

전류구동 CMOS 다치 논리 회로설계 최적화연구

The Optimization of Current Mode CMOS Multiple-Valued Logic Circuits

최 재 석

Jai-Sock Choi

요 약

전류모드 CMOS 회로기반 다치 논리 회로가 최근에 구현되고 있다. 본 논문에서는 4-치 Unary 다치 논리 함수를 전류모드 CMOS 논리 회로를 사용하여 합성하였다. 전류모드 CMOS(CMCL)회로의 덧셈은 각 전류 값들이 회로비용 없이 수행될 수 있고 또한 부의 논리 값은 전류흐름을 반대로 함으로써 쉽게 구현이 가능 하다. 이러한 CMCL 회로 설계과정은 논리적으로 조합된 기본 소자들을 사용하였다. 제안된 알고리즘을 적용한 결과 트랜지스터의 숫자를 고려하는 기존의 기법보다 더욱 적은 비용으로 구현할 수 있었다. 또한 비용-테이블 기법의 대안으로써 Unary 함수에 대해서 범용 UUPC(Universal Unary Programmable Circuit) 소자를 제안하였다.

Abstract

The implementation of Multiple-Valued Logic(MVL) based on Current-Mode CMOS Logic(CMCL) circuits has recently been achieved. In this paper, four-valued Unary Multiple-Valued logic functions are synthesized using current-mode CMOS logic circuits. We properly make use of the fact that the CMCL addition of logic values represented using discrete current values can be performed at no cost and that negative logic values are readily available via reversing the direction of current flow. A synthesis process for CMCL circuits is based upon a logically complete set of basic elements. Proposed algorithm results in less expensive realization than those achieved using existing techniques in terms of the number of transistors needed. As an alternative to the cost-table techniques Universal Unary Programmable Circuit (UUPC) for a unary function is also proposed.

Key words : Multiple-Valued Logic (MVL), Current Mode CMOS, Cost-table, Universal Unary Programmable Circuit(UUPC), Logic Design.

I. 서 론

최근 다치 논리회로설계기법으로 주목받고 있는 전류모드 CMOS 논리(CMCL) 회로는 전력소모, 칩 점유율이 낮고, 동작특성이 매우 우수하므로 IIL(Integrated InjectIo n Logic)과 같은 기존의 전류모드 회로설계기법을 쉽게 응용할 수 있어 다치 논리함수를 구현하는데 적합하다 [1-4]. CMOS를 사용한 다치 논리회로설계의 장점은 첫째 기존의 2치 CMOS에서의 풍부한 설계경험, 둘째 아날로그 동작을 수행하는 능력, 셋째 SoC(System on Chip)와 같은 One chip VLSI에서 2치 논리회로와 혼용하여

설계할 수 있다. 이는 다치 논리 공정이 기존의 2치 논리 회로 공정과 매우 유사하므로 가능하다. 넷째 CMCL 회로의 논리회로 구현성 등이다[3]. 비용-테이블 합성, 분해 방법은 다치 논리 회로설계에 CCD(Charge Coupled Device) 회로를 사용하면서 시작되었고 CMCL 회로설계의 비용-테이블 개선되어 적용되었다[1,5,9]. 본 논문에서는 다치 전류모드 CMOS 회로설계 방법의 설계비용과 역 전류동작이 가능한 부-회로를 제안하여 기존의 방법을 개선하였고, 이 부-회로들을 사용함으로써 기존의 비용테이블보다 더욱 유연한 비용-테이블을 회로설계에 적용할 수 있다. 제안된 부-회로는 T-gate[10]와 기능면에서 동일하고 PLA와 같은 다-입력/다-출력 조합 논리 회로를 설계하는데 유용한 Universal Unary Programmable Circuit(UUPC)를 제안한다.

*인덕대학 정보메카트로닉스과

접수 일자 : 2005. 5. 6 수정 완료 : 2005. 7. 25

논문 번호 : 2005-2-1

II. 기본 회로구성

2.1 전류모드 CMOS 기본 회로요소

전류모드 CMOS 논리회로에서 변수들은 연속되지 않은 일정한 전류 값에 의해 표현되고 팬 아웃은 하나로 제한한다. 영역비용함수(area cost function)는 Kerkhoff와 Robroek[6]에 의해 칩 면적을 최소화 하는 방법으로 CCD회로에 처음 적용하였다. 전류모드 CMOS 논리회로 설계에서의 비용함수는 회로구현에 사용하는 트랜지스터 수 (TC: Transistor Count)에 비례하고 이 수로써 필요한 칩 영역을 측정할 수 있다[1,7]. 본 논문에서는 전류모드를 사용하므로 기존의 방법과는 달리 합산기와 감산기를 구현하기 위한 비용은 0 이 된다. 비용-테이블에서 사용하는 기본 전류모드 CMOS 회로는 다음과 같으며 이는 많은 전류모드 CMOS 논리회로들을 분석한 후 회로구현에 가장 적합한 것들로서 선택한 것이다[1.4.7.8].

그림 1은 전류모드 CMOS 기본 논리회로에 대한 기호, 논리식, 회로구현에 대해 보여준다.

① 합산기(sum) : 전류모드를 사용한 회로에서의 합산은 매우 간단한데 이는 둘 또는 그 이상의 신호를 결합하여 만들어지는 연결연산자이다. 합산의 회로 구현은 다수의 입력과 하나의 출력사이의 단순한 상호 연결이다. 따라서 비용은 0 이다.

② 상수 발생기(constant generator) : 상수 발생기는 출력이 $-(r-1)$ 부터 $(r-1)$ 사이의 임의의 양수 혹은 음수값의 형태로 발생하고 부의 논리 값 i 는 전류량 i 와 같고 전류의 흐름 방향만 반대인 논리 값을 표현한다. 상수 발생기의 회로는 병렬로 연결된 MOS N-형 트랜지스터, P-형 트랜지스터로 구성한다. 트랜지스터의 게이트는 기준전압에 연결되며 기준전압은 임의로 생성될 수 있다.

③ 미러(mirror, N-형, P-형) : 미러는 단일입력- 다출력 논리 연산자인데 N-형과 P-형이 있다. 미러는 BNDP(Bound-Positive)와 BNDN(Bound-Negative)을 사용하여 다음과 같은 대수적 연산이 가능하다.

$$BNDP(x) = MAX(0, x)$$

$$BNDN(x) = MIN(0, x)$$

이러한 미러들의 회로는 그림과 같이 입력전류의 부의 값을 출력하며 입력전류의 어떠한 정수 곱을 출력할 수 있다. 입력전류의 부(양)의 값은 NMOS(PMOS) 미러에

이름	기호	논리식	회로구현
합산기		$Y = x_1 + x_2 + \dots + x_n$	
상수 발생기		$Y = -b$	
		$Y = b$	
미러		$Y_i = -b_i * u$ $U = MAX(x, 0)$	
		$Y_i = -b_i * u$ $U = MIN(x, 0)$	
전류스위치		$Y = 0$ if $u=0$ x_1 o.w. $U = MAX(x_2, 0)$	
		$Y = 0$ if $u=0$ x_1 o.w. $U = MIN(x_2, 0)$	

그림 1. 기본 전류모드 CMOS 논리회로

Fig. 1. Basic Current-Mode CMOS Logic elements.

서 0을 기준으로 절단된다. 절단된 값은 출력부에서 반대 부호로 바뀌고 상수에 의해 곱해진다.

④ 전류스위치(threshold element) : 제어 신호 값이 0 이 하이면 NMOS 소자에서 입력을 출력으로 연결시키고 0 이 상이면 PMOS에서 입력을 출력으로 연결시킨다.

2.2 전류모드 CMOS 회로를 이용한 다치 논리함수 합성 방법

단일변수 함수설계에서 다치 논리함수 합성기술로는 수학적 합성방법과 컴퓨터를 이용한 알고리즘 합성방법이 있다[1]. 본 논문에서는 컴퓨터를 이용한 알고리즘 합성 방법 중 비용-테이블 합성, 분해방법을 사용한다[6,7].

이 방법은 CCD 회로설계에 처음 사용하였으며[6] 다양한 용도에 응용이 가능한 방법으로써 비용-테이블에 포

함되는 함수들에 의해 전류모드 CMOS를 비롯한 다른 소자를 이용한 회로에도 적용할 수 있는 장점이 있다.

2.2.1 비용-테이블 방법

본 논문에서 주어진 4차 1변수 함수를 전류모드 CMOS 논리회로로 구현하면 그림 2와 같이 비용-테이블 방법으로 다차 논리회로로 설계가 가능하다. 그림 2는 입력 부, 부-함수 구현부와 출력부로 구성된다. 입력부는 입력변수 x를 P-형 미러를 사용하여 부-함수 실현부로 분산시킨다. 부-함수 구현부는 비용-테이블 함수들로 구성되며 제시된 부-함수들을 사용하여 구현한다. 이는 Q1, Q2, ..., Qn 으로 표기하였다. 출력부는 함수를 생성하는 합산기로 구성된다.

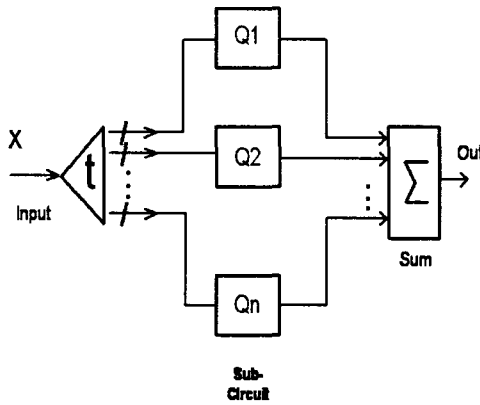


그림 2. 비용-테이블을 이용하여 1 변수 함수를 구현하는 CMCL회로

Fig. 2. The CMCL Circuit for Unary function with Cost-Table approach

본 논문에서 제안하는 부회로의 구조를 그림 3에 나타내었다.

① 그림 3의 부-회로 1은 논리동작이 파라미터 b0, b1, b2와 t1, t2에 의해 표현된다. 출력 값은 미러에 의해 절단되지만 부의 값이 되지는 않는다(N-형 미러). 동일하게 P-형 미러 에서는 양의 값이 되지 않는다.

$$f(x) = -b_2 \cdot BNDN[K + (-b_0 \cdot x)] : (N\text{-형 미러})$$

$$f(x) = -b_2 \cdot BNDN[K + (-b_0 \cdot x)] : (P\text{-형 미러})$$

$$\text{상수 } K = \begin{cases} -b & \text{if } t = p \\ b & \text{if } t = n \end{cases}$$

비용은 회로에 사용된 트랜지스터의 총수에 입력신호의 양을 더한 값이므로 $b + b_1 + b_2 + 1$ 이다.

② 두 미러의 직렬중속접속이 부-회로2와 같이 구현되며 얻어지는 함수의 범위는 step-up, step-down, 그리고 delta literal 등 많은 함수들을 포함한다.

$$f(x) = -b_4 \cdot BNDN[L + (-b_5 \cdot x) - b_2 \cdot BNDN(K + (-b_0 \cdot x))]$$

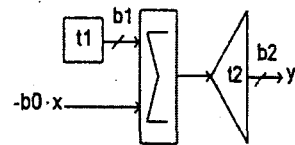
: N-형 미러

N-형 미러일 때

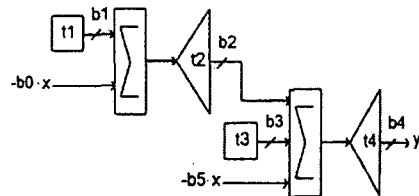
$$\text{상수 } K = \begin{cases} -b & \text{if } t = p \\ b & \text{if } t = n \end{cases}$$

전체 비용은 $b_0 + b_1 + b_2 + b_3 + b_4 + b_5 + 2$ 이다.

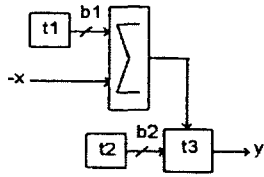
③ 부-회로 3과 부-회로 4는 레벨 검출기(threshold-detector)와 역 레벨 검출기(inverse threshold detector)로 동작한다. 부-회로 3은 그림 3과 같이 상수발생기와 패스-트랜지스터로 구현될 수 있다. 상수 발생기를 갖는 레벨 검출기는[8,9] CCD 기본회로를 이용하여 설계된바 있다. 부-회로3은 입력 x가 t1에서의 출력과 같거나 더 클 때 게이트 출력이 패스-트랜지스터 t3에 의해 상수 발생기 t2로 연결되어 출력부에서 상수 t2의 출력이 출력되고, 부-회로4는 입력 x가 t1에서 출력과 같거나 더 클 때 입력 x가 출력된다. 부회로3의 회로비용은 $2 + b_1 + b_2$ 이며, 부회로4의 회로비용은 $5 + b_1$ 이다. 역 레벨 검출기(또는 역 리터럴 함수 발생기)의 동작은 레벨 검출기의 역 동작이다. 이는 레벨 검출기를 수정함으로써 구현될 수 있다. 즉, 입력 x가 t1이 출력보다 작거나 같으면 상수발생기의 t2나 입력 x가 출력이 되고, 그렇지 않으면 출력은 0 이다.



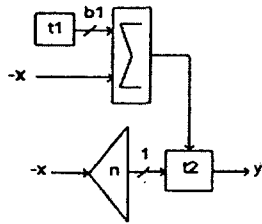
a) 부-회로 1



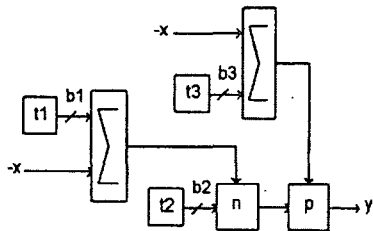
b) 부-회로 2



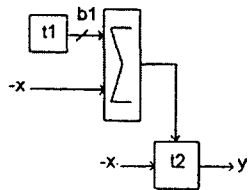
c) 부-회로 3



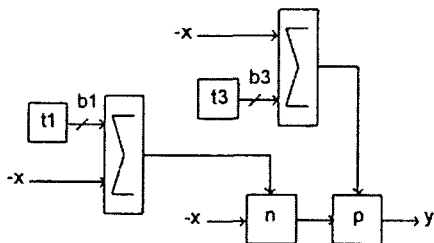
d) 부-회로 4



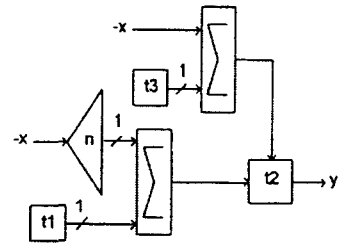
e) 부-회로 5



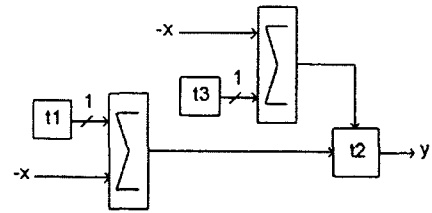
f) 부-회로 6



g) 부-회로 7



h) 부-회로 8



i) 부-회로 9

그림 3. 제안된 부-회로
Fig. 3. Proposed sub-circuits

$TD(m, x) =$ 상수 C 또는 입력 x , if $m < x$
0, Otherwise.

$ITD(m, x) =$ 상수 C 또는 입력 x , if $m > x$
0, Otherwise.

④ 위의 부회로 3을 수정한 회로가 부회로5 이다. 입력 x 가 $t1$ 과 $t3$ 출력의 사이 값이면 상수 $t2$ 또는 입력 x 가 출력되고, 그렇지 않으면 출력은 0 이다. 회로 비용은 $4+b1+b2+b3$ 이다. 부-회로5의 동작은 리터럴 함수 발생기와 유사하며, 다치 멀티플렉서인 UUPC(Universal Unary Programmable Circuit)나 T-gate, PLA를 구현하는데 유용하다.

$MTD(m1, m2, x) =$ 상수 C 또는 입력 x ,
if $m1 < x < m2$
0, Otherwise.

⑤ 부회로 6은 레벨 검출기, 역 레벨 검출기의 동작을 한다. 그리고 부호함수들을 도출하기 위해서 입력 $-x$ 를 미리 동작 없이 그대로 사용한다. 입력 x 가 $t1$ 에서의 출력보다 더 크거나 같으면 $-x$ 가 출력되고, 그렇지 않으면 출력은 0 이다. 부회로 7의 회로비용은 $3+b1$ 이다.

⑥ 부-회로 7은 수정 레벨 검출기(modified threshold

detector)의 동작을 한다. 부호함수의 출력을 위해 입력 $-x$ 를 그대로 사용한다. 입력 x 가 $t1$ 과 $t2$ 출력의 사이 값이면 $-x$ 가 그대로 출력되며, 그렇지 않으면 출력은 0 이다. 비용은 $5+b1+b3$ 이다.

⑦ 부-회로 8은 위의 부-회로들에서 얻을 수 없는 함수들을 구현할 수 있는데, 이는 입력 x 와 상수 $t1$ 의 합에 의해서이다. 이로 인해서 새로운 비 부호함수들과 부호함수의 생성이 가능하다. 비용은 7 이다.

⑧ 부-회로 9는 입력 $-x$ 와 상수발생기 출력 $t1$ 의 합이 $-x$ 와 $t3$ 의 합에 의해서 제어된다. 비용은 5이다. 제안된 부-회로들로부터 함수기능이 다양하고 비용감소가 큰 비용-테이블을 설계할 수 있다.

2.2.2 제안된 비용-테이블

제안된 부-회로들로부터 표 1의 비용-테이블을 얻을 수 있다. 이 비용-테이블로부터 기존의 방법보다 더 적은 회로비용을 갖는 함수를 구현할 수 있다. 모든 4치 함수들은 제안된 부-회로들을 그림 2의 회로구조에서 조합하여 구현할 수 있다.

2.3 검색 알고리즘

표 1의 비용-테이블로터 256개의 4치 함수들을 합성하기 위해 제안된 알고리즘은 아래와 같으며, 이 알고리즘은 비용-테이블의 부호, 비 부호함수들을 모두 고려하여 수행한다는 장점을 가지고 있다. 부호 부-함수는 부의 천이(Negative transition)를 갖는 함수를 구현할 수 있기 때문에 총 평균비용의 현저한 감소효과를 기대할 수 있다. 기 언급한 내용들을 요약하여 비용-테이블 함수를 검색하는 알고리즘은 다음과 같다.

단계 1. 비용-테이블에서 함수들과 그 비용들을 차례로 모두 읽어 들인다.

단계 2. 테이블에 있는 함수 모두에 대해 지정된 횟수만큼 반복적으로 수행을 계속하도록 설정한다. 함수들의 가항을 더하고 그 비용들도 모두 더하고, 부호함수도 알고리즘 수행에 포함한다.

단계 3. 가능한 모든 함수에 대해 검색하고, 해당 조건을 충족하는 함수들을 선택한다. 반복적으로 수행을 계속하면서, 더 낮은 비용을 갖는 동일함수가 도출될 때는 해당 함수에 그 비용을 새로 할당한다. (256개의 1변수 함수 모두에 대해서 수행한다.)

단계 4. 원하는 횟수만큼 모두 탐색하고 나면 결과를 출력하고 평균비용을 계산한다.

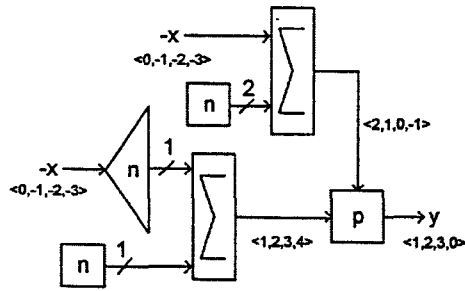
단계 5. 256개 1 변수 함수의 평균비용이 가장 낮은 비용-테이블을 선택하고, 함수 합성은 선택된 테이블을 기준으로 한다.

표 1. 제안된 비용-테이블
Table 1. Proposed cost table

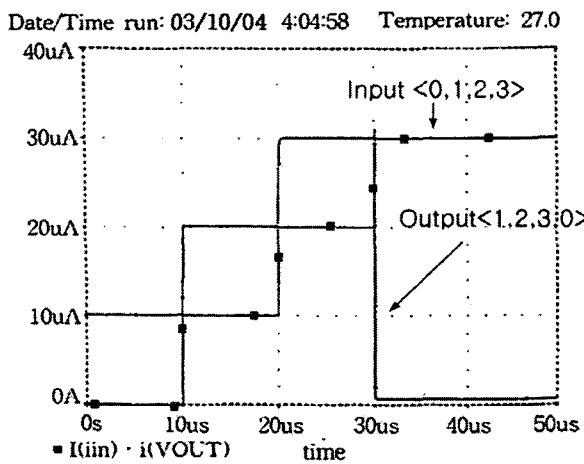
번호	부-함수	비용	35	0 1 1 0	8	70	0 0 -2 -2	6
1	-1 -1 -1 -1	1	36	0 2 0 0	8	71	0 0 -3 -3	7
2	1 1 1 1	1	37	0 0 -1 -3	8	72	0 0 0 -1	6
3	0 1 2 3	3	38	0 -3 0 0	9	73	0 0 0 -2	7
4	1 0 0 0	3	39	0 0 1 0	9	74	-1 0 0 0	3
5	-1 0 0 0	4	40	0 1 2 1	9	75	-2 0 0 0	4
6	0 0 1 2	4	41	0 2 2 0	9	76	-3 0 0 0	5
7	0 1 1 1	4	42	0 3 0 0	9	77	-1 -1 0 0	4
8	1 1 0 0	4	43	1 2 1 0	9	78	-2 -2 0 0	5
9	2 0 0 0	4	44	0 2 1 0	10	79	-3 -3 -3 0	6
10	-2 0 0 0	5	45	3 2 2 3	10	80	-1 -1 -1 0	5
11	-2 0 0 0	5	46	0 0 2 0	10	81	-2 -2 -2 0	6
12	0 -1 -2 -3	3	47	0 3 3 0	10	82	-3 -3 -3 0	7
13	0 0 0 1	5	48	3 2 1 2	10	83	-1 0 1 0	8
14	0 0 1 1	5	49	0 0 -3 0	11	84	-2 -1 0 1	7
15	0 2 2 2	5	50	0 0 3 0	11	85	-1 0 1 2	6
16	1 1 1 0	5	51	2 3 1 0	11	86	0 -1 0 1	8
17	2 2 0 0	5	52	0 3 2 1	12	87	1 0 -1 0	6
18	3 0 0 0	5	53	1 3 1 0	12	88	1 0 -1 -2	4
19	0 1 0 0	6	54	0 3 1 0	13	89	2 1 0 -1	5
20	0 0 -1 -2	6	55	0 1 3 0	14	90	0 1 0 -1	6
21	0 0 0 2	6	56	-3 0 0 0	6	91	-2 -3 0 0	6
22	0 0 2 2	6	57	-3 -2 -1 0	6	92	-1 -2 0 0	5
23	0 3 3 3	6	58	0 -1 -1 0	8	93	2 1 0 0	6
24	2 2 2 0	6	59	0 -2 -2 0	9	94	3 2 0 0	7
25	3 3 0 0	6	60	0 -3 -3 0	10	95	-1 -2 -3 0	6
26	-3 -1 0 0	7	61	0 0 -1 0	9	96	0 0 -1 -2	5
27	0 -1 0 0	7	62	0 0 -2 0	9	97	0 2 1 0	7
28	0 0 1 3	7	63	0 -1 -2 0	5	98	0 0 1 0	8
29	0 0 2 3	7	64	0 0 -2 -3	5	99	0 -2 -1 0	9
30	3 3 3 0	7	65	0 0 0 -3	6	100	-3 -2 0 0	9
31	0 -2 0 0	7	66	0 -1 -1 -1	4	101	1 2 0 0	7
32	0 0 0 3	7	67	0 -2 -2 -2	5	102	2 3 0 0	8
33	3 3 3 0	7	68	0 -3 -3 -3	6	103	1 2 3 0	8
34	0 -2 0 0	8	69	0 0 -1 -1	5	104	3 2 1 0	6

2.4 부-함수 시뮬레이션

본 논문에서 제시한 9개의 부-회로를 이용하여 입력 값 <0,1,2,3>에 대하여 4개의 모듈에 SPICE 시뮬레이션을 행하였다. 그림 4의 경우 입력은 X이며 -X로 표시된 부분은 부의 값을 나타내므로 <0,-1,-2,-3>이 된다.



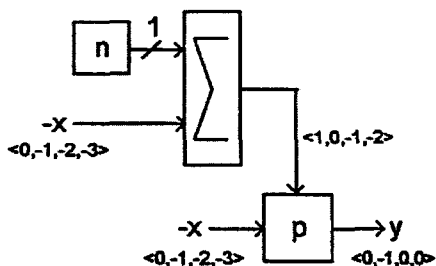
a) 입력 $\langle 0, 1, 2, 3 \rangle$ 출력 $\langle 1, 2, 3, 0 \rangle$을 갖는 부-함수



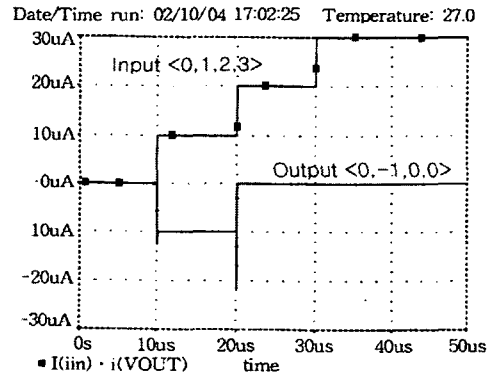
b) 부-함수 $\langle 1, 2, 3, 0 \rangle$ 시뮬레이션결과

그림 4. 입력 $\langle 0, 1, 2, 3 \rangle$에 대한 부-함수 $\langle 1, 2, 3, 0 \rangle$ 시뮬레이션 결과

Fig.4. Simulation result of sub-circuit $\langle 1, 2, 3, 0 \rangle$ for input $\langle 0, 1, 2, 3 \rangle$



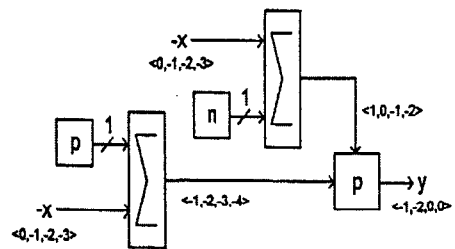
a) 입력 $\langle 0, 1, 2, 3 \rangle$ 출력 $\langle 0, -1, 0, 0 \rangle$을 갖는 부-함수



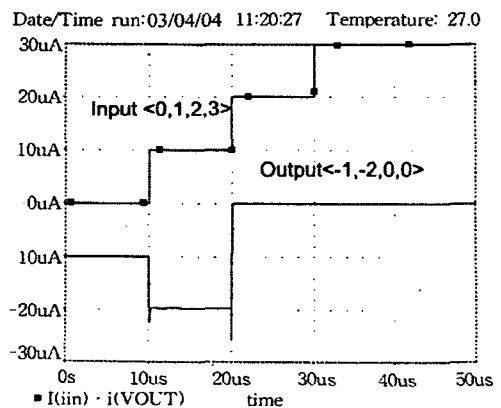
b) 부-함수 $\langle 0, -1, 0, 0 \rangle$ 시뮬레이션결과

그림 5. 입력 $\langle 0, 1, 2, 3 \rangle$에 대한 부-함수 $\langle 0, -1, 0, 0 \rangle$ 시뮬레이션 결과

Fig.5. Simulation result of sub-circuit $\langle 0, -1, 0, 0 \rangle$ for input $\langle 0, 1, 2, 3 \rangle$



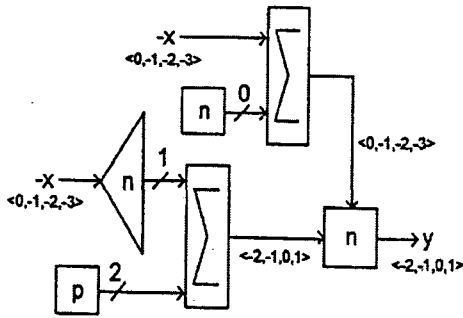
a) 입력 $\langle 0, 1, 2, 3 \rangle$ 출력 $\langle -1, -2, 0, 0 \rangle$을 갖는 부-함수



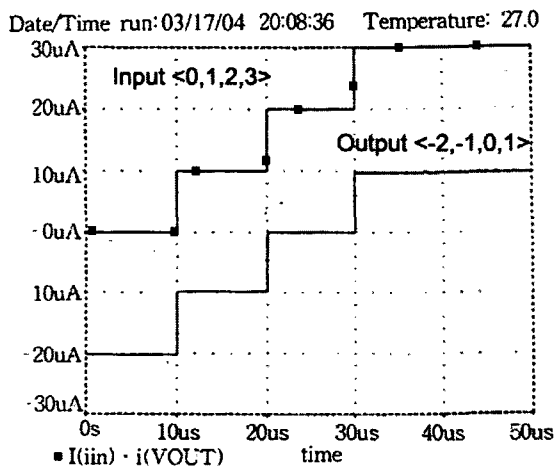
b) 부-함수 $\langle -1, -2, 0, 0 \rangle$ 시뮬레이션결과

그림 6. 입력 $\langle 0, 1, 2, 3 \rangle$에 대한 부-함수 $\langle -1, -2, 0, 0 \rangle$ 시뮬레이션 결과

Fig.6. Simulation result of sub-circuit $\langle -1, -2, 0, 0 \rangle$ for input $\langle 0, 1, 2, 3 \rangle$



a) 입력 <0,1,2,3>, 출력 <-2,-1,0,1>을 갖는 부-함수



b) 부-함수 <-2,-1,0,1> 시뮬레이션결과

그림 7. 입력 <0,1,2,3>에 대한 부-함수 <-2,-1,0,1> 시뮬레이션 결과

Fig.7. Simulation result of sub-circuit <-2,-1,0,1> for input <0,1,2,3>.

합산기를 거쳐 전류 스위치의 규칙에 따라 결과 값 <1,2,3,0> 값을 얻게 된다. 그림 5, 그림 6, 그림 7의 경우에도 동일한 동작원리에 의해 결과 값을 얻을 수 있다. 4개의 모듈 모두 동일한 입력 값을 인가하여 각각의 모듈 별로 PSPICE 시뮬레이션을 수행하였으며, 5μ CMOS 공정기술에 의해 파라미터를 결정한다. MOS의 회로 모델은 Level 3을 사용한다. 논리 값은 전류 값 0을 기준으로 10μA 단위로 논리 값을 정한다.

2.5 전류모드 CMOS Universal Unary Programmable Circuit (UUPC)

UUPC는 모든 1-변수 또는 다-변수 다차 논리함수를 생성할 수 있고, 규칙적인 구조로 인해 회로를 쉽게 구현할 수 있다. 또한, UUPC의 회로구조는 T-gate 와 같은

다차 논리소자로 쉽게 적용될 수 있다. 그러나 일반적인 회로구현이 매우 용이한 반면 비용이 많이 드는 단점이 있다. UUPC는 멀티플렉서와 유사하며, 한 개의 제어 입력 값이 나머지 r 개의 입력들이 출력으로 선택되는 것을 결정한다. 그림 8. a) 는 4차 UUPC를 나타낸다. 그림에서 XC는 4차 제어(선택)입력으로 <0,1,2,3>의 값들을 갖는다. XC = 0 일 때 입력 X0 가 선택되고, XC= 1 일 때 X1이 선택된다. 내부구조는 그림 8. b)와 같다.

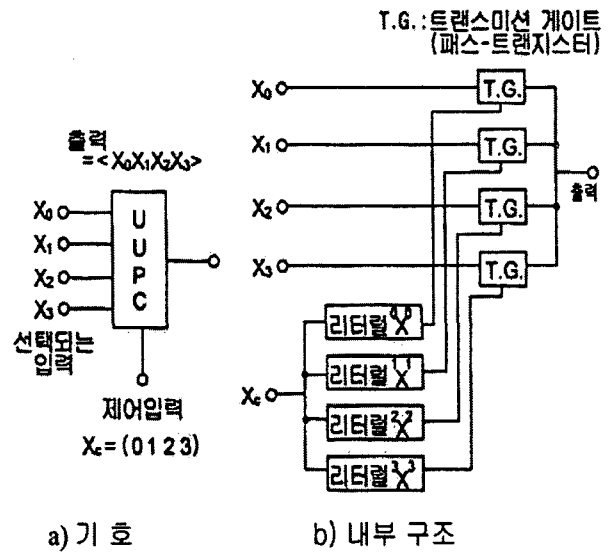


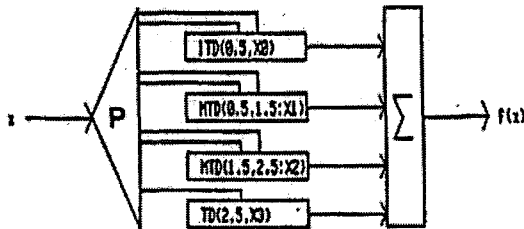
그림 8. UUPC(Universal Unary Programmable Circuit)
Fig. 8. UUPC(Universal Unary Programmable Circuit)

[정의]

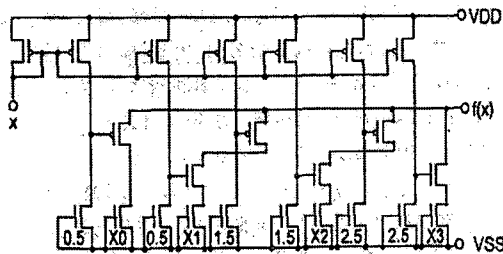
- TD(m,x) = 상수 C 또는 입력 x, m<x
0, Otherwise.
- MTD(m1,m2,x) = 상수 C 또는 입력 x, m1<x<m2 이면
0, Otherwise
- ITD(m, x) = 상수 C 또는 입력 x, 만일 x<m 이면
0, Otherwise

본 논문에서 제안한 UUPC의 회로는 그림 9와 같다. 안정된 전류공급을 위하여 Depletion mode MOSFET를 전류발생기로 사용하였으며 출력되는 전류의 상대 레벨 값을 함께 표기하였다. x=0 이면 역 레벨 검출기인 ITD(0.5, x0)가 동작하여 출력 f(x)는 x0가 된다. 출력 f(x)는 x=1일 때 x1, x = 2 일 때 x2, x = 3 일 때 x3이다. 함수 f(x)에 대한 UUPC의 전체 비용은 Q(f(x)) = 24+x0+x1+x2+x4 이다. 제어 입력 x의 레벨을 검출하는 상수 발생기들이 중복 사용됨을 알 수 있는데, 상수발생기의 출력 K=0.5가 ITD(0.5, x0)와 MTD(0.5, 1.5, x1)에

서 $x=0$ 와 $x=1$ 의 레벨 값을 검출하기 위해 사용되며, $K=1.5$, $K=2.5$ 도 각각의 제어 입력 값의 레벨을 검출하기 위해 사용된다. 따라서, 회로가 중복되는 부분을 생략하도록 설계한다면 UUPC의 구현비용은 상당히 감소할 수 있다.



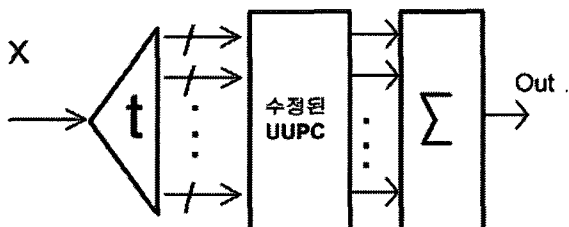
a) UUPC 기호



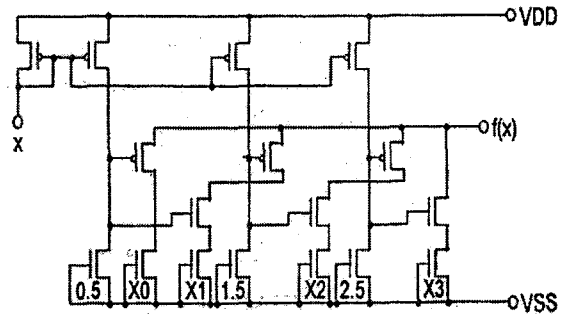
b) UUPC 회로도

그림 9. 제안된 UUPC 회로
Fig. 9. Proposed circuit of UUPC

본 논문에서 제안한 방법을 적용하면 그림 9의 회로는 그림 10의 회로와 같이 개선되고, 전체비용은 $Q(f(x))=16.5+x_0+x_1+x_2+x_3$ 가 된다. 전류모드 CMOS 논리회로에서 레벨검출은 그림 1의 기본회로들로 쉽게 구현할 수 있으며, CCD를 사용한 UUPC(UICPC)에서의 레벨검출은 전류모드 CMOS UUPC보다 상대적으로 더 복잡하며 더 높은 비용이 필요하다.



a) 개선된 UUPC 기호



b) 개선된 UUPC 회로

그림 10. 개선된 UUPC
Fig. 10. Modified UUPC

2.6 비교 및 검토

본 논문에서 제안한 부-회로를 이용하여 다치 논리함수를 설계한 결과, 기존의 방법[9]보다 256개의 1변수 함수 중 37.1%에 해당되는 95개의 함수의 비용이 절감되었으며, 평균비용 면에서 7.7% 감소되었다.

표 2. 평균비용 비교

Table 2. Compared Table for average cost

	Y.H.Chang & J.T.Butler	M.H.Abd-El-Barr & M.I.Mahroos	본 논문
평균 비용	12.12	10.37	9.63

제안된 부-회로와 비용 최소화 알고리즘을 통하여 단위 소자의 수를 줄임으로써 내부결선 감소에 의한 칩 밀도 감소, 이에 따른 처리속도 향상과 전력소모 감소, 레벨 저하 방지효과와 회로 동작의 안정성 향상을 기대할 수 있다. 표 2는 기존 방법과의 평균비용을 비교한 결과이며, 그림 11은 256개 1변수 함수의 각 함수비용과 개수의 상대적인 비교를 통하여 비용 절감효과를 나타내었다. 본 논문의 경우 함수비용이 낮은 부분의 함수 개수가 기존의 방법보다 상대적으로 많으므로 전체 회로비용 면에서 본 논문의 회로 비용 절감효과를 알 수 있다.

III. 결론

본 논문에서 제안한 부-회로들을 사용한 비용-테이블에서, 전류모드 CMOS 논리회로에서 낮은 비용의 부호부-함수는 더 적은 비용으로 비 부호함수를 구현할 수 있고, 비용-테이블 내에 비 부호함수, 부호 함수가 함께 포함되므로 평균 구현비용이 개선되었다.

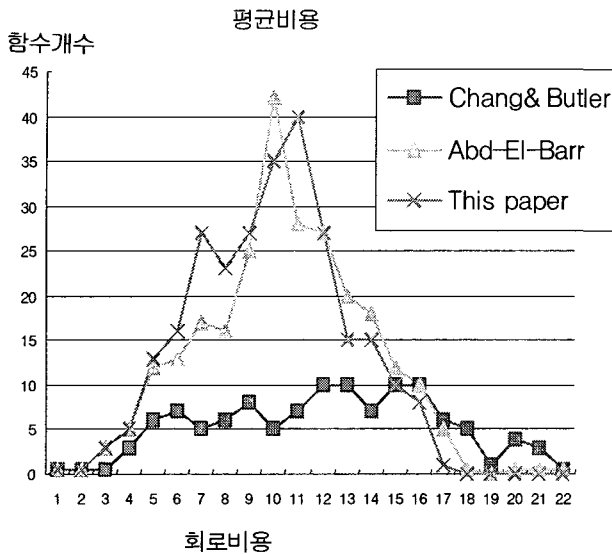


그림 11. 함수비용-개수 비교

Fig. 11. Comparisons of cost-number of functions

256 개의 4치 1변수 함수의 경우 제안된 방법은 기존의 방법에 비해 전체 1 변수함수들 중 37.1 %의 구현비용을 감소시키고, 전체 구현비용을 최소 7.7% 감소시킬 수 있었다. 또, 제안한 부-회로들을 이용하여 설계한 다치 멀티플렉서인 UUPC의 경우 전류공급기의 불필요한 중복사용을 개선함으로써 회로 비용 면에서 최대 31.25%, 최소 20.83% 가 절감되어 평균 25%의 회로비용 감소효과가 얻어진다. 향후과제로는 UUPC를 개선한 새로운 종류의 모듈을 개발하여 좀더 적은 비용으로 다치 논리 회로를 설계하기 위한 연구가 필요하다.

참고 문헌

[1] S.P Onneweer, H.T.Kerhoff and J.T,Butler "Structural computer-aided design of current-mode CMOS logic circuits," Proc. of ther 18th ISMVL, pp.21-30, may 1988.
 [2] F.J.Pelayo, A.Prieto, A.Lloris and J.Ortega."CMOS Current-mode Multivalued PLAs," IEEE Trans. Circuit and Systems, 1991, vol.38, No.4 pp.434-441.
 [3]T.S.Kawahito, M.Kameyama and T.Higuchi "VLSI-oriented bi-directional current-mode arithmetic circuits based on the radix-4 signed digit number system." Proc. of the 16th Inter. Symp. on Multiple-Valud Logic. pp.70-77, May. 1986.
 [4] T.yamakawa."CMOS multivalued circuits in hybrid mode,"Proc. of the 15th ISMVL.,pp144-151, May, 1985.
 [5] M.H.Abd-El-Barr and M.I.Mahroos, "On the

Synthesis of MVL Functions for Current-Mode CMOS Circuits Implementation," Proc. of the 22th ISMVL, pp.221-228, 1992.

[6] H.A.J.Robroek, "The Synthesis of MVL-CCD circuits," M.Sc. Report no.12.3936, Twente University of Technology, Enschede, The Netherlands, Dec. 1981.
 [7] Y.H.Chang and J.T.Butler, "The Design of Current Mode CMOS Multiple-Valued Circuits," Proc. of the 21 th ISMVL, pp.130-138, May, 1991.
 [8] M.Kameyama, T.Sekibe and T.Higuchi, "Design of highly parallel residue arithmetic circuits based on multiple-valued bidirectional current mode MOS technology," Proc. of the 18th ISMVL, pp.6-13, May, 1988.
 [9] M. H. Abd-El-Barr and A. Al-Mutawwa, "Cost -Analysis of 4-valued Uanry Functions Implemented using Current-Mode CMOS Circuits," Proc. of the 30th ISMVL, pp.215-220, May, 2000.
 [10] 최재석 외 3, "코드할당에 의한 다치 논리함수의 모듈러 함수 분해에 관한 연구" 대한전자공학회 논문지-C, Vol.35 No.7 pp.78-91, 1998.



최재석(Jai-Sock Choi)

1988 인하대학교 전자공학과 공학사
 1990 인하대학교 전자공학과 (공학석사)
 1997 인하대학교 전자공학과 (공학박사)
 1990~1995 (주)유니온 시스템 개발 연구소

1996.09~1998.06 우진전자통신(주) 연구소

1998.10~1999.02 한국EIT(주) 연구소장

1999~ 현재 인덕대학 정보메카트로닉스과 조교수

관심분야 : 다치 논리회로설계, 스위칭이론,

마이크로프로세서응용, 시스템프로그래밍