Quatrième colloque national du GDR SOC-SIP (System-On-Chip — System-In-Package)
mercredi 9 juin – vendredi 11 juin 2010
Ecole Nationale Supérieure de l'Electronique et de ses Applications, Cergy

Hardware Dependability: Fault Tolerance to the Rescue?

Jean Arlat, Jacques-Henri Collet and Yves Crouzet
{firstname.lastname}@laas.fr

Université de Toulouse

LAAS-CNRS

cnrs

Outline

- Introduction
  —> Emerging Paradigms in Hardware Technology
  —> Brief Historical Perspective on Fault Tolerance
- Yield Enhancement
- On-Line Error Handling
- Testing/Evaluation Issues (wrt Fault Tolerance Mechanisms)
- Concluding Remarks

Where Do We Stand?

The "More Moore" (Top-Down) Trend

- Process variations ↗
- Manufacturing (lithography, testing) costs ↗
- Yield ↘
- Prob. defects get undetected ↗
- Impact of defects ↗

- Frequency ↗, Power dissipation ↗
- Parameter variation ↗
- Power supply voltage ↘
- Soft Error Rate ↗

Crosscutting Challenge 5: Reliability & Resilience

- ***Relaxing the requirement of 100% correctness for devices and interconnects may dramatically reduce costs of manufacturing, verification, and test.***
  Such a paradigm shift will likely be forced in any case by technology scaling, which leads to more transient and permanent failures of signals, logic values, devices, and interconnects.

  Several example issues are as follows. (1) Below 65nm, single-event upsets (soft errors) impact field-level product reliability, not only for embedded memories, but for logic and latches as well. (2) Methods for accelerated lifetime testing (burn-in) become infeasible as supply voltages decrease (resulting in exponentially longer burn-in times); even power demands of burn-in ovens become overwhelming. (3) Atomic-scale effects can demand new "soft" defect criteria, such as for non-catastrophic gate oxide breakdown or highly resistive vias.

- ***In general, automatic insertion of robustness into the design will become a priority as systems become too large to be functionally tested at manufacturing exit.***
- ***Potential solutions include automatic introduction of redundant logic and on-chip reconfigurability for fault tolerance, development of adaptive and self-correcting or self-healing circuits, and software-based fault- tolerance.***

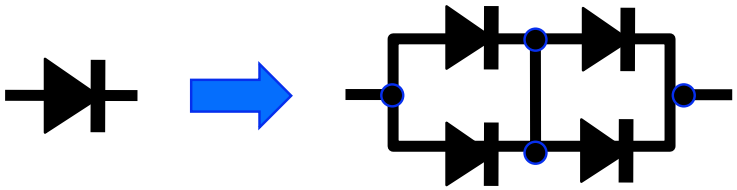
Source: International Technology Roadmap for Semiconductors, 2009 Edition — Design [http://www.itrs.net]

New Paradigms are Emerging

- Move away from the Basic “Frequency & Size” Rationales
- From “100% Correct” to “Less than Perfect” Circuits…
- Resilience via Incorporation of Redundancy Techniques to cope with Manufacturing Defects and Runtime Faults
- Memory: Static and On-line Degradable-Reconfigurable Circuits — Extensive application of ECC (Hamming, SEC-DED, Reed-Solomon, Turbo Codes, etc.)
- Processor: From “X-Scalar” to “Vectorial” Multi-Core Architectures, featuring “Natural” Reconfiguration Capabilities
- …

Incorporating Fault Tolerance Features
Level of Application of Redundancy?

- **1950's: Elementary Devices**
  [Moore&Shannon 56, VonNeuman 55]

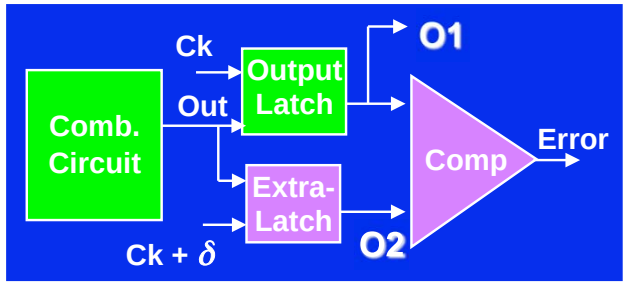


- **1960's-1970's:**
  - **Inter-Chip Level Level**
    IBM series, Bell, NASA,Raytheon, …
  - **Intra-Chip Level Functional Blocks**
    Self-checking μP [Crouzet & Landrault, FTCS 79]

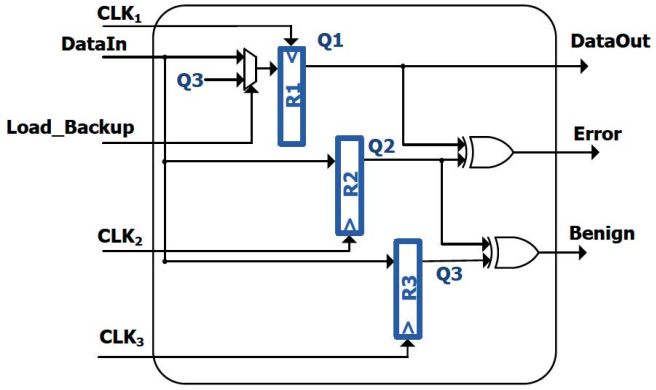
- **1980's: Inter-Chip Level Modular Redundancy**
  - **Intel iAPX 432 "Master-Checker Piggy-Packing Combination"**
  - **Applications in Aircraft Critical Computer Systems**
    - Airbus 320: Command-Monitor (COM-MON) Pairs
    - Boeing 777: Command-Monitor-Standby Lanes (Motorola, Intel, AMD μprocessors)

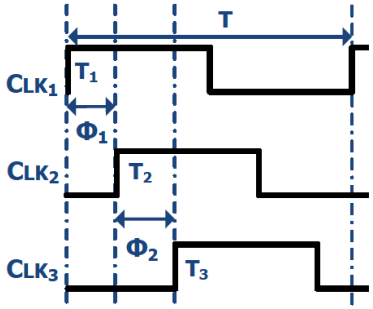
- **1990's: Intra-Chip Level Functional Blocks (ALU, Communication, Memory)**
  - **SPARC v7-based Atmel ERC32 —** ECC checks, Signature Monitoring, etc.
    —> SPARC v8-based Open Source "LEON" — Fault-Tolerant version
    - Gaisler & ESA: www.gaisler.com/cms/index.php?option=com_content&task=view&id=338&Itemid=231

Incorporating Fault Tolerance Features
Level of Application of Redundancy?



- **2000's: Basic component Level (Flip-Flop)**
  - **Soft Error**
    - **Detection basic principle) ——>** [Anghel *et al.*, DATE 2000]
    - **Error Detection & Recovery:**
      - GRAAL paragdigm [Nicolaïdis, ITC 2007]
      - Triplication of Flip-flops and Skewed Clocks [Avirneni *et al.*, DSN 2009] *****
    - **Iroc RoCS81**: rad tol LEON 2.1)
    - **ARM**: Optimizations/Extensions: Tolerance of Delay Faults ARM (Razor Scheme – 2004)
    - **Intel**: Scan Chain Protection — 2004 and coping with NBTI Faults — 2006

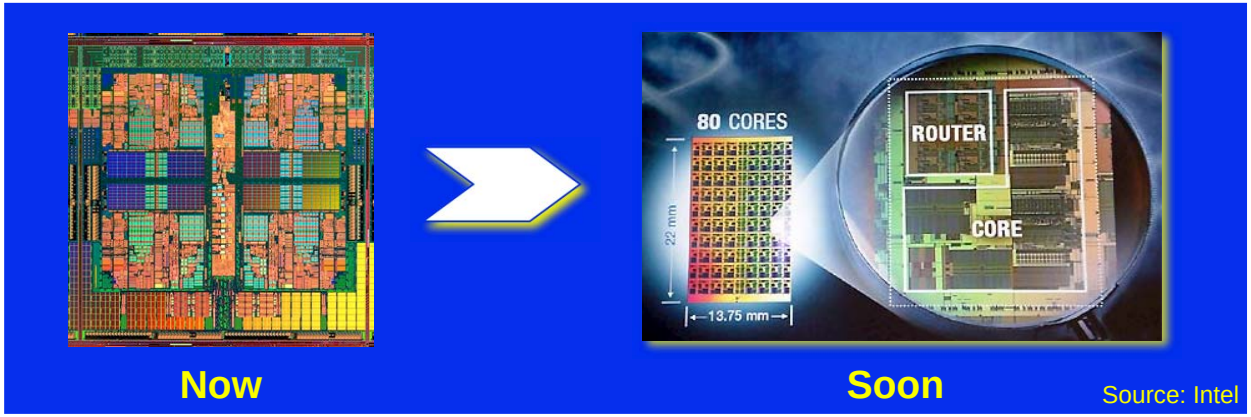


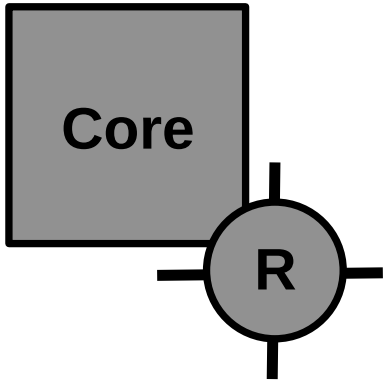


CASE	R_1	R_2	R_3	ERROR	BENIGN	RECOVERY
I	√	√	√	0	0	No Recovery
II	×	√	√	1	0	Load R_2 or R_3 into R_1
III	√	×	√	1	1	No Recovery
IV	√	√	×	0	1	No Recovery

2010's —> : New Technologies for Transistor Devices (CNT,…) ?

Towards Multi-[Many!]-Core Architectures





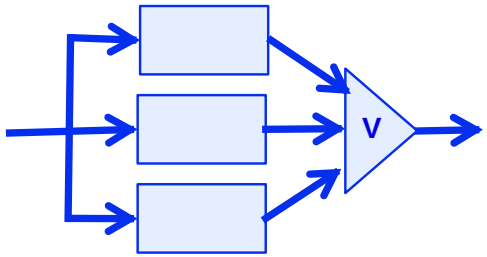
- **Multi-Core: ↗ performance while coping with power dissipation issues (very high clock frequency)**
- **Still, ↘ transitor size for including many of such cores —> significant % of defective cores (more than 10-20%?)**
- **Current context:**
  - **Chips are sorted according to frequency**
  - **Single core processor = "Downgraded" dual core circuits …**

How to go further?

- **Yield Enhancement**
- **On-line reconfiguration**

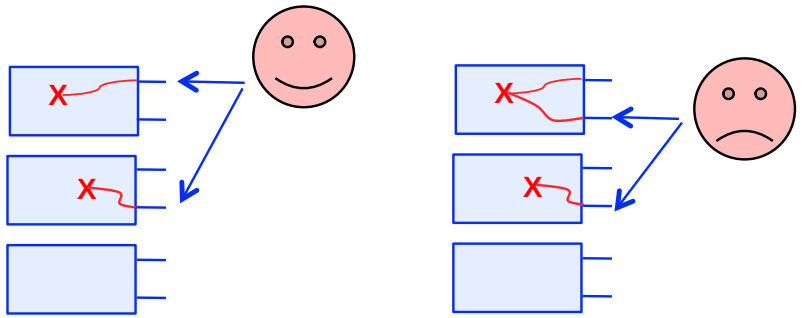
Yield Enhnacement: Triple Modular Redundancy

■ Basic Principle



- Application to Logic Cores
  (Memory ECC ≈ 100% coverage)

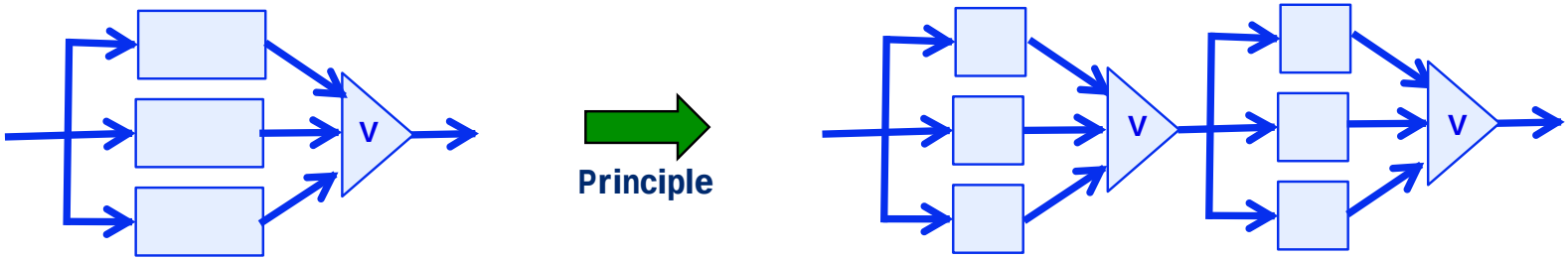
■ Coverage of Fault Tolerance?



Separate failures vs. Common failures

—> Figure of Merit (Reliability & Area) -> Limited Improvement

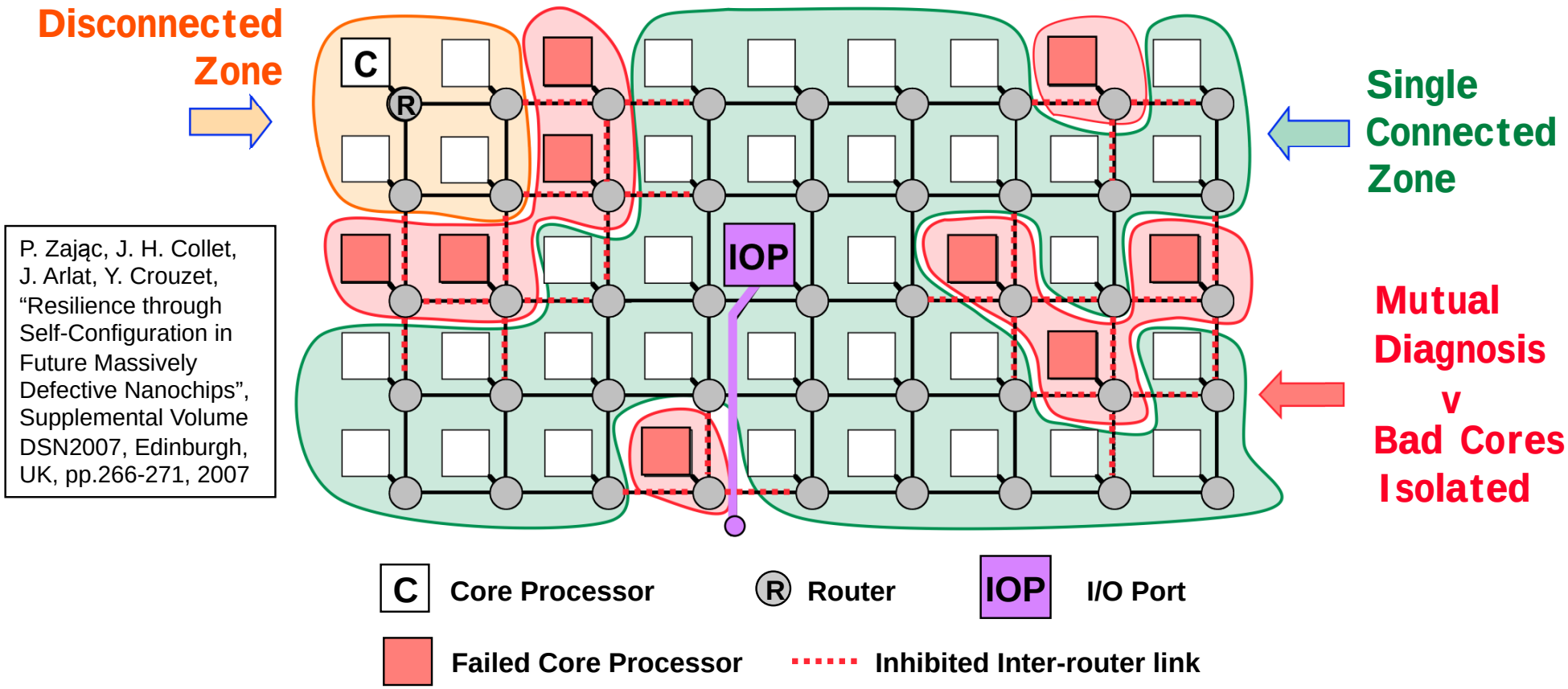
■ Chip Partitioning —> Reduction of combinational depth of replicated parts



Julien Vial *et al.*, Using TMR Architectures for SoC Yield Improvement
1st Int. Conf. on Advances in System Testing and Validation Lifecycle, pp. ,155-160, Porto, Portugal, 2009

Example Target Architecture

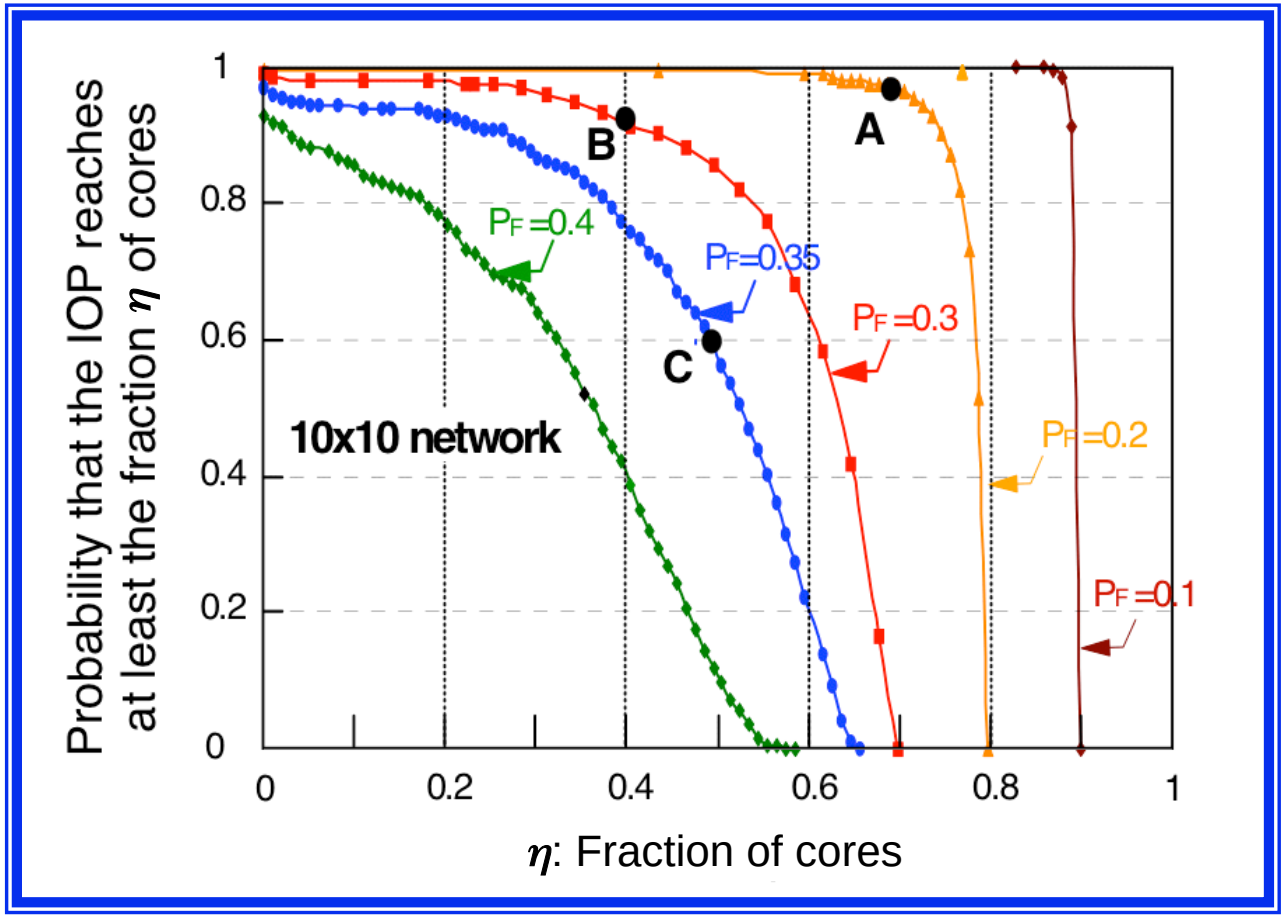
(5x9-node Network — Connectivity: 4)



Example of Basic Routing Scheme: Contract Net Protocol (CNP)

- **Step 1: The IOP broadcasts a Request Message across the Single Connected Zone (flooding, possibly inside a propagation radius). Each core adds the route to each forwarded message.**
- **Step 2: Each core sends an Acknowlegement Message to the IOP, which follows the RM route in the opposite direction.**
- **Step 3: The IOP stores the discovered routes in a special buffer (Valid Route Buffer).**

Example of Analysis

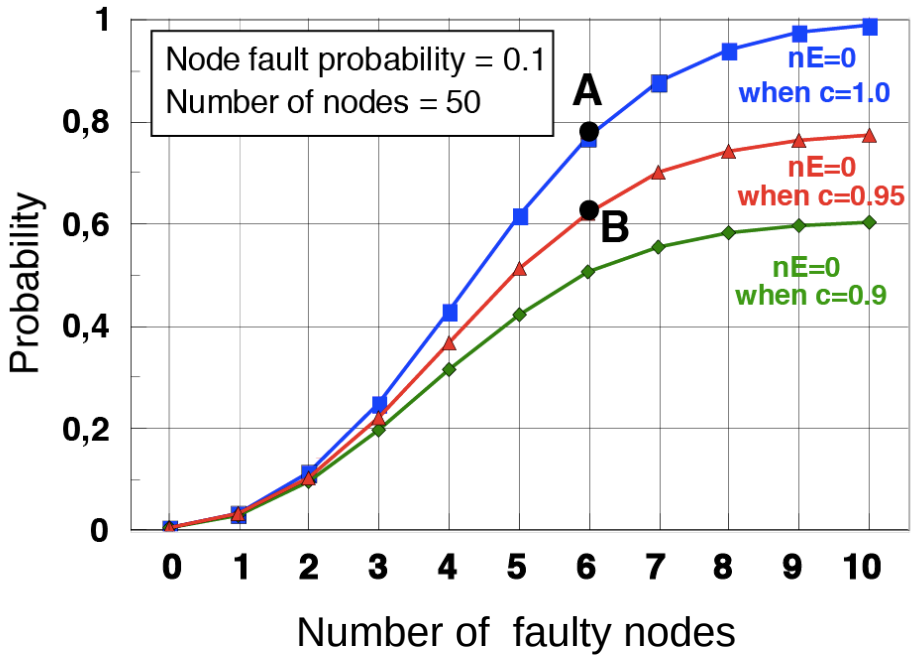


Point A (X_A = 0.68 and Y_A = 0.96) : the probability is approximately Y_A = 0.96 that the IOP reaches at least η = 68% of all cores when the core probability of failure P_F = 0.2.

Analysis of the Impact of Several Features

- I/O Ports = Hard Core (single point of failure) and Communication Bottleneck
  - —> ↗ Reliability of I/O Ports (Redundancy)
  - —> ↗ Number of I/O Ports
  - —> Location of I/O Ports
  - —> ↗ Connectivity of I/O Ports wrt Adjacent Nodes

- (Imperfect) Coverage —> of the Diagnosis
  - A: ≈ 80% of chips feature ≤6 failed nodes (perfect covergae)
  - B: ≈ 60% of chips feature ≤6 failed nodes

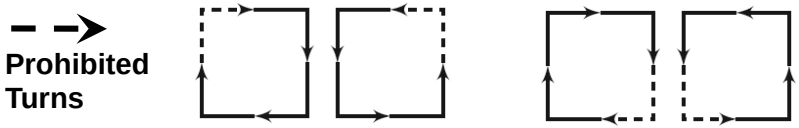


[Collet et al. IEEE TDSC, 2010]

On-Line Reconfiguration Issues

- **Error Detection and Fault Diagnosis**
  - Mutual Checks (Exchange of I am Alive Messages)
  - Applicable for "Fail-Silent Nodes" (strong assumption!)

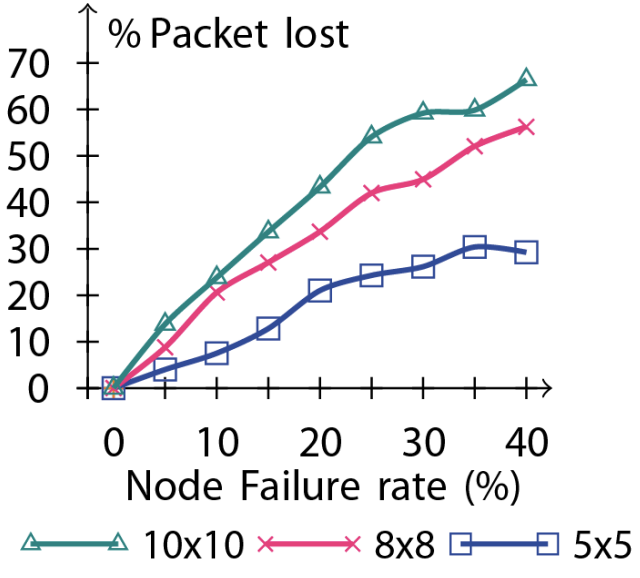
- **Adaptive Routing**
  - Avoid Table-based Algorithm (Hard Core)
  - Build upon Multiprocessor Domain Solutions (e.g., wormhole based NARA Alg.)
  - Combination of North Last & South Last Strategies



  - Fault Free Case : No message lost

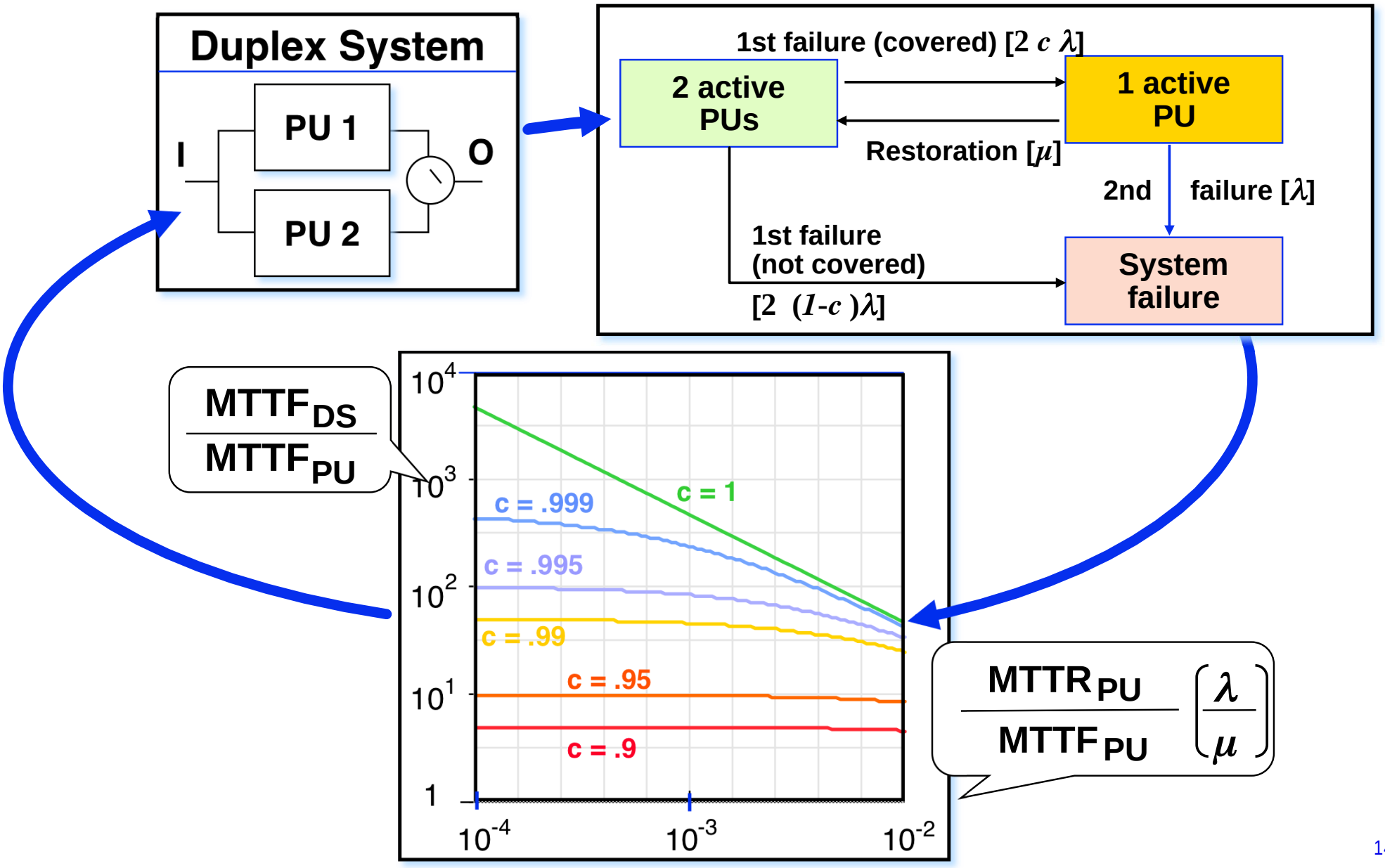
  - But, in presence of faults … ☹ !

When normal routing options are exhausted, the message buffered (current node —> Virtual Source) and then resent

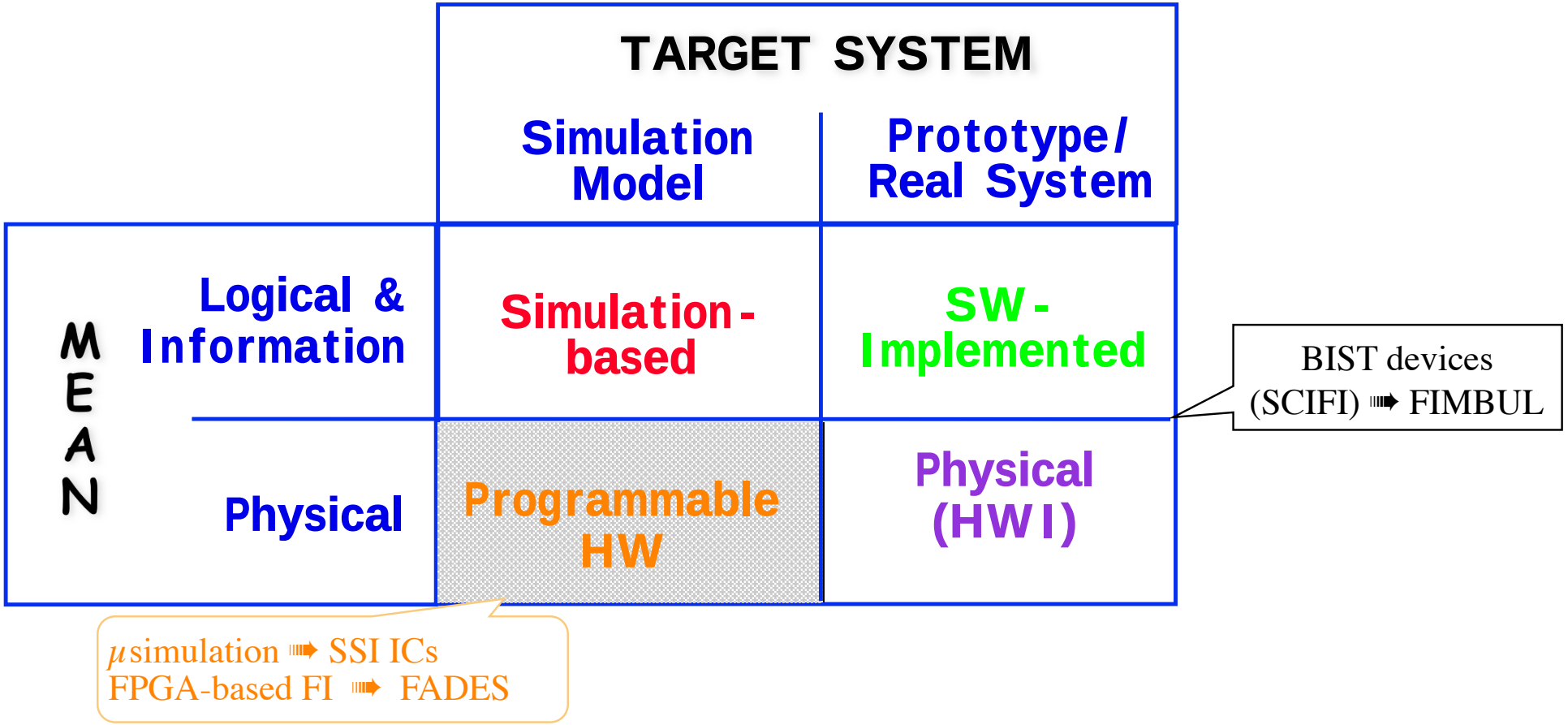


[Cunningham & Avresky, HPCA 1995; Chaix et al., NCA 2010]

Impact of FT Coverage on Dependability



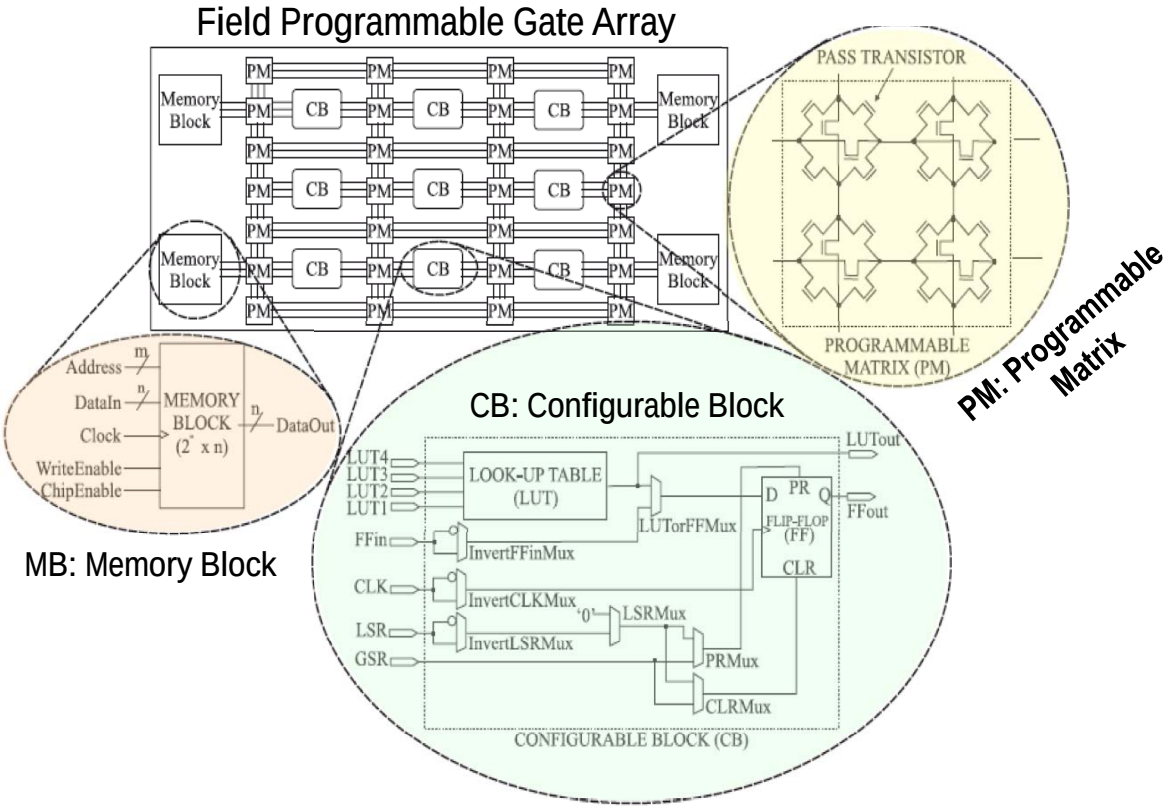
Fault Injection:
A Pragmatic Approach for Testing Fault Tolerance Mechanisms



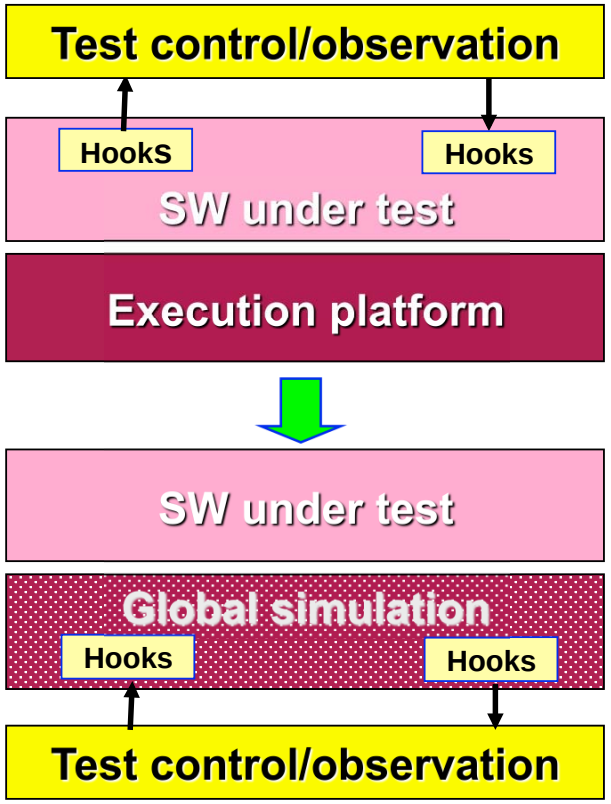
HW-Fault Injection

- Limitation of capabilities of SWIFT techniques wrt HW-level
- Increase of dependability concerns at HW level

- **FPGA-based FI technique**
  [De Andrés *et al*, IEEE TVLSIS 2008]



- **Virtual execution platform (incl. Processor and RT OS)**



F **= stuck-at, open, short, bit-flip, delay, etc.**

Concluding Remarks

- **Security-related Issues**
  - Crypto chips Assessment
  - Testability *vs.* Security (BIST devices —> potential vulnerabilities)
- **Application-level Issues**
  - Deployment of SW Applications on many-core architectures
  - Adaptation and Reconfiguration aspects
- **Increased Interdependencies between Hardware Matters and Software & Communication protocols Aspects**
- **"Beyond Moore" Technologies**
  - New "Transistor, Bit" Devices: CNT, Nanowires, Biomolecules, Organic molecules, …
  - Intrinsically unreliable !
- **The Wheel is Spinning…**

Hardware Dependability: Fault Tolerance to the Rescue!
Once again… ☺

To Probe Further

- L. Anghel, M. Nicolaïdis, “Cost Reduction and Evaluation of a Temporary Faults Detecting Technique”, IEEE/ACM DATE 2000, pp. 591-597, 2000
- L. Anghel, M. Nicolaidis, N. Achouri, "Built In Self Repair Techniques for Based on ECC Codes to Cope with Memories Affected by High Defect Densities”, IEEE VLSI Test Symp. 2004, April 2004.
- J. Arlat, M. Aguera, L. Amat, Y. Crouzet, J.-C. Fabre, J.-C. Laprie, E. Martins and D. Powell, “Fault Injection for Dependability Validation — A Methodology and Some Applications”, IEEE TSE, 16 (2), pp.166-182, 1990
- J. Arlat, A. Costes, Y. Crouzet, J.-C. Laprie, D. Powell, “Fault Injection and Dependability Evaluation of Fault-Tolerant Systems”, IEEE ToC, 42 , (8), pp. 913–923, 1993
- J. Arlat, Y. Crouzet, Y. Deswarte, J.-C. Fabre, J.-C. Laprie, D. Powell, “Tolérance aux fautes”, Encyclopédie de l'informatique et des systèmes d'information (partie 1), pp.241-270, Vuibert, 2006
- N.D.P. Avirneni, V. Subramanian, A. K. Somani, “Low Overhead Soft Error Mitigation Techniques for High-Performance and Aggressive Embedded Systems”, IEEE/IFIP DSN-2009, pp.
- A. Benso, P. Prinetto (Eds.), Fault Injection Techniques and Tools for Embedded Systems Reliability Evaluation, Kluwer, 2003
- J.H. Collet, P. Zajac, M. Psarakis, D. Gizopoulos, "Chip Self-Organization and Fault Tolerance in Massively Defective Multi-core Arrays", IEEE TDSC, 2010 (To appear)
- Y. Crouzet, C. Landraultn ”Design of Self-Checking MOS-LSI Circuits - Application to a Four-Bit Microprocessor”, IEEE FTCS-9, Madison, Wisconsin (USA), 1979, pp. 189-192
- C. Cunningham, D. Avresky, “Fault-tolerant Adaptive Routing for Two-dimensional Meshes,” IEEE Symp. on High-Performance Computer Architecture, (HPCA), pp. 122–131, 1995
- D. DeAndrés, J.C. Ruiz, D. Gil, P. Gil, “Fault Emulation for Dependability Evaluation of VLSI Systems”, IEEE ToVLSIS, 16 (4), pp., 422-431, 2008
- P. Folkesson, S. Svensson, J. Karlsson, “A Comparison of Simulation Based and Scan Chain Implemented Fault Injection”, IEEE FTCS-28, pp.284-293, 1998
- D. Hély, F. Bancel, M.-L. Flottes, B. Rouzeyre, “Secure Scan Techniques: A Comparison”, IEEE IOLTS'06, pp.119-124, 2006
- J. Lee, M. Tehranipoor, C. Patel, J. Plusquellic, “Securing Designs Against Scan-Based Side-Channel Attacks”, IEEE TDSC, 2007
- R. Leveugle, “Early Analysis of Fault-based Attack Effects in Secure Circuits”, IEEE ToC, 56 (10), pp.1431-1434, 2007
- E.F. Moore, C.E. Shanon, “Reliable Circuits Using Less Reliable Relays”, J. Franklin Institute, pp. 181-208, 281-297, 1956
- T. Munakata (Ed.), “Beyond Silicon : New Computing Paradigms”, CACM, 50 (9), pp. 30-72, 2007
- M. Nicolaïdis, “GRAAL: A New Fault Tolerant Design Paradigm for Mitigating the Flaws of Deep Nanometric Technologies,” IEEE ITC, p. 1–10, 2007
- J. Von Neumann, “Probabilistic Logics and the Synthesis of Reliable Organisms from Unreliable Components”, Automata Studies, C.E. Shannon, J. McCarthy, Eds., pp. 43-98, 1955
- P. Zając, J. H. Collet, J. Arlat, Y. Crouzet, “Resilience through Self-Configuration in Future Massively Defective Nanochips”, IEEE/IFIP DSN-2007 (Supplemental Volume), Edinburgh, UK, pp.266-271, 2007