

테스트 용이성을 고려한 VLSI 設計 方式

林 寅 七

漢陽大學校 工科學科 電子工學科 教授(工博)

I. 序 論

VLSI가 대규모화, 복잡화 되어감에 따라 이에 대한 테스트 문제는 지극히 심각한 문제로 대두되고 있다. 금후 집적도가 점점 높아질 것으로 예상됨으로 테스트 문제의 해결 방안은 더욱 중요한 문제로 부각 될 것이다. 집적회로의 테스트 패턴 생성에 필요한 계산기 사용 시간은 회로 규모가 커질수록 지수함수적으로 급증하여, 종래의 방식으로는 수천 게이트를 초과하는 순서회로의 경우에는 테스트가 거의 불가능하게 된다. 이에 따라 IBM에서는 테스트가 용이한 회로 구조로 LSSD(level sensitive scan design) 방식을 개발하여 최근의 LSI 컴퓨터에 채용하고 있고, 그외에도 built-in test 방식을 비롯한 테스트 용이한 LSI/VLSI 설계 방식이 등장하였으며 앞으로도 이에 대한 연구가 더욱 활발해 질 것으로 예상된다.

본고에서는 VLSI 테스트를 용이하게 할 수 있도록 하는 각종 설계 방식에 관하여 서술하고, CMOS의 테스트와 PLA(programmable logic array)의 testable design에 비추어 따로 언급한다.

II. 테스트 방법

1. 일반적인 테스트 방법

일반적인 테스트 방법은 exhaustive testing, structured testing, functional testing의 세 가지로 나눌 수 있다.⁽¹⁾ Exhaustive testing 방법은 가장 단순한 방법으로 모든 조합 가능한 상태를 입력 테스트 벡터로 만들어 수행하는 방법이며, 조합 회로에서 입력수가 n일 때 테스트 패턴의 수는 2^n 개가 된다. 순서 회로에서는 입력수가 n이고 기억 소자수가 m이면 테스트 패턴의 수는 최소 2^{n+m} 개가 된다. 이 방법은 테스트 패턴 생성은 쉬우나, LSI/VLSI에서는 테스트 패턴의 수가 너무 많아지게 되므로 실현 가능성이 없는 방법이라 하겠다.

Structured testing 방법은 exhaustive testing 방법에서 불필요한 테스트 패턴을 제거하여 테스트 패턴의 수를 줄이도록 한 방법이다.

Functional testing 방법은 어떤 회로의 기능수행이 올바르게 회로가 정상이라고 판단하는 방법이다.

2. LSI/VLSI 테스트 방법

LSI/VLSI 테스트 방법을 특성에 의해 분류하면 다음과 같은 세 가지로 나눌 수 있다.⁽²⁾

1) Ad-Hoc Testability Approach

특별히 주어진 회로의 특성에 알맞게 그때 그때 임기응변적으로 테스트 할 수 있게끔 하는 것으로 특정회로중 테스트를 쉽게 할 수는 있으나 모든 회로에 일반적으로 적용할 수는 없다. 여기서 응답 평가에 대한 하나의 유용한 방법으로 signature analysis 방법이 있다. 이 방법은 테스트 패턴에 대한 응답을 자체적으로 알아내도록 고안된 방법으로서, 그림1에 LFSR(linear feedback shift register)을 이용한 signature analysis 방법을 나타냈다.⁽¹⁰⁾ L_1 은 A에 의해 동작되고 B에 의해 L_2 가 동작되어 신호가 다음 단으로 전해져, 그 출력은 exclusive-OR 회로에 연결되어 고장을 검출한다. LFSR은 매 cycle 같은 출발점을 갖도록 초기화시키고 응답이 들어와서 LFSR을 거쳐 나오면 고장을 검출할 수 있다.

2) Systematic and Structred Testability Approach

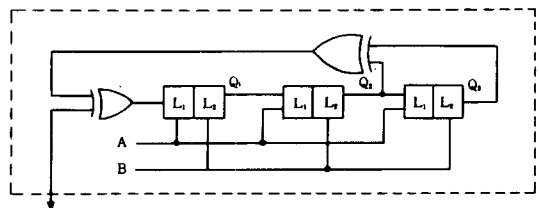


그림 1. Signature analysis

회로의 구조에 기초를 둔 체계적인 테스트 방법으로 자동테스트 패턴 생성이 효과적이며 경제적으로 실현될 수 있다. scan test 방법, random access scan 방법, LSSD 방법 등이 여기에 속한다.

3) Self Test and Built-In Test Approach

회로 특성을 이용하거나 부가 회로를 사용하여 테스트를 용이하게 하는 방법으로 self test나 BILBO (built-in logic-block observer)를 이용한 방법들이 여기에 속한다.

한편 LSI/VLSI 테스트 방법을 테스트하는 시간에 의해 분류하면 concurrent 테스트와 explicit 테스트로 나눌 수 있다.¹¹⁾ concurrent 테스트에서는 정상적인 user-application 입력 패턴들이 diagnostic 패턴들로 작용하므로 테스트가 정상 동작과 동시에 진행된다. Explicit 테스트에서는 테스트 입력으로 특별한 입력 패턴이 인가되므로 정상 동작과 테스트가 각각 다른 시간에 이루어지며, 일반적으로 테스트 패턴 생성, 테스트 패턴인가, 응답 처리 등 세 가지 과정으로 이루어진다.

현재는 부가 하드웨어의 양이 많아지고, 설계가 복잡하므로 concurrent 테스트가 많이 사용되지 않으나, LSI/VLSI의 복잡도가 증가할 수록 테스트 절차를 UUT(unit under test)에 집어 넣는 경향이므로 점차 사용이 증가 될 것이다.

LSI/VLSI 테스트에 있어 어떤 회로를 테스트하기 위해서는 그 회로를 외부에서 조정할 수 있어야 하고 (controllability), 이것에 의해 원하는 출력이 나오는 지를 관찰 할 수 있어야 한다. (Observability) 이 개념을 도입한 설계 과정을 살펴보면 그림2와 같다.¹²⁾

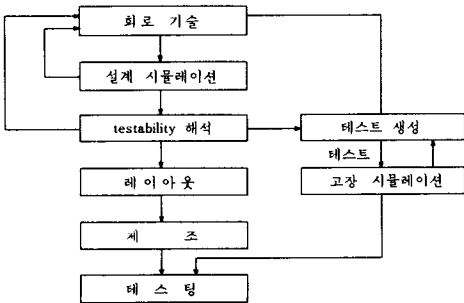


그림 2. Testability 설계 과정

Ⅲ. 구조화 설계방법 및 자동 테스트

1. Scan design

1) Scan Design의 원리

Scan design은 기억소자들을 포함하는 순서논리회로에 있어서 테스트 생성문제의 복잡성을 감소시킬 수 있는 매우 유용한 방법이다.¹³⁾ 이 방법은 테스트시 기억소자들이 쉬프트 레지스터의 형태로 구성되어져 쉬프트 기능을 갖게 됨으로써 이 기억소자들의 값을 외부에서 임의로 제어 (controllability) 할 수 있고, 그 기억소자들의 상태를 외부에서 관찰 (observability) 할 수 있게 해준다.

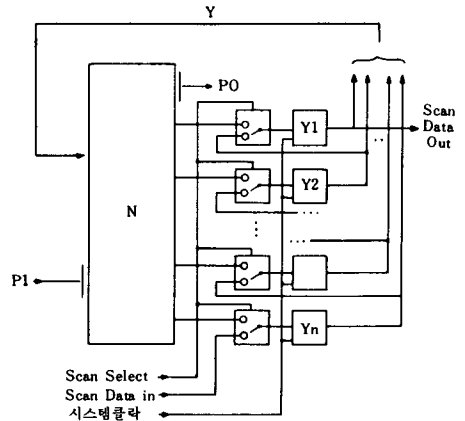


그림 3. Scan path의 원리

그림 3은 이 Scan path 동작을 임의의 순서회로에 적용하는 방법을 나타내고 있다. 여기서 모든 기억소자들이 연결되어 한 개의 scan path를 형성하고 있음을 볼 수 있다. 이 그림에서 각 기억소자들은 공통 scan select 신호로 연결된 2-way switch에 의해 그 동작이 제어된다. Scan select 신호가 "LOW" 상태일때 switch는 조합회로의 출력들을 기억소자 입력들에 연결시켜서 회로는 정상동작 (normal mode)을 수행하게 된다.

또한 Scan select 신호가 "high" 상태일때 기억소자들은 조합 회로와의 연결이 끊어진 채 serial-in, serial-out의 쉬프트 레지스터로 재구성되어 scan path 동작을 수행하게 된다. 이때 serial 데이터 입력은 "scan data in", serial 데이터 출력은 "scan data out" 이라고 한다. 따라서 scan mode에서 단지 scan data in 을 통해 입력값을 연속적으로 인가해 주고 시스템 클락으로 쉬프트 레지스터를 동기시킴으로써 각 기억소자의 값을 임의로 제어할 수 있다.

2) Random Access Scan (RAS)

Random access scan 설계 기술의 원리는 각각의 기

억소자들이 독립적으로 set 또는 preset 될 수 있고, 또 그 출력값들을 관찰할 수 있도록 개별적인 addressing을 허용해 주는 것이다. 이 원리는 그림 4에 나타나 있으며, 그림 5는 하나의 특별한 형태의 기억소자(이 경우는 preset/clear addressable latch)를 나타내 주고 있다.

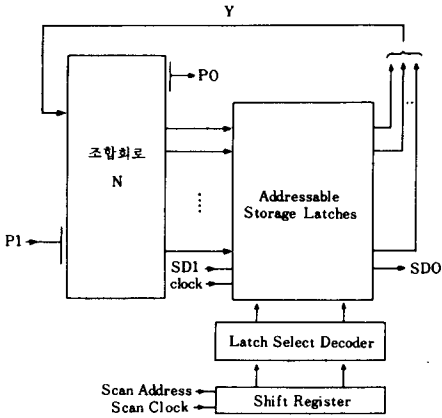


그림 4. Random-access scan

되고 이때 system data(D)의 변화는 system clock (CLK)이 low상태일때 출력 Q로 전달된다. Scan 동작은 scan clock(SCLK)에 의해 제어되며, 이때, 시스템은 high 상태로 유지되어야 한다. 원하는 래치가 선택되면 그래치의 값은 scan data in(SDI)을 통해 입력되는 값으로 세트 될 수 있으며, 그 값은 scan data out(SDO)에서 관찰할 수 있다.

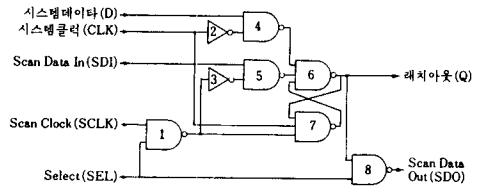


그림 6. Polarity-hold addressable 래치

Random access scan은 scan path를 포함하지 않는다는 점에서 기본 scan-path 개념과는 다르다. Random access scan의 커다란 단점은 테스트 입력값들을 래치들속에 세트시키고 그 응답값을 관찰하는데 너무 많은 시간이 필요하다는 것이다.

3) LSSD(Level Sensitive Scan Design)

① 원리

LSSD는 level sensitive design과 scan design 개념을 함께 갖는 설계 방식이다.^[2,3,4,13] Level sensitive 라함은 회로나 게이트의 지연시간, 상승시간, 하강시간 등에 관계없이 회로의 clock 레벨과 허용된 입력 신호의 변화에 따라서 회로가 동작되는 것을 말한다. Scan design에 대해서는 앞의 III. 1. 1)에서 설명한 바와 같다.

LSSD를 구성하는 기본적인 요소는 그림 7에 표시한 쉬프트 레지스트 래치(SRL)이다. 이래치의 정상 상태 동작은 다음과 같다. 입력 데이터 D는 클락C가 1일때만 래치에 저장되어 안정한값을 갖는다. 따라서 클락C에 의하여 D가 입력되게 되며 래치의 출력은

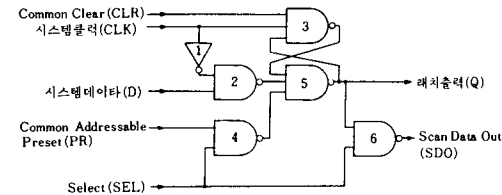


그림 5. Preset/clear addressable latch

그림 4에서 각각의 래치들은 scan-address 쉬프트 레지스터 값을 입력으로 하는 latch select decoder의 출력에 따라 선택된다. 그림 5의 preset/clear addressable 래치의 여러 가지 동작은 다음과 같다.

- (a) 래치 clear 동작 ; CLR=0, PR=0, CLK=1; Q=0
- (b) 정상 래치 동작 ; CLR=1, PR=0, CLK=0; Q=D
- (c) 래치 preset 동작 ; CLR=1, CLK=1, PR=1; Q=1
- (d) 래치 값 관찰; 원하는 래치의 값을 scan data out(SDO)을 통해 출력시킬 수 있다.

그림 6은 RAS에서 사용할 수 있는 또 다른 addressable 래치이며, 이래치의 동작은 다음과 같다. 먼저, 정상 동시에는 scan clock(SCLK)이 low상태로 유지

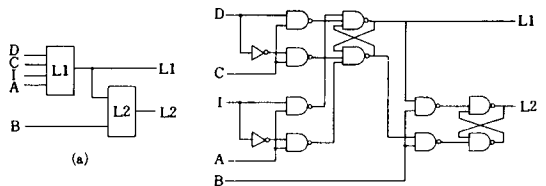


그림 7. 쉬프트 레지스터 래치(SRL)

D값과 같은 한 개의 값만 갖는다. 이때 클락 A, B는 0 값을 유지한다. 테스트 상태의 동작시에는 래치 L1, L2가 쉬프트 레지스터로 동작을 한다. 클락 A에 의하여 쉬프트 데이터 IN은 L1에 저장되고, 클락 B에 의하여 L1에 저장된 값이 L2로 쉬프트되어 저장된다. 이때 클락 A, B는 0 과 1 을 교대로 반복하는 클락으로서, 래치 L1과 L2의 동작을 각각 조정하게 되고 이때 클락C는 0 값을 유지하게 된다.

그림 8은 two-phase 시스템 클락을 갖는 LSSD system에 대한 일반적인 구조를 나타내고 있다. 이회로는 double-latch system이라고 불리는 그이유는 network N의 모든 시스템 입력들은 L1를 경유하여 L2에서 선택되기 때문이다. 클락 C2가 발생하면 SRL들의 몇몇 래치들은 L1 래치들에 저장되었던 값들로 인해 그 상태가 변화할 것이다. 이러한 변화들은 조합회로 N을 거쳐 전달되어야 하며 클락 C1이 발생하기 전에 X_1, X_2, \dots, X_n 에서 안정되어야 한다.

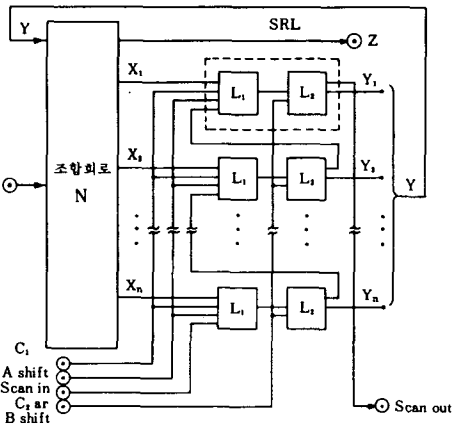


그림 8. LSSD 이중 래치 설계

그림 8의 조합회로 N부분에 대한 테스트는 다음과 같은 방법으로 수행된다.

- i) 일정한 테스트 패턴이 SRL (Y_1, Y_2, \dots, Y_n) 들로 쉬프트되어 주입력에 인가된다.
- ii) 신호들이 조합회로 N을 통과할 수 있는 충분한 시간이 경과한 후, 클락 C1은 X_1, X_2, \dots, X_n 신호들을 SRL들의 L1래치들에 저장할 수 있도록 충분히 high 상태를 유지한다.
- iii) L1 래치들속의 패턴들이 L2 래치들로 쉬프트 아우트되어 기대치 (expected response) 와 비교된다.

이러한 LSSD 방정식에서, 테스트시 그림 7의 쉬프트 레지스터 래치의 두개의 출력값들이 독립적으로 제어되지 못하는 결점을 해결하고, 테스트 패턴 생성을 용이하게 하며, 고장 검출율을 향상시키는 새로운 병렬 쉬프트 레지스터 래치의 설계 방식도 제안되었다.¹⁵⁾ 이 래치를 H. Fujiwara 등이 제안한 고장검출이 용이한 부가 PLA에 적용하므로써, 순서 논리회로의 PLA에서도 함수 독립적인 테스트 집합의 사용을 가능케하여 테스트 패턴 생성을 용이하게 하고 테스트 패턴의 수를 감소시킬 수 있다.

② 회로의 분할

조합 회로의 게이트 수가 N일때 자동테스트 생성을 위한 계산 시간은 N^2 에 비례한다. 테스트 패턴 생성 CAD로서의 한계는 게이트 수 3,000~5,000개이므로 대규모 조합회로를 분할하여 독립적으로 테스트하는 방법이 경제적이다. 이때 분할하는 방법은 첫째, 플립 플롭 또는 출력 단자를 출발점으로 backtrace 하여 플립 플롭이나 주입력에 도달할 때 까지의 원추형 회로가 단위 분할 회로가 되도록 분할한다. 둘째, 단위 분할 회로의 규모가 3,000~5,000 게이트를 초과할 경우 (backtrace가 bus 회로에 도달할 경우)를 대비하여 control 플립플롭(CF)을 backtrace 도중에 삽입한다. (NEC : DA conf. 1978). CF와 같은 목적의 selective control 회로를 삽입할 수도 있다 (IBM : Semiconductor Test Symp. 1978).

③ 메모리가 내장된 LSI/VLSI 테스트

이때의 테스트 순서는 플립플롭 → 메모리 → 조합회로이며, 주 입력과 플립플롭에 인가한 테스트 패턴이 메모리의 입력 라인에 가해지도록 한다. 또한 메모리의 출력 데이터를 회로의 출력과 플립 플롭을 통해 관찰하여 테스트한다. 이러한 테스트 방법은 테스트 시간이 많이 걸리며 테스트 패턴 생성에도 문제가 있다. 그래서 NEC에서는 LSI/VLSI의 메모리 부분과 타회로를 완전히 분리할 수 있는 분리용 회로를 추가하는 방법을 제안하였다. 이 방법은 회로량은 증가하나 테스트는 용이하다 (Semiconductor Test Conf. 1978).

4) 전달 지연 테스트

LSI 회로가 logic 테스트에 합격되었다고 하더라도 회로의 delay time에 이상이 있을 경우 정해진 값의 clock rate로써 정상적인 동작을 하지 않는 경우가 흔히 있다.

종래에는 정밀한 지연 테스트가 불가능했으며, 다만 clock rate를 바꾸어 가면서 기능 테스트만을 하였다. 그러나 LSSD 방법을 사용함으로써 이러한 지

연 테스트의 가능성이 매우 높아졌으며, 이는 LSSD의 큰 장점중의 하나로 간주되고 있다.

일반적으로 순서 논리회로에서는 지연 테스트를 위하여 다음과 같은 경로를 포함하게 된다.

- (a) 주 입력(primary input) → 조합회로→출력
- (b) 주 입력 → 조합회로→플립플롭
- (c) 플립플롭 → 조합회로→출력
- (d) 플립플롭 → 조합회로→플립플롭

LSSD 방법을 사용하면 종래에는 테스트가 곤란했던 (b), (c), (d)의 지연 테스트(delay test)가 가능하다. 또한 LSSD 방식을 응용함으로써 sequential PLA 회로에서의 효과적인 지연 테스트가 가능한 설계 방식도 제안되었다.^[11]

5) Scan Design의 장점 및 단점

① 장점

(a) 설계확인(Design Validation)과 Timing 분석
 일련의 디자인 rule의 위배성에 대한 자동적인 checking을 수행함으로써 설계 확인 작업과 timing분석이 간단화 된다.

(b) 테스트 패턴 생성

Scan 디자인 된 회로에서는 다만 조합 회로에 대한 테스트 생성만이 요구된다.

(c) 테스트 패턴 평가

Fault simulation에 의해 그 평가가 수행된다면 simulator는 단지 조합회로에 인가된 테스트 패턴에 대한 fault-coverage를 평가하면 된다.

(d) 테스트 인가와 고장진단(Fault Diagnosis)

회로를 scan mode에서 동작시킴으로써 feedback 경로를 open 시킬 수 있으므로 틀린 진단이나 애매한 진단이 나올 수 있는 주 원인을 제거해 준다. 테스트 인가에 있어서의 장점은 회로가 tester에 대한 uniform 하고 well-define 된 interface를 제공해 준다는 것이다. 즉, set-up instructions, scan-path test, 조합회로 test 등의 test program의 구성이 크게 다르지 않는다는 것이다.

② 단점

(a) 부가 하드웨어의 요구

여분의 주입력/출력 access pin들과 논리 디바이스들이 요구된다. 특히 LSSD의 경우 4 개의 부가 pin (SI, A, B, C)이 요구되며 부가 게이트들의 증가비율은 4 ~20% 정도이다.

(b) 테스트 시간의 증가

Scan design 된 회로를 테스트하기 위해서는 회로를 scan-path mode에서 normal mode로 바꾸어 주어

야 하며 각각의 테스트가 load되고 인가 될 때 마다 이 과정이 반복되어야 한다.

(c) 설계자의 자유에 대한 제약

Scan design 방법은 전통적인 논리설계에 주어졌던 자유에 제약을 준다. 특별히 설계자는 speed를 높이기 위한 비동기적인(asynchronous) 설계를 할 수 없게 된다.

6) 테스트 자동 생성 프로그램의 고속화(PODEM)

LSSD회로(조합회로 부분)에 대한 테스트 자동 생성을 고속화하는 방법으로서 PODEM(path oriented decision making)이 제안 되었다.^[12] 이 방법은 종래의 D-algorithm등의 경로 활성화법을 바탕으로 한 것으로 회로의 외부 입력에 대해서만 논리값을 할당하고 테스트 생성 과정에 모순이 발생하는 확률을 적게 하며 모순이 발생했을 때는 빠르게 새로운 외부 입력 값을 할당할 수 있다. 이 방법은 최악의 경우에 모든 가능한 입력 변수값을 할당하게 되므로 검출 가능한 고장에 대한 테스트를 구하는 완전한 알고리즘의 하나이다.

다음 그림9는 PODEM에 의한 테스트 생성을 나타낸 것이다.

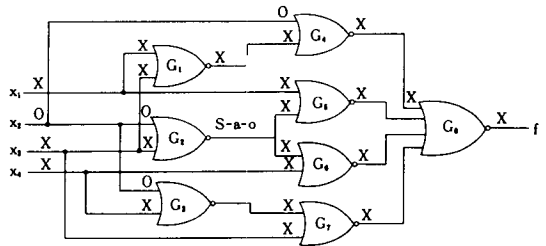


그림 9. PODEM에 의한 테스트 생성

우선 게이트 G₂의 출력 신호선의 s-a-o고장을 검출하기 위해 회로 입력 x₂에 0을 설정한다. 이때 게이트 G₂, G₃, G₄의 제 1 입력이 0으로 된다(그림9). 다음에 게이트 G₂의 출력을 D로 하기 위해 회로 입력 x₃에 0을 설정하면 게이트 G₁, G₂, G₃의 제 2 입력이 0으로 되고 게이트 G₂의 출력 및 G₅, G₆의 입력에 D가 나타난다. 이때 (x₁, x₂, x₃, x₄) = (X, 0, 0, X), 게이트 G₁부터 G₆의 입력은 각각 (X, 0), (0, 0), (0, X), (0, X), (X, D), (D, X), (X, 0), (X, X, X, X)이다. 다음은 게이트 G₅를 통해서 D를 전달해 주기 위해서 입력 x₁을 0으로 한다. 이때 (x₁, x₂, x₃, x₄) = (0, 0, 0, 0), G₁부터 G₆까지의 입력은 각각 (0, 0), (0, 0), (0, X), (0, 1),

(0, D), (D, X), (X, 0), (0, \bar{D} , X, X)이다. 그 다음 게이트 G_6 을 통하여 D를 전달해 주기 위해서 회로 입력 x_4 를 1로 하면 D가 출력에 전달되지 않는다. 이때 $(x_1, x_2, x_3, x_4) = (0, 0, 0, 1)$, G_1 부터 G_6 까지의 입력은 각각 (0, 0), (0, 0), (0, 1), (0, 1), (0, D), (D, 1), (0, 0), (0, \bar{D} , 0, 1)이다. 다시 x_4 의 값을 0으로 하므로써 테스트 생성이 가능하다. 즉, 게이트 G_2 의 출력 신호선의 s-a-0 고장을 검출하기 위한 테스트 입력은 (0, 0, 0, 0)가 된다.

이 방법은 종래 D-algorithm보다 빠른 방법이며, 최근에는 PODEM 보다도 더 고속화가 가능한 방법으로써 회로내 분지점 fanout에 착안한 algorithm(FAN)도 제안 되었다.

2. Self-Test Technique

1) Built-In Test 방법

① Built-In Test의 기본 단위

Built-in test 방법은 종래의 테스트 방법에서 문제 시 되었던 고장 modeling과 테스트 생성 알고리즘이 필요없고 테스트 응답 처리를 위한 비용 또한 매우 적게 든다는 점에서 현재 많은 연구가 진행되고 있다.^{8,10)}

Built-in test에 대한 기본 형태는 그림10과 같이 테스트 될 회로(circuit under test : CUT)의 입력측과 출력측에 테스트 패턴 생성과 테스트 응답처리를 위한 부가회로를 각각 삽입시키는 것이다. 이 부가 회로를 BILBO(built-in logic block observer)라고 부르며 입력측 BILBO는 TPG(test pattern generator)이고 출력측 BILBO는 SR(signature register)이다. 즉, TPG에서 테스트 패턴을 생성하여 테스트할 회로(CUT)로 입력시키며 그 응답은 SR에서 처리하게 된다. 이 SR의 신호는 fault-free 응답과 비교기에서 비교되어 테스트된 회로의 고장 발생여부를 알리는 go/no-go 신호를 발생시킨다.

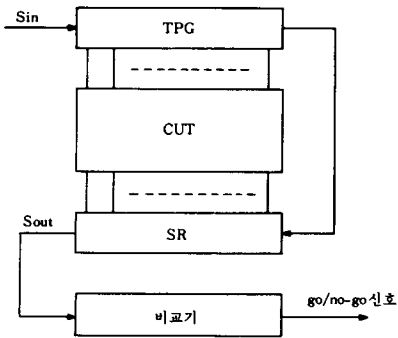


그림10. Built-in test의 기본 형태

② TPG와 SR

그림10의 테스트 패턴 생성 회로는 그림11과 같이 플립플롭의 열과 선형귀환으로 이루어지는 쉬프트 레지스터(linear feedback shift register : LFSR)로써, $2^n - 1$ (n 은 플립플롭수)개의 pseudorandom 패턴을 반복적으로 생성하게 된다. 즉, 모든 플립플롭의 값이 동시에 0이 아닌 register 초기치를 입력시킨후 클럭 펄스를 연속적으로 가하면 pseudorandom 패턴이 생성된다. 그러나 실제 테스트에서는 1주기만을 사용한다.

이때 LFSR의 귀환 경로는, 귀환 경로를 다항식으로 표시했을 때 primitive 다항식이 되도록 결정하여야 한다.

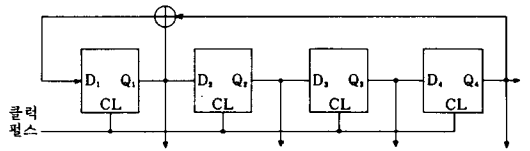


그림11. 테스트 패턴 생성회로

Built-in test의 테스트 응답 처리 방법은 CRC(cyclic redundancy check)코드를 이용하는 "signature analysis"를 사용하고 있다. 이 signature analysis를 수행하는 회로 역시 primitive 귀환 경로를 갖는 LFSR인데, TPG와 다른점은 그림12와 같이 매 클럭마다 테스트 응답 입력이 병렬로 exclusive-OR 게이트를 통해 입력된다는 점이다.

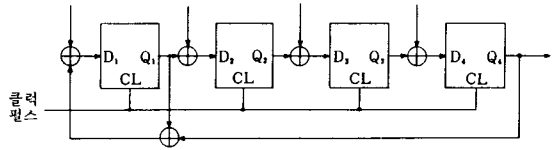


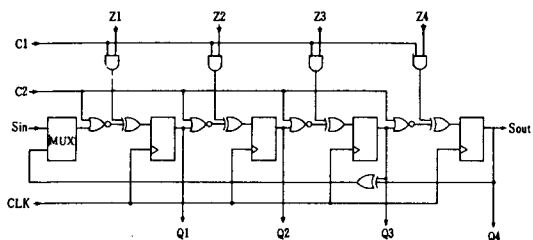
그림12. 테스트 응답 처리 회로

이런점에서 이 회로를 MISR(multiple input signature register)이라 부르기도 하며 이 MISR의 초기치는 어떠한 논리조합 값을 주어도 관계없다.

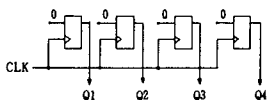
③ 다기능 BILBO

Built-in test의 테스트 패턴은 pseudo random 패턴이나 2진 카운터 회로에서 발생되는 패턴을 사용하기 때문에, LSI/VLSI 회로에 적용하였을 경우 입력되는 테스트 패턴의 수가 많아져 테스트 시간이 길어

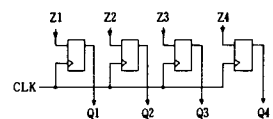
진다. 이러한 점은 큰 회로를 몇 개의 모듈로 회로 분할하여 테스트함으로써 해결할 수 있다. 이 경우 분할된 각 모듈에 대해 built-in test의 기본 형태와 같이 2개씩의 BILBO를 삽입하면 테스트를 위한 부가회로가 과다해진다. 그래서 각 모듈사이에 한개씩의 BILBO만을 삽입시켜 이 문제점을 해결하는데, 이때 BILBO는 다기능 즉, TPG, MISR, scan path 등의 기능을 갖고 이를 외부에서 제어해 줄 수 있어야 했다. 이 다기능 BILBO의 전형적인 형태는 그림13(a)와 같다. 이 다기능 BILBO는 2개의 제어신호 C_1, C_2 를 사용하며 이 제어신호에 따른 다 기능 BILBO의 동작은 그림13(b)~(e)와 같다. 즉, 이 다기능 BILBO는 정상 동작시의 단순한 래치를 비롯하여 테스트 동작시의 LFSR, scan path, reset 등의 기능을 모두 갖추고 있어서 대규모 회로를 분할하여 테스트할 경우 사용이 가능하다.



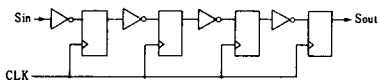
(a) 다기능 BILBO의 형태



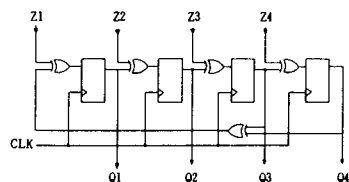
(b) Reset mode : $C_1=0, C_2=1$



(c) Normal latch mode : $C_1=1, C_2=1$



(d) Scan path mode : $C_1=0, C_2=0$



(e) LFSR mode : $C_1=1, C_2=0$

그림13. 다기능 BILBO

그러나 그림13(a)의 다기능 BILBO는 그림13(e)의 LFSR mode에서 TPG와 MISR의 기능이 완전히 분리되지 않는다. 즉, 이 다기능 BILBO가 TPG로 동작하기 위해서는 외부 입력 $Z_1 \sim Z_4$ 를 모두 "0"으로 고정시켜야 한다. 이 문제점으로 인해 그림13(a)의 다기능 BILBO는 C_1, C_2 두개의 외부 제어 신호만으로 BILBO의 기능을 완전히 제어할 수 없어 실제회로의 분할 테스트에 적용되기가 어렵다.

현재 이러한 문제점을 개선하여 외부 제어 입력만으로 다기능 BILBO의 기능을 완전히 제어할 수 있는 다기능 BILBO가 발표되어 있으며, 이 다기능 BILBO를 사용함으로써 분할된 여러개의 모듈을 한번에 테스트 할 수 있는 일괄 테스트까지 가능하게 되었다.^{10,11}

3. Totally Self-Checking(TSC)

임의의 회로를 테스트하기 위해서는 테스트를 위한 부가 회로가 필요하다. 그런데 테스트를 위한 부가회로에 고장이 발생하면 회로의 테스트가 곤란하며 회로의 정상동작에도 커다란 영향을 미친다. 이러한 문제점을 해결하기 위해 totally self-check 개념을 이용하고 있다.

TSC 회로는 부호화 입력과 출력을 가지며 정상동작시 check 회로에서 정동작과 오동작을 지시해 준다.

임의의 회로가 TSC 회로인지는 Anderson¹⁴에 의해 주어진 다음의 정의에 의해 결정된다.

[정의 1] 회로가 모든 입력코드에 대해 잘못된 코드출력을 발생시키지 않으면 고정집합(F)에 대해서 그 회로는 fault secure이다.

[정의 2] 회로가 최소한 1개의 입력코드에 대해서 비코드 출력을 발생시키면 F에 대해 self-testing 이다.

[정의 3] 회로가 fault secure와 self-testing 성질을 만족하면 그 회로는 TSC이다.

TSC 회로의 일반적인 구조는 그림14와 같으며 error indication은 회로가 정상동작시 정동작과 오동작을 알려준다.

그림15는 그림14의 TSC 회로를 기능회로와 검사회

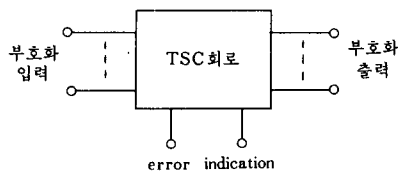


그림14. TSC 회로의 구조

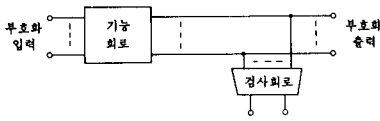


그림15. TSC model

로 분리하여 나타내었다.

그림15에서 기능회로는 부호화 입력에 대해 부호화 출력을 발생시키고, 검사회로는 부호화 출력에 대해 error indication 신호를 발생시켜야 하므로 기능회로와 검사회로는 TSC성질을 가져야 한다. 즉, 검사회로는 기능회로가 self-testing 성질을 갖도록 필요한 모든 테스트를 제공해야만 한다.

이러한 TSC 회로는 동시고장 검출(concurrent error detection)이 가능한 회로 설계 방법이며, 회로가 정상 동작시 error indication에 의해 permanent fault 뿐만 아니라 transient fault도 검출할 수 있다. 그러나 TSC 회로는 설계가 어렵고 부가회로가 많이 요구된다는 단점이 있다.

4. CMOS 회로의 테스트

COMS는 낮은 전력소모, 고집적도 등의 성질로 LSI/VLSI의 중요한 구성소자로 등장하고 있으며, 앞으로 그 사용범위는 계속 확대될 전망이다.

CMOS는 본질적으로 출력 low, 출력 high, 출력 open의 3가지의 상태가 존재할 수 있으며 출력 open은 출력이 high impedance로 되는 고장상태를 말한다. 즉, 개개의 트랜지스터에서 open 상태가 발생하므로서 생기는 고장 형태로 이를 stuck-open 고장이라 하고, 이러한 고장이 생길 경우 CMOS 회로의 charge store 기능으로 인해 전상태 값을 그대로 갖게 되므로, 조합 회로가 순서회로와 같은 동작을 행한다. 따라서 이와 같은 stuck-open 고장의 검출은 하나의 테스트 패턴으로 불가능하며, matching sequence라 불리는 두개의 테스트 시퀀스가 필요하다.

그림16은 그 입력 CMOS NAND 게이트 회로이고, "X" 표시가 발생가능한 stuck-open 고장이다. 여기서, 1 트랜지스터의 stuck-open 고장이 발생한다면 그림16의 회로는 그림17과 같이 될 것이다.

그림17에서 입력(X, Y)에 (0, 1)이 인가되면 V 트랜지스터의 stuck-open 고장으로 인하여 출력이 high impedance 상태가 되고, 이때의 정상출력을 F_{n+1} 이라 하면 고장시 출력은 F_n 이 된다. 즉, 고장이 발생하면 전 상태의 값을 가짐을 나타낸다.

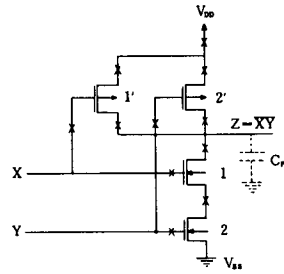


그림16. 그 입력 CMOS NAND 게이트 회로

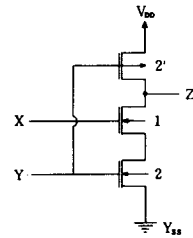


그림17. 그림16에서 1' 트랜지스터 고장시 회로

이때 1' 트랜지스터의 stuck-open 고장을 검출할 수 있는 matching sequence는 초기조건 (1, 1) 및 그 다음 패턴 (0, 1)로 이루어진다. 즉, (X, Y) = (1, 1)이 인가되면 정상출력은 0가 되고, 그 다음 (X, Y) = (0, 1)이 인가 되었을 때 1' 트랜지스터의 stuck-open 고장으로 인해 출력이 0 값을 계속 유지하면 (X, Y) = (0, 1)이 인가 될 때의 정상출력과 구별되므로 고장 검출이 가능하다.

CMOS 회로의 고장 검출은 이러한 방법으로 모든 트랜지스터의 stuck-open 고장을 검출할 수 있는 테스트 시퀀스를 구하는 것이다. 일반 CMOS 회로에 대하여 연결 그래프를 이용하여 테스트 시퀀스를 구하는 방법도 제안되고 있으며^(6,8), 테스트 입력 패턴이 다음 패턴으로 바뀔때 동시에 바뀌지 않거나, 또는 회로의 path의 길이가 다르기 때문에 생기는 Unequal delays 등에 의하여 time skew가 발생할 때는 고장을 검출할 수 없는 점을 해결하기 위하여 Domino CMOS 회로 방식을 사용하는 방법도 발표되고 있다.⁽⁸⁾

5. Testable PLA Design

PLA(programmable logic array)는 크게 decoder, AND array 및 OR array의 3부분으로 구성되어 있다. 그림18은 일반적인 PLA를 나타내고 있다.

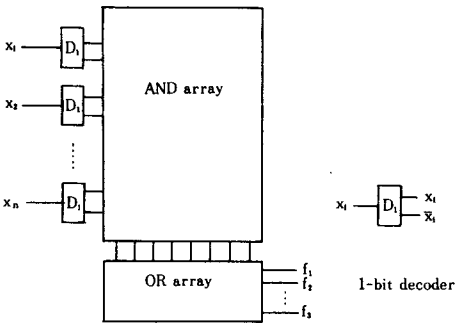


그림18. 일반적인 PLA 구조

PLA의 신뢰도를 향상시키기 위한 방법으로 하드웨어를 추가시켜 테스트를 용이하게 하는 설계방식이 있으며 다음과 같은 특징을 갖는다.¹⁵⁾

- 1) PLA의 입력단자수를 n, 적항수를 m이라면 테스트 입력패턴과 이에 대응하는 테스트 출력패턴은 PLA의 크기 n, m에 의해서만 결정되고, PLA의 AND array 및 OR array의 접속패턴에 의존하지 않는 함수독립적 테스트 집합 갖는다. 따라서 PLA의 함수에 상관없이 별도의 테스트 패턴 생성이 필요없게 된다.
- 2) 이러한 함수 독립적 테스트 집합의 테스트수와 테스트 길이는 $0(n+m)$ 이다. 부가 하드웨어가 없는 보통의 PLA에서는 테스트수가 일반적으로 $0(nm)$ 인 것을 생각하면 여기에서의 검사수는 대단히 적고, 테스트하는 시간을 대폭적으로 단축할 수 있다.

하드웨어를 부가한 테스트가 PLA를 용이한 그림19에 나타낸다. 그림에서와 같이 그 부가논리는 1개의 쉬프트레지스터, 2개의 EOR cascade, AND array에 1개의 열과 OR array에 1개의 행을 추가하여 구

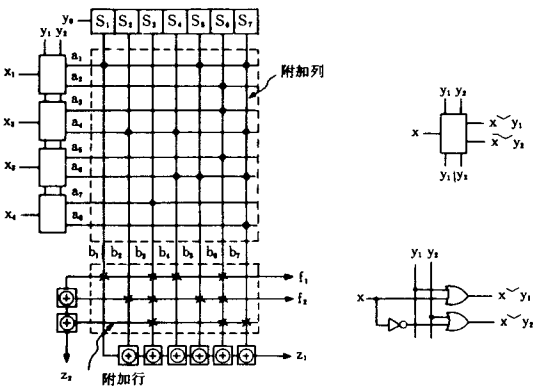


그림19. 고장 검출용 회로가 부가된 PLA

성된다. 이 경우 AND array에 부가한 최우단의 열에는 AND array에서의 각 행의 소자 접속수가 기수개가 되도록 소자접속을 해준다. 마찬가지로 OR array에 부가된 최하단의 행에는 OR array의 각 열의 소자 접속수가 기수개가 되도록 소자접속을 한다.

OR array의 i번째 입력신호 b_i 는 쉬프트레지스터 S_i 의 값에 따라 영향을 받게 된다.

$$b_i = P_i \cdot S_i \quad i=1, 2, \dots, m$$

여기서 P_i 는 AND array의 i번째 적항이다.

고장 검출용 회로가 부가된 PLA는 다음과 같은 특징을 갖는다.

- 1) 쉬프트레지스터는 선택하는 열에는 1을, 그 나머지 열에는 0을 인가하여 AND array의 임의의 열을 선택할 수 있다(그림20(a) 참조).
- 2) 제어선 y_1, y_2 를 갖는 수정된 decoder는 AND array의 임의의 행을 활성화하는데 사용된다(그림20(b) 참조).
- 3) OR array 아래의 EOR cascade는 AND array의 활성화된 행의 단일 고장을 검출하는 parity checker이다(그림20(c) 참조).
- 4) OR array 좌측의 EOR cascade는 OR array의 활성화된 열의 단일고장을 검출하는 parity checker이다(그림20(d) 참조).

위와 같은 고장검출용 회로가 부가된 PLA의 성질을 이용하여 다음과 같은 단일고장을 검출하는 함수 독립적 테스트 집합을 구할 수 있다.

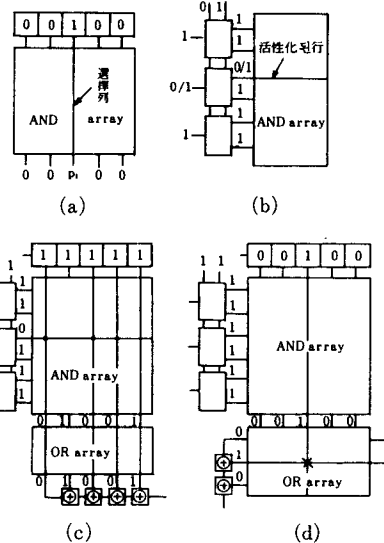


그림20. 고장검출용 회로가 부가된 PLA의 특징

- 1) Decoder, AND array, OR array 내의 출력선, gate의 stuck-at 고장
- 2) AND array와 OR array의 점접고장

표 1. 함수독립적 테스트 집합 $A_{n,m}$

I_n	$x_1 \dots x_1 \dots x_n y_1 y_2 s_1 \dots s_j \dots s_m$ ----- 0 0 0	$z_1 z_2$ 0 0
$j=1, 2, \dots, m$		
I_{2j}^0	0 0 0 1 0 0 0 1 0 0	1 1
I_{2j}^1	1 1 1 0 1 0 0 1 0 0	1 1
$i=1, 2, \dots, n$		
I_{2i}^0	1 1 0 1 1 0 1 1 1 1	$\epsilon_m -$
I_{2i}^1	0 0 1 0 0 1 0 1 1 1	$\epsilon_m -$

위와 같은 고장을 검출하기 위한 테스트 집합 $A_{n,m}$ 은 표 1과 같다. 여기서 n 은 입력수, m 은 AND array의 열수이고 ϵ_m 은

$$\epsilon_m \begin{cases} 0 & m \text{이 우수} \\ 1 & m \text{이 기수} \end{cases}$$

이다.

테스트집합 $A_{n,m}$ 으로부터 다음과 같은 고장검출용 회로가 부가된 PLA에 대한 테스트 시퀀스를 구할 수 있다.

$$a_{n,m} = I_{11} I_{21}^0 I_{22}^0 \dots I_{2m}^0 I_{21}^1 I_{22}^1 \dots I_{2m}^1 U_1 U_2 \dots U_{m-1} I_{31}^0 I_{32}^0 \dots I_{3n}^0 I_{32}^1 \dots I_{3n}^1$$

여기서 테스트 입력 U_i 는

$$x_1 \dots x_n y_1 y_2 s_1 s_2 \dots s_j s_{j+1} \dots s_m z_1 z_2$$

$$U_i = \text{----- } 1 \ 1 \ 1 \ 1 \ \dots \ 1 \ 0 \ \dots \ 0 \ \epsilon_m -$$

이다.

테스트 시퀀스 $a_{n,m}$ 은 PLA내의 모든 단일 stuck-at 고장, 점접고장 뿐만 아니라 슈프트레지스터의 기능도 테스트 할 수 있다. 이 테스트 시퀀스는 PLA의 크기 n, m 에만 의존하므로 함수 독립적이 되며, 그 길이는 $2n+3m$ 이 된다.

이러한 부가 하드웨어를 사용하여 테스트를 용이하게 하는 설계방식은 부가 하드웨어량이 많게 되고, 테스트 입력제어를 위한 부가회로의 사용으로 정상동작시 시간치연이 발생하게 된다.

이러한 문제점을 해결하기 위해, 부가 하드웨어가 적고 이로인한 영향이 회로에 미치지 않으며 테스트 패턴 생성이 용이한 설계방식도 제안되고 있다.^{11,17)}

결 언

LSI/VLSI의 집적도가 놀라울 정도로 증가함에 따

라 종래의 방법으로는 테스트가 불가능하게 되었으며, 테스트 패턴의 수와 비용이 과다하게 요구되므로 testability에도 문제가 생기게 되었다. 이러한 LSI/VLSI를 테스트하기 위해서 설계시 테스트를 고려하여 설계하는 테스트 용이한 설계의 개념 및 방법에 대하여 서술하였다. Random access scan, LSSD, BILBO, TSC 등에 대하여 구체적으로 언급하였으며, 각각의 방법들은 서로가 장단점을 갖고 있다.

CMOS 회로 및 PLA는 그 중요성 및 특수성으로 인하여 따로 언급하였다.

앞으로는 VLSI/VLSI의 대규모화, 복잡화에 따라 고장의 종류 및 수효를 고려하여 체계적인 구조를 가지면서도 자동으로 테스트 패턴을 생성할 수 있는 테스트 용이한 설계 방법이 연구 개발되어야 할 것이다.

參 考 文 獻

- [1] M.S. A badir and H.K. Reghbati, LSI Testing Techniques, *IEEE Micro*, pp. 34-51, Feb., 1983.
- [2] E.I. Muchldorf and A.D. Savkar, "LSI logic testing-an overview," *IEEE Trans. Computers*, vol. C-30, no.1, pp. 1-17, Jan., 1981.
- [3] T.W. Williams and K.P. Parker, "Design for testability-a survey," *IEEE Trans. on Computers*, vol. C-31, no.1, pp. 2-15, Jan., 1982.
- [4] E.B. Eichelberger and T.W. Williams, "A logic design structure for LSI testability," *J. Design Automat Fault-Tolerant Comput.*, vol. 2, pp. 165-178, May, 1978.
- [5] 한석봉, 조상복, 임인철, "Test 용이성을 고려한 LSI/VLSI 논리설계 방식과 PLA에의 응용", 대한전자공학회지, 제 21 권 제 3 호, pp. 26-33, 5, 1984.
- [6] Chiang, Kuang-Wei and Vranesic, Zvonko G., On Fault Detection in CMOS Logic Networks, *IEEE 20th D.A. Conf.*, pp. 50-56, 1983.
- [7] Chandramouli, R., *On Testing Stuck-Open Faults*. The 1983 International Symposium on Fault Tolerant Computing, Milano, Italy, June 28-30, 1983.
- [8] 조상복, 임인철, "CMOS Complex Gates의 테스트 생성 알고리즘", 대한전자공학회지, 제 21 권 제 5 호, pp. 55-60, 9, 1984.

- [9] Brend Konemann, et al., *Built-in Logic-Block Observation Techniques*. IEEE Test Conference,
- [10] Robert A. Frohwork, *Signature Analysis: A New Digital Field Service Method*. H.P. Jour., pp. 2-8, May, 1977.
- [11] 김동욱, 한석봉, 임인철, "Built-in test 방식을 이용한 LSI/VLSI testable design", 대한전자공학회 추계종합학술대회 논문집, vol. 6, no. 2, pp. 219-222, 11, 1983.
- [12] 可児賢二, 川西宏, 船津重宏, "超 LSI CAD의 基礎", オーム社, pp. 184-187, 1983.
- [13] R.G. Bennetts, *Design of Testable Logic Circuits*, Addison-Wesley, 1984.
- [14] D.A. Anderson and G. Metze, "Design of totally selfchecking check circuits for m-out-of n codes," *IEEE Trans. on Computers*, vol. C-22, no.3, Mar., 1973.
- [15] H. Fujjwara and K. Kinoshita, "A Design of programmable logic array with universal test," *IEEE Trans. Computers*, vol. C-30, pp. 823-828, Nov., 1981.
- [16] 김은성, 조상복, 임인철, "테스트가 용이한 PLA 설계 방식과 테스트 생성", 대한전자공학회 추계 종합학술대회 논문집, vol. 7, no. 2, 11, 1984.
- [17] H. Fujiwara, "Design for testability in PLAs," 일본전자통신학회논문지, vol. J 60-D, no. 9, 9, 1983.
- [18] 이종철, 임인철, "Delay testing의 용이성을 고려한-Sequential PLA의 Testable design", 대한전자공학회 추계종합학술대회 논문집, vol. 7, no. 1, pp. 255-259, 7, 1984. *

◆ 用 語 解 說 ◆

超格子 (Supperlattice)

서로 다른 결정재료를 수nm씩 교대로, 주기적으로 배열한 구조의 결정 격자, 예를 들면, GaAlAs 계에서 Al의 조성비율이 수 nm마다 주기적으로 변화되는 다층결정을 제작한다면, band 구조가 변형되고, 보통의 Al가 Ga이 불규칙적으로 배열하고 있는 물질과는 다른 물리적 특성을 나타내는 새로운 물질이 형성된다.

Stoichiometry

화학 반응에서 생성물의 양적 관계에 관한 것으로, GaAs 화합물의 경우 Ga와 As의 mole비가 1대1로 된 상태를 지칭한다. 실제의 GaAs 결정은, 제작 조건에 의존하고, Ga과잉 또는 As과잉의 결정으로서 안정하게 존재한다.

空格子點 (Vacancy)

결정내의 격자점이 점유되어 있지 않는 격자 결합의 일종.

DLTS (Deep Level Transient Spectroscopy)

결정내의 깊은 준위(캐리어 포획중심)의 에너지와 포획률을 평가하는 방법, GaAlAs에서는, 특히 깊은 준위의 불순물 농도가 결정성장 조건에 강하게 의존하고 있고, 결정질의 평가법으로서 이용된다. 온도를 走査하면서 접합용량의 시간변화를 측정하므로써 깊은 준위의 정보를 얻는다.

Photo luminescence

광여기에 의한 재결합 발광의 Spectre를 관찰하고, 발광 peak energy로부터 반도체의 혼합결정비와 불순물의 종류를 추정하는 방법, 절대 강도와 spectre폭으로 부터 결정의 광학특성을 평가 할 수 있다.

EPMA (Electron Probe Micro Analysis)

물질의 전자線을 조사하여, 발생하는 특성 X선을 분석하므로써 물질의 정량元素분석을 행하는 방법. Photoluminescence의 peak波長으로부터 평가하는 조성비 분석보다도 직석보다도 직접적 측정으로 미소면적의 측정도 가능.