

論理回路 기능검사를 위한 入力信號 算出

(Test pattern Generation for the Functional Test of Logic Networks)

趙 廷 完*, 洪 元 模**

(Chc, Jung Wan and Hong, Won Mo)

要 約

이 論文에서는 Boolean difference를 利用하여 combinational 및 sequential 論理回路에서 發生하는 機能的인 故障에 對한 test pattern을 얻는 方法을 研究하였다.

이 方法은 test pattern을 얻고자 하는 回路의 Boolean 함수의 Boolean difference를 계산하므로써 체계적으로 test pattern을 얻는 節次를 보여주고 있다. 컴퓨터에 依한 實驗結果에 依하면 이 方法은 combinational 회로 및 asynchronous sequential 回路에 適合하며, clock이 있는 flip flop을 適當히 模型化함으로써 이 方法을 synchronous sequential 회로에도 適用할 수 있음이 입증되었다.

Abstract

In this paper, a method of test pattern generation for the functional failure in both combinational and sequential logic networks by using extended Boolean difference is proposed. The proposed technique provides a systematic approach for the test pattern generation procedure by computing Boolean difference of the Boolean function that represents the Logic network for which the test patterns are to be generated. The computer experimental results show that the proposed method is suitable for both combinational and asynchronous sequential logic networks. Suitable models of clocked flip flops may make it possible for one to extend this method to synchronous sequential logic networks.

1. 序 論

디지털 시스템이 점점 複雜하여짐에 따라 이를 維持 및 補修하는 일은 어렵고도 重要的인 일이 되었다. 이러한 디지털 시스템 내에서 發生하는 故障를 체계적이며 자동적으로 찾을 수 있는 方法에 對하여 記述하고자 한다.

시스템의 故障이라 함은 올바른 動作을 하지 아니하는 狀態가 發生하였음을 意味한다. 디지털 시스템 내에서 發生하는 故障에는 여러가지가 있겠으나 論理回路에서 發生하는 故障으로는 結線이 생략되어 있거나 導線사

이에 short가 있거나, diode가 open 혹은 short 되어 있거나 transistor의 각단자사이의 open 또는 short에 의한 誤動作 등의 여러가지가 있다. 이러한 要因에 依한 故障은 一時的으로 不安全하게 나타날 수도 있으나 本論文에서는 같은 條件下에서는 항상 發生하는 영구적인 故障으로서 gate의 入力 혹은 出力단자가 電源 또는 ground에 short되어 있거나 結線사이의 open으로 因한 入力단자의 floating 등에 依한 것으로 gate의 入力과 出力단자들의 stuck-at-1(S-A-1), stuck-at-0(S-A-0)로 集約할 수 있는 故障에 局限하기로 한다. 以上과 같은 故障이 存在함을 확인하기 위하여는 適當한 信號를 入力단자에 加하여 出力단자에서 올바른 結果가 나오는 가를 살펴야 한다. 이 때에 入力에 必要한 信號를 test pattern이라 한다.

* 正會員 韓國科學院 Korea Advanced Institute of Science.

** 準會員 三星電子(株) Samsung Electronics Co. 接受日字 1976年 2月 25日

Combinational 回路의 경우 test pattern의 수는- 入力變數들이 가질 수 있는 모든 조합의 수만큼 생각 할 수 있겠으나 이것은 n개의 入力變數를 갖는 회로에 對하여 2ⁿ이므로 實用性이 없다. 가능한 限 모든 故障에 적용되는 最少限의 test pattern을 찾는 方法에 對하여는 Armstrong[6]의 equivalent normal form에 依한 方法, Roth[10]의 D-algorithm, Sehsu와 Freeman[12]의 模型 等の 여러 方法이 제시되었으며 Marino[2], Hsiao와 Chia[7], Cho[9] 등은 Boolean difference[8]를 적용하였다. 本 論文에서는 Boolean difference를 이용하여 combinational은 물론 sequential 回路에 적용할 수 있는 algorithm을 다음과 같은 가정하에 論한다.

첫째, 論理回路內에 故障은 단 하나만이 存在한다. 이것은 여러개의 故障이 同時에 存在할 경우 이들의 상쇄작용으로 出力단자에 이 故障들이 감추어 질 가능성이 있기 때문이다. 둘째, 論理回路가 기능상 重複(redundancy) 되어 있지 아니하다. 이것은 重複된 回路에 故障이 發生할 경우 出力단자에는 아무런 영향을 미칠 수 없기 때문이다. 셋째 故障은 앞서 論한 gate의 入力 혹은 出力단자의 S-A-1과 S-A-0에 限한다.

2. Boolean Difference

論理回路 M의 入力를 x₁, x₂, ..., x_n 出力을 y₁, y₂, ..., y_m 이라 하고 이 回路의 기능을 Boolean 함수로 1 ≤ j ≤ m 에 對하여 식 (1)과 같이 나타낼 수 있다.

$$y_j = F_j(x_1, x_2, \dots, x_n) \tag{1}$$

식(1)의 入力 x_i에 對한 Boolean difference는 식(2)와 같다[8]. 식(2)에서 ⊕는 論理

$$\frac{dy_j}{dx_i} = F_j^-(x_1, x_{i-1}, x_i, x_{i+1}, x_n) \oplus F_j^+(x_1, x_{i-1}, x_i, x_{i+1}, x_n) \tag{2}$$

函數 exclusive OR를 意味한다. 식 (1)과 (2)는 모든 y_j에 對하여 成立하므로 以下에서는 편의상 하나의 出力 y = y_j에 對하여서만 論한다. 식(2)에서 x_i는 論理變數이므로 그 값이 1이나 0만을 가질 수 있으므로 식 (2)는 식 (3)과 같이 표현할 수 있다.

$$\frac{dF}{dx_i} = F(x_1, x_{i-1}, 1, x_{i+1}, x_n) - F(x_1, x_{i-1}, 0, x_{i+1}, x_n) \tag{3}$$

이제 단일 論理回路 M의 x_i 入力단자로 부터 y_j 출력단자에 이르는 回線을 test한다고 하자, 그러면 우선 y_j가 x_i에 종속하도록 x₁, ..., x_{i-1}, x_{i+1}, ..., x_n의 조건을 구한 후 x_i에 주어진 入力이 y_j에 그대로 나타나기를 관찰하면 된다. 이 조건은 식(3)에서 $\frac{dF}{dx_i} = 1$ 로 놓

고 풀은 결과이다. 즉 식(3)에서 $\frac{dF}{dx_i} = 1$ 이라 함은 F가 x_i=1과 x_i=0을 대입하였을 때 그 값이 다르다는 것이다.

3. Test Pattern

가. Combinational 論理回路

그림 1은 간단한 combinational 論理回路이다. 그

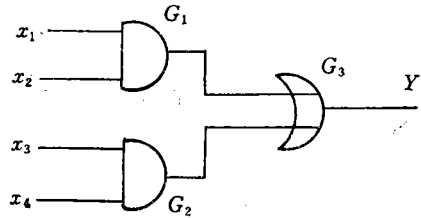


그림 1. Combinational 논리회로
Fig.1. Combinationallogic circuit

림 1에서 x₁에 加해진 入力신호는 x₁-G₁-G₃-Y의 回線을 따라 出力 Y에 傳해진다. 이렇게 x₁에서의 신호가 出力 Y에 나타나는 가를 관찰하려면 入力 x₁에만 出力 Y가 종속하여야 되므로 이 조건이 $\frac{dY}{dx_1} = 1$ 인 것이다 이러한 경우 x₁-G₁-G₃-Y의 回線을 sensitizing path라 한다. 그러므로 x₁에 對한, 즉 x₁-G₁-G₃-Y 回線에 對한 test pattern은 x₁의 S-A-0에 對한 x₁ · $\frac{dY}{dx_1} = 1$ 과 x₁의 S-A-1에 對한 $\bar{x}_1 \cdot \frac{dY}{dx_1} = 1$ 이 되는 것이다.

Boolean difference $\frac{dY}{dx_1}$ 은 chain rule[9]에 依하여 식 (4)과 같이 나타낼 수 있다.

$$\frac{dY}{dx_1} = \frac{dY}{dG_3} \cdot \frac{dG_3}{dG_1} \cdot \frac{dG_1}{dx_1} \tag{4}$$

식(4)에 依하면 $\frac{dY}{dx_1}$ 은 각 gate 들의 入力에 對한 Boolean difference를 알면 임의의 sensitizing path에 對한 Boolean difference를 求할 수 있으며 test pattern도 自動적으로 求해 진다.

그림 2와 같이 G₁의 出力에서 分岐點이 있는 後 재수렴하는 경우 x₂에서 Y에 이르는 回線 x₂-G₁-G₂-Y와 x₂-G₁-G₃-Y의 두가지가 있다.

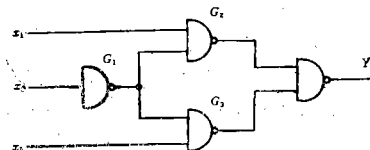


그림 2. 分岐 및 재수렴 回路
Fig. 2. Reconverging paths

따라서 $\frac{dY}{dx_2}$ 는 各 回線에 對하여 모두 求할 수 있으며 이때의 Boolean difference도 역시 식(4)과 같이 chain rule에 依하여 求할 수 있다. 이와 같이 하여 求한 test pattern은 入力뿐만 아니라 그 入力단자를 포함하는 모든sensitizing path上에서의 S-A-O 및 S-A-1 故障를 檢査할 수 있다.

나. Sequential 論理回路

Sequential 論理回路는 그림 3과 같은 asynchronous 모형으로 나타낼 수 있다.

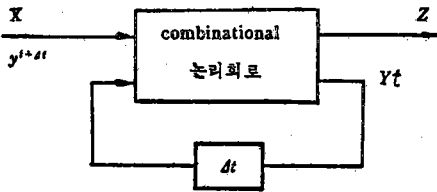


그림 3. Sequential 논리회로의 모형
그림 Fig.3. Sequential circuit model

그림 3에는 delay에 依한 feedback이 있으므로 test pattern의 순서가 重要한 問題가 된다. 入力信號가 충분히 오래동안 변하지 아니하고 $x^{t+1}=Y^t$ 가 되는 안정한 상태를 가정하였을 때 feedback된 入力を primary input로 고려하면 asynchronous sequential 回路는 combinational 回路와 同一하게 취급할 수 있으므로 combinational 回路에서의 Boolean difference가 마찬가지로 적용된다.

Feedback된 入力は 以前의 상태에 따라 결정되므로 단일 path sensitizing 條件에서 어느 set state s^k 가 必要로 된다면 test pattern을 加하기에 앞서 이 set state s^k 가 되도록 다른 入力を 事前에 加하여야 된다 이 入力を homing sequence[7]라 한다. Homing sequence H는 X를 入力, Y를 feed back 入力이라 할때 $S^k = F_1(X) + F_2(X, Y)$ 로 이루어질 때 $F_1(X)$ 를 필요로 하는 값이 되도록 X의 값을 계산함으로써 求할 수 있다. 이와 같이 하여 sequential 回路에서 Boolean difference는 식 (5)와 같이 나타낼 수 있으며 test pattern도 combinational 回路의 경우와 마찬가지로 구할 수 있다.

$$\frac{dY}{dx_i} = F(x_{i1}, s^i) \oplus F(x_{i0}, s^k) \quad (5)$$

식 (5)에서 $x_{i1} = (x_1, \dots, x_i, \dots, x_n)$, $x_{i0} = (x_1, \dots, \bar{x}_i, \dots, x_n)$ 이며 H_0 와 H_1 을 homing sequence, $f\{p_i\}$ 를 입의 path $\{p_i\}$ 의 Boolean difference의 한 term이라 하면 S-A-O의 test pattern은 H_0 와 $x_i f\{p_i\} = 1$ 이고 S-A-1의 test pattern은 H_1 과 $\bar{x}_i f\{p_i\} = 1$ 이다.

Synchronous sequential 論理回路에서는 信號의 전달이 단순한 delay에 의하여 이루어질 때 생기는

race, hazard의 問題를 해결하기 위하여 clock 入력이 있다.

이 入力は high에서 low 또는 low에서 high로 변하는 순간에 信號를 단계적으로 進展시키므로 앞에서 論한 바와는 다른 transition의 意味를 考慮하여야 한다.

SR f/f (Set Reset flip flop)의 진위표는 표 1과 같이 표현된다.

여기서 出力 Y는 $Y = S + \bar{R}y$ 로 표시된다. 이와 같은 관계는 clock이 한 번, 즉 pulse가 하나 入력이 되는 것을 전제로 한다.

<표 1.> RS f/f 진위표

시간 t에서의 入力	시간 t에서의 상태	다음(시간 t+1)의 상태	
S	R	y	Y
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	d
1	1	1	d

이런 入力は 없어야 하므로 don't care로 出力을 表示한다.

즉 sensitizing path에는 f/f을 지날 때 마다 한 번씩의 clock入력이 필요로 된다. 그런데 Boolean difference는 level의 入력을 변수로 한 함수에 대해서 얻어지는 것이므로 clock이 level의 入력으로 표현되도록 하여야만 앞에서 論한 Boolean difference를 利用한 test pattern을 얻어낼 수가 있다.

이제 sensitizing path上的의 f/f에서 clock의 入력이 必要로 되는 경우를 $C=1$, 그렇지 않은 경우를 $C=0$ 라고 표시하자.

그러면 Synchronous RS f/f의 出力 Y는 $Y = (S + \bar{R}y)C$, $\bar{Y} = (\bar{S}_0R + \bar{y})C$ 로 나타낼 수 있다. 이것을 그림 4와 같은 asynchronous모형으로 바꾸어서 생각을 한다 이제 R로부터 Y에 이르는 path의 sensitizing 條件을 구하면 $S=0, y=1, C=1$ 이 된다. 따라서 test pattern은

$$\begin{matrix} S & R & C^* \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{matrix} \text{이며 } y=1 \text{ 이 되는 homing sequence는}$$

$\frac{S}{1} \frac{R}{0} \frac{C^*}{1}$ 이 된다. 단, 여기서 C^* 는 必要한 pulse 入力의 수를 말한다.

따라서 homing sequence 및 tnst pattern은

$$SAO \begin{Bmatrix} S & R & C \\ 1 & 0 & 1 \\ 0 & 1 & 1 \end{Bmatrix}$$

$$SAI \begin{Bmatrix} S & R & C \\ 1 & 0 & 1 \\ 0 & 0 & 1 \end{Bmatrix} \text{ 이 된다.}$$

또한 入力 S 로 부터 出力 Y 에 이르는 path의 sensitizing 조건은 $\bar{R} \cdot y = 0, C=1$ 에서 $y=0, C=1$ 을 택하여 homing sequene 및 test pattern은

$$SAO \begin{Bmatrix} S & R & C \\ 0 & 1 & 1 \\ 1 & 0 & 1 \end{Bmatrix}$$

$$SAI \begin{Bmatrix} S & R & C \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{Bmatrix} \text{ 이 된다.}$$

이 test pattern에서 첫째, 셋째, 줄은 homing sequence이고 둘째, 넷째 줄은 S-A-O 또는 S-A-I의 test pattern이 된다. 또한 C 가 나타내는 수는 level high의 의미가 아닌 必要한 pulse의 수이므로 master slave f/f의 경우 1에서 0의 transition을 의미하며 positive edge trigger f/f의 경우 0에서 1의 transition을 의미한다. 이러한 test pattern은 實際의 RSf/f를 test 할 수 있는 入力이 됨을 알 수 있다.

test pattern을 계산하는 途中 $R=1, S=1$ 이 되는 경우를 배제하면 synchronous RS f/f은 0와 같이 模型化할 수가 있는 것이다.

이와 같은 模型을 임의의 回路 内の f/f에 대치하였을 때, 이 回路의 入力과 出力 사이의 임의의 path를 sensitize하려면 (단 入力으로부터 出力에 이르는 path sensitize하지 아니한다. 이것은 이미 Boolean difference를 적용할 의미를 상실했기 때문이다) path 위에서 지나게 되는 f/f의 수효만큼의 clock 入力이 必要하게 된다. 이것은 $C=1$ 이 必要로 되는 C 이 入力단자의 수를 세어서 이것을 이 때에 必要로 되는 clock의 수로 삼는다는 것을 의미한다.

이것은 homing sequence를 구하는 경우에도 마찬가지로 적용을 한다.

이와 같이 하여 SRf/f를 模型化하면 SRf/f는 combinational 회로와 마찬가지로 생각할 수 있으므로 앞에서 論한 test pattern을 求하는 方法을 그대로 적용할 수가 있다.

이러한 synchronous f/f에 direct set 또는 direct clear의 入力이 있을 경우, 이 入力들은 clock의 入力이 必要하지 아니하므로 그림 4에서의 Y 에 $\bar{D}S, \bar{Y}$ 에 $\bar{D}C$ 를 OR하여 出力을 표시할 수 있다. $\bar{D}S$ 및 $\bar{D}C$ 는 0에서 set 및 clear가 됨을 의미한다. 즉 식 (6) 및

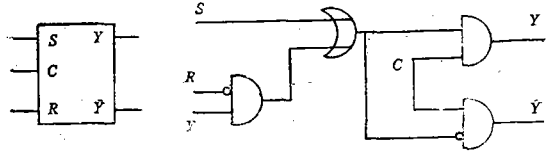


그림 4. SRF/F asynchronous 모형

Fig. 4. Asynchronous model of a SR F/F

식 (7)로 나타낼 수 있다. 이외에 Df/f 및 SRf/f의 경우 이들의 入力에 대한 出力의 함수 관계는 식 (8), (9), (10), (11)과 같이 표현 할 수 있으며 이에 의하여 各 경우에 必要한 模型을 임의의 회로 內에 代置하여 homing sequence 및 test pattern을 求할 수가 있다.

$$SR \text{ f/f} \quad Y = (S + \bar{R}y) \cdot C + \bar{D}S \quad (6)$$

$$\bar{Y} = (\bar{S} + R\bar{y}) \cdot C + \bar{D}\bar{C} \quad (7)$$

$$D \text{ f/f} \quad Y = D \cdot C + \bar{D}S \quad (8)$$

$$\bar{Y} = \bar{D} \cdot C + \bar{D}\bar{C} \quad (9)$$

$$JK \text{ f/f} \quad Y = (Jy + \bar{K}) \cdot C + \bar{D}S \quad (10)$$

$$\bar{Y} = (\bar{J}\bar{y} + K\bar{y}) \cdot C + \bar{D}\bar{C} \quad (11)$$

4. 實驗 및 結果

前節에서 論한 Boolean difference를 利用한 test

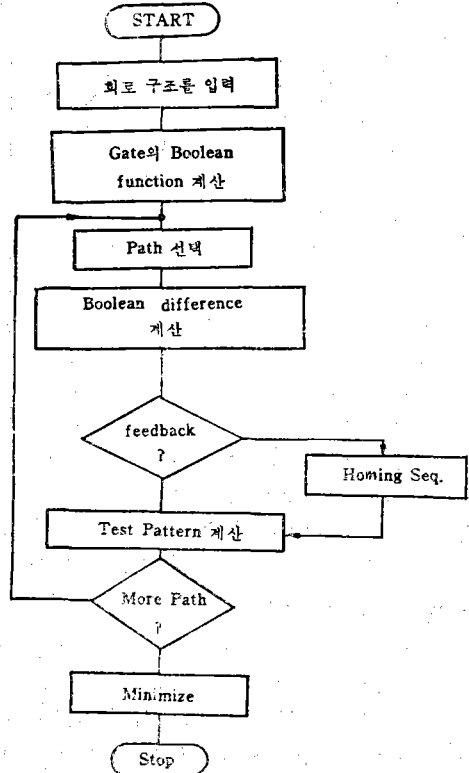


그림 5. Boolean Difference algorithm flowchart
FIG. Flowchart of the Boolean difference algorithm

pattern을 찾는 algorithm의 flowchart는 그림5와 같다.

그림 5는 combinational과 sequential 회로에 모두 적용되는 것으로 sequential 회로의 경우 feedback 회선을 잘라서 그 한쪽은 primary input로 간주하여 Boolean difference를 계산하며 combinational 회로와

다른 점은 homing sequence를 구하여야 하는 것이다. 그림 7의 회로는 그림 6[13]의 state flow를 가진 회로로서 그림 8과 같은 모형으로 나타낼 수 있다.

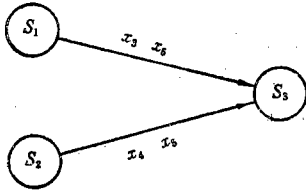


그림 6. State flow 모형
Fig.6. State flow diagram

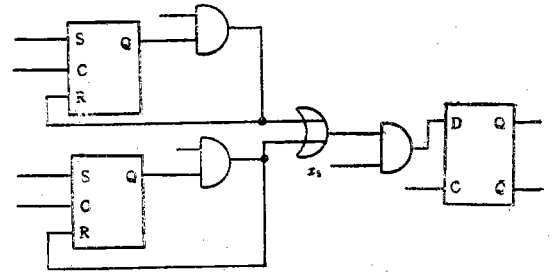


그림 7. 그림 6의 회로
Fig.7. Realization of Fig.6

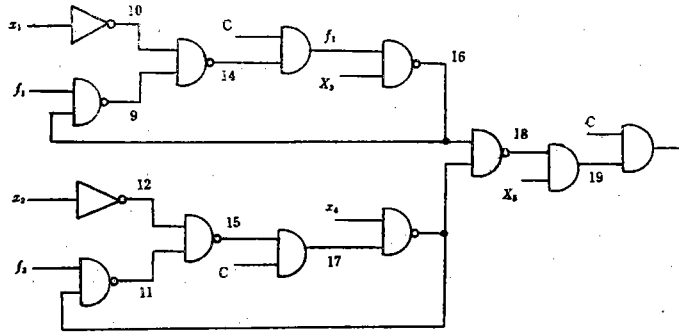


그림 8. 그림 7의 D-type F/F 이천의 모형
Fig.8. Model of Fig.7 excluding D-type F/F

표 2는 그림 7의 회로의 test pattern을 구한 컴퓨터 必要한 clocking의 수가 나타나 있다. 즉, 표 2에서 터에 의한 실험 결과이다. 各 path에 對한 pattern 및 test 5의 경우 $X_1=1, X_2=0, X_3=1, X_4=0, X_5=1, N^0$

〈표 2.〉

그림 7 회로의 Test Pattern.

test	description	X_1	X_2	X_3	X_4	X_5	No. of. Clocks
1	S-A-1 fault of path (19-5)	1	1	1	1	0	2
2	S-A-0 fault of path (19-5)	1	1	1	1	1	2
3	S-A-1 fault of path (19-18-16-3)	1	0	0	0	1	2
4	S-A-0 fault of path (19-18-16-3)	1	0	1	0	1	2
5	S-A-1 fault of path (19-18-16-14-10-1)	1	0	1	0	1	1
6	S-A-0 fault of path (19-18-16-14-10-1)	0	1	0	1	1	2
7	S-A-1 fault of path (19-18-17-4)	1	0	1	0	1	1
8	S-A-0 fault of path (19-18-17-4)	0	1	0	1	1	2
9	S-A-1 fault of path (19-18-17-15-12-2)	0	1	0	1	1	1
10	S-A-0 faeltof path (19-18-17-15-12-2)	0	0	0	1	1	2
		0	1	0	1	1	1
		0	1	1	1	1	2

of clock 1은 신호 16이 0의 state에 있도록 하는 homing sequence이며 그 아래의 test pattern은 path 19-18-16-14-10-1의 S-A-1 fault를 찾아 내기 위한 test pattern이다.

5. 結 論

本論文에서 고안한 test pattern을 찾는 방법은 다음과 같은限界가 있다. 첫째, 重複된 回路(redundancy)에 對한 對策이 없다. 물론 重複된 回路가 있을 경우 그중 하나가 故障이 나더라도 정상 動作이 可能하므로 이러한 경우의 clock 혹은 고장진단은 무의미하다. 둘째, 回路의 크기가 클수록 컴퓨터의 기억장치의 요구가 증대한다. 이 문제는 실제적인 문제로서 보조기억장치를 사용할 경우 거의 무한대의 기억용량을 제공할 수 있으므로 학술적으로 문제는 아니다. 셋째, clock이 있는 f/f의 경우 clock을 primary input로 가정하고 있다. 이 문제는 가장 심각한 문제로서 실제로 한개의 f/f의 出力이 다른 f/f의 clock 入力에 연결되어 있을때의 문제이다. 그러나 回路의 動作이 순수히 asynchronous하게 動作하도록 하였다면 별 문제는 없다. 넷째, homing sequence가 길 경우 문제이다. 이 문제도 역시 둘째번째 문제와 마찬가지로 실용적인 문제로서 학술적인 문제는 아니다.

본 論文에서 고안한 방법의 특징은 이 方法으로 얻은 test pattern은 故障를 진단하기 위한 充分한 정보를 주고 있으며 또한 이러한 algorithm에 依하여 求한 test pattern은 microprogram에 依한 컴퓨터 diagnostic 프로그램을 만들 수 있다. Sequential 回路를test 하는 데는 아직도 完全하고도 간단한 方法이 개발되지 않고 있으므로 더욱 깊은 研究가 必要하다.

參 考 文 獻

1. Melvin. A. Bruer, Design Automation of Digital Systems, Prentice Hall, Englewood Cliffs N.J., 1972.
2. Peter N. Marinos, "Derivation of Minimal Complete Sets of Test Input Sequences Using Boolean Differences," IEEE Trans. on Computer, Vol. C-20, Jan. 1971, pp. 25-32.
3. A.C.L. Ching, I.S. Reed and A.V. Banes,

- "Path Sensitization, Partial Boolean Difference, and Automated Fault Diagnosis," IEEE Trans on Computer, Vol. C-21, Feb. 1971 pp. 189-195.
4. Stephen S. Yau and Yu-Sham Tang, "An Efficient Algorithm for Generating Complete Test Sets for Combinational Logic Circuits", IEEE Trans. on Computer, Vol. C-20, Nov. 1971, pp. 1245-1251.
5. Donald L. Moon, "Bipolar Logic Test Pattern Generation", Computer Design, March 1971, pp. 63-73.
6. D.B. Armstrong, "On Finding a Nearly Minimal Set of Fault Detection Tests for Combinational Logic Tests Sets," IEEE Trans on Elect. Computer, Feb. 1962, pp. 66-73.
7. M.Y. Hsiao and D.K. Chia, "Boolean Difference for Fault Detection in Asynchronous Sequential Machines," IEEE Trans. on Computer, Vol. C-20, Nov. 1971, pp. 1356-1361.
8. L. Bearnson, M. Hsiao and F. Sellers, "Analyzing Errors with Boolean Difference, IBM Technical Report, TROO. 1619. July 1967.
9. J.W. Cho, "Combinational Logic Diagnostic Test Generation Program," IBM Product Assurance Report, CPA 22. XX-X00017, Nov. 1970.
10. J.P. Roth, "Diagnosis of Automata Failure: A Calculus and A Method, "IBM Journal of R. and D., July 1966, pp. 278-288.
11. R.G. Benetts, D.C. Brittle, A.C. Prior and J.L. Waobington, "A Modular Approach to Test Sequence Generation for Large Digital Networks," Digital Processes, Vol. 1, Spring 1975, pp. 3-23.
12. Herbert Y. Chang, Eric Manning and G. Metzger, Fault Diagnosis of Digital Sytems, Wiley-Interscience, 1970.
13. John B. Peatman, The Design of Digital Systems, McGraw-Hill, 1972.