3rd IEEE International On-Line Testing Workshop

Sofitel Capsis Beach Resort, Aghia Pelaghia headland, Crete, Greece

July 7-9, 1997

Sponsored by

IEEE Computer Society

in cooperation with

Dept of Computer Eng. & Inform.
University of Patras
26 500 Patras, Greece
E-mail: nikolosd@cti.gr

Dept of Computer Science
Texas A&M University
College Station, TX 77843-3112
E-mail: pradhan@cs.tamu.edu

Reliable Int. Systems Group, TIMA
46 Av. Felix Viallet
38031 Grenoble, France
E-mail:michael.nicolaidis@imag.fr

NCSR «Demokritos»
Aghia Paraskevi
15310 Athens, Greece
E-mail:paschali@iit.nrcps.ariadne-t.gr

Failure Detection and Boundary Scan: a Pseudo On-line approach (POST)

JOSÉ MIGUEL L. VIEIRA dos SANTOS
ISEP- Instituto Superior de 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:
The Pseudo On-Line Boundary Scan Failure DeTection (POST) solution proposed, uses an enhanced BST infrastructure, fully IEEE 1149.1 compatible, to enable a restricted on-line detection of pre-defined circuit conditions and response capturing. Able to inject some corrected values, POST may improve Fault-Tolerant applications.
Keywords:Fault-Tolerance, On-Line Test, DFT, BST.

1- Introduction

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

In Fault-Tolerant (**FT**) systems [2,3,4] reliability is the prime concern. These *"systems with the ability to survive to faults"*[5], (are build 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 *fault detection* in normal system operation. However, having a OFL nature, the BST structure is really inefficient in On-Line (**ONL**) applications, and solutions providing *BST synchronisation to system logic* are required. An additional reason to recommend enhancements in the BST infrastructure, if significant benefits are within sight, is that the major limitation in present day VLSI is usually the package pin number and not the internal circuit density.

POST is a *partial design diversity* FT solution, involving:

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

b) a technique to verify In-Out circuit relations ONL.

c) Traditional FT designs may be supervised, but providing also *output vector injection*, POST may, alone, extend the FT concept to low cost designs.

The paper follows with a background review (2), POST presentation (3) and dynamic analysis (4). Output vector injection (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 usually 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**) or *software* (**SW**) Functional Blocks (**FB**) available, replaces the active FB if it fails. Fault detection is here necessary, 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. *Fail-Safe* circuits with no error detection become weak in VLSI, and *Self-Checking* circuits need coding techniques and *Checkers*, meaning that the base system needs 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 [6].

A proprietary solution to automatic event detection and capturing in a BST environment referred by Whetsel [9], addresses objectives different from POST and off-chip implementation, therefore leading to higher cost in terms of PCB area and to larger design cycles. A concurrent testing technique in a BIST environment was proposed by Saluja [10]. 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 FT purposes. B^2UBIST for Self-Checking boards, using BST for OFL detection of simple faults, and concurrent checker analysis, was also introduced recently [12].

3- POST: A General View

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 [13]. Providing *independence of the main system and ability to run fast tests ONL*, it may additionally perform structural tests during power-up.

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.

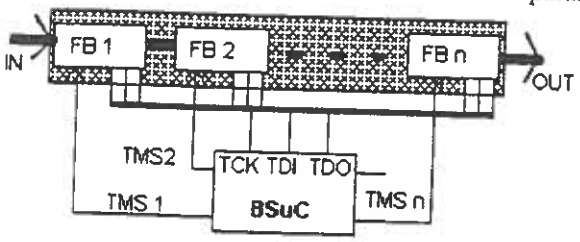

Fig.01- Independent control of FB TAPs in the Chain

3.1-Definitions and Terminology

The structures we consider in our methodology are:

BSμC- the on-board dedicated BST-controller, in a 28 pin package. It has near 2500 gates (memory not included) and 16 TMS lines allowing independent FB TAP access and faster scan solutions (2 may be used for hierarchical BST communication). TCK may be any submultiple of the BSμC CLK, and TCK ≤ CLK ≤ 33MHz. This CLK and the system CLK may be equal or different.

FB- a Functional Block, BST/POST (almost) completely testable, for which a set of test vectors (**VT**) can be generated at an acceptable cost/coverage ratio. 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 group of FBs, confined by Chain inputs and outputs, with no relation to the BST chains. In the fig.01 FBs are in series (concerning signal flow), but this is not mandatory. *Clusters* in the Chain may be tested off-line.

MODULE- usually it will be a single CHAIN and a BSμC dedicated to the FBs (not necessarily in series), 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, as seen by users, delivering the service stated by the project specification. Usually built up of Modules, a simple System may be a single Module.

FBs are defined in the Chain by testability criteria (C&O). Modules allow to add the FT concept.

For each **FB** we have (Fig.02):

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.

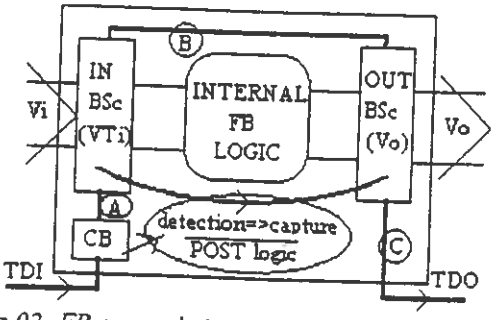

Fig.02- FB general view, with CB and POST logic

3.2- Given a working FB, we look for the capability to:

- *pick-up a VTi from the set pre-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 resulting 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.02):

a) If IN and OUT BScs 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, may be placed in locations A, C or preferably in B, where they need not to be shifted in, because their pre-loaded values have no effect.

3.3- Capture Bit (**CB**) is a simplified BS shadow cell with no pin connection, in the first position of the BS Register (*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 load with 0, making POST "*transparent*" to the BST infrastructure.

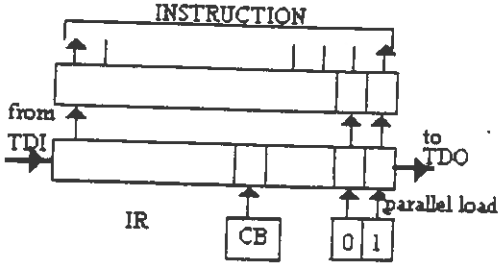

Fig.03- Reading CB through an IR operation

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. The simplified fluxogram presented is easy to follow.

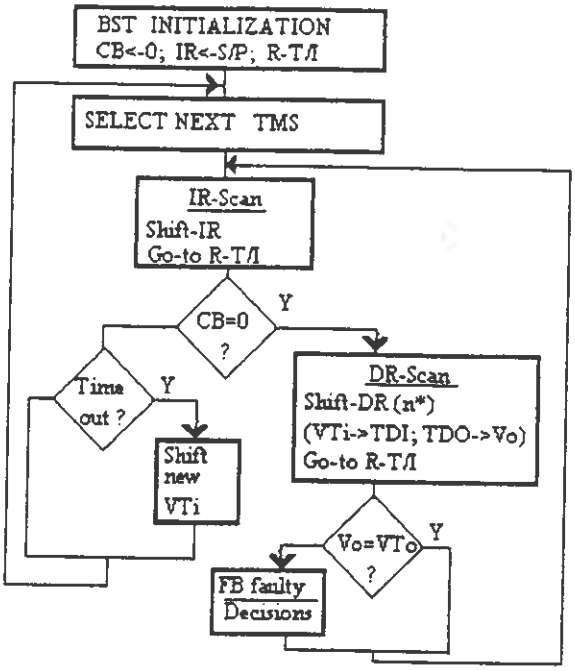

Fig.04- Simplified POST fluxogram.

Notice:
S/P: Sample/ Preload instruction (std).
R-T/I: Run-Test/ Idle state (std).
n= number of bits in the largest chain (VTi, Vo).*
A delay defined in 4.2. 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.

3.4- 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 (no R2 nor M2). With standard cells additional capabilities become available (6.1.a). According to the type of BS cells used, means a 10-15% overhead to the std BST HW.

c) Detection pulse will depend on the system itself; POST control logic has a short 3 gate delay between FB IN and OUT. A valid Vo must be captured and so an additional *delay* may be necessary.

d) 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).

e) POST logic avoids destruction of the captured Vo, when a DR-read operation is initiated to fetch it.

4- Dynamic Analysis

Timings will vary from system to system, but to estimate reasonable figures consider a MODULE, single CHAIN with 8 FB; each FB with *max[VTi,Vo]*=16 (16 In+16 Out cells allowed for POST, and only 16 TCK cycles needed for a VTi+Vo shift cycle). In this Chain 8*(16+16)=256 Test-points are supervised, not necessarily *primary inputs or outputs. Note that this limit respects only to the FB I/O cells relevant for FT POST needs!*

4.1-FB inspection time

Assuming 8-bit IRs (a usual value), and TCK= 33MHz, we have for each FB:
DR scan cycle =(2*5)+(3+16+2)= 31 TCK cycles = 0.94μs
IR scan cycle = (2*6)+(3+8+1) = 24 TCK cycles = 0.73μs
The above values are obtained as (X)+(Y), where:
X= number of TCK cycles needed by TAP state machine.
Y= number of the BSμC CLK cycles [13].
With 1μs per FB majoring other instructions average delay, each FB *inspection time* will be:
To read CB → **ΔTi**=0.73+1= 1.73μs
To read CB and load new VTi →ΔTi=1.73+0.94=2.67μs
The average ΔTi is 2.3μs, and with 8 FBs in the Chain each FB will be *inspected* every 18μs, corresponding to 55KHz (16KHz if TCK=10MHz).

4.2- VTi detection time.

For a 16 input FB, a maximum of 64K different Vi are possible. We shall consider the BSμC CLK equal to the system CLK, but they can be entirely different. Two cases may be considered in normal operation:

a) All Vi have a *similar probability of existence.*
Considering a clocked (synchronous) system, the maximum statistical interval for a Vi to appear, at a 33MHz CLK is:

ΔTm= 65536 VT/ 33 VT/μs= 1986μs ≈ 2ms

and the *mean* interval: **ΔTd**= (1/2)*ΔTm≈**1ms.**
This is the *average detection interval* that a VTi is expected to wait into the IN BScs *until detection.*

b) *The Vi have different probability of existence.*
Qualitative information about Vi frequency distribution may allow a faster test of a restricted vector set. Liquid level control, temperature, speed and several automotive functions are examples where "*the system spends 90% of the time in less than 10% of the possible input combinations*", a kind of "*digital quiescent range*". Additionally many systems have input combinations that may never appear. Considering a restricted range of 10% of all Vi: ΔTd≈1ms/10=**100μs.**
This restricted set is not necessarily a limitation to the *fc*; the most usual vectors will be tested faster, and Vo verified in normal operating conditions. Many real-world cases stay between the above figures, and in slow systems, POST may capture VTi+Vo and verify them continuously.

4.3-POST and Fault Detection

POST is not really an ONL solution, 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. Then POST is ON-Line in a *discrete* (not continuous) way. 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 [14], from basic system control, elevator, automotive, train, up to plane and space craft control circuits, with *output update rates* no faster than 10-20Hz, exceptionally 50HZ, and frequently in the Hz or sub-Hz range. Considering a *ΔTou*=50ms, the above *ΔTd* values allow POST to verify 50 and 500 VTi+VTo *pairs* respectively. If the verified pairs provide a *fc* of, say, *96%*, this is the probability of the output to be error free. In general, the Chain must be designed so that the VTi verification time is shorter than the acceptable error confinement delay (see 6.2.c).

4.4- VTi Selection

Some VTi are a sub-set of the simulation functional test vectors, and others may need to be determined. Automatic VTi+VTo generation 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: $\forall_i$, $\alpha_i = 1$.

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, where N=*ΔTou/ΔTd*.

e) Estimate the global *fc*. In some cases a fc=100% may be obtained with M<N vectors.

f) If a VTi is not detected, the BSμC must replace it after ΔTm, otherwise POST rate will degrade.

g) In cases 4.2.b, some "*not frequent*" Vi must be included, so that input excursion extreme cases may be supervised.

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- POST and Fault-Tolerance

An enhancement to POST, developed with 4 gates only, allows a pre-shifted VTo to be placed (*injected*) at the FB outputs, when the triggering VTi=Vi condition happens. 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, allow a word to be changed, at system speed, with the real ICs and without the need to build prototypes for FT debug.

b) ***Partial replacement of a pre-detected defective FB.*** Uploading selected VTi/VTo pairs into POST BScs, allow these VTo to be available at the FB outputs when expected, even with the FB internal logic disabled. The Chain may still provide a *Graceful Degradation* mode of operation, one VTi/VTo pair at a time.

This *partial Fault-Tolerance* may be extended to *more than one FB, and also to the chain as a whole.* 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*); a failure warning will be reported, but repair may be delayed.

6- FINAL REMARKS

6.1-Enhancements to POST

a) In cases where not all the bits of VTi have to, or can, be defined, with the standard IN BScs, the R2 flip-flop will be available to store a mask bit, previously shifted.

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.* IN and OUT BSc, with detection capability, reset 2 CB cells.

c) POST may be used as *Pseudo-CHECKER.* Only OUT BScs are needed; the BSμC may continuously verify captured Vo. A new instruction is desirable so that the other cells are bypassed.

6.2- Limitations of POST:

a) When a fault behind the FB disturbs Vi detection, the BSμC may still detect faults *internal to* the FB (Vo is not correct) and *external* if some Vi are not appearing as expected. This method may be applied to faults in Clusters.

b) 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.

c) Higher detection rates will be possible reducing the FB I/O number. Given TCK and the system CLK frequencies, and the desired *ΔTou*, the maximum recommended number of FB in/outputs to be supervised by POST may be determined, and taken in consideration.

d) POST looks for a single vector at a time in each FB, but it will be looking for as many vectors as the number of FB in the Chain.

6.3- Strengths of POST

a) *Independence:* POST has *no effect at all on the Chain performance*. The Chain may be designed and tested without the BSμC, to be added later. The same is valid the BSμC fails and a watchdog disables it.

b) *Easy of design and test*: the Chain may be designed as a whole and tested. 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.

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) POST may be used in several ways. The basic two are:

-Passive supervision: POST only signals the existence of detected faults, and eventually disables the output of the Chain in Modules with redundant schemes.

-Active supervision: FB output is updated only if POST doesn't disagree, and signal progression may be followed inside a Chain. Speed of operation is lowered but reliability is improved, and correction may be used.

e) *Retry:* in order to deal with temporary faults, POST may retry the detection of a detected fault.

f) *Partial Fault-Tolerance*: when VTo injection is used, to bypass a defective FB as referred above.

g) *Field Report:* the BSμC may keep a trace of all detected failure situations, stored in memory.

h) *Reliability*: mass production and extensive use may allow the BSμC to be a highly dependable IC.

6.4- POST and Sequential circuits

In its actual stage, POST is interesting mainly for combinational circuits. However some characteristics seem to be promising for sequential cases:

- FB may be a fictitious entity, circuits may be designed as a Chain, and POST cells will only supervise chosen parts.
- the possibility of masking some VTi bits and the BSμC ability to accept more than one VTo for each VTi, also helps to deal with sequential systems.
- working as Pseudo-Checker, POST may deal with complex sequential blocks, watching only the outputs.

6.5- Future Directions

a) How do BS cells influence the FB/ Chain reliability?

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? And with 2 BSμC, one for each Chain?

c) Since BST can disconnect FB outputs, how far can reconfiguration be exploited? And should it address all FB outputs or only those detected defective?

d) How effective POST can be, acting as the KILL area in reconfigurable hierarchical architectures [8]?

CONCLUSION

People involved with Fault-Tolerance usually believe that "*Murphy was an optimist*", and additional resources are inserted in traditional FT architectures, providing the required redundancy but also reducing global reliability. This may not necessarily be the best way to deal with FT in many "*critical but not life critical*" real-time systems. POST allows a BST enhanced infrastructure to verify FB I/O operation in a near ONL way, *without disturbing system operation* and with a *fault-coverage* and some FT as corollary. With a good cost/benefit relation, 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. 135-141.

10- K.K.Saluja, R.Sharma, C.R. Kime "A Concurrent Testing Technique for Digital Circuits", 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 SC Boundary Scannable Boards", *Proc.of ITC*,1992, IEEE, pp.372-81.

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

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