

Exclusive OR-AND-OR 배열을 이용하는 3단계 PLA 에서의 고장 검출

Fault Detection in the 3-Level PLA using Exclusive OR-AND-OR array

황 희 용
 진 영 문 *
 조 영 보

서울 대학교
 전자계산기 공학과

1. 서론

Programmable Logic Array (PLA) 는 조합 논리 회로의 기능을 경제적으로 실현시킬 수 있는 논리 소자로서, Fleisher 와 Maissel (1) 에 의하여 제안 되었으며 구체적인 배열 구조로 되어 있으므로 LSI 나 VLSI 설계에 그 응용도가 높다.

PLA 는 일반적으로 AND 배열 및 OR 배열로 구성되는 2단계 PLA 이나, 2단계 PLA 설계시 AND 게이트 입력수의 증가 및 보수 입력을 위한 NOT 게이트의 사용등이 문제점으로 제기됨으로서 이러한 문제점을 해결하기 위하여 Exclusive OR-AND-OR (EXANOR) 배열로 이루어지는 3단계 PLA가 Cho, Lee 와 Hwang (2) 에 의해 제안 되었다.

2단계 PLA 의 고장 검출에 있어서 Smith (3) 는 sharp (#) 를 이용하여 시험 입력을 구하였으며, Hong 과 Ostapko (4) 는 PLA 의 입력 decoder 부분이 2입력 decoder 로 된 경우에 대한 시험 입력을 구하는 방법을 제안 하였으며, Agarwal (5) 은 SAE (stuck-at equivalence) 회로를 이용하여 다중 고장 (multiple fault) 의 검출에 대하여 논하였다.

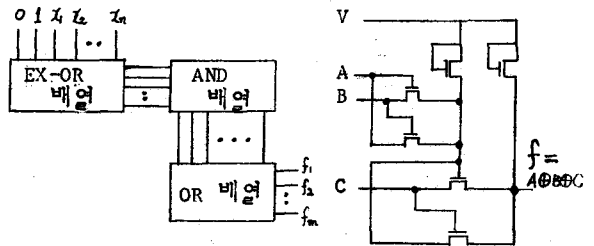
그러나 위의 논문 (3), (4), (5) 는 2단계 PLA 의 고장검출 방법에 대하여만 논한 것으로서, EXANOR 로 구성되는 3단계 PLA 의 고장검출에 있어서는 위에서 제안된 방법을 동일하게 적용시킬 수가 없다.

그러하여 본 논문에서는 EXANOR 을 사용하는 3단계 고장검출 방법을 새로이 제시 하였으며, 특히 Exclusive-OR 게이트 입력단 에서의 시험 입력을 효율적이고 정확하게 만들기 위하여, 변

형된 sharp 연산자인 ESHARP (\oplus) 연산자를 새로이 고안 하였으며 이를 시험집합 알고리즘에 적용 하였다.

2. 본론

1) 3단계 PLA 의 구조



(a) 3단계 PLA 의 구조 (b) EX-OR 의 MOS 구성
 그림 1. 3단계 PLA

3단계 PLA 의 구조 및 MOS 구성은 그림 1과 같다. 이러한 3단계 PLA 는 입력 변수가 n개일 경우에 n개의 입력 및 0, 1 의 입력이 사용되어 입력선의 수는 (n + 2) 개가 된다.

3단계 PLA 를 설계하는 자세한 방법은 참고문헌 (2) 를 참조하기 바란다.

2) 3단계 PLA 의 고장 모형

Exclusive-OR (이하, EOR 로 표시함) 게이트 입력, AND 게이트 입력 및 OR 게이트 입력이 다른 곳으로 잘못 연결되거나 연결이 끊어지는 경우가 발생한다고 가정할 경우에 3단계 PLA 에서의 고장모형은 다음 5종류의 논리적 고장 집합으로 나뉘어진다.

- ① ED (Exclusive-OR Disappearance) Fault
 입력 변수와 EOR 게이트간의 연결이 끊어짐.

② EA (Exclusive-OR Appearance) Fault

입력 변수가 EOR 게이트로 잘못 연결됨.

③ G (Growth) Fault

EOR 게이트와 AND 게이트간의 연결이 끊어짐.

④ S (Shrinkage) Fault

EOR 게이트가 AND 게이트로 잘못 연결됨.

⑤ D (Disappearance) Fault

AND 게이트와 OR 게이트간의 연결이 끊어짐.

⑥ A (Appearance) Fault

AND 게이트가 OR 게이트로 잘못 연결됨.

3) 시험집합 생성 알고리즘

앞에서 제시한 고장집합 $F = EDUGUDU$

$EA \cup S \cup A$ 에 대한 시험집합 T 를 만드는 알고리즘은 다음 6개 단계로 구성된다.

① 시험되는 PLA 에 대한 규격을 배워외역택로 입력 시킨다.

② 비트 마킹킹 (bit marking) 여부 에 의하여 아직 검출되지 않은 고장의 $f \in F$ 를 선택한다.

③ f 에 대한 시험집합 (test set) t 를 만들어, 이를 T 에 추가 시킨다.

④ 아직 고려치 않은 고장이 있으면 단계 2로 간다.

⑤ 고장이 없는 (fault-free) PLA 에 대한 시험 결과를 결정한다.

⑥ 시험집합 T 및 정확한 시험 결과를 출력 시킨다.

이중 단계 3에 대하여 중점적으로 설명하기로 한다. 단계 3에서는 단일 고장에 대한 시험 집합 생성을 위하여 패스 감응 방법 (path - sensitization method) 를 사용 하였으며, 3단계 PLA 에 대한 패스 감응 방법을 자동화 하기 위하여 큐브 표시 및 ESHARP 연산자를 사용 하였다.

ESHARP 연산자는 EOR 게이트 입력단에서 어느 한 입력 cube 의 조건을 만족하면서 다른 한 입력 cube 에 대한 EOR 게이트 출력이 0 이 되는 조건을 동시에 만족 하는 입력 변수값을 결정 하는 경우에 사용하며 그 동작은 다음과 같다.

$a_i \oplus b_i$	b_i		
	0	1	X
a_i	0	Z	Z
	1	1	Z
	X	1	Z

(소자별 ESHARP 표)

i) $\$$ 가 발생치 않은 경우로서

① 모든 i 에 대하여 Z 인 경우: $a \oplus b = a$

② 한 i 에 대하여 $a_i \oplus b_i = 0, 1$ 로 될 경우:
 $a \oplus b = (a_1, a_2, \dots, b_i, \dots, a_n)$

③ i 및 j 에 대하여 $a_i \oplus b_i = 0, 1$ 및 $a_j \oplus b_j = 0, 1$ 일 경우:

$a \oplus b = (a_1, a_2, \dots, b_i, \dots, b_j, \dots, a_n)$
 또는 $(a_1, a_2, \dots, \bar{b}_i, \dots, \bar{b}_j, \dots, a_n)$

ii) $\$$ 가 한번 발생한 경우로서

① 나머지 모든 i 에 대하여 Z 인 경우:
 $a \oplus b = \emptyset$

② 한 i 에 대하여 $a_i \oplus b_i = 0, 1$ 로 될 경우:
 $a \oplus b = (a_1, a_2, \dots, \bar{b}_i, \dots, a_n)$

iii) $\$$ 가 두번 발생한 경우: $a \oplus b = a$ 임.

여기서는 지면 관계상, 6가지 고장 모형중 가장 대표적인 ED 고장일 경우의 시험집합 생성 과정에 대하여만 살펴본다.

단계 1: 우선 고장이 발생한 EOR 게이트의 출력이 정상일 경우의 출력과는 다르게 되도록 입력을 인가 하여야 한다. 이 경우 고장이 발생한 입력 단자를 1로 놓는다. 즉 $C_j \rightarrow X$ 로 될 경우, $C_j = 1$ 로 하며 C_i 중에서 C_j 를 제외한 모든 소자를 모두 X로 놓고, 이러한 cube 를 C' 라 한다.

단계 2: 고장의 전파 (error propagation)

다음 단계로 고장은 출력으로 전파되어야 한다. 만약 $d_k = 1$ 이면 고장 출력은 k 번째 곱셈항 (product term) 으로 전파되며, $d_k = 1$ 이면 고장 출력은 l 번째 출력으로 전파 된다. 그리하여 이를 만족 하는 k, l 값 중에서 하나를 선택한다.

단계 3: 선의 정당화 (line justification)

또한 고장의 출력으로 제대로 전파되기 위하여는 AND 게이트의 다른 입력들은 1, OR 게이트의 다른 입력들은 0 이어야 하므로 OR 게이트로 부터 거슬러 올라가는 (back tracking) 경우에는 해당 되는 EOR 게이트의 출력이 모두 1 이어야 한다.

즉, $d_k = 1$ (단, $n \neq k$) 를 만족 하는 곱셈항 d^n 중 에서 $d_k = 1$ (단, $n \neq k$) 을 만족 하는 C^m 들로 구성되는 백업용 \bar{C} 이라 하면, 이들 큐브 에 포함된 EOR 게이트 출력중 하나 이상은 0 이어야 한다.

뿐만, c_i 중 에서 $d_i = 1$ (단, $i \neq 1$) 을 만족 하는 c_i 패턴 구성되는 배열을 (Γ_i) 이라 하면, 이들 큐브 역 소변 EOR 게이트 출력은 모두 1이 되어야 한다.

EOR 게이트 출력이 0이 되기 위하여는 입력 cube 중 0 또는 1인 소자를 모두 취하므로 하여 입력 시 적어 하거나, EOR 게이트 출력이 1이 되기 위하여는 입력 cube 의 소자중 에서 첫번째로 나타나는 0 또는 1인 소자는 그대로 두며, 두번째로 나타나는 0 또는 1인 소자가 있을 경우, 이 두번째 소자를 취하므로 만든 후에 입력 시 적어 하거나.

결과, 배열 (Γ_i) 에 대하여 0 또는 1인 소자를 모두 취하므로 취한 배열을 (Γ_i') 라 하면, 배열 (Γ_i) 에 대하여 두번째로 나타나는 0 또는 1인 소자를 취하므로 취한 배열을 (Γ_i'') 라 하면, 이에 대하여 ESHARP 연산자를 적용 시키면 된다.

그리하여 구하는 시험 입력은 다음과 같이 된다.

$$(c') \oplus (\Gamma_i') \oplus (\Gamma_i'')$$

또한 배열들 $(\Gamma_i') = \{c', c', c'\}$, $(\Gamma_i'') = \{c', c', c'\}$

라 하면,

$$\begin{aligned} \text{시험입력} &= (c') \oplus (\Gamma_i') \oplus (\Gamma_i'') \\ &= (c' \oplus c', c' \oplus c', c' \oplus c') \oplus (c' \oplus c' \oplus c') \\ &= c' \oplus c' \oplus c' \oplus c' \oplus c', c' \oplus c' \oplus c' \oplus c' \oplus c', \\ & \quad c' \oplus c' \oplus c' \oplus c' \oplus c' \text{ 이 된다.} \end{aligned}$$

단계 4 : 배열 이 배열이 공집합 (ϕ) 이면, 단계 2로 가서 또 다른 k, l 값을 선택 한다. 이와 같이 선택된 모든 k, l 값에 대하여 만들어진 배열이 모두 공집합 (ϕ) 이면, 해당 고장에 대한 시험 입력은 존재하지 않는다고 할 수 있다.

3. 결론

본 논문에서는 3단계 PLA 에서의 고장검출을 용이하게 하기 위하여 5가지 형태의 고장모형을 제시하였으며, 이러한 5가지의 고장모형을 검출 하는 시험집합 생성 알고리즘 (test set generation algorithm) 을 고안하였으며, 알고리즘 구성에 있어서 EOR 배열에서의 시험집합 생성을 위하여 새로운 ESHARP 연산자를 구성 하였다.

또한 알고리즘에 의해 만들어지는 시험집합 (test set) 의 크기가 전체적 시험법 (exhaustive testing) 을 적용할 경우에 만들어지는 시험집합의 크기에 비하여 1/18로 줄어들음을 보였다. 그리고 3단계 PLA에 대한 또 다른 요구사항인 3단계 PLA 내의 고장 위치

(fault location) 단별이 될 것이다. 최근의 기술적인 진보에 의하여 난수 논리 회로(random logic circuit) 에 대한 고장위치 단별은 이전에 비하여 중요하지 않게 된 반면에, PLA 에 있어서는 사용되지 않는 리던던트 (redundant) 한 부분을 이용하여 프로그램 에러 (programming error) 들을 수정하는 것이 가능하므로, 3단계 PLA 내에서의 고장 위치 단별은 중요하다고 할 수 있다.

그리하여 3단계 PLA 에 있어서, 고장 위치를 단별한 후에, 해당되는 고장에 무관하므로 3단계 PLA 를 리프로그래밍 (reprogramming) 함으로써, 3단계 PLA 소자를 계속적으로 사용하는 것이 가능하게 될 것이다.

참 고 문 헌

1. H. Fleisher and L. I. Maissel : "An Introduction to logic array", IBM J. Res. Develop., vol. 19, pp 98-109, Mar. 1975
2. D. S. Cho, J. W. Lee and H. Y. Hwang : "The three-level PLA design using EXANOR", KIEE, vol 32, No1, pp 13-23, Jan. 1983
3. J. E. Smith, "Detection of faults in programmable logic arrays", IEEE Trans. on comp., vol. c-28, no 11, pp 845-853, Nov. 1979
4. D. L. Ostapko and S. J. Mong, "Fault analysis and test generation for programmable logic arrays", IEEE Trans. on comp., vol c-28, no 9, pp 617-626, Sep. 1979
5. V. K. Agarwal, "Multiple fault detection in PLA's", IEEE Trans. on comp., vol c-29, No. 6, pp 513-522, June 1980
6. Tsutomu Sasao : "Multiple-valued decomposition of generalized boolean functions and the complexity of programmable logic arrays", IEEE trans. comp., vol c-30, pp 635-643, Sept 1981
7. S. C. Lee, Modern switching theory and digital design, Prentice-Hall Inc., 1973