

論文98-35C-11-2

시험성 분석 기법(ITEM)의 부분 스캔 성능 평가 (Partial Scan Performance Evaluation of Iterative Method of Testability Measurement(ITEM))

金炯局*, 李宰勳**, 閔炯福**

(Hyoung Kook Kim, Jae Hoon Lee, and Hyoung Bok Min)

요약

검사용이성 분석에서는 회로의 모든 선에서 제어율과 관측율 값을 계산하고 이를 기반으로 결함 시험도를 평가한다. 검사용이성 분석은 응용에 따라 제어율과 관측율 값을 이용하기도 하고, 결함 시험도 값을 사용하기도 한다. 검사용이성 분석 알고리즘 ITEM은 이미 결함 시험도 측정 관점에서 평가되었다. 하지만 부분 스캔과 같은 응용 분야를 위해 회로 내의 각 선들에 대한 제어율과 관측율 값도 중요한 의미를 가지므로 평가되어야 한다. 본 논문에서는 회로내의 각 선들에 대한 검출율 관점에서 STAFAN과 ITEM을 비교 평가하기 위해, 플립플롭을 스캔함에 따른 전체 회로의 검사용이성 영향을 분석하는 민감도 분석을 이용한 검사용이성 부분 스캔 기법을 통해 간접적으로 ITEM을 평가하였다. ITEM에 의해서 구해진 검사용이성은 STAFAN에 의해 구해진 것과 거의 유사한 값을 유지하였지만, 빠른 실행 시간을 보였다. ITEM은 부분 스캔과 실행 시간에 민감한 크기가 큰 회로에 있어서 효과적인 것으로 판단된다.

Abstract

Testability analysis computes controllabilities and observabilities of all lines of a circuit and then evaluates fault coverage. The values of controllability and observability as well as fault coverage produced by testability analysis are used for applications of testability analysis. ITEM was evaluated as a fault coverage tool. But the values of controllability and observability at all lines of circuits must be estimated as a performance measure of testability tools for another application such as partial scan. In this paper, partial scan method based on sensitivity analysis which estimates relative improvement of detectability of circuits after scanning a flip-flop is used for performance evaluation of ITEM. Performance of ITEM, with respect to testability values on each net, has been measured by comparing ITEM and STAFAN. Partial scan performance achieved by ITEM is very similar to that of STAFAN, but ITEM takes less CPU time. Therefore ITEM is very efficient for partial scan application because ITEM runs faster for very large circuits in which execution time is critical.

* 正會員, 三星電子 메모리 開發 事業部 應用 技術팀

DRAM企劃

(SAMSUNG Electronics. Memory Technology & Product Division)

** 正會員, 成均館大學校 電氣工學科

(School of Elec. computer Eng. Sungkyunkwan Univ.)

※ 본 연구는 1997년도 한국 과학 재단 연구비 지원에 의한 결과임(과제 번호 96-0102-16-01-3).

接受日:1998年7月22日, 수정완료일:1998年9月29日

I. 서론

오늘날 VLSI 기술의 발달로 시스템의 기능과 신뢰성에 급격한 향상을 가지고 온 반면, 회로 복잡도의 증가로 회로에 대한 검사를 더욱더 어렵게 만들었고, 칩의 생산단가에서 차지하는 검사 비용을 증가시켰다. 따라서 검사 비용을 절감하기 위해 회로의 검사 단계에서부터 검사를 고려하는 것이 일반적인 추세이다.

회로에 고장이 발생했을 때, 임의의 검사 신호를 이용하여 그 고장을 검출할 수 있는 확률을 결함 시험도

(fault coverage)로 정의한다.

결함 시험도를 정확하게 계산하기 위해 결함 시뮬레이터(fault simulator)를 사용한다. 그러나 결함 시뮬레이터의 실행 시간이 회로를 구성하는 게이트 수의 제곱에 비례하므로^[1] 칩의 집적도가 높아질수록, 실행 시간은 급격하게 증가한다. 이러한 문제점을 해결하기 위해 근사값을 사용하거나 확률적인 개념을 도입하는데 이러한 기법을 검사용이성 분석(Testability Analysis)이라 한다. 이를 회로의 설계 단계에서부터 적용하여 결함 시뮬레이션에 들어가는 비용, 검사 신호 생성 및 검사 장비 비용 등에 대한 영향을 미리 분석할 수 있고, 또한 분석 자료를 기초로 회로를 경제적으로 검사하려 할 때, 예상되는 문제점을 초기에 진단 가능하다^[2].

검사용이성 분석 알고리즘으로 많이 사용되는 것으로서 COP^[3]과 STAFAN^[1]이 있다. COP은 검사용이성을 확률적으로 분석하고 실행 시간이 빠르다는 장점이 있는 반면 순차 회로에 적용할 수 없다. STAFAN은 일정 양의 검사 신호를 입력으로 하여 이에 대한 논리 시뮬레이션을 수행하고 이로 인한 결과를 통해 회로의 각 선들에 대한 정보를 통계적으로 구하여 검사용이성 분석을 한다. 즉 정확한 정보를 얻기 위해서는 충분한 양의 검사 신호를 필요로 하며, 이로 인해 논리 시뮬레이션 시간에 많은 시간이 소요되어 현재와 같은 크기가 큰 회로에 대해서는 적용하기 불가능한 반면, 순차 회로에 적용 가능하다. 이러한 문제가 발생하여 COP의 빠른 계산 방식과 순차 회로에 적용 가능한 STAFAN의 후진선(feedback line) 분석의 특성을 적용하고 초기화 불가능한 플립플롭을 분석하여 정확도를 향상시킨 ITEM^[4]이 제안된 바 있다.

결함 시험도 관점에서 ITEM의 성능을 [4]에서 평가하였으나 검사용이성이란 회로의 각 선의 검출율(제어율 및 관측율)이란 측면도 있다. 본 논문에서는 STAFAN과 ITEM을 이용하여 회로 내의 각 선들에 대한 검출율을 구하고 이 정보를 이용하여 회로 전체의 검사용이성을 구한다.

따라서 검사용이성을 향상시킬 수 있는 메모리 소자를 대상으로 부분 스캔을 수행하고, 결과로 발생하는 회로에 대한 결함 시험도와 부분 스캔 소요시간을 분석하여 두 분석 알고리즘을 각 선들에 대한 검사용이성의 정확성 관점에서 평가해보고자 한다.

II. 시험성 분석 기법(ITEM)의 개요

ITEM^[4]은 계산 속도는 빠르지만 순차 회로에 적용이 곤란한 COP과 속도는 느리지만 순차 회로를 해석할 수 있는 STAFAN을 서로 통합, 개선하였다. ITEM에서는 기본 게이트의 제어율과 관측율에 구하는데 있어서 COP의 계산 방식을 채택하였다.

1. ITEM에 추가된 제어율과 관측율의 정의 및 X-제어율

순차회로에 대한 해석을 위해서 플립플롭에 대한 제어율과 관측율 계산법을 추가하였다. 플립플롭은 입력 클럭과 입력단과 출력단으로 구성이 되며, 클럭에 상승 신호가 발생할 때, 입력단 값에 따라 출력단 값이 결정되고 또한 이 값이 관측된다. 그리고 입력단 값이 변경된 후 클럭에 상승 신호 중 최초 한 개가 출력단에서 관측된다. 그리하여 ITEM에서는 클럭에 상승 신호가 발생할 확률과 입력단 값이 변경될 확률을 이용하여 플립플롭에 대한 제어율과 관측율 값을 결정한다. 그리고 XOR 게이트에 대하여, 게이트의 입력 중에서 논리값 1의 개수에 의해 게이트의 출력단을 결정되는 것을 고려하여 제어율과 관측율 계산법을 제시하였다.

또한 각 게이트의 전달 지연과 후진선 등으로 인하여 회로 내에 존재할 수 있는 논리값 X에 대한 제어율을 새로 도입하여 회로의 각 신호선에 대한 논리값 X로 제어될 확률을 구한다.

2. ITEM의 반복 연산을 통한 후진선 해석 및 초기화 불가능한 플립플롭 고려

순차 회로에서는 후진선으로 인해 회로 내에 루프가 형성된다. ITEM에서는 회로내의 루프에 대한 해석을 위해 각 신호선을 가지선, 후진선, 전진선 그리고 횡단선 4가지 종류로 분류하고 이 정보를 이용하여 회로내의 루프를 구성하는 부분들을 반복적 연산을 통해 각 선들에 대한 제어율과 관측율 값을 결정하게 된다. 반복 횟수는 실험적으로 결정되며 반복 횟수를 충분히 크게 하면 모든 신호선들의 제어율과 관측율이 일정한 수치에 수렴하지만, 플립플롭은 클럭에 상승 신호가 있을 때 동작하기 때문에 반복 횟수를 제한해야 하는데 ITEM^[4]에서는 60을 제안하고 있다.

순차 회로에서는 후진선이나 재수렴 하는 선들과 초기화되지 않는 플립플롭으로 인해, 회로 내부에 논리

값 X가 계속 존재한다.

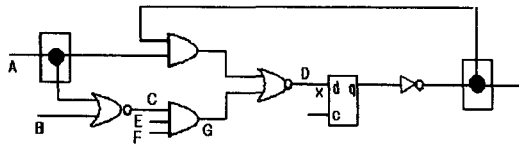


그림 1. 초기화되지 않는 플립플롭의 존재
Fig. 1. Existence of uninitializable flip-flop.

그림 1에서 A에 논리 1이 인가되면 플립플롭의 출력단은 논리값 X로 계속해서 유지가 된다. 이런 회로를 반복법으로 해석할 경우 큰 오차가 예상이 된다. 이러한 오차를 제거하기 위해 초기화되지 않는 플립플롭을 먼저 검색하고 해당 플립플롭 출력의 논리값 0과 1의 제어율을 각각 0으로 논리값 X에 대한 제어율을 1로 고정한다. 그리고 반복 횟수를 1로 고정하고 회로를 분석한다.

III. 부분 스캔을 이용한 성능 평가

STAFAN과 ITEM을 검사용이성 관점에서 평가하기 위해 본 논문에서는 Prashant S. Parish가 제안한 CoPS^[5]의 민감도 분석(sensitivity analysis)을 이용한 부분 스캔 기법을 이용한다.

1. 부분 스캔을 통한 성능 평가 이유

STAFAN과 ITEM을 비교하는 가장 이상적인 방법은 N개의 입력 단자를 가지는 회로에 대해서 입력에서 발생할 수 있는 모든 입력 신호 2^N 개를 인가하여 논리 시뮬레이션이나 결합 시뮬레이션을 수행하고 이에 따른 결과로서 각 신호선의 제어율과 관측율을 구해 각각의 알고리즘으로 측정한 결과와 비교 평가하는 것이다. 그러나 모든 입력 신호를 감안한다는 것은 현실적으로 불가능하다. 그리하여 본 논문에서는 간접적인 방법으로서 민감도 분석을 통한 검사용이성 부분 스캔 기법을 채택하였다.

2. 민감도 분석을 통한 부분 스캔 기법

민감도 분석이라 함은 회로내의 플립플롭을 스캔하고 이에 따른 회로의 변화에 대해서 전체 회로에 대한 검사용이성 영향을 분석하는 것이다. 그림 2의 알고리즘은 CoPS^[5]에서 제안한 부분 스캔 알고리즘이다. 먼저 회로의 모든 선에 대한 검출율을 구하고 이 값들을 모두 합하여 전체 회로의 검사용이성(TCF)을 구

한다. 인쪽의 루프는 한번에 하나의 플립플롭에 대해서 분석을 한다. Scan(j)는 플립플롭 j를 스캔 플립플롭으로 변환하는 것이고, 이는 플립플롭을 최종 입력단과 최종 출력단으로 변환하는 것과 같다. 이렇게 변환을 취한 다음, 수정된 회로에 대해 TCF_j를 구하고 임시로 스캔된 회로에 대한 민감도 ξ_j 를 구한 다음, 플립플롭을 다시 예전의 상태로 되돌려 놓는다(Unscan(j)). 이렇게 모든 플립플롭이 한번씩 스캔될 때까지 TCF를 구하고 여기서 민감도가 가장 큰 값을 선택하여 해당 플립플롭을 실제로 스캔하고 TCF를 선택한 플립플롭에 대한 값으로 재 설정하고 스캔 플립플롭의 집합에 포함을 시킨다. 그리고 위에서 행한 방식을 남은 플립플롭에 다시 적용하여 모든 플립플롭이 스캔될 때까지 반복 수행한다.

Partial Scan()

```

Compute Controllability and Observability
Compute TCF of the circuit
n = number of unscanned FFs
while(TCF > 0 and n > 0)
  for every unscanned FF j
    Scan(j)
    TCFj = new TCF
    Compute Sensitivity :  $\xi_j = TCF - TCF_j$ 
    Unscan(j)
  best = FF with maximum  $\xi_j$ 
  Scan(best)
  Add best to scan FF list
  n = n - 1
  Recompute best
  TCF = TCFbest
Return Scan FF list

```

그림 2. 부분 스캔 알고리즘, CoPS

Fig. 2. Algorithm of Partial Scan, CoPS.

3. 성능 평가

실험의 대상 회로는 ISCAS89 benchmark 회로^[6]를 이용하였다. 주어진 회로에 대하여 STAFAN과 ITEM을 이용하여 회로의 검사용이성을 분석하고 이를 기초로 30%, 50% 그리고 70% 부분 스캔을 수행하여 각각의 알고리즘을 이용하여 나온 결과 회로에 대하여 결합 시뮬레이터 HOPE^[5]를 이용하여 결합 시험도를 구하고 부분 스캔하는데 소요된 시간을 비교하였으며 각각의 부분 스캔의 결과에 대해 식(1)과 식(2)을 이용하여 평균 제곱 오차와 검출된 결함의 평균을 구하였다.

$$\text{평균 제곱 오차} = \frac{\sqrt{\sum (FC_s - FC_i)^2}}{\text{전체 회로의 수}} (\%) \quad (1)$$

$$\text{검출된 결함시험도의 평균} = \frac{\sum DF_{\text{회로}i}}{\sum F_{\text{회로}i}} \times 100 (\%) \quad (2)$$

식(1)의 FCs와 FCi는 각각의 회로에 대한 STAFAN과 ITEM을 이용해서 부분 스캔한 회로에 대한 결함 시험도이며, 식(2)의 $DF_{\text{회로}i}$ 는 회로 i에 대한 검출된 결함의 총수이며 $F_{\text{회로}i}$ 는 회로 i에 존재하는 모든 결함의 수를 뜻한다. 그림 3은 실험의 전체 흐름도이다.

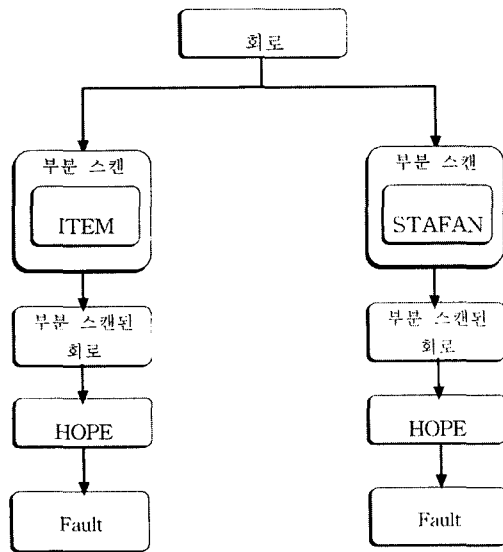


그림 3. 부분 스캔 흐름도

Fig. 3. Flowchart of Evaluated Partial Scan.

IV. 실험 결과

본 논문의 검사용이성 검출 알고리즘 평가 방법인 민감도 분석을 이용한 부분 스캔 알고리즘을 C언어로 구현하였으며 64MB의 주메모리를 갖는 Sun Ultra Sparc-1 workstation에서 실험이 수행되었다. 실험의 대상회로는 ISCAS89 benchmark회로^[6]를 사용하였다.

1. STAFAN의 입력 신호 수 결정

STAFAN은 입력된 신호에 대한 논리 시뮬레이션을 수행하면서 각 신호선에 대한 정보를 수집한다. 그리고 수집된 정보를 사용하여 필요한 확률 값들을 계산한다.

따라서 회로에 대한 검사용이성에 대한 측정을 하는데 있어서 입력되는 신호의 수에 의해 민감한 영향을 받으며 실행시간도 영향을 받는다.

이와 같은 입력되는 신호의 수를 결정하기 위하여, 본 논문에서는 ISCAS89 benchmark회로에 대하여 검사 신호수를 10, 18, 32, 56, 100, 178, 316, 562 그리고 1000으로 변경하면서 각각에 대한 검사용이성을 구하며 부분 스캔을 수행하였다.

표 1. 검사 신호수에 대한 s1423.bench회로의 부분 스캔 결과

Table 1. Partial scan result of s1423.bench for input vector number.

회로 (PI수/DF수)	Partial Scan 비율 (%) (DFF 수)	input vector 수	CPU time (sec)	Fault Coverage (%)
s1423 (17/74)	30 (22)	10	231.15	70.495
		18	359.567	69.835
		32	559.683	69.241
		56	778.783	70.825
		100	1362.983	70.957
		178	2416.683	72.409
		316	4251.467	74.059
		562	7562.433	73.993
		1000	13522.867	73.927
	50 (37)	10	360.833	74.983
		18	568.017	79.142
		32	862.383	80.528
		56	1238.3	80.066
		100	2184.55	81.320
		178	3884.15	80.198
		316	6815.417	80.528
		562	12098.933	80.264
		1000	21396.35	81.386
	70 (51)	10	559.5	90.297
		18	896.55	88.779
		32	1095.533	83.564
		56	1587.033	81.122
		100	2759.367	80.924
		178	6067.9	80.528
		316	8744.217	80.462
		562	15404.433	80.528
		1000	26888.483	80.462

표 1은 s1423.bench회로에 대하여 STAFAN을 이용하여 각각의 검사 신호수에 따른 부분 스캔 결과를 제시하였다. 결과에 나타난 바에 의하면, 검사 신호수에 의해서 결함 시험도가 진동함을 관찰할 수 있다. 특히 검사 신호수가 적을수록 결함 시험도의 편차가 크다. 그러나 검사 신호수가 178 이후부터는 편차가

작아지고 일정한 값에 수렴함을 실험 결과에서 나타내었다. 이러한 현상은 다른 회로에 대해서도 유사하게 나타났다. 그리하여 본 논문에서는 STAFAN을 이용해 검사용이성을 구하는데 필요한 적절한 검사 신호수를 실험 수행시간 및 결함 시험도의 편차 등을 고려하여 200으로 정하였다.

다음으로 ITEM과 STAFAN의 부분 스캔 성능을 평가하기 위하여, 본 논문은 각각의 알고리즘을 이용하여 30%, 50% 그리고 70% 부분 스캔을 수행함으로써 부분 스캔 결과로 구한 회로에 대한 결함 시험도와 부분 스캔에 소요된 시간을 비교함으로써 평가를 하였으며 그 결과는 4.2절부터 4.4절에서 제시하였다.

실험을 진행하는데 있어서, ITEM에서 제안하고 있는 바에 의해 정확한 검사 용이성 분석을 하기 위해 반복 횟수를 60으로 고정하여 실험에 적용하였으며, 또한 반복 횟수를 1로 고정하여 실험에 적용하여 각각의 차이를 비교하였다.

2. 30% 부분 스캔 결과

표 2와 표 3은 STAFAN과 ITEM을 이용하여 30% 부분 스캔하였을 때의 결과이다. 표 2에서 FC는 부분 스캔을 수행한 생성된 회로에 대한 결함 시험도를 나타내며 DF는 검출된 결함의 수를 뜻한다.

표 2. 30% 부분 스캔시 결함 시험도 결과

Table 2. Fault Coverage Result for 30% partial scan.

회로 이름	총결함수	STAFAN		ITEM			
		FC(%)	DF	반복횟수 = 1		반복횟수 =60	
		FC(%)	DF	FC(%)	DF	FC(%)	DF
s382.bench	399	88.471	353	76.942	307	83.709	334
s420.1.bench	455	23.077	105	26.813	122	26.813	122
s444.bench	474	66.878	317	81.646	389	66.878	317
s526.bench	555	68.649	381	70.450	391	68.649	381
s526n.bench	553	68.716	380	70.524	390	68.716	380
s641.bench	467	96.360	450	96.574	451	95.981	448
s713.bench	581	89.673	521	90.878	528	90.189	524
s953.bench	1079	97.127	1048	97.034	1047	96.200	1038
s1196.bench	1242	94.042	1168	94.042	1168	94.042	1168
s1238.bench	1355	89.299	1209	89.225	1209	89.225	1209
s1423.bench	1515	73.927	1120	64.356	975	72.409	1097
s5378.bench	4603	76.298	3512	70.280	3235	73.104	3365
s9234.bench	6923	50.079	3467	44.634	3090	58.212	4030
결함 총 수	20201		14031		13302		14413
검출 평균(%)			69.457		65.848		71.348

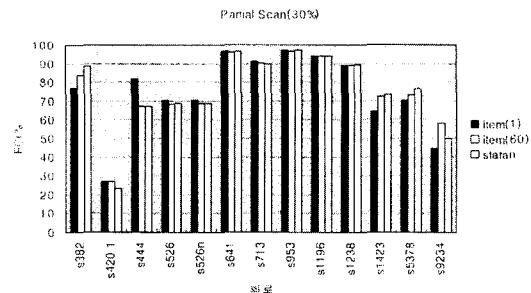


그림 4. 부분 스캔 30% 수행시 각 회로에 대한 결함 시험도 비교

Fig. 4. Fault Coverage Comparison for 30% partial scan.

표 2를 그림으로 나타내면 그림 4와 같다. 결과에 나타난 바에 의하면 ITEM알고리즘을 반복 횟수 1로 고정시키고 부분 스캔을 수행하였을 때, STAFAN 알고리즘을 이용한 부분 스캔 결과의 결함 시험도에 있어서 s382.bench와 s5378.bench회로에 대해서 다른 회로에 비해 낮은 결함 시험도를 보이는 반면 s444.bench회로에 대해서는 높은 결함 시험도를 나타낸다. 이 두 알고리즘을 이용한 결함 시험도 결과의 평균 제곱 오차는 1.765%로 오차를 나타낸다. 그리고 검출 평균은 65.848%로 STAFAN의 69.457%에 비해 약간 낮은 검출도를 나타낸다.

다시 ITEM알고리즘의 반복 횟수를 60으로 고정시킨 후 부분 스캔을 적용하였을 때, 결함 시험도에 평가에 있어서 s5378.bench 회로에 대해서 낮은 값을 보이고 s9234.bench 회로에 대해서는 높은 값을 보인다. 하지만 대체로 비슷한 결함 시험도를 갖는다. STAFAN 알고리즘을 이용해서 얻은 회로에 대한 결함 시험도의 평균 제곱 오차는 0.829%로서 대체적으로 비슷한 결함 시험도를 유지함을 나타낸다. 그리고 검출 평균은 71.348%로서 반복 횟수가 1일 때와 STAFAN을 이용하였을 때에 비해 높은 검출도를 보였다.

하지만 속도 면에 있어서는 STAFAN을 이용해서 부분 스캔하는 데 소요시간에 비해 반복 횟수를 1로 고정된 다음 ITEM을 이용한 부분 스캔을 수행하는데 소요되는 시간이 평균 98.191% 감소했으며, 반복 횟수를 60으로 고정된 ITEM을 이용한 부분 스캔을 수행하는데 소요되는 시간은 평균 55.678%가 감소하였다. 이는 STAFAN을 이용하였을 때, 입력 신호수를

200으로 하였기 때문에 논리 시뮬레이션 실행 시간이 ITEM을 반복 횟수를 1로 고정하였을 때에 비해 최대 200배의 시간이 소요되기 때문이다. 표 3은 각 알고리즘에 이용하여 30%부분 스캔을 수행하였을 때의 소요시간을 나타낸다. 표 3을 그림으로 나타내면 그림 5와 같다. 단, 그림 5의 종축의 단위는 실행시간에 대한 로그를 취한 값이다. 그리고 향상율이란 각 반복 횟수에 대한 ITEM을 이용하여 부분 스캔하였을 때 소요되는 시간이 STAFAN을 이용하여 부분 스캔을 수행하는데 소요되는 시간에 비해 단축 정도를 나타낸다.

3. 50% 부분 스캔 결과

표 4와 표 5는 STAFAN과 ITEM을 이용하여 50% 부분 스캔을 수행하였을 때의 결과 값이다. 표 4에 나타난 바에 의하면 반복 횟수를 1로 고정한 후 ITEM을 부분 스캔에 적용하였을 때와 STAFAN을 이용하였을 때 결과 값의 평균 제곱 오차는 2.936%이다. 그리고 검출 평균은 74.887%로 STAFAN의 75.040%과 비슷한 수치를 보인다.

표 3. 30% 부분 스캔시 실행 시간

Table 3. Execution Time for 30% partial scan.

회로 이름	STAFAN CPU(sec)	ITEM			
		반복 횟수 = 1		반복 횟수 = 60	
		CPU(sec)	감소율(%)	CPU(sec)	감소율(%)
s382.bench	47.450	0.967	97.962	29.317	38.215
s420.1.bench	33.333	0.683	97.951	0.683	97.951
s444.bench	48.117	1.117	97.678	34.417	28.472
s526.bench	53.867	1.217	97.741	41.616	22.743
s526n.bench	53.867	1.317	97.763	41.650	29.247
s641.bench	89.900	1.283	98.572	34.967	61.105
s713.bench	96.450	1.333	98.618	39.733	58.805
s953.bench	251.550	4.033	98.397	134.850	46.392
s1196.bench	137.917	1.983	98.562	2.383	98.272
s1238.bench	140.700	2.000	98.579	2.417	98.282
s1423.bench	2542.733	39.033	98.465	1334.133	47.531
s5378.bench	65129.667	1211.967	98.139	34208.983	47.475
s9234.bench	177130.000	3429.933	98.064	90228.7	49.060
평균감소율			98.192		55.657

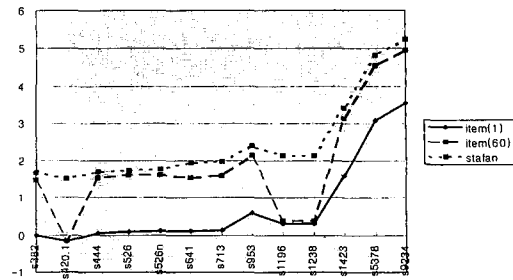


그림 5. 부분 스캔 30% 수행시 소요된 CPU 시간 비교

Fig. 5. Executed CPU time comparison as 30% partial scan.

표 4. 50% 부분 스캔시 결함 시험도 결과

Table 4. Fault Coverage Result for 50% partial scan.

회로 이름	총결 합수	STAFAN		ITEM			
				반복 횟수 = 1		반복 횟수 = 60	
		FC(%)	DF	FC(%)	DF	FC(%)	DF
s382.bench	399	96.241	384	97.494	389	89.223	356
s420.1.bench	455	43.397	197	43.297	197	43.297	197
s444.bench	474	77.637	368	94.937	450	69.831	331
s526.bench	555	67.748	376	90.270	501	67.748	376
s526n.bench	553	67.812	375	90.416	500	67.812	375
s641.bench	467	97.859	457	97.859	457	96.788	452
s713.bench	581	91.566	532	89.673	521	90.878	528
s953.bench	1079	97.127	1048	97.034	1047	96.200	1038
s1196.bench	1242	94.525	1154	94.525	1174	94.525	1174
s1238.bench	1355	89.520	1213	89.520	1213	89.520	1213
s1423.bench	1515	80.198	1215	78.683	1189	80.726	1223
s5378.bench	4603	77.536	3569	73.343	3376	78.145	3597
s9234.bench	6923	61.369	4251	59.391	4114	63.909	4427
결함 총수	20201		15159		15128		15287
검출 평균(%)			75.040		74.887		75.674

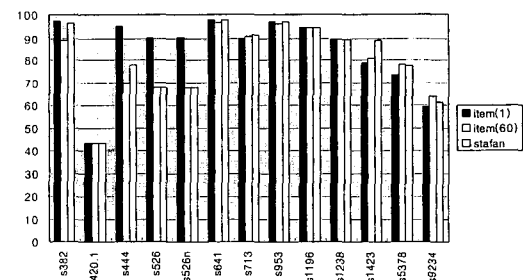


그림 6. 부분 스캔 50% 수행시 각 회로에 대한 결함 시험도 비교

Fig. 6. Fault Coverage Comparison as 50% partial scan.

그리고 반복 회수를 60으로 고정한 후 ITEM을 부분 스캔에 적용하였을 때의 경우 STAFAN을 적용하

였을 때, 두 결과 값의 평균 제곱 오차는 1.064%이며 검출 평균은 75.674%로 STAFAN보다 다소 높으나 비슷한 수치를 갖는다.

표 5는 50% 부분 스캔을 수행하였을 때 각각의 알고리즘에 대한 부분 스캔 실행 시간이다. 반복 횟수를 1로 고정하고 ITEM을 부분 스캔에 적용하였을 때 STAFAN을 적용하였을 때에 비해 98.358%의 실행 시간 감소를 보였으며, 반복 횟수를 60으로 하여 ITEM을 적용하였을 때는 58.838%의 감소를 보였다. 또한 30% 부분 스캔시 STAFAN를 기준으로 한 평균 실행 시간과 비교하여 감소율이 커짐을 관찰할 수 있다. 그림 7은 표 5를 그래프로 표현한 것이다.

표 5. 50% 부분 스캔 시 실행시간 결과
Table 5. Execution Time for 50% partial scan.

회로 이름	STAFAN CPU(sec)	ITEM			
		반복 횟수 = 1		반복 횟수 = 60	
		CPU(sec)	감소율(%)	CPU(sec)	감소율(%)
s382.bench	90.817	1.367	98.495	43.367	52.248
s420.1.bench	55.750	1.067	98.086	1.067	98.086
s444.bench	76.583	1.583	97.932	51.233	33.101
s526.bench	85.033	1.783	97.903	62.133	26.931
s526n.bench	95.033	1.783	98.124	62.017	34.742
s641.bench	139.217	1.900	98.635	55.783	59.931
s713.bench	160.900	2.017	98.747	64.050	60.193
s953.bench	411.400	5.983	98.546	211.260	48.649
s1196.bench	215.133	3.033	98.590	3.567	98.342
s1238.bench	212.833	3.167	98.512	3.633	98.293
s1423.bench	3931.683	57.683	98.535	1997.200	49.202
s5378.bench	99322.700	1753.967	98.234	49845.033	49.815
s9234.bench	304082.450	5127.367	98.314	135754.980	55.356
평균 감소율			98.358		58.838

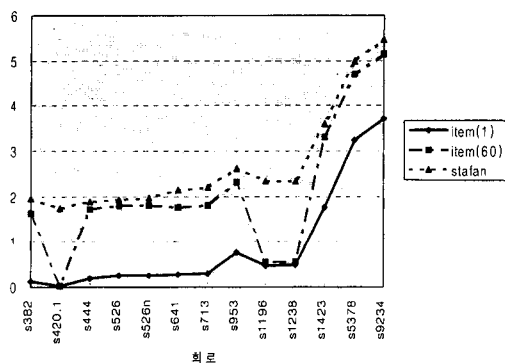


그림 7. 부분 스캔 50% 수행시 소요된 CPU 시간 비교
Fig. 7. Executed CPU time comparison as 50% partial scan.

4. 70% 부분 스캔 결과

표 6과 그림 8은 각 알고리즘을 적용하여 70% 부분 스캔을 수행하였을 때의 결과 값이다. 먼저 반복 횟수를 1로 고정하여 ITEM을 부분 스캔에 적용했을 때와 STAFAN를 적용하였을 때 두 결과 값의 평균 제곱 오차는 0.485%로서 거의 차이가 없음을 나타낸다. 그리고 검출 평균은 80.932%로서 80.244%인 STAFAN에 비해 약간 높은 수치를 보였다.

표 6. 70% 부분 스캔시 결함 시험도 결과
Table 6. Fault Coverage Result for 70% partial scan.

회로 이름	총결함수	STAFAN		ITEM			
		FC(%)	DF	반복 횟수 = 1		반복 횟수 = 60	
				FC(%)	DF	FC(%)	DF
s382.bench	399	99.749	398	100.000	399	99.248	396
s420.1.bench	455	54.725	249	54.725	249	54.725	249
s444.bench	474	96.835	459	97.046	460	69.831	331
s526.bench	555	93.333	518	96.577	536	92.793	515
s526n.bench	553	93.490	517	96.745	535	92.948	514
s641.bench	467	98.073	458	98.287	459	97.430	455
s713.bench	581	91.566	534	92.083	535	90.706	527
s953.bench	1079	96.756	1048	96.664	1047	96.849	1038
s1196.bench	1242	96.135	1194	96.135	1194	96.135	1194
s1238.bench	1355	91.292	1237	91.292	1237	91.292	1237
s1423.bench	1515	83.366	1263	85.941	1302	92.112	1244
s5378.bench	4603	73.452	3551	73.343	3376	78.225	3603
s9234.bench	6923	69.063	4784	72.47	5020	71.07	4923
결함 총수	20201		16210		16349		16226
검출 평균(%)			80.244		80.932		80.323

그러나 반복 횟수를 60으로 고정된 후 부분 스캔에 적용하였을 때 STAFAN을 적용하였을 때와 비교를 하면, 두 결과 값의 평균 제곱 오차는 2.222%이며 검출 평균은 80.323%로서 STAFAN과 비슷한 수치이다.

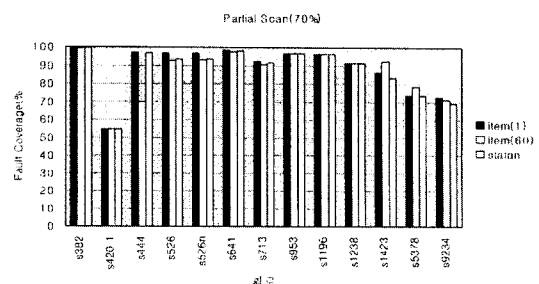


그림 8. 부분 스캔 70% 수행시 각 회로에 대한 결함 시험도 비교
Fig. 8. Fault Coverage Comparison as 70% partial scan.

표 7. 70% 부분 스캔 시 실행시간 결과
Table 7. Execution Time for 70% partial scan.

회로 이름	STAFAN CPU(sec)	ITEM			
		반복 횟수 = 1		반복 횟수 = 60	
		CPU(sec)	감소율	CPU(sec)	감소율
s382.bench	90.883	1.65	98.185	53.350	41.298
s420.1.bench	68.700	1.583	97.696	53.350	97.596
s444.bench	98.717	1.867	98.109	62.750	36.434
s526.bench	109.233	2.200	97.986	76.600	29.875
s526n.bench	110.050	2.233	97.971	76.500	30.486
s641.bench	181.900	2.417	98.671	69.700	61.682
s713.bench	189.533	2.600	98.628	79.267	58.178
s953.bench	467.050	7.350	98.426	261.510	44.008
s1196.bench	290.067	3.517	98.788	4.267	98.529
s1238.bench	259.033	3.767	98.546	4.300	98.340
s1423.bench	5029.500	69.733	98.614	2381.267	52.654
s5378.bench	124298.970	1756.200	98.587	60165.417	51.596
s9234.bench	357314.200	6216.033	98.260	169232.3	52.638
평균 감소율			98.344		57.955

표 7은 70% 부분 스캔을 수행하였을 때의 실행 시간이다. 속도에 있어서는 반복 횟수를 1로 고정하고 ITEM을 부분 스캔에 적용하였을 때 평균 98.344%의 감소를 보였으며 반복 횟수를 60으로 하여 적용하였을 때는 속도는 평균 57.955% 감소하였다. 그림 9는 70% 부분 스캔을 수행하였을 때의 시간을 그래프로 나타내었다.

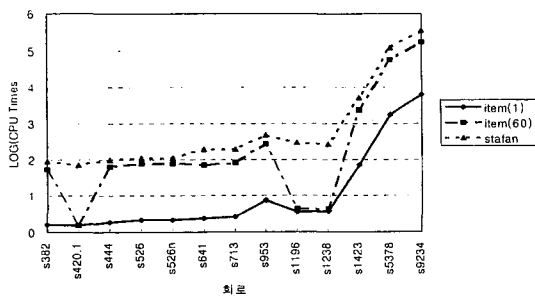


그림 9. 부분 스캔 50% 수행시 소요된 CPU 시간 비교

Fig. 9. Executed CPU time comparison for 50% partial scan.

각 부분 스캔의 결함 시험도를 비교해보면 ITEM 알고리즘을 적용하였을 때, 반복횟수가 1일 때와 60일

때에 결함 시험도의 약간의 차이가 발생함을 알 수 있다. 그림 10은 반복횟수에 의한 결함 시험도의 차를 나타내었다.

그림 10은 반복 횟수가 60일 때 부분 스캔을 수행했을 때 결함 시험도와 반복 횟수를 1로 하였을 때 결함 시험도의 차에 대한 그림이다. Y축의 0을 기준으로 아래쪽에 위치할수록 반복 횟수가 1일 때 결함 시험도가 더 좋음을 나타내고 있고 위쪽일수록 반복 횟수가 60일 때 결함 시험도가 더 좋음을 나타내고 있다.

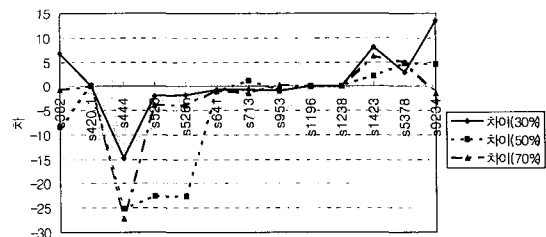


그림 10. 반복 횟수에 의한 결함 시험도 차
Fig. 10. Fault Coverage Difference by Iteration.

그림 10에서 보이는 바와 같이 s420.1.bench 회로를 제외한 s382.bench에서부터 s526n.bench 회로까지는 반복횟수가 1일 때 효과적인 부분 스캔을 하였고 s641.bench부터 s1196.bench 회로까지는 거의 차이가 없음을 나타내었지만 s1238.bench 회로 이후부터는 반복횟수가 60일 때 더 효과적인 부분 스캔을 하였음을 알 수 있다.

ITEM은 결함 시험도를 구하는데 있어서 초기화가 불가능한 플립플롭이 존재하는 회로를 제외하고는 반복횟수를 60으로 할 것을 제안하고 있다. 하지만 부분 스캔하는데 적용하는데 있어서는 반복 횟수를 1로 고정하고 수행하여도 60으로 고정하였을 때와 결함 시험도에 있어서 별 차이가 없음을 위의 수행한 결과에서 알 수 있다.

실험 결과 중 평균 제곱 오차, 평균 결함 검출률 및 누적 실행 시간을 표 8에 나타내었다. 표 8에서 두 번째 행은 STAFAN의 결함 시험도를 기준으로 하여 식(1)에 의한 결과 값이고, 세 번째 행은 STAFAN과 ITEM에 대한 식(2)의 결과 값이다. 그리고 네 번째 행은 전체 회로에 대한 CPU 실행 시간의 누적 시간이며, 괄호 안의 값은 STAFAN의 실행 시간을 기준으로 한 ITEM 실행시간의 감소율을 나타낸다.

표 8. 최종 실험 결과

Table 8. Experiment Result.

* STAFAN의 결함 시험도를 기준으로 한 값

** STAFAN의 실행 시간을 기준으로 한 값

비교 항목	부분 스캔율	STAFAN	ITEM	
			반복 횟수=1	반복 횟수=60
*평균 제곱 오차	30%		1.765	0.829
	50%		2.936	1.064
	70%		0.485	2.222
평균 결함 검출율	30%	69.479	65.848	71.348
	50%	75.040	74.887	75.674
	70%	80.244	80.932	80.323
누적 실행 시간 (*평균 감소율%)	30%	245760.551	4696.866 (98.192)	126133.849 (55.567)
	50%	408879.532	6962.600 (98.358)	188155.326 (58.838)
	70%	488507.833	8071.150 (98.344)	232468.833 (57.955)

표 8에 나온 결과 값에 의하면, 평균 제곱 오차는 3%이내 임을 알 수 있으며, 검출율은 STAFAN과 ITEM이 비슷함을 나타내고 있다. 하지만 실행 시간에 있어서 STAFAN에 비해 반복 횟수를 1로 고정 한 후 부분 스캔을 수행하였을 때는 98% 이상, 반복 횟수를 60으로 고정 한 후 부분 스캔을 수행하였을 때는 55%이상의 감소를 나타낸다.

본 논문의 실험 결과에 의하면, 순차 회로의 검사용 이성 측정에 있어서 통계적인 분석 방법인 STAFAN에 비해 확률적 분석 방법인 ITEM을 이용하는 것이 STAFAN과 비슷한 검사용이성을 유지하면서 빠른 시간 내에 분석할 수 있음을 알 수 있다. 특히 크기가 큰 순차 회로를 대상으로 효과적인 부분 스캔을 수행함을 나타내었다.

V. 결 론

본 연구에서는 통계적으로 검사용이성을 구하는 STAFAN과 확률적으로 검사용이성을 구하는 ITEM을 비교하였다. STAFAN은 회로의 검사용이성을 구하는데 소요되는 시간은 입력 신호의 수와 회로의 시뮬레이션 하는데 걸리는 시간의 곱으로서 회로가 증가함에 따라 실행시간이 커진다. 또한 통계적인 정확도를 유지하기 위해 검사 신호의 개수도 증가시켜야 한다. 즉 기술의 발전으로 인해 현재와 같은 초대형 회

로의 경우 적용하기 어려워짐을 알 수 있다. 이에 비해 기본적으로 ITEM은 COP의 검사용이성 측정 방법을 이용함으로 회로의 크기에 선형적으로 비례하여 실행시간이 증가한다. 하지만 ITEM에는 순차 회로를 해석하기 위해 포함된 후진선 검사에 필요한 시간과 초기화 불가능한 플립플롭을 찾는 알고리즘의 추가에 의해 이보다는 더 증가한다. 하지만 STAFAN과 달리 입력 신호수에 영향을 받지 않으므로 실행시간이 현저히 감소함을 알 수 있다.

특히 본 논문에서 실행한 실험 결과에 의하면 STAFAN을 이용하여 부분 스캔을 수행하였을 경우 논리 시뮬레이션 실행 시간이 회로의 크기에 민감하게 반응하여 부분 스캔 실행 시간이 기하급수적으로 증가함을 알 수 있었다. 하지만 ITEM은 단순히 시뮬레이션을 수행하는 것이 아니라 검사용이성을 구하기에 앞서 회로의 특성을 파악하고 이를 검사용이성 분석하는데 이용하므로 실행 시간이 현저하게 줄어들며 정확한 검사용이성 분석을 수행한다. 특히 실행 시간에 있어서 STAFAN에 비해 55%~98%의 시간을 단축할 수 있었으며 크기가 큰 회로에 있어서는 ITEM을 이용하였을 때 STAFAN에 비해 부분 스캔에 더 효과적임을 알 수 있다.

이제까지의 결과 값으로 볼 때 ITEM이 STAFAN에 비해 결함 시험도 측면뿐만 아니라 제어율과 관측율 관점에서도 우수함을 알 수 있었으며 특히 ITEM을 큰 회로에 적용하였을 때 더 효율적이라는 것을 알 수 있었다. 따라서 이 알고리즘을 설계 시부터 적용한다면 회로의 전체적인 정보를 얻을 수 있을 뿐만 아니라 검사비용의 절감에도 많은 도움이 있을 것으로 판단된다.

참 고 문 헌

- [1] S. K. Jain and V. D. Agrawal, "Statistical Fault Analysis", IEEE Design & Test of Computers, vol. 2, pp. 38-44, February 1985.
- [2] Miron Abramovici, Melvin A. Breuer and Arshur D. Friedman, *Digital Systems Testing And Testable Design*, IEEE press, pp. 343-346, 1990.
- [3] F. Brglez, "On Testability Analysis of

- Combinational Networks," Proceedings of the International Symposium on Circuits and Systems, pp. 221-225, May 1984.
- [4] 민형복, 김신태, 이재훈, "플립플롭의 초기화 가능성을 고려한 디지털 회로에 대한 고장 검출물의 평가 기법", 대한전자 공학회 논문지, 제35권 C편 제4호, pp. 263-272, 1998. 4
- [5] Prashant S. Parikh and Miron Abramovici, "Testability-Based Partial Scan Analysis", Journal of Electronic Testing, pp. 61-70, 1995.
- [6] F. Brglez, D. Bryan, and K. Kozminski, "Combinational profiles of sequential benchmark circuits," Proceedings of International Symposium on Circuits and Systems, pp. 1929-1934, May 1989.

 저 자 소 개



金 炯 局(正會員)

1996년 성균관대학교 공과대학 전기공학과(공학사). 1998년 성균관대학교 공과대학 전기공학과(공학석사). 1998년 2월 ~ 현재 삼성전자 반도체 ME 개발 사업부. 관심분야는 VLSI CAD / Testing

李 宰 勳(正會員) 第 35卷 C編 第 4號 參照

閔 炯 福(正會員) 第 35卷 C編 第 4號 參照