

CAM(Content Addressable Memory)의 병렬테스팅을 위한 Built-in 테스트회로 설계에 관한 연구

正會員 趙 鉉 默* 正會員 朴 魯 京** 正會員 車 均 鉉*

A Study on the Built-in Test Circuit Design for Parallel Testing of CAM(Content Addressable Memory)

Hyon Mook Cho*, Nho Kyung Park**, Kyun Hyon Tchah* *Regular Members*

要 約

본 논문에서는 CAM에서 발생하는 모든 PSF(Pattern Sensitive Fault)를 검사하기 위한 알고리즘과 테스트회로를 설계하였다. 즉, 짧은 시간에 최소의 부가회로를 이용하여 외부의 장비에 의존하지 않고 테스트하는 내장 테스트회로를 설계하였다. 부가적으로 첨가된 회로로는 병렬비교기와 오류검출기가 있고, 병렬테스팅을 위해서 수정된 디코더를 사용하였다. 또한, 효과적인 테스트패턴을 구하기 위해 Eulerian path의 구성방법에 대해서도 연구를 수행하였다. 결과적으로, 본 논문에서 사용한 알고리즘을 사용하면 워드수에 관계없이 $325 + 2b$ (b: 비트수) 만큼의 동작으로 CAM의 모든 내용을 테스트할 수 있다. 전체 회로중에서 테스트회로가 차지하는 면적은 약 7.5% 정도가 된다.

ABSTRACT

In this paper, algorithm and built-in test circuit for testing all PSF(Pattern Sensitive Fault) occurring in CAM(Content Addressable Memory) are proposed. That is, built-in test circuit that uses minimum additional circuit without external equipment is designed. Additional circuit consists of parallel comparator, error detector, and modified decoder for parallel testing. Besides, the study on eulerian path for effective test pattern is carried out simultaneously. Consequently, using proposed algorithm, we can test all contents of CAM with $325 + 2b$ (b: number of bits) operations regardless of number of words. The area occupied by test circuit is about 7.5% of total circuit area.

I. 서 론

반도체 기술의 발달로 메모리 분야에서의 집적도

증가가 가속화됨에 따라 짧은 시간내에 메모리를 테스트할 수 있는 기술이 요구되고 있다. 높은 집적도의 메모리 칩에서 인접한 메모리 셀들간의 누설전류에 의해서 많은 오류가 발생하게 되어 이웃하는 셀의 내용을 변하게 하는데 이러한 기능적 오류의 형태를 PSF라고 한다. 이러한 PSF는 기존의 stuck-at fault와 더불어 메모리 칩에서 중요한 오류형태를 구성하게 된다. 따라서, 이러한 오류를 짧은 시간에 최소의 부가회로를 이용하여 외부의 장비에 의존하지 않고

*高麗大學校 電子工學科

Dept. of Electronic Eng., Korea University

**湖西大學校 情報通信工學科

Dept. of Information Telecommunication Eng., Hoseo University

論文番號: 9402

接受日字: 1994年 1月 5日

테스트하는 내장 테스트 회로설계와 알고리즘이 필요하게 되었다.^{[1][4]} 본 논문에서는 CAM(Content Addressable Memory) 설계경험을 바탕으로 칩상의 테스트를 수행하는 Testable CAM 회로의 설계와 테스트 알고리즘에 대해서 기술한다. 즉, 기존의 CAM에 변형된 디코더 회로, 병렬 비교기회로, 오류 검출기회로등의 부가회로를 추가한 16 워드×8 비트의 Testable CAM을 설계하였다. 전체 구성은 메모리 셀 어레이, 데이터/마스킹 레지스터, 변형된 디코더, 감지 증폭기, 병렬 비교기, 오류 검출기등으로 이루어진다. 또한, 병렬 테스트를 효과적으로 수행하기 위해서 최적의 테스트 패턴을 제공하는 Eulerian 경로에 대해서도 연구를 수행하였다.^{[2][3][5][6][7]}

본 논문에서는 설계한 각 회로에 대해서 논리, 회로 시뮬레이션을 수행한 후 ETRI 3um n-well CMOS 설계규칙으로 레이아웃을 수행하였다.

본 논문에서 사용한 테스트 알고리즘은 CAM에 있어서, 워드수에 관계없이 $325 + 2b$ (b: 비트수)만큼의 동작으로 모든 내용을 테스트할 수 있다.

II. Testable CAM의 설계

본 논문에서 설계한 Testable CAM의 전체 블럭도를 그림 1에 나타내었다.

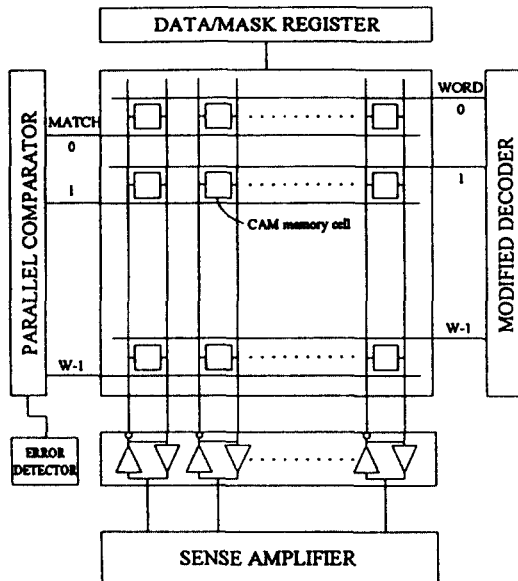


그림 1. Testable CAM의 전체 블럭도
Fig 1. Overall block diagram of testable CAM

각 블럭별 동작을 간단히 살펴보면 다음과 같다. 데이터/마스킹 레지스터는 저장 또는 매치 동작시에 입력 데이터를 받아들이는 부분이고, 감지 증폭기는 읽기 동작시 속도를 증가시키기 위한 회로이며 병렬 비교기와 오류 검출기는 테스트 모드에서 매치 데이터와 셀의 내용을 비교하여 오류를 감지하는 회로이다. 변형된 디코더는 빠른 테스트를 위해서 여러개의 워드선을 한번에 선택할 수 있게 하는 회로이다. 메모리 셀회로는 6개의 트랜지스터를 사용하는 스택형 RAM 구조와 데이터 매칭동작을 위한 4개의 트랜지스터로 구성된 Exclusive OR 구조를 갖고 있다. 그림 2는 CAM의 메모리 셀이고, 이 회로 아래부분의 4개의 NMOS가 Exclusive OR 구조를 형성한다.

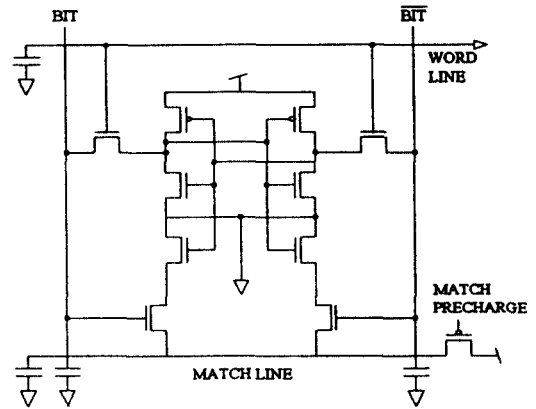
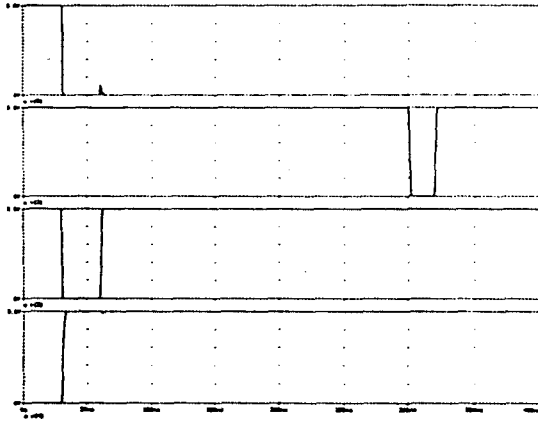


그림 2. CAM의 메모리 셀
Fig 2. Memory cell of CAM

이 회로에서 읽기나 저장할 때의 동작은 기존의 RAM과 같다. 매칭동작인 경우, 매치선은 미리 5V로 선충전되고 메모리 셀에 1이 저장되어 있을 때 비트선으로 1이 입력되면 회로아래부분의 4개의 트랜지스터가 형성하는 두개의 경로가 끊어져서 매치선은 방전되지 않고 1로 유지된다. 만약, 메모리 셀의 내용과 비트선의 내용이 다르다면 두 경로중 하나가 접지로 연결되어 매치선은 방전된다. CAM 메모리 셀에 대한 시뮬레이션 결과는 그림 3에 나타내었다.



v(2) : Bit line v(3) : $\overline{\text{Bit line}}$
v(4) : memory cell content

그림 3. CAM 메모리 셀의 시뮬레이션 결과
Fig 3. Simulation result of CAM memory cell

III. Testable CAM의 구성회로

기존의 CAM 구성회로에 병렬비교기, 오류검출기, 변형된 디코더회로를 부가하여 Testable CAM을 설계하였다.

1. 병렬비교기와 오류검출기

병렬비교기와 오류검출기의 회로를 그림 4에 나타내었다. 병렬비교기와 오류검출기는 병렬로 선택된 매치선과 연결되어, 매치데이터와 셀의 내용을 비교하여 0이나 1로 같은 값을 갖는지를 검사한다. 만일, 선택된 매치선이 다른 매치선과 하나라도 다른 값을 갖는다면 오류검출기가 오류신호인 $ERROR = 1$ 을 출력시킨다. N-채널 트랜지스터들인 T_1, T_2, \dots, T_m 은 매치선들이 모두 동시에 1인지를 검사한다. 반면에, P-채널 트랜지스터들인 P_1, P_2, \dots, P_m 은 이러한 신호들이 모두 0인지를 검사하게 된다. 트랜지스터 T_0 와 P_0 는 선충전 트랜지스터이고, T_d 는 방전 트랜지스터로서 충전기간 동안에는 단락되고 방전기간동안인 ϕ_2 동안에 turn on 된다. 병렬비교기와 오류검출기의 동작을 예들들어 설명하면 다음과 같다.

(1) 선택된 매치선에 해당하는 값들이 1로서 같은 값을 가지는 경우 :

ϕ_1 이 high 일때 T_0, P_0 스위치가 연결되어서 N-트랜지스터, P-트랜지스터는 VDD 값이 된다. 또한 S_1, S_2 스위치도 연결된다. 입력이 모두 1이므로 N-트랜지스터들이 모두 연결되며, ϕ_2 가 high일때 T_d 를 통해서 N-트랜지스터노드의 VDD 전압이 방전되고, P-트랜지스터는 여전히 VDD 전압을 유지하고 있다. S_1 스위치가 on된 상태에서 S_2 스위치는 off 되어서 오류 검출기입력은 0이 된다. 선택된 매치선에 해당하는 값들이 모두 0일 경우도 유사한 동작이 이루어진다.

(2) 선택된 매치선에 해당하는 값들이 같지 않은 경우 :

이 경우에는 몇개의 P-트랜지스터가 연결되고, 나머지는 단락된다. N-트랜지스터의 경우도 마찬가지이다. 따라서, ϕ_1 이 high가 되면서 VDD로 충전되어 있던 N, P-트랜지스터의 노드값이 ϕ_2 가 high 일때 N, P-트랜지스터를 통해서 방전되므로 S_1, S_2 스위치는 모두 단락되고 S_3, S_4 가 on 되어 오류 검출기의 입력은 1이 된다. 오류 검출기의 동작은 $\overline{TEST} = 0$ 이 되었을 때, OR 게이트의 출력에 따라서 오류신호를 발생하게 된다. 정상적인 동작을 수행하는 저장기간에는 $TEST = 1$ 로 되어 0을 유지한다. 본 연구에서는 매치선을 다섯개의 집합으로 나누어 검사를 수행하므로 다섯개의 비교신호를 하나의 집합에 있는 메모리 셀들의 내용이 동일한지를 검사하게 된다. 만일, 같지 않은 경우에는 다섯개의 입력 OR 게이트를 통해서 오류검출기를 구동시켜 오류를 감지한다. 트랜지스터 S_1, S_2, S_3 는 OR 게이트 입력을 구성한다. 만일, 선택된 모든 매치선의 값이 0이나 1이면, S_1 이나 S_2 중 하나가 도통되어 OR 게이트 입력으로 0가 들어가게 된다. OR 게이트의 출력은 트랜지스터 V_0, V_1, V_2, V_3, V_4 로 이루어진 오류 검출기로 연결된다. 오류검출기의 출력은 매치선 집합의 값이 모두 같으면 $ERROR = 0$ 을 출력하게 된다. 선택된 매치선 집합의 값들이 같지 않으면, S_1 과 S_1 가 모두 단락되고 S_3, S_4 에 의해 OR 게이트입력으로 1이 들어가게 된다. 따라서, 오류 검출기는 $ERROR = 1$ 을 출력한다. 정상적인 동작을 수행하는 동안에는 $TEST = 1$ 로 되어 0을 유지한다.

병렬비교기의 시뮬레이션 결과를 그림 5에 나타내었다.

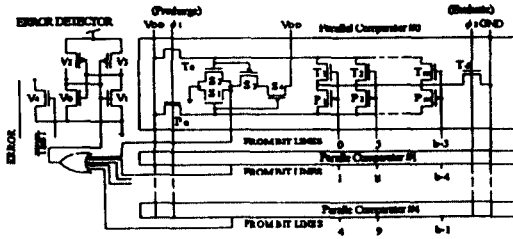
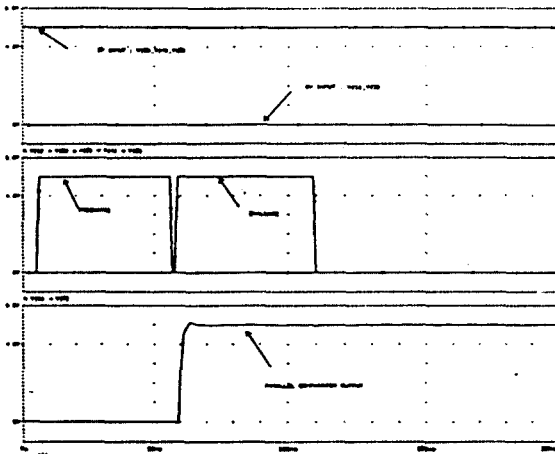


그림 4. 병렬비교기와 오류검출기의 회로도
Fig 4. Circuit diagram of parallel comparator and error detector



v(1),v(2) : 0v Input v(3),v(4),v(5) : 5v Input
v(6) : Precharge v(7) : Evaluate
v(11) : Parallel comparator output

그림 5. 병렬비교기의 시뮬레이션 결과
Fig 5. Simulation result of parallel comparator

2. 변형된 디코더 회로

변형된 디코더 회로를 그림 6에 나타내었다. 변형된 디코더 회로는 워드선을 $j = w \pmod{5}$ 에 의해서 5개의 집합으로 나누어 선택을 하게된다. 정상적인 동작에서 $\overline{SELECT} = 1$ 이고, 테스트동작시에는 $\overline{SELECT} = 0$ 이 된다. 트랜지스터 Q_1, Q_2, Q_3, Q_4 로 디코더회로가 구성된다. 테스트시에 트랜지스터 Q_5, Q_6 는 디코더출력이 입력 어드레스에 관계없이 $\overline{SELECT} = 0$ 일때 동작한다. 정상적인 동작시에는 $\overline{SELECT} = 1$ 이 되어

디코더출력은 입력어드레스 A_1, A_2, A_3, A_4 에 의해서 선택된다.

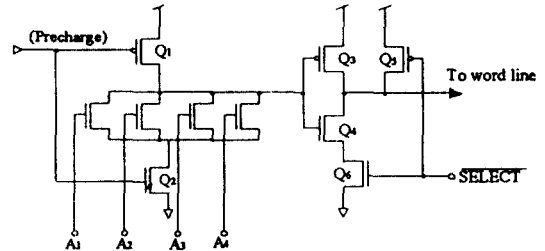
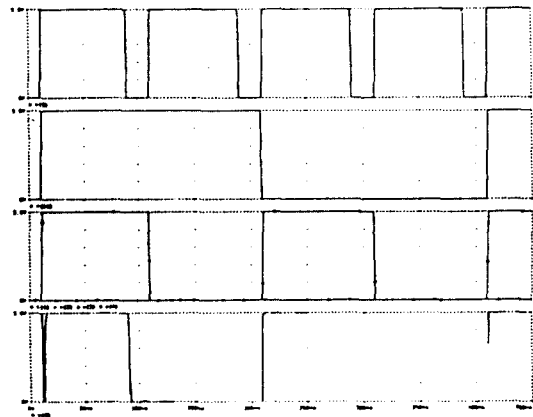


그림 6. 변형된 디코더의 회로도
Fig 6. Circuit diagram of modified decoder

변형된 디코더의 시뮬레이션 결과는 그림 7에 나타내었다.



v(5) : Precharge v(1),v(2),v(3),v(4) : Input
v(11) : \overline{SELECT} v(9) : output

그림 7. 변형된 디코더의 시뮬레이션 결과
Fig 7. Simulation result of modified decoder

IV. CAM의 병렬 테스트를 위한 알고리즘

1. Test 알고리즘의 기본이론

C_n 으로 정의되는 n-비트 CAM은 b개의 비트선과 w개의 워드선으로 $n = w \times b$ 형태의 매트릭스로 표현된다. $B = \{0, 1, 2, \dots, b-1\}$ 는 C_n 에서의 비트선의 집

합이며, $W = \{0, 1, 2, \dots, w-1\}$ 은 워드선의 집합이다. $\text{Pair}(i, j)$ 는 i 번째 비트선과 j 번째 워드선의 교점에서의 셀 주소를 나타낸다. 이러한 셀을 C_{ij} 로 표시하고 셀 C_{ij} 의 상태는 S_{ij} 로 표시한다. 본 논문에서 고려하는 PSF 모델은 제한된 이웃의 셀 동작에 의해서 하나의 셀에 오류가 생긴다고 생각한다. 테스트 알고리즘의 설명을 위해서 다음을 정의한다.^[8]

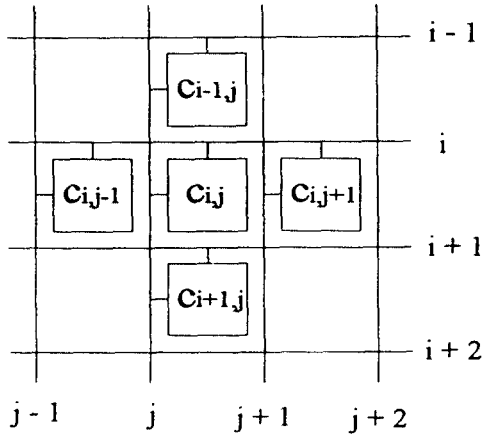


그림 8. 기준셀과 Type 1 이웃
Fig 8. Base cell and Type 1 neighborhood

정의 1: 인접한 4개의 셀들에 둘러싸인 C_{ij} 를 기준 셀이라고 하고, 인접한 셀들 $C_{i\pm 1, j}$ 와 $C_{i, j\pm 1}$ 을 포함해서 이러한 형태의 셀의 구조를 Type 1 이웃이라고 정의한다. 기준셀을 없앤 인접한 셀들의 집합을 N_1 으로 정의한다. 그림 8에 Type 1 이웃을 나타내었다.

정의 2: C_{ij} 의 값이 N_1 의 고정된 패턴에 의해서 오류가 생기면 그러한 오류를 SPSF(Static Pattern Sensitive Fault)라고 정의한다.

정의 3: N_1 중의 어떤 한 셀의 값이 0에서 1, 1에서 0으로 변함으로써 C_{ij} 의 값이 변하게 될때 그런 오류를 DPSF(Dynamic Pattern Sensitive Fault)라고 정의한다. 각 메모리 셀 C_{ij} 는 양의 정수 k 로 할당된다. Type 1 이웃에 대해서는 $k = (2j + i) \pmod{5}$, $k = 0, 1, 2, 3, 4$ 에 의해서 할당된다. 그림 9에 Type 1 이웃에 대한 셀번호 할당을 나타내었다. 예를들어 $j = 2$ 이고 $i = 4$ 인 경우 $k = (2j + i) \pmod{5}$ 에 의해 k 는 3으로 설정됨을 알 수 있다. 각 이웃의 내용은 이진 상태 벡터 $\langle s_{k-1}, s_{k-2}, \dots, s_j \dots s_0 \rangle$ 으로 표현되는데,

여기에서 s_j 는 $j = \{0, 1, 2, \dots, k-1\}$ 중의 하나로 할당된 셀의 상태를 나타낸다. 위의 예에서 $k=3$ 이므로 이진 상태 벡터는 $\langle s_2 s_1 s_0 \rangle$ 로 표현된다. 크기가 k 인 이웃의 상태 벡터는 이웃에 저장될 수 있는 모든 가능한 패턴들을 나타내는 노드로 구성된 상태 공간 그래프로 기술될 수 있다. $k=3$ 을 갖는 이웃에 대한 상태 공간 그래프를 그림 10에 나타내었다. 이진상태벡터 $\langle s_2 s_1 s_0 \rangle$ 로 표현할 수 있는 모든 가능한 경우는 000에서부터 111까지의 8가지 경우가 된다.

i \ j	0	1	2	3	4	5	6
0	0	1	2	3	4	0	1
1	2	3	4	0	1	2	3
2	4	0	1	2	3	4	0
3	1	2	3	4	0	1	2
4	3	4	0	1	2	3	4
5	0	1	2	3	4	0	1

그림 9. Type 1 이웃에 대한 셀번호 할당
Fig 9. Cell number assignment for Type 1 neighborhood

$k=3$ 을 갖는 이웃에 대한 상태 공간 그래프에서 노드는 m 으로 숫자화 되는데 여기에서, $m = \sum s_j \cdot 2^j$ 이고, $0 \leq m \leq 2^k - 1$ 이 된다. 인접한 셀에서의 0나 1을 쓰는 천이 저장동작 W_r 에 의해서 야기되는 상태의 변화는 에지(edge)로 나타낸다. 노드의 라벨들은 비트패턴을 나타내고, 화살표가 있는 에지는 1 비트 천이 저장을 나타낸다. 두개의 화살표로 된 에지는 SPSF를 검사하고, 하나의 화살표로 된 에지는 DPSF를 검사하는 천이 저장이다.(여기에서, $i=2$ 가 기준 셀이라고 가정한다.) 테스트를 완전하게 하기 위해서는 각각의 에지를 모두 경유하면서 테스트길이를 가장 짧게 하는 경로를 택해야 하는데 이러한 경로를 Eulerian 경로라고 한다. 본 논문에서는 이러한 Eulerian 경로를 이용해서 Type 1 이웃에 대한 모든 가능한 비트패턴을 표 1과 같이 구해서 테스트를 수행한다.

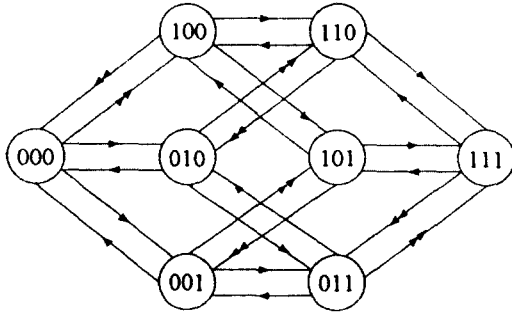


그림 10. k=3을 갖는 이웃의 상태 공간 그래프
Fig 10. State space graph of neighborhood(k=3)

표 1. Type 1 이웃에서의 가능한 비트패턴
Table 1. Bit pattern of Type 1 neighborhood

op num	S ₄ S ₃ S ₂ S ₁ S ₀	op num	S ₄ S ₃ S ₂ S ₁ S ₀	op num	S ₄ S ₃ S ₂ S ₁ S ₀	op num	S ₄ S ₃ S ₂ S ₁ S ₀
0	00000	40	01001	80	10010	120	01001
1	01000	41	01000	81	10110	121	11001
2	11000	42	01010	82	11110	122	11000
3	10000	43	01011	83	11010	123	01000
4	10001	44	01111	84	01010	124	01010
5	11001	45	01110	85	01110	125	11010
6	01001	46	01100	86	00110	126	11011
7	00001	47	01101	87	00010	127	01011
8	00011	48	11101	88	00011	128	01001
9	01011	49	11100	89	00111	129	01011
10	11011	50	11110	90	01111	130	01111
11	10011	51	11111	91	01011	131	01101
12	10010	52	11011	92	11011	132	00101
13	11010	53	11011	93	11111	133	00111
14	01010	54	11000	94	10111	134	00011
15	00010	55	11001	95	10011	135	00001
16	00110	56	10001	96	00011	136	10001
17	01110	57	10000	97	10011	137	10011
18	11110	58	10010	98	10010	138	10111
19	10110	59	10011	99	00010	139	10101
20	10111	60	10111	100	00000	140	11101
21	11111	61	10110	101	10000	141	11101
22	01111	62	10110	102	10001	142	11011
23	00111	63	10101	103	00001	143	11001
24	00101	64	10001	104	00101	144	11000
25	01101	65	10101	105	10101	145	11010
26	11101	66	11101	106	10100	146	11110
27	10101	67	11001	107	00100	147	11100
28	10100	68	01001	108	00110	148	10100
29	11100	69	01101	109	10110	149	10110
30	01100	70	00101	110	10111	150	10010
31	00100	71	00001	111	00111	151	10000
32	00101	72	00000	112	01111	152	00000
33	00100	73	00100	113	11111	153	00010
34	00110	74	01100	114	11110	154	00110
35	00111	75	01000	115	01110	155	00100
36	00011	76	11000	116	01100	156	01100
37	00010	77	11100	117	11100	157	01110
38	00000	78	10100	118	11101	158	01010
39	00001	79	10000	119	01101	159	01000
						160	00000

CAM을 위한 테스트 알고리즘은 Type 1 이웃의 모든 SPSF를 검사하는 알고리즘이다. 따라서, j개의 매치선은 j(mod5)를 이용해서 집합화된다. j 그룹의

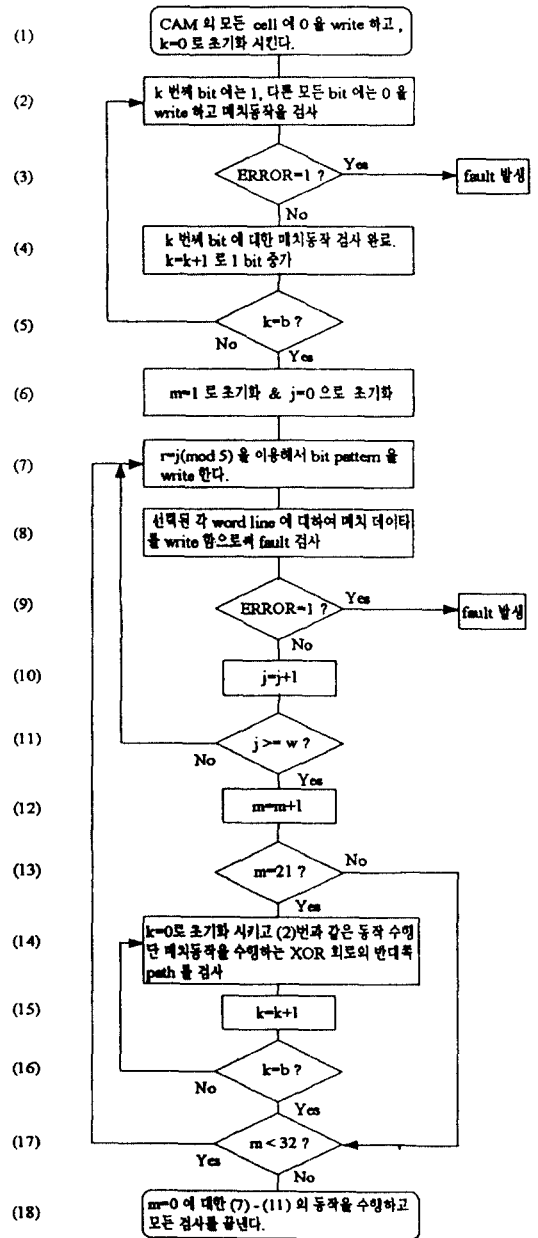


그림 11. CAM을 검사하기 위한 알고리즘
Fig 11. Algorithm for CAM testing

매치선은 $j(\text{mod}5)$ 을 이용해서 집합화 된다. j 그룹의 매치선들이 하나의 동일한 값을 갖지 않고 0이나 1을 갖는다면, 병렬비교기를 통한 오류검출기에서 오류를 검출한다.

CAM에서는 모든 비트선들이 병렬로 저장되기 때문에, 저장시에 DPSF가 없게 되므로 DPSF를 검사할 수가 없다. CAM에서는 Type 1 어웃에 대해 2개의 비트패턴들을 이용해서 검사하게 된다. 따라서 CAM의 모든 셀에 대해 SPSF를 검사하기 위해서 표 1의 처음 32개의 패턴만이 필요하게 된다. 표 1에서 동작 m 은 선택된 워드선에 숫자 m 의 비트패턴을 저장하는 동작이다. 하나의 워드선에 대한 비트수는 $b > 5$ 이므로 패턴 m 은 CAM 셀 C_{ij} 에 표 1에서의 상태 s_{ij} 가 쓰여지게 된다. 여기에서, $p = (2j + i)(\text{mod}5)$ 이다.

이 식은 그림 9의 셀번호 할당과 같다. 셀에 패턴 m 이 저장 될때, $b > 5$ 이상일때는 m 이 반복적으로 쓰여지게 된다. $j(\text{mod}5) = 0$ 인 워드선에 쓰여지는 패턴을 m^0 라고 나타낸다. 반면, $j(\text{mod}5) = r > 0$ 이면 패턴 m 은 2^r 만큼 왼쪽으로 순환 쉬프트된다. 이런 변형된 패턴은 k 번 반복된다. 그러한 패턴을 m^k 로 나타낸다. CAM을 검사하기 위한 알고리즘을 그림 11에 나타내었다. 이 알고리즘은 $325 + 2b$ 번의 동작으로 모든 PSF를 검사한다. (1)에서 모든 메모리 셀을 0으로 초기화 시킨다. (2) (5)는 메모리 셀에서 매치 동작을 수행하는 XOR 회로의 한쪽 경로를 구성하는 2개의 트랜지스터의 드레인과 소오스 부분의 단락여부를 검사하는 과정이다. 또한, (13) (16)은 XOR 회로의 또다른 한쪽 경로를 구성하는 트랜지스터의 오류를 검사하는 과정이다. (6)에서 표 1의 첫번째 비트패턴을 결정하고, 그 패턴에 대해서 변형된 디코더를 통해서 5개의 그룹으로 묶인 워드선에 해당하는 셀의 오류를 검사하게 된다. 따라서, (6) (10)에서 $m = 1$ 인 패턴에 대한 모든 셀들의 오류를 검사하게 된다. 전체 알고리즘을 수행하는데 필요한 테스트 복잡도를 구해보면 다음과 같다.

1. (1)에서 변형된 디코더를 통해서 5번의 저장 동작이 필요하게 된다. : 5 테스트 복잡도
2. (2) (5), (13) (16)에서는 1 비트에 대해서 차례로 수행하므로 비트수의 2 배에 해당하는 테스트 복잡도가 필요하다. : $2b$ 테스트 복잡도
3. (6) (12)에서 21개의 패턴을 m^k 동작을 통해서 매치 데이터를 저장하게 된다. : $(21 \times 5) \times 2 = 210$ 테스트 복잡도

4. (17)에서 나머지 11개의 패턴에 대한 (6) (12)를 수행하게 된다. : $(11 \times 5) \times 2 = 110$ 테스트 복잡도

따라서, 본 연구의 알고리즘을 이용하여 CAM의 모든 PSF를 검사하는데 $325 + 2b$ 의 테스트 복잡도가 필요하게 된다.

V. 결 론

본 논문은 CAM에서 모든 PSF를 검사하는 알고리즘을 적용하여, 칩상에서의 검사를 위해 기존의 CAM에 부가적인 회로를 첨가하여 Testable CAM 회로를 설계하였다. 또한, 병렬 테스트를 효과적으로 수행하기 위해서 Eulerian 경로를 구하는 프로그램도 작성을 하였다. 본 논문의 알고리즘을 이용하면 워드수에 관계없이 $325 + 2b$ (b : 비트 수)의 테스트 복잡도만으로 모든 PSF를 검사하게 된다. 각 회로에 대해서 SPICE를 이용하여 회로 시뮬레이션을 수행하였다. 또한, APOLLO workstation 상의 QUICKSIM을 이용하여 논리 시뮬레이션을 수행하였으며, 레이아웃은 ETRI $3\mu\text{m}$ n well CMOS 설계규칙을 이용하여 수행하였다. Testable CAM의 전체 레이아웃은 그림 12에 나타내었다. 칩의 총 면적은 $4986 \mu\text{m} \times \mu\text{m}$ 이고, 입력보호회로와 출력버퍼회로를 제외한 대

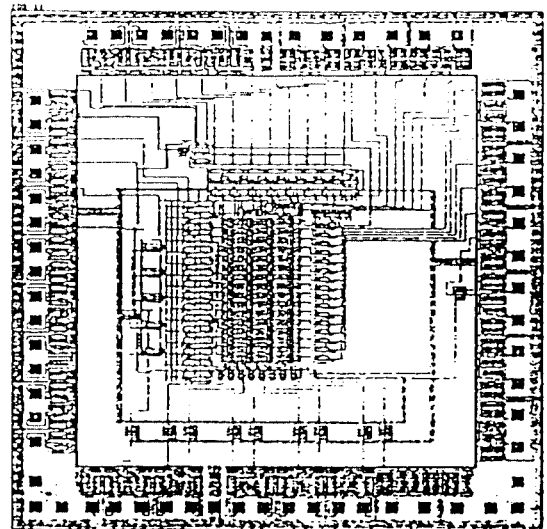


그림 12. 전체 레이아웃
Fig 12. Layout

부사용면적은 4014um × 4014um인데, 이중 테스트를 위해 부가된 회로는 약 300um × 300um(전체회로의 약 7.5% 정도)이다. 본 논문은 이전의 연구에 비해서 보다 간단한 내부회로를 내장하고 병렬테스팅을 수행함으로써 테스트에 소요되는 시간을 절약할 수 있는 장점이 있다고 하겠다.

참 고 문 헌

1. T.Kohonen, "Content-Addressable Memories," Spinger-Verlag, 1987.
2. H.Kadota, "An 8-kbit Content Addressable and Reentrant Memory," IEEE J.Solid state circuits, vol.sc-20, No 5, pp.951-957, Oct.1985.
3. T.Ogura, "A 4-bit Associative Memory LSI," IEEE J.Solid state circuits vol.sc-20, No 6, pp.1277-1282. Dec.1985.
4. T.Koo, "Integrated-Circuit Content Addressable Memories," IEEE J.Solid state circuits, vol. sc-5, pp.208-215, Oct.1970.
5. S.R.Jones, "A 9-kbit Associative Memory for High-Speed Parallel Processing Applications," IEEE J.Solid state circuits, vol.23, No.2, pp. 543-548, Apr.1988.
6. S.i.Chae, "Content Addressable Memory for VLSI Pattern Inspection," IEEE J.Solid state circuits, vol.23, pp.74-78, No 1, Feb.1988.
7. T.ogura, "A 20-kbit Associative Memory LSI for Artificial Intelligence Machines," IEEE J. Solid state circuits, vol.24, No.4, pp.1014-1020, Aug.1989.
8. D.S.Suk and S.M.Reddy, "Test Procedures for a Class of Pattern Sensitive Faults in Semiconductor Random Access Memories," IEEE Trans.on Comput, vol.C-29,pp.419-429. June. 1980.
9. 차균현, 박노경외, "연상메모리의 자동설계에 관한 연구," 한국통신학회 논문지 제15권 제10호 1990.
10. 차균현, 박노경외, "Content Addressable and Reentrant Memory(CARM)의 설계에 관한 연구," 한국통신학회 논문지 제16권 제1호 1991.
11. 차균현, 박노경외, "연상메모리 설계 및 제작에 관한 연구," 한국통신학회 논문지 제16권 제2호 1991.
12. 차균현, 박노경외, "객체지향형 프로그래밍을 이용한 CAM 생성기 구현에 관한 연구," 한국통신학회 논문지 제16권 제12호 1991.



趙 鉉 默(Hyon Mook Cho)正會員
1965年 8月 25日生
1989年 2月:高麗大學校 電子工學
科 工學士
1991年 2月:高麗大學校 電子工學
科 工學碩士
1993年 2月:高麗大學校 電子工學
科 工學博士課程修了

※主關心分野: VLSI/CAD, 通信回路設計等.

朴 魯 京(Nho Kyung Park)正會員
1958年 1月 8日生
1984年 2月:高麗大學校 電子工學
科 工學士
1986年 2月:高麗大學校 電子工學
科 工學碩士
1990年 2月:高麗大學校 電子工學
科 工學博士

1988年 2月~現在:湖西大學校 情報通信工學科 副教授

※主關心分野: VLSI/CAD, 畫像處理用 집 設計等.



車 均 鉉(Kyun Hyon Tchah)正會員
1939年 3月 26日生
1965年:서울大學校 電氣工學科 工
學士
1967年:美國 일리노이大學校 工學
碩士
1976年:서울大學校 工學博士
1987年~現在:高麗大學校 電子工
學科 教授

※主關心分野: CAD 및 通信시스템等.