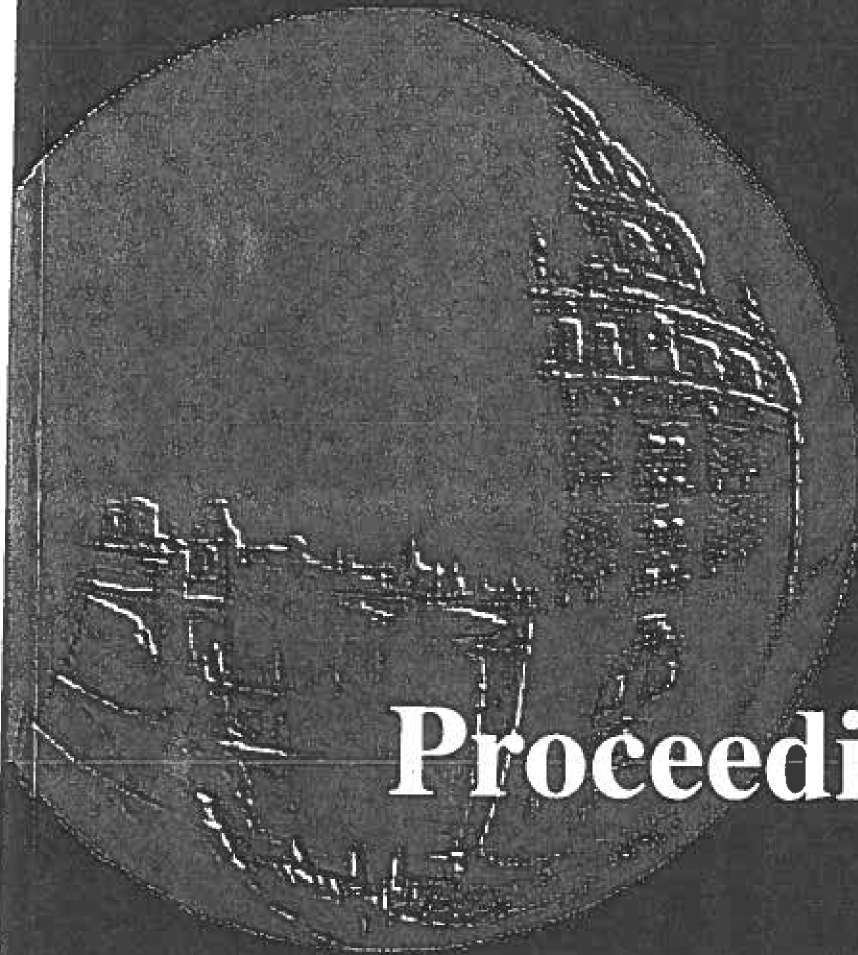
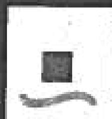
DCIS'97

XII Design of Circuits and Integrated Systems Conference



Proceedings



Escuela Superior de Ingenieros.

Sevilla, November 18-21, 1997

DCIS '97

The *POST* Approach to On-line Failure Detection Based on BST

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:

An enhanced BST infrastructure, fully IEEE 1149.1 std compatible, enables on-line detection of pre-defined circuit conditions, opening the way to real-time monitoring Fault-Tolerant designs. A restricted Fault-Tolerance capability is also possible, with POST ability to inject a set of corrected output values.

Keywords: On-line Test, BST, Fault-Tolerance, DFT.

1- Introduction

The Boundary Scan Test (**BST**) technology was defined in the IEEE 1149.1 **std** [1], and is now widely accepted for Off-Line (**OFL**) tests. ASIC libraries and CAD tools give to design and test engineers a quick path to this powerful infrastructure, and many ICs are available with BST.

In Fault-Tolerant (**FT**) systems [2,3,4] reliability is the prime concern. These *"systems with the ability to survive to faults"*[5], (are built to) exhibit a high capability to hide many types of *failure modes*, and therefore require additional testability solutions. As the BST infrastructure will be present in an increasing number of circuits for structural tests, it may be expected to help improve *failure detection* during circuit operation. However having an OFL nature, the BST structure is really inefficient in On-Line (**ONL**) operation, and solutions providing *BST-to-system logic synchronisation* are required. An additional reason to recommend enhancements in the BST infrastructure, if significant benefits are within sight, is that pin-limited ICs predominate over core limited ICs.

The solution proposed POST (*Pseudo On-line Scan-based failure deTection*) is a *partial design diversity* approach to FT, involving:

a) an enhancement to the BST infrastructure allowing expected conditions to be detected *without disturbing normal system operation*;

b) a technique to verify ONL In-Out circuit relations; traditional designs (FT or not) may be supervised.

c) Ability to *inject* some corrected outputs; POST may, alone, extend the FT concept to low cost designs.

The paper follows with a background review (2), POST presentation (3) and failure detection analysis (4). Fault-Tolerance features (5) and final remarks (6) conclude it.

2-Background

The two major traditional FT architectures, *Static* and *Dynamic*, usually assume *single fault models* [6,7,8].

Static or **Masking** FT is widely based upon *Triple Modular Redundant* with Voting (**TMR**) architectures or on *Error Correcting Codes (***ECC***)*, and the errors are *masked* with the help of the redundant information. Fault *detection is* unnecessary in these schemes, but to avoid *fault latency* the circuits must be (periodically) checked.

In *Dynamic* or **Reconfiguration** schemes, one of the spare replicated *hardware* (**HW**) Functional Blocks (**FB**) available, replaces the active FB if it fails. Fault detection is necessary here, continuously or at "reasonable" time intervals, according to the *error confinement* delay acceptable in the application.

Additional detection problems still arise from the fault type, which may be *permanent* or *temporary*, and in this case *intermittent* or *transient*. Transient faults are usually the most difficult type to detect.

Most practical FT approaches presently available still have some problems. TMR and *Fail-Safe* circuits, with no error detection, become weak in VLSI, where production defects are usually multiple and common mode failures in replicated structures have a high probability of simultaneous occurrence. *Self-Checking* circuits need coding techniques and *Checkers*, meaning that the base system has to be redesigned. A time interval to allow all the input vectors to be exercised, is also supposed so that faults may be safely detected.

Previous work relating BST applied to failure detection areas, to our knowledge, can be summarised as follows: a proprietary solution to automatic event detection and capturing in a BST environment referred by Whetsel [9], addresses off-chip implementation and different objectives, therefore leading to higher cost in terms of PCB area and to larger design cycles. Its usefulness for ONL failure detection is also handicapped by the triggering and capture methods, mainly directed for debug or stop-and-go modes. A concurrent testing technique in a BIST environment was proposed by Saluja [10], but not 1149.1 compliant and depending on the completion of the test (a real handicap); internal test vector storage makes the solution potentially dependent on production and common mode defects.

The FIBS technique proposed by Chau [11] allows BST based fault injection, providing independent s@ (*stuck-at*) and *s-open* pin control, but ONL synchronisation is however not supported, which greatly restricts the usefulness of this approach for practical FT purposes. B²UBIST for Self-Checking boards, using BST for OFL detection of simple faults and concurrent checker analysis was also introduced recently by Lubaszewski and Courtois [12], simplifying coding requirements but is not 1149.1 compliant.
The most recent designs using BST for ONL failure detection assume modules with (two) replicated circuits, one of them being disconnected and tested OFL, while the other does its job. Problems to disconnect and synchronise both circuits make the solution not compatible with the 1149.1 std [13]; other solution uses replicated ASICS exchanging checked information between them, which requires additional lines, has synchronisation problems and provides no protection against common mode defects [14].

3- POST: A General View

3.1-Definitions and Terminology

Fault-Tolerance always means additional cost because some redundancy is necessary, and an on-board dedicated BST-controller (**BSµC**) may therefore be considered, such as the one presented in [15]. It is a dedicated ASIC (2000 gates and 28 pins) with 16 TMS lines, allowing independent FB TAP access and faster scan solutions compared to a single BST chain. TCK may be any submultiple of the BSµC CLK, and TCK ≤ CLK ≤ 33MHz. This CLK and the system CLK may be equal or different. Providing independence of the main circuit and fast scan operations in ONL conditions, it may additionally perform OFL tests during power-up or when required.
Four basic levels are considered in our methodology, beginning with the circuit, designed according to the specification and optimised. Then:
FB- Functional Blocks are identified by testability or functional criteria, to be BS/POST (almost) completely testable. A FB may be one (or more) integrated circuit (IC), and multiple-FB ICs are possible using internal scan chains. In practice some FB I/Os may be *not relevant* for a given FT problem, and this is exploited to achieve speed improvements.
CHAIN- the testable circuit, with FBs in series, parallel or interleaved. *Clusters* in the Chain may be tested off-line and as referred in 6.1.
MODULE- usually it will be a single CHAIN and a BSµC dedicated to the FBs, but for FT purposes it may have two or more, redundant Chains. Physically a Module is considered to be a board (PCB).
SYSTEM- the top level, delivering the service stated by the project specification. Usually built up of Modules, a simple System may be a single Module.

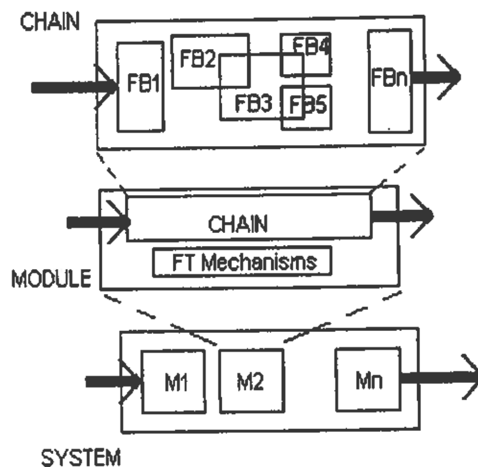


Fig.01- Hierarchical design relations

FBs are defined to allow for testability concepts (C&O) in the Chain. Modules, expected to be error self-contained, add the FT concept.
Off-Line tests can take place using a single BST chain. This means long Test Vectors (**VT**), complex to generate, store, analyse, and reducing test speed. Internal PROM releases BSµC pins for independent TMS lines, to provide parallel TAP access to the necessary FBs for each application, leading to powerful and faster solutions. The other TAP signals TCK, TDI and TDO are tied in parallel.

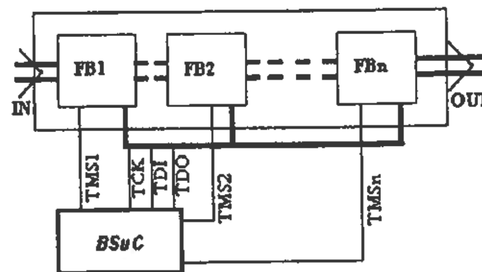


Fig.02- Independent control of FB TAPs in the Chain

For each **FB** we have (Fig.03):
Vi= input Vector, arriving at the FB inputs.
Vo= output Vector, captured at the FB outputs.
VTi= input Test Vector, expected.
VTo= output Test Vector, expected as the result of VTi.

fc- fault coverage of each VTi+VTo pair, or set of pairs. A fault is, for our purposes, any kind of *defect*, permanent or temporary, in the FB. On-line failure detection deals with *defect coverage*, but being more difficult to quantify we shall refer fault coverage.

3.2- FB level operation
Given a working FB, we look fo
- pick-up a VTi from the set stor
- shift VTi into the BST chain of
- wait for a similar Vi to appear
- **detect** a VTi=Vi condition and
- compare Vo with a stored, exp
POST is an 1149.1-compatit
requirements, and optimum
suggest two arrangements:

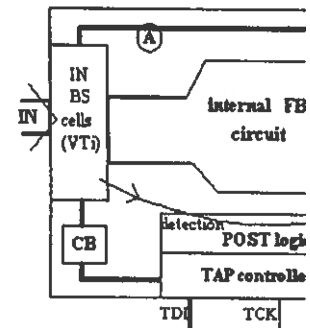


Fig.03- FB general view, with

a) If IN and OUT BS cells are
number of required TCK cycles
of VTi or Vo vectors. b) BS ce
placed in location A need not be

The Capture Bit (**CB**) is a sim
with no pin connection (*valid u*
for OFL tests simply means one
to activate POST, CB resets wit
BS sample or if loaded w
"*transparent*" to the BST infrastu

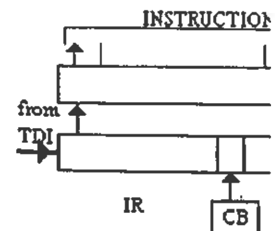
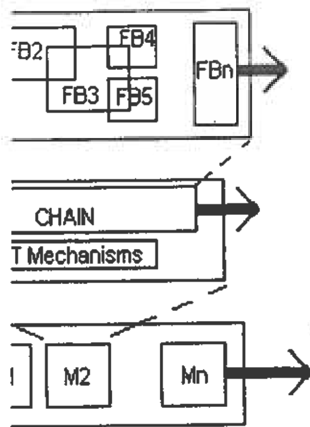


Fig.04- Reading CB throu

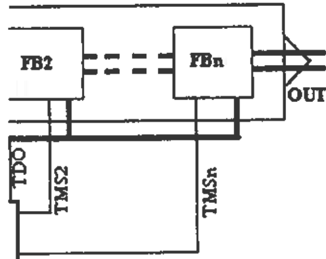
The mandatory **Sample/Preloa**
read CB, load VTi and read Vo.
through IR cycles, is made avai
IR bits, and the S/P mode is ma
capture. Only the first Vo is ca
VTi=Vi condition happens betw

POST relies on a careful anal
machine time diagrams and c
compatibility to the std is maint
a) IN and OUT BScs store
changed and *no delay is introdu*



erarchical design relations

ıllow for testability concepts (C&O) dules, expected to be error self- T concept.
ake p[illegible] using a single BST chain. Test [illegible]ectors (**VT**), complex to nalyse, and reducing test speed. leases BSμC pins for independent ovide parallel TAP access to the each application, leading to powerful . The other TAP signals TCK, TDI ı parallel.



ent control of FB TAPs in the Chain

ıve (Fig.03):
ırriving at the FB inputs.
, captured at the FB outputs.
ector [illegible]ected.
Vecto.., expected as the result of VTi.

of each VTi+VTo pair, or set of pairs. our purposes, any kind of *defect*, nporary, in the FB. On-line failure ith *defect coverage*, but being more y we shall refer fault coverage.

3.2- FB level operation

Given a working FB, we look for the capability to:
- pick-up a VTi from the set stored in the BSμC ROM,
- shift VTi into the BST chain of the FB,
- wait for a similar Vi to appear at the FB inputs,
- detect a VTi=Vi condition and capture the Vo,
- compare Vo with a stored, expected, VTo.

POST is an 1149.1-compatible solution to these requirements, and optimum operating conditions *suggest* two arrangements:

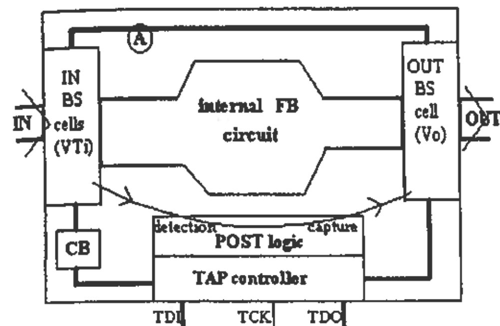


Fig.03- FB general view, with CB and POST logic

a) If IN and OUT BS cells are grouped as shown, the number of required TCK cycles is defined by the largest of VTi or Vo vectors. b) BS cells of pins not relevant placed in location A need not be shifted in or out.

The Capture Bit (**CB**) is a simplified BS shadow cell with no pin connection (*valid under std point 10.1.1.g, for OFL tests simply means one more 0*). Set with VTi to activate POST, CB resets with every detection, every BS sample or if loaded with 0, making POST "*transparent*" to the BST infrastructure.

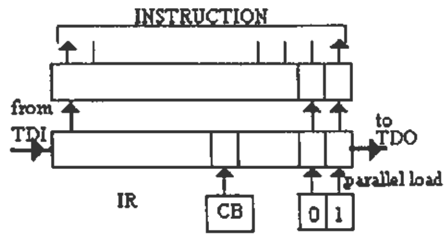


Fig.04- Reading CB through an IR operation

The mandatory **Sample/Preload** (S/P) mode is used to read CB, load VTi and read Vo. The CB bit value, read through IR cycles, is made available in one of the free IR bits, and the S/P mode is maintained when waiting a capture. Only the first Vo is captured if more than one VTi=Vi condition happens between each BSμC access.

POST relies on a careful analysis of the TAP state machine time diagrams and control signals, so that compatibility to the **std** is maintained, and:
a) IN and OUT BScs store the bits without being changed and *no delay is introduced in the signal path*.
b) POST was developed with the *simplified* BScs, providing observability-only, which is enough for ONL tests. With standard cells, additional capabilities become available (see 6.a).
c) Detection may happen at any moment, so special precautions were needed to ensure a stable CB information, provided by POST for FB inputs *synchronous* or *asynchronous* to the CLK (and TCK).
d) POST logic avoids destruction of the captured Vo, when a DR-read operation is initiated to fetch it.

The simplified fluxogram presented is easy to follow.

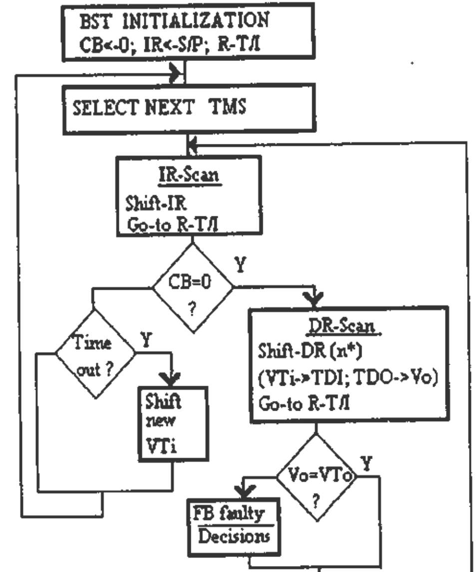


Fig.05- Simplified POST fluxogram.

Notice:
S/P: Sample/ Preload instruction (std).
R-T/I: Run-Test/ Idle state (std).
n= number of bits in the largest group (VTi, Vo).*
An estimated delay is allowed to detect each VTi. Waiting a capture, the BSμC may verify if the captured Vo equals VTo, and proceed with a new FB analysis.

3.3- Module level detection

In point 3.2 the detection and capture mechanism at FB level was presented. To analyse the behaviour of POST at Module level we shall consider a single-Chain comprising *F* FBs with independent TMS signals provided by the BSμC. Define **n** as the number of bits (relevant to POST) of the largest test vector VTi or VTo, for all FBs. The BSμC shifts in the VTi's and polls the FBs to search for Vi=VTi matches.
The optimal detection (and capture) rate occurs when a new match takes place in each FB during each polling cycle. To optimise capture rate it is necessary to consider F, n, the number of *TCK cycles* required by the BSμC and the *average time* for an input match.

Writing and solving the equations relating these four parameters, we can design the chart below, with the number of TCK cycles as a function of F and n. The leftmost line represents the *polling frequency (fp)* in KHz, related to TCK, and represented here for the maximum TCK frequency of the BSμC, 33 MHz.
The analysis above gives the worst case (lower) polling frequency because **n** is taken for all FBs. Regarding practical designs all the VTi/VTo bit number will be known, and so the exact solution may be computed.

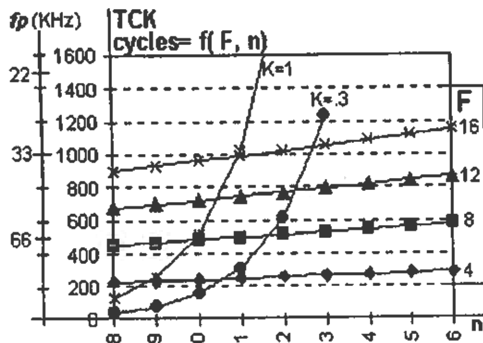


Fig.06- TCK cycles as a f(F,n)

This information allows to find the best solution for each circuit. If a Chain needs F=8 FBs, and no more than n=10 bit VTs, then each polling cycle will required about 500 TCK cycles, and <u>each FB</u> will be inspected at about 66KHz (if F=12, n<15, the fp will be near 40KHz).
In the equations referred above was included an additional *"probability factor"* K∈[0, 1], representing those Vi that are really required to be verified. The worst case is for K=1, if there is no information about Vi frequency distribution or if all Vi have the same probability of existence. The corresponding curve, divides the chart in 2 regions, the left one with probability of capture greater than 50%; as such the solutions must fit in this region. Nevertheless the right region may give faster solutions if a restricted number of VT pairs provides a high *fc*.
When the application allows an offset-based search (as is the case for most physical systems) or if the inputs change slowly, the K factor may be lower than 1; the left area enlarges, and increasing F or n the average fault latency interval may be reduced. The curve when only 30% of the possible Vi need to be verified, is represented. This restricted set is not necessarily a limitation to the *fc*: as the relation Vi-Vo is verified (and not only Vo), the most usual vectors will be tested faster this way, and Vo verified in operating conditions, providing a safety mode. Many real-world cases stay between the above figures, and in slow systems, POST may capture VTi+Vo and verify them.

4 -POST and failure detection

POST is ON-Line in a *discrete* (not continuous) way, which explains the *Pseudo ON-Line* expression. Detection and Capture are really ONL, but faults may only be detected by the BSμC after reading Vo. Now we claim that:

a) *If the output of a FB is updated at intervals ΔTou (Time output update, constant or variable), and*
b) *If a number N of VTi, with a fc =X%, is verified between two consecutive output updates,* then:
c) ***X will be the probability of a fault to be detected, generating no erroneous FB output,*** or:
c') ***the probability of the output to be error-free is X%.***
Proof: *the first part of assertion b) is accepted valid in Off-Line Tests, by nature. During useful and normal life of digital systems, faults are accepted to happen slowly and one at a time, many orders of magnitude slower than ΔTou, meaning that conclusion c) must be accepted if assertion b) is verified.*◊

Many examples of real-time systems may be found in the literature [16], from basic system control, elevator, automotive, train, up to plane and space craft control circuits, for which the acceptable fault latency interval usually lies in the 10-100ms interval. According to the figures in point 3.3, something between 500 and 10.000 VTi/VTo pairs may be expected to be verified, for each FB during the acceptable fault latency intervals, providing an efficient failure detection mechanism.
In general purpose circuits, the number of FBs in the Chain must be defined so that the verification time is shorter than the acceptable error confinement delay, defined according to the application requirements.

4.2- VTi Selection

The VTi/VTo pairs are a sub-set of the simulation functional test vectors. Automatic VTi+VTo selection tools are being developed, according to the following general rules:
a) estimate α_i, the *probability of existence* of each Vi, in normal operation; if not known: $\forall_i$, α_i=1.
b) calculate β_i, the *fc* for each Vi+Vo pair,
c) order VTi by their *efficiency coefficient* $\gamma_i = \alpha_i * \beta_i$
d) select the VTi for the *verification rate* desired: the N Vi with higher γ_i
e) Estimate the global *fc*. In some cases a fc=100% may be obtained with M<N vectors, the depth of the circuit being the dominant factor.
f) If a VTi is not detected, the BSμC must replace it after an interval related to the average delay for an input match, otherwise POST rate will degrade.

POST efficiency will be optimised if VTi checking restarts from the top of the above list after each output update, or at cyclic intervals when justified.

5- Fault-Tolerance featur

We have seen in point 3
mechanism. But there is anc
detection of an error (Vo≠VTo)
switch for the *VTo injection*
VTi+VTo are shifted in, and wl
condition occurs (VTi=Vi), VTc
outputs o the FB.
Two immediate applications:
a) ***System self-synchronised***
debug purposes, avoiding additic
Anticipated coded informatio
processor, allows FB outputs to
speed, with the real ICs and wi
prototypes for debug and validati
b) ***Discrete replacement of a def***
Uploading selected VTi/VTo pai
allows the VTo to be available
expected, bypassing the FB inten
The BSμC may enter in a FT *s*
Vi and searching the correct VT
one) to be injected. The Chai
Graceful Degradation mode of o
pair at a time. Software impleme
mode are named *imprecise comp*
to provide in (safe) time, if no
least a valid and safe approach
control logic may still provide
shuttle service between entrance
restaurant (*the origin of POST*
printer logic will still print slowl
only. In both cases a failure wa
but repair may be delayed.

6- Strengths, Limitations a

Some enhancements to POST ar
search process must always be d
level, otherwise the BSμC will
fault coverage will decrease. As
on reading Vi and search VTi/V
have a restricted practical interes
enhancements:
a) In cases where not all the bits
be defined, using the standard IN
will be available to store a ma
VTi=Vi match may be controlled
b) If the BSμC is not required t
can not be grouped, POST may
read. VTi and VTO will be shifte
BSc groups, with detection capab
the BSμC may follow the operati
c) POST may be used as *Pse*
OUT BScs are needed, and the B
verify the captured Vo. A new ins
bypass other cells, increasing the
d) VTi detection may be gr
qualitative information, or i
information is passed to the

ilure detection

in a *discrete* (not continuous) way, the *Pseudo ON-Line* expression. iture are really ONL, but faults may y the BSμC after reading Vo. Now

f a FB is updated at intervals ΔTou 'e, constant or variable), and of VTi, with a fc =X%, is verified cutive output updates, then:
robability of a fault to be detected, neous FB output, or:
of the output to be error-free is X%.
t of assertion b) is accepted valid in nature. During useful and normal life faults are accepted to happen slowly ;, many orders of magnitude slower ning that conclusion c) must be on b) is verified.◊

f real-time systems may be found in from basic system control, elevator, up to plane and space craft control the acceptable fault latency interval 10-1 s interval. According to the 3, something between 500 and 10.000 y be expected to be verified, for each acceptable fault latency intervals, ent failure detection mechanism.
e circuits, the number of FBs in the fined so that the verification time is acceptable error confinement delay, to the application requirements.

irs are a sub-set of the simulation ctors. Automatic VTi+VTo selection eveloped, according to the following

e *probability of existence* of each Vi, n; if not known: $\forall_i$, α_i=1.
e *fc* for each Vi+Vo pair,
eir *efficiency coefficient* $\gamma_i = \alpha_i * \beta_i$
for the *verification rate* desired: the N

obal *fc*. In some cases a fc=100% may M<N vectors, the depth of the circuit it factor
t det , the BSμC must replace it elated to the average delay for an input POST rate will degrade.

will be optimised if VTi checking top of the above list after each output c intervals when justified.

5- Fault-Tolerance features of POST

We have seen in point 3 the failure detection mechanism. But there is another possibility: upon detection of an error (Vo≠VTo) the POST structure may switch for the *VTo injection mode*. In this mode VTi+VTo are shifted in, and when the triggering input condition occurs (VTi=Vi), VTo will be injected at the outputs o the FB.
Two immediate applications:
a) ***System self-synchronised*** *Fault-Injection* for FT debug purposes, avoiding additional HW fault injectors. Anticipated coded information from the system processor, allows FB outputs to be changed, at circuit speed, with the real ICs and without the need to build prototypes for debug and validation of FT designs.
b) ***Discrete replacement of a defective FB***
Uploading selected VTi/VTo pairs into BS/POST cells, allows the VTo to be available at the FB outputs when expected, bypassing the FB internal logic.
The BSμC may enter in a FT *survival* routine, reading Vi and searching the correct VTo (or the best approach one) to be injected. The Chain may still provide a *Graceful Degradation* mode of operation, one VTi/VTo pair at a time. Software implementations of this survival mode are named *imprecise computations* [17], and aim to provide in (safe) time, if not the correct output at least a valid and safe approach. A defective elevator control logic may still provide a continuous, blind, shuttle service between entrance level and the last floor restaurant (*the origin of POST idea*), or a defective printer logic will still print slowly and a single font type only. In both cases a failure warning will be reported, but repair may be delayed.

6- Strengths, Limitations and Research

Some enhancements to POST are possible, but the VT search process must always be done at BS/POST cells level, otherwise the BSμC will be overcharged, and fault coverage will decrease. As such, solutions based on reading Vi and search VTi/VTo in memory, seem to have a restricted practical interest. Some of the possible enhancements:
a) In cases where not all the bits of VTi have to, or can, be defined, using the standard IN BScs the R2 flip-flop will be available to store a mask bit, and the input VTi=Vi match may be controlled at bit level.
b) If the BSμC is not required to read Vo, or the cells can not be grouped, POST may be used with *no Vo read*. VTi and VTO will be shifted into the IN and OUT BSc groups, with detection capability. With 2 CB cells, the BSμC may follow the operation.
c) POST may be used as *Pseudo-CHECKER*. Only OUT BScs are needed, and the BSμC may continuously verify the captured Vo. A new instruction is desirable to bypass other cells, increasing the scan speed.
d) VTi detection may be greatly improved with qualitative information, or if anticipated coded information is passed to the BSμC by the main processor, allowing a VTi to be loaded just before it is expected to occur.

6.1- Strengths of POST

POST has two ways to detect errors (and defects), the basic one being output mismatch (VTo≠Vo), which provides detection of defects internal to the FB. But each FB Input group may be seen as a kind of checker to the previous part of the Chain, and so the BSμC may also look for an unreasonably low number of input matches, which may be a result of defects in the circuit behind the FB. This may be applied to clusters.
Other strengths of POST are:
a) *Independence:* POST has *no impact on the circuit performance*. The Chain may be designed and tested without the BSμC, to be added later. The same is valid if the BSμC fails and a watchdog disables it.
b) *Easy of design and test*: the Chain may be designed as a whole, and no coding methods are required. POST BSc will then be routed through desired (FT relevant) test points; so the *number of vectors to test the FB is not increased* by POST. The increased testability provided by the FBs allows the number of VT required to test the Chain/circuit to be lower than with a global solution, bringing non negligible speed improvements.
c) *Upgrading:* If operating conditions change or a better set of VTi is defined, by deeper simulation or monitoring field operation, the BSμC ROM may be reprogrammed.
d) *Retry:* in order to deal with temporary faults, POST may be allowed to retry the detection of a some errors.
e) *Discrete Fault-Tolerance* provided by VTo injection.
f) *Field Report:* the BSμC may keep a trace of all detected failure situations, stored in memory, which may be really helpful for field engineers.
g) According to the type of BS cells used, POST means a 10-15% overhead to the std BST. So POST will probably mean about 1% overhead to the circuit, excluding the BSμC, obviously external.

6.2- Limitations of POST:

a) *Single VTi search at a time*
As the scan cells can store a single bit, the resulting average fault-latency will be a lower bound of a multiple VTi search, but this seems not to be compatible with the 1149.1 std. Nevertheless the number of VTi searched at a given time is equal to the number of FBs in the Chain.
b) POST and Sequential circuits
In its actual stage, POST is interesting mainly for combinational circuits. Sequential circuits, mainly their depth, restrict the usefulness of POST to FBs with 2 or 3 Flip-flops. However some characteristics seem to be promising for sequential designs, such as the possibility of masking some VTi bits and the BSμC ability to accept more than one VTo for each VTi. As Pseudo-Checker, POST may deal with complex sequential blocks, watching the outputs only.

6.3- Future Research

a) Alternative implementations to reduce the average fault latency, and enhancements for sequential circuits are being considered.

b) Can we envisage the use of the BSμC as a VOTER, in a Module with 2 replicated Chains? Will the Module behave as a TMR-like system, with POST helping disagreement decisions? Is the VTi/Vto memory storage still required?

c) What is the behavior of a TMR Module supervised by POST, if the first failed Chain is kept in a degraded operation mode. Is the Module able to sustain a second Chain failure without degrading performance.

d) How do BS cells influence the FB/ Chain reliability? The figures known point to a 3-5% increase in the failure rate, but this problem and BS cells protection against circuit injury needs further analysis.

CONCLUSION

Technology enhancements and cost reduction has allowed Fault-Tolerance features in medium-cost designs, some of them referred as "*critical but not life critical*" systems. Errors and faults must be detected as soon as possible, but not with the requirements of life-critical designs, usually error masking or fail-stop mechanisms, meaning additional resources to provide the high redundancy required ("*Murphy was an optimist*"), but reducing circuit reliability also.

POST allows a BST enhanced infrastructure to verify on-line the circuit I/O operation, *without disturbing system operation* and with a *fault-coverage* and a restricted FT as corollary. With a good cost/benefit relation, fully 1149.1 compliant and requiring no additional instructions, POST may expand BST applications into FT systems design. Actual test speed is being determined with respect to benchmarks, and results concerning fault latency, area overhead and scan insertion timing will be presented in a future paper.

ACKNOWLEDGEMENTS

We would like to thank Gustavo Alves for his helpful suggestions in writing this paper.

REFERENCES:

1- *IEEE Standard 1149.1 Test Access Port and Boundary-Scan Architecture*, IEEE Inc, NY, 1990.

2- Parag K. Lala, *Fault Tolerant & Fault Testable HW Design*, Prentice/Hall International, 1985.

3- Victor P. Nelson & Bill D. Carroll, *Tutorial: Fault-Tolerant Computing*, IEEE Computer Society, 1987.

4- D.P.Siewiorek,"Architecture of Fault -Tolerant Computers: An Historical Perspective", *Proc. of IEEE*, V.79, no.12, Dec.1991.

5- A. Avizienis, "Fault-Tolerance: the survival attribute of digital systems," *Proceedings of the IEEE V66*, N 10, pp. 1109-25,1978.

6-M.Nicolaidis,"Shorts in Self-Checking Circuits",*JETTA* 1,pp. 257-73,1991.

7-C. E.Stroud, A.E.Barbour, "Testability and Test Generation for Major Voting F-T Circuits," *JETTA*, 4, pp. 201-14, 1993.

8-C.Thibeault, Y.Savaria, J-L.Houle,"Test Quality Hierarchical Defect-Tolerant Integrated Circuits," *JETTA* 3, pp. 93-102,1992.

9- Lee Whetsel, "Event Qualification: a Gateway to At-Speed Testing," *Proceedings of the ITC* 90, pp. 135-41.

10- 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.

11-Savio Chau, "Fault Injection Boundary Scan Design for Verification of F-T Systems," *Proc. of the ITC*, pp. 677-82, 1994.

12-M. Lubaszewski, B.Courtois,"On the design of Self-Checking Boundary Scannable Boards", *Proc.of ITC*,1992, IEEE, pp.372-81.

13 -T.J.Chakraborty, On-line Test Method Using BS, 3rd IEEE IOLTW, Crete, Greece, 1997, pp156-9.

14 - C. Kuntzsch, F.Mayer, K.Ronge, A Novel Approach for an On-Line Selftest Architecture using ASIC Circuits in a Multi-Channel System,3rd IEEE IOLTW, Crete, Greece, 1997, pp165-8.

15- Ferreira,JM,Pinto,FS and Matos, JS"A Modular Architecture for Board-Level BIST of BS Boards", *Proc. of the EuroASIC Conference*, June 1992.

16-International Simposium on FT Computing, Digest of Papers, IEEE Computer Society Press, Vols. 15-24.

17 - J.W.Liu, W-K Shih, K-J Lin, R. Bettati and J-Y Chung, "Imprecise Computations", Proc. Of the IEEE, Jan 1994, pp. 83-94.

High-Level Test S

J. C. Ferr
[1]INE
Pr
[2]FE

Abstract

Boundary-scan based test
1149.1 and P1149.4) can prov
mixed-signal board testing th
life cycle. The present paper
test specification and test pla
ronments. It describes a mixe
tion language and the associat
egy. The paper includes
examples.

1. Introduction

The growing number of
and their increasing complexit
the constant need for add
required by ever more deman
by new technological de
manufacturing, packaging an
difficulty of testing mixed-sig
accordingly and is adversel
market.

In this paper we consider
related to high-level mixed-
We assume the boards ar
boundary-scan based test bus
1149.1 standard [1], extended
nals through the use of spec
[2,3] and/or by the addition of
described in the P1149.9 st
This approach builds on the s
standard for digital systems, c
ferent phases of a product's li
the use of low cost test equipm

Several support ICs for
(BST) have been describe
[2,3,5,6] and some are ava
They range from relatively
improve the observability a
digital boards [5] to sophisti
[8]. BST-based test suppor
capabilities are described in [2

The testing of mixed-sign

A. C. Leão is currently w
Semicondutores, Vila de Cond