

論文94-31B-1-7

3치 범용 논리 모듈 U_h 에 의한 빠른 논리 합성

(Fast Synthesis based on Ternary Universal Logic Module U_h)

金榮建*, 金鍾五**, 金興壽*

(Young Gun Kim and Jong O Kim and Heung Soo Kim)

要約

ULM U_h 를 이용한 논리함수의 합성은 주어진 함수의 canonic Reed-Muller 확장계수에 의거하여 구성된다. 본 논문은 3치 ULM U_h 를 이용한 3치 함수의 빠른 합성 방법을 제안하였다. 3치 n -변수 함수의 합성시 본 논문에서 제안된 회로비용 계산과 합성 방법에 의하여 ULM U_h 의 수가 최소화 되도록 제어 입력 변수의 순서가 결정된다. 따라서 ULM U_h 를 이용한 3치 함수 합성시 최적의 회로 실현이 가능하게 되며 ULM U_h 를 이용한 3치 함수의 합성에도 적용될 수 있다. 모든 제어 입력 변수의 순서를 선택하는데 필요한 검색 횟수는 $(n+2)(n-1)/2$ 가 된다.

Abstract

The logic function synthesis using ULM U_h is constructed based on canonic Reed-Muller expansion coefficient for a given function. This paper proposes the fast synthesis algorithm using ULM U_h for ternary function. By using circuit cost and synthesis method of proposed in this paper, order of control input variable with minimum number of ULM U_h can be decided in the synthesis of n -variable ternary function. Accordingly, this method enables to optimum circuit realization for ternary function synthesis using ULM U_h and can be applied to ternary function synthesis using ULM U_h . The complexity of search for select the order of all control input variables is $(n+2)(n-1)/2$.

1. 서론

다치논리가 조합회로의 설계에 적용되기 위하여는

*正會員, 仁荷大學校 電子工學科
(Dept. of Elec. Eng., Inha Univ.)

**正會員, 東洋工業專門大學 電子科
(Dept. of Elec. Eng., Dongyang Tech.
College)

接受日字 : 1993年 5月 21日

함수적 완전성, 유연성, 회로의 실현성을 모두 만족하여야 한다. Reed-Muller 대수는 이러한 성질을 모두 만족하므로 다치논리에서 Reed-Muller 확장에 관한 연구가 꾸준히 진행되고 있다. 이 분야에서는 다치논리의 함수적 표현, 함수의 최소화, 합성 방법 등이 연구되고 있다. 현재 논리 설계 방법에서는 모듈성이 상당히 강조되고 있는 추세이므로 다치논리 설계에서도 범용 모듈을 이용하여 간단하고 규칙적인 형태를 갖는 회로의 실현에 연구가 집중되고 있다. 다치논리의 대표적인 ULM(Universal Logic

Module)으로는 T-gate와 U_r 가 있으며 3차 ULM 중 하나인 U_r 는 3차 canonic Reed-Muller 확장에 의거하여 설계되었기 때문에 모든 3차 함수는 ULM U_r 로 실현이 가능하게 된다. 그러나 U_r 는 기본 모듈 U_h 로 분해가 가능하며 U_r 는 2개의 U_h 또는 2개 이하의 U_h 로 실현될 수 있으므로 3차 함수의 합성시 U_r 보다 U_h 를 이용한 합성 방법이 효율적이다.^[2,3]

U_r 와 U_h 의 표현식은 다음과 같으며 각각의 기호를 그림 1에 보였다.

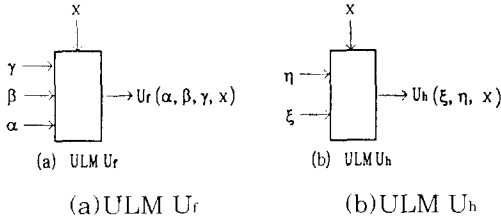


그림 1. 3차 ULM U_r 와 U_h 의 기호

Fig. 1. Symbol of ternary ULM U_r and U_h .

기본 모듈을 이용한 회로의 합성에는 나무구조에 의한 설계 방법이 모든 스위칭 함수에 효과적으로 적용되며^[4], 이러한 형태의 최적 합성은 각단의 제어 입력변수의 순서 할당 선택에 의하여 결정된다.^[1,3,4] ULM을 이용한 3차 함수의 합성에 대하여 몇가지 방법이 제안되었으나^[1,3,5] 이 방법들에는 약간의 결점을 내포하고 있다. Tokmen과 Hurst의 방법은 각각의 제어 입력 변수에 대한 Reed-Muller 확장 계수를 구해야 하는 번거로움과 최적의 입력 제어 변수 할당이 어려우며, Hu와 Wu의 방법은 Reed-Muller 확장 계수 map에 의존하므로 변수의 수가 3 이상일 경우 부적합 하다. Fei와 Zhuang의 방법은 위의 결점을 해결하였으나 U_r 만을 고려한 방법이므로 U_h 를 이용할 경우에는 최적의 합성이 어렵게 된다. 본 논문에서는 이러한 결점을 모두 해결할 수 있으며 U_r 와 U_h 에 모두 적용될 수 있는 합성 방법을 제안하였다. 2장에서는 n -변수 3차함수가 ULM U_r 또는 U_h 로 실현될 수 있음을 보였으며, 3장에서는 회로의 비용을 계산하기 위하여 Reed-Muller 확장계수로 부터 유도되는 비용행렬(cost matrix)을 소개하였다. 제 4장에서는 최적의 회로비용을 계산하기 위한 조건과 조건에 따르는 제약 사항을 제시하고 증명하였으며 이것을 근거로 회로의 비용을 계산하였다. 5장에서는 합성 방법을 서술하고 이에 대한 적용 예를 들어 기존의 방법들과 비교하므로써 본 방법의 타당성을 보였다.

II. ULM을 이용한 3차 함수의 합성

P 차 논리 시스템에서 n 변수 함수의 Reed-Muller 확장은 식(1)과 같이 표현된다.

$$f(X_1, X_2, \dots, X_n) = \sum_{i=0}^{p-1} C_i \prod_{j=1}^n X_j^{i_j} \quad (1)$$

여기서 $C_i \in GF(p)$ 이고, X_j 는 변수 X_j 의 i 번째 멱승을 나타내며, 모든 연산은 $GF(p)$ 이다. 식(1)에 의하여 임의의 n 변수 3차 함수의 Reed-Muller 확장은

$$f(X_1, X_2, \dots, X_n) = C_0 X_1^0 X_2^0 \dots X_n^0 \oplus C_1 X_1^0 X_2^0 \dots X_n^1 \oplus \dots \oplus C_j X_1^{e_1} X_2^{e_2} \dots X_n^{e_n} \oplus \dots \oplus C_{3^n-1} X_1^2 X_2^2 \dots X_n^2 \quad (2)$$

와 같이 된다.

여기서 $X_1^0 = 1, X_1^1 = X_1, X_1^2 = X_1 \cdot X_1 (i=1, 2, \dots, n,)$

$$C_j, e_j \in \{0, 1, 2\} \quad (j=0, 1, 2, \dots, 3^n-1)$$

$$(j)_{10} = (e_1 e_2 e_3 \dots e_n)_3$$

식(2)는 식(3), (4)와 같이 분해하여 다시 쓸 수 있다.

$$f(X_1, X_2, \dots, X_n) = (C_0 \oplus C_1 X_n \oplus C_2 X_n^2) \oplus (C_3 \oplus C_4 X_n \oplus C_5 X_n^2) X_{n-1} \oplus \dots \oplus (C_{3^n-3} \oplus C_{3^n-2} X_n \oplus C_{3^n-1} X_n^2) X_1^2 X_2^2 \dots X_{n-1} \quad (3)$$

$$= [C_0 \oplus (C_1 \oplus C_2 X_n) X_n] \oplus [C_3 \oplus (C_4 \oplus C_5 X_n) X_n] X_{n-1} \oplus \dots \oplus [C_{3^n-3} \oplus (C_{3^n-2} \oplus C_{3^n-1} X_n) X_n] X_1^2 X_2^2 \dots X_{n-1} \quad (4)$$

식(3), (4)로부터 n 변수 3차 함수는 U_r 와 U_h 로 실현될 수 있음을 확인 할 수있다. U_h 모듈을 이용하여 n 단계의 나무구조로 3차 함수를 실현 할 때 필요한 U_h 모듈의 최대 수는 $2 \sum_{i=1}^n 3^{i-1} = (3^n-1)$ 이 된다.

III. RM계수에 의한 비용행렬

[정의 1] n 단계로 구성된 나무구조 합성에서 임의의 i 번째 단계의 ULM 모듈출력을 $g(i, j)$ ($i=1, 2, \dots, n; j=0, 1, 2, \dots, 3^{n-1}$)로 나타내며, $g(0, j)=C_j$ 이다.

식(4)는 X_n 이 첫번째 단계의 제어 변수일 경우에 해당되며, 정의 1에 의하여 다음과 같이 표현될 수 있다.

$$f(X_1, X_2, \dots, X_n) = g(1, 0) \oplus g(1, 1) X_{n-1} \oplus g(1, 2) X_{n-1}^2 \oplus \dots \oplus g(1, 3^{n-1}-1) X_1^2 X_2^2 \dots X_{n-1} \quad (5)$$

식 (4)와 (5)로 부터 다음의 식이 유도된다.

$$\begin{aligned}
 g(1,0) &= C_0 \oplus C_1 X_n \oplus C_2 X_n^2 \\
 g(1,1) &= C_3 \oplus C_4 X_n \oplus C_5 X_n^2 \\
 g(1,3^{n-1}-1) &= C_{3^{n-3}} \oplus C_{3^{n-2}} X_n \oplus C_{3^{n-2}} X_n^2
 \end{aligned} \tag{6}$$

회로의 비용을 계산하기 위하여 식(6)의 RM계수를 식(7)과 같은 행렬로 나타내고 이것을 비용행렬(cost matrix)^[5]이라 한다.

$$[g(1,0)g(1,1)\cdots g(1,3^{n-1}-1)] = [1 X_n X_n^2] \begin{bmatrix} C_0 & C_3 & \cdots & C_{3^{n-1}} \\ C_1 & C_4 & \cdots & C_{3^{n-2}} \\ C_2 & C_5 & \cdots & C_{3^{n-3}} \end{bmatrix} \tag{7}$$

일반화된 비용행렬을 구하기 위하여 임의의 변수 X_i 가 첫 단계의 제어 입력 변수라 가정하면 식(7)은 다음과 같이 표현된다.

$$\begin{aligned}
 & [g_i(1,0) g_i(1,1) \cdots g_i(1,3^{n-1}-1)] \\
 &= [1 X_i X_i^2] \begin{bmatrix} g(0, L_0) & g(0, L_1) & \cdots & g(0, L_k) & \cdots & g(0, L_{3^{n-1}-1}) \\ g(0, M_0) & g(0, M_1) & \cdots & g(0, M_k) & \cdots & g(0, M_{3^{n-1}-1}) \\ g(0, N_0) & g(0, N_1) & \cdots & g(0, N_k) & \cdots & g(0, N_{3^{n-1}-1}) \end{bmatrix} \\
 &= [1 X_i X_i^2] [G_i(0)] \tag{8}
 \end{aligned}$$

여기서 $(k)_{10} = (e_1 e_2 e_3 \cdots e_{i-1} e_{i+1} \cdots e_n)_3$
 $(L_k)_{10} = (e_1 e_2 e_3 \cdots e_{i-1} 0 e_{i+1} \cdots e_n)_3$
 $(M_k)_{10} = (e_1 e_2 e_3 \cdots e_{i-1} 1 e_{i+1} \cdots e_n)_3$
 $(N_k)_{10} = (e_1 e_2 e_3 \cdots e_{i-1} 2 e_{i+1} \cdots e_n)_3$ 이다.

만일 임의의 변수 X_h 가 2번째 단계의 제어 입력 변수라 가정하면 식(7)은 다음과 같이 표현된다.

$$\begin{aligned}
 & [g_h(2,0)g_h(2,1)\cdots g_h(2,3^{n-2}-1)] \\
 &= [1 X_h X_h^2] \begin{bmatrix} g(1, L_0) & g(1, L_1) & \cdots & g(1, L_k) & \cdots & g(1, L_{3^{n-1}-1}) \\ g(1, M_0) & g(1, M_1) & \cdots & g(1, M_k) & \cdots & g(1, M_{3^{n-1}-1}) \\ g(1, N_0) & g(1, N_1) & \cdots & g(1, N_k) & \cdots & g(1, N_{3^{n-1}-1}) \end{bmatrix} \\
 &= [1 X_h X_h^2] [G_h(0)] \tag{9}
 \end{aligned}$$

여기서 $(k)_{10} = (e_1 e_2 e_3 \cdots e_{h-1} e_{h+1} \cdots e_{i-1} e_{i+1} \cdots e_n)_3$
 $(L_k)_{10} = (e_1 e_2 e_3 \cdots e_{h-1} 0 e_{h+1} \cdots e_{i-1} e_{i+1} \cdots e_n)_3$
 $(M_k)_{10} = (e_1 e_2 e_3 \cdots e_{h-1} 1 e_{h+1} \cdots e_{i-1} e_{i+1} \cdots e_n)_3$
 $(N_k)_{10} = (e_1 e_2 e_3 \cdots e_{h-1} 2 e_{h+1} \cdots e_{i-1} e_{i+1} \cdots e_n)_3$ 이다.

또한 임의의 변수 X_w 가 i 번째 단계의 제어 입력 변수라 가정하면 식(8), (9)와 같이 반복적인 방법으로 다음의 일반화된 비용행렬의 식을 얻을 수 있다.

$$[g_w(i,0)g_w(i,1)\cdots g_w(i,3^{n-i}-1)]$$

$$\begin{aligned}
 &= [1 X_w X_w^2] \begin{bmatrix} g(i-1, L_0) & g(i-1, L_1) & \cdots & g(i-1, L_k) & \cdots & g(i-1, L_{3^{n-i}-1}) \\ g(i-1, M_0) & g(i-1, M_1) & \cdots & g(i-1, M_k) & \cdots & g(i-1, M_{3^{n-i}-1}) \\ g(i-1, N_0) & g(i-1, N_1) & \cdots & g(i-1, N_k) & \cdots & g(i-1, N_{3^{n-i}-1}) \end{bmatrix} \\
 &= [1 X_w X_w^2] [G_w(i-1)] \tag{10}
 \end{aligned}$$

[정의 2] $g_w(i,0), g_w(i,1), \dots, g_w(i,3^{n-i}-1)$ 이 i 번째 단계의 함수라면 이것을 실현하기 위한 회로비용은 비용행렬 $[G_w(i-1)]$ 로 부터 구하여진다.

IV. 회로의 비용

[정의 3] 단일변수 3치 함수는 27가지 다른 형태의 함수가 존재하며 이를 실현하는 U_h 모듈의 수는 0, 1, 2개로 구분될 수 있으므로^[2] 각각의 함수를 실현하는데 소요되는 U_h 모듈의 수에 따라 다음과 같이 3개의 그룹으로 분류한다.

- {G₀} = {(- 0 0) (0 1 0)}
 - {G₁} = {(- 0 1) (- 2 0) (1 1 0) (2 1 0) (0 1 1)}
 - {G₂} = {G₀}와 {G₁}을 제외한 나머지
- 여기서 - 는 don't care이다.

[정의 4] 비용행렬 $[G(i-1)]$ 에서 3개의 계수로 구성된 임의의 두개의 열벡터를 V_r 와 V_r' 라 하고, V_r' 에서 U_h 형태로 분해하여 2개의 계수로 구성된 열벡터 V_{h1} 과 V_{h2} 를 다음과 같이 정의한다.

$$\begin{aligned}
 V_r &= [g(i-1, L_k), g(i-1, M_k), g(i-1, N_k)] \\
 V_r' &= [g'(i-1, L_k), g'(i-1, M_k), g'(i-1, N_k)] \\
 V_{h1} &= [g'(i-1, L_k), g'(i-1, M_k)] \\
 V_{h2} &= [g'(i-1, M_k), g'(i-1, N_k)]
 \end{aligned}$$

[정리 1] V_{h1} 이 V_r 의 부분집합이 될 필요충분 조건은 $g'(i-1, L_k) = g(i-1, M_k), g'(i-1, M_k) = g(i-1, N_k), g'(i-1, N_k) = 0$ 이다.

증명 : V_{h1} 은 2개의 열벡터 $g'(i-1, L_k)$ 와 $g'(i-1, M_k)$ 만으로 구성되므로 이것에 대응되는 열벡터 V_r 는 다음과 같다.

- (1) $g'(i-1, L_k) = g(i-1, L_k), g'(i-1, M_k) = g(i-1, M_k)$ 인 경우
 - ① $g'(i-1, N_k) = 0$ 이면 $U_r' = g(i-1, L_k) \oplus g(i-1, M_k) X \notin U_r$
 - ② $g'(i-1, N_k) \neq 0$ 이면 $U_r' = g(i-1, L_k) \oplus g(i-1, M_k) \oplus g'(i-1, N_k) X \notin U_r$
- (2) $g'(i-1, L_k) = g(i-1, M_k), g'(i-1, M_k) = g(i-1, N_k)$ 인 경우
 - ① $g'(i-1, N_k) = 0$ 이면 $U_r' = g(i-1, M_k) \oplus g$

$$(i-1, N_k)X \in U_r$$

- ② $g'(i-1, N_k) \neq 0$ 이면 $U_r' = g(i-1, M_k) \oplus (g(i-1, N_k) \oplus g'(i-1, N_k)X)X \notin U_r$ 이므로 V_{h1} 이 V_r 의 부분집합이 될 필요충분 조건은 $g'(i-1, L_k) = g(i-1, M_k)$, $g'(i-1, M_k) = g(i-1, N_k)$, $g'(i-1, N_k) = 0$ 이다.

[정리 2] V_{h2} 가 V_r 의 부분집합이 될 필요충분 조건은 $g'(i-1, M_k) = g(i-1, M_k)$, $g'(i-1, N_k) = g(i-1, N_k)$, $g'(i-1, L_k) = \text{don't care}$ 이다.

증 명 : V_{h2} 는 2개의 열벡터 $g'(i-1, M_k)$ 와 $g'(i-1, N_k)$ 만으로 구성되므로 이것에 대응되는 열벡터 V_r 는 다음과 같다.

- (1) $g'(i-1, M_k) = g(i-1, L_k)$, $g'(i-1, N_k) = g(i-1, M_k)$ 인 경우
- ① $g'(i-1, L_k) = 0$ 이면 $U_r' = (g(i-1, L_k) \oplus g(i-1, M_k)X)X \notin U_r$
- ② $g'(i-1, L_k) \neq 0$ 이면 $U_r' = g'(i-1, L_k) \oplus (g(i-1, L_k) \oplus g(i-1, M_k)X)X \notin U_r$
- (2) $g'(i-1, M_k) \neq g(i-1, M_k)$, $g'(i-1, N_k) = g(i-1, N_k)$ 인 경우
- ① $g'(i-1, L_k) = 0$ 이면 $U_r' = (g(i-1, M_k) \oplus g(i-1, N_k)X)X \in U_r$
- ② $g'(i-1, L_k) \neq 0$ 이면 $U_r' = g'(i-1, L_k) \oplus (g(i-1, M_k) \oplus g(i-1, N_k)X)X \in U_r$ 이므로 V_{h2} 가 V_r 의 부분집합이 될 필요충분 조건은 $g'(i-1, M_k) = g(i-1, M_k)$, $g'(i-1, N_k) = g(i-1, N_k)$, $g'(i-1, L_k) = \text{don't care}$ 이다.

따라서 비용행렬 $[G(i-1)]$ 에 의하여 결정되는 함수 $g(i, k)$ 를 실현하는데 있어서 제한 조건에 따라 생략될 수 있는 U_h 모듈의 수는 아래와 같이 5가지가 존재한다.

- (1) $V_r \in \{G_0\}$ 이면 2개의 U_h 모듈이 생략된다.
- (2) $V_r \in \{G_1\}$ 이면 1개의 U_h 모듈이 생략된다.
- (3) $V_r \in \{G_1\}$ 이고 동일한 두 V_r 는 1개의 U_h 모듈이 생략된다.
- (4) $V_r \in \{G_2\}$ 이고 동일한 두 V_r 는 2개의 U_h 모듈이 생략된다.
- (5) V_{h1} 또는 V_{h2} 가 동일한 경우 1개의 U_h 모듈이 생략된다.

[정의 5] 함수 $g(i, k)$ 를 실현하는데 소요되는 U_h 모듈의 수 $T(g(i))$ 를 회로비용으로 정의하고, 최대로 소요되는 U_h 모듈의 수와 생략할 수 있는 U_h 모

듈의 수를 각각 $M(g(i))$, $D(g(i))$ 로 표기한다.

위의 제약 조건에 의하여 $(g_w(i, 0) \ g_w(i, 1) \ g_w(i, 3^{n-1}-1))$ 을 실현할 때 생략되는 U_h 모듈의 수 $D(g(i))$ 는 다음 식에 의하여 계산된다.

$$D(g(i)) = \sum_k (g(i, k)) = \sum_k T_k + \sum_k O_k + S1 + S2 + S3 \quad (11)$$

$$\text{여기서 } T_k = \begin{cases} 2 & \text{if } V_r \in \{G_0\} \\ 0 & \text{otherwise} \end{cases}$$

$$O_k = \begin{cases} 1 & \text{if } V_r \in \{G_0\} \\ 0 & \text{otherwise} \end{cases}$$

S1: $V_r \in \{G_2\}$ 인 동일한 열벡터가 2개 이상 존재할 때 생략되는 U_h 모듈의 수.

S2: $V_r \in \{G_1\}$ 인 동일한 열벡터가 2개 이상 존재할 때 생략되는 U_h 모듈의 수.

S3: V_{h1} 또는 V_{h2} 인 동일한 열벡터가 2개 이상 존재할 때 생략되는 U_h 모듈의 수이며 이때 조건 (1), (3), (4)항에 해당되는 열벡터는 제외된다.

또한 $M(g(i))$ 와 $T(g(i))$ 는 다음과 같다.

$$M(g(i)) = 2 \cdot 3^{n-1} \quad (i=1, 2, 3, \dots, n) \quad (12)$$

$$T(g(i)) = M(g(i)) - D(g(i)) \quad (13)$$

식(12)에서 $M(g(i))$ 는 변수의 수 n 과 i 에 따라 항상 일정하므로 $T(g(i))$ 는 $D(g(i))$ 에 종속된다. 따라서 $D(g(i))$ 가 최대일때 회로비용이 최소가 되며 $T(g(i))$ 의 값이 i 번째 단계를 실현하는 U_h 모듈의 수가 된다.

V. 합성 알고리즘

ULM U_h 를 사용한 n 변수 3차 함수를 실현하는 회로의 비용은 각 단계에서 제어 변수의 선택에 따라 결정되며, 결정된 회로의 비용은 이후 단계의 제어 변수 순서에 관계없이 일정하다.^[5] 이러한 결과를 이용한 최소의 회로비용을 갖는 3차 함수의 논리 합성 과정은 다음과 같다.

- (1) X_1, X_2, \dots, X_n 을 차례로 선택하여 비용행렬 $[G(0)]$ 로 부터 각각의 변수에 대한 회로비용을 식(11)을 이용하여 구한 후 $D(g(1))$ 이 최대인 변수 X_i 를 첫 단계 제어 변수로 선택한다. 만일 $D(g(i))$ 가 같은 값을 갖는 경우 제어 변수 선택의 우선순위는 $\sum T_i, \sum O_i, S1, S2, S3$ 순 이고 모든 단계에 적용된다.

- (2) 식(10)에 의하여 첫 단계 함수 $g_i(1, 0) \ g_i(1, 1) \ g_i(1, 3^{n-1}-1)$ 를 구한다.

(3) $X_1, X_2, \dots, X_{i-1}, X_{i+1}, \dots, X_n$ 을 차례로 선택하여 비용행렬 $[G(1)]$ 로 부터 각각의 변수에 대한 최로 비용을 식(11)을 이용하여 구한 후 $D(g(2))$ 가 최대인 변수 X_n 를 두번째 단계 제어 변수로 선택한다.

(4) 식(10)에 의하여 두번째 단계 함수 $g_n(2,0)$ $g_n(2,1)$ $g_i(2,3^{n-2}-1)$ 를 구한다.

(5) $(n-1)$ 번째 단계 제어 변수가 결정될 때 까지의 과정을 반복하여 각 단계의 제어 변수를 선택한다.

(6) 최종 단계 함수에 의하여 U_n 모듈로 구성된 나무구조 회로를 실현 한다.

[예 1] U_n 모듈을 사용하여 아래의 3차 논리 함수 $f_1(X_1, X_2, X_3)$ 를 실현한다.

$$f_1(X_1, X_2, X_3) = 1 \oplus 2X_2 \oplus X_1 \oplus X_1X_2 \oplus 2X_1X_2 \oplus 2X_1X_2X_3 \oplus X_1X_2^2 \oplus X_1^2X_2 \oplus 2X_1^2X_2 \oplus 2X_1^2X_2X_3 \oplus 2X_1^2X_2^2 \oplus X_1^2X_2^2X_3$$

C_3 계수의 벡터는 다음과 같이 주어진다.

$$C = [100200000110220100111222010]^T$$

(1) (a) X_1 이 첫 단계 제어 입력 변수인 경우

$$[G_1(0)] = \begin{bmatrix} 1 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 2 & 2 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 2 & 2 & 2 & 0 & 1 & 0 \end{bmatrix}$$

식 (11)에 의하여 생략되는 U_h 모듈의 수를 구한다.

$$D(g(1)) = \sum_k T_k + \sum_k O_k + S_1 + S_2 + S_3 = 4 + 3 + 0 + 1 + 2 = 10$$

(b) X_2 가 첫 단계 제어 입력 변수인 경우

$$[G_2(0)] = \begin{bmatrix} 1 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 1 \\ 2 & 0 & 0 & 2 & 2 & 0 & 2 & 2 & 2 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

$$D(g(1)) = \sum_k T_k + \sum_k O_k + S_1 + S_2 + S_3 = 6 + 4 + 2 + 3 + 0 = 15$$

(c) X_3 가 첫 단계 제어 입력 변수인 경우

$$[G_3(0)] = \begin{bmatrix} 1 & 2 & 0 & 1 & 2 & 1 & 1 & 2 & 0 \\ 0 & 0 & 0 & 1 & 2 & 0 & 1 & 2 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 2 & 0 \end{bmatrix}$$

$$D(g(1)) = \sum_k T_k + \sum_k O_k + S_1 + S_2 + S_3 = 10 + 2 + 0 + 0 + 2 = 14$$

위에서 $D(g(1))$ 이 가장 큰 값을 갖는 변수 X_2 가 첫 단계 제어 입력 변수로 선택된다. 따라서 첫 단계 함수는 다음과 같이 된다.

$$(g_2(1,0)g_2(1,1)\dots g_2(1,8)) = [1 X_2 X_2^2] \begin{bmatrix} 1 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 2 & 2 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 2 & 2 & 2 & 0 & 1 & 0 \end{bmatrix} = [1 \oplus 2X_2, 0, 0, 1 \oplus 2X_2 \oplus X_2^2, 1 \oplus 2X_2, 1 \oplus 2X_2 \oplus X_2^2, 1 \oplus 2X_2]$$

(2) (a) X_1 이 두번째 단계 제어 입력 변수인 경우

$$[G_1(1)] = \begin{bmatrix} 1 \oplus 2X_2 & 0 & 0 \\ 1 \oplus 2X_2 \oplus X_2^2 & 1 \oplus 2X_2 & 0 \\ 1 \oplus 2X_2 & 1 \oplus 2X_2 \oplus X_2^2 & 1 \oplus 2X_2 \end{bmatrix}$$

$$D(g(2)) = \sum_k T_k + \sum_k O_k + S_1 + S_2 + S_3 = 0 + 0 + 0 + 0 + 0 = 0$$

(b) X_3 가 두번째 단계 제어 입력 변수인 경우

$$[G_3(1)] = \begin{bmatrix} 1 \oplus 2X_2 & 1 \oplus 2X_2 \oplus X_2^2 & 1 \oplus 2X_2 \\ 0 & 1 \oplus 2X_2 & 1 \oplus 2X_2 \oplus X_2^2 \\ 0 & 0 & 1 \oplus 2X_2 \end{bmatrix}$$

$$D(g(2)) = \sum_k T_k + \sum_k O_k + S_1 + S_2 + S_3 = 2 + 0 + 0 + 0 + 1 = 3$$

위에서 $D(g(1))$ 이 가장 큰 값을 갖는 변수 X_3 가 두번째 단계 제어 입력 변수로 선택된다. 따라서 두번째 단계 함수는 다음과 같이 된다.

$$(g_3(2,0)g_3(2,1)g_3(2,2)) = [1 X_3 X_3^2] \begin{bmatrix} 1 \oplus 2X_2 & 1 \oplus 2X_2 \oplus X_2^2 & 1 \oplus 2X_2 \\ 0 & 1 \oplus 2X_2 & 1 \oplus 2X_2 \oplus X_2^2 \\ 0 & 0 & 1 \oplus 2X_2 \end{bmatrix} [(1 \oplus 2X_2) (1 \oplus 2X_2 \oplus X_2^2) \oplus (1 \oplus 2X_2)X_3 (1 \oplus 2X_2) \oplus (1 \oplus 2X_2 \oplus X_2^2)X_3 \oplus (1 \oplus 2X_2)X_3^2]$$

표 1. 각 단계에서 생략된 U_h 모듈의 수

Table 1. Number of omitted U_h module in each level.

Level	Control variable	Tokmen Hurst ^[1]	Hu Wu ^[3]	Fei Zhuang ^[9]	This paper
1st	X_1	4	8	4	10
	X_2	6	12	6	15
	X_3	10	12	10	14
2nd	X_1	0	0	0	0
	X_3	0	2	0	3

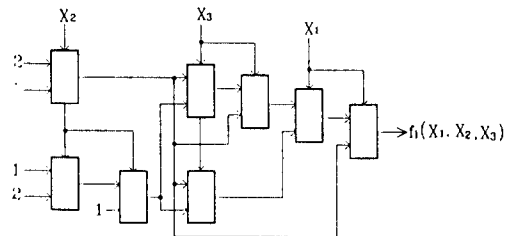


그림 2. U_h 모듈을 사용한 $f_1(X_1, X_2, X_3)$ 의 실현
Fig. 2. Realization of $f_1(X_1, X_2, X_3)$ by using U_h module.

위 식에 의하여 U_h 모듈을 사용한 나무구조 회로 실현을 얻을 수 있다. 표 1에서는 각 단계 별로 생략되는 U_h 모듈의 수를 기존 방법과 비교 하였으며, 그림 2는 함수 f₁(X₁, X₂, X₃)의 실현 회로이다.

[예 2] U_h모듈을 사용하여 아래의 3차 논리 함수 f₂(X₁, X₂, X₃)^[3]를 실현한다.

$$f_2(X_1, X_2, X_3) = 1 \oplus 2X_1 \oplus X_2 \oplus X_1X_2^2 \oplus 2X_1^2 \oplus 2X_1^2X_2 \oplus X_1^2X_3^2 \oplus X_1X_2X_3 \oplus X_1X_2^2X_3 \oplus X_1X_2X_3^2 \oplus 2X_1X_2^2 \oplus X_1X_2^2X_3 \oplus 2X_1X_2^2X_3^2 \oplus X_1^2 \oplus 2X_1^2X_2 \oplus X_1^2X_3^2 \oplus 2X_1^2X_2X_3 \oplus 2X_1^2X_2X_3^2 \oplus 2X_1^2X_2^2X_3$$

C₃계수의 벡터는 다음과 같이 주어진다.

$$C] = [120101221011111212121222020]$$

(1) (a) X₁이 첫 단계 제어 입력 변수인 경우

$$D(g(1)) = \sum_k T_k + \sum_k O_k + S1 + S2 + S3 = 0 + 4 + 4 + 0 + 3 = 11$$

(b) X₂가 첫 단계 제어 입력 변수인 경우

$$D(g(1)) = \sum_k T_k + \sum_k O_k + S1 + S2 + S3 = 0 + 3 + 2 + 1 + 3 = 9$$

(c) X₃가 첫 단계 제어 입력 변수인 경우

$$D(g(1)) = \sum_k T_k + \sum_k O_k + S1 + S2 + S3 = 0 + 4 + 0 + 0 + 3 = 7$$

X₁이 첫 단계 제어 입력 변수로 선택되고 첫 단계 함수는 다음과 같이 된다.

$$(g_1(1,0)g_1(1,1) \dots g_1(1,8)) = [1 X_1 X_1^2] \begin{bmatrix} 1 & 2 & 0 & 1 & 0 & 1 & 2 & 2 & 1 \\ 0 & 1 & 1 & 1 & 1 & 1 & 2 & 1 & 2 \\ 1 & 2 & 1 & 2 & 2 & 2 & 0 & 2 & 0 \end{bmatrix}$$

$$= [1 \oplus X_1^2 \quad 2 \oplus X_1 \oplus 2X_1^2 \quad X_1 \oplus X_1^2 \quad 1 \oplus X_1 \oplus 2X_1^2 \quad X_1 \oplus 2X_1^2 \quad 1 \oplus X_1 \oplus 2X_1^2 \quad 2 \oplus 2X_1 \quad 2 \oplus X_1 \oplus 2X_1^2 \quad 1 \oplus 2X_1]$$

(2) (a) X₂가 두번째 단계 제어 입력 변수인 경우

$$D(g(2)) = \sum_k T_k + \sum_k O_k + S1 + S2 + S3 = 0 + 0 + 0 + 0 + 0 = 0$$

(b) X₃가 두번째 단계 제어 입력 변수인 경우

$$D(g(2)) = \sum_k T_k + \sum_k O_k + S1 + S2 + S3 = 0 + 0 + 0 + 0 + 0 = 0$$

표 2. 각 단계에서 생략된 U_h 모듈의 수

Table 2. Number of omitted U_h module in each level.

Level	Control variable	Tokmen Hurst ^[1]	Hu Wu ^[3]	Fei Zhuang ^[5]	This paper
1st	X ₁	4	10	4	11
	X ₂	2	6	2	9
	X ₃	0	5	0	7
2nd	X ₁	0	0	0	0
	X ₂	0	0	0	0

변수 X₂와 X₃의 (D(g2))의 값이 같으므로 모두 두 번째 단계 제어 입력 변수로 가능하다. 표 2에서는

함수 f₂(X₁, X₂, X₃)의 실현시 각 단계 별로 생략되는 U_h 모듈의 수를 기존 방법과 비교 하였다.

VI. 결론

3차 논리 함수 합성시 ULM U_h를 이용하는 것이 ULM U_r를 이용하는 것보다 유연성이 있고 실현이 단순화되는 장점을 갖고 있다. 본 논문은 ULM U_h를 기본 모듈로 사용한 3차 n-변수 논리 함수의 합성 방법에 관한 것이다. 함수 합성시 제안된 ULM U_h의 생략 조건을 이용하여 회로비용을 계산하고 합성 방법을 제시하므로써 각 단계별로 최적의 회로 합성이 실현되며, 결정된 제어 입력 변수에 대한 회로비용의 값은 실현되는 U_h 모듈의 수를 나타낸다. 따라서 기존에 제안된 합성 방법보다 향상된 회로 실현을 할 수 있으며 회로비용의 계산은 행렬에 의하여 처리되므로 변수의 수에 제약을 받지 않는다. 본 논문에서 제시된 ULM U_h의 생략 조건은 ULM U_r의 생략 조건을 모두 만족하므로 ULM U_r 합성에도 적용이 가능하다.

參考文獻

[1] V.H.Tokeman and S.L.Hurst, "A Consideration of Universal Logic Modules for Ternary Synthesis based upon Reed-Muller Coefficients," *Proc. 9th ISMVL*, pp. 248-256, 1979.

[2] Chen.X and Wu.X, "The Synthesis of Ternary Function under Fixed Polarity and Ternary I2L Circuits," *Proc. 13th ISMVL*, pp. 424-429, 1983.

[3] Z.Hu and X.Wu, "The Logic Synthesis using Ternary Universal Logic Module U_hs," *Proc. 17th ISMVL*, pp. 250-259, 1987.

[4] A.D.Singh, J.R.Armstrong, F.G.Gray, " Combinational and Sequential Multivalued Logic Design Using Universal Iterative Tree Structures," *Proc. 9th ISMVL*, pp. 182-189, 1979.

[5] Benchu Fei and Nan Zhuang, "Fast Logic Synthesis based upon Ternary Universal Logic Module U_f," *Proc. 22th ISMVL*, pp. 401-407, 1992.

[6] M.Kameyama and T.Higuchi, "Synthe-

sis of Optimal T-gate Networks in Multiple-Valued Logic," *Proc. 9th ISMVL*, pp. 190-195, 1979.

[7] David Green, *Modern Logic Design*, Addison-Wesley, 1986.

[8] Fang Yang, "Fast Synthesis of Q-valued Function based on Modulo-Algebra Expansion," *Proc. 16th ISMVL*, pp. 36-41, 1986.

著者紹介



金榮建(正會員)

1977年 2月 인하대학교 전자공학과 졸업. 1981年 2月 인하대학교 대학원 전자공학과 공학석사. 1984年 3月 ~ 1993年 2月 부천 전문대학 전자계산과 부교수. 1989年 8月 ~ 현재 인하대학교 대학원 전자공학과 박사과정. 주관심 분야는 회로 및 시스템 설계, 다치논리 회로 설계, 퍼지논리 등임.



金鍾五(正會員)

1980年 2月 인하대학교 전자공학과 졸업. 1982年 2月 서울대학교 대학원 제어계측공학과 공학석사. 1982年 11月 ~ 1989年 9月 금성사 정보기기연구소 선임연구원. 1990年 9月 ~ 현재 인하대학교 대학원 전자공학과 박사과정. 1989年 9月 ~ 현재 동양공업전문대학 전자과 부교수. 주관심 분야는 회로 및 시스템 설계, 다치논리 회로 설계, 퍼지논리, fault detection 등임.

金興壽(正會員) 第29卷 B編 第11號 參照
현재 인하대학교 전자공학과 교수