ASIC 설계를 위한 원격 테스트 시스템

윤 도현, 강 성호
연세대학교 전기공학과
전화: 361-2774 / 팩스: 313-8053

Tele-test Systems for ASIC Design

Doehyun Yoon, Sungho Kang
Dept. of Electrical Eng. Yonsei Univ.
Email: dhyoon@dopey.yonsei.ac.kr, shkang@yonsei.ac.kr

Abstract

In this paper, Tele-Test System for ASIC Design is constructed. It consists of the server, and the clients. The server and clients are implemented by Java. Using Java RMI system, the remote access via information network is implemented.

In this Tele-test system, fault simulation, test pattern compaction, test pattern generation, and path-delay fault test generation services are implemented. All service can be performed parallel by network access.

I. 서론

반도체의 테스트는 주어진 제품이 제대로 동작되는지 알아보는 작업이다. 제작자의 입장에서는 제품을 출고하기 전에 불량품을 가려냄으로써 제품의 신뢰도를 높여 줄 필요가 있으며 사용자의 입장에서도 구입한 제품 또는 사용중인 제품이 정상적인 동작을 하는지 확인할 필요성이 있다. 앞서 살펴 본 바와 같이 칩이 보다 고집적화 되고 복잡화함에 따라 테스트에 소요되는 비용이 급격히 증가하여 테스트의 중요성이 점차 부각되고 있다. 복잡해지고 많은 시간을 필요로 하는 테스트를 보다 쉽게 하기 위해 칩의 설계에서부터 테스트를 고려하는 테스트 용이화 방식이 현재 많이 사용되고 있다. 하지만 설계자들은 이러한 기술에 익숙하지 못하기 때문에 초고속통신망을 이용하여 설계된 회로에 가장 최적의 테스트 용이화 회로를 합성시키는 것은 칩의 설계 시간 단축뿐 아니라 비용 절감에 많은 이득을 얻을 수 있다.

최근 개인용 컴퓨터의 멀티미디어 컴퓨터로의 발전과 인터넷의 확산이 급속하게 이루어지고 있다. 이와 더불어 통신망도 음성이나 동영상을 주고받을 수 있을 만큼 많은 용량을 처리할 수 있게 발전되었다. 통신망의 발전은 언제 어디서나 원하는 정보나 자료를 구할 수 있게 하였다. 또한 화상회의나 원격 가상교육과 같은 먼 곳에 있는 사람과의 정보교환도 가능하게 하였다. 따라서 본 논문에서 목표로 하는 것은 이러한 발전된 초고속 통신망을 이용하여 원격 테스트 시스템을 구현하는 것으로 주어진 회로에 대한 테스트 패턴 생성, 고장 시뮬레이션, 패턴 압축, 지연 고장 테스트 패턴 생성 등 초고속 정보 통신망 상에서 원격으로 동작되는 테스트 시스템을 구현하는 것을 목표로 한다.

본 논문은 1998년도 정보통신부의 초고속정보통신 응용기술개발사업의 결과입니다.

II. 원격 테스트 시스템의 구조

원격 테스트 시스템은 사용자의 요구에 따라 주어진 회로에 대한 테스트 패턴을 생성 할 수 있다. 본 장에서는 원격 시스템의 설계 시 고려되었던 주요 사항들을 기술하고, 설계된 각 모듈의 기능과 구현 사항에 대해 기술한다.

원격 테스트 시스템은 서버와 클라이언트로 구성되는데, 서버와 데이터 베이스 부분은 Java를 이용하여 구현되었다. 그림 1에 전체 시스템의 구조를 나타내었다. 서버와 클라이언트간의 통신은 TCP/IP 소켓을 사용하는데, Java의 RMI[1]를 이용하여 클라이언트에서 서버 상의 원격 객체를 제어하는 방식을 사용하였다. 서버는 ServerService와 Server의 두 Java 클래스로 구성되어 있는데, ServerService 클래스는 서버에서 클라이언트 측에 제공하는 서비스를 정의하는 클래스이고, Server 클래스는 ServerService에서 정의된 서비스를 실제로 구현하고, 클라이언트와 데이터 베이스와의 접속을 관리하는 역할을 한다. 몇몇 서비스는 Java로 구현되지 않고, 서버에서 C프로

그램으로 제작되어 컴파일된 형태를 가지므로 이러한 경우 Server는 클라이언트에서 요청한 입력을 이용하여 서버상의 프로그램을 구동시킨 후에 결과만 클라이언트에 전달하는 역할을 한다. 그림 2에 Server의 역할을 나타내었다.

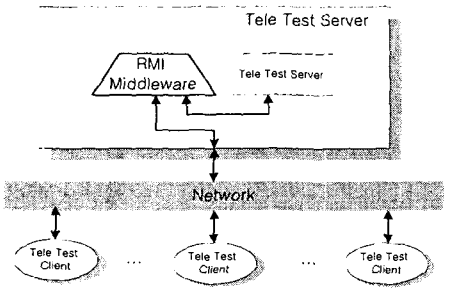

그림 1. 원격 테스트 시스템의 구조

Fig. 1. Architecture of Tele-test System

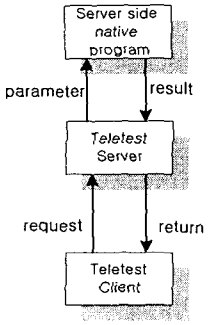

그림 2. 원격 테스트 서버의 동작

Fig. 2. Behavior of Server

앞서 설명하였듯이 Server에서 제공하는 서비스는 모두 RMI를 이용하여 구현되었으므로, 클라이언트에서 서버를 이용할 때는 이 RMI를 이용하여 원격 클래스의 메쏘드를 실행시키는 방법으로 서버에서 제공하는 서비스를 이용할 수 있다.

원격 테스트 클라이언트는 서버와는 다르게 순수하게 Java를 이용하여 구현되었다. 먼저 Client 클래스는 최초에 클라이언트를 실행시킬 때 이용되는 클래스로 네트워크 환경을 초기화시키고, 서버와의 접속을 시도한 후에 ClientFrame 클래스를 실행시키는 역할을 한다. ClientFrame 클래스는 Java의 JFrame을 이용하여 구현된 윈도우 프레임으로 이 GUI 환경을 이용하여 사용자는 원하는 입력을 설정하고, 원하는 서비스를 선택하여 실행 시킬 수 있다.

III. 원격 테스트 시스템에서 제공하는 서비스

원격 테스트 서비스는 일반적인 반도체 설계용 CAD에서 제공되는 서비스를 원격 환경에서 이용할 수 있도록 해준다. 이를 위한 테스트 프로그램들은 모두 Sun SPARC workstation 상에서 C 프로그램으로 구현되었고, 원격 테스트 서버에서는 클라이언트에서 입력한 파라미터들을 가지고 서버상의 프로그램을 구동시킨 후에 이 결과를 클라이언트에 돌려주게 된다. 이러한 모든 작업은 RMI를 이용하여 구현되었고, 또한 실행 될 때 개별적인 쓰레드를 생성하므로, 클라이언트에서 원격 객체를 이용하여 실행시킬 수 있고, 또한 여러 작업을 동시에 진행시킬 수 있다.

이 원격 테스트 서비스에서 제공할 수 있는 서비스로는 다음과 같은 작업들이 있다.

· 고장 시뮬레이션: 회로에 사용자가 지정한 테스트 패턴을 인가하여, 인가한 테스트 패턴이 회로내의 고착 고장을 어떤 것을 검출할 수 있는지를 시뮬레이션 한다. 고장 시뮬레이션이 끝난 후에는 고착 고장에 대한 검출율 이외에 인가한 테스트 패턴으로 검출되지 못한 고장들의 목록을 결과로 표시한다. 이 서비스를 구현하기 위하여 병렬 패턴 단일 고장 전파(PPSFP: Parallel Pattern Single Fault Propagation)의 방법[2]을 이용하여 회로의 크기가 크더라도 빠르게 고장 시뮬레이션을 수행할 수 있었다.

· 테스트 패턴의 생성: 회로 내의 고착 고장에 대한 테스트 패턴을 생성해 주는 서비스로 의사 무작위에 의한 방법, 결정론적 테스트 패턴 생성 방법, 그리고 2 단계 테스트 패턴 생성 방법의 세 가지 방법을 이용할 수 있다. 의사 무작위 테스트 패턴을 이용한 방법은 테스트 패턴 생성에 많은 시간이 걸리지 않는 장점이 있는 반면에 의사 무작위 저항 고장과 같은 검출되기 어려운 고장을 검출하기 위해서는 너무나 많은 시간이 소모된다는 단점이 있다. 반면에 결정론적 테스트 패턴 생성 방법은 단일 고착 고장을 가정하고, 고장 활성화(Fault Excitation)와 고장 전파(Fault Propagation)를 이용하여 가정한 고장을 검출할 수 있는 테스트 패턴을 생성한다. 그러나 이러한 방법은 시간이 아주 오래 걸리고, 특히 테스트 패턴 생성 도중에 시도 철회 횟수가 많을수록 시간이 오래 걸리므로 실제에서는 각 테스트 패턴 생성 시마다 최대 시도 철회 횟수를 설정하게 된다. 2 단계 테스트 패턴 생성 방법은 이 두 가지 방법의 장점을 결합하여 처음에 의사 무작위 테스트 패턴을 생성하여 검출되기 쉬운 고장을 검출한 후에, 나머지 고장에 대하여서만 결정론적 테스트 패턴 생성을 하여

효율적으로 테스트 패턴 생성을 수행한다. 원격 테스트 시스템에서는 이 3 가지 방법을 모두 이용하여 테스트 패턴을 생성할 수 있는데, 2 단계 테스트 패턴 생성 시에는 처음에 한꺼번에 32개의 의사 무작위 테스트 패턴을 생성하면서 각 32개의 테스트 패턴이 고장을 하나도 검출하지 못하는 경우가 4번 이상 연속적으로 발생하게 되면, 나머지 고장들에 대해서는 결정론적 테스트 패턴을 생성하였다. 결정론적 테스트 패턴 생성 시에는 최대 시도 철회 횟수를 100으로 하는 PODEM 알고리듬[3]을 이용하였다.

· 테스트 패턴의 압축: 실제 생산 시에는 같은 테스트 패턴의 길이가 짧을수록 테스트에 걸리는 시간이 줄어들게 된다. 따라서 고장 검출율을 그대로 유지한 상태로 테스트 패턴의 수를 줄이는 테스트 패턴 압축 기법을 적용해서 주어든 테스트 패턴을 이용해야 한다. 원격 테스트에서 적용한 테스트 패턴 압축 기법은 SOCRATES[4]에서 제시된 역순서 고장 시뮬레이션(Reverse Order Fault Simulation) 기법을 이용하여 빠르고 효율적으로 테스트 패턴의 수를 줄일 수 있었다.

· 지연 고장에 대한 테스트 패턴 생성: 반도체의 속도가 점차로 빠르게 되면서 기존의 고착 고장에 의한 테스트만으로는 반도체의 이상 유무를 확인 할 수 없게 되었다. 따라서 지연 고장 모델이 제시되었고, 이 지연 고장은 회로내의 신호의 진행 지연이 모델링된 지연 이상으로 생긴 결함으로 정의 할 수 있다. 이 지연 고장에 대한 테스트를 위해서는 두 개의 테스트 패턴이 필요한데, 첫 번째 테스트 패턴은 회로 내의 소자들을 초기화시키고, 두 번째 테스트 패턴은 회로는 회로 내의 소자에 상승 천이 내지는 하강 천이를 만들게 되고, 정해진 시간이 지난 후에 이 두 번째 테스트 패턴에 의한 출력의 변화를 회로의 출력에서 관찰할 수 없다면 이 회로는 지연 고장이 발생했다고 판단할 수 있게 된다. 원격 테스트에서는 지연 고장을 위하여 회로에 대한 경로를 생성하는 경로 생성 서비스와 이 서비스에서 생성된 경로에 대한 경로 지연 고장 테스트 패턴을 생성하는 두 서비스가 있다. 이 두 서비스를 이용해서 주어진 회로에 대한 경로 지연 고장을 테스트 할 수 있는 테스트 패턴을 원격으로 생성할 수 있다.

이상의 서비스는 모두 서버 상의 프로그램으로 구현되었으므로 아주 빠르게 실행 되게 된다. 원격 테스트 서버와 클라이언트에서는 이 프로그램들을 구동시키고, 결과를 받아 보게 되는데, 이러한 시스템을 Java의 RMI를 이용하여 구현함으로써 최적화된 분산 환경 시스템을 구현할 수 있다. 즉, 고가의 CAD 툴과 고성능의 서버 환경에서만 구동 될 수 있는 프로그램을 실제 수행은 서버 상에서 하고, 클라이언트에서는 원격 객체를 이용하여 원격 구동을 시킴으로써 저가의 PC 환경에서 구동할 수 있게 되었고, 그러면서도 실제 프로그램의 수행은 서버 상에서 이루어지므로 성능의 저하는 전혀 나타나지 않게 되었다.

VI. 시험 및 구현 환경

원격 테스트 시스템은 서버만이 원격 테스트 서비스를 구현하기 위해서 특정 아키텍처에 적용되는 프로그램을 사용하였고, 나머지 부분은 모두 Java로 구성되었기 때문에 서버를 제외한 클라이언트는 어떠한 환경에서도 구현 될 수 있다. 본 시스템의 테스트를 위해서 서버에는 256MB의 주 메모리를 갖는 SUN SPARC workstation을 이용하였고 원격 테스트 서버를 위해서 JDK1.2beta 버전의 Java 가상 머신을 서버 상에 설치하였다. 원격 테스트 클라이언트는 Java 가상 머신이 설치된 모든 환경에서 실행이 가능한데, 실제 실험에는 Pentium II PC에 JDK1.2 버전의 Java 가상 머신이 설치된 환경에서 클라이언트를 실행하였다. 그리고 전체 원격 테스트 시스템 실험은 연세대학교 초고속 선도 시험망을 이용하였다. 그림 3에 원격 테스트 시스템의 구현 및 시험 환경을 나타내었다.

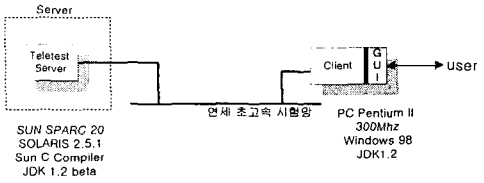

그림 3. 구현 및 시험 환경

Fig. 3. Tele-test Experimental Environment

이러한 환경에서 원격 테스트 시스템을 구동시키면 클라이언트 측에서 처음에 그림 4와 같은 화면을 볼 수 있다. 그림 4의 여러 가지 서비스 중에서 예를 들어 ISCAS 85 성능 측정 기준 회로 중 c432회로에 대한 테스트 패턴을 네트워크 상에서 실행하게 되면 클라이언트에서 입력한 회로를 서버 측에서 받아서 테스트 패턴을 생성하게 된다. 테스트 패턴 생성이 끝난 후에 클라이언트 측에서는 그림 5의 결과 요약과 그림 6의 테스트 패턴을 얻을 수 있다. 이러한 모든 작업은 개별적인 쓰래드로 구성되어서 병렬로 다른 작업과 병렬로 이루어질 수 있고, 따라서 한꺼번에 여러 개의 작업을 동시에 진행시킬 수도 있다.

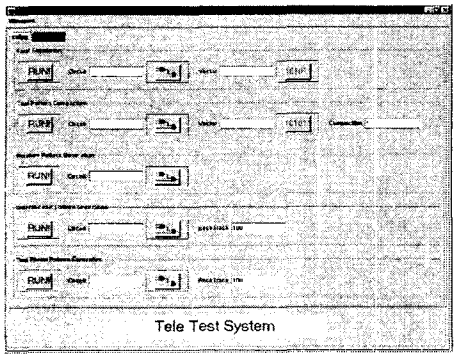

그림 4. 원격 테스트 시작 화면
Fig. 4. First Scene of Tele-test System

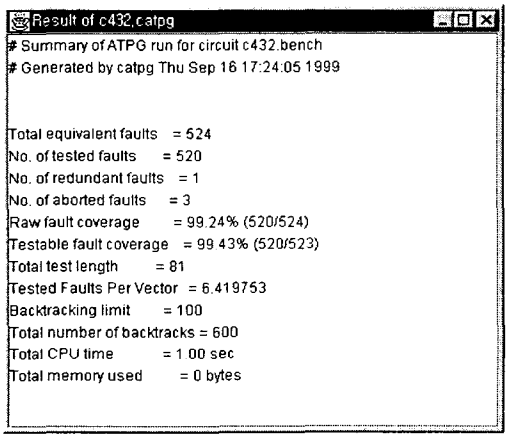

그림 5. c432에 대한 테스트 패턴 생성 결과
Fig. 5. Test generation result for c432

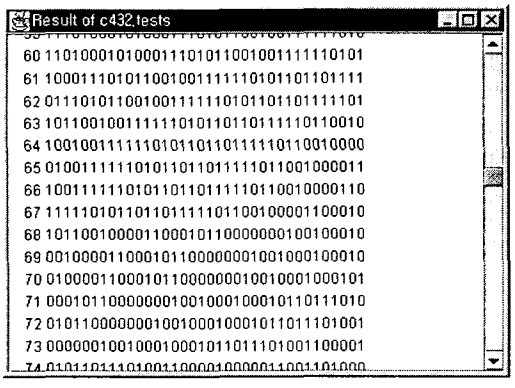

그림 6. c432에 대한 테스트 패턴
Fig. 6. Generated Test Patterns for c432

V. 결론 및 향후 연구방향

본 논문에서는 주문형 반도체의 설계를 위한 원격 테스트 시스템을 구현하였다. 이 시스템에서 주어진 회로에 대한 테스트 패턴 생성, 고장 시뮬레이션, 패턴 압축, 경로 생성, 그리고 지연 고장에 대한 테스트 패턴 생성을 RMI 시스템을 이용하여 원격으로 제어할 수 있도록 구현하였다. 이러한 시스템을 이용하여 고성능의 서버와 저성능의 클라이언트간의 통신을 이용하여 고가의 서버급 컴퓨터와 CAD 툴이 필요한 작업을 네트워크를 이용하여 효율적으로 작업할 수 있는 저비용의 분산 환경을 구축하였다. 이 원격 테스트 시스템은 RMI와 Java를 이용하여 구현되었으므로 클라이언트는 플랫폼에 상관없이 Java 가상 머신이 구현된 어떠한 컴퓨터에서라도 수행이 가능하다. 이러한 연구 결과를 바탕으로 앞으로는 서버와 클라이언트간의 더 많은 상호 작용을 바탕으로 하는 원격 테스트 시스템의 연구가 필요하고, 더 나아가서는 주문형 반도체의 전체 설계, 시뮬레이션, 검증, 테스트 작업을 통합한 효율적인 분산형 반도체 설계 환경의 구축이 필요하다. 이러한 통합된 네트워크 환경에서의 설계, 테스트 환경이 구축된다면 고가의 CAD 툴의 사용을 최소화하고, 서버 상에 설치된 CAD 툴만으로 전체 네트워크에 연결된 많은 사용자가 손쉽게 설계 환경에 접근이 가능할 것이다. 또한 반도체의 집적도가 증가하고 회로의 크기가 커짐에 따라서 여러 설계 팀이 동시에 나누어서 작업을 하게 되는데, 이러한 설계 팀들 사이의 자료 교환, 공동 작업 및 에러 수정 작업에도 많은 도움을 주리라고 예상한다.

참고 문헌

[1] http://java.sun.com/marketing/collateral/javarmi.html

[2] J. A. Waicukauski, et. al., "Fault Simulation for Structured VLSI," *VLSI Systems Design*, pp. 20-32, Dec. 1985.

[3] P. Goel, "An Implicit Enumeration Algorithm to Generate Tests for Combinational Logic Circuits," *IEEE Trans. on Computer*, vol. C-30, no. 3, pp. 215-222, 1981.

[4] M. H. Schulz, E. Trischler, and T. M. Sarfert, "SOCRATES: A Highly Efficient Automatic Test Pattern Generation System," *IEEE Trans. on Computer-Aided Design*, vol. 7, no. 1, pp.126-137, Jan. 1988.