

## 내장형 다중 RAM 모듈용 원형 BIST 생성기\*

장 종 권

컴퓨터정보통신공학부

### <요 약>

ASIC에 내장된 다중 RAM 모듈의 테스트를 위하여 BIST(Built-In Self Test)기법을 이용한 내장형 다중 RAM 모듈용 범용 BIST 생성기를 설계하였다. 본 논문에서 제안한 범용 BIST 생성기는 주어진 내장 RAM 모듈의 사양과 적용되는 테스트 알고리즘에 따라 이에 부합되는 BIST 회로를 VHDL 코드로 자동 생성하는 설계 자동화 도구로서, 각 모듈 단위로 설계되어 회로의 추가 개발 및 재사용이 가능하다. 뿐만 아니라, 직렬 인터페이스 기법을 사용하여 부가적인 핀 수를 줄였으며, BIST 회로 공유 기법의 도입으로 다중 RAM 테스트 시 다양한 사양의 RAM 테스트에 적용이 쉽도록 설계하였다.

---

## A Generic BIST Builder of Embedded Multiple RAM Modules

Jong-Kwon Chang

Division of Computer Engineering & Information Technology

### <Abstract>

In this paper we propose a generic BIST builder of the embedded multiple RAM modules in ASICs. The BIST circuitry is automatically generated according to the specification of the target RAM Modules and the test algorithms applying to them. The BIST is designed using the top-down technique and, thus, has several advantages in the area of the selection of test algorithm, the development of the circuitry, and the reuse of the circuitry. In addition, we have modified the existing serial interfacing approach to obtain smaller

---

\* 본 연구는 1997년도 울산대학교 연구비 지원에 의해 수행되었음.

additional BIST circuitry and higher fault coverage and better BIST sharing of the target RAM Modules in ASICs.

## I. 서 론

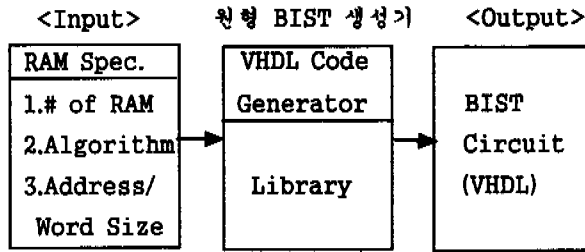
오늘날 수많은 통신용 ASIC 칩은 하나의 칩 속에 다수의 다양한 메모리가 내장되어 있다. ASIC 내부의 랜덤 로직 고장 검출도가 스캔 디자인 기술에 의해 향상되었음에도 불구하고 내장 메모리의 테스트는 외부 테스트패턴 인가 방식의 시간 한계성과 RAM 입출력 핀의 공간 제약성으로 인하여 여전히 어려운 문제로 남아 있다.<sup>[1,2]</sup> 또한, 칩이 고집적화 되어 가는 추세 속에서 하나의 칩 속에 내장된 여러 개의 RAM 모듈을 보다 정확하고 효율적으로 테스트하는 문제 또한 중요하다 할 수 있다.<sup>[3,4,5]</sup> ASIC 칩의 랜덤 로직 회로망의 테스트는 구조적 DFT(Design for Testability) 기법인 스캔 기법을 채택하여, 칩 외부의 PI(primary input), PO(primary output), SI(scan input), SO(scan output)에 테스트 패턴을 인가하여 테스트를 수행할 수 있다.<sup>[1,2,3]</sup> 즉, 랜덤 로직의 고장 검출도를 향상시킬 수 있다. 하지만, ASIC 칩에 내장된 메모리 소자의 테스트는 다음과 같은 문제점으로 인하여 외부 테스트를 수행할 수 없다. 첫째, 메모리의 구조적 특성상 고장 모델이 일반적으로 사용되는 고착고장 모델로는 검출할 수 없는 복잡한 고장 형태가 발생한다. 따라서, 메모리의 고장을 검출하기 위해서는 많은 양의 테스트 패턴을 인가해야만 하는데, 외부에서 테스트 패턴을 인가하는 방식은 패턴인가 시간이 많이 소요될 뿐만 아니라 다양한 메모리 소자의 고장을 검출하는데 한계성이 있다. 둘째, ASIC에 내장된 메모리 소자의 어드레스, 데이터, 읽기/쓰기 제어 신호를 ASIC의 외부 입출력 핀에서 직접적으로 제어 및 관측할 수가 없다. 즉, 외부에서 ASIC에 내장된 메모리의 고장을 검출하기가 없다.

BIST 기법은 칩 내부에 부가적인 다이 면적을 차지하지만, 다음과 같은 장점이 있다.<sup>[6,7,8,9,10,11]</sup> 첫째, VLSI 칩 테스트의 복잡도를 덜어 준다. 둘째, VLSI 설계 공정(CMOS, BiCMOS, Bipolar)에 알맞은 고장 모델이 설정된 후 BIST 회로가 제작되므로 테스트 패턴의 질이 향상된다. 셋째, 부가적인 외부 테스트 장비 없이 칩, 보드, 시스템 수준의 테스트를 고속화 할 수 있다. 이에 본 논문에서는 ASIC에 내장된 다중 RAM 모듈의 테스트를 위하여 BIST 기법을 이용한 내장형 다중 RAM 모듈용 BIST 생성기(이하 원형 BIST 생성기)를 설계하였다.

## II. 원형 BIST 생성기의 구성

원형 BIST 생성기는 내장형 RAM 모듈의 사양과 적용되는 테스트 알고리즘에 따라 이에 부합하는 BIST 회로를 VHDL 코드로 자동 생성하는 설계 자동화 도구이며 VHDL 코드 생성기와 테스트 알고리즘 라이브러리의 두 부분으로 구성된다. ASIC에 내장된 RAM의 개수가 유동적일 때 VHDL로 기술될 BIST 회로의 포트 수 또한 가변적이 된다. 하지만, VHDL로는 위에서 언급한 바와 같이 고정되지 않은 포트 수를 기술하기 어렵다. 이 문제를 해결하기 위해 C언어를 이용하여 VHDL 코드를 생성하는 프로그램인 VHDL 코드 생성기를 사용하였다. 이러한 가변적인 수의 포트를 자동적으로 생성하는 것은 원형 BIST 생성기 제작 시 발생한 구현상

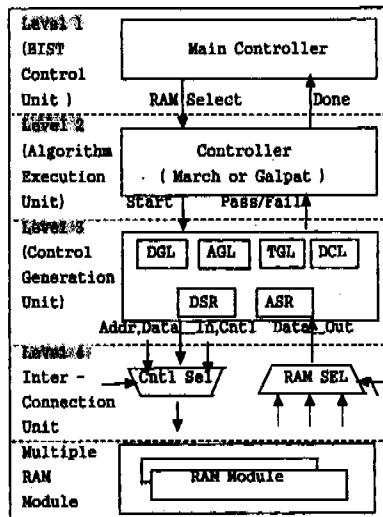
의 문제점으로 VHDL 코드 생성기의 개발로 그 해결이 가능하였다. 그리고, 테스트 알고리즘 라이브러리 부분은 모든 RAM 모듈에 공통으로 적용되는 테스트 알고리즘을 라이브러리화하여 BIST 내부의 제어 유닛에서 선택적으로 사용할 수 있게 하였다. 원형 BIST 생성기(그림1)는 입력으로 RAM의 개수, 사용될 테스트 알고리즘, 그리고 각각의 RAM에 대한 어드레스 와 워드 크기를 받아들여 이에 맞는 다중 RAM 모듈을 위한 BIST 회로를 생성한다.



(그림 1) 원형 BIST 생성기의 입·출력 구조  
 (Fig. 1) The I/O Structure of BIST Builder

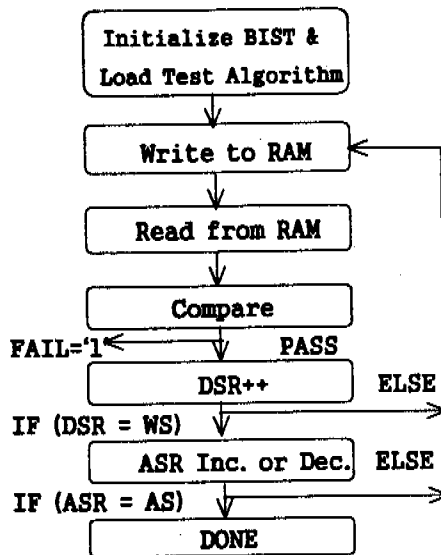
### III. 원형 BIST 생성기의 계층구조

원형 BIST 생성기의 BIST회로(그림 2)와 같이 계층적인 구조로 생성되며 테스트 알고리즘의 변경에 따른 회로의 변경, 회로 개발 및 회로의 재사용이 용이성을 고려하여 설계된다.



(그림 2) BIST 계층 구조  
 (Fig. 2) Hierarchy of BIST Design

또한, BIST 회로의 부가적인 부분을 최소화하고 보다 높은 고장 검출도 및 내장 RAM 모듈에 대한 BIST 회로 공유를 개선하기 위하여 기존의 직렬 인터페이스<sup>[7]</sup> 기법을 적용하였다. 본 논문에서 제안된 BIST 계층 구조(그림 2)는 네 부분으로 구성되었다. Level 1은 BIST Control Unit으로 가장 탑 레벨(top level)에 위치하여 BIST 회로의 메인 컨트롤을 담당하는 부분으로 외부 테스트 신호가 설정되면 테스트 하고자 하는 RAM 모듈들을 선택한 뒤, 하위 Algorithm Execution Unit에서 선택된 테스트 알고리즘을 이용하여 각각의 RAM 모듈에 테스트를 수행하도록 제어 신호를 보내고, 이것으로부터 알고리즘 수행 결과에 따라 Pass/Fail 신호와 테스트 종결 신호를 외부로 출력하는 역할을 수행한다. 이 회로는 FSM(Finite State Machine)으로 구현되며 각각의 RAM에 대하여 하나의 상태가 인가되어 테스트 알고리즘을 수행한다. Level 2인 Algorithm Execution Unit의 흐름도(그림 3)는 Level 1로부터 스타트 신호와 어드레스 크기에 따라 테스트 알고리즘을 수행하게 된다.



(그림 3) Algorithm Execution Unit의 흐름도  
(Fig. 3) The flow of the Algorithm Execution Unit

또한 이때 선택된 RAM 모듈과의 테스트 데이터 직렬 인터페이스가 이루어지게 된다. Level 3인 Control Generation Unit은 AGL(Address Generation Logic), DGL(Data Generation Logic), TGL(Timing Generation Logic), DCL(Data Comparison Logic)과 상태 레지스터인 ASR(Address Status Register), DSR(Data Status Register)로 구성되며 Level 2의 제어 신호에 따라 어드레스, 데이터, 제어 신호등이 RAM으로 적용되고 테스트 결과는 상위 레벨인 Algorithm Execution Unit으로 전달된다. Level 4인 Interconnection Unit에서는 외부 편인 테스트가 설정되면, 회로는 테스트 모드로 전환되고, RAM에 인가되는 시스템의 모든 신호는 멀티플렉스에 의해 차단되어 BIST 회로에서 생성된 신호가 RAM에 인가된다. Level 1과 4는 BIST Builder에서 설명한 VHDL 코드 생성기에 의해, Level 2와 3은 알고리즘 라이브러리에 의해 제작된다.

#### IV. 원형 BIST 생성기의 기능 및 구현

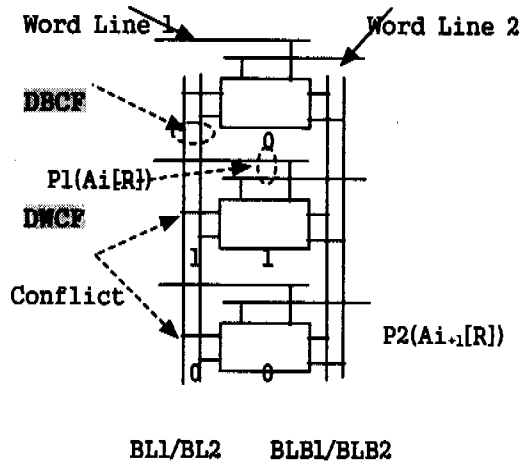
알고리즘 라이브러리에 의해 제작되는 BIST 생성기의 기능은 다음과 같다. Level 3에서 생성되는 각 부분 중 AGL은 RAM에 인가되는 어드레스를 테스트 알고리즘에 따라서 생성하는 유닛으로 Enable 신호가 인가되면 클럭에 동기 하여 업/다운 카운트를 수행하여 이 결과를 RAM의 어드레스로 사용한다. 또한 DGL은 테스트 알고리즘의 각 수행단계에서 필요한 데이터를 생성하며 이 데이터는 직렬 인터페이싱 기법을 이용하여 각 RAM 모듈에 인가되는데 기본 동작은 쉬프트 오퍼레이션이다. 여기서 생성된 데이터(0 또는 1)는 SI을 통하여 RAM으로 인가되고 직렬 연결을 거쳐 SO로 나오게 된다. 쉬프트 오퍼레이션 수행 부분은 멀티플렉스로 설계되었으며 BIST-ON 신호(= test signal)에 따라 테스트 데이터 또는 정상 입력 데이터가 입력된다. 맨 처음 입력되는 데이터는 Serial - In으로 나타나는 RAM 입력 데이터이고, 맨 마지막으로 나오는 데이터는 Serial-Out에 해당되는 데이터이다. DCL(Data Comparison Logic)에서는 RAM으로부터 출력된 이 Serial-Out 데이터 값과 Control 즉, 알고리즘 실행 유닛에서 생성된 값 즉, 고장이 없을 경우 기대되는 값을 비교하여 발생한 고장을 찾게 되는 것이다.

TGL(Timing Generation Logic)은 동작 특성에 맞는 제어 신호를 출력해야 한다. 이를 위해서 인가되는 읽기/쓰기 및 RAM Enable 신호뿐만 아니라 BIST 회로의 동기를 위한 제어 신호도 생성하는 모듈이다. RAM 제어 신호에서 가장 중요한 필요 조건은 읽기/쓰기 동작 시 신호 각각의 지연 조건을 만족해야한다는 것이다. 일반적으로 이러한 타이밍 조건을 만족시킬 수 있는 논리 회로의 설계는 CAD 설계 환경에서 지연 소자를 사용하여 구현이 가능하지만, ASIC 제조 공정을 이용하거나 프로그램 가능한 VLSI 소자를 이용하더라도 RAM 제어 신호의 정확한 지연 조건을 만족시키기에는 어려움이 따르고, 실제 클럭의 동작 주파수에 따라 제어 신호 발생에 관련된 지연 소자 모두를 변경해야만 한다. 따라서 본 논문에서는 RAM 제어 신호의 지연 조건에 독립적인 타이밍 생성기를 설계하여 BIST 회로에 도입하였다. 지연 조건에 독립적인 타이밍 생성기의 장점은 클럭의 동작 주파수가 RAM 입/출력 신호의 최대 지연 값보다 큰 조건만 만족하면 ASIC 제조 시에 정확한 지연 소자 개발에 대한 부담을 줄일 수 있고, 각 ASIC 소자의 클럭 변화로 인한 타이밍 생성기의 재 설계가 필요 없어 BIST 회로 동작의 안정성을 기할 수 있다.

상태 레지스터인 ASR은 RAM에 한 워드가 인가 될 때마다 어드레스의 상태를 저장하여 RAM 전체에 데이터가 인가되면 알고리즘 실행 유닛을 동기 시켜 테스트 알고리즘이 수행되도록 한다. 또한 DSR은 한 워드가 RAM에 직렬 인터페이싱 하는 상태를 저장하는 유닛으로, TGL의 동기신호에 의해 동작하며 한 워드가 RAM에 인가되면 ASR을 동기 시켜 테스트 알고리즘이 상태를 유지하도록 한다.

#### V. 고장 모델 및 테스트 알고리즘

알고리즘 실행 유닛에 사용된 알고리즘 중 March의 경우 Dual Port RAM 구조상 주로 발생하는 고장 인 DWCF(Dual port Word line Coupling Fault)와 DBCF(Dual port Bit line Coupling Fault)를 검출하기 위한 새로운 테스트 알고리즘인 DPMarch가 제안되었다. 먼저 DWCF와 DBCF의 고장 모델과 검출조건(그림 4)은 다음과 같다.

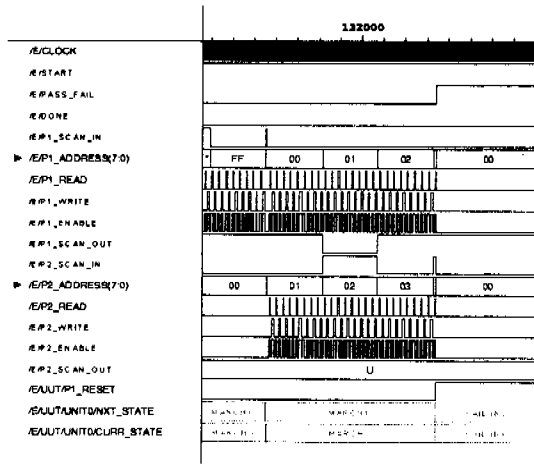


(그림 4) DWCF와 DBCF의 고장모델 및 검출조건  
 (Fig. 4) Fault Model and Detect condition of DWCF and DBCF

DWCF의 검출 조건은  $i = 0$  에서  $WS(\text{Word Size})-1$ 까지 메모리 셀  $C_i$ 에 0/1 토글 값을 입력한 후, 셀  $C_i$ 와  $C_j$ 에 대하여  $P1(A_i[R])$ 와  $P2(A_{i-1}[R])$ 를 수행하여 값을 다시 읽는다. (여기서 셀  $C_i$ 와  $C_j$ 는 워드 라인  $i$ 와 워드 라인  $j$ 에 연결된 인접한 메모리 셀을 가리키며,  $P(A_j[R])$ 는 포트  $i$ 에 관련된 오퍼레이션을,  $A_j$ 는 어드레스  $j$ 를,  $R$ 은 메모리 리드 오퍼레이션을 의미한다.) 만약 메모리 셀 중 DWCF가 존재하면  $WD1_i$ 와  $WD2_{i-1}$ 은 동시에 ON되며, 포트  $i$ 의 워드 라인  $j$ 를 나타내는  $WD_j$ 는 서로 다른 논리 값을 가지며 충돌하게 되는데 이것으로 존재하는 고장을 검출하게 된다.

DBCf의 테스트 패턴은 dual port RAM의 비트 라인 구조에 따라 결정되는데, BL1과 BL2가 인접한 메모리 구조인 경우 DBCf의 test 조건은 DWCF와 동일하다. 그러나, BL1과 BLB2가 인접한 구조인 경우 메모리 셀  $i = 0$ 부터  $i = WS - 1$ 까지 0 또는 1의 값을 연속적으로 기록하게 된다. 만약 DBCf가 존재하면  $WD1_i$ 와  $WD2_{i-1}$ 은 동시에 ON된다. 따라서 서로 다른 값이 충돌하게 되어 고장을 검출할 수 있게 된다.

그림5는 Synopsys Simulator를 이용하여, BIST 생성기에 의해 생성된 BIST회로가 고착고장이 발생한 RAM을 시뮬레이션한 결과를 나타낸다.



(그림 5) 고착 고장에 대한 시뮬레이션 결과  
 (Fig 5) The Simulation Result of Stuck-at Faults

## VI. 결 론

본 논문에서 제안된 BIST Builder는 Logic Level에서 하이 레벨 언어인 VHDL을 이용하여 기술 독립적인(Technology Independent)회로를 생성하고 테스트 알고리즘을 라이브러리로 만든 후 다양한 사양의 다중 RAM 모듈뿐만 아니라 라이브러리로 기술된 테스트 알고리즘도 필요에 따라 선택할 수 있도록 하였다. VHDL의 Generic Parameter와 C 언어를 이용한 VHDL 생성기의 개발로 다양한 RAM의 개수와 다중 RAM의 다양한 어드레스 크기나 워드 크기에도 BIST 회로나 RAM 모델의 수정이 필요 없는 범용 적이고, 이식성 높은 BIST 생성기를 구현하였으며 하나의 BIST회로가 다중의 RAM을 테스트하기 때문에, BIST회로로 인한 추가적인 칩 다 이 면적을 최소화할 수 있다는 장점 외에도 BIST 생성기의 제작으로 인하여 BIST회로 제작에 소요되는 많은 시간과 노력을 단축시킬 수 있을 것으로 기대된다. BIST 생성기 각 부분을 설계한 후, 테스트 벤치에 의한 시뮬레이션을 수행한 결과 BIST 생성기에 의해 생성된 BIST 회로는 선택된 알고리즘에 따라 RAM에 발생한 임의의 고장을 검출하여 외부로 Fail/Pass 신호를 발생시켰다.

결론적으로 본 논문에서 제안된 BIST 생성기는 ASIC에 내장된 다중 RAM모듈의 테스트를 용이하게 하며 회로의 재사용과 이식성(Flexibility)에 중점을 두어 메모리의 사양과 수행할 알고리즘에 따라 자동적으로 BIST 회로를 생성하였다.

## VII. 참고문헌

- [1] D.R.Aadsen, H.N.Scholz, and Y.Zorian, Automated BIST for Regular Strategies Embedded in ASIC Devices, AT&T Technical Journal, pp.97~108, May/June, 1990.
- [2] T.W.Williams and K.P.Parker, Design for Testability A Survey, Proc.of IEEE, Vol.71, No.1, Jan.1983
- [3] F.Tsui, Testing of Memory Parts, IBM Technical disclosure bulletin, Vol.25, No.3 A, pp. 12 16~1227, Aug., 1982
- [4] 서승모, 차세대 ASIC MEMORY 개발 현황 및 전망, 전자공학회지, 제20권, 제 11호, pp. 37~45, 11, 1993.
- [5] R.Collett, ASIC Design in the 1990s: Tools, Technologies and Tradeoffs, Nikei Electronics Asia, pp. 39~45, Oct., 1992.
- [6] R.Dekker, F.Beenker, and L.Thijssen, Realistic Built-In Self test for Static RAMs, IEEE Design & Test of Computers, pp.26~34, Feb., 1989.
- [7] S.K.Jain and C.E.Stroud, Built-In Self Testing of Embedded Memories, IEEE Design & Test, pp.27~37, Oct., 1986.
- [8] Z.Sun and L.T.Wang, "Self-Testing of Embedded RAMs," IEEE Test Conf., pp. 148~156, 1984
- [9] K.K.Saluja, S.H.Sung, and K.Kinoshita, "Built-In Self Testing RAM: A Practical Alternative," IEEE Design & Test, pp. 42~51, Feb., 1987.
- [10] B.Nadeau-Dostie, A. Silburt, and V.K.Agarwal, "serial Interfacing for Embedded-Memory Testing," IEEE Design & Test of Computers, pp. 52~63, Apr., 1990.
- [11] M.Franklin and K.K.Saluja, "Built-in Self-Testing of Random-Access Memories," IEEE Computer, pp. 45~56, Oct., 1990.