EECS 579 Term Projects

A. Outline

This document describes the EECS 579 term project requirements and deadlines, and lists some possible topics. Proposals for projects on other topics are encouraged. You may also propose modified versions of the suggested projects.

Project Selection

Select a project (possibly modified) from this list below, or propose a new project. Projects may be proposed by individual students or by a team of two students. (The latter is intended for experimental projects only). You will find relevant reference material in the text, the course reserve file in the library, and on the World Wide Web.

.

Proposal

Submit a short proposal (two typed pages maximum, excluding appendices for the main proposal) **.** In the event that your first choice is not accepted, also state a second choice and be prepared to expand it into a full proposal later, if asked. Every effort will be made to give students their first choice, but a reasonable distribution of the topics is required. Projects will be assigned by the instructor based on the quality of the proposals. Your proposal should have the following format:

1. Title, student name(s) and e-mail address(es)
   Project outline: Describe your objectives, method of attack, expected results, and other relevant data.
   Time schedule (List dates of major stages or “milestones” in the execution of the project.)
   References (List only those items actually consulted in preparing the proposal)
   Work division (This applies to two-person projects only; indicate clearly who will do what.)
2. Brief alternative proposal (Give a title and a 1-paragraph description of your back-up project).

Progress Report

A one-page written progress report should summarize how the project is progressing, describe any unexpected difficulties encountered, and describe any significant changes in the project plans.

Oral Presentation

Short oral presentations of the projects to the class will be scheduled
Presentation time will be 15 minutes or so per project.

Final Written Report

The project will be documented in a term report **.** This report will be a major factor in determining project grades. The term reports should be of professional quality and be in the usual format of a technical report or a journal paper. Experimental projects may have shorter reports, with detailed material such as program listings, sample runs, etc. placed in appendices.

All **term reports** should be computer-printed and have the following standard format:

Title page (project title, authors, date, abstract)
Abstract (half-page summary of the report)
Sec. 1. *Introduction* (problem definition, background material, prior work, goals, methods)
Secs. 2 through *k*. (body of the report in as many sections as needed—usually 3 to 5)
Sec. *k* + 1. *Conclusions* (evaluation of results, lessons learned, and suggestions for improvement or further work)
References (use the style of the *IEEE Transactions* journals)
Appendices (if appropriate, such as experimental data, program code, etc.)

B. Project Suggestions

A list of possible term projects follows. *The cited references should be viewed as a starting point only* in the project selection process.

Note that there is a huge Bibliography (covering up to the year 1999 or so) at the end of Bushnell and Agrawal.

Experimental Projects

1. Implementation of the combinational FAN algorithm

Write a program for any available computer system to implement the FAN ATPG algorithm for large combinational circuits. Your program should be able to generate efficiently a complete test set for all SSL faults in a given circuit. Incorporate all (or most of) the speedup heuristics of the FAN algorithm. Also incorporate some simulation features to allow one test to detect many faults. Run your program on test cases, including the ISCAS-85 benchmark circuits, and compare your program's performance to published data on other programs in the literature. **Ref.** Text, Chap. 7.

2. Implementation of the sequential PODEM algorithm

Write a program for any available computer system to implement a version of the PODEM algorithm for small to mid-sized sequential circuits. The primary goals here are accuracy and complete SSL fault coverage, rather than fast computing speed. Run your program on some test circuits, including some of the smaller ISCAS-89 benchmarks, and compare your program's performance to published data on other programs. **Refs.** Text, Chap. 8.

3. Comparison and evaluation of two commercial ATPG programs

We have many commercial electronic CAD tools here at Michigan from major CAD vendors like Mentor Graphics, Synopsys and Cadence. The goal of this project is to identify and evaluate the performance and cost of two ATPG programs by analyzing them running them on some large benchmark circuits, both combinational and sequential. **Refs.** The EECS CAD support home page: http://www.eecs.umich.edu/dco/ecad/ Follow the links for each vendor for tool descriptions, supported platforms, start-up instructions, etc. Variants of this project might be to evaluate analog testing or fault simulation tools.

4. Lab testing of ICs using ATE

This project is aimed primarily at students who have taken a VLSI design class (EECS 427 or 627), and (ideally) have designed and fabricated a chip that needs to be tested. We have a high-performance Hewlett-Packard IC tester in the Solid-State Circuits Lab to use as the testing tool. Chips designed by others may be used if available and adequately documented. *You can only propose this* *project if you have identified an actual set of chips that you can use!* This project will involve deriving tests and applying them to the said chips for functional correctness, and performing such parametric measurements as schmoo plots.

Survey-oriented Projects

5. Test generation using parallel processing techniques

Study the suitability of general-purpose or special-purpose parallel computers for speeding up test generation. What computer architectures and algorithm types are available and suitable for test generation at various abstraction levels? Opportunities exist to do some experimentation (testing ideas or implementing a complete program) using various parallel computer systems to which we may have access via CAEN, including networks of Unix workstations (NOWs). Familiarity with parallel processing techniques is recommended for this project. **Ref.** P. Banerjee: *Parallel Algorithms for Computer-Aided Design*, Prentice-Hall, 1994

6. Analog and mixed-signal testing techniques.

This research project will address some key testing problems in circuits that combine analog and digital components, such as digital signal processors (DSPs). Although digital testing (the focus of EECS 579) is well understood, the same cannot be said for analog testing. Simple, easy-to-use fault models like the SSL model exist for digital testing, but the modeling of analog faults is far more difficult and poorly understood. Built-in self-test (BIST) is now making inroads in commercial digital designs; however, mixed-signal BIST remains in a rudimentary state.This project will investigate the modeling of analog faults and the development of BIST methods for mixed-signal applications. **Ref.** Text. Chap. 10–11.

7. Testing circuits composed of IP (intellectual property) cores

The problems associated with testing large predesigned digital modules, such as microprocessors and I/O interface controllers, is a hot topic in digital design. Cores are typically functional blocks whose internal implementation details are only partially specified. Testing such cores is difficult, especially when they are combined with other cores and user-designed logic. The goal of this project is to survey the issues associated with testing cores, analyze the main problems, and to propose some solutions to the problem of high-coverage testing of embedded cores in system- on-chip (SOC) designs. **Refs**. (1) Text Chap. 18. (2) H. Kim and J.P. Hayes: “Realization-independent ATPG for designs with unimplemented blocks,” *IEEE Trans. Computer-Aided Design*, vol. 20, pp.290–306, Feb. 2001.

8. On-line testing of complex digital ICs

For critical applications such as car or airplane control, on-line fault detection and monitoring is as important as manufacturing testing This project involves developing and implementing a design plan for on-line testing of a new digital IC of the system-on-chip (SOC) type. This plan should include both periodic testing for permanent hardware faults as well as continuous testing for transient faults. As it is not always possible to test all parts of an IC continuously, a rational plan for prioritizing faults should be provided. The overall plan should contain provisions for internal hardware modifications as well as external testing. For example, if the chip contains a microprocessor, it is expected that the microprocessor will play a key role in testing. **Ref**. H. Al-Asaad et al.“Online BIST for embedded systems,” *IEEE Design & Test*, vol. 15, no. 4, pp.17 -24, Oct.-Dec. 1998.

9. Nanoelectronics and the end of Moore’s Law

It is believed that if Moore’s Law continues as it has since the 1960’s, transistors will reach the size of an atom around 2020. At that point CMOS circuits as we know them will no longer function and ICs will be built in entirely new ways. Various new “nano” technologies, such as spintronics, (bio)molecular circuits, quantum circuits, carbon nanotubes, etc. are being suggested as possible replacements for CMOS. This project will explore in depth these proposed nanoelectronic technologies, with emphasis on the testing and reliability problems that are likely to accompany them. **Refs**: Search the web in general and IEEE Xplore in particular.

Research-oriented Projects

10. Synthesis of efficient deterministic test generators for BIST

Most BIST techniques for logic circuits use low-cost (pseudo) random test-pattern generators. This is not so easy to do in the case of non-random deterministic test generation, where a specific set of tests is to be produced, e.g., tests previously generated by PODEM. Many large, practical circuits can be fully tested (100% SSL fault coverage) with small but carefully selected test sets. For BIST, these tests must be generated on-chip. Storing them in a ROM is one possibility, but it usually takes a lot of hardware (chip area), which can be unacceptable. This project will investigate the design of small sequential circuits (special-purpose FSMs) that generate a specific, deterministic set of test patterns, while not requiring a lot of logic. **Ref:** Text, Chap. 15.

11. Byzantine faults and their detection

These are defined as faults that present different symptoms to different observers. A basic example is a bridging fault that creates an intermediate voltage level which may be read as 0 by some gates and as 1 by others. Such faults are very difficult to model or detect. In the past they have been considered rare and largely ignored. According to Driscoll in Ref. 2, however, such faults are becoming much more common as IC devices get smaller. Indeed he predicts that PCs will have to be designed to tolerate such faults a decade from now, otherwise they won’t operate! This project will explore the nature of Byzantine faults and the methods needed to detect and tolerate them. **Refs.** (1) D.B Lavo et al.;”Beyond the Byzantine generals: unexpected behavior and bridging fault diagnosis,” *Proc. Intl. Test Conf.*, pp.611 -619, 1996. (2) K. Driscoll et al.: “Byzantine fault tolerance, from reality to theory,” Proc. SAFECOMP 2003, pp. 235-248 (See Prof. Hayes for a copy of this second paper.).

The following projects are directly or indirectly motivated by quantum computing.

12. Test generation for reversible combinational circuits

Reversible logic circuits, where the output uniquely determines the input, allow computation with arbitrarily small power dissipation; they are also essential components for quantum computing. Reversibility can significantly simplify testing. This project will study testing problems for reversible, combinational non-quantum circuits, in particular, generating (near) minimal test sets for more powerful fault models such as the single and multiple cell fault models. The problem of designing reversible circuits to improve their testability is also an interesting (and fairly tractable) one. **Ref**. K.N. Patel, J.P. Hayes and I. Markov, “Fault testing for reversible circuits,” *Proc. VLSI Test Symp*., pp.410-416, April 2003. (This is available at J.P. Hayes’s web site. It is the first and perhaps the only paper so far to address the reversible ATPG problem, mainly using the stuck-at fault model.)

13. Test generation for reversible logic arrays

(Read the preceding item first) This project will study testing problems and testability properties of iterative array (ILA) circuits, which are composed of identical but reversible logic cells that are connected in uniform fashion. A well known example of an irreversible ILA is a ripple-carry adder. Some ILAs are known to require a constant number of tests independent of the array length; this is called C-testability. ILA models are also useful when testing general sequential circuits, since they correspond to a time-frame expansion of the circuit. This project will investigate reversible iterative logic arrays (ILAs) which model both combinational and sequential circuits. It will attempt to extend known results about irreversible ILAs to reversible ILAs using the stuck-at fault model (and perhaps the single cell fault model). It will also look at ways to modify such ILAs to increase their testability. **Refs**. (1) K.N. Patel, J.P. Hayes and I. Markov, "Fault testing for reversible circuits," *Proc. VLSI Test Symp*., pp.410-416, April 2003. (2) Sec. 8.3.3 of Abramovici, Breuer and Friedman, *Digital systems testing and testable design,* 1990 (which is in the course reserve file).

14. Error models for quantum circuits

Quantum circuits represent a revolutionary new way to compute and communicate that has great long-term potential. Some important problems, notably prime factorization, can be solved exponentially faster using quantum techniques. Quantum devices are extremely small and are subject to noise (called decoherence) that cause a quantum state to decay rapidly. Thus they are more error-prone than classical (non-quantum) circuits. This project will expire the normal and faulty behavior of quantum circuits for various proposed quantum technologies including ion traps and NMR. It will propose some fault/error models and determine how they might be detected. This project requires a solid math background, especially in linear algebra, in order to read the relevant literature. No specific expertise in quantum mechanics is needed (but certainly won't hurt if you have it).

Refs. (1) J.P. Hayes: "Tutorial: basic concepts in quantum circuits," Slides of a talk presented at the 40th Design Automation Conf, June 2003 (This is available at J.P. Hayes's web site). (2) M.A. Nielsen and I.L. Chuang: *Quantum computation and quantum information*, Cambridge Univ. Press, 2000 (This text book is in the EECS 579 library reserve file.)

C. Sources of Information

Books

(1) Textbook (Bushnell and Agrawal).

(2) Course reserve file

1. Bushnell, Michael L. and Vishwani D. Agrawal [Required textbook]
   Essentials of electronic testing for digital, memory, and mixed-signal -- Kluwer, 2000.

2. Abramovici, Miron, Melvin A. Breuer and Arthur D. Friedman
   Digital systems testing and testable design -- New York: IEEE Press, 1990.

3. Bardell, Paul H, William H. McAnney and Jacob Savir
   Built-in test for VLSI: pseudorandom techniques -- New York: Wiley, 1987.

4. Crouch, Alfred L
   Design-for-test for digital IC's and embedded core systems, Prentice Hall, 1999.

5. Jha, Niraj K. and S. Gupta.
   Testing of digital systems -- Cambridge University Press, 2003.

6. Nadeau-Dostie, Benoit (ed.)
Design for at-speed test, diagnosis and measurement -- Kluwer, Boston, 2000.

7. Nielsen, Michael A. and Isaac L. Chuang
Quantum computation and quantum information, New York: Cambridge Univ. Press, 2000.

8. Pradhan, Dhiraj K.
Fault-tolerant computer system design, Prentice Hall, 1996.

Journals

IEEE Design and Test magazine
IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems
Journal of Electronic Testing (JETTA)
IIEEE Transactions on VLSI Systems

Conference Proceedings

International Test Conference (ITC) Proceedings (Oct.)
VLSI Test Symposium (VTS) Proceedings (May)
Design Automation Conference (DAC) Proceedings (July)

All the above are annual IEEE-sponsored research-oriented conferences, and are accessible on-line.

The World Wide Web

This is a vast, unstructured source of information on all topics. Recent papers (or relevant Ph.D. theses and technical reports) are easily found and copied from the Web. Search by topic, author name, university or company name. Using the “IEEE Xplor” web site at *http://ieeexplore.ieee.org/Xplore/DynWel.jsp* you can directly access electronic versions of all conference and journal papers published in the last 15 years or so by the IEEE.