Consiglio Nazionale delle Ricerche

Reprint

A Self-Diagnosis Algorithm for Hypercube Networks and its Application to VLSI Circuit Testing

Piero Maestrini and Paolo Santi

In: Design and Diagnostics of Electronic Circuits and Systems - DDECS'98 Second International Workshop (Szczyrk, Poland, 1998)

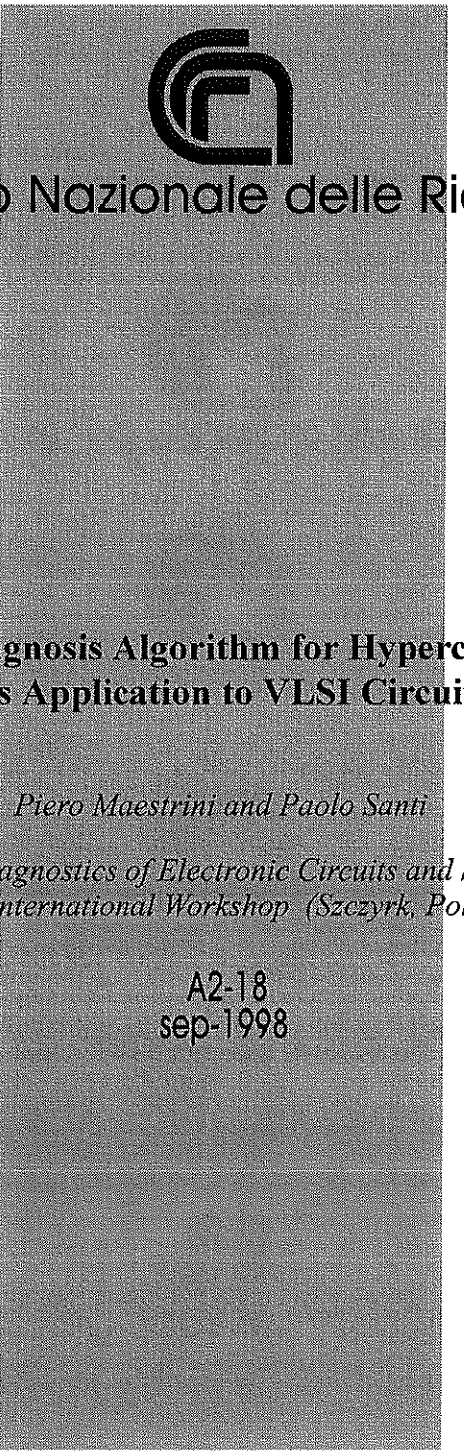
A2-18
sep-1998

I.E.I.
ISTITUTO DI
ELABORAZIONE DELLA
INFORMAZIONE

Pisa

COVER DESIGN: silvia rigobon

I.E.I. via S.Maria, 46 - 56126 PISA
Tel + 39 50 593414 Fax + 39 50 554342 E.mail biblio@iei.pi.cnr.it



Consiglio Nazionale delle Ricerche

*A S*elf-Diagnosis Algorithm for Hypercube Networks and its Application to VLSI Circuit Testing

Piero Maestrini and Paolo Santi

In: Design and Diagnostics of Electronic Circuits and Systems - DDECS'98
Second International Workshop (Szczyrk, Poland, 1998)

A2-18
sep-1998

A Self-Diagnosis Algorithm for Hypercube Networks and its Application to VLSI Circuit Testing

Piero Maestrini and Paolo Santi
Department of Computer Science, University of Pisa
Istituto di Elaborazione dell'Informazione del CNR, Pisa
maestrin@iei.pi.cnr.it, santi@iei.pi.cnr.it

Abstract

In this paper we introduce a self-diagnosis algorithm for hypercube-connected systems. The algorithm produces a diagnosis which is provably correct if the number of faulty units in the system is less than a threshold T_σ, asserted by the algorithm itself. Although the diagnosis may be incomplete, simulations show that the expected number of unidentified units is very small. The application of the diagnosis strategy to the manufacturing test of VLSI chips is also considered.

1. Introduction

The self-diagnosis approach, introduced by Preparata et al. [1], relies upon interprocessor *tests*, where every processing element (the *tested* unit) is tested by at least another processing element (the *testing* unit) directly connected to it. Every test involves two units. Units may be in a state of faulty or non-faulty. A test is performed as follows:

- the testing unit requests the tested unit to run a test program;
- the tested unit returns an output to the testing unit;
- the testing unit, based on the response of the tested unit, generates a binary outcome. The outcome is 0 if the test passes and 1 if the test fails.

The test invalidation rule, defined in Table 1, assumes that the test is totally reliable if the testing unit is non-faulty and totally unreliable if the testing unit is faulty.

Testing unit's state	Tested unit's state	Test outcome
Non-faulty	Non-faulty	0
Non-faulty	Faulty	1
Faulty	Non-faulty	0 or 1
Faulty	Faulty	0 or 1

Table 1. Invalidation rule

A multiprocessor system can be represented by an undirected graph $\mathcal{G}=(\mathcal{V},\mathcal{E})$, called *system graph*, where $\mathcal{V}$ is the set of vertices (also called units, nodes, processors or processing elements) and $\mathcal{E}$ is the set of edges. Edges represent communication links between units in the system.

The set of tests utilized for the purpose of diagnosis is defined by the directed graph $\mathcal{DG}=(\mathcal{V},\mathcal{E}')$, where $\mathcal{E}'=\{[u,v]$ such
that unit u tests unit $v\}$[1]. $\mathcal{DG}$ is called the *diagnostic graph* of the system. Units u and v must be adjacent, i.e. $[u,v]\in\mathcal{E}' \Rightarrow (u,v)\in\mathcal{E}$. The test of node v performed by node u is denoted by $u \xrightarrow{\alpha} v$, where α is the test outcome. We use notation $u \xleftarrow{\alpha\ \beta} v$ to collectively denote a test of v performed by u with outcome β and a test of u performed by v with outcome α.

The set σ of all test outcomes is called *syndrome*. It is immediate from Table 1 that any given fault set may yield different syndromes; conversely, any given syndrome may derive from different fault sets.

Given the syndrome, the *diagnosis algorithm* produces a diagnosis of the system, i.e. classifies each units as either

faulty, *non-faulty* or *unknown*. The diagnosis algorithm is supposed to be executed by an external and reliable computer, called the *diagnoser*. The diagnoser collects the syndrome from the units in the system and then executes the diagnosis algorithm.

Let $\mathcal{V}f$, $\mathcal{F}$ and $\mathcal{G}$ denote the actual fault set, the set of units identified as faulty and the set of units identified as non-faulty by the diagnosis algorithm, respectively. Of course, $\mathcal{F}\cap\mathcal{G}=\emptyset$. The diagnosis is said to be *correct* if $\mathcal{F}\subseteq\mathcal{V}f$ and $\mathcal{G}\subseteq\mathcal{V}-\mathcal{V}f$ (i.e. no unit which is actually non-faulty is diagnosed as faulty and no unit which is actually faulty is diagnosed as non-faulty); it is said to be *complete* if $\mathcal{F}\cup\mathcal{G}=\mathcal{V}$. A diagnosis which is both correct and complete is called *one-step diagnosis*.

It has been proved [1] that the one step diagnosis is possible if the cardinality of $\mathcal{V}f$ is less than, or equal to, the *one-step diagnosability* of the system, a parameter which is limited above by the minimum indegree of nodes in $\mathcal{G}$. When applied to regular systems, where each processor is connected to a small, and generally constant, number of neighbors, this condition is quite restrictive. For example, the one-step diagnosability of a toroidal rectangular grid is 4; this means that, regardless the number of units in the system, correct and complete diagnosis is always possible only if the number of faults does not exceed 4. Given this limitation, the research interest has shifted toward diagnosis algorithms which are complete but non correct, or correct but not complete. The former approach is called *probabilistic diagnosis*: the diagnosis algorithm produces a complete diagnosis which is correct with high probability. This probability decreases as $\#\mathcal{V}f$ increases. The latter approach, which is going to be described in details in the next sections, produces a diagnosis which is proved to be correct under very general conditions, but is usually incomplete.

2. Self-diagnosis approach to VLSI testing

The self-diagnosis theory may find a promising application in the "wafer scale" manufacturing test of VLSI. The goal of the diagnosis is the identification of good integrated circuits (IC's) within the wafer, which will be packaged, while the faulty circuits will be discarded.

The related technologies are among the most critical in the manufacturing process of integrated circuits. The usual approach requires testing of individual IC's by means of a *test machine*, which is assumed to be fault-free. The test machine establishes connections to the inputs and output pins of individual IC's, supplies an input sequence and compares the received output sequence with the expected sequence, which may be stored in the test machine itself or may be drawn from a certified fault-free chip, called *golden unit*.

This approach has some disadvantages. First of all, a wafer may contain hundreds of IC's which have to be tested sequentially and the probing unit has to be repositioned every time on a new IC in the wafer. This implies that the time required to test an entire wafer may be quite large, and it tends to increase as the technology advances because of the increasing complexity of the test. Furthermore, the test should be performed at the operating speed of IC's to be accurate. However, the test computer is unlikely to match the speed of the IC's under test and, if even this is the case, its design should be frequently updated in order to keep the pace with the advances in IC's technology.

It has recently been suggested [2] that self-diagnosis could replace the usual testing strategies in the wafer scale testing, considering that the IC's are arranged in a regular pattern within the wafer. Rather than testing the IC's individually by means of the test machine, the IC's themselves could perform mutual tests. To perform the tests, adjacent IC's could be assigned the same job (in principle this simply means that they receive a common input sequence) and the outputs could be compared. To a certain extent, this implies the possibility of performing the tests simultaneously. The role of the test machine would be reduced to supplying the input sequence, collecting the test outcomes (one bit for every comparison) and executing the diagnosis algorithm.

In order to implement self-diagnosis strategies on the wafer, some hardware support is needed: in fact, a wafer can be seen as an array of identical components, which should be complemented with comparators, communication links, power supply, ground, clock and so on. Although fulfilling these requirements consumes some additional silicon area, the testing of the IC's may become more cost effective with respect to the traditional methods. Another advantage is the possibility of testing the chips at the maximum sustainable speed.

On the other hand, application to wafer-scale testing of IC's poses a serious challenge to the theory of self-diagnosis since:

- the expected fraction of faulty chips in the wafer can be as large as 50%, depending on the complexity of the chip.
- in order to be feasible, the interconnection structure to be implemented on the wafer has to be regular and the degree of the nodes has to be small.

Given these constraints, the diagnosis algorithms cannot provide a diagnosis which is both correct and complete.

Hence, all the strategies proposed

In [2], Rangarajan et al. proposed two dimensional mesh. A test is c test sequence is applied to each diagnosis which is complete but decreases significantly as the leng

In [4], Huang et al. proposed an a IC's whose outputs agree, and de fraction (very close to 1) of fault-f size of the system. This means th the production of defecting chips.

In [10], Maestrini et. al. propose produced by this algorithm is cor which is asserted by the algorithm function $T(N)$, which depends uni actual number of faults may be fa simulated with systems of differ evaluate the expected value of T_σ.

n		10% of faulty un
64	$E(T_\sigma)$	63
	$E(n_d)$	0
256	$E(T_\sigma)$	255
	$E(n_d)$	1
1024	$E(T_\sigma)$	1017
	$E(n_d)$	1

Table 2. Simulations resu over a sample of 500 si uniform.

Simulations show that, the diagn large number of faults. However, unidentified units grows significa wafer), about one third of them m that the state of some chips on the using traditional technique or m production process is unpaired.

As one could easily expect, the decreases with the increase of the scale testing of IC's, the self diagr nodes is larger than 4. On the othe possible, if the proposed approach

In [3], LaForge et al. introduced lying on the border of the structur of units that declare each other fau algorithm diagnoses correctly, wit arbitrary fraction of fault-free unit interconnection structure should b

In the next section we introduce from the one proposed in [10], pro the wafer is less than a threshol incomplete, simulations results re small.

Hence, all the strategies proposed for the VLSI manufacturing test are either not complete or not correct.

In [2], Rangarajan et al. proposed a diagnosis strategy based on voting. The chips on the wafer are connected in a two dimensional mesh. A test is composed by a sequence of test vectors, each with an associated fault coverage. The test sequence is applied to each chip, and the outcome of adjacent chips are compared. The strategy produces a diagnosis which is complete but may be incorrect. The authors show that the probability of incorrect diagnosis decreases significantly as the length of the test sequence increases.

In [4], Huang et al. proposed an algorithm for mesh-connected systems. The strategy identifies *factions* of at least 3 IC's whose outputs agree, and declares them fault-free. It is shown that the algorithm identifies correctly a constant fraction (very close to 1) of fault-free units and a constant fraction (very close to 1) of faulty units, irrespective of the size of the system. This means that a small fraction of faulty units is allowed to be diagnosed incorrectly, leading to the production of defecting chips.

In [10], Maestrini et. al. proposed a deterministic algorithm for toroidal mesh-connected systems. The diagnosis produced by this algorithm is correct provided the number of faulty units in the system is less than a threshold $T\sigma$, which is asserted by the algorithm itself. It has been shown that for arbitrary syndrome σ, $T\sigma$ is bounded below by a function $T(N)$, which depends uniquely on the size N of the system. It has been proved that $T(N) \in \Theta(N^{1/2})$. Since the actual number of faults may be far above the syndrome independent bound $T(N)$, the diagnosis algorithm has been simulated with systems of different sizes where different numbers of faults are uniformly distributed in order to evaluate the expected value of $T\sigma$. Simulations results are listed in Table 2.

n		10% of faulty units	20% of faulty units	30% of faulty units	40% of faulty units	50% of faulty units
64	E(Tσ)	63.6	63.2	61.3	57.1	50.8
	E(n_d)	0.0	1.0	2.0	4.8	11.8
256	E(Tσ)	255.9	255.8	247.7	221.8	174.1
	E(n_d)	1.2	1.3	3.3	20.6	75.4
1024	E(Tσ)	1017.3	1008.4	989.8	845.6	604.2
	E(n_d)	1.1	2.2	12.1	104.5	417.5

Table 2. Simulations results: E(Tσ) and E(n_d) are the average of values of Tσ and n_d, respectively, over a sample of 500 simulations; n_d is the number of unidentified units. Distribution of faults is uniform.

Simulations show that, the diagnosis provided by the algorithm may expected to be correct in the occurence of a large number of faults. However, as the percentage of faulty units in the wafer approaches 50%, the number of unidentified units grows significantly: for a mesh of 256 units (which corresponds to the typical situation on the wafer), about one third of them may remain unidentified. When applied to VLSI testing, incomplete diagnosis means that the state of some chips on the wafer remain unknown. These chips may be tested individually by a test computer using traditional technique or may be considered faulty and discarded. In both cases, the effectiveness of the production process is unpaired.

As one could easily expect, the percentage of units whose state is left unidentified by the diagnosis algorithm decreases with the increase of the degree of nodes in the system graph. This implies that, when applied to wafer-scale testing of IC's, the self diagnosis algorithm should be based on interconnection structures where the degree of nodes is larger than 4. On the other hand, the degree of the interconnection structure should be increased as little as possible, if the proposed approach has to be cost effective.

In [3], LaForge et al. introduced a diagnosis strategy for nearly regular systems, where all units (excepting those lying on the border of the structure) have an arbitrary and constant degree. Their strategy, based on the aggregation of units that declare each other fault-free, produces a diagnosis which is neither correct nor complete. However, their algorithm diagnoses correctly, with probability approaching 1 as the system size increases, all the faulty units and an arbitrary fraction of fault-free units. Unfortunately, in order to obtain accurate results, the degree of the underlying interconnection structure should be quite large.

In the next section we introduce an algorithm for hypercube-connected systems. The algorithm, which is derived from the one proposed in [10], produces a diagnosis which is provably correct provided the number of faulty chips in the wafer is less than a threshold $T\sigma$, which is asserted by the algorithm itself. Although the diagnosis may be incomplete, simulations results reported in section 5 show that the expected number of unidentified units is very small.

3. The diagnosis algorithm

In this section we introduce a self-diagnosis algorithm for hypercube-connected systems.

An hypercube system can be represented by an undirected graph $\mathcal{H}=(\mathcal{V},\mathcal{E})$, where $\#\mathcal{V}=N=2^n$ and $\#\mathcal{E}=n2^{n-1}$ for some integer n greater than 1. The integer n is the dimension of the hypercube, and N is the total number of units in the system. Each node of the hypercube is labeled with a n digit binary number. Units are connected based on their Hamming distance: edge (u,v) exists iff $d_H(u,v)=1$, where d_H denotes Hamming distance. The set of tests utilized for the purpose of diagnosis is defined by the directed graph $\mathcal{DH}=(\mathcal{V},\mathcal{E}')$, where $\mathcal{E}'=\{[u,v]: (u,v)\in\mathcal{E}\}$.

The diagnosis algorithm is divided into three steps: *Local Diagnosis*, *Fault-Free Core Identification* and *Augmentation*. Every step of the algorithm is described in a separate subsection.

3.1 Local Diagnosis

The objective of local diagnosis is to classify every unit as either Z (unidentified unit), F (faulty unit) or D (dual unit). The state of Z-units remains unidentified and will be determined in the successive steps of the algorithm. F-units are known to be faulty. D-units are defined in disjoint pairs with the property that, for every pair, at least one unit is faulty. The sets of units classified Z, F or D are denoted by $\mathcal{Z}$, $\mathcal{F}$ and $\mathcal{D}$, respectively.

Unit classification is based upon the following Lemma:

Lemma 1: Let u and v be adjacent units;

a) if $u \xleftarrow{1\ 0} v$ *then u is faulty;*

b) if $u \xleftarrow{1\ 1} v$ *then at least one unit among u and v is faulty;*

c) if $u \xleftarrow{0\ 0} v$ *then u and v are in the same state (that is, both units are either faulty or non-faulty);*

d) if v is faulty and $u \xrightarrow{0} v$ *then u is faulty.*

Proof: Immediate from the invalidation rule. □

Local Diagnosis proceeds as follows. Initially we identify F-units (case a) and d) of Lemma 1). Then we identify the maximum number of D-units (case b) of Lemma 1), which are defined in disjoint pair, by finding the maximum matching in the undirected subgraph $\mathcal{DH}'=(\mathcal{V}-\mathcal{F},\mathcal{E}'')$, where $\mathcal{E}''=\{(u,v): [u,v]\in\mathcal{E}', [v,u]\in\mathcal{E}'$ and both edges are labeled with 1$\}$. This problem can be easily solved using one of the many algorithms in literature, observing that graph $\mathcal{DH}'$ is bipartite, since it is a subgraph of $\mathcal{H}$, which is bipartite [5]. Finally we define the set $\mathcal{Z}$ as the set of units which were classified as neither F nor D in the preceding steps.

3.2 Fault-Free Core Identification

At the end of local diagnosis, each node in the system has been classified as either F, D or Z-unit.

In the second step, the subgraph $\mathcal{H}'$ of $\mathcal{H}$ induced by the units classified as Z is considered. The set of all the units in every connected component of $\mathcal{H}'$ is defined as an *aggregate*. Z-units which are adjacent only to F and D-units are trivial aggregates. The collection of all the aggregates is denoted by $\{\Gamma_1,\Gamma_2,\ldots,\Gamma_k\}$.

An important property of aggregates is immediate from Lemma 1: all of their units are in the same state. However, we cannot decide whether the actual state is faulty or fault-free.

Letting α be the maximum of $\#\Gamma_1,\#\Gamma_2,\ldots,\#\Gamma_k$, the Fault-Free Core (denoted $\mathcal{FFC}$) is defined as the union set of all the aggregates of cardinality α.

The $\mathcal{FFC}$ plays a fundamental role in the diagnosis algorithm, since it will be shown that, under the conditions stated in the next section, all units in the $\mathcal{FFC}$ are non-faulty. Consequently, every test performed by nodes in the $\mathcal{FFC}$ is completely reliable.

3.3 Augmentation

The third step of the algorithm is aimed to augmenting both sets $\mathcal{FFC}$ and $\mathcal{F}$, by identifying the state of the largest possible number of nodes in sets $\mathcal{D}$ and $\bigcup_{i=1,\ldots,k}\Gamma_i - \mathcal{FFC}$.

Augmentation is based on the tests of units in $\mathcal{V}-\mathcal{FFC}-\mathcal{F}$ performed by units in $\mathcal{FFC}$. For each test, the outcome is considered and:

- if the test outcome is 0, then v is identified as non-faulty and $\mathcal{FFC}$ is redefined as $\mathcal{FFC}\cup\{v\}$. If v belongs to some aggregate Γ_i, then all the units belonging to Γ_i are identified as non-faulty and included in $\mathcal{FFC}$.
- if the test outcome is 1, then v Γ_i, then all the units belonging
- if any unit v which was previo F. If v belongs to some aggreg

4. Diagnosis correctness a

In this section we show that the number of faulty units in the sys threshold is asserted by the diag depends only on the size N of the

Diagnosis correctness relies upon property guarantees that there ex units belonging to the $\mathcal{FFC}$ are act

Lemma 2. If $\#\mathcal{V}<N/2$, *there exists*

Proof. The number of Z-units in t $\mathcal{F}$ is $\#\mathcal{F}$ and the number of faulty maximum matching and at least exists at least one unit classified (possibly singleton) aggregate.

Theorem 1. *Given any syndrome* $T_\sigma=\#\mathcal{F}+\#\mathcal{D}/2+\alpha$, *and* α *is the n*

Proof. Suppose that there exists a in $\mathcal{F}$ is $\#\mathcal{F}$ and the number of fault is at least $\#\mathcal{F}+\#\mathcal{D}/2+\alpha$. Since $\#\mathcal{V}$

The bound T_σ depends on the ca turn, such numbers are dependent been derived in [8] by a worst cas

$$T(N) = \min_{1\le\alpha\le\max}\left\{\frac{}{2\left(\right.}\right.$$

$$i=\min\left\{j: \sum_{k=0}^{j}\binom{n}{k}\ge\alpha\right\}.$$

Unfortunately, $T(N)$ cannot be de proved [8] that $T(N)$ is $\Omega(N/\log^2$ approximations used in the worst

If the cardinality of the actual fa incomplete. In fact, whenever w $\mathcal{FFC}$ set, their state remains uni situation is depicted in Figure 1.

In the worst case, a circuit of diagnosis. It follows that the ma diagnosis is $n=\log N$.

Theorem 3. *The diagnosis produ system is less than n.*

The result stated by Theorem 3 hypercube of dimension n is n. F faulty chips in a wafer can be

- if the test outcome is 1, then v is identified as faulty and $\mathcal{F}$ is redefined as $\mathcal{F}\cup\{v\}$. If v belongs to some aggregate Γ_i, then all the units belonging to Γ_i are identified as faulty and included in $\mathcal{F}$.
- if any unit v which was previously classified as either Z or D tests with outcome 0 a F-unit u, then v is classified F. If v belongs to some aggregate Γ_i, then all the units belonging to Γ_i are identified as faulty and included in $\mathcal{F}$.

4. Diagnosis correctness and completeness

In this section we show that the diagnosis produced by the algorithm described above is correct provided that the number of faulty units in the system is less than a threshold T_σ, which depends on the actual syndrome σ. This threshold is asserted by the diagnosis algorithm itself. A lower bound $T(N)$, which holds for any syndrome and depends only on the size N of the system is also reported.

Diagnosis correctness relies upon two conditions: the former guarantees that there exist at least one Z-unit, and this property guarantees that there exists at least one (possibly singleton) aggregate. The latter guarantees that all the units belonging to the $\mathcal{FFC}$ are actually fault-free.

Lemma 2. If $\#\mathcal{V}_f<N/2$, there exists at least one aggregate.

Proof. The number of Z-units in the system is given by the expression $N-\#(\mathcal{F}\cup\mathcal{D})$. Since the number of faulty units in $\mathcal{F}$ is $\#\mathcal{F}$ and the number of faulty units in $\mathcal{D}$ is at least $\#\mathcal{D}/2$ ($\mathcal{D}$ is constructed by considering units matched by a maximum matching and at least one unit in each pair is faulty), it follows that $\#(\mathcal{F}\cup\mathcal{D})\le 2\#\mathcal{V}_f<N$. It follows that there exists at least one unit classified as Z at the end of the first step of the algorithm. So there exists at least one (possibly singleton) aggregate. ❑

***Theorem 1.** Given any syndrome σ, all the units in the Fault-Free Core are fault-free, provided $\#\mathcal{V}_f<T_\sigma$, where: $T_\sigma=\#\mathcal{F}+\#\mathcal{D}/2+\alpha$, and α is the maximum of the aggregate cardinalities.*

Proof. Suppose that there exists an aggregate Γ_i in the $\mathcal{FFC}$ composed of faulty units. Since the number of faulty units in $\mathcal{F}$ is $\#\mathcal{F}$ and the number of faulty elements in $\mathcal{D}$ is at least $\#\mathcal{D}/2$, then the total number of faulty units in the system is at least $\#\mathcal{F}+\#\mathcal{D}/2+\alpha$. Since $\#\mathcal{V}_f<\#\mathcal{F}+\#\mathcal{D}/2+\alpha$ by the hypothesis, this leads to a contradiction. ❑

The bound T_σ depends on the cardinalities of sets $\mathcal{F}$ and $\mathcal{D}$ and on the maximum of the aggregate cardinalities; in turn, such numbers are dependent on the actual syndrome σ. The following syndrome independent bound $T(N)$ has been derived in [8] by a worst case analysis:

$$T(N) = \min_{1\le\alpha\le\max}\left\{\frac{N}{2\left(1+\frac{n\alpha}{mf(\alpha)}\right)}+\alpha\right\}, \quad \text{where} \quad \max = \sum_{k=0}^{\left\lfloor\frac{n-1}{2}\right\rfloor}\binom{n}{k}, \quad mf(\alpha)=\alpha\cdot\frac{n-2i}{i+1}, \quad \text{and}$$

$$i=\min\left\{j:\sum_{k=0}^{j}\binom{n}{k}\ge\alpha\right\}.$$

Unfortunately, $T(N)$ cannot be determined analytically; however, it has been evaluated numerically, and it has been proved [8] that $T(N)$ is $\Omega(N/\log^2 N)$. It should be noted that this bound is very pessimistic, because of some rough approximations used in the worst case analysis.

If the cardinality of the actual fault set is less than T_σ, then the diagnosis is correct, although it is quite likely to be incomplete. In fact, whenever we have a circuit of faulty units encircling a set of units which do not belong to the $\mathcal{FFC}$ set, their state remains unidentified (unless some of them have been previously identified as F-unit). This situation is depicted in Figure 1.

In the worst case, a circuit of n faulty units enclosing an unidentified unit is sufficient to lead to incomplete diagnosis. It follows that the maximum number of faults that the system can tolerate in order to ensure complete diagnosis is $n=\log N$.

***Theorem 3.** The diagnosis produced by the diagnosis algorithm is complete if the total number of faulty units in the system is less than n.*

The result stated by Theorem 3 agrees with [11], in which it is shown that the one-step diagnosability of an hypercube of dimension n is n. However, this result is very poor, since as mentioned above the expected fraction of faulty chips in a wafer can be as large as 50%. A realistic evaluation of the diagnosis completeness has been

obtained using simulation. Simulation results are reported in the next section.

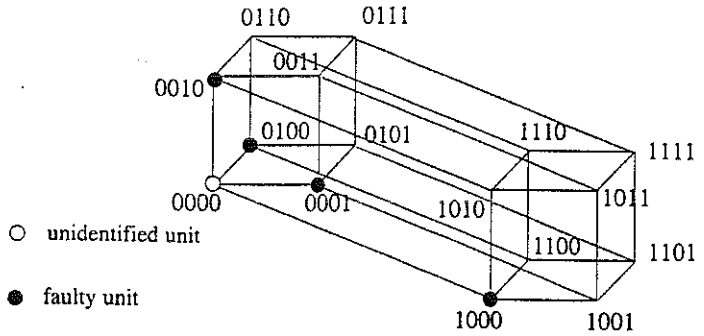


Figure 1. Incomplete diagnosis: a circuit of faulty units encloses an unidentified unit.

5. Simulations results and application to VLSI testing

The algorithm described above is powerful enough to be applied to VLSI testing. In fact, the algorithm produces a diagnosis which is provably correct provided the number of faulty chips in the wafer is less than the syndrome dependent bound Tσ. The threshold Tσ has been evaluated by means of simulation for different system sizes (see Table 3). For every system size, fault sets of different cardinalities have been distributed uniformly over the node set. For every fault set, the syndrome dependent bound Tσ has been determined by executing the diagnosis algorithm and the average E(Tσ) has been calculated over a sample of 1000 fault sets. As it is seen from Table 3, irrespective of the size of the system, the expected value of Tσ is very large even when the fraction of faulty units is 50%.

It is also seen that the variance is very small. This means that the diagnosis generated by the algorithm is correct with high probability even in the occurence of a large number of faults in the wafer. Of course, diagnosis may be incomplete. The expected number of chips which the algorithm is unable to diagnose has also been evaluated by means of simulation. The results (Table 4) are quite encouraging: even if the fraction of faulty chips in the wafer is 50%, the expected number of unidentified units is very small and the variance is also very small.

N		10% of faulty units	20% of faulty units	30% of faulty units	40% of faulty units	50% of faulty units
64	E(Tσ)	63.8 (0.25)	63.8 (0.202)	63.6 (0.789)	63.5 (0.590)	62.8 (1.547)
256	E(Tσ)	255.9 (0.083)	255.8 (0.285)	255.6 (0.521)	255.4 (0.706)	255.0 (0.962)
1024	E(Tσ)	1023.9 (0.152)	1023.8 (0.190)	1023.7 (0.405)	1023.6 (0.681)	1023.0 (1.303)
16384	E(Tσ)	16383.9 (0.074)	16383.8 (0.234)	16383.6 (0.359)	16383.6 (0.465)	16383.0 (1.133)

Table 3. Simulations results for hypercube systems. E(Tσ) denotes the expected value of Tσ over a sample of 1000 simulations. The variance is reported in parentheses.

N		10% of faulty units	20% of faulty units	30% of faulty units	40% of faulty units	50% of faulty units
64	$E(n_d)$	0 (0)	1.0 (0)	1.0 (0)	1.13 (0.175)	1.35 (0.456)
	$E(c_p)$	100%	99.9%	98.3%	89.5%	62.8%
256	$E(n_d)$	0 (0)	1.0 (0)	1.0 (0)	1.05 (0.043)	1.34 (0.373)
	$E(c_p)$	100%	99.9%	99.3%	90.8%	57.4%
1024	$E(n_d)$	0 (0)	0 (0)	1.0 (0)	1.03 (0.042)	1.28 (0.309)
	$E(c_p)$	100%	100%	99.6%	94.2%	62.7%
16384	$E(n_d)$	0 (0)	0 (0)	1.0 (0)	1.04 (0.043)	1.27 (0.300)
	$E(c_p)$	100%	100%	99.9%	97.7%	63.3%

Table 4. Simulations results for hypercube systems. $E(n_d)$ denotes the expected number of unidentified chips in the wafer over a sample of 1000 simulations. The variance is reported in parentheses. $E(c_p)$ denotes the expected ratio of complete diagnosis.

Simulations show that the algorithm described above can be used to diagnose faulty chips in a wafer. In order to

implement it on a wafer, some consi

First of all, we have to consider the
a wafer is in the order of hundre
interconnection would be at most 8.
total number of units in the system,

Moreover, faults in the additional
considered. The testing hardware
procedure is as follows. Each units
the task along each communication
generating the test outcome, which
are fault-free, we have a fault mod
simply assume than whenever two
respect to our algorithm, this simpl
not affect the worst case analysis
unavoidable faults affecting the com
test involves two distinct comparato

6. Concluding remarks

A self-diagnosis algorithm for hy
diagnosis which is proved to be co
dependent bound (Tσ). However, th
bound Tσ is very large and that the
when the expected fraction of faulty
applied to VLSI testing, where the e

References

1. Preparata, F.P, Metze, G. and C
IEEE Trans. Comput., vol. EC-16, p

2. Rangarajan, S., Fussel, D., M
vol. 39 n. 2, pp. 195 - 205, Februar

3. LaForge, L.E., Huang, K., Ag
Transaction on Computer vol. 43 n

4. Huang, K., Agarwal, V.K., L
Structures and Its Application to VL
44 n.4, pp.363-372, April 1995.

5. Saad, Y. and Schultz, M.H., "
pp. 867-872, July 1998.

6. Chen, C., Agrawal, D.P. an
Interconnection Network with Area
1332-1344, Dec. 1993.

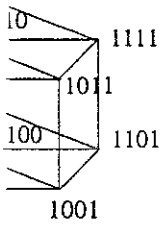
7. Maestrini, P., Santi, P., "Self-L
Symposium on Reliable and Distribu

8. Rubino, R., Santi, P., "A Self
33, IEI, Pisa, 1998.

9. Chessa, S., Maestrini, P., Mar
Arrays: Survey and Evaluation", *Pro*
1995.

10. Chessa, S., Maestrini, P., "Con
27, IEI, Pisa, June 1995.

11. Armstrong, J. R., Gray, F. G.,
Computers, vol. C-30, no. 8, pp. 587-



ies an unidentified unit.

sting. In fact, the algorithm produces a
n the wafer is less than the syndrome
ulation for different system sizes (see
en distributed uniformly over the node
d by executing the diagnosis algorithm
As it is seen from Table 3, irrespective
e fraction of faulty units is 50%.

s generated by the algorithm is correct
he wafer. Of course, diagnosis may be
o diagnose has also been evaluated by
fraction of faulty chips in the wafer is
e is also very small.

40% of faulty units	50% of faulty units
63.5 (0.590)	62.8 (1.547)
255.4 (0.706)	255.0 (0.962)
1023.6 (0.681)	1023.0 (1.303)
16383.6 (0.465)	16383.0 (1.133)

the expected value of Tσ over a

40% of faulty units	50% of faulty units
1.13 (0.175)	1.35 (0.456)
89.5%	62.8%
1.05 (0.043)	1.34 (0.373)
90.8%	57.4%
1.03 (0.042)	1.28 (0.309)
94.2%	62.7%
1.04 (0.043)	1.27 (0.300)
97.7%	63.3%

notes the expected number of
ns. The variance is reported in

ose faulty chips in a wafer. In order to

implement it on a wafer, some considerations have to be made.

First of all, we have to consider the feasibility of an hypercube structure on the wafer. Since the number of chips on a wafer is in the order of hundreds of units (100-2300 units usually), the degree of the underlying hypercube interconnection would be at most 8. An hypercube of dimension n, can be laid out in area $O(N^2)$, where $N=2^n$ is the total number of units in the system, using a two dimensional layout [6].

Moreover, faults in the additional hardware included on the wafer for the purpose of self-diagnosis have to be considered. The testing hardware is composed by comparators wired on each communication link. The testing procedure is as follows. Each units is assigned the same task. Upon task completion, each unit sends the output of the task along each communication channel. Each comparator receives the output of two units and compares them, generating the test outcome, which is 0 if the outputs agree, and 1 otherwise. If we assume that all the comparators are fault-free, we have a fault model slightly different from the model described in the Introduction, in which we simply assume than whenever two units test each other, the outcomes of the tests can only be 0-0 or 1-1. With respect to our algorithm, this simply means that no unit can be identified as faulty in the first step. Since this does not affect the worst case analysis, the previously reported results remain valid. In order to account for the unavoidable faults affecting the comparators, we may associate comparators to each chip under test, such that every test involves two distinct comparators. So, faults affecting comparators are treated as faults affecting the chips.

6. Concluding remarks

A self-diagnosis algorithm for hypercube-connected systems has been introduced. The algorithm produces a diagnosis which is proved to be correct provided the number of faulty units in the system is less than a syndrome dependent bound (Tσ). However, the diagnosis may be incomplete. Simulations show that the syndrome dependent bound Tσ is very large and that the expected number of units left unidentified by the algorithm is very low even when the expected fraction of faulty units is as large as 50%. This means that the algorithm is a good candidate to be applied to VLSI testing, where the expected fraction of faulty chips is usually large.

References

1. Preparata, F.P, Metze, G. and Chien, R.T., "On the Connection Assignment Problem of Diagnosable Systems". *IEEE Trans. Comput.*, vol. EC-16, pp. 848 - 854, Dec. 1967.

2. Rangarajan, S., Fussel, D., Malek, M., "Built-in Testing of Integrated Circuit Wafers", *IEEE Trans. Comp.* vol. 39 n. 2, pp. 195 - 205, February 1990.

3. LaForge, L.E., Huang, K., Agarwal, V.K., "Almost Sure Diagnosis of Almost Every Good Element", *IEEE Transaction on Computer* vol. 43 n. 3, pp. 295 - 305, March 1994.

4. Huang, K., Agarwal, V.K., LaForge, L., Thulasiraman, K., "A Diagnosis Algorithm for Constant Degree Structures and Its Application to VLSI Circuit Testing", *IEEE Transaction on Parallel and Distributed Systems* vol. 44 n.4, pp.363-372, April 1995.

5. Saad, Y. and Schultz, M.H., "Topological Properties of Hypercubes", *IEEE Trans. on Computers*, vol. C-37, pp. 867-872, July 1998.

6. Chen, C., Agrawal, D.P. and Burke, J.R., "dB Cube: a New Class of Hierarchical Multiprocessor Interconnection Network with Area Efficient Layout", *IEEE Trans. on Parallel and Distributed Systems*, Vol. 4, pp. 1332-1344, Dec. 1993.

7. Maestrini, P., Santi, P., "Self-Diagnosis of Processor Arrays Using a Comparison Model", *Proc. 14'th SRDS-Symposium on Reliable and Distributed Systems*, Bad Neuenahr, Germany, pp.218-228, September 1995.

8. Rubino, R., Santi, P., "A Self-Diagnosis Algorithm for Hypercube Connected Systems", *Internal Report B4-33*, IEI, Pisa, 1998.

9. Chessa, S., Maestrini, P., Mangione, M., Polacci, S., Santi, P., "Self-Diagnosing Algorithms for Processor Arrays: Survey and Evaluation", *Proc. 2th ICAuto International Conference on Automation*, Indore India, December 1995.

10. Chessa, S., Maestrini, P., "Correct and Almost Complete Diagnosis of Processor Arrays", *Internal Report B4-27*, IEI, Pisa, June 1995.

11. Armstrong, J. R., Gray, F. G., "Fault Diagnosis in a Boolean n-cube Array of Microprocessors", *IEEE Trans. Computers*, vol. C-30, no. 8, pp. 587-590, August 1981.