

*Authors:*

**Ahmet Kemal Goksel** is head of the Microsystems Peripheral Development department at AT&T Information Systems in Holmdel, New Jersey. He is responsible for development of peripheral chips for the WE® 32000 series microprocessors. He received the B.S. in electrical engineering from Robert College in Turkey and the M.S. and Ph.D. in electrical engineering from the University of Virginia. He joined AT&T in 1977.

**Warren T. Sekino** is a supervisor in the Microsystems Processor Development department at AT&T Information Systems in Murray Hill, New Jersey. He works on development of design

verification and computer-aided engineering for VLSI chips. He received the B.S. in mathematics from Utah State University and the M.S. in computer science from the University of Utah. He joined AT&T in 1970. **William W. Troutman** is a supervisor in the Microcomputer Design department at AT&T Bell Laboratories in Murray Hill, New Jersey. He is responsible for design of 32-bit-microprocessor peripheral circuits and ultra-high-speed MOS custom circuits. He received the Ph.D. in engineering, applied mathematics, and theoretical physics from Cambridge University. He joined AT&T in 1966.

## TOOLS AND TECHNIQUES FOR VLSI QUALITY

### Introduction

Product quality assurance is a key element in sustaining leadership in state-of-the-art integrated circuit technology and transcends merely assuring error-free chips; it also includes assuring attractive features, rigorous electrical characteristics, high tolerance of process variations, and consideration for economic manufacturing.

Assuring the quality of silicon integrated circuits is an extremely challenging endeavor, particularly in light of the complexity and size of today's very large scale integrated (VLSI) chips, which have more than 170,000 transistors and match the complexity of the mainframe computers of less than two decades ago. [More than 10,000 transistors on a chip is regarded as VLSI, more than 1000 transistors is large scale integration, more than 100 transistors is medium scale integration (MSI), and less than 100 transistors is small scale integration (SSI).]

To assure quality, AT&T closely couples the VLSI design methodology with a verification process. The methodology is a well structured, top-down process, with well defined levels of abstraction. Each level requires a design implementation with appropriate design tools and a complementary design verification process. Furthermore, regression is employed: each simulation level of the methodology is compared to the previous simulation level before work continues on the next design level. Thus, quality assurance is interwoven in the design methodology and is integral to develop-

**Table I. VLSI Design Process and Design Verification**

Design phase	Model	Tool	Quality assurance
Architectural	Behavioral simulator	C language	Tests, reviews, analysis
Functional	Functional simulator	C, AIDE, PLA tools, checker	Tests, reviews, regression
Logic	Logic model	LAMP, DRAW, checker	Tests, synthesis, regression
Circuit/timing	Circuit model	MOTIS, ADVICE, checker	Tests, regression, analysis, synthesis
Layout	Layout model	Circuit extractor, LTX, CTL-LCC	Analysis
Wafer/chip	Wafer	Test machine	Tests, fault analysis
Integration and system certification	Chip	Test facilities	Tests, regression
Product engineering	Chip	Test machine	Tests, analysis, parameteric characterization

ing a high-quality VLSI chip. Breadboarding, using discrete parts, is not part of this method—all verification is performed on large, complex software models (Table I), which give results more quickly and are highly accurate.

This paper reviews the quality assurance techniques and tools used in the Microsystem Laboratory at AT&T Information Systems and the Silicon Device Development Laboratory at AT&T Bell Laboratories to develop a series of VLSI chips. It describes the design process, emphasizing the tools and techniques to assure the quality of the design. It discusses a highly successful VLSI chip development based on the process.

### Design Process

The design process consists of eight levels: architectural, functional, logic, circuit/timing, layout, wafer/chip, integration/system, and product engineering. Table I illustrates the levels, models, tools, and the methods of quality assurance.

**Architectural Design.** During the architectural design, product goals are set after extensive discussions with customers and a feasibility study of the technology that will be state-of-the-art during the life of the chip. The external view of the chip as well as the major internal features are then specified in enough detail to provide information to logic designers and test engineers.

Extensive reviews of each feature are held by the architects, developers, verifiers, and customers to determine the impact and needs.<sup>1</sup> Thorough analysis of all proposed features is conducted in this phase; if necessary, the features are simulated to determine the performance impact and the feature advantages. Silicon testability issues are explicitly discussed and evaluated as features of the chip during this phase.

A high-level simulator, a behavioral simulator written in a high-level programming language,<sup>2</sup> is implemented to realize the proposed features and to determine the system impact of the chip. Behavioral simulators typically represent the algorithms of the features where it is easy to detect design flaws in the features and establish initial performance goals. The simulator is then integrated with other behavioral simulators of other chips to ensure consistent protocol.

Design verification tests are generated during this phase to verify the complete set of features on the behavioral simulator. To obtain good coverage, the tests are generated from a

---

test specification document and the tests results are verified through a list of expected results. The tests, test results, and test specifications are thoroughly reviewed and inspected. Though this is a manual process, it is the simplest, most flexible, and most accurate method. The tests have been invaluable in identifying problems in the features. The output of the simulator is also used as a “golden yardstick” for verification of other models later in the design phase.

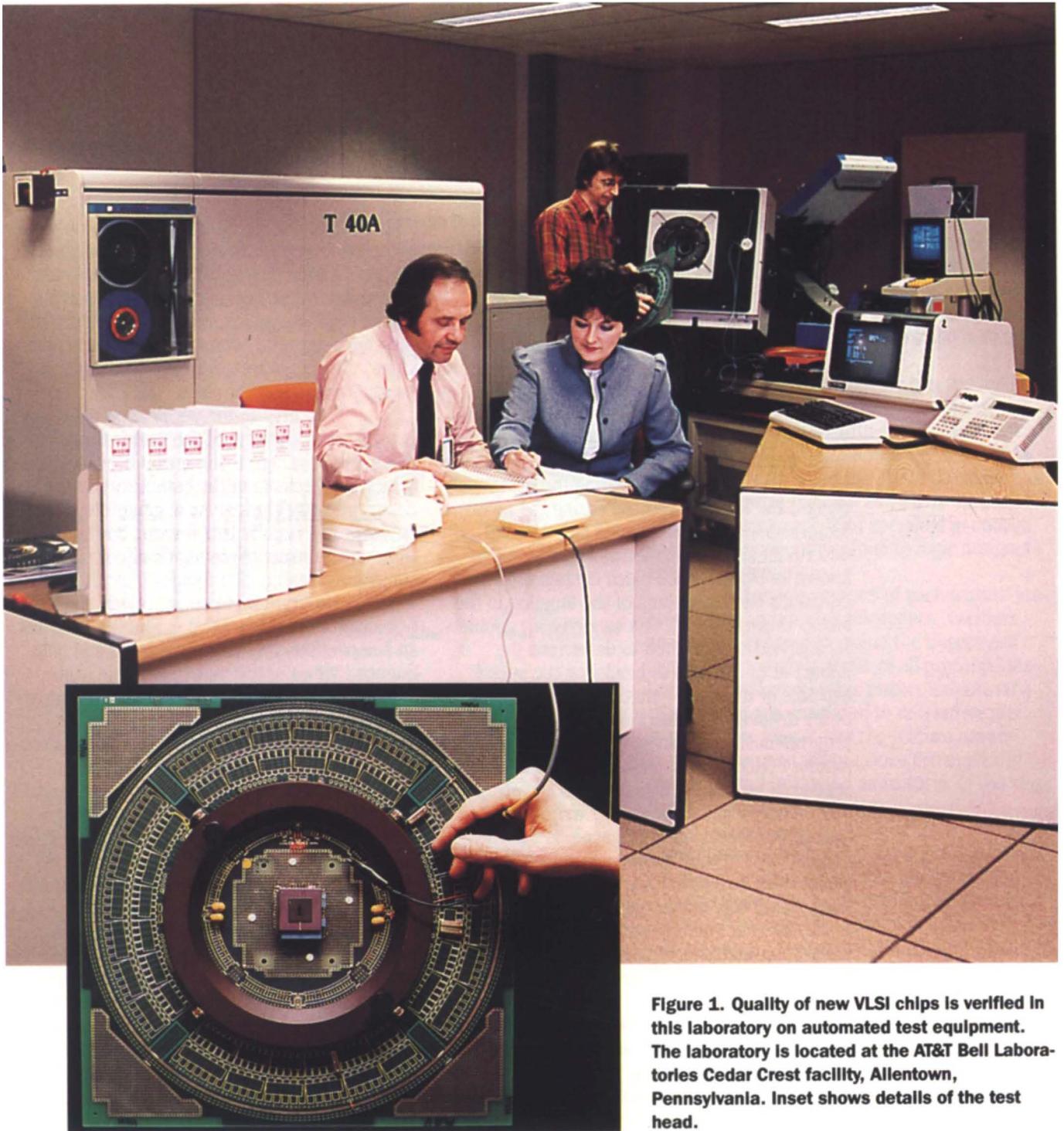
**Functional Design.** During the functional design, the features created during the architectural design are subdivided into detailed descriptions, and specified as logical entities known as blocks. Each block contains the logic equations and algorithms of the function in the block. In parallel with this subdivision, geometric analysis is conducted to determine the impact of the functional blocks on the speed and size of the chip. Functional blocks are generally divided into several logic blocks: programmable logic array (PLA), read-only memory (ROM), data paths [i. e., regular logic, registers, arithmetic logic unit (ALU), etc.], and random logic. A simulator, written in a high-level programming language, is the embodiment of the functional specification and is called the functional simulator. Quality assurance during this design phase is intended to assure correct implementation of the algorithms and partitioning of the features. The quality is assured through reviews, analysis, and regression.

The functional simulator is verified by regressing the functional simulator against the architectural simulator, i. e., comparing results of identical tests. The functional simulator is further tested to stress the implementation of the blocks and the interfaces between the blocks.<sup>3</sup> For example, data path tests would be

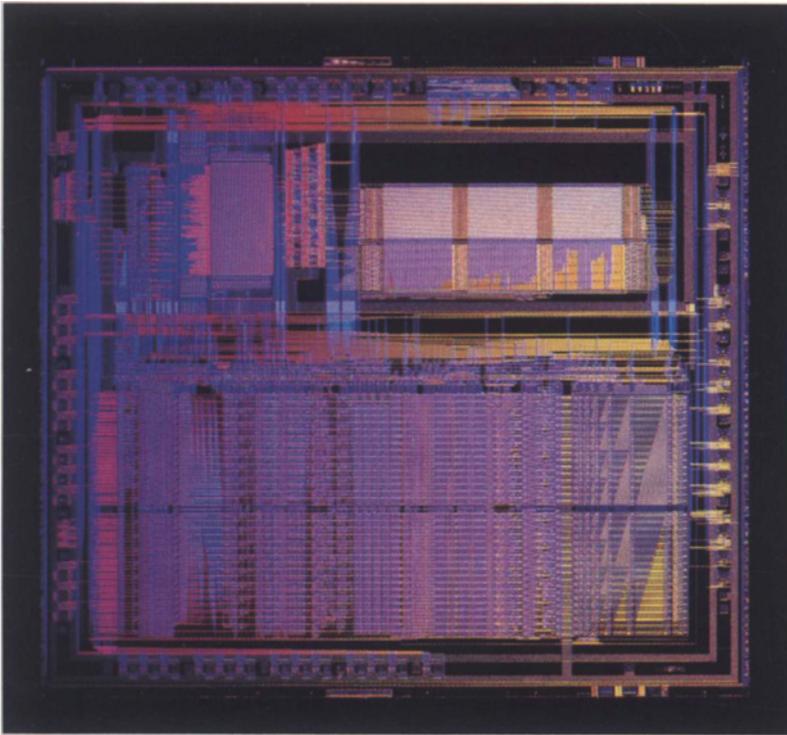
heavily stressed during this phase. To obtain complete test coverage of the implementation, the verification engineer must gain thorough knowledge of the design through extensive design reviews and inspections. To ensure that the chip is correctly communicating with other chips, the chip is tested in a system environment along with other chips.

Essential to the verification operation is a full understanding of the test coverage. Complete coverage is the goal. Often this is not achievable because of the complexity of the design implementation. As a substitute, the engineer has regular test reviews and inspections and a means of automatically checking simulation results. The tool for checking is a software program that postprocesses simulation results of protocol tests. The test reviews and inspections rely on independent and uninfluenced views of the test coverage. Often these reviews reveal additional tests and prove to be invaluable to the process.

**Logic Design.** After the functional specification has been designed and verified, the logic at the gate level for each block is designed and verified. Quality assurance during this phase is accomplished through test, regression, and synthesis. The design and generation is accomplished through one of two methods: synthesis or schematic capture. By synthesis, some random logic in the control section is directly generated from a functional simulator through a logic synthesis tool. The logic design is also captured through a schematic capture system, known as DRAW, and logic generated by the capture system—the latter requiring manual intervention. To verify its correctness, each logic block is placed in the corresponding functional block in the functional simulator and the results of tests are compared. The logic simulator is LAMP. The complete logic model is



**Figure 1. Quality of new VLSI chips is verified in this laboratory on automated test equipment. The laboratory is located at the AT&T Bell Laboratories Cedar Crest facility, Allentown, Pennsylvania. Inset shows details of the test head.**



**Figure 2. The WE<sup>®</sup> 32106 math accelerator unit contains 150,000 transistors on one silicon chip.**

then regressed against the functional simulator to ensure that the logic accurately reflects the functional design.

**Timing/Circuit Design.** Verifying the timing of a chip is extremely important to the design of the chip. The timing is verified through analysis, regression, and synthesis. The timing analysis of the VLSI design is accomplished primarily with MOTIS, a multilevel timing simulation system, and ADVICE, a circuit level simulator.<sup>4-6</sup> The synchronous delay feature of MOTIS is used to locate analytically all suspected slow paths between pairs of clocked points. In addition, a small set of tests are run on MOTIS to exercise additional paths

suspected of being slow by the logic designers. All paths indicated as being slow, as well as other asynchronous paths, are then simulated with ADVICE. Additional data that lend accuracy to the timing verification are the location of the gates and the sizes of the transistors.

Clock lines on a VLSI chip require special attention during both layout and circuit analysis. Clock lines are usually routed separately from all other conductors on the chip. For this routing, special attention is paid layout configurations that may induce clock skew. Tight control of clock skew is generally achieved on VLSI chips by careful control of the layout and accompanying detailed ADVICE analysis.

Circuit verification is provided by a series of tools that includes graphical layout editors. These systems are run hierarchically, with "coarse" design-rule checks being run before the checks for "finer" design rules.

**Layout Design.** After logic design the chip must then be translated into a specific geometric pattern of transistors and wires. The end product of the layout effort is the detailed circuit information that is required to write the x-y mask. During the initial stages of layout design, in conjunction with the functional design, the final floor plan of the chip is configured. The floor plan is divided into several logic macros: PLA, ROM, data path (i.e., regular logic, registers, ALU, etc.), or random logic. While layout for memories and PLAs is done with a set of automatic tools, the data path and random logic are done with interactive tools in the style of gate matrix symbolic design with a parameterized cell generator.

The parameterized cell generator contains design rules of various technologies and can, therefore, produce families of logic gates

where each set of gates is tied to a specific design-rule generation.<sup>7</sup> Various tools are used to check violations of design rules internal to the cell. A set of tools is then used to assure that these cells are connected with respect to the design rules of the particular technology. The blocks are interconnected with LTX,<sup>8</sup> an interactive routing tool. After the chip has been completely interconnected, a final check for design rule violations is made on the entire chip.

A layout verification tool, CTL-LLC, is used to verify the layout design against logic design. As the layout of the gates develops, and as cells are interconnected to realize larger logic functions, it is essential to guarantee that the logic as laid out corresponds exactly to the logic designer's intent. The circuit description is extracted from the layout design by using a circuit extractor. CTL-LLC then takes as input the circuit description and compares this against the logic description.

**Wafer/Chip.** Until this phase of the design process, the chip design is realized as various software models. The objective during this phase of the design is to identify fully operational silicon chips. The operational chips are identified by running tests with high fault coverage over the chip. Faults are gates that are stuck at one or zero, or that have an unconnected input.

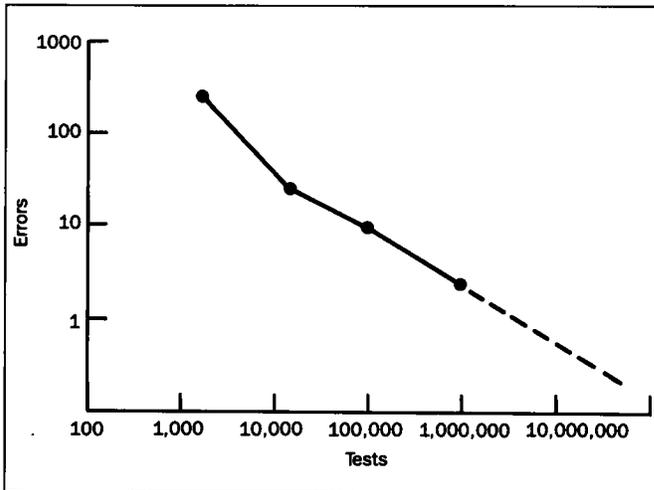
Fault coverage is the statistical analysis of the confidence level of locating all faults. Typically, these tests find about 90 percent of the possible faults. Past data indicate that this fault coverage detects virtually all defective chips.

**Integration and System Certification.** Integration and system test is a continuation of the presilicon verification process except that the chip model is a silicon model. Because the

actual speed of a chip is orders of magnitude faster than the simulation models, the chip is exhaustively tested during this phase (Figure 1). Often, millions of tests are used to stress all aspects of the chip. Integration testing is the process of verifying that the silicon does indeed accurately represent the requirements as defined during the architectural design. The silicon is placed in a test facility, the tests are downloaded from a host system, and the chip is exercised using these tests.

System test, on the other hand, is the final certification of the chip before it is committed to production manufacturing. The purpose of system tests is to instantiate the customer's view of the chip design and show that the chip operates properly within a fully operational environment. Typically, the chips are tested in customer-like environments where the chip is subjected to system loads and stress loads. This would include stand-alone and system environments. Alpha and beta tests are an integral part of this phase. (In alpha tests, the chip is released to a single "friendly" user. In beta tests, the scope of chip use is expanded but still carefully controlled.) At completion of this phase the confidence level is extremely high that the chip is free of errors in layout and logic, and it can be stated that the chip is certified for production.

**Product Engineering.** To ensure economic manufacturability of new chip designs, extensive characterization of the silicon is accumulated. The characterization data provide assurance that the design is not affected by variations in fabrication parameters over the lifetime of the product. Generally, a sample of at least 100 wafers (representative of all parts of the fabrication window) is tested and the results studied to determine if the yield of the



**Figure 3. As the design of a chip is improved, errors become harder to detect. As a result, many more test simulations are needed to find fewer and fewer errors.**

design is sensitive to any fabrication variation. Functional, timing, and input-output parameters are automatically plotted to identify deviations from expected values. Of particular interest are the severe environmental conditions, i.e., lowest voltage, highest temperature, and worst-case clock skew. The design is proven-in when the results have been thoroughly studied and accepted.

After the design has been proven-in, the chip design is transferred into production. As the production chips are tested, the number of good chips (yield) is constantly scrutinized to assure consistency with well-accepted yield models. If discrepancies occur between expected and measured yields, then the yield loss is investigated beginning with historical data. As the processing of a particular VLSI design matures, major yield loss defects are identified, and corrective actions are implemented as soon as possible. Special attention is also given to problems associated with test program hardware and accuracy and integrity

of the automatic test systems that are used to determine probe and package yield.

Another facet of high-quality product is the careful study of field returns. Field returns are retested and examined, and a complete failure-mode analysis is conducted to determine the cause of failure. If the cause is identified in the fabrication process, then immediate corrective steps are outlined and acted on.

#### **An Application: WE 32106 Math Accelerator Unit**

The development of the WE 32106 math accelerator unit (MAU) is a product of the design methodology and its companion design verification, as described in this paper. The MAU (Figure 2) is a chip that meets the full IEEE floating-point standards. It has 150,000 transistors and is capable of a half-million floating-point operations per second. The development cycle, from architectural design through system certification, was completed in 13 months. As the design of the chip approached actual silicon, the rate of finding errors decreased, and the errors became more difficult to pinpoint because of the second- and third-order effects of the causes. The rate of finding errors was the inverse of the number of tests that were used (Figure 3).

It is important to note that all errors were found prior to system certification and that the chip was free of errors in both layout and logic on the first design iteration. Quality assurance methods were exhaustively applied at all levels; this included testing, reviews and inspections, and extensive use of verification tools. Finding errors in this manner benefited not only the quality of the chip, but also the cost and schedule of the chip development. The MAU development reinforces the adage: "Finding errors early in the design cycle

---

directly impacts the development cost and schedule." We can say that the quality of the MAU product can be directly attributed to the thorough design verification at all levels of the design methodology.

#### Summary

The tools and techniques to assure quality in the VLSI design process are described in this paper. Particular attention is paid to ensuring a robust design through verification, synthesis, and regression. The tools and techniques have been successfully applied to many chips such as the WE 32106 math accelerator unit, as well as other projects currently under development.

#### Acknowledgments

The work described in this document represents the effort of many people in several different organizations who have developed these tools and techniques. To these people we owe a debt of gratitude.

#### References

1. D. E. Blahut, R. H. Krambeck, H-F. S. Law, H. Shichman, and H. C. So, "Hierarchical Design Methodology for a Single Chip 32-bit Microprocessor," 1982 International Conference on Circuits and Computers, New York, September 1982.
2. B. W. Kernighan and D. M. Ritchie, *The C Programming Language*, Prentice-Hall, Inc., Englewood Cliffs, N. J., 1978.
3. M. J. Myer, P. Agrawal, and R. G. Pfister, "VLSI FSM Design System," *Proceedings of the 21st Design Automation Conference*, June 1984.
4. V. B. Agrawal *et al.*, "Mixed Mode Simulation," *Proceedings of the 17th Design Automation Conference*, Minneapolis, June 1980, pp. 618-625.
5. L. Nagel, "ADVICE For Circuit Simulation," *Proceedings of the 1980 International Symposium on Circuits and Systems*, Houston, April 1980.
6. M. Shoji, "Electrical Design of the BELLMAC™ - 32A Microprocessor," 1982 International Conference on Circuits and Computers, New York, September 1982.
7. A. D. Lopez and H-F. S. Law, "A Dense Gate Matrix Layout Method for MOS VLSI," *IEEE Transactions on Electronic Devices*, Vol. ED-27, August 1980, pp. 1671-1675.
8. A. E. Dunlop, "Automatic Layout of Gate Arrays," *Proceedings of the 1983 International Symposium on Circuits and Systems*, Newport Beach, Calif., May 1983, pp. 1245-1248.

(Manuscript received December 4, 1985)

MARCH/APRIL 1986 • VOLUME 65 • ISSUE 2