

Faults Diagnosis Methodology for the WaferNet Interconnection Network

Yan Basile-Bellavance¹, Yves Blaquière² and Yvon Savaria¹

¹: GR2M, École Polytechnique de Montréal,

²: Département d'Informatique, Université du Québec à Montréal

Abstract - In this paper, the interconnection network (WaferNet) which is part of an active and reconfigurable prototyping board, named WaferBoard™, is analyzed to derive efficient defect diagnosis. The WaferNet structure spans an entire silicon wafer that inevitably contains defects, due to the nature of the microfabrication process, and defect management strategies are inserted in the design flow. Defects must be accurately located to efficiently reconfigure the circuit around them. Key differences between a conventional printed circuit board and WaferNet justify the proposed diagnosis methodology. A sequential walking-one algorithm and a broadcast algorithm are proposed to locate shorts or stuck-at faults in the network. It is shown that dedicated hardware architectures must be integrated in the network to locate those defects in a reasonable time. Analysis shows that the proposed diagnosis time complexity is $O(n^2)$, where n is the number of cells in the matrix. An upper bound time limit is calculated that depends on both the size and the number of faults in the circuits.

I. INTRODUCTION

This paper proposes an architecture and methodology for diagnosis of shorts and stuck-at defect in a recently proposed wafer-scale reconfigurable mesh network called WaferNet. This network is part of the WaferBoard™ technology [1] for rapid prototyping and validation of digital systems. It embeds a reconfigurable active substrate called WaferIC™ [2] that is obtained from photo-repetition of reticles. Each reticle contains an array of cells and thousands of configurable NanoPads allowing for easy alignment-insensitive placement of integrated circuits (ICs) on its surface. Any IC pin can then be actively interconnected through the configurable WaferNet. Each cell is interconnected with its neighbor cells in the four directions (Fig. 1). Each cell contains two programmable structures: a crossbar and an array of 4×4 NanoPads, interconnected as shown in Fig. 1 and Fig. 2. Each cell can redirect at most two IC pin signals, contacted with NanoPads, to other cells. All cells are daisy-chained and configured through custom scan chains (Fig. 1(f) and (g)).

The WaferNet architecture is detailed in [3], but no work on its diagnosability has been published. The problem of WaferNet diagnosis is similar to testing FPGA interconnect resources covered in [4]. However, the WaferIC™ does not contain programmable logic blocks, but only configurable crossbars and configurable I/O buffers. Furthermore, WaferNet spreads an entire wafer rather than a large silicon die. It is well known that wafer-scale circuits require defect tolerance to make them economically feasible [5].

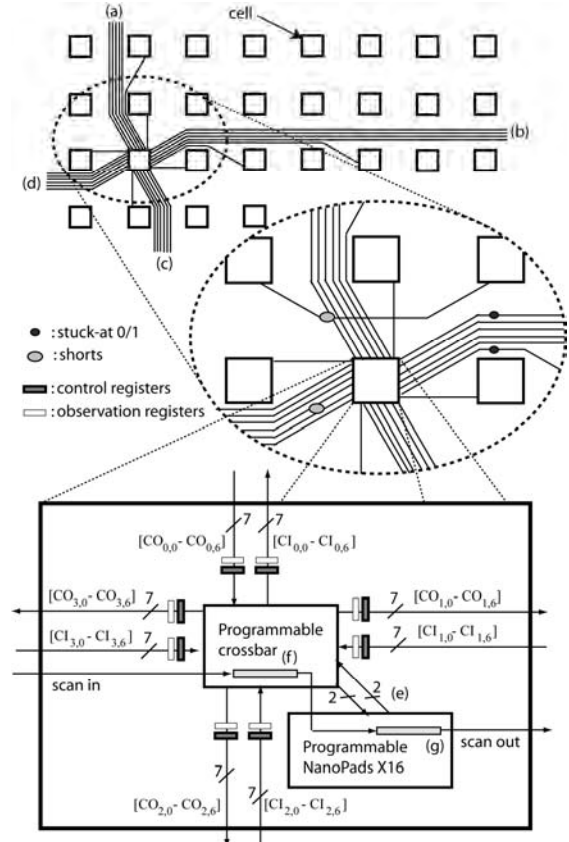


Fig. 1. Hierarchical structure of the WaferNet. The intercellular unidirectional network is deployed in 4 directions: (a) North, (b) East, (c) South and (d) West. A scan chain spans all the cells of each reticle of the WaferIC ((f) and (g)).

A dedicated tests and diagnosis method must be designed for the WaferNet architecture that differs from classical checker-board applied to PCB. This paper characterizes the resources added to the WaferNet to make it affordably diagnosable. It also covers the algorithms that allow detecting stuck-at 0/1 and short defects (affecting both parallel and perpendicular directions, as shown on Fig. 1) in interconnects. Only stuck-at 0/1 are diagnosed in the crossbars and no short diagnosis in the crossbar is done. Wafer-scale diagnosis is also addressed in this paper, which brings more stringent time and area constraints.

Section II details the proposed diagnosis methodology for the WaferNet. Section III analyzes its time complexity. Diagnosis time depends on both the number of scan chains

(CUT). For example, a long interconnect between a distant crossbar source and the $CI_{3,3}$ input terminal crosses 4 cells ($L=4$), and the control scan coming from the input port CI_0 or CI_1 does not cross any cell ($L=0$) because it is included in the CUT.

1) *Test Type A*: This test leverages local control and observation registers, and for each '1' applied on a crossbar input port, the crossbar is configured in broadcast mode (one-to-all configuration). For example, if a fault (3 (b)) is located in zones h1, h2 of the input port $CI_{3,3}$, then the fault can be observed when the broadcast is applied to port $CI_{3,3}$. This test phase can be applied concurrently on all CUTs.

2) *Test Type B*: It consists of applying the well known walking '1' algorithm [8], which reveals short and open faults on each interconnect as well as any stuck-at defects. With the proposed method, shorts can only be diagnosed between network interconnects and not between any trace outside of the network. A walking '1' (and walking '0') sequence is applied at the transceiver side of interconnects using control scan registers, signals are then captured at the crossbar outputs in the observation registers and shifted out for diagnosis.

To diagnose a short or a stuck in the network, it is necessary to shift out the content of the CUT observation register outside of the WaferIC to a test controller for each position of the walking-one. During the application of the walking '1', it is required to force a '0' on all control registers of the network. Such precaution enables the diagnosis of shorts on any pair of interconnect (parallel or perpendicular). Two other factors must be taken into account for adding concurrency to the test type B:

The concept of influence cone: for each CUT, there is a zone where it is required to force a '0' to avoid the overlap of potential multiple shorts in the network. Such influence cone is limited by the maximal extent of the incoming links times two (64 cells).

The resulting required gap between concurrent CUTs: each concurrent CUT must keep a security gap between two CUTs to avoid shorts overlap. A security gap of 2 times 64 cells is sufficient.

3) *Test type C*: To narrow down the possible locations of faults (shorts or stuck), further test sequences are needed. This test sequence is applied only where the CUT appears to contain faults detected with test type A or B. Test type C consists of a global broadcast test over long interconnects. The same one-to-all configuration is forced on the CUT. The difference comes from control registers. Instead of using local control register, the test uses control registers coming from the distant cells at the other end of the long interconnect. For example, there is an overlap between the two stuck-at-one defects 'd' and 'b' (Fig. 3 (d) and (b)). These overlapped faults propagate only to port $CO_{3,3}$ on test type B. To reveal fault 'd', test type C is used. If a '1' is seen on every capture register of the CUT, as shown in Fig. 4, then it is possible to conclude that a stuck-at-one exists on the interconnect (Fig. 4(d)).

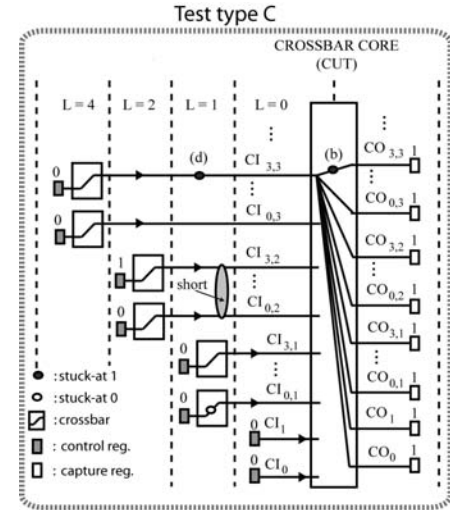


Fig. 4: Test type C is applied only on suspicious cells. An interconnect defect (d) result in capturing a '1' on any capture register.

C. Diagnosis Resolution with reduced overhead

Overhead is the ratio of the area dedicated to test and diagnosis compared to the total area used by the circuit under test. Each cell of the network occupies about $540 \times 540 \mu\text{m}$, and 50 % of its area is dedicated to digital circuits; the remaining area is occupied by power management and analog circuits. Significant efforts were dedicated to reducing the complexity of test and diagnosis circuits, with only 61 single-bit registers and some configurable 2 to 1 multiplexers dedicated to the task. The proposed design can be generalized to arbitrary size regular network of crossbar for SoC or wafer scale system.

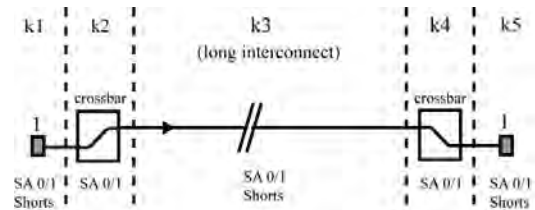


Fig. 5: a generalized definition of the diagnosis resolution in a regular network of interconnected crossbar.

Fig. 5 illustrates specific parts of the network structure ($k1$ to $k5$) under test. They define the considered granularity for diagnosis purpose. An extensive analysis of the diagnosis resolution of the proposed method in the presence of multiple shorts or multiple stuck at (SA) is beyond the scope of this paper. If the number of faults is low, for some mature microfabrication process, the probability to have multiple interacting faults on some interconnects is low. In such situation, the proposed method allows to diagnose faults located in the $k1$, $k3$ and $k5$ regions. The proposed diagnosis methodology is unable to differentiate shorts between $k1$, $k3$ and $k5$. On the other hand, SAs can be easily differentiated as shown in the test type C section. Regions $k2$ and $k4$ are only diagnosed for SA.

III. COMPLEXITY ANALYSIS AND RESULTS

The WaferIC consists of a concatenation of regular cells in a matrix ($m \times m$) included on each reticule of a whole wafer. The number of JTAG ports on each reticule has been fixed to two. Diagnosis of reticules can be executed concurrently (i.e. star JTAG architecture). Each cell contains 4 types of registers, as shown in Table 1, for a total of 375 registers per cell. For every partitioned scan chain, all register cells are connected together, and all cells are connected together by inter-cellular scan chains forming a global reticule scan chain as shown in Fig. 1 and 2. The most efficient multiple scan chain partition (i.e. 2) is implemented as follows:

- Scan chain 1: every crossbar test registers (i.e. 61) are daisy chained together. The total number of registers spanning an entire reticule (R_1) is $R_1 = 61 m^2$.
- Scan chain 2: the remaining configuration registers (375-61=314) are daisy chained together. The total number of registers spanning an entire reticule is $R_2 = 314 m^2$.

Such partitioning allows configuration or diagnosis to be done independently to improve efficiency of both operations. The diagnosis time depends on the clock period p .

1) *Diagnosis time for test type A*: The total test time (T_A) for the “Test type A” is $T_A=2(30)R_1p$. There are 30 input terminals in each crossbar, thus, 30 test vectors is applied to the crossbar. The factor 2 is introduced in the diagnosis time because the broadcast algorithm includes two passes forcing ‘1’ then ‘0’ on each input terminal. Each test can be done concurrently because the test is local.

2) *Diagnosis time for test type B*: The total test time is $T_B = (2(64/m)+1)^2(R_1V_B+m^2R_2)p$. The term (R_1V_B) defines the number of ‘1’ of the walking sequence applied to every output terminals ($V_B = 30 m^2$) times the number of registers in the network. The term (m^2R_2p) represents the total number of crossbar re-configurations. One reconfiguration is needed for every CUT in the network. The factor $(2(64/m)+1)$ is the number of reticules included in the influence cone to avoid overlap and to keep the diagnosis integrity. The influence cone is in fact a square, thus a power of 2 is applied to this slow down factor.

3) *Diagnosis time for test type C*: The number of test vectors for test type C is determined by the number of faults per reticule, f , (short or SA). The number of test vector (V_C) is $30fR_1$. Only the worst case is considered, when there is only one fault per CUT, which required a new test for every fault discovered. Thus, $T_C=(30)2f(R_1+R_2)p$. Similarly to the diagnosis time for test type B, there is one crossbar reconfiguration for every CUT. This is why R_2 appears in the formula. No concurrency is applied to test type C.

The total diagnosis time $D = T_A + T_B + T_C$ for a clock period p of 10MHz is summarized in Table II. The total diagnosis time is dominated by the test type B, which grows as $O(m^4)$. And the diagnosis time for the test type A is the order of microseconds and is negligible.

TABLE II :DIAGNOSIS TIME D (@10 MHZ)

m	$T_A + T_B$ (s)	Nb. of fault short or SA per reticule (f)	T_C (s)	D (s)
8	507.7	0	0	507.7
		30	4.3	512
		60	8.6	516.3
16	2276.6	0	0	2276.6
		30	17.3	2293.9
		60	34.6	2311.2

IV. CONCLUSIONS

This paper proposed a deterministic diagnosis methodology for the new WaferNet reconfigurable network. The WaferNet needs a specific diagnosis strategy, which differs from related classical PCB diagnosis, such as the checkerboard algorithm. Time complexity analysis is demonstrated to be $O(n^2)$, where n is the number of cells in the matrix ($n = m^2$). Furthermore, the diagnosis time depends on the number of faults found with test type C. The proposed method can be useful for any type of regular network of crossbar for SoC or wafer scale integration with a relatively low diagnosis test overhead. Research is ongoing to improve diagnosis efficiency by applying test concurrency with more sophisticated algorithms.

ACKNOWLEDGEMENTS

The authors would like to thank NSERC and Gestion TechnoCap Inc. for their financial support, and CMC Microsystems for providing design tools and support.

REFERENCES

- [1] Norman, R., U.S. Patent Application Number 11/611,263.
- [2] R. Norman, O. Valorge, Y. Blaquiere, E. Lepercq, Y. Basile-Bellavance, Y. El-Alaoui, R. Prytula, Y. Savaria, “An Active Reconfigurable Circuit Board”, NEWCAS 2008
- [3] Lepercq E., Valorge O., Basile-Bellavance Y., Savaria Y., Blaquiere Y., “An Interconnection Network for a Novel Reconfigurable Circuit Board”, NEWCAS 2008, June, pp. 129-132, 2008.
- [4] A. Doumar, H. Ito, “Detecting, diagnosing, and tolerating faults in SRAM-based field programmable gate arrays: a survey”, IEEE Trans. on VLSI Systems, vol. 11, no. 3, June 2003.
- [5] Brewer, J.E., “Promises and Pitfalls of WSI”, in Wafer Scale Integration, Kluwer, pp.1-29, 1989.
- [6] R. Jain, A. Mukherjee, and K. Paul, “Defect-aware design paradigm for reconfigurable architectures,” in Proc. of the 2006 emerging VLSI technologies and architectures, Karlsruhe, Germany, Mar 2-3, 2006.
- [7] M. Lu, Y. Savaria, B. Qiu, J. Taillefer, “IEEE 1149.1 based defect and fault tolerant scan chain for wafer scale integration,” 18 th Proc. IEEE International Symp. On Defect and Fault Tolerance in VLSI systems, 2003.
- [8] A. Hassan, J. Rajski, V. K. Agarwal, “Testing and Diagnosis of Interconnects using Boundary Scan Architecture,” 1988, International Test Conference.