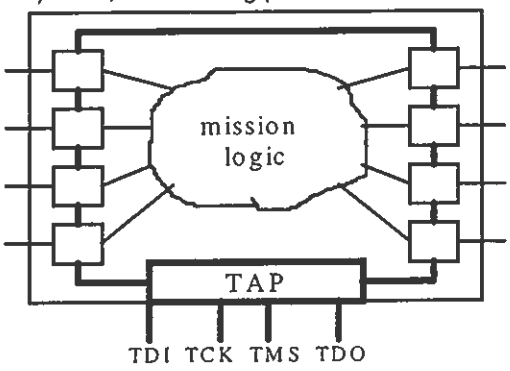

Fig.02- The BST TAP and scan cells

The std includes mechanisms for preventing many faults in TAP connections to affect the system. IC producers are able to insert native BST chains with almost no overhead, and user BST usually lies in the 5-10% overhead range.
From the specification designers will build a circuit, that must be tested (for structural defects) and debugged (looking for design errors). FT designs require in addition the verification of their FT capabilities, and their ability to hide faults may allow them to accumulate, which is not compatible with the single fault model. This implies most of the times the need to test/debug each sub-module of the FT system independently, and BST may play here an important role, an additional reason to include it in FT designs. Required for structural tests, the infrastructure may be expected to help improve functional verification on-line. However its off-line nature is really inefficient in on-line applications, requiring solutions to provide BST synchronisation to system logic.

Previous work relating BST and FT is scarce and usually applies BST for testing the circuits disabled from normal operation, or as a way to bring internal checker signal to the external. Some interesting examples are:
The C-BIST concurrent testing technique presented by Saluja *et al* [8], assumes test completion which may not occur and depends on common defects. Cheng and Agrawal [9] refer partial scan designs to break synchronous sequential circuits, improving testability. McHugh and Whetsel [10] suggested Parity and an additional pin for Interrupts, to increase instruction reliability and allow to the BST controller a fast way to know internal scan errors. Wagner and Williams [11] showed that functional testing may be enhanced by concurrent sampling, and suggest the BST.

The B2UBIST architecture for Self-Checking modules, using BST for OFL detection of single faults and checker analysis [12], is an evolution of the UBIST methodology, allowing to reduce SC coding requirements. The checkers are monitored through the TAP, but this approach is not entirely compatible to the IEEE 1149.1-std.
Chackraborty [13] and Kuntzsch et al [14] work with Duplex modules, referring isolation and synchronisation problems.

4- Concurrent Scan Test

4.1- Previous considerations

The FTCS22 "State-of-the-Practise in Fault-Tolerance" work-group [15] recommends that *FT modules be designed according to standard methodologies, so that they can be reused in other applications, reducing design cost and allowing a justified exhaustive debug and test, to obtain highly dependable modules.*
On the other hand the overhead of the FT solution must be minimum, as it impacts the circuit MTBF. Experience shows that the error latency interval (delay between fault occurrence and error detection) and the redundant detection overhead required walk in reverse directions: if the application accepts a small latency interval the hardware overhead may be reduced.
A Partial Design Diversity (PDD) design may then be a solution, and allow systems to survive in a *Grateful Degradation* mode, with a lower bound of operation, when the application does not justify the cost of replication as for most earth based systems (automotive, train and industrial control).

These were the main ideas in our work, together with BST reuse on-line, directing the solution to reduce the impact of the overhead on the MTBF and time delay. In this environment, BST will be mainly user defined, directed for ASICs and FPGAs where replication has few interest. Our search was focused on a PDD solution joining the advantages of a low cost partial approach, to the reliability of having the mission circuit operation verified through an independent process.

4.2- Architecture

Concurrent Scan Test (**CST**) is a powerful evolution of *POST* originally presented in [16]. The basic idea is to store the best available set of deterministic test patterns into the BST controller and compare them with the actual operation Test Vectors (**VT**) through the BS cells. The enhancement developed, fully IEEE 1149.1 compatible, allows this operation to be transparent to the circuit, with no impact at all on the mission circuit performance, and can be applied to IC or PCB levels.

Each circuit may be verified as a single or multiple Functional Block (**FB**) according to Fig.03 and 04, providing a concurrent functional verification through the scan chain.

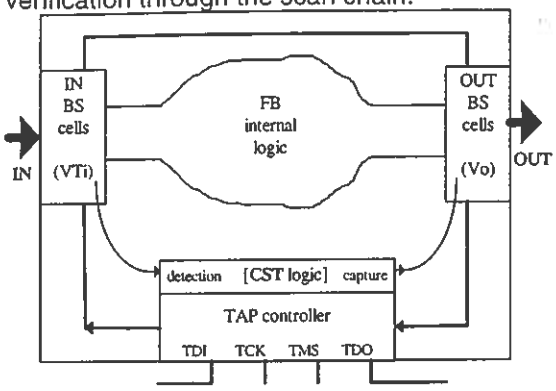

Fig.03- FB general view

CST is able to search simultaneously several combinations (32 or more) of the single input vector (VTi). The resident BST-controller (**BSμC**) is derived from the one presented in [17], redesigned and optimised for concurrent operation. With 28 pins, the internal ROM (1-4KB) releases pins for 16 TMS lines to provide independent TAP access for each application FB's, also provides the ability to run tests concurrently and may also perform structural tests during power-up or when required.

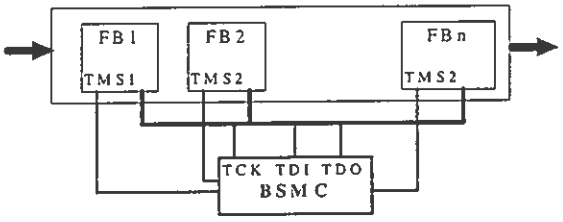

Fig.04- Independent TAP control

The RISC-like software (6 instructions only) may be equal for most applications, and only the VT need to be programmed for each design, into the ROM. Derived from the test patterns defined for functional verification (debug), VTs are now reused justifying the cost to search for the best deterministic set. A wide use and mass production will allow a complete debug, high reliability and low cost BSμC.

CST has 2 important features:
- *low overhead*: CST adds only 10-15% (depending on the type of cells used) to the std BST. Having in mind that BST overhead usually stays below 10%, CST means about 1% additional hardware to the mission circuit, now supervised concurrently with virtually no impact on the MTBF.
- *Independence*: the nature of CST allows the mission circuit to be freely designed and optimized, providing its highest reliability.

Functional simulation will define nodes required to be verified, and scan cells will be inserted accordingly, without the need to redesign the mission circuit. From here two things result:
- the mission circuit may work alone and be tested without the BSµC. This is also valid if the BSµC fails and is disabled by a Watchdog.
- the number of VT is not increased, contrary to most other FT solutions. This will speed up the concurrent verification process.
These features are really unusual with traditional FT solutions, and besides the BSµC may learn some VT's on the field, and store a trace of detected failure situations.

CST doesn't work by random capture; VT's stay into the input BS cells for the time they will statistically show up. This and simultaneous multiple search improves input matches. The analysis of CST operation shows that the number of VT statistically verified is compatible with the number of VT required for typical designs and the latency intervals usually accepted. The chart below shows the number of VTs statistically verified in 10ms for each FB. **F** is the number of FBs and **n** is the average number of inputs of all FBs. The *detection Time* Td_C and Td_N lines are derived for Constant and Normal input patterns distributions, and design solutions should fit into their left region.

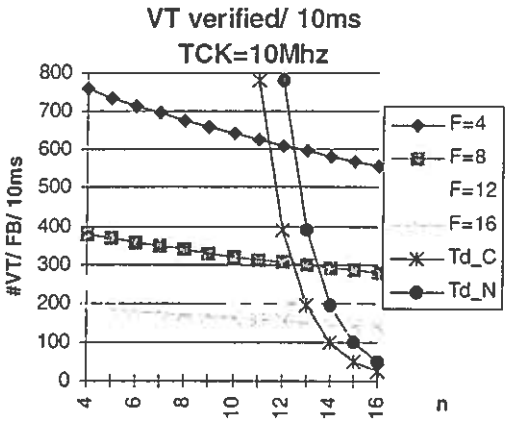

Fig.05- Number of VT verified

Besides the supervision ability CST allows pre-shifted VTo to be placed at the FB outputs, when the input triggering VTi=Vi condition happens, which may be used for Fault-Injection debug purposes and Discrete replacement of a defective FB to provide a Graceful Degradation mode of operation, one VTi/VTo pair at a time. This discrete Fault-Tolerance may allow a step above the Fail-stop solution, and be used as a second FT mechanism in most critical FT designs. CST will provide a failure warning, but repair may be delayed.

Conclusion

Enhancing the BST infrastructure capabilities, a low overhead solution provides failure detection in cost effective designs, and may allow systems to continue working in the presence of faults, a **survival** mode with a fraction of traditional solutions cost. After circuit design, optimisation and FB definition, the ATPG process and scan cells routing can be completely automatic, up to the final BSµC ROM programming.
Sequential logic is the major limitation and new solutions are being addressed. For the moment sequential designs with CST are required to be Duplex: hardware comparison of outputs confines the error immediately, and CST may disable the wrong circuit.

REFERENCES

1 - J-C Laprie, "Dependable Computing and Fault Tolerance: Concepts and Terminology", IEEE FTCS15 Digest of Papers, pp2-11, 1985.
2- *IEEE Standard 1149.1 Test Access Port and Boundary-Scan Architecture*, IEEE Inc, NY, 1990.
3- Parag K. Lala, *Fault Tolerant & Fault Testable HARDWARE Design*, Prentice/Hall International, 1985.
4-M.Nicolaidis,"Shorts in Self-Checking Circuits",*JETTA* 1, pp. 257-73, 1991.
5-C.Thibeault, Savaria,.Houle,"Test Quality Hierarchical Defect-Tolerant ICs," *JETTA*, 3, pp. 93-102, 1992.
6 - Santos, JMV, "F-T: How Interesting in Industrial Equipment", ISIE'97, Student Forum, pp7-11, 1997.
7 - B. Courtois, "Failure Mechanisms, Fault Hypothesis and Analytical Testing of LSI-NMOS Circuits", VLSI Conference, pp.341-50, 1981.
8 - K.K.Saluja, R.Sharma, C.R. Kime "A Concurrent Testing Technique for Digital Crcuits", IEEE Trans. on CAD, V.7, N°12, pp.1250-60, Dec 1988
9 - K-T Cheng and V.D. Agrawal, "An Economical Scan Design for Sequential Logic Test Generation", *FTCS-19 D. of Papers*, IEEE C. Society Press, 1989, pp28-35.
10- C. M. Maunder and R. E. Tulloss, editors. *The Test Access Port and B-S Architecture*, IEEE, 1990, Chpt. 20 by: P.F. McHugh and L. Whetsel, pp 205-213.
11- K. D. Wagner and T. W. Williams, "Enhancing Board Functional Self-Test by Concurrent Sampling", IEEE *Proc.of ITC 1991*, pp. 633-40.
12-M. Lubaszewski, B.Courtois,"On the design of SC BS Boards", *Proc.of ITC*,1992, IEEE, p.372-81.
13 -T.J.Chakraborty, On-line Test Method Using BS, *3rd IEEE IOLTW*, Crete, Greece, 1997, pp156-9.
14 - C. Kuntzsch et al, "A Novel Approach for an On-line Selftest Arch. using ASIC Circuits in a Multi-Channel System", *3rd IEEE IOLTW*, Crete, Greece, 1997, pp165-8.
15- International Simposium on FT Computing, digest of papers, IEEE Computer Society Press, 1992.
16- Santos J.M.V, Ferreira J.M.M, "POST: Pseudo On-line Boundary Scan Failure DeTection", *3rd IEEE Int'l On-line Testing Workshop*, Greece, July 1997.
17- Ferreira,JM, Pinto,FS and Matos, JS, "A Modular Architecture for Board-Level BIST of BS Boards", *Proc. of the EuroASIC Conference*, June 1992.