

Domino CMOS NOR-NOR Array Logic 와

Testable Design에 관한 연구

* 이종호 조상복

울산대학교 전자 및 전산기 공학과

A study on the Testable Design of Domino

CMOS NOR-NOR Array Logic

Joong Ho, Lee Sang Bock, Cho

Dept. of Electronic and Computer Engineering, university of ulsan

abstract

This paper proposes testable design method of Domino CMOS NOR-NOR Array Logic design method. Previous Domino CMOS PLA method is composed of 2 level NAND-NAND Logic. Because NOR-NOR Logic is realized by a parallel circuit, this method can prevent delay time each level and DNOR-PLA include testable circuit system that DNOR-PLA circuit. DNOR-PLA testable algorithm is realized on Prime(Primos) in Pascal language and DNOR-PLA circuit is simulated by PSPICE.

1. 서론

VLSI 기술에 PLA 와 같은 Array Logic 이 많이 사용되고 있고 PLA 방식 중에서도 Domino-CMOS Logic 설계 방식은 CMOS 의 특징을 가지면서 chip area 와 속도가 개선 되고 NMOS 기술을 그대로 이용 할 수 있는 방식으로 발전하기 시작했다. [1,2,3,4,5] 그런데 지금까지의 PLA 구조 방식에 있어 입력 함수가 많아 질수록 transistor 들의 직렬 연결로 인한 capacitance 효과로 인해 delay time 이 문제가 되어 왔다. [2,6] 본 논문에서는 이런 현상을 방지하기 위해 CMOS-Domino Logic 의 NAND-NAND 구조를 NOR-NOR 구조로 바꾸어 줌으로서 기본의 NAND-NAND Logic 함수들의 직렬 연결로 인한 커페시터 효과와, 그로인한 pre charge transistor 와 evaluate transistor 간의 커페시터 양의 불균형과 delay time 의 불균형을 방지할 수 있는 방식을 제안한다. 제안되는 CMOS Domino NOR-NOR Logic은

기존의 NAND-NAND Logic에 비해 첫째단의 인버터가 바뀜으로 그만 큼 트랜지스터 수가 감소 되고 징적도도 높아지며 capacitance 효과를 제거 함으로서 응답 속도가 빨라 지게 된다. 또한 이 회로의 테스터 생성 알고리즘 및 테스터 절차를 제안한다.

2. Domino CMOS Logic

Domino CMOS 회로는 Fig.1과 같이 함수 형성을 위한 NMOS 함수 block과 clocking gate, 그리고 inverter로 구성되어 있다. [2] Domino CMOS Logic은 CMOS 와 같이 전력 소모가 적을 뿐만 아니라, chip arrer가 작아서 징적도가 높고 구조적 특징으로 인해 하나의 클럭 폴스에 의해 함수 값을 얻을 수 있으므로 timing 문제에 있어서 다른 디내믹 회로 보다 안정하며, 응답 속도 또한 1.5-2 배 정도 빠르다는 잇점이 있다.

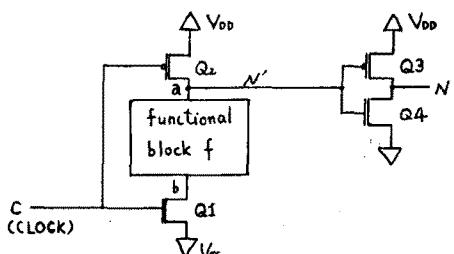


Fig.1

Fig.1에서 clock이 1 일 때 함수 f 중 true path가 형성되면 a 와 b 사이에 path가 형성되어 N1은 f 와 Q1에 의해 0 이 된다. 이때 NOT gate(Q3 와 Q4)에 의해 출력 N 이 1 이 된다. 즉 함수 block 이 "true"이고 clock 이 1 일 때만 변화가 일어 나고 그 외에는

형상 0 으로서 CMOS-Domino Logic 의 속도는 논비 값 1 이 진화 되는 시간에 의해 결정 된다. 그런데 이 Domino CMOS Logic 은 NAND-NAND Logic 이기 때문에 각 함수의 길이가 길어 질수록 true path 를 형성 하기 위한 트랜지스터 갯수가 많아 진다. 즉 true path 를 형성 하기 위해 트랜지스트의 직렬 연결이 길어져서 커피시턴스가 커진다. Domino-CMOS Logic 의 구조적 특징으로 인해 precharge 될 때는 P-MOS 트랜지스트 하나만 통과 하게 되고 evaluate 될 때는 여러 단의 N-MOS 트랜지스트를 통과 해야 하므로 여기서 발생 되는 커피시턴스 망의 불균형은 delay time 의 불균형을 초래 하여 이 현상이 심각 해지면 clock 의 일정 시간 동안 "1" 의 상태로 triggering 할동안 evaluate 트랜지스터의 출력을 얻을 수 있을 뿐만 아니라, 출력을 얻었다 해도 응답 속도가 늦어진다는 문제점이 있다.

3. Domino CMOS 회로의 NOR-NOR 설계

최근 융용 되고 있는 PLA 보는 CMOS 기술 인데 특히 David A. Hodges 와 Horace G. Jackson 은 기존의 CMOS 보다 우수한 특장을 지닌 CMOS-Domino Logic 을 이용하여 AND-OR 형의 2 level 구조로 된 Domino PLA 를 구성하였다. 이것은 속도가 매우 빠르고, 고집적도를 가지며, 전력 소모 가 매우 적은 장점을 가지며, 대부분 NMOS 트랜지스트들이 사용된다. 그런데 입력 함수들이 길어짐에 따라 Domino CMOS 의 NAND-NAND 구조로 인한 소자간의 직렬 연결로 delay time 의 불균형이 생겨 원치 않는 출력을 얻을 수 있는데 이를 방지하기 위해 모든 소자를 병렬 연결 하는 NOR-NOR 구조로 설계 할 수 있다.

<장의 1> 입력 가능한 decoder 수를 Y(입력 가능한 최대의 수)라 놓고 각 입력을 X 라 놓고 여러 입력들의 조합을 Py 라 놓으면 이것은 출력 함수가 된다.

$$Py = X_1' + X_2' + \dots + X_n$$

$$\equiv X_1 X_2 \dots X_n$$

장의 1 에 의해 가능한 출력(F)을 병렬 식으로 표시하면 아래와 같다.

$$F_1 = P_{11} + P_{12} + \dots + P_{1Y}$$

$$F_2 = P_{21} + P_{22} + \dots + P_{2m}$$

$$\dots$$

$$F_n = P_{n1} + P_{n2} + \dots + P_{nk}$$

여기서 $m \leq Y$, $k \leq Y$ 이고 n 는 출력 가능한 최대의 수이다.

<장의 2> 위와 같이 표현된 함수에서 동일한 열(column)에 속하는 직항들을 하나의 직항군이라 한다. 이때 위의 함수는 Y 개의 직항군을 갖는다.

<Domino CMOS NOR-NOR 구조 설계 방식>

- 1) 각 출력 함수는 각각 하나의 CMOS-Domino Logic 의 함수 블럭 내에 실현 된다. 즉 함수 블럭은 NMOS 의 Nor Logic 으로 직항선을 구성하여 병렬 연결하며, 전체적으로 array 구조를 이룬다.
 - 2) 함수내의 동일한 직항들은 동일군에 둔다. 예를 들면 F_1 과 F_2 에 동일 직항이 있을 경우, $P_{11} = P_{n1}$ 또는 $P_{12} = P_{21}$ 가 되도록 같은 일에 둔다.
 - 3) 첫번째 직항군부터 차례로 직항선을 구성한다.
 - 4) 각 출력 함수에 해당하는 직항선들을 각각 병렬 연결하여, 함수 블럭을 구성하고, CMOS-Domino Logic 으로 연결 한다.
 - 5) 같은 출력군에 해당하는 함수들은 한군으로 묶어서 (4)와 똑같이 한다.
 - 6) 출력은 PMOS 와 함수 사이에서 연결 된다.
- 위의 설계 방식의 예로 n=3 이고, Y=4 인 입력과 출력을 가지는 함수를 Domino CMOS NOR-NOR 회로로 Fig.2 에 나타 내었다. 아래는 그 함수를 나타낸 것이다.

$$F_1 = X_1 X_2 + X_1 X_2' X_3' X_4$$

$$F_2 = X_1' + X_2' X_3$$

$$F_3 = X_1 X_2 + X_1 X_4'$$

여기서 각 직항선은 array 구조로 구성되어 있어 PLA 의 AND array 와 같은 기능을 수행 한다. 또한 각 직항선은 병렬로 연결되어 OR logic 을 실현 시킴으로서, PLA 의 OR array 기능을 대신한다. 그래서 Domino-CMOS NOR-NOR 구조는 AND-OR logic 의 PLA 와 동일한 기능이 실현될 수 있다.

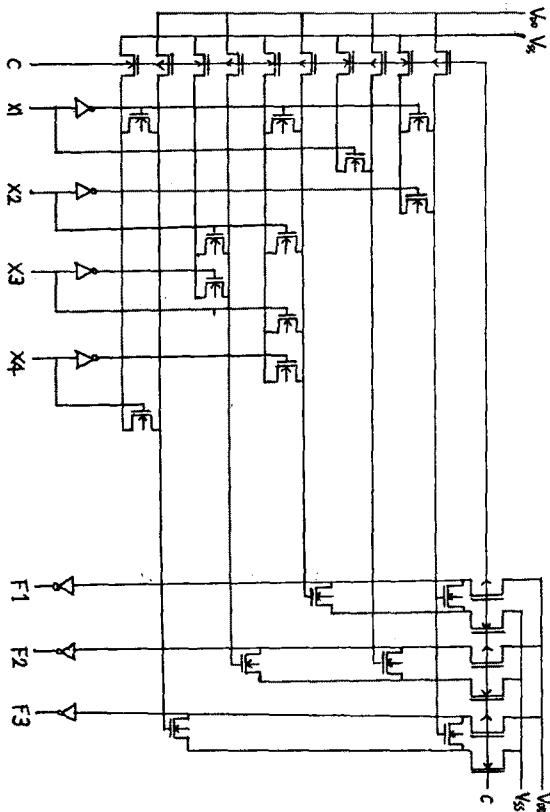


Fig2. Domino CMOS NOR-NOR Logic Circuit

4. Domino CMOS NOR-NOR 구조의 delay time 과 특징의 정량적 분석

Domino CMOS NAND-NAND Logic 의 truth path 생성에서 발생되는 delay time 의 불균형을 방지하기 위해 NOR-NOR Logic 으로 재 구성하였다. 즉 function block 을 병렬 연결 함으로 인해 true path 의 길이가 길어져도 상관 없이 트랜지스터 한개에 해당되는 커피시티 효과 밖에 나타내지 않으므로 delay time 의 불균형을 아주 효과적으로 제거 할 수 있다. 그렇게 된으로 인해 정상 동작을 얻을 수 있게 된다. 아래에 정량적인 delay time 을 계산했다.

<정의3> 입력이 m개 일 때 P 형 소자 한개에 해당하는 최소의 rise time 을 T_r , N 형 MOS 소자 전체에 해당하는 소자의 fall time 을 T_f 타 한다. 여기서 트랜지스터의 기생 커피시티 효과는 무시 한다.

. 정의 3에 의해 Domino CMOS NAND-NAND Logic 의 전자 delay time ;

$$. T_r = R_p(C_d + C_l)$$

$$R_p = 4/(B_p V_{dd}): \text{resistance of p-device in gate}$$

B_p is the MOS transister gain factor

C_d = capacitance of a unit drain area

C_l = other load capacitance on gate (routing and fan-out)

$$. T_f = m R_n(C_d + C_l)$$

$$R_n = 4/(B_n V_{dd}): \text{resistance of n-device in gate}$$

. Domino CMOS NOR-NOR Logic 의 delay time;

$$. T_r = R_p(C_d + C_l)$$

$$. T_f = R_n(C_d + C_l)$$

위에서 보는 바와 같이 rise time 은 NOR-NOR Logic 이나 NAND-NAND Logic 이나 같음을 알 수 있다. 그리고 fall time 은 NOR 구조가 NAND 구조보다 m(입력 갯수) 배나 개선 됐다는 것을 알 수 있다. 그러면 다른 어떤 Logic 방식 들과 비교 해보자. Domino CMOS NAND Logic 보다 invert 가 줄어 들었으므로 인해 속도가 빨라지고, 전력 소모가 적고, 집적도가 높아진다. 그리고 simplified Domino-CMOS (SDC)[5] 에 비하면 SDC 는 clock 의 조절이 Domino CMOS NOR-NOR Logic 보다 복잡하다. 그리고 SDC 는 기생 커피시티 헌상을 일으킨다. 그리고 CMOS Domino Logic Array(CLA) [5] 와 비교하면 CLA 는 1 level Logic 으로서 출력을 얻을 수 있어 소자당은 줄어 들지만 중복 되는 함수가 많아 지면 중복 되는 모두를 설계 해야 한다는 단점이 있기 때문에 Domino-CMOS NOR-NOR Logic 은 위의 설계 방식 들의 단점을 보완 할 수 있다.

5. Testable Design Domino CMOS

NOR-NOR Logic

Domino CMOS NOR-NOR 회로의 test 도 CLA 의 방식과 같이 Design 할 수 있다. 우선 Domino CMOS NOR-NOR 회로의 고장 예를 살펴 보자.

. 부가 회로에서의 고장

1. Shift Register 의 stuck-at 고장

2. 트랜지스터의 단신 고장 (stuck-open) 과 도통 고장 (stuck-on)

. pull-up 및 pull-down 회로의 고장

3. stuck-on 고장

4. first level (입력단) 의 n 형 과 p 형 트래지스터의 stuck-open 고장

5. second level (출력단) 의 n 형과 p 형 트랜지션은 각 적항근을 구별한 것이다.

스터의 stuck-open 고장

6. Stuck-at 고장(출력선)

.array에서의 고장

7. 입력단의 stuck-at 고장

8. missing device 고장 (트랜지스터가 존재해야 될

부분에 없는 것)

9. extra device 고장 (트랜지스터가 없어야 될 부분

에 있는 것)

10. 트랜지스터의 stuck-open or stuck-on 고장

위와 같이 고장 예를 살펴 보았다. 그러면 test 칠차에 관해 알아 보자. 테스트를 용이하게 하기 위해서 Shift Register 를 사용한 부가 회로를 추가시켰다. 정상 동작 일때는 모두 "1"로 두어서 test 할 수 있게 했다.

임의의 함수를 실현한 Domino CMOS NOR-NOR Logic 의 PLA 특성 행렬 개념을 도입하여 다음과 같이 정의 한다.

<정의4> n개의 입력과 k개의 적항선을 가진 Domino CMOS NOR-NOR Logic에 대하여 다음과 같은 요소들로 구성되는 ($k \times n$) 행렬을 Domino CMOS NOR 의 입력 특성 행렬이라 한다.

$X(i,j) = 1$; true bit line 과 적항선(i)의 교점에 소자 존재

$X(i,j) = 0$; complement bit line 과 적항선(i)의 교점에 존재

$X(i,j) = -$; 어느 bit line 과의 교점에도 존재하지 않을 때

<정의5> k개의 적항선과 m개의 출력을 가진 Domino CMOS에 대하여 다음과 같은 요소들로 구성되는 ($k \times m$) 행렬을 Domino CMOS NOR 의 출력 특성 행렬이라 한다.

$f(i,j) = 1$; 적항선(i) 와 출력선(j), 즉 F_j 에 연결 되었을 때.

$f(i,j) = 0$; 출력선 F_j 에 적항선(i) 이 연결 되지 않았을 때.

위의 정의로부터 Domino CMOS NOR-NOR에 대한 입력 및 출력 특성 행렬을 구하여 중복되는 적항끼리 묶어서 같은 적항으로 만든 행렬이 Fig.3 이다. 점

X1 X2 X3 X4 F1 F2 F3

1 1 - - 1 - 1

0 - - - - 1 -

----- -----

1 0 0 1 1 - -

- 0 1 - - 1 -

1 - - 0 - - 1

Fig.3 modification of the personality matrices

<테스터 집합 생성 알고리즘>

(절차1) Fig.3 의 특성 행렬로부터 테스터 패턴 $T(i, j)$ 를 생성한다.

<단계1> $X(i)$ 의 don't care 인 입력을 모두 1로 두어 테스터 입력 t_1 으로 한다. 이때는 정상 출력 $F(i)$ 가 된다.

<단계2> $X(i)$ 의 don't care 인 입력을 모두 0으로 두어 t_2 를 한다. 이때도 정상 출력 $F(i)$ 가 된다. don't care가 없으면 t_2 는 필요 없다.

<단계3> $X(i)$ 의 don't care 가 아닌 부분에 대해 하나씩 그 보수 값을 취하여 t_3 입력 패턴으로 한다. 출력은 $F(i)'$ 가 된다.

$X(i)$: 입력 특성 행렬의 i 번째 행

$F(i)$: 출력 특성 행렬의 i 번째 행

(절차2) $i=1$ 부터 $i=k'$ 까지 절차 1 을 반복 수행후 종료 한다. 여기서 j 는 i 번째 행이 속해 있는 product line 의 행 번호이다.

Fig.3 으로 부터 table2 에 testable set 를 작성했고 Fig.4는 testable circuit 이다.

X1 X2 X3 X4 F1 F2 F3

-----|-----

t1 : 1 1 0 0 1 - 1

T(1,1) t2 : 1 1 1 1 1 - 1

t3 : 1 0 - - 0 - 0

: 0 1 - - 0 - 0

-----|-----

t1 : 0 0 0 0 - 1 -

T(2,1) t2 : 0 1 1 1 - 1 -

	t3 : 1	- - -	- 0 -
	t1 : 1	0 0 1	1 - -
	t3 : 0	0 0 1	0 - -
T(3,2)	: 1	1 0 1	0 - -
	: 1	0 1 1	0 - -
	: 1	0 0 0	0 - -
	t1 : 0	0 1 0	- 1 -
T(4,2)	t2 : 1	0 1 1	- 1 -
	t3 : -	1 1 -	- 0 -
	: -	0 0 -	- 0 -
	t1 : 1	0 0 0	- - 1
T(5,2)	t2 : 1	1 1 0	- - 1
	t3 : 0	- - 0	- - 0
	: 1	- - 1	- - 0

Table.2 testable set for the Domino CMOS NOR-NOR Logig in fig.4

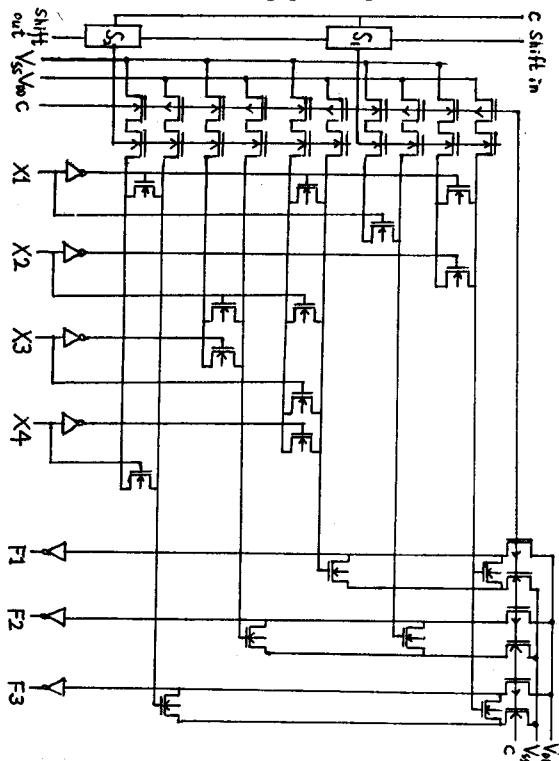


Fig.4 Testable Design of the Domino CMOS NOR-NOR Logic circuit

테스트 집합으로 수행 되는 전체 테스트 절차는 다음과 같다. Shift Register 의 상태를 테스트하려는 bit(출력 근)에만 1 을 인가하고 나머지는 0 으로 한다. 그런후에 테스터 패턴 T(i,j)의 입력을 n개의 외부 입력에 인가하고, m개의 외부 출력을 통해 관찰하여 테스터 패턴의 출력과 비교 확인 한다.

5. 결 론

기존의 NAND-NAND Logic 뿐만이 아니라 모든 PLA 구성 방법에서 각 합수 구성에서 트랜지스터들의 직렬 연결로 인한 커페시티스 영향으로 delay time 의 불균형을 재거함으로서 fall time 이 m(입력 소자 갯수)배나 빨라짐으로 인해 출력의 응답이 안정 되고 속도도 빨라지는 Domino CMOS NOR-NOR Logic 의 설계 방식을 제안하였다. 아울러 부가 회로를 사용하여, 테스터가 용이 하도록 설계 하였고, 이에 대한 테스터 집합 생성 알고리듬과 테스터 절차를 제안하였다. 이 Test 방식은 고전적인 stuck-on 고장 및 stuck open 고장 하지도 모두 고려 함으로서 매우 높은 검출율을 기대 할 수 있다.

앞으로 Domino CMOS 회로에 대한 기술이 발전함에 따라 Domino CMOS NOR-NOR 방식이 많이 사용 되리라 기대 된다.

참 고 문 헌

- [1] Fredrick J.Hill and Gerald Peterson "Switching theory & Logical Design", 1985.
- [2] Neil West and Karen Eshraghin, "Principle of CMOS VLSI Design. A System Perspective", 1985.
- [3] 김용두, 조상복, 임인철 "CMOS PLA 설계 및 stuck-open 고장 검출" 대한 전자 공학회 추계 종합 학술 대회논문집, vol.8, No.2, pp547-550, 1985.11.
- [4] 김용두 "CMOS PLA 설계 및 stuck-open 고장 검출에 대한 연구" 한양 대학교 산업대학원 석사 학위 논문, 1985.11.
- [5] Lance A.Glasser & Daniel W.Dobber Publ "The Design and Analysis of VLSI Circuit", 1985.