

論文94-31B-6-8

근본 출력에 근거한 고장 모의실험

(A Fault Simulation Method Based on Primary Output)

李商高*, 朴圭皓**

(Sang Seol Lee and Kyu Ho Park)

要約

본 논문에서는 조합 회로의 근본 출력에 근거한 고장 모의실험 방법을 제안한다. 결정적 검사 신호 발생에서 각 검사 신호는 하나씩 얻어진다. 얻어진 신호는 검사중인 회로의 근본 입력에 인가되어 고장을 모의실험하게 된다. 고장은 각 근본 출력의 관점에서 추적 검출된다. 각 근본 출력으로 나타난 고장 검출은 검출을 위한 단어의 해당 비트 위치에 반영되고, 동적 고장 전파로 재수렴 줄기에 대한 효율적인 고장 검출이 이루어진다. 여러 개의 벤치마크 회로에 대한 고장 모의실험을 해 보았다. 그 결과 활성화 해야 되는 게이트의 수가 근본 출력에 근거하지 않는 방법에 비해 현저히 줄어들음을 확인하여 제안된 고장 모의실험 알고리즘의 성능 향상을 입증했다.

Abstract

In this paper, we propose a fault simulation method based on primary output in combinational circuit. In the deterministic test pattern generation, each test pattern is generated incrementally. The test pattern is applied to the primary inputs of circuit under test to simulate faults. We detect the faults with respect to each primary output. The fault detection with respect to each primary output is reflected by the corresponding bit in the detection words, and efficient fault detection for the reconvergent fan-out stem is achieved with dynamic fault propagation. As an experimental result of the fault simulation with our method for the several bench mark circuits, we illustrated the good performance showing that the number of gates to be activated is much reduced as compared with other method which is not based on primary output.

Key words : fault simulation, combinational circuit, primary output, dynamic fault propagation

1. 서론

* 正會員, 圓光 大學校 電氣 工學科

(Dept. of Elec. Eng., Wonkwang Univ.)

** 正會員, 韓國科學技術院 電氣 및 電子工學科

(Dept. of Electrical Engineering, KAIST)

接受日字 : 1993年 10月 8日

반도체의 고집적화와 대량 생산은 완제품에서의 오동작을 방지하는 사전 검사의 중요성을 인식시켜 왔다. 반도체 검사는 점점 더 중요한 위치를 차지하고 있고, 반도체 검사에 있어 고장 모의실험(fault

simulation)은 얻어진 검사 신호(test pattern)로 검출할 수 있는 모든 고장을 모의 실험으로 찾아내는 과정이다. 많은 고장 모의실험 방법들이 고안되어 왔고 [1, 2, 8, 10, 13], 특히 컴퓨터 단어의 각 비트에 서로 다른 병렬 신호(parallel pattern)들을 대응시켜 병렬로 고장 모의실험하는 방법들이 큰 성과를 얻었다. [8] 그런데 결정적 검사 신호 발생(deterministic test pattern generation) 과정에서는 검사 신호가 하나씩 얻어지므로 [5, 4, 9, 11] 병렬 신호를 이용한 방법들은 무작위 검사 신호 발생(random test pattern generation) 과정에서 이용된다. [8, 13] 결정적 검사 신호 발생 과정에서 얻어진 개별 신호로 고장 모의실험을 수행할 때 연산의 기본단위는 단어(word)가 된다. 고장 모의실험 과정중에 한 신호만을 한 단어에 대응시켜 처리하는 방법은 그 단어에 포함시킬 수 있는 정보 용량을 비효율적으로 사용하는 결과가 된다. 스캔 설계(scan design)로 조합 회로에 대한 검사 기법의 유용성이 한층 강조되고 있다. 본 논문에서 조합 회로의 고장 모의실험을 주어진 신호에 대해 근본 출력과 연관시켜 단어를 비트 단위로 세분화 하여 처리한다. 이 방법에 의하면 고장 모의실험중 단어의 한 비트는 대응된 근본 출력에 대한 고장 검출(fault detection) 정보를 반영하고 있다. 재수렴 줄기의 고장이 검출 되려면 그 고장이 근본 출력중 한 곳 이상으로 전달 됨을 확인해야 한다. 근본 출력에 근거한 방법에 의하면 줄기의 가지들에 있는 근본 출력에 대한 고장 정보만을 이용하여 간단히 재수렴 줄기의 고장 검출 여부를 알아낼 수도 있다. 만일 재수렴 줄기의 고장 검출이 가지들의 정보만으로 확인될 수 없으면, 근본 출력에 근거한 동적인 고장 전파로 확인한다. 근본 출력에 근거한 동적인 고장 전파를 이용하면, 고장 검출을 확인하기 위해 활성화 되어 하는 게이트들의 수가 정적인 고장 전파보다 적게 된다. 활성화 되는 게이트가 많을 수록 고장 모의실험 시간이 늘어난다. 이와 같이 근본 출력에 근거한 고장 검출을 이용하면 그렇지 않은 경우에 비하여 효율적인 고장 모의실험을 수행할 수 있다.

본 논문의 구성은 2 장에서 고장 모의실험에 필요한 정의와 고장 검출상의 성질들을 기술하고, 3 장에서 근본 출력(primary output)에 근거한 재수렴 줄기의 고장 검출을 소개한다. 4 장에서는 제안된 근본 출력에 근거한 고장 모의실험 알고리즘에 대해 설명하고, 5 장에서는 주어진 벤치마크 회로에 대해 [3] 알고리즘의 성능 향상을 실험을 통해 입증한 후, 6 장에서 결론을 짓는다.

II. 정의 및 고장 검출 성질

주어진 검사 신호에 대해 어느 선의 고장이 검출을 확인하기 위해서는 그 고장이 근본 출력까지 전파되어 관측됨을 확인해야 한다. 이 장에서는 고장 모의 실험 알고리즘의 설명을 위해 용어들을 정의하고 고장 검출에 대한 성질을 다룬다.

[정의 1] 어느 줄기가 서로 다른 가치를 통해서 동일한 게이트에 도달할 수 있을 때, 이 줄기를 재수렴 줄기(reconvergent fan-out stem)라 한다. 줄기가 서로 다른 어떤 가치를 통해서도 동일한 게이트에 도달할 수 없을 때, 이 줄기를 비수렴 줄기(non-reconvergent fan-out stem)라 한다.

그림 1의 예제 회로에서 재수렴 줄기는 c 와 i 이고 비재수렴 줄기는 m 이다.

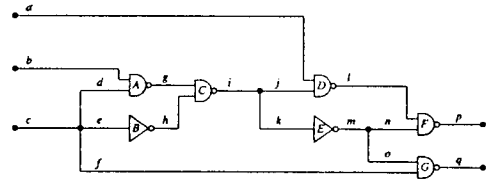


그림 1. 예제 회로

Fig. 1. An example circuit.

[정의 2] 재수렴 줄기 s 의 일차 탈출 게이트 e 는 다음과 같이 정의된다.

- (1) s 로 부터 서로 다른 경로를 통해 e 에 도달되고,
- (2) s 로 부터 반드시 e 를 통해 도달되는 근본 출력이 적어도 하나 존재한다.

그림 1에서 재수렴 줄기 c 의 일차 탈출 게이트는 C 와 G 이고, 재수렴 줄기 i 의 일차 탈출 게이트는 F 이다.

[정의 3] 재수렴 줄기 s 의 일차 줄기 영역(primary stem region)은 s 에서 도달되고 일차 탈출 게이트에 이르는 모든 선과 게이트들 그리고 s 의 일차 탈출 게이트들로 구성된다.

그림 1에서 재수렴 줄기 c 의 일차 줄기 영역은 선 $d, e, f, g, h, i, k, m, o$ 와 게이트 A, B, C, E, G 로 구성되고, 재수렴 줄기 i 의 일차 줄기 영역은 선 j, k, l, m, n 과 게이트 D, E, F 로 구성된다.

[정의 4] 재수렴 줄기 s 의 탈출 게이트(exit gate)는 다음과 같이 재귀적으로 정의된다,

- (1) s 의 일차 탈출 게이트는 s 의 탈출 게이트이다.
- (2) s 와 탈출 게이트 사이에 있는 재수렴 줄기의 한 탈출 게이트를 e 라 할 때, s 로 부터 반드시 e

를 통해 도달되는 근본 출력이 적어도 하나 존재하면 e 도 s 의 탈출 게이트이다.

그림 1에서 재수렴 줄기 c 의 탈출 게이트는 C, F, G 이고, 재수렴 줄기 i 의 탈출 게이트는 F 이다.

[정의 5] 재수렴 줄기 s 의 줄기 영역(stem region)은 s 에서 도달되고 탈출 게이트에 이르는 모든 선과 게이트들 그리고 s 의 탈출 게이트들로 구성된다.

그림 1에서 재수렴 줄기 c 의 줄기 영역은 선 $d, e, f, g, h, i, j, k, l, m, n, o$ 와 게이트 A, B, C, E, F, G 로 구성되고, 재수렴 줄기 i 의 줄기 영역은 선 j, k, l, m, n 과 게이트 D, E, F 로 구성된다.

어느 선 l 이 근본 출력에 도달되는 정보를 $R(l)$ 로 표시하고 도달성이라 하자. $R_i(l)$ 은 선 l 과 i 번째 근본 출력 사이의 경로 존재 여부에 대한 진위를 나타낸다. 만일 $R_i(l) = 1$ 이면 선 l 과 i 번째 근본 출력 사이에 경로가 존재하고, $R_i(l) = 0$ 이면 선 l 과 i 번째 근본 출력 사이에 경로가 존재하지 않는다. $|R_i(l)|$ 은 진위를 정수화한 표시이다.

[성질 1] f 를 줄기 s 의 한 가지라 하고 N_{fout} 을 가지의 수라 할 때,

$$\exists i \left(\sum_{j=1}^{N_{fout}} |R_i(f_j)| > 1 \right)$$

이면, 줄기 s 는 재수렴 줄기이다.

(증명) 위 식은 i 번째 근본 출력에 도달되는 경로가 2 개 이상의 서로 다른 가지를 통해 시작됨을 나타낸다. 그런데 한 개의 근본 출력에 공통적으로 도달하려면 이러한 경로는 어 디선가 만나야 한다. 즉 서로 다른 가지를 통해 동일한 게이트에 도달하므로 정의 1에 의해 s 는 재수렴 줄기이다.

[성질 2] f_j, f_k 를 줄기 s 의 가지라 하고 N_{fout} 을 가지의 수라 할 때,

$$\exists i \left(\left(\sum_{j=1}^{N_{fout}} |R_i(f_j)| = 1 \right) \wedge \left(R_i(f_k) = 1 \right) \right)$$

이면, 줄기 s 는 가지 f_k 를 통해 i 번째 근본 출력에 다른 가지와 재수렴하지 않고 도달하고, 이 때 f_k 를 비재수렴 가지라 하자.

(증명) f_k 이외의 다른 가지를 통해 i 번째 근본 출력에 도달할 수 없으므로 줄기 s 는 가지 f_k 를 통해 i 번째 근본 출력에 다른 가지와 재수렴하지 않고 도달한다.

그림 1의 예제 회로에서 p 를 첫 번째 근본 출력, q 를 두 번째 근본 출력이라 하자. 표 1은 각 근본 출력으로의 도달성과 줄기의 재수렴 여부를 나타내고

있다. 줄기 c 는 첫 번째 근본 출력 p 와 두 번째 근본 출력 q 에 대하여 재수렴한다. 줄기 i 는 근본 출력 p 에 대해서는 재수렴하지만 q 에 대해서는 재수렴하지 않는다.

표 1. 도달성과 재수렴 여부

Table 1. Reachability and reconvergence.

선	R_1	R_2	$\sum R_1 $	$\sum R_2 $	비고
q	0	1			근본 출력 p에 재수렴하는 줄기
p	1	0			
o	1	1			
n	1	1			
m	1	1	1	1	
l	1	1	1	1	
k	1	1	1	1	
j	1	1	1	1	
i	1	1	2	1	
h	1	1	1	1	
g	1	1	1	1	근본 출력 q에 재수렴하는 줄기
f	1	1	1	1	
e	1	1	1	1	
d	1	1	1	1	
c	1	1	2	3	
a	1	1	1	1	

여러 가지 고장 모델중 선이 단일한 값으로 고정된다고 가정하는 단일값 고정 고장 모델(single stuck-at fault model)이 실제 고장 상황을 대변하는 가장 많이 사용되는 모델중의 하나이다.^[12] 어느 선 l 이 0으로 고정된 고장을 간단히 I_{s0} 라 하고 1로 고정된 고장을 I_{s1} 라 하자. 어느 검사중인 회로에서 선의 갯수가 n 이라 할 때 고장 축약(fault collapsing) 이전의 고장의 수는 $2n$ 이다.^[7] 주어진 한 검사 신호에 대한 고장이 없는 모의실험(fault free simulation)중 어느 선 l 의 값이 0 이라면 I_{s1} 만이 검출 대상이 되고 그 고장의 검출 여부를 검출성 $D(I_{s1})$ 로 나타내자. 주어진 검사 신호에 따라 선 l 의 검출 대상이 되는 고장의 검출성은 $D(I_{s0})$ 혹은 $D(I_{s1})$ 중 하나로 유일하게 결정된다. 그러므로 선 l 의 고장 검출성을 단순히 $D(l)$ 라 하자. 주어진 검사 신호에 대한 선 l 의 고장 검출 확인은 고장 l 이 한개 이상의 근본 출력까지 전파됨을 확인해하 한다. i 번째 근본 출력에 대한 검출성을 $D_i(l)$ 라 하면

$$D(l) = (D_i(l), \dots, D_i(l), \dots, D_{npo}(l))$$

이다. 이 식에서 npo 는 근본 출력의 수이다. $D_i(l)$ 은 npo 비트중 한 비트 만이라도 1이라면 해당 근본 출력으로 검출됨을 나타낸다. 선 l 로부터 근본 출력 방향으로 도달되는 선중 하나를 m 이라 할 때 $D^m(l)$ 은 고장 l 이 선 m 까지 전파되는지 여부를 나타낸다. 만일 npo 개의 근본 출력을 $p_1, p_2, \dots, p_n, p_o$ 라 하면

$$D(l) = D^{p1}(l) \vee D^{p2}(l) \vee \dots \vee D^{npo}(l) \quad (1)$$

이다. \vee 는 비트와 비트 끼리의 OR 연산자이다. 따라서

$$\begin{aligned} D^{n1}(I.) &= (D_1(I.), 0, \dots, 0) \\ D^{n2}(I.) &= (0, D_2(I.), \dots, 0) \\ &\dots\dots \\ D^{nnp}(I.) &= (0, 0, \dots, D^{npp}(I.)) \end{aligned}$$

이다.

표 1의 비교에 분류되었듯이 회로에서 선의 종류는 3 가지가 있다. 근본 출력, 게이트의 입력 그리고 줄기이다. 만일 선 l 이 i 번째 근본 출력이라 하면

$$\begin{aligned} D(I.) &= (0, \dots, D_i(I.), \dots, 0) \\ &= (0, \dots, 1, \dots, 0). \end{aligned}$$

만일 선 l 이 게이트의 입력이라 하고 선 o 가 그 게이트의 출력이라 하면

$$D(I.) = D^o(I.) \wedge D(o).$$

\wedge 는 비트와 비트 끼리의 AND 연산자이다.

[정리 1] 만일 선 l 이 비재수렴 줄기라 하고 $f_1, \dots, f_{N_{fout}}$ 을 줄기 l 의 가지들이라 하면

$$D(I.) = D(f_{1,*}) \vee D(f_{2,*}) \vee \dots \vee D(f_{N_{fout},*}).$$

(증명) 비재수렴 줄기의 가지들은 서로 다른경로를 통해 만나지 않는다. 즉 $R(f_i) \wedge R(f_j) = 0, i \neq j$ 이므로 상호 배타적이어서 $D(f_{i,*}) \wedge D(f_{j,*}) = 0$ 이다. 따라서 위 식이 성립한다.

그러나 l 이 재수렴 줄기이면 $D(I.)$ 을 구하기가 간단하지 않다. 재수렴하는 가지들의 검출성을 안다 하더라도 서로 상쇄할 수 있기 때문에 정확한 고장 검출을 확인하기 위하여 고장 전파를 해 보아야 한다.

III. 근본 출력에 근거한 재수렴 줄기의 고장 검출

재수렴 줄기 l 에 대한 고장 $I.$ 의 검출을 확인하는 가장 단순한 방법으로 식 (1)을 구현하여 확인할 수 있다. 그러나 모든 근본 출력까지 고장을 전파하여 확인하는 방법은 비효율적이다. 고장 전파 영역을 축소하여 효율을 꾀할 수 있다. 고장 전파 영역을 축소시키는 방법에는 2 가지가 있다. 하나는 정적인 방법이고 다른 하나는 동적인 방법이다. 정적인 방법은

본격적인 고장 모의실험 이전에 고장 전파 영역을 결정해 놓는 방법이고 동적인 방법은 고장 모의실험중에 검출된 고장 검출 상황에 따라 고장 전파 영역이 결정된다. 전자의 방법보다 후자의 방법이 더 많이 고장 전파 영역을 축소시키는 효과를 갖는다.

1. 정적인 고장 전파

이 방법은 근본 출력에 근거하지 않고 진행된다. Tulip이라 명명되는 한 방법은 줄기 영역에서 비줄기 영역으로 나가는 선들을 탈출선(exit line)이라 하고 상호 고장전파를 상쇄시킬 수 있는 영역은 줄기 영역 안으로 국한되므로 줄기 영역 안에서만 고장 전파를 실시한다. 재수렴 줄기 l 의 줄기 영역에서 나가는 탈출 선들을 e_1, \dots, e_n 이라 할 때,

$$D(I.) = (D^l(I.) \wedge D(e_1)) \vee \dots \vee (D^{D^n}(I.) \wedge D(e_n))$$

로 고장 검출성이 확인된다.^[8]

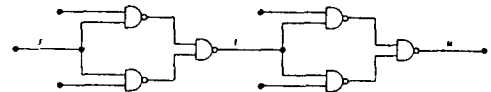


그림 2. 정적인 고장 전파 예제 회로

Fig. 2. An example circuit for static propagation.

예를 들면 그림 2에서 재수렴 줄기 s 의 고장 s 를 검출하기 위해 근본 출력 u 까지 고장을 전파하여 검출하기 보다는 탈출 선 t 까지 고장 전파하여 $D(s.)$ 를 얻은 후, $D(s.) = D(s.) \wedge D(t.)$ 을 통하여 $D(s.)$ 를 알아낸다.

줄기 영역이 일차 줄기 영역의 확장임에 착안하여 고장 전파 영역을 일차 줄기 영역까지로 축소시킨 방법도 있다.^[6]

이 방법들은 근본 출력까지 고장을 전파하지 않고 상호간에 간섭을 주지 않는 선 즉 탈출 선까지 고장을 전파하는 방법이다. 이 방법들은 재수렴 줄기에 대한 검출성을 결정하고자 할 때, 재수렴 줄기의 줄기 영역 혹은 일차 줄기 영역까지 고장 전파를 통해 $D(I.)$ 을 결정한다.

2. 동적인 고장 전파

동적인 고장 전파 방법에서는 줄기 영역까지 고장 전파를 하지 않아도 고장 검출을 결정할 수 있다. 제안된 고장 모의실험은 고장 검출을 주어진 검사 신호

에 대해 근본 출력에서 근본 입력(primary input) 까지 차례로 수행한다. 고장 모의실험중 재수렴 줄기에 대한 고장 검출이 필요할 때, 현재까지 알려진 고장 검출 정보를 이용하여 손쉽게 고장 검출을 확인할 수 있다. 근본 출력에 대해 추적되어 온 검출 정보를 이용하여 동적인 고장 전파를 효율적으로 성취할 수 있다. 재수렴 가지들의 고장 전파가 상호 간섭될 수 있는 영역은 줄기 영역으로 국한된다. 그런데 줄기 영역 안에서 고장 전파를 하지 않아도 수렴 줄기의 고장을 검출할 수도 있다. 예를 들어 그림 1에서 재수렴 줄기 i 의 고장 i 를 검출하고자 할 때 가지 k 의 도달성

$$R(j) = (R_1(j), R_2(j)) = (1, 0)$$

$$R(k) = (R_1(k), R_2(k)) = (1, 1)$$

와 성질 2로부터 줄기 i 는 가지 k 를 통해 다른 가지와 재수렴 없이 근본 출력 q 까지 도달할 수 있다. 그러므로 $D_2(k) = 1$ 이라면 $D_2(i) = 1$ 이다. 즉 정적인 방법에 의해 $D'(k)$ 와 $D'(i)$ 를 구하지 않고서도 고장 i 를 확인할 수도 있다.

어느 선 i 의 근본 출력에 근거한 고장 검출 수행 여부를 표시하기 위해 수행성 $E(i)$ 을 도입한다. 고장 i 이 i 번째 근본 출력으로 고장 검출 수행되었으면 $E_i(i)$ 는 1이다.

[정리 2] f_j, f_k 를 줄기 s 의 가지라 하고 N_{out} 가지의 수라할 때,

$$\exists i((\sum_{j=1}^{N_{out}} R_i(f_j)=1) \wedge (R_i(f_k)=1 \wedge (E_i(f_k)=1)))$$

이면 $D(s) = D_i(k_k)$ 이다.

(증명) 항 $\exists i((\sum_{j=1}^{N_{out}} R_i(f_j)=1) \wedge (R_i(f_k)=1))$ 은 성질 2로부터 가지 f_k 가 다른 가지들과 재수렴하지 않음을 나타낸다. 항 $E_i(f_k)=1$ 은 가지 f_k 의 고장 검출이 i 번째 근본 출력에 대해 이미 수행되었음을 나타낸다. 따라서 i 번째 근본 출력에 대한 고장 i 의 검출성은 고장 f_k 의 i 번째 근본 출력에 대한 고장 검출성과 같아진다.

정리 2에 의거한 고장 검출 확인은 줄기 영역에 대해 고장 전파를 전혀 수행하지 않고도 고장을 검출 결정할 수 있는 경우이다. 그런데 정리 2만으로는 재수렴 줄기에 대한 고장 검출 확인을 다하지 못할 때가 발생한다. 정리 2로 고장의 검출이 되지 않았고, 고장을 전파하여 검출을 확인할 여지가 있을 경우에는 줄기 영역의 일부 혹은 전부에서 동적인 고장 전파하여 검출성을 확인한다.

정의 4에 의해 재수렴 줄기 s 에서 그의 탈출 게이트 e 를 거쳐야만 도달되는 근본 출력이 적어도 하나 존재한다. 그 근본 출력을 재수렴 줄기 s 에 대한 탈출 게이트 e 의 탈출 근본 출력이라 하자.

[정의 3] 재수렴 줄기 s 의 어느 탈출 게이트를 e 라 하고 o 를 e 의 출력이라 할 때, i 번째 근본 출력이 s 에 대한 e 의 탈출 근본 출력이고

$$(E_i(o)=1) \wedge (D_i^o(s)=1)$$

이면 $D_i(s) = D_i(o)$ 이다.

(증명) i 번째 근본 출력으로의 고장 s 의 전파 경로가 o 를 경유할 수밖에 없고, 고장 전파를 통해 $D_i^o(s) = 1$ 로써 s 이 선 o 까지 전파됨이 확인되었고, $E_i(o) = 1$ 로써 o 의 고장 o 에 대한 검출이 이미 수행되었으므로 $D_i(s) = D_i(o)$ 이다.

그림 1의 회로에 검사 신호 $a=1, b=1, c=1$ 이 주어졌다고 하자.

$$D(i) = (D_1(i), D_2(i)) = (1, 0)$$

$$E(i) = (E_1(i), E_2(i)) = (1, 0)$$

인 상태에서 c 의 고장 c , 즉 c_{sa0} 의 검출을 확인하기 위해 고장 전파를 줄기 영역에서 시행중, 탈출 게이트 C 의 출력 i 까지 검출성이 $D'(s) = (D_1(s), D_2(s)) = (1, 1)$ 임을 알았다 하자. 첫 번째 근본 출력이 c 에 대한 C 의 탈출 근본 출력이고, $E_1(i) = 1$ 이고 또한 $(D_1(s) = 1$ 이므로, 정리 3에 의해 $D_1(s) = D_2(i) = 1$ 이다. 그러므로 재수렴 줄기 s 의 고장 s 은 검출된다. 즉 재수렴 줄기 c 의 고장 검출을 줄기 영역의 일부만의 고장 전파로 확인할 수 있다.

IV. 근본 출력에 근거한 고장 모의 실험 알고리즘

개괄적인 고장 모의 실험 순서는 그림 3과 같다.

재수렴 줄기에 대한 탈출 게이트와 탈출 근본 출력을 찾는다.

for 주어진 검사 신호들

 고장없는 모의 실험 수행

 고장 검출 수행

 고장 떨굼 수행

endfor

그림 3. 고장 모의 실험 알고리즘

Fig. 3. Algorithm of fault simulation.

그림 3의 고장 모의 실험 알고리즘에서는 우선 본

격적인 고장 모의 실험을 진행하기 전에 재수렴 줄기에 대한 탈출 게이트와 탈출 게이트의 탈출 근본 출력을 찾아 재수렴 줄기에 연결 리스트로 만든다. 그 후 각각의 주어진 검사 신호들에 대하여 고장이 없음을 가정하고 모의 실험하여 정상적인 동작을 알아낸다. 그리고 주어진 검사 신호에 대해서 고장 검출을 근본 출력에서 근본 입력까지 차례로 확인한다. 주어진 검사 신호에 의해 확인된 고장 검출을 이용하여 더이상 검출이 불필요한 고장들을 떨군다.

```

for 근본 출력에서 근본 입력에 이르는 각 선 l
  if( l = i 번째 근본 출력)
    D(l.) ← (0, ..., 1, ..., 0)
    E(l.) ← (0, ..., 1, ..., 0)
  elseif( l = 게이트 입력)
    D(l.) ← D'(l.) ∧ D(α)      /* α는 게이트의 출력 */
    E(l.) ← E(α)
  elseif( l = 비재수렴 줄기)
    D(l.) ← D(f1) ∧ ... ∧ D(fMout) /* fi는 가지, 1 ≤ i ≤ Mout */
    E(l.) ← E(f1) ∧ ... ∧ E(fMout)
  elseif( l = 재수렴 줄기)
    재수렴_줄기_고장_검출(l)
  endif
endif
endfor
    
```

그림 4. 고장 검출 알고리즘

Fig. 4. Algorithm of fault detection.

선 l의 고장 l에 대한 검출성 D(l.)과 수행성 E(l.)을 컴퓨터 연산 단위의 한 단어에 대응시켜 알고리즘이 동작된다. 또 D_i(l.)과 E_i(l.)은 D(l.)과 E(l.) 단어의 i 번째 비트에 대응되어 처리된다. 그림 4는 근본 출력부터 근본 입력까지 차례로 재수렴 줄기를 제외한 선의 고장 검출을 확인하는 알고리즘이다. 검출 대상이 근본 출력의 고장이면 항상 검출된다. 게이트 입력의 고장이면 그 게이트를 넘어 고장이 전파

```

알고리즘 재수렴_줄기_고장_검출(l)
for l의 가지 fi, 1 ≤ i ≤ Mout
  if( fi = 비재수렴 가지)
    D(l.) ← D(l.) ∧ D(fi)
    E(l.) ← E(l.) ∧ E(fi)
  endif
endif
endfor

/* 정리 3을 이용한 줄기 영역에서의 고장 전파 */
if( D(l.) = 0 )
  while( l의 고장 영역 전파 )
    g ← 고장 전파중의 게이트 /* α는 게이트 g의 출력 */
    if( g = 탈출 게이트 )
      M ← g의 독립 탈출 비트
      D(l.) ← M ∧ D'(l.) ∧ E(α) ∧ D(α)
      E(l.) ← M ∧ E(α)
      if( D(l.) ≠ 0 )
        return
      endif
    endif
  endwhile
endif
endfor
    
```

그림 5. 재수렴 줄기의 고장 검출 알고리즘

Fig. 5. Algorithm of fault detection at reconvergent fan-out stem.

될 수 있고 게이트 출력의 고장이 검출될 수 있을 때 검출된다. 비재수렴 줄기의 고장은 가지의 고장이 하나라도 검출될 수 있으면 검출된다.

재수렴 줄기에 대한 고장 검출은 그림 5에 나타나 있다. 한 근본 출력으로만 고장 전파가 확인되면 고장 검출이 확정되므로, 비재수렴 가지를 통해 고장 검출이 확인되면 줄기 영역에서 고장 전파는 필요없다. 고장 검출이 확인되지 않아 고장 전파가 필요하다면 도중에 정리 3을 만족하는 상황이 되어 검출이 확인되면 더 이상의 전파는 필요없다.

따라서 줄기 영역 전체에 걸쳐 고장 전파를 하지 않더라도 재수렴 줄기의 고장을 검출을 결정할 수 있다.

V. 실험 및 결과

재수렴 줄기의 고장 검출을 확인하기 위해서 필요에 따라 줄기 영역 안의 게이트들을 활성화 시켜 고장 전파를 해야한다. 따라서 고장 모의실험중 활성화 되는 게이트의 수가 적으면 적을 수록 알고리즘의 성능은 우수하다고 말할 수 있다. 고장 모의실험 알고리즘의 수행시간은 활성화 되는 게이트의 수에 비례한다. 제안된 알고리즘은 동적으로 결정되는 고장 전파 영역으로 인해 재수렴 줄기의 고장 검출에 활성화 되어야 하는 게이트의 수를 현저히 감소시킨다 여러 개의 벤치마크 회로에 대하여 고장 모의실험중에 활성화 되는 게이트의 수를 구하여 표 2에 보였다. 표 2에서 PO는 근본 출력(Primary Output)이다. 기존의 정적인 고장 전파 방법과 비교하여 평균 2 배의 성능 향상을 얻었고, 회로 c6288의 경우 최대 12 배의 향상된 성능을 얻었다.

표 2. 실험 결과

Table 2. Experimental results.

회로	게이트 수	근본 출력 수	고장 수	검사 신호	고장 검출률(%)	활성화 된 게이트 수 PO 근거	활성화 된 게이트 수 PO 비근거	성능 향상
c432	196	7	524	54	99.24	18436	50124	2.72
c499	243	32	758	52	98.68	30013	39194	1.31
c800	443	26	942	58	100.00	33076	4486	1.34
c1355	587	32	1574	87	99.49	64924	136754	2.11
c1908	913	25	1879	126	99.52	132214	249457	1.89
c3540	1719	22	3428	177	96.00	837105	1556484	1.86
c6288	2448	32	7744	34	99.56	145624	1785420	12.26

VI. 결론

출력에 근거한 고장 모의실험 알고리즘을 제안하였다. 이 알고리즘에서는 주어진 검사 신호에 대하여 근본 출력에 근거한 고장 검출을 해 나간다. 동적인 고장 전파 기법을 함께 적용하여 재수렴 줄기의 고장

검출에 좋은 성능을 발휘한다. 재수렴 줄기의 고장 검출 확인시에 줄기 영역의 모든 게이트들을 활성화 하지 않아도 고장 검출을 확인할 수 있다. 실험을 통해 근본 출력에 근거하지 않는 방법과 성능을 비교하여 보았다. 여러 개의 벤치마크 회로에 적용해 본 결과 약 2 배의 성능 향상을 확인할 수 있었다.

參考文獻

- [1] M. Abramovici, P.R. Menon, and D.T. Miller, "Critical path tracing: An alternative to fault simulation," *IEEE Design & Test of Computers*, vol.1, no.1, pp. 83-93, Feb. 1984.
- [2] D.B. Armstrong, "A deductive method of simulating faults in logic circuits," *IEEE Trans. on Computers*, vol.C-21, no.5, pp. 464-471, May 1972.
- [3] F.Brglez and H.Fujiwara, "A neutral netlist of 10 combinational benchmark circuits and a target translator in fortran," in Proc. International Symposium on Circuits and Systems, pp. 663-698, 1985.
- [4] H.Fujiwara and T.Shimono, "On the acceleration of test generation algorithms," *IEEE Trans. on Computers*, vol.C-32, no.12, pp.1137-1144, Dec. 1983.
- [5] P.Goel, "An implicit enumeration algorithm to generate tests for combinational logic circuits," *IEEE Trans. on Computers*, vol.C-30, no.3, pp. 215-222, Mar. 1981.
- [6] S.S. Lee and K.H. Park, "An efficient fault simulation method for reconvergent fan-out stem," *IEICE Trans. on Information and Systems*, vol.E76-D, no.7, pp. 771-775, July 1993.
- [7] E.J. McCluskey and F.W. Clegg, "Fault equivalence in combinational logic networks," *IEEE Trans. on Computers*, vol.C-20, no.11, pp.1286-1293, Nov. 1971.
- [8] F.Maamari and J.Rajski, "A method of fault simulation based on stem regions," *IEEE Trans. on Computer-Aided Design*, vol.CAD-9, no.2, pp. 212-220, Feb. 1990.
- [9] J.P. Roth, "Diagnosis of automata failures: A calculus and a method," *IBM Journal of Research and Development*, vol.10, no.4, pp. 278-291, July 1966.
- [10] M.H. Schulz and F.Brglez, "Accelerated transition fault simulation," in Proc. of the 24th Design Automation Conference, pp. 779-784, 1985.
- [11] 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.
- [12] C.Timoc, M.Buehler, T.Griswold, C. Pina, F.Scott, and L.Hess, "Logical models of physical failures," in Proc. International Test Conference, pp. 546-553, 1983.
- [13] J.A. Waicukauski, E.B. Eichelberger, D.O. Forlenza, E.Lindbloom, and T. McCarthy, "A statistical calculation of fault detection probabilities by fast fault simulation," in *IEEE International Conference on Computer-Aided Design*, pp. 779-784, 1985.
- [14] T.W. Williams and K.P. Parker, "Design for testability -- a survey," *Proc. of the IEEE*, vol.71, no.1, pp. 311-325, Jan. 1983.

 著 者 紹 介



李 商 高(正會員)

1960年 4月 18日生. 1984年 2月 고려대학교 전자공학과 졸업(공학사). 1984年 ~ 1986년 삼성전자 연구소 연구원. 1989年 2月 한국과학기술원 전기및전자공학과(석사). 1994年 2월 한국과학기술원 전기및전자공학과(박사). 1994年 3月 ~ 현재 원광대학교 전기공학과 전임강사. 주관심 분야는 컴퓨터구조, 설계자동화, 병렬처리 등임.

朴 圭 皓(正會員) 第 29卷 B編 第 4號 參照

현재 한국 과학기술원 전기 및 전자공학과 교수