

논문 2005-42SD-3-5

가변 길이의 다중 특성 다항식을 사용하는 LFSR을 이용한 새로운 Reseeding 방법

(A New Reseeding Methodology Using a Variable-Length Multiple-Polynomial LFSR)

양 명 훈*, 김 유 빈*, 이 용*, 박 현 태*, 강 성 호**

(Myung-Hoon Yang, Youbean Kim, Yong Lee, Hyuntae Park, and Sungho Kang)

요 약

본 논문에서는 가변 길이의 다중 특성 다항식을 사용하는 LFSR (MP-LFSR)을 사용한 새로운 reseeding 방법을 제안한다. 제안된 reseeding 방법에서 많은 수의 specified bits를 가진 test cube는 높은 차수의 다항식으로 인코딩되고 반면에 적은 수의 specified bit를 가진 test cube는 낮은 차수의 다항식으로 인코딩 된다. 따라서 각 test cube에서 specified bit의 숫자에 따라서 인코딩 되는 data의 크기를 최적으로 줄일 수 있다. 가변 길이의 MP-LFSR은 기존의 MP-LFSR를 간단히 수정하여 구현이 가능하고 인코딩 데이터에 1 비트만을 추가하여 여러 개의 특성 다항식을 제어할 수 있다. 면적이 큰 ISCAS'89 벤치 회로에 대한 실험 결과는 제안된 방법이 비교적 작은 크기의 하드웨어 오버헤드로써 이전의 방법들 보다 좋은 인코딩 효율을 보여준다.

Abstract

This paper proposes a new reseeding methodology using a variable-rank multiple-polynomial linear feedback shift register (MP-LFSR). In the proposed reseeding scheme, a test cube with large number of specified bits is encoded with a high-rank polynomial, while a test cube with a small number of specified bits is encoded with a low-rank polynomial. Therefore, according to the number of specified bits in each test cube, the size of the encoded data can be optimally reduced. A variable-rank MP-LFSR can be implemented with a slight modification of a conventional MP-LFSR and Multiple Polynomial can be represented by adding just 1 bit to encoding data. The experimental results on the largest ISCAS'89 benchmark circuits show that the proposed methodology can provide much better encoding efficiency than the previous methods with adequate hardware overhead.

Keywords : test, Linear feedback shift register (LFSR) reseeding, encoding efficiency

I. 서 론

최근에는 고도의 미세 공정 기술 개발로 인하여 하나의 반도체 칩 내에 프로세서 코어, 큰 용량의 내장된 메모리, 아날로그 코어, 그리고 RF 코어와 같은 여러 개의 IP(intellectual property) 코어를 내장하여 시스템을

구성할 수 있는 시스템온칩(SoC: System-on-a-chip)이 가능하게 되었다. IP 코어가 점점 복잡해지고 또한 하나의 칩에 내장되는 IP 코어의 숫자가 증가함에 따라서 SoC를 테스트하기 위한 테스트 데이터의 양은 기하급수적으로 증가하고 있다^[1]. 그러나 입출력 채널의 용량과 속도, 그리고 ATE(automatic test equipment)의 데이터 메모리 용량의 제한으로 인하여 많은 양의 테스트 데이터를 SoC에 인가하는 것은 매우 어려운 일이다. 내장된 자체 테스트 기법(BIST: Built-In Self Test)은 이러한 문제의 해결을 위한 하나의 좋은 해결책이 될 수 있을 것이다^{[2][3][4][5]}. 그러나 IP 코어를 제공하는 업체로

* 학생회원, ** 평생회원 연세대학교 전기전자공학과
(Department of Electrical and Electronic
Engineering, Yonsei University)

※ 본 논문은 정보통신부의 출연금으로 수행한 IT
SoC 핵심설계인력양성 사업의 수행결과입니다.
접수일자: 2005년1월24일, 수정완료일: 2005년2월24일

부터 제공받는 IP 코어가 항상 BIST-ready 코어가 아니며 이러한 경우 BIST 회로의 내장을 위하여 IP 코어를 재설계해야 하는 문제가 생긴다. 테스트 디컴프레션 방법은 많은 양의 테스트 데이터를 줄이기 위한 하나의 방법이 될 수 있다. 결정 패턴의 test cube에 대한 테스트 디컴프레션에 대한 연구는 이전부터 많이 진행되어 오고 있다^{[6][7][8][9][10][11][12]}. 결정론적 패턴(Deterministic Pattern)의 test cube에 대한 테스트 디컴프레션은 적은 양의 인코딩된 데이터를 칩 외부로부터 받아서 칩 내부의 하드웨어에서 이 데이터를 많은 양의 테스트 데이터로 디코딩하여서 주사 경로를 통해서 테스트 대상 회로(CUT: Circuit under Test)에 인가하는 방법이다. LFSR reseeding을 사용한 최초의 테스트 디컴프레션 방법은 Koenemann에 의해서 제안되었다^[7]. LFSR에 seed가 로드되면 LFSR은 주사 경로로 결정론적 테스트 패턴을 인가할 수 있게 된다. 각 주사 경로 가운데 가장 긴 주사 경로의 길이를 L이라고 하면 주사 경로에 테스트 패턴을 채우기 위해서 L 클럭 주기만큼 쉬프트 동작을 수행해야 한다. 각각의 LFSR의 seed는 서로 다른 테스트 패턴을 생성하게 되기 때문에 ATPG (automatic test pattern generation)에 의해서 생성되는 결정론적 테스트 패턴을 생성할 수 있는 LFSR의 seed를 결정하기 위해서는 LFSR의 특성 다항식에 기반을 둔 선형 방정식을 풀어야 한다.

LFSR의 길이는 기본적으로 S_{max} 에 의해서 결정된다. S_{max} 는 각 테스트 패턴의 specified bit의 숫자 가운데 가장 큰 값을 나타낸다. Koenemann에 따르면 선형 방정식의 해가 존재하지 않을 확률이 10^{-6} 이하가 되기 위해서는 LFSR의 길이가 $S_{max} + 20$ 이 되어야 함을 수학적으로 증명하였다^[7]. 또한 Hellebrand는 MP-LFSR을 사용하는 경우 선형 방정식의 해가 존재하지 않을 확률이 10^{-6} 이하가 되기 위해서는 LFSR의 길이가 $S_{max} + 1$ 이 되어야 함을 보였다^[8]. 테스트 패턴의 집합에서 specified bit의 숫자의 변동이 심하기 때문에 테스트 패턴의 인코딩 효율성을 떨어뜨리게 되고 이러한 문제를 해결하기 위한 많은 방법들이 기존에 제안되었다^{[8][9][10][11][12]}.

다중 특성 다항식을 사용하는 LFSR을 이용하는 reseeding 방법이 [8]과 [9]에서 제안되었으며 [10]과 [11]에서는 seed의 저장을 위해서 할당하는 메모리를 최적화하기 위해서 가변 길이의 seed를 사용하는 reseeding 방법이 제안되었다. 그리고 [12]에서는 LFSR

의 특성 다항식의 차수보다 짧은 길이의 seed를 사용하여 테스트 데이터의 크기를 줄이는 reseeding 방법이 소개되었다.

본 논문에서는 가변 길이의 다중 특성 다항식을 이용하는 LFSR을 사용하는 새로운 reseeding 방법을 제안한다. 제안하는 reseeding 방법은 기존의 다중 특성 다항식을 이용하는 LFSR을 간단히 수정하여서 구현이 가능하며 가능한 한 짧은 길이의 특성 다항식을 사용함으로써 인코딩 효율을 높일 수 있도록 해준다.

ISCAS'89 벤치 회로에 대한 실험 결과는 제안하는 reseeding 방법이 비교적 작은 하드웨어 오버헤드를 가지고 기존의 다른 reseeding 방법에 비해서 향상된 인코딩 효율성을 제공함을 보여준다.

II. 제안된 Reseeding 아키텍처

이전의 연구에서 제안된 LFSR에 기반을 둔 테스트 디컴프레션 방법들은 각 test cube를 고정된 길이의 LFSR에 대한 seed로 인코딩을 하게 된다. 따라서 이러한 방법들의 인코딩 효율성은 S_{max} 에 의해서 제한된다. 만약 많은 수의 specified bit를 가진 test cube는 차수가 높은 다항식으로 인코딩되고 적은 수의 specified bit를 가진 test cube는 낮은 차수의 다항식으로 인코딩된다면 인코딩 효율을 상당히 개선할 수 있을 것이다. 이러한 점을 고려하여 테스트 디컴프레션 방법을 구현하기 위해서 서로 다른 차수의 다항식을 MP-LFSR에서 사용할 수 있도록 하는 새로운 reseeding 방법을 제안한다.

그림 1은 가변 길이의 MP-LFSR의 구조를 나타내는 블록 다이어그램이다. 테스트 데이터 가운데서 특성 다항식의 차수와 계수에 관한 정보를 받아서 Polynomial

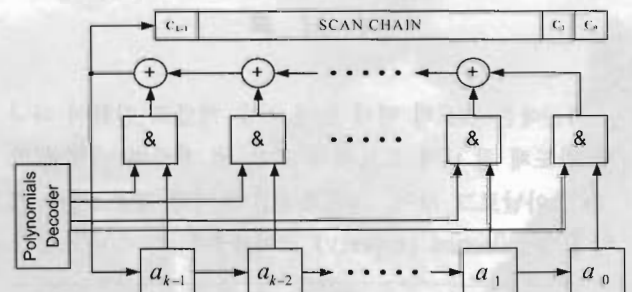


그림 1. 가변 길이의 특성 다항식을 이용한 LFSR의 블록 다이어그램
Fig. 1. Block Diagram of a Variable-Length Multiple Polynomial LFSR.

Decoder 블록에서 그 데이터를 디코딩하여 되먹임 경로에 있는 XOR 게이트에 연결된 AND 게이트의 입력단의 값을 설정하여서 여러 가지 차수와 계수를 가지는 LFSR을 구현할 수 있다. 즉 Polynomial Decoder 출력 가운데 1을 가지는 값이 어느 위치부터 시작되는지에 따라서 특성 다항식의 차수가 결정되고 그 이후부터의 값에 따라서 특성 다항식의 각 항의 계수가 결정된다. 그림에서 알 수 있듯이 가변 길이의 특성 다항식을 사용하는 MP-LFSR은 기존의 MP-LFSR과 거의 유사한 구조를 가진다.

그림 2는 제안하는 reseeding 방법에 대한 블록 다이어그램을 나타내고 있다. 본 논문에서 제안하는 아키텍처는 MP-LFSR, 특성 다항식 디코더, 테스트 제어기, rank 카운터, poly 카운터, 패턴 카운터, 비트 카운터로 구성된다. 각 특성 다항식은 seed의 길이, 즉 특성 다항식의 길이별로 인코딩된다. 각 seed 데이터는 특성 다항식을 나타내기 위한 "next_poly" 비트들과 같이 저장된다. "next_poly" 비트는 특성 다항식의 계수나 차수가 변하는지 아닌지의 여부를 나타내게 된다. Test Controller는 "next_poly" 비트를 로드해야 할 경우가 되면 Update_poly_count 신호를 활성화 시켜서 NB(Next Bit) 레지스터에 저장된 "next_poly" 비트의 값을 poly_count_enable 신호를 통해서 polynomial counter 블록으로 전달하게 된다. Polynomial counter

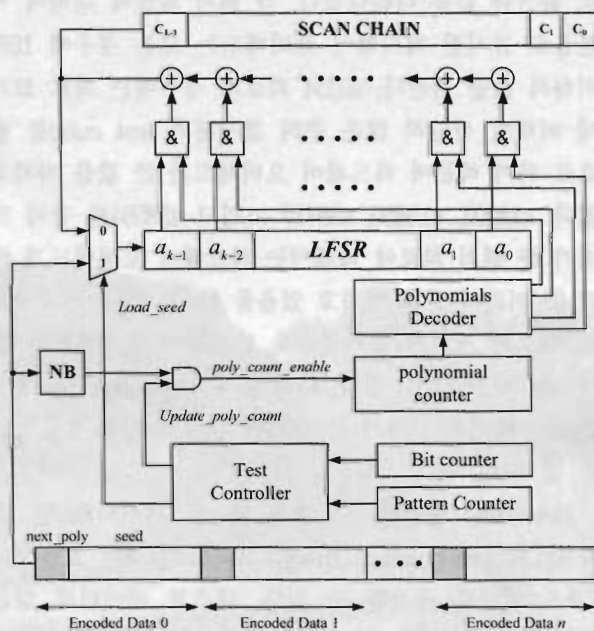


그림 2. 제안된 Reseeding 방법을 구현하기 위한 하드웨어 아키텍처

Fig. 2. Hardware Architecture for implementation of Proposed Reseeding.

블록은 poly_count_enable 신호가 1일 때마다 1씩 카운터를 증가시키게 되고 이 값에 의해서 Polynomial decoder는 현재 사용할 특성 다항식을 결정하게 된다.

다른 차수의 특성 다항식을 사용한다는 것은 가변 길이의 seed를 사용할 수 있다는 의미이다. 즉 각 seed를 LFSR로 로드하기 위한 클럭의 수가 각 seed의 길이에 따라서 달라진다. LFSR에 새로운 seed를 채우기 위한 클럭의 수는 특성 다항식의 차수에 의존하게 됨은 명백하다. 따라서 새로운 seed를 LFSR에 채우기 위해서는 특성 다항식의 차수만큼의 쉬프트 동작이 필요하게 된다. 이 때 필요한 쉬프트 동작의 횟수는 특성 다항식의 길이에 따라서 달라지게 된다. Test Controller 블록은 비트 카운터의 값이 필요한 쉬프트 동작의 횟수와 같아질 때까지 쉬프트 동작을 하도록 제어하게 된다.

III. Seed 인코딩 프로세스

그림 3은 앞에서 설명한 reseeding을 위한 테스트 데이터 인코딩 프로세스의 과정을 나타내고 있다. 검출이 쉬운 고장(easy-to-detect fault)의 경우에는 의사임의 패턴(pseudorandom pattern)을 적용하여서 검출하게 된다. 그리고 주어진 결정론적 패턴에 대해서는 최초의 LFSR의 길이는 $s - 10$ 으로 선택되고 이 값이 그림 3에서의 polynomial_rank의 값으로 선택된다. 여기서 s 는 주어진 test cube에서의 specified bit의 숫자이다. 주어진 길이의 특성 다항식을 가지는 LFSR이 test cube와 일치하는 패턴을 생성하기 위한 LFSR의 seed 값을 구하기 위한 선형 방정식이 구성되고 이 선형 방정식의 해가 LFSR의 seed값이 된다. 만약 주어진 선형 방정식

```

pseudo_random_testgen();
deterministic_testgen();
for (all test cubes){
  s = number_of_specified_bits(current_test_cube)
  polynomial_rank = s - 10
  while (SEED_FOUND == FALSE) {
    for (i=0; i<16 | SEED_FOUND == FALSE; i++) {
      select_one_polynomial(polynomial_rank);
      form_a_system_of_equations();
      solve_a_system_of_equation();
    }
    if (SEED_FOUND == FALSE) then
      polynomial_rank++;
  }
}
    
```

그림 3. 제안된 인코딩 프로세스

Fig. 3. Proposed encoding process.

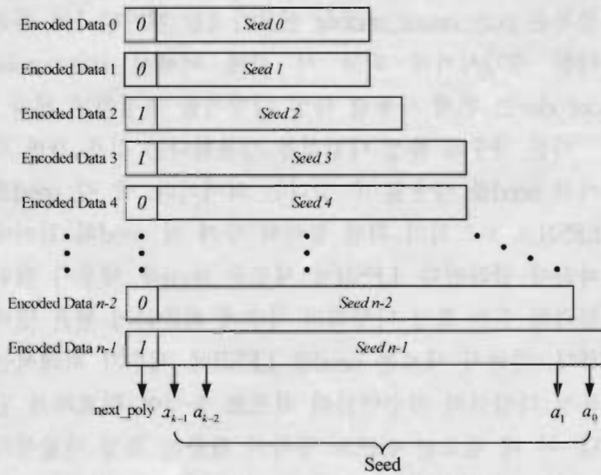


그림 4. 제안된 Reseeding을 이용한 인코딩 데이터
Fig. 4. Encoded Test Data using Proposed Reseeding.

의 해가 존재하지 않으면 선형 방정식의 해가 존재할 때까지 특성 다항식을 변경하면서 이 과정을 반복하게 된다. 본 논문에서는 각 길이별로 16개의 다중 특성 다항식을 사용한다. 만약 16개의 특성 다항식에 대해서 모두 선형 방정식의 해를 찾을 수 없는 경우에는 *polynomial_rank*를 1만큼 증가시킨 후에 위의 과정을 반복하게 된다. 이와 같은 과정을 거치게 되면 각 test cube에 대해서 가능한 가장 짧은 길이의 특성 다항식을 가지는 LFSR을 사용하여 테스트 패턴을 생성할 수 있게 되므로 LFSR의 seed 값으로 encoding되는 테스트 데이터의 길이를 최소화 할 수 있게 된다.

그림 4는 그림 3에 나타난 인코딩 프로세스에 의해서 생성된 인코딩 데이터의 예를 보여주고 있다. 각 인코딩 데이터는 "next_poly" 비트와 Seed 값으로 구성된다. "next_poly" 비트가 1인 경우에는 특성 다항식의 차수나 계수가 변하는 것을 의미하게 된다. 이 값이 1인 경우마다 앞에서 설명한 바와 같이 Polynomial Counter의 값이 증가하게 되고 Polynomial Counter의 값에 따라서 Polynomial Decoder 블록에서 디코딩을 수행하여 현재 Seed가 사용될 LFSR의 특성 다항식을 결정하게 된다.

IV. 실험 결과

앞에서 설명한 MP-LFSR의 reseeding 방법에 대한 실험은 ISCAS'89 벤치 회로들 가운데 크기가 비교적 큰 회로에 대해서 수행하였다. 각 회로에 대해서 쉽게 검출할 수 있는 고장에 대해서는 10K개의 의사임의 패턴을 적용하여 테스트를 수행하고 100%의 고장 검출을

표 1. 제안된 방법의 하드웨어 오버헤드
Table 1. Hardware Overhead of Proposed Scheme.

회로 이름	게이트 수	하드웨어 오버헤드	
		(게이트 수)	백분율 오버헤드
s5378	4271	447	10.47
s9234	8579	1273	14.84
s13207	14260	727	5.10
s15850	16280	963	5.92
s38471	38011	2006	5.28
s38584	37554	1160	3.10

을 얻기 위해서 의사임의 패턴으로 검출하지 못하고 남아있는 임의패턴 저항고장(random pattern resistance fault)들에 대해서 결정론적 테스트 패턴을 생성하기 위해서 ATPG를 수행하였다. 각 test cube들은 대응하는 선형 방정식을 통해서 seed로 인코딩되었다.

표 1은 제안된 reseeding 방법을 사용한 경우의 하드웨어 오버헤드를 나타내고 있다. 하드웨어 오버헤드는 2 입력 NAND 게이트의 크기를 하나의 게이트로 가정했을 경우의 게이트 숫자로 나타낸 값이다. 표 1에서 두 번째 열은 ISCAS'89 벤치 회로의 게이트 수를 나타내고 있으며 세 번째 열은 제안된 reseeding 아키텍처의 게이트 수를 나타내고 있다.

제안된 reseeding 방법에 대한 하드웨어 아키텍처는 Synopsys사의 Design Compiler 툴을 이용하여 합성되었다. 표의 마지막 열에는 하드웨어 오버헤드를 백분율로 환산한 값을 나타내었다. 각 벤치 회로에 대하여 백분율로 표시한 하드웨어 오버헤드는 모든 경우에 15% 이하의 값을 가진다. s9234 회로의 경우에는 회로 크기에 비해서 상당히 많은 수의 결정론적 test cube를 필요로 하기 때문에 하드웨어 오버헤드가 큰 값을 가지고 있다. s13207, s15850, s38417 그리고 s38584와 같이 크기가 큰 벤치 회로에 대해서는 하드웨어 오버헤드가 대략 6%이하의 값을 가지고 있음을 알 수 있다. 전체적으로 회로의 크기가 증가할수록 하드웨어 오버헤드가 줄어드는 것을 알 수 있다. 따라서 제안된 reseeding 방법은 하드웨어의 크기가 큰 회로에 대해서 더욱 효율적임을 알 수 있다.

Reseeding 방법의 효율성은 테스트 데이터의 양, 압축률(CR: compression ratio), 그리고 인코딩 효율성의 3가지 인자로 평가할 수 있다. 테스트 데이터의 양은 인코딩된 데이터를 저장하기 위한 메모리의 크기이다. 인코딩 데이터의 크기는 ATPG에서 생성된 test cube set의 크기에 큰 영향을 받기 때문에 다양한 reseeding 방법을 평가하기에는 충분하지 않다. CR은 결정론적

표 2. 기존의 reseeding 방법[8]과 제안된 reseeding 방법의 비교

Table 2. Comparison of previous scheme [8] and proposed reseeding scheme.

회로		본 논문		[8]	① ÷ ②
이름	주사 셀의 개수	테스트 데이터 ①	테스트 데이터 ②	테스트 데이터 ②	
s5378	214	362	726	726	0.50
s9234	247	5,162	6,923	6,923	0.75
s13207	700	1,333	3,570	3,570	0.37
s15850	611	2,870	6,528	6,528	0.44
s38471	1664	21,058	24,283	24,283	0.87
s38584	1464	2,238	3,406	3,406	0.66

테스트 패턴 전체를 저장하기 위해서 필요한 저장 용량을 인코딩된 테스트 데이터의 양으로 나눈 값이다. CR 값은 reseeding 방법을 평가하는데 있어서 단순히 테스트 데이터의 양에 비해서는 좋은 방법이라고 할 수 있다. 그러나 대부분의 LFSR에 기반을 둔 reseeding 방법은 test cube에서의 specified bit의 숫자에 따라서 성능의 차이를 보이게 된다. 특히 specified bit의 숫자가 적은 경우에는 CR 값은 reseeding 방법의 효율성에 무관하게 높아질 수 있다. LFSR에 기반을 둔 reseeding 방법을 평가하기에 가장 좋은 인자는 인코딩 효율(encoding efficiency)이다. 인코딩 효율은 test cube의 전체 specified bit의 숫자를 인코딩된 데이터의 전체 비트 수로 나누어서 계산하게 된다^[9]. 제안된 reseeding 방법의 실험 결과는 MP-LFSR을 이용한 방법인 [8]과 [9], 가변 길이의 seed를 이용하는 방법인 [11], 그리고 partial reseeding을 이용하는 [12]와 비교하였다. 여기서 test cube 데이터와 생성된 seed 데이터와 효율을 비교하기 위해서 인코딩 데이터에서 "next_poly" 비트는 제외하였다.

표 2는 제안된 reseeding 방법과 [8]에서 사용한 MP-LFSR reseeding 방법의 비교 결과를 나타내고 있다. [8]에서의 실험 결과가 테스트 데이터양에 대해서만 나타내어서 이를 기준으로 제안된 방법과 비교하고 있다. 두 가지 경우 모두 10K개의 의사임의 패턴을 인가한 후에 남은 고장에 대해서 생성된 결정론적 테스트 패턴에 대한 reseeding 결과이다. 표에서 나타난 바와 같이 제안된 방법이 모든 벤치 회로에 대해서 [8]의 결과에 비해서 13% - 63% 정도 적은 테스트 데이터를 필요로 함을 알 수 있다.

표 3은 제안된 reseeding 방법과 [9]에서 사용한

표 3. 기존의 reseeding 방법[9]와 제안된 reseeding 방법의 비교

Table 3. Comparison of previous scheme [9] and proposed reseeding scheme.

회로		본 논문		[9]	① ÷ ②
이름	인코딩 효율 ①	인코딩 효율 ②	인코딩 효율 ②	인코딩 효율 ②	
s5378	1.23	0.90	0.90	0.90	1.37
s9234	1.15	0.97	0.97	0.97	1.19
s13207	1.17	0.53	0.53	0.53	2.21
s15850	1.14	NA	NA	NA	NA
s38471	1.12	0.94(100K)	0.94(100K)	0.94(100K)	1.19
s38584	1.18	0.94(100K)	0.94(100K)	0.94(100K)	1.26

표 4. 가변 길이 reseeding 방법 [11]과 제안된 reseeding 방법의 비교

Table 4. Comparison of variable length scheme[11] and proposed reseeding scheme.

회로		본 논문		[11]		① ÷ ②
이름	테스트 데이터	CR ①	테스트 데이터	CR ②	테스트 데이터	
s5378	362	16.0	NA	NA	NA	NA
s9234	5,162	10.0	5,346	4.8	5,346	2.08
s13207	1,333	51.5	5,877	20.96	5,877	2.46
s15850	2,870	29.6	6,316	5.42	6,316	5.46
s38471	21,058	50.3	16,796	7.72	16,796	6.52
s38584	2,238	107.3	3,996	19.05	3,996	5.63

MP-LFSR reseeding 방법의 비교 결과를 인코딩 효율 측면에서 나타내고 있다. 여기서도 앞에서와 마찬가지로 10K개의 의사임의 패턴을 인가한 후에 검출되지 않은 고장에 대해서 생성된 결정론적 테스트 패턴에 대한 reseeding 결과이다. 다만 [9]에서 s38417과 s38584의 경우에는 100K개의 의사임의 패턴을 인가한 후의 결과만이 주어지고 있다. 제안된 reseeding 방법이 인코딩 효율 측면에서 19% - 121% 정도 [9]의 결과보다 향상되었음을 알 수 있다.

표 4는 제안된 reseeding 방법과 가변 길이의 reseeding 방법^[11]의 CR 값의 비교 결과를 보여 주고 있다. 여기서도 결과는 역시 10K개의 의사임의 패턴을 인가한 후의 결과이다. 표에서 보면 s38417을 제외한 모든 회로에서 제안된 방법이 더 적은 테스트 데이터를 필요로 함을 알 수 있다. s38417 회로의 경우에는 제안된 방법이 더 많은 테스트 데이터를 필요로 하지만 더 큰 CR 값을 가지고 있다. 이것은 제안된 방법의 test

표 5. partial reseeding 방법 [12]와 제안된 reseeding 방법의 비교

Table 5. Comparison of partial reseeding scheme[12] and proposed reseeding scheme.

회로	본 논문					[12]					① ÷ ②	
	이름	Test cube 개수	Specified bit 개수	Smax	테스트 데이터	인코딩 효율 ①	Test cube 개수	Specified bit 개수	Smax	테스트 데이터		인코딩 효율 ②
s5378		27	444	19	362	1.23	30	493	18	502	0.982	1.25
s9234		208	5,921	52	5,162	1.15	138	4,674	61	5,013	0.932	1.23
s13207		100	1,563	30	1,333	1.17	157	2,824	24	3,008	0.938	1.25
s15850		139	3,265	37	2,870	1.14	167	5,092	38	5,204	0.978	1.17
s38471		637	23,524	92	21,058	1.12	340	23,984	85	24,513	0.978	1.15
s38584		164	2,649	54	2,238	1.18	62	2,848	55	2,942	0.968	1.22

set의 크기가 [11]보다 더 크다는 것이다. 제안된 방법의 CR 값이 더 작기 때문에, 만약 s38417 회로에 대해서 같은 test cube에 대해서 인코딩을 수행한다면 제안된 방법이 더 작은 크기의 인코딩 데이터를 생성하게 될 것이다. 모든 벤치 회로에 대해서 제안된 방법이 108% - 552% 정도로 우수한 CR 값을 나타냄을 알 수 있다.

표 5에서는 10K개의 의사임의 패턴 인가 후에 제안된 reseeding 방법과 partial reseeding 방법^[12]의 결과의 비교를 나타내고 있다. 제안된 방법의 인코딩된 테스트 데이터가 s9234를 제외한 모든 벤치 회로에 대해서 partial reseeding 방법의 경우보다 적음을 알 수 있다. s9234의 경우에는 테스트 데이터가 partial reseeding의 경우보다 약간 많은데 이것은 제안된 방법에서 사용된 인코딩되기 이전의 데이터에서 specified bit의 숫자가 partial reseeding보다 많기 때문이다. 즉 s9234 회로의 경우에는 partial reseeding 방법이 적은 수의 test set에 대해서 인코딩을 수행하였기 때문에 테스트 데이터의 크기가 적어진 것이다. 그러나 제안된 방법은 s9234의 경우 인코딩 효율 면에서는 partial reseeding의 경우보다 23%정도 향상됨을 알 수 있다. 따라서 같은 test set에 대해서 인코딩을 수행할 경우에는 제안된 방법이 더 적은 테스트 데이터를 생성할 것이다. 표에서 알 수 있듯이 모든 경우에 대해서 제안된 reseeding 방법이 partial reseeding보다 15%-25% 정도 향상된 인코딩 효율을 보이고 있다.

IV. 결 론

본 논문에서는 가변 길이의 다중 특성 다항식을 사용하는 LFSR에 기반을 둔 새로운 reseeding 방법을 제안하였다. 제안된 방법은 많은 수의 specified bit을 가지는 test cube에 대해서는 높은 차수의 특성 다항식을 사용하고 적은 수의 specified bit을 가지는 test cube에 대해서는 낮은 차수의 특성 다항식을 사용하게 된다. 따라서 제안된 reseeding 방법은 specified bit의 최대값을 사용하는 기존의 방법에 비해서 test set에서 specified bit의 변화가 많더라도 각 test cube의 specified bit의 숫자에 대해서 가능한 낮은 차수의 특성 다항식을 적용할 수 있기 때문에 인코딩 효율이 높은 효율적인 테스트 데이터를 생성할 수 있다. 제안된 reseeding 방법에 대한 하드웨어 아키텍처는 기존의 MP-LFSR을 간단히 수정하여서 구현이 가능하며, 따라서 기존의 MP-LFSR을 이용하는 reseeding 방법보다 약간 증가된 하드웨어 오버헤드를 가지게 되며 회로가 커질수록 작은 크기의 하드웨어 오버헤드를 가지게 된다. ISCAS'89 벤치 회로에 대한 실험 결과는 제안된 reseeding 방법이 기존의 reseeding 방법보다 훨씬 향상된 인코딩 효율을 얻을 수 있음을 보여준다.

참 고 문 헌

- [1] A. Chandra and K. Chakrabarty, "System-on-a-chip test-data compression and decompression architecture based on Geology codes," *IEEE*

- Trans. Computer-Aided Design of Integrated Circuits and Systems*, Vol. 20, pp. 355-368, March 2001.
- [2] P. H. Bardell, W. McAnney, and J. Savir, *Built-in test for VLSI: Pseudo-Random Techniques*. New York: Wiley, 1987.
- [3] V. D. Agrawal, C. R. Kime, and K. K. Saluja, "A tutorial on built-in self-test - Part 1: Principles," *IEEE Design & Test of Computers*, Vol. 10, pp. 73-82, March 1993.
- [4] V. D. Agrawal, C. R. Kime, and K. K. Saluja, "A tutorial on built-in self-test - Part 2: Applications," *IEEE Design & Test of Computers*, Vol. 10, pp. 69-77, July 1993.
- [5] H. -J. Wunderlich and Y. Zorian, *Built-In Self Test (BIST): Synthesis of Self-Testable Systems*. Norwell, MA: Kluwer, 1997.
- [6] H. -S. Kim, J. -K. Lee, and S. Kang, "A new multiple weight set calculation algorithm," *Proc. Int. Test Conf.*, pp. 878-894, 2001.
- [7] B. Koenemann, "LFSR-coded test pattern for scan designs," *Proc. Eur. Test Conf.*, pp. 237-242, 1991.
- [8] S. Hellebrand, B. Reeb, S. Tarnick, and H. -J. Wunderlich, "Pattern generation for a deterministic BIST scheme," *Proc. Int. Computer-Aided Design Conf.*, pp. 88-94, 1995.
- [9] S. Hellebrand, J. Rajski, S. Tarnick, S. Venkataraman, and B. Courtois, "Built-in test for circuits with scan based on reseeding of multiple-polynomial linear feedback shift registers," *IEEE Trans. Computers*, Vol. 44, pp. 223-233, Feb. 1995.
- [10] N. Zacharia, J. Rajski, J. Tyszer, and J. A. Waicukauski, "Two-dimensional test decompressor for multiple scan design," *Proc. Int. Test Conf.*, pp. 186-194, 1996.
- [11] J. Rajski, J. Tyszer, and N. Zacharia, "Test data decompression for multiple scan designs with boundary scan," *IEEE Trans. Computers*, Vol. 47, pp. 1188-1200, Nov. 1998.
- [12] C. V. Krishna, A. Jas, and N. A. Touba, "Test vector encoding using partial LFSR reseeding," *Proc. Int. Test Conf.*, pp. 885-893, 2001.

저 자 소 개



양 명 훈(학생회원)
 1996년 연세대학교 전기공학과
 학사 졸업.
 1998년 연세대학교 전기공학과
 석사 졸업.
 2004년 삼성전자 System LSI
 사업부 선임연구원.

2005년 현재 연세대학교 전기전자공학과
 박사과정.

<주관심분야 : DFT, BIST, SoC 설계>



김 유 빈(학생회원)
 2002년 서강대학교 컴퓨터학과
 학사 졸업.
 2004년 서강대학교 컴퓨터학과
 석사 졸업.
 2005년 현재 연세대학교 전기전자
 공학과 박사 과정.

<주관심분야 : BIST, DFT, SoC Test>



이 용(학생회원)
 2003년 연세대학교 전기전자
 공학과 학사 졸업.
 2003년 연세대학교 아식설계 공동
 연구소 연구원
 2005년 현재 연세대학교 전기전자
 공학과 석사 과정.

<주관심분야 : DFT, BIST, Test Compression>



박 현 태(학생회원)
 2004년 연세대학교 전기전자
 공학과 학사 졸업.
 2005년 현재 연세대학교 전기전자
 공학과 석사 과정.

<주관심분야 : DFT, SoC 설계>



강 성 호(평생회원)
 1986년 서울대학교 제어계측
 공학과 학사 졸업.
 1988년 The University of Texas,
 Austin 전기 및 컴퓨터
 공학과 석사 졸업.
 1992년 The University of Texas,

Austin 전기 및 컴퓨터공학과 박사 졸업
 1992년 미국 Schlumberger Inc. 연구원
 1994년 Motorola Inc. 선임 연구원
 현재 연세대학교 전기전자공학과 교수

<주관심분야 : SoC 설계 및 SoC 테스트>