

Test Strategies For a 40Gbps Framers SoC

Hans T. Heineken and Jitendra B. Khare

Sacramento Design Center
Ample Communications Inc.
Sacramento, CA 95827

Abstract

This paper describes DFT/DFD/DFM strategies implemented on a 40Gbps framer chip. The device is a 1500 pin, over 10M gate SoC with multiple PLLs/DLLs and 2.5GHz IOs. Some novel techniques were required to ensure quality and manufacturability.

1. Introduction

As IC technology shrinks, designers are putting highly complex functionality on a single chip [1]. In the ASIC world, System on Chip (SoC) designs are now increasingly common, especially in the communications area. Telecom designs are combining functions performed by multiple chips in the previous generation, leading to increased system efficiency and decreased costs. Current technologies have resulted in telecom chips with greater than ten million gates, analog and digital IP cores, large memories and high speed interfaces. In addition, these SoCs have large pin counts and complex, expensive packages.

Testing telecom SoCs has multiple problems. First, large pin counts and high speed interfaces require expensive testers not readily available in test houses. Even if available the costs can be prohibitive for fabless companies that out-source testing. As a result, quality test strategies must be designed to enable chip testing on less expensive main-stream testers.

Second, telecom SoCs can cost thousands of dollars each, and go into systems costing \$50,000 or more. In such cases, chips are expected to have field escape rates of less than 100 DPM. This is a major hurdle to overcome: state-of-the-art SoCs tend to have low yields, implying lower reliability and higher chances of test escapes [2,3,4,5]. Hence, to ensure low DPM numbers, two things are necessary: design for manufacturability (DFM) methods to improve yield; and design for testability (DFT) strategies to rigorously test die in order to weed out escapes.

Third, the complexity of SoC devices makes it harder to bring up designs on the bench for system-level validation. However, given market pressures, allocating several months for design debug is not possible. This is especially true if a new spin is necessary to make a device samplable. In order to shorten the validation process to two to three

weeks, it is necessary to implement Design for Debug (DFD) features on chip.

This paper discusses the DFT/DFD/DFM features that were implemented on a large telecom SoC, internally referred to as Blackbird. It also describes the various area and design complexity trade-offs that went into the design process. Section 2 begins with an overview of the Blackbird SoC design. Section 3 talks in detail of the various DFT/DFD features and their trade-offs. Section 4 describes the DFM methods implemented on the chip to improve yield and reliability. Conclusions are provided in Section 5.

2. SoC Overview

The Blackbird is a STS-768 SONET framer, designed for 40Gbps packet-over-SONET (POS) applications. The device can also operate as four STS-192 framers for quad 10Gbps operation, which enables a smooth migration from 10Gbps to 40Gbps with minimal changes to the line card. The chip also includes data engines for PPP (Point-to-Point Protocol) encapsulation and HDLC (High Level Data Link Control) data framing, both in the 40Gbps and 10Gbps modes. In traditional non-SoC designs, framer and data engine functions are performed by two different ICs.

In a typical application, the device interfaces to four network processors on the system side, and an optical module on the line side (Figure 1) which connects to a SONET ring.

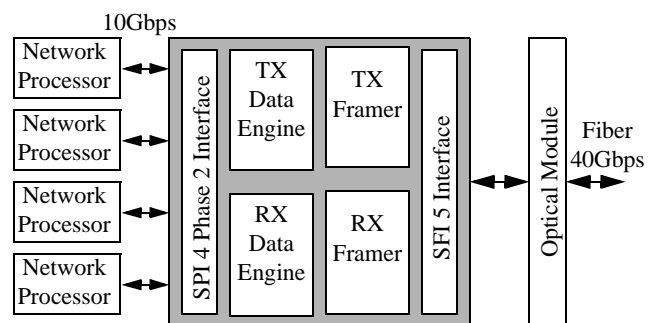


Figure 1. Blackbird block diagram.

The system side interface is based on the SPI-4 Phase 2 [6] specification, and consists of 64 LVDS inputs and outputs operating at 800 MHz data rate (400MHz DDR). The

optical interface adheres to the SFI-5 [7] standard. On both the transmit and receive side, this interface consists of 4 2.5GHz clocks and 20 IOs operating at 2.5GHz. The transmit interface operates off a single 2.5GHz PLL. However, due to board-level skew tolerance built into the SFI-5 specification, the receive side consists of 4 PLLs and 20 DLLs, each operating at 2.5GHz. The device has over 10 million gates, and about 180Kbytes memory on chip. It is manufactured in a 9-metal 0.13um CMOS process. The asic has a total of 1500 pins, and is packaged in a 45mmx45mm Flip Chip BGA. The chip power consumption is 8 Watts. Figure 2 shows a die photo of the blackbird.

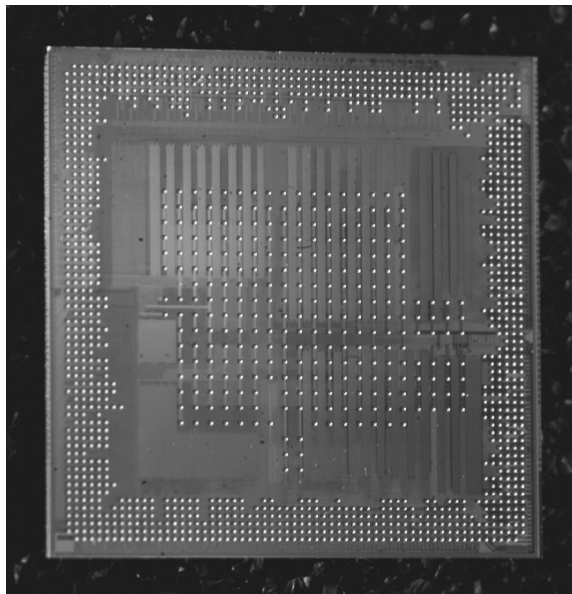


Figure 2. Blackbird die photo.

3. Design for Test and Debug Features

As indicated in Section 1, a rigorous test methodology was needed to ensure that the Blackbird met DPM requirements. This section discusses the Blackbird's DFT and DFD implementation in detail.

3.1 Scan and Memory BIST

To meet DPM requirements, full scan methodology was an obvious choice for the SoC, especially given that the bulk of the clock domains were under 200MHz (i.e timing was less of an issue). However, the design with its 371,000 flip-flops taxed the limitations of the ATPG and gate level simulation tools.

The final design used 85 scan chains and test pattern generation took 6 days on a 64-bit 400MHz Ultra-Sparc machine. The resulting fault coverage was over 99.5%. Due

to the presence of multiple clock domains, static timing could not be used to verify scan vectors on the final delay-annotated gate-level netlist. Instead, gate-level simulation was employed, taking close to 10 CPU days. The use of full scan added 3% to the random logic area.

Standard off-the-shelf memory BIST controllers were used to test the memories. Each block had its own independent controller, programmable via the JTAG interface. All memories on chip were 2-port FIFO structures with ports operating asynchronously. Since most testers are unable to provide two asynchronous clocks, memories were not tested at speed.

3.2 Iddq

Iddq has become an established test methodology in the ASIC world to catch defects and reliability problems not caught by other test methodologies [8,9,10,11]. However, the use of Iddq for the Blackbird presented a problem. Two types of leakage currents exist in the 0.13um manufacturing process - subthreshold currents for transistors in the off state, and gate leakage for on transistors. Given the number of transistors on the chip (greater than 50M) it was likely that the leakage current would swamp any Iddq information.

The effectiveness of Iddq can be significantly enhanced by the use of current signatures [12,13]. However, it was uncertain whether the size of the current signature steps could be adequately resolved in the presence of all the background current in the Blackbird.

To overcome this potential problem, it was decided to break the power grid on the die into multiple power grids and to isolate them from each other. As is typical, the analog power grid was separated from the digital grid. In addition, the digital power grid itself was broken up. The design had nine digital sub-blocks, and blocks were grouped into the various power grids. The top level circuitry was also divided into these grids. Each grid can be operated independently for Iddq, thereby reducing background current and enhancing fault detectability.

Breaking up the power grid required considerable work at the design level. First, the gate-level netlist had to be broken into multiple power grids in order for the LVS verification to work. Second, a complex IR-drop analysis had to be performed to ensure that voltage levels did not change abruptly between connected gates working on different grids. Third, considerable effort had to go into the package design to maintain the multiple grid scheme at the package-level. This required close interaction between RTL and package designers, and was possible only because package design was done in-house. Finally, Iddq patterns had to be generated independently for the logic associated with each grid.

The multiple power plane strategy took time and effort to develop, but is now incorporated into a standard design flow for use in future products.

3.3 Functional Test (Tester on Chip)

Telecom ICs generally consist of a transmit path and a receive path. As a result, Pseudo Random Binary Sequence (PRBS) data generators and verifiers are commonly used in telecom designs to test the design at both the chip and system level [14]. By employing loopbacks between transmit and receive paths at various points in the system or chip, problems can be isolated to a particular die or to a functional block within a die.

Such PRBS generators and verifiers can be used for at-speed functional testing. However, the nature of this solution does not allow for an exhaustive functional test. In addition, PRBS systems give only limited visibility into design problems (it can isolate failing blocks, but not failing logic). As a result, it was decided to build a more complex data generator and verifier system into the Blackbird.

3.3.1 Data Packet Generator (DPG)

The data generator on the Blackbird is designed to generate HDLC packets instead of PRBS data. This allows the chip to be tested under realistic data conditions. The packet generator is located at the beginning of the data path, right after the SPI-4.2 interface, and before the transmit data engines (Figure 3). It is highly programmable and can generate packets of various sizes, forms and patterns. The sizes can take the following options:

- *fixed size packet* - size of packet is fixed and specified by the user.
- *incremental size packet* - size of the generated packet increments with each clock cycle (by 4 bytes for 10Gbps; 16 bytes for 40Gbps).
- *pseudo random size packet* - packet size varies as set by a 32 bit PRBS.

The patterns within the packets can take on several formats. The basic pattern size is 32 bits for 40Gbps and 8 bits for the 10Gbps modes. The patterns can be of three types:

- *fixed patterns* - patterns with a user specified value.
- *incremental patterns* - pattern values incremented with each clock cycle.
- *pseudo random patterns* - patterns generated by a 32 bit PRBS.

The inter-packet gap is also programmable, as is the number of packets to be generated. A continuous mode also exists where the packets are continuously generated.

All the above options were extremely useful for design

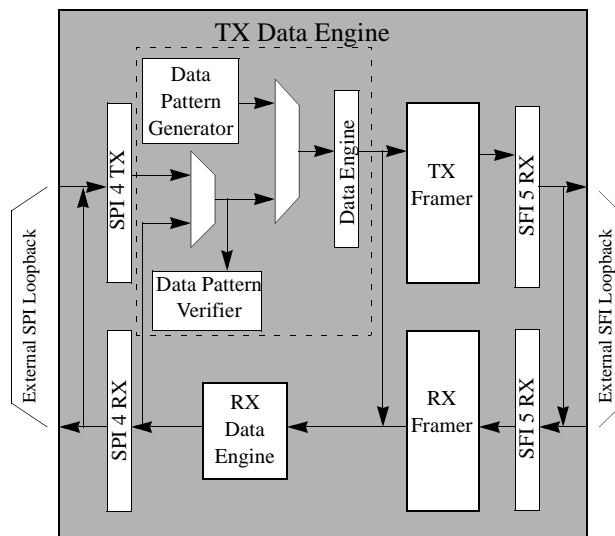


Figure 3. Data Packet Generator/Verifier Loopbacks.

debug. The fixed pattern mode was used to bring up the chip on the bench in two days. Other modes were used to test chip performance for small and large packets, and different data rates. The incrementing pattern mode was instrumental in isolating signal coupling failures in the system. In total, about seven days were required to bring up the chip on the bench.

Functional testing is performed using the random packet, random data mode. Preliminary results indicate that this test is able to catch failures not detected by other tests.

3.3.2 Error Injector

The packet generator has an option to introduce simple bit errors in the generated patterns. The intentional errors change the patterns slightly from their original values as expected by the verifier. This option is used as a simple test to determine whether the verifier is functioning and is able to measure bit error rates accurately.

The die also has a more sophisticated error injector on chip. This module injects errors directly into the SONET frames. Errors can be injected randomly within a frame or on a given bit or byte, and can be repeated across bytes or frames. An error can also be injected in specific locations of the frame, e.g., in the Section Overhead, Line Overhead, or Synchronous Payload Envelope. In addition, the duration (number of cycles) in which the errors are injected is controllable. The primary purpose of the error injector is not so much to test the packet verifier, but to provide a mechanism to test bit error rates at the system level.

3.3.3 Data Packet Verifier (DPV)

The purpose of the packet verifier is to receive the

packets produced by the generator, and to measure error bit rates, error byte rates, and number of good packets. In addition, at the system level, it is able to measure the number of times synchronization is lost when data is sent across the SONET ring.

Since data coming to the verifier is not PRBS generated but packets of various types and sizes, the verifier works by incorporating a full second packet generator within its module. This second generator is identical to the original packet generator: both have the same PRBS's for packet size, pattern, and packet gap. When turned on, options programmed for the first generator are similarly applied to the second generator. The verifier also sits after the SPI interface on the transmit side. In this way, it is able to verify data going through both the SFI and SPI interfaces in external or internal loopback.

The second generator, internal to the verifier, switches on as soon as data from the original generator arrives at the verifier. At this point incoming data is compared to the locally generated data. If patterns match within a user defined bit error limit, the verifier goes into "sync" mode. The verifier then starts to derive the bit error rates and byte error rates. (Bit and byte error rates are counted to distinguish between single errors and clusters of errors.)

If the bit error rate goes above a user defined limit for a given number of cycles, then the verifier goes out of sync and assumes the packet is bad. It then attempts to resync with new incoming data. The advantage of allowing sync to occur even with the arrival of errors is that the user can acquire an accurate measurement of data dropping by the communication medium, e.g., fiber. During functional test, however, when there is no fiber, error rates and out of sync rates must be zero. Otherwise the die is rejected.

The packet verifier is much bigger than the generator and has an impact on die size. The verifier gate count goes higher due to two reasons. First, it has to incorporate a full packet generator to be able to sync with incoming data. Second, it has to deal with packed data. The data engines on the transmit side are supposed to pack packets back-to-back to increase the efficiency of the network. However on the receive side, unpacking is done such that inter-packet gaps are not always maintained. This creates problems for the verifier. The internal data bus on the chip is 32 bytes - the data verifier has to look at all the 32 bytes in parallel to weed out idle bytes and identify the next data bytes.

The packet generator and verifier thus act as a built-in tester and can be used for at-speed functional test. This negates the need for expensive testers with large per pin memory and high speed IOs.

The total gate count of the packet generator and veri-

fier is close to 300,000 gates. This is approximately 3% of the standard cell area of Blackbird.

3.4 IO BIST

As indicated in Section 2, the chip has 5 PLLs and 20 DLLs operating at 2.5GHz. In addition, the transmit and receive paths both have 20 IOs operating at similar speeds. While testers working at 2.5GHz are available, most are limited to testing only one or two pins at that speed. Using testers with 40-50 2.5GHz IOs is prohibitively expensive.

To mitigate this problem, an IO BIST feature was incorporated into the design. The BIST can be programmed to send 16 four bit patterns to each SFI-5 output in a continuous manner. On the receive side, the BIST can capture data from each SFI-5 input. Thus, in an external loopback mode, with inputs tied to outputs, all 2.5GHz IOs can be tested simultaneously. Since the clock output frequency is 622MHz (quarter speed), the entire SFI-5 interface can therefore be tested on a standard tester with a maximum speed of 622 MHz.

The IO BIST was also useful for debug - it was used to bring up the SFI interface on the bench within days. It has also caught failures - the complexity of the SFI-5 interface makes it susceptible to parametric failures.

3.5 BurnIn Mode

A burnin mode was added to the chip to ensure reliability [15,16]. In this mode, scan chains are made active through on-chip LFSRs, and the memory BIST loops continuously through its patterns. The overall toggle coverage was estimated to be about 98%.

3.6 Additional Diagnostics

Beside the diagnostic ability provided by DPG/DPV, other diagnostic mechanisms were also built into the chip. The SoC has a 16 bit diagnostic bus to bring out data from various design points. In addition, a very large number of shadow registers were used to capture data at internal points. These registers can be read through the CPU interface.

4. Design for Manufacturability

As seen above much effort was employed enhancing the testability and debug of the product. In addition the manufacturability of the device was also enhanced to increase the product yield. Yield is also related to test escapes [2,4,17,18,19], and the higher the yield the lower the probability of a faulty die reaching the field. Manufacturability enhancements were performed in a number of areas including: relaxed design rules, redundant vias, and metal fill.

4.1 Design Rule Modification

In previous work it has been noted that relaxing (i.e., increasing) the metal pitch in the technology file of the place and route tools can significantly reduce the susceptibility of a design to defects [20,21,22,23,24]. Often this can be done without a penalty in die area. In fact relaxed metal routing pitch tends to force routers to more evenly distribute interconnects across metal layers (rather than concentrating interconnects on the lower metal layers).

Yield increases owing to relaxed metal pitch have been estimated on some designs to be as high as 15% [23]. For this design, the optimal increase in metal pitch without die area penalty was determined experimentally. The subsequent yield increase was found to be over 25%. The main reason for such a large increase is that the yield was low to begin with, owing to the large size of the design.

An additional benefit of the increased metal pitch is the reduction in crosstalk. In this design, crosstalk analysis using commercial tools detected virtually no problems at the block level. Only at the top level was the layout modified for crosstalk, and then only in a few places. Thus timing closure was significantly simplified using the relaxed metal spacing.

Figure 4 shows an example layout with relaxed design rules. Only metal layers 2, 3, and 4 are displayed along with their vias. Note that only metals 3 and 4 use relaxed design rules; metal 2 uses minimum design rules.

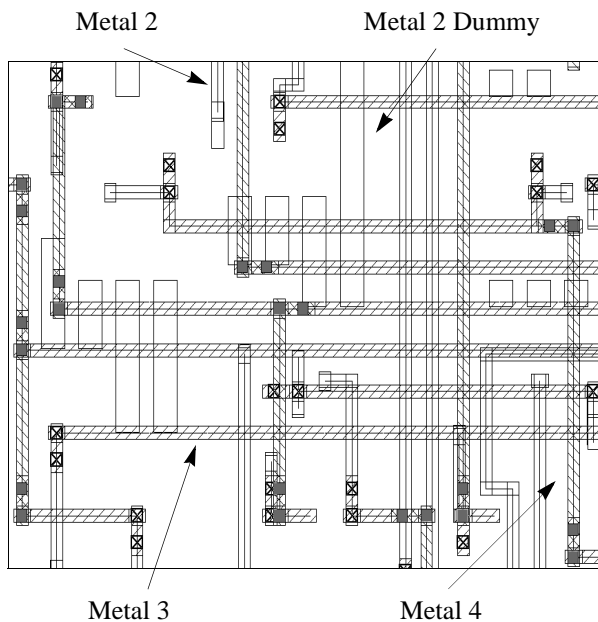


Figure 4. Example layout with relaxed design rules, redundant vias and dummy metals.

4.2 Redundant Vias

Via failures can be an issue in the 0.13 μ m process due to dual damascene copper processing. In order to minimize this problem, it was decided to apply redundant vias liberally. In the analog circuitry much effort was spent manually inserting redundant vias into the layout. In the digital areas, a software program was developed to automatically insert vias after the place and route was finished. The program would look at every via, identify if it was possible to make it redundant and then make the fix. This program was written in-house because at the time of tapeout, commercial options did not exist or worked poorly. The program was able to apply redundancy to 75% of the vias. (For the upper metal layers redundancy was applied to over 90% of the vias.) This number was high mainly because of the use of relaxed metal spacings. Figure 4 shows examples of redundant vias in the metal 2 and metal 3 layers.

4.3 Metal Fill

Inserting metal fill in layouts is done to minimize over or under etching during chemical polishing and is a relatively common practice. This is typically done consistently over the digital circuitry and more sparingly over the analog circuitry. The latter is more susceptible to charges collected on the dummy metal polygons (less of a problem in a copper process than in an aluminum process).

As with the insertion of the redundant vias, the metal fill insertion was done with an in-house program. In the digital areas this program was liberal with its metal insertion. In the analog areas, owing to the high speed nature of the design, the program skipped adjacent layer(s) on those parts designated as sensitive. In very sensitive areas the analog designers were asked to insert their own fill or dummy structures where used. Figure 4 shows examples of metal 2 dummies in the digital area.

5. Summary

This paper describes the DFT/DFD/DFM implementation on a 10-plus million gate, 1500 pin Telecom ASIC. In addition to standard DFT techniques, novel methods were used to ensure high test quality. These included on-chip packet generators/verifiers for at-speed functional testing, and multiple power planes for Iddq testing. In addition, high speed IO BIST was implemented to enable testing on standard testers.

DFM techniques were also used on the ASIC to enhance yield by 25%, and subsequently to improve reliability. In addition, DFD techniques were applied that substantially accelerated system bring-up on the bench, thereby reducing time-to-market

Acknowledgments

The authors would like to acknowledge the efforts of

Ed Lennox, Poly Palamuttam, Ashok Raman, Amal Reddy, Charles Ouyang, and Manuel D'Abreu in the implementation of the test strategies.

References

- [1] W. Maly, "Cost of silicon viewed from VLSI design perspective," *31st. Design Automation Conference*, pp. 135-142, June 1994.
- [2] V. Agrawal, S. Seth and P. Agrawal, "Fault coverage requirement in production testing of LSI Circuits," *IEEE Journal of Solid-State Circuits*, SC-17(1), pp 57-61, Feb. 1982.
- [3] T. W. Williams and N.C. Brown, "Defect level as a function of fault coverage," *IEEE Transactions on Computers*, C-30, 1981, pp 508-509.
- [4] D.D. Gaitonde, J. Khare, D.M.H. Walker, and W. Maly, "Estimation of reject ratio in testing of combinatorial circuits," *Proc. IEEE VLSI Test Symp.*, pp 319-325, April 1993.
- [5] F. Corsi, S. Martino, T.W. Williams, "Defect level as a function of fault coverage and yield," *Proceedings of European Test Conference*, 1993, pp 507-508.
- [6] "System Packet Interface Level 4 (SPI-4) Phase 2: OC-192 System Interface for Physical and Link Layer Devices," *Implementation Agreement OIF-SPI4-02.0*, Jan 2001.
- [7] "Serdes Framer Interface Level 5 (SFI-5): Implementation Agreement for 40Gbps Interface for Physical Layer Devices," *OIF Document oif2001.145.1*, Apr 2001.
- [8] P. Nigh and W. Maly, "Test Generation for Current Testing," *IEEE Design and Test of Computers*, Vol. 7, No. 1, Feb. 1990, pp 26-38
- [9] R.C. Aitken, "Diagnosis of Leakage Faults with Iddq," *J. Electronic Testing: Theory and Applications*, vol. 3, no. 4, Dec. 1992, pp 367-376.
- [10] P. Nigh, W. Needham, K. Butler, P. Maxwell and R. Aitken, "An experimental study comparing the relative effectiveness of functional, scan IDDQ, and delay-fault testing," *VLSI Test Symposium*, April 1997, pp. 459-463.
- [11] P. Nigh, W. Needham, K. Butler, P. Maxwell, R. Aitken and W. Maly, "So what is the optimal test mix? A discussion of the SEMATECH methods experiment," *ITC*, Nov. 1997.
- [12] A. Gattiker and W. Maly, "Current signatures," *IEEE VLSI Test Symposium*, April 1996, pp. 112-117.
- [13] A. Gattiker and W. Maly, "Current signatures: application," *ITC*, Nov. 1997.
- [14] "Error Performance Measuring Equipment Operating at the Primary Rate and Above,". *The International Telegraph and Telephone Consultative Committee, Recommendation O.151*, Oct 1992.
- [15] D.L. Denton and D.M. Blythe, "The impact of burn-in on IC reliability," *J. Environmental Sci.*, vol. 29, no. 1, pp. 19-23, Jan. 1986.
- [16] L.M. Leemis and M. Beneke, "Burn-in models and methods: A review," *IIE Trans.*, vol. 22, no. 2, pp. 172-180, June 1990.
- [17] F. Kuper, J. van der Pol, E. Ooms, R. Wijburg, W. Koster, D. Johnston, "Relation between yield and reliability of integrated circuits: experimental results and application to continuous early failure rate reduction programs," *Proceedings IRPS*, pp. 17-21, 1996.
- [18] C. Glenn Shirley, "A defect model of reliability," *Tutorial Notes, International Reliability Physics Meeting*, 1996.
- [19] W. Kuo and T. Kim, "An overview of manufacturing yield and reliability modeling for semiconductor products," *Proceedings of the IEEE*, Vol. 87, No. 8, Aug. 1999.
- [20] W. Maly and J. Deszczka, "Yield estimation, model for VLSI artwork evaluations", *Electron Letters*, vol. 19, no. 6, pp. 226-227, March 1983.
- [21] C. H. Stapper, "Modeling of integrated circuit defect sensitivities," A. V. Ferris-Prabhu, "Modeling the critical area in yield forecasts," *IEEE J. Solid-State Circuits*, vol. SC-20, no. 4, pp. 874-878, Aug. 1985.
- [22] H. T. Heineken and W. Maly, "Interconnect yield model for manufacturability prediction in synthesis of standard cell based designs," *Proc. of IEEE Custom Integrated Circuits Conference*, 1996, pp. 368-373.
- [23] C. Ouyang, K. Ryu, H. Heineken, J. Khare, S. Shaikh and M. d'Abreu, "Wire Planning for Performance and Yield Enhancement," *Proc. of IEEE Custom Integrated Circuits Conference*, 2000.
- [24] C. Ouyang, H. T. Heineken, J. Khare, S. Shaikh, and M. d'Abreu, "Maximizing wafer productivity through layout optimizations", *Proceedings of VLSI'00*, Jan. 2000.