# Defect Localization Using Physical Design and Electrical Test Information


Zoran Stanojevic
Dept. of Electrical Engineering
Texas A&M University
College Station TX 77843
Tel: (979) 862-6610
Fax: (979) 847-8578
Email: zoran@cs.tamu.edu

Hari Balachandran
Texas Instruments, Inc.
Mixed Signal Products
Dallas TX
Tel: (214) 480-7602
Fax: (214) 480-7676
Email: harib@ti.com

D. M. H. Walker
Dept. of Computer Science
Texas A&M University
College Station TX 77843
Tel: (979) 862-4387
Fax: (979) 847-8578
Email: walker@cs.tamu.edu

Fred Lakhani
SEMATECH
Yield Management Technology
2706 Montopolis Drive
Austin TX 78741
Tel: (512) 356-7011
Fax: (512) 356-7640
Email: fred.lakhani@sematech.org

Sri Jandhyala
Texas Instruments, Inc.
Mixed Signal Products
Dallas TX
Tel: (214) 480-7598
Fax: (214) 480-7676
Email: srij@ti.com



## Abstract

*In this work we describe an approach of using physical design and test failure knowledge to localize defects in random logic. We term this approach computer-aided fault to defect mapping (CAFDM). An integrated tool has been developed on top of an existing commercial ATPG tool. CAFDM was able to correctly identify the defect location and layer in all 9 of the chips that had bridging faults injected via FIB. Preliminary failure analysis results on production defects are promising.*



## Keywords

Defect diagnosis, testing, failure analysis, fault diagnosis, fault localization, defect localization, fault isolation, defect isolation


## 1. Introduction

Fault localization or fault isolation is the process of identifying a region within an integrated circuit (IC) that contains a circuit fault, such as a short or open circuit. This region must be small enough that the defect causing the fault can be found and analyzed. This is very important for debugging new products, ramping yields, identifying test escapes and reliability problems in customer returns, and resolving quality assurance (QA) part failures. This problem is made more difficult by more interconnect layers and increased layout density, which often preclude a direct view of a defect site. This makes application of traditional defect localization methods. The International Technology Roadmap for Semiconductors projects the complexity of fault isolation to grow by 142 times over the next 15 years [1].

Current state-of-the-art practice for localizing defects in logic circuits is to use a stuck-at fault diagnosis approach. A set of passing and failing vector outputs is captured, typically at wafer test, and fed into the diagnosis system. The diagnosis produces a ranked list of nets on which a stuck-at fault best explains the observed failure patterns. A list of logically equivalent nets is also provided. An example of logical equivalence is that a stuck-at-1 fault on an input to an OR gate is logically equivalent to a stuck-at-1 fault on the OR gate output. Stuck-at diagnosis tools are commercially available as part of an automatic test pattern generation (ATPG) system, such as Mentor Graphics FastScan™ and IBM TestBench™. The list of suspect nets can be visualized in the chip layout database using a tool such as Knights Technology LogicMap™ [2][3]. Since even individual nets can be large, the idea case is that a "hit" was seen in wafer inspection, in that a defect was observed near one of the suspect nets. This location can then be examined with a scanning electron microscope (SEM). This procedure is shown in Figure 1. Unfortunately such inspection information is only available for a small fraction of all chips. The result is that the SEM search time for even a few nets can be quite lengthy, dominating total fault isolation time. A further challenge is that since signal nets often run only on interior layers, delayering must be performed prior to the inspection of a given layer. This delayering process can remove the sought-after defect, and precludes the application of additional test vectors to aid the search process.

Bridging faults are the most common faulty behavior caused by defects. However, stuck-at faults are not a very good model for bridging faults [4]. As a result, the stuck-at diagnosis can produce poor results, in that the real faulty nets are far down the list of suspects and so are never examined due to resource constraints, or they are not on the list at all. As a result, a significant fraction of the time the diagnosis fails. This wastes failure analysis lab resources, reduces the rate of yield learning, and can make QA quite difficult since typically there are only a small number of failed devices to examine.

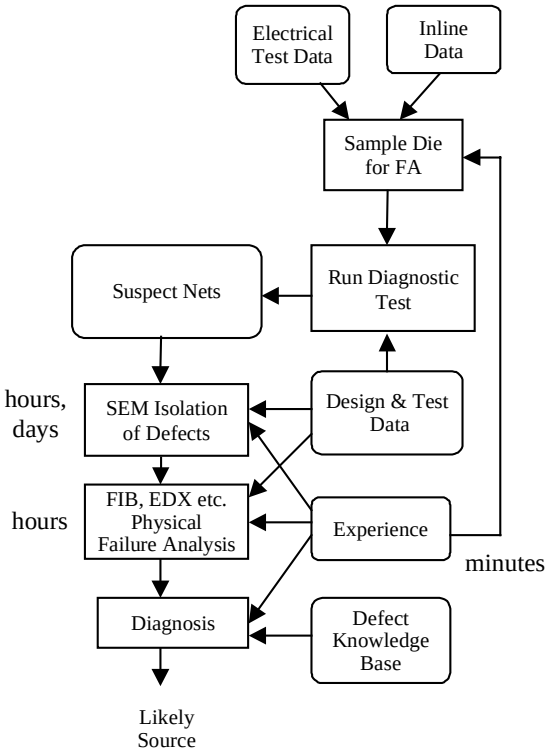


Figure 1. Current Defect Sourcing Approach

The solution approach used in this work is *computer-aided fault to defect mapping* (CAFDM). Rather than replace the mature stuck-at diagnosis infrastructure, our goal is to combine this test information with physical design information to reduce the SEM search area. As shown in Figure 2, we use physical design information to identify locations and layers within the circuit where short and open circuit faults can occur. This information is used to filter and improve the suspect net list provided by stuck-at diagnosis. The result is that the mature stuck-at diagnosis infrastructure is used, while its primary disadvantages are overcome. The project metrics are shown in Table 1. The metrics are interrelated in that by using layer and location information to reduce the search area, we also reduce the fault isolation time and failure rate.

In the remainder of this paper we describe the CAFDM approach and our experimental results. Section 2 describes previous work on defect diagnosis. Section 3 presents the CAFDM approach. Section 4 describes experimental results and Section 5 concludes.

## 2. Previous Work

Several methods have been suggested to improve diagnosis of bridging faults using the stuck-at fault model. A vector that detects a bridging fault also detects an associated stuck-at fault [5]. Composite signatures can be constructed for each potential bridging fault by the union of the four associated single stuck-at signatures. However this technique results in an unacceptable rate of misleading diagnoses. An improvement is to use only realistic faults, and better match restriction [6], which was demonstrated experimentally [7]. Using several different fault models can improve the diagnosis results [8]. These improved approaches require the construction of a large fault dictionary. In particular, it is often a requirement that one of the bridged nets be included in the dictionary in order to achieve a successful diagnosis. Since there are hundreds of thousands of nets in today's large industrial circuits, the dictionary construction can be very expensive.

Table 1. Project Metrics

| Description | Baseline | Goal |
|---|---|---|
| Failed Diagnosis | 30% | <15% |
| Physical Layer Ranking | No | Yes |
| Physical Location Ranking | No | Yes |
| Small Suspect List (< 10) | 80-95% | 90-99% |
| Average Localization Time | Days | Hours |

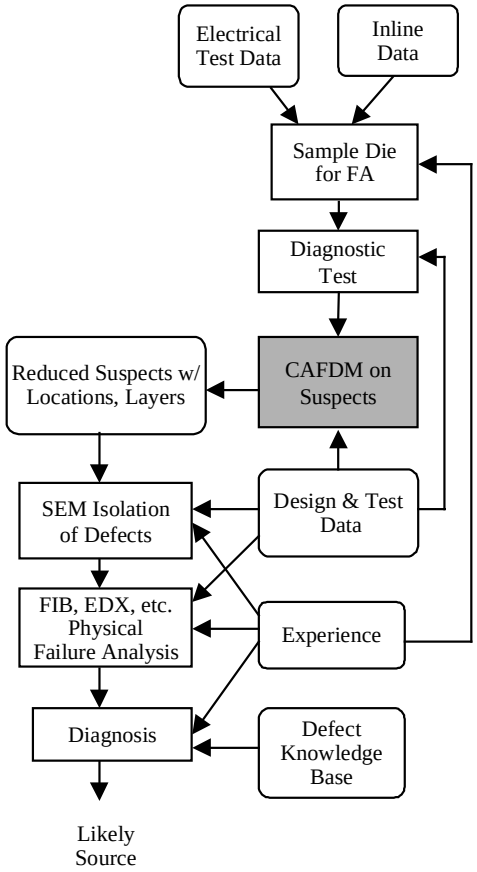


Figure 2. CAFDM modification of existing fault diagnosis approach

The best bridging fault diagnosis results can be achieved by modeling the bridging fault behavior at the circuit level, and using layout information to identify adjacent nets that could be involved in a bridge [9]. The drawback is the need for a completely different diagnostic infrastructure, such as a bridging fault simulator [10]. The separate infrastructure substantially increases the cost of diagnosis.

## 3. Computer-Aided Fault to Defect Mapping

In CAFDM we assume that with the addition of bridging faults we can accurately model the behavior of most faults, and that the addition of physical design information will reduce the search area sufficiently to achieve small SEM search time.

### 3.1. Fault Extraction

The initial CAFDM implementation is shown in Figure 3. A fault extractor identifies the critical areas where short circuits could occur on the suspect nets, including their locations and layers. We only extract short circuits since open circuits can be modeled reasonably well with the stuck-at fault model.

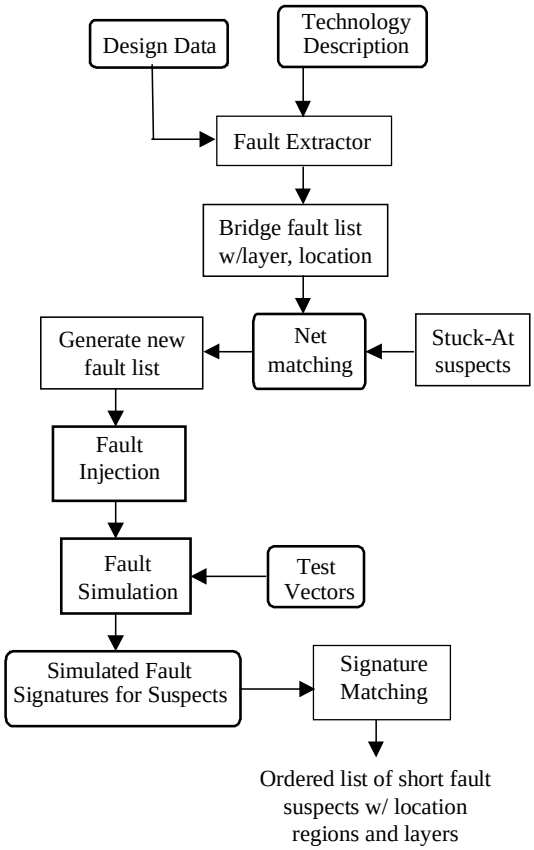


Figure 3. CAFDM Implementation

Previous work on fault extraction has tended to focus on accuracy of the critical areas associated with each fault [11][12]. However these approaches are too slow to extract bridging faults for a large industrial circuit in a reasonable amount of time. One challenge is that hierarchical techniques [11] work poorly on the essentially random routing in an ASIC-style logic design. In addition for the purposes of fault diagnosis it is more important to identify potential bridging faults rather than to accurately predict their probability.

To solve this problem, we have developed a new fault extractor, FedEx, with a goal of extracting all two-node bridging fault sites that are possible for a maximum defect size. This is similar to the goal of the AMD fault extractor [13]. The maximum defect size is chosen to be large enough to so that Fedex enumerates all likely bridges.

FedEx uses a scanline approach to circuit and fault extraction. The geometry is sorted in decreasing y coordinate. A horizontal scanline is passed down over the geometry. As geometry intersects the scanline, it is inserted into associated data structures and nets are extracted. Since devices are not involved with bridging faults, they are not explicitly extracted. However to ensure separate nets, active areas are split by the gate layer.

As extracted geometry leaves the scan line, it is placed into a trailing window, and critical areas computed between nets, using rectilinear approximations. These approximations are accurate to within about 20%. Each bridging fault site includes the layer, bounding box of the critical area, and weighted critical area. There can be multiple sites between a pair of nets, due to the finite size of the analysis window. The critical area weighting is done using the $1/x^3$ defect size distribution, so that when divided by the chip area, the weighted area is the relative probability of failure. In its prototype version, FedEx can extract 500K transistors of logic in about one hour of CPU time and 350MB of memory on an engineering workstation. We expect substantially higher performance with further optimizations.

### 3.2. Fault Diagnosis

For each chip being diagnosed, FastScan uses a set of passing and failing vectors to generate a set of suspect nets. This typically takes less than one minute on million-transistor designs on a typical workstation. The Verilog net names must then be translates to layout net names. The design methodology must provide this mapping. These nets are then matched against the bridging fault list to identify the list of potential bridging faults.

The potential bridging faults and stuck-at faults are injected into the netlist one at a time and simulated with the test vectors. The Wired-AND, Wired-OR, and Wired fault models are used, since any one might best model the fault behavior. The Wired model propagates an X condition on the bridged nodes if they are driven to opposite values.

The Hamming distance (the number of bit positions in all vectors that differ) between the simulation output and the tester output is computed, and sorted in increasing order of distance. We chose Hamming distance rather than number of vector matches to achieve higher resolution. An X output (from a Wired fault) has a distance of 0.5 from a tester 0/1 output. Only the smallest Hamming distance for each bridging fault model is recorded. The result is a ranked list of suspects. For bridging faults the layer and location information for all the fault sites between that net pair are given.

For faults with the same Hamming distance, further ranking is done using the weighted critical area, with faults with larger critical area being ranked first, since they are more likely to occur. Rather than ranking first by Hamming distance and then critical area, it may be more appropriate to use a weighted combination of metrics, so as to rank based on expected search time. Rather than modify the ranking, all this information is provided to the user and left to their judgement. The ranked list of suspects is then used to perform the SEM search.

Limiting the number of reported suspects requires discarding matches below some threshold. For our prototype implementation we did not discard any suspects, leaving it to the user to ignore suspects that were poor matches to the measured data. This also allowed them to stop searching if there was a large jump in Hamming distance from one suspect to the next.

## 3.3. Stuck-At Diagnosis Failure

One problem with using stuck-at diagnosis to filter the possible suspect nets is that it can fail. This happens when the fault behavior is not a good match to the stuck-at model. The result is that the real faulty nets may not be on the reported suspect nets. This situation occurs in 10% or more of faulty chips. In our CAFDM methodology, this results in the Hamming distance of all the stuck-at and bridging fault suspects exceeding a user-defined threshold. This situation could also occur if the simulated fault models are not a good match for the fault behavior, which is a failure for our approach. An example is when multiple faults are present in the chip.

We can handle the case of stuck-at diagnosis failure by using a *backconing* procedure. For each faulty output, the logic *cones* are identified, that is, all the nets for which there is a circuit path to the output. All the potential bridging faults to these nets are identified. This procedure is repeated for all faulty outputs in all vectors. These bridging fault lists are intersected to find the potential bridges in which one side or the other has a circuit path to all faulty outputs. The idea is that this bridge can potentially explain all faulty outputs. The resulting fault list is used as input to the fault simulation procedure described in the previous section. The backconing procedure should be used only when stuck-at diagnosis fails since the backconing procedure takes additional time, and the potential bridging fault list is larger than that obtained from stuck-at diagnosis.

# 4. Experimental Results

A case study using the CAFDM software was carried at out at Texas Instruments (TI), Mixed Signal Products Division, Dallas TX. The target chip for diagnosis was a streaming audio controller. It is a 1M-transistor, mixed-signal design, implemented in a 250-nm technology. This design has about 500K transistors of digital logic and a small amount of analog circuitry, and the remainder memory arrays. The die size is 3.75 mm by 3.75 mm. The digital logic uses a full scan design, which permits FastScan diagnosis. There are 26 953 labeled nets subject to diagnosis. Since the memory arrays are diagnosed separately, they are removed for FedEx analysis.

Using a maximum defect size of 2.25 microns, FedEx extracted 2.8M bridging fault sites. Of these there are 1.6M unique bridging pairs. The average critical area for each fault site is 5 $\mu m^2$, smaller on the diffusion and poly layers, and larger on the upper metal layers, due to their long parallel runs. The largest area is 0.64 $mm^2$, which occurs on metal4 on the global reset line, which itself has 93,376 bridging pairs. This large critical area is caused by the fact that the area is computed as the bounding box containing all critical areas in that region between the net pair at the maximum defect size. Thus an L-shaped critical area will have a bounding box with an area much larger than the actual critical area. The weighted critical area is computed using the actual critical area shapes. We chose not to report more complex critical area shapes so as to avoid overwhelming the user with data. In practice the user will navigate to a suspect net location intersected by the critical area bounding box.

## 4.1. Controlled Experiments

A set of controlled and production experiments were conducted. A focused ion beam (FIB) was used in the

controlled experiments to inject bridging faults at known locations into 13 chips and determine how well the CAFDM software performed in identifying the locations. All faults were injected on the metal4 layer. For the production experiment, 10 chips that failed wafer scan test were analyzed with CAFDM and one sent to failure analysis.

It was not possible to directly measure diagnosis time since the FIBed defect locations were known and equipment limitations prevented automatic navigation to the predicted locations in the production chip. We use the predicted fault site area as a surrogate for search time.

The results of the controlled experiments are summarized in Table 2. Only 9 chips are listed since one chip failed its scan chain test, and three chips erroneously had the FIB injected into a scan chain node. These can be diagnosed using binary search with the SEM. In the table the first column identifies the chip. The second column lists how many of the bridged nets (nodes) appear on the FastScan stuck-at fault list, and their rank on the list. For example, for FIB3, only one of the bridged nets appears on the suspect list at position #6. The third column lists the bridged net ranking after CAFDM. The fourth column lists the critical area bounding box.

Table 2. FIB Experiment Results

| Chip | FastScan™ | CAFDM | Search Area |
|---|---|---|---|
| FIB1 | one node (#1) | bridge (#1) | 27 μm² |
| FIB3 | one node (#6) | bridge (#1) | 263 μm² |
| FIB4 | two nodes (#4, #12) | bridge (#1) | 86 μm² |
| FIB6 | none | bridge (#1) | 91 μm² |
| FIB7 | two nodes (#1, #20) | bridge (#1) | 61 μm² |
| FIB8 | two nodes (#16, #21) | bridge (#1) | 54 μm² |
| FIB10 | one node (#1) | bridge (#1) | 4 μm² |
| FIB11 | two nodes (#1, #2) | bridge (#1) | 57 μm² |
| FIB13 | one node (#1) | bridge (#1) | 400 μm² |

In eight chips, at least one of the bridged nets appears on the stuck-at suspect list. All of these chips were successfully diagnosed by CAFDM, with the real bridge being ranked #1 on the suspect list, with a very close match to actual chip behavior. In seven of those chips the #2 suspect was a much worse match and would not even be considered for diagnosis. In the eighth chip the top two suspects had the same small Hamming distance. The critical area of the real bridge was much larger than the other bridge, and so the real bridge was ranked #1. In all samples the Wired-AND bridging model gave the best results. Neither of the bridged nets appeared in the stuck-at suspect list for chip FIB6. Backconing successfully identified the bridge.

We are interested in how CAFDM reduces the SEM search area. The potential bridges to the stuck-at suspects for FIB3 are highlighted in Figure 4. The arrow points to the bridge location. As can be seen, the suspect nets extend over too large an area to search by SEM. Figure 5 highlights the top-ranked bridged nets found by CAFDM. The cross in the middle of the figure is the bridge location. One net is about 3 mm long, and would take a while to search by SEM. But since the two nets are adjacent in only a small region, the critical area is only 263 $\mu m^2$, about 5 fields of view in the SEM. It is also predicted to be on metal4. So location and layer information greatly reduce the search area.

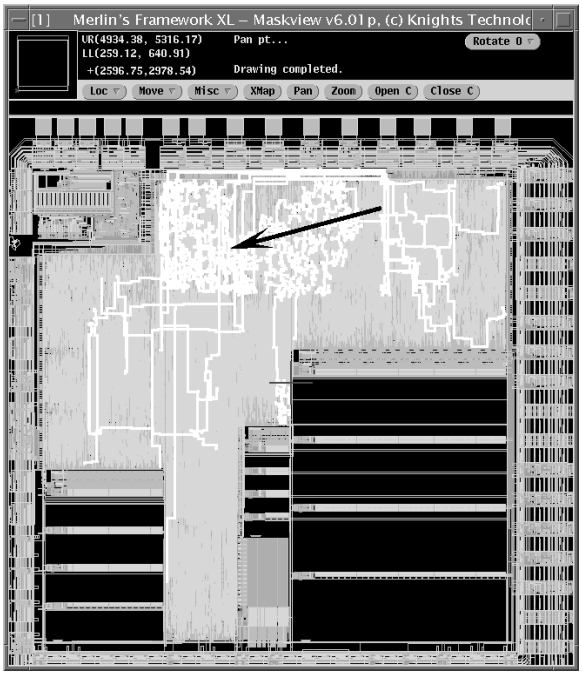


Figure 4. FIB3 potential bridging fault nets

In the controlled experiment, CAFDM was able to achieve all of our metrics. We assume the search time would be very small due to the small search area. However these experiments were on FIBed bridges that have low resistance, and are well modeled by one of our fault models. Results for real defects are unlikely to be as good.

## 4.2. Production Experiments

The production experiments were conducted on 10 devices that failed wafer scan test. These devices passed all other wafer tests. Out of the 10 devices, 3 fail only under maximum voltage conditions and are being reserved for later analysis. CAFDM analysis was performed on the remaining 7 devices, predicting a small search area.

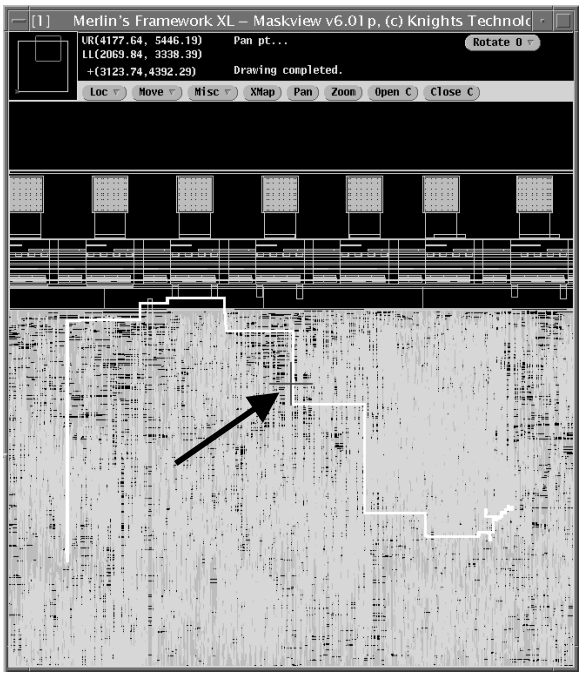


Figure 5. FIB3 CAFDM predicted and actual bridging fault location

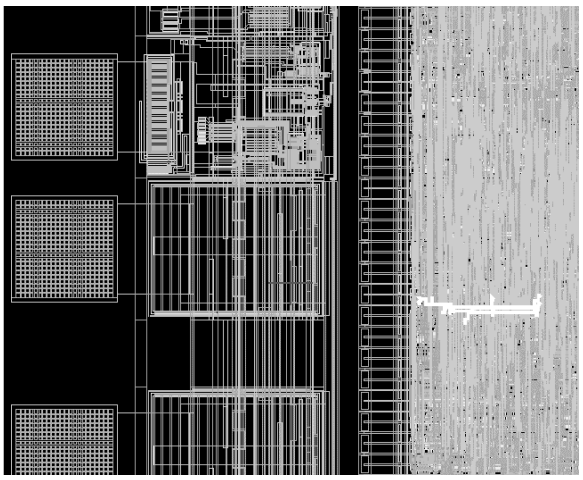

Figure 6. CHIP4 potential bridging fault nets

CHIP4 was selected for failure analysis. In this device two bridges were ranked as the top candidates, explaining most failing vectors while other candidates explained no failing vectors. The bridged lines are shown in Figure 6. The first bridge is predicted to be on the metal2 or metal3 layers, while the second bridge is predicted to be on the metal1 or poly layers. As can be seen, the search region is small enough to perform physical failure analysis (PFA) within hours.

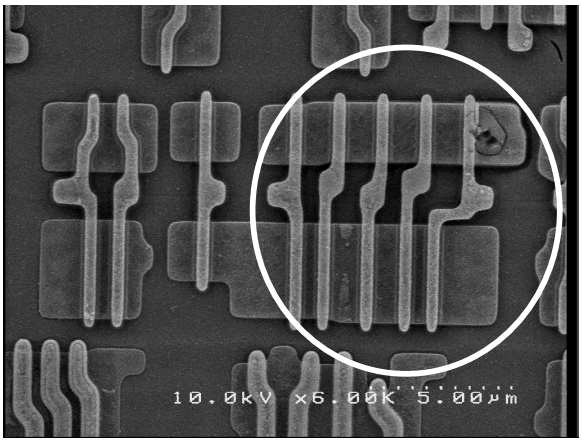


Figure 7. Defects observed at fault site

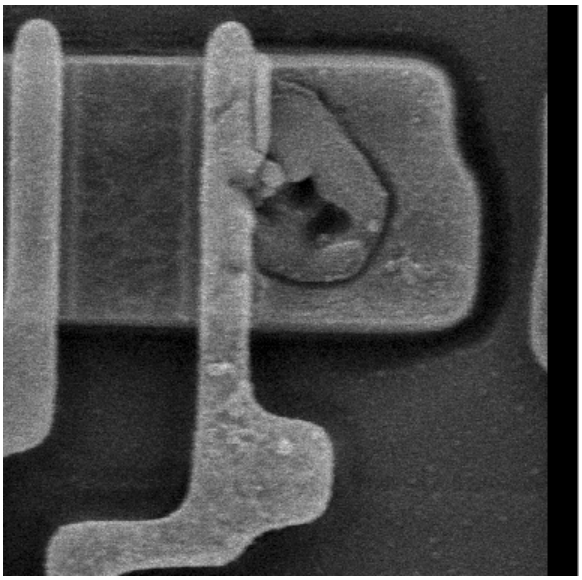

Figure 8. Close-up of gate-diffusion defect

Failure analysis of CHIP4 was promising, but inconclusive. A misunderstanding and deprocessing problems resulted in some metal layers being removed before they could be completely inspected. Thus it could not be determined if any defect was present in the predicted locations on some of the predicted layers. However as Figure 7 shows, defects were observed in a nearby gate, with close-ups in Figures 8 and 9. These were not at the predicted locations or on the predicted layers, but it is very unlikely for independent defects to be near the predicted fault site. So we believe that these defects are associated with the observed faulty behavior. We plan on performing failure analysis on the remaining devices to gain a better understanding of CAFDM performance on production defects.

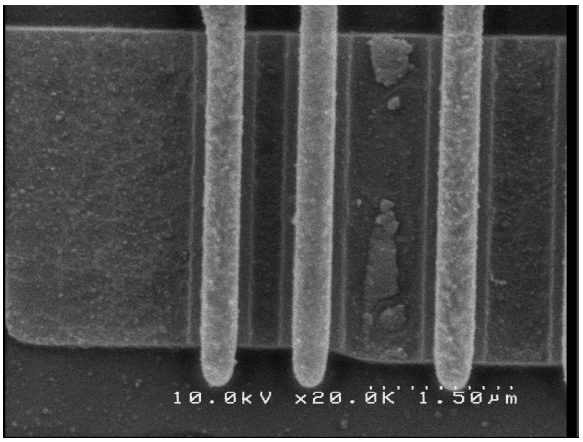


Figure 9. Close-up of material observed on diffusion

## 5. Conclusions and Future Work

In this work we have described a computer-aided fault to defect mapping (CAFDM) approach to defect diagnosis, that combines physical design and electrical test information to localize the fault and thus reduce diagnosis time. We were able to achieve our project metrics on our controlled experiments and initial results from our production experiments are promising. More production experiments are needed to fully evaluate the CAFDM approach.

Our experience has identified several improvements that need to be made to CAFDM to achieve project metrics in a production environment. One is to include the dominant bridging fault model. A recent sutdy [14] showed that 53% of all bridging faults behaved as dominant faults. This can also be seen in that half of the stuck-at diagnoses in the FIB experiments identified only one of the bridged nets as a suspect. In a dominant fault, the dominant net will not be identified as suspect. An open question is whether more complex, and more costly models, such as voting models, are required to achieve sufficient diagnostic accuracy on most bridges.

Another improvement is to reduce the per-chip CAFDM diagnosis time. Most nets are small, and so have few bridging faults that must be simulated. But some of the suspect nets in our production samples have hundreds of potential bridges, and simulating all of these can take 5-10 CPU hours. The solution we are pursuing is to modify the netlist so that the bridging behaviors look like stuck-at faults, and the stuck-at diagnosis engine can accurately rank suspects. This has the advantage that bridging and stuck-at behaviors are considered simultaneously. The pattern fault capability of TestBench avoids the need for netlist modification, since it can model many bridging fault behaviors.

## Acknowledgments

This work was funded by International Sematech under project DRTB002 and by an internship for Zoran Stanojevic in the Design and Test Technology Group, Mixed Signal Products, Texas Instruments, Inc. This work benefited from the advice and support of Craig Force of Texas Instruments, and Sagar Sabade, a Texas A&M student interning at TI.

## References

[1] Semiconductor Industries Association, *International Technology Roadmap for Semiconductors*, 1999.

[2] H. Balachandran, J. Parker, G. Gammie, J. Olson, C. Force, K. Butler and S. Jandhyala, “Expediting Ramp-to-Volume Production”, *IEEE International Test Conference*, Atlantic City, NJ, September 1999, pp. 103-112.

[3] H. Balachandran, J. Parker, D. Shupp, S. Butler, K. Butler, C. Force and J. Smith, “Correlation of Logical Failures to a Suspect Process Step”, *IEEE International Test Conference*, Atlantic City, NJ, September 1999, pp. 458-466.

[4] C. Hawkins, J. Doen, A. Righter, and F. Ferguson, “Defect Classes - An Overdue Paradigm for CMOS IC Testing”, *IEEE International Test Conference*, Washington DC, 1994. pp. 413-425.

[5] S. Millman, E. McCluskey, and J. Acken, “Diagnosing CMOS Bridging Faults with Stuck-at Fault Dictionaries”, *IEEE International Test Conference*, Washington DC, 1990, pp. 860-870.

[6] D. Lavo, T. Larrabee, B. Chess, and F. Ferguson, “Diagnosing Realistic Bridging Faults with Single Stuck-at Information”, *IEEE Transactions on Computer-Aided Design of Circuits and Systems*, 1998, pp. 255-268.

[7] J. Saxena, K. Butler, H. Balachandran, and D. Lavo, ”On Applying Non-Classical Defect Models to Automated Diagnosis”, *IEEE International Test Conference*, Washington DC, October 1998, pp. 748-757.

[8] D. Lavo, T. Larrabee, B. Chess and I. Harinto, "Probabilistic Mixed-Model Fault Diagnosis," *IEEE International. Test Conference*, Washington DC, October 1998, pp. 1084-1093.

[9] S. Naik and W. Maly, “Computer-Aided Failure Analysis of VLSI Circuits Using IDDQ Testing,” *IEEE VLSI Test Symposium*, Atlantic City NJ, April 1993, pp. 106-108.

[10] V. Sar-Dessai and D. M. H. Walker, “Resistive Bridge Fault Modeling, Simulation and Test Generation,” *IEEE International Test*

*Conference*, Atlantic City NJ, September 1999, pp. 596-605.

[11] D. Gaitonde and D. M. H. Walker, "Hierarchical Mapping of Spot Defects to Catastrophic Faults - Design and Applications", *IEEE Transactions on Semiconductor Manufacturing*, vol. 8, no. 2, 1995, pp. 167-177.

[12] S. Zachariah, S. Chakravarty and C. Roth, "A Novel Algorithm to Extract Two-Node Bridges", *ACM/IEEE Design Automation Conference*, Los Angeles, CA, June 2000, pp. 790-793.

[13] R. Fetherston, I. Shaik and S. Ma, "Testability Features of the AMD-K6 Microprocessor", *IEEE Design and Test of Computers*, July-September 1998, pp. 64-69.

[14] S. Ma, I. Shaik and R. Fetherston, "A Comparison of Bridging Fault Simulation Methods", *IEEE International Test Conference*, Atlantic City NJ, September 1999, pp. 587-595.