

삼각 셀의 정의와 효율적인 GRM 계수 생성 기법

정회원 나기수*, 윤병희*, 변기영**

Definition of Triangle Cell and Effective Generating methodology of Generalized Reed-Muller Coefficients

Gi-Su Na*, Byoung-Hee Yoon*, Gi-Young Byun** *Regular Members*

요약

본 논문에서는 삼각 셀을 이용하여 2^n 개의 서로 다른 극수를 갖는 새로운 GRM 계수의 생성 기법을 제안한다. 기존의 GRM 계수 생성방법으로는 주어진 RM계수를 변환행렬과 연산하는 Green 등의 방법과 기본 전달행렬들을 반복 적용하여 다른 극수들을 구하는 Besslich 등의 방법이 대표적이다. 본 논문에서 효율적인 GRM 계수의 생성을 위하여 삼각 셀을 정의하였고, 삼각 셀의 첫 행에 $2n$ 개의 주어진 RM계수를 나열한 후 고정된 수식에 의해 하위 열에 순차적 모듈로 합을 행하는 병렬형 방법이다. 제안한 연산 기법의 효율성을 예증하기위해 기존의 기법들과 비교하였고, 비교결과 n 개의 입력 변수에서 모든 극수의 GRM 계수들을 구하는데 같은 시스템 복잡도 ($\log_2 n$) T_X 에 대하여 Besslich 등의 기법은 $2^{n-1} \times (2^n - 1)$ 개의 2입력 Ex-OR가 필요한 반면에 본 논문에서 제안한 기법은 $2 \times (n-1)$ 변수의 Ex-OR 개수 + 3^{n-1} 개의 2입력 Ex-OR만을 필요로 한다.

ABSTRACT

In this paper, we propose the method to derive new GRM(Generalized Reed-Muller) coefficients for each 2^n polarities using Triangle cell. As the existing methods to generate GRM coefficients, there are Green's method to operate transform matrix with a given RM coefficient and Besslich's method to get other polarities using basic transfer matrices repeatedly. In this paper, Triangle cell is defined so as to obtain GRM coefficients efficiently. After arranging 2^n given RM coefficients of a first row of Triangle cell, sequence modulo sum is performed in parallel to low column by a fixed numerical formula. To prove the efficiency of proposed arithmetic method, it is compared with Besslich's method. As the compared result, to calculate GRM coefficients of all polarities to n input variables, Besslich's method needs $2^{n-1} \times (2^n - 1)$ two-input Ex-ORs and the proposed method needs $2 \times (\text{the number of Ex-ORs for } n-1 \text{ variables}) + 3^{n-1}$ for the same system complexity - $(\log_2^n)T_X$.

Keyword : polarities, triangle cell, coefficients, RM, GRM

I. 서론

지난 수십 년간 2진 논리에 기초한 집적회로기술의 발전은 소자의 집적능력과 기능당 가격비율을 비약적으로 향상시켰으며, 이로써 논리시스템 및 컴퓨터 하드웨어분야의 눈부신 발전을 이룩할 수 있게 되었다. 그러나 회로의 집적도 향상으로 인한 시스템 복잡도 증가는 고밀도 집적회로를 구현하기 위해 해결되어야 할 중대한 문제점으로 부각되고

있다. 시스템 복잡도는 연산회로에 사용되는 단위 연산 게이트의 수에 의한 회로 복잡도와 입력된 신호가 출력에 이르기까지 단위 연산 게이트에서 소비되는 지연 시간을 총칭하며 고속, 집적화 및 대용량 연산 시스템 개발을 위해 이에 대한 연구가 활발히 진행되어 왔다^[1-3].

시스템의 입력과 출력 사이의 관계를 표시하는 함수 형태는 입력 값의 조합에 의해 출력 값이 주어지는 진리표를 일반화한 연산 영역과 입력변수를

* 인하대학교 전자공학과 회로및시스템 연구실(hskim@inha.ac.kr)

** 가톨릭대학교 정보통신 전자공학부(gybyun@catholic.ac.kr)

논문번호: 040097-0302, 접수일자: 2004년 3월 2일

함수적으로 표현한 함수영역으로 구분하여 함수 해석이 가능하며, 이들 두 영역 사이의 변환은 RM 전개식을 이용한다^[4]. RM을 이용한 AND/Ex-OR 구현의 주된 응용은 산술, 통신회로, 에러제어를 위한 코딩구조, 동기화 시스템, 테스트와 같은 여러 응용 분야에서 찾아 볼 수 있는데^[5~6], RM 전개식을 사용하는 이점은 소자 수와 게이트 상호 연결 수에 있어서 타 함수의 논리 회로 실현보다 경제적이며 회로 구현이 용이하다는 점이다.

RM 전개식에 의하여 표현된 함수 영역의 함수는 유일한 함수로 존재하는 것이 아니며, 입력 변수들에 대한 입력의 형태를 달리하면 새로운 형태의 정규화된 함수를 구성 할 수 있다^[7~8]. 이와 같이 n 개의 변수에 대하여 2^n 개의 서로 다른 극수에 대한 계수를 구하는 과정을 일반화된 Reed-Muller 변환, 간략히 GRM 변환이라 하며 이를 이용한 여러 가지 기법들이 제안된 바 있다. Besslich 등^[9~10]은 Ex-OR 연산을 이용하여 GRM 계수의 생성 방법을 극수의 변화에 따르는 계수간의 관계로 맵(map)상에서 해석하여 순차적인 GRM 계수를 구하는 방법을 제안하였다. Wu 등^[11]은 모듈러 연산을 기반으로 Kronecker 곱을 적용한 전달행렬을 연산하여 해당 극수의 GRM 계수를 구하고 IL을 이용하여 3차 함수로 실현, 회로에 적용하는 연구를 발표하다. 그러나 Wu 등의 기법은 변수의 개수가 증가함에 따라 전달행렬의 크기가 선형적으로 비례하기 때문에 이로 인한 연산의 복잡성이 따르는 단점이 있다. Green 등^[12~13]은 2차 함수에 대하여 Kronecker 곱에 의한 전달행렬을 구하고 이를 적용하여 RM 변환과 GRM 계수를 구하였는데 이 방법은 연속적으로 극수 생성이 가능하며 계수 생성 연산식이 간단하고, 특히 3차나 4차 함수에 효과적인 GRM 계수 생성 방법이지만 직렬형의 알고리즘으로 계수 생성에 있어서 많은 수의 연산자가 필요하며 계수 생성 시간도 상대적으로 오래 걸린다는 단점이 있다.

이러한 연구 활동을 토대로 본 논문에서는 순차적 계수 생성의 기본이 되는 단일변수에 대한 삼각셀을 정의하였고 이를 순차적으로 반복 적용한 n 변수에 대한 GRM 계수 생성 알고리즘은 제안하였다. 이 방법은 순차적이고 반복적으로 계수를 전개하여 GRM 계수를 생성할 수 있으며 기존에 발표된 Besslich, Green 등의 방법에 비해 보다 적은 수의 연산자로 구현이 가능하다는 장점이 있다.

II. Reed-Muller 계수와 GRM 계수

본 장에서는 RM 계수와 GRM 계수의 수식적인 생성방법에 대하여 논의한다.

1. RM 계수의 생성

일반적으로 RM 변환에 의하여 연산 영역의 데이터를 함수 영역으로 변환하는 과정을 위한 전달 행렬을 먼저 구하고 연산영역의 계수들과 전달 행렬 간의 연산을 통하여 함수 영역의 계수를 얻을 수 있다. 역으로 함수 영역의 계수들을 연산 영역의 값으로 변환할 때에는 역행렬을 이용하여 연산영역에서의 계수들을 얻을 수 있다. 이를 그림으로 표현하면 그림 1과 같다.

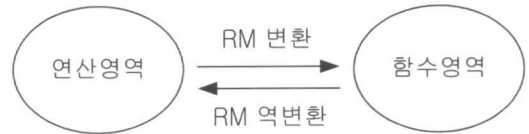


그림 1. 전달 행렬에 의한 영역사이의 변환
Fig. 1. Exchange of domains by transform matrix

단일변수 x_1 에 대한 연산 영역 함수식을 단일 변수 RM 전개식, 즉 함수 영역으로 변환할 때의 전달 행렬을 나타내기 위해 단일 변수 부울 함수식을 이용하면 식 (1)과 같다.

$$f(x_1) = \overline{d_0 x_1} + d_1 x_1 \quad (1)$$

여기서, d_0, d_1 은 $\in GF(2)$ 이다.

식(1)에서 드모르간의 법칙을 적용하면 식(2)와 같은 결과를 얻을 수 있다.

$$f(x_1) = \overline{\overline{d_0 x_1} \cdot d_1 x_1} \quad (2)$$

GF(2) 연산에서는 $\overline{\overline{a}} = a \oplus 1$ 또는 $\overline{a \oplus 1} = a$ 이므로 식(2)을 다시 모듈러-2 연산식으로 전개하면 식 (3)과 같다.

$$\begin{aligned}
 f(x_1) &= (d_0(x_1 \oplus 1) \oplus 1) \cdot (d_1 x_1 \oplus 1) \oplus 1 \\
 &= d_0 d_1 x_1 \oplus d_0 d_1 x_1 \oplus d_0 x_1 \oplus d_0 \oplus d_1 x_1 \\
 &= d_0 \oplus (d_0 \oplus d_1) x_1 \quad ; \text{연산영역} \\
 &= c_0 \oplus c_1 x_1 \quad ; \text{함수영역} \quad (3)
 \end{aligned}$$

식(3)에서 연산영역의 계수들을 함수영역의 계수들과의 관계는 $d_0 = c_0$, $d_0 \oplus d_1 = c_1$ 이므로 이를 행렬로 표현하면 식(4)와 같다.

$$\begin{bmatrix} c_0 \\ c_1 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \cdot \begin{bmatrix} d_0 \\ d_1 \end{bmatrix} \text{ over GF}(2) \quad (4)$$

이 때, 단 변수에서 연산영역의 계수 d_0 와 함수영역의 계수 c_0 와의 관계는 식(4)에서 나타난 것처럼 식(5)와 같은 행렬 형태로 나타나게 되고 이를 변환행렬(transform matrix)이라 한다.

$$T_0 = [1] \quad , \quad T_1 = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \quad (5)$$

식(5)를 n 변수에 대한 변환행렬로 확장하여 표현하면 식(6)과 같다.

$$T_n = \begin{bmatrix} T_{n-1} & 0 \\ T_{n-1} & T_{n-1} \end{bmatrix} \quad (6)$$

2. GRM상수의 생성

앞 절에서 논의되었던 RM 전개식은 모듈러-2의 정규화 된 곱의 합 형식으로 표현 가능하다. RM 전개식에 의하여 표현된 함수는 다치논리의 함수영역에서 유일한 것이 아니다. 만약 입력 변수 x_i 를 $\overline{x_i}$ 로 대체한다면 다른 형태의 정규화 된 형식으로 표현된다. 이와 같이 n 개의 입력변수에 대해 2^n 개의 서로 다른 입력형태가 만들어지며 이에 대한 RM계수를 구하는 것이 GRM 변환이다.

GRM 계수 값의 결정은 보통 연산영역에서 극수에 대한 GRM 계수를 구하는 방법이 있지만 극수가 0인 fixed polarity를 구하고, 극수를 확장시켜 GRM 계수를 구하는 것이 최적화 된 극수를 구하는데 더욱 효과적이다. 극수의 표현은 n -bit 이진수를 십진수로 변환하여 표현한다. 예를 들어, 극수가 4 bit 이진수 0101로 주어졌다면 이때의 극수의 표

현은 극수 $p = 5$ 가 된다. 원래의 RM 계수는 극수 $p = 0$ 인 GRM 계수와 같다.

2.1 단일변수에서의 GRM상수의 생성

단일변수($n=1$)인 경우 GRM 계수를 구하기 위해서는 식(7)과 같은 단일 변수 행렬식을 사용한다.

$$f(x_1') = c_0 \oplus c_1 x_1' \quad (7)$$

x_1' 는 x_1 의 입력형태가 x_1 이거나 또는 $\overline{x_1}$ 인 2가지 형태 중에 하나를 의미한다.

① $x_1' = x_1$ 인 경우;

$x_1' = x_1$ 을 식(7)에 대입하면 GRM 함수의 일반식은 식(8)과 같이 표현된다.

$$\begin{aligned}
 f(x_1') &= c_0 \oplus c_1 x_1 \\
 &= a_0 \oplus a_1 x_1 \quad (8)
 \end{aligned}$$

식(8)는 극수가 $p=0$ 인 경우를 말하며, 계수들간의 관계를 변환행렬을 이용해서 표현하면 식(9)과 같다.

$$\begin{bmatrix} a_0 \\ a_1 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \end{bmatrix} \text{ 또는 } a = Z_0 c \quad (9)$$

② $x_1' = \overline{x_1}$ 인 경우;

모듈러-2 성질에 의하여 $\overline{x_1} = x_1 \oplus 1$ 이 되고 식(7)은 식(10)과 같이 변환된다.

$$\begin{aligned}
 f(x_1') &= c_0 \oplus c_1 \overline{x_1} = c_0 \oplus c_1 (x_1 \oplus 1) \\
 &= a_0 \oplus a_1 x_1 \quad (10)
 \end{aligned}$$

식(10)은 극수가 $p=1$ 인 경우를 말하며, 계수들간의 관계를 변환행렬을 이용해서 표현하면 식(11)과 같다.

$$\begin{bmatrix} a_0 \\ a_1 \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \end{bmatrix} \text{ 또는 } a = Z_1 c \quad (11)$$

2.2 다변수에서의 GRM상수의 생성

$n=2$, 즉 변수가 2개인 경우에 대하여 보면 극수는 2^2 개의 경우가 발생한다.

식(12)과 같이 2변수 함수가 주어졌을 때, 변수 x_2 와 변수 x_1 의 조합에 의해 극수는 4개가 존재한다.

$$f(x_2, x_1) = c_0 \oplus c_1 x_1 \oplus c_2 x_2 \oplus c_3 x_2 x_1 \quad (12)$$

이때 주어진 모든 극수를 표현하면,

① $x_2' = x_2 \quad x_1' = x_1$, 극수 $p=0$ 인 경우;
 $f(x_2', x_1') = f(x_2, x_1) = c_0 \oplus c_1 x_1 \oplus c_2 x_2 \oplus c_3 x_2 x_1$
 $= a_0 \oplus a_1 x_1 \oplus a_2 x_2 \oplus a_3 x_2 x_1$

② $x_2' = x_2 \quad x_1' = \overline{x_1}$, 극수 $p=1$ 인 경우;
 $f(x_2', x_1') = f(x_2, \overline{x_1})$
 $= c_0 \oplus c_1(x_1 \oplus 1) \oplus c_2 x_2 \oplus c_3 x_2(x_1 \oplus 1)$
 $= (c_0 \oplus c_1) \oplus c_1 x_1 \oplus (c_2 \oplus c_3) x_2 \oplus c_3 x_2 x_1$
 $= a_0 \oplus a_1 x_1 \oplus a_2 x_2 \oplus a_3 x_2 x_1$

③ $x_2' = \overline{x_2} \quad x_1' = x_1$, 극수 $p=2$ 인 경우;
 $f(x_2', x_1') = f(\overline{x_2}, x_1) = c_0 \oplus c_1 x_1 \oplus c_2(x_2 \oplus 1) \oplus c_3(x_2 \oplus 1)x_1$
 $= (c_0 \oplus c_2) \oplus (c_1 \oplus c_3) x_1 \oplus c_2 x_2 \oplus c_3 x_2 x_1$
 $= a_0 \oplus a_1 x_1 \oplus a_2 x_2 \oplus a_3 x_2 x_1$

④ $x_2' = \overline{x_2} \quad x_1' = \overline{x_1}$, 극수 $p=3$ 인 경우;
 $f(x_2', x_1') = f(\overline{x_2}, \overline{x_1}) = c_0 \oplus c_1(x_1 \oplus 1) \oplus c_2(x_2 \oplus 1) \oplus c_3(x_2 \oplus 1)(x_1 \oplus 1)$
 $= (c_0 \oplus c_1 \oplus c_2 \oplus c_3) \oplus (c_1 \oplus c_3) x_1 \oplus (c_2 \oplus c_3) x_2 \oplus c_3 x_2 x_1$
 $= a_0 \oplus a_1 x_1 \oplus a_2 x_2 \oplus a_3 x_2 x_1$

각각의 경우마다 계수 a_i 는 달라지며 이 때 계수

C_i 와의 관계는 변환행렬 Z 로 나타난다.

① 극수 $p=0$

$$Z_{00} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} Z_0 & 0 \\ 0 & Z_0 \end{bmatrix}$$

② 극수 $p=1$

$$Z_{01} = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} Z_1 & 0 \\ 0 & Z_1 \end{bmatrix}$$

③ 극수 $p=2$

$$Z_{10} = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} Z_0 & Z_0 \\ 0 & Z_0 \end{bmatrix}$$

④ 극수 $p=3$

$$Z_{11} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} Z_1 & Z_1 \\ 0 & Z_1 \end{bmatrix}$$

위의 4개의 경우를 살펴보면 모든 경우마다 행렬

은 단일변수 변환행렬 Z_0 와 Z_1 의 조합으로 이루어져 있다는 것을 알 수 있다. 단일변수에 대한 GRM변환행렬 Z_0 와 Z_1 를 이용하여 n 변수에 대한 임의의 극수 $p=i$ 의 GRM 변환행렬에 대한 일반식은 식(13)과 같다. 이 때 변환행렬은 $2^n \times 2^n$ 의 차수를 갖는 행렬이며 단일변수 변환 행렬의 kronecker 곱에 의하여 생성된다.

$$Z_{n, n-1, n-2, \dots, 1} = Z_{\langle i \rangle} = Z_n \otimes Z_{n-1} \otimes \dots \otimes Z_1$$

(13)

여기서 $n, n-1, \dots, 1$ 은 0 또는 1이며 \otimes 는 Kronecker 곱이다.

예를 들어 행렬 A 와 B 가 있다고 가정한다.

$$A = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1q} \\ a_{21} & a_{22} & \dots & a_{2q} \\ \vdots & \vdots & \dots & \vdots \\ a_{p1} & a_{p2} & \dots & a_{pq} \end{bmatrix}$$

$$B = \begin{bmatrix} b_{11} & b_{12} & \dots & b_{1q} \\ b_{21} & b_{22} & \dots & b_{2q} \\ \vdots & \vdots & \dots & \vdots \\ b_{p1} & b_{p2} & \dots & b_{pq} \end{bmatrix}$$

행렬 A 와 행렬 B 의 Kronecker 곱 $A \otimes B$ 는 식(14)과 같이 정의한다.

$$A \otimes B = \begin{bmatrix} a_{11}[B] & a_{12}[B] & \dots & a_{1q}[B] \\ a_{21}[B] & a_{22}[B] & \dots & a_{2q}[B] \\ \vdots & \vdots & \dots & \vdots \\ a_{p1}[B] & a_{p2}[B] & \dots & a_{pq}[B] \end{bmatrix}$$

(14)

만약, 행렬 A 가 $p \times q$ 이고 행렬 B 가 $r \times s$ 이면 $A \otimes B$ 는 $pr \times qs$ 의 차수를 갖는다. Kronecker 곱을 표시하는 \otimes 는 결합법칙은 적용되지만 교환법칙은 적용되지 않는다.

$$A \otimes (B \otimes C) = (A \otimes B) \otimes C = A \otimes B \otimes C$$

$$A \otimes B \neq B \otimes A$$

Kronecker 곱의 성질을 이용하면 RM 계수의 변환행렬 T_n 을 단일변수 변환행렬 T_1 으로 표현 가능하다.

$$\begin{aligned} T_n &= T_1 \otimes T_{n-1} = T_1 \otimes T_1 \otimes T_{n-2} \\ &= T_1 \otimes T_1 \otimes \dots \otimes T_1 \end{aligned} \quad (15)$$

식(15)과 같이 n 변수 일 때 변환행렬 T_n 은 단일변수 변환행렬 T_1 을 n 번 Kronecker 곱을 하면 얻을 수 있다.

III. 삼각 셀을 이용한 GRM 계수 생성

본 장에서는 단일변수, 2변수, 3변수에 대한 삼각 셀을 정의하고 이를 반복적으로 적용한 Triangle 방법을 이용한 n 변수 GRM 상수 생성 알고리즘을 논의한다.

1. 단일변수에 대한 삼각 셀의 정의

단일변수에 대한 RM함수는 식(7)에 의해 표현된다.

$$f(x_1) = c_0 \oplus c_1 x_1$$

식(7)에서 극수에 대한 GRM 상수를 구해보면 다음과 같다.

① 극수 0의 GRM 상수식

$$f(x_1) = c_0 \oplus c_1 x_1 = a_0 \oplus a_1 x_1$$

식 (8)으로부터 $a_0 = c_0$, $a_1 = c_1$ 임을 알 수 있다.

② 극수 1의 GRM 상수식

$$\begin{aligned} f(x_1) &= c_0 \oplus c_1 (x_1 \oplus 1) \\ &= (c_0 \oplus c_1) \oplus c_1 x_1 \\ &= a_0 \oplus a_1 x_1 \end{aligned}$$

식(10)으로부터 $a_0 = c_0 \oplus c_1$, $a_1 = c_1$ 임을 알 수 있다.

위의 식(8)과 식(10)으로부터 그림 2와 같은 삼각 셀을 도출할 수 있다.

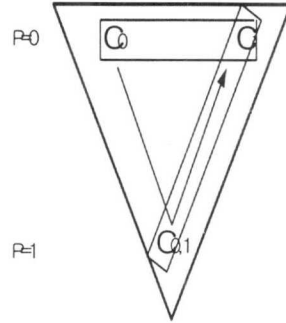


그림 2. 단일변수 삼각 셀.

Fig. 2. Triangle cell for single variable.

그림 2에서 $c_{0,1}$ 은 $c_0 \oplus c_1$ 을 의미한다.

$P=0$ 행에 위치한 사각기호 안의 c_0, c_1 이 극수 0의 GRM 계수이고, $P=1$ 행에 위치한 비스듬한 사각기호 안의 계수 $c_0 \oplus c_1$ 과 c_1 이 극수 1의 계수가 됨을 알 수 있다. 따라서 그림2를 이용하여 단일변수에 대한 GRM 상수들을 구할 수 있다.

그림 2에서 표현한 단일변수에 대한 삼각 셀 형태를 회로로 구현하면 그림 3과 같이 나타낼 수 있다. 그림 3은 3변수 일 경우 밑줄 친 부분이 각각 단일변수에 대하여 주어진 $p = 0$ 의 계수를 서로 Ex-OR하여 $p=1$ 인 계수를 구해낼 수 있는 구조를 가진다.

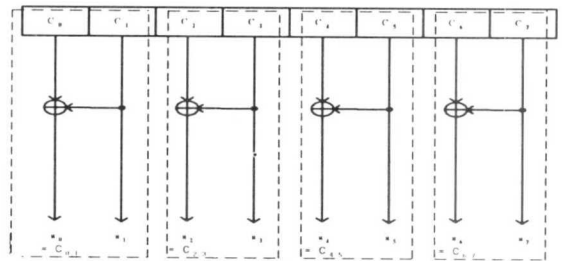


그림 3. 단일변수에 대한 기본적인 변환 회로
Fig. 3. The fundamental transforms circuit for one-variable.

2. 2변수에 대한 삼각 셀 적용

2변수에 대한 RM함수는 식 (12)을 변수의 차 순으로 정리하여 식(16)과 같이 정리할 수 있다.

$$f(x_2, x_1) = (c_0 \oplus c_1 x_1)$$

$$\oplus (c_2 \oplus c_3 x_1) x_2 = g_0^1 \oplus g_1^1 x_2 \tag{16}$$

식 (16)에서 보면 g_0^1, g_1^1 은 각각 단일변수에 대한 RM 함수식과 같은 형태를 갖고 있으며(위첨자 1은 단일변수) 단일변수일 때 적용되어진 방법을 2변수에서도 동일하게 확장 적용할 수 있음을 의미한다.

2변수에서의 GRM 함수식을 전개하면 모든 극수에 대하여 다음과 같이 나타낼 수 있다.

① 극수 0의 GRM 함수식

극수 0의 계수 ; c_0, c_1, c_2, c_3 (RM의 계수와 동일, g_0^1, g_1^1 의 삼각형)

g_0^1, g_1^1 에 대한 삼각 셀 내에서 $p = 0$ 행에 있는 계수들로 표현이 가능하다.

② 극수 1의 GRM 함수식 ; $f(x_2, x_1) = (c_0 \oplus c_1(x_1 \oplus 1)) \oplus (c_2 \oplus c_3(x_1 \oplus 1))x_2$

$$= ((c_0 \oplus c_1) \oplus c_1 x_1) \oplus ((c_2 \oplus c_3) \oplus c_3 x_1) x_2$$

극수 1의 계수 ; $c_0 \oplus c_1, c_1, c_2 \oplus c_3, c_3$ (g_0^1, g_1^1 의 삼각형)

③ 극수 2의 GRM 함수식 ; $f(x_2, x_1) = c_0 \oplus c_1 x_1 \oplus c_2(x_2 \oplus 1) \oplus c_3 x_1(x_2 \oplus 1)$

$$= ((c_0 \oplus c_2) \oplus (c_1 \oplus c_3)x_1) \oplus (c_2 \oplus c_3)x_1 x_2$$

극수 2의 계수 ; $c_0 \oplus c_2, c_1 \oplus c_3, c_2, c_3$ ($g_0^1 \oplus g_1^1, g_1^1$ 의 삼각형)

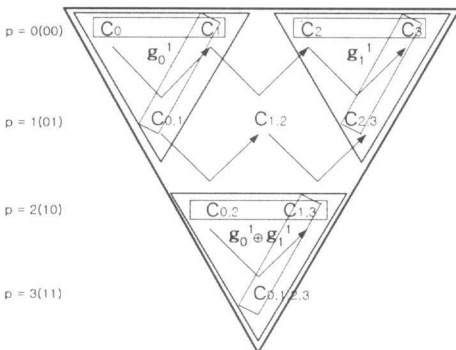


그림 4. 2변수 삼각 셀.
Fig. 4. Triangle cell for 2-variable.

④ 극수 3의 GRM 함수식 ; $f(x_2, x_1) = (c_0 \oplus c_1(x_1 \oplus 1)) \oplus (c_2 \oplus c_3(x_1 \oplus 1))(x_2 \oplus 1)$

$$= ((c_0 \oplus c_1 \oplus c_2 \oplus c_3) \oplus (c_1 \oplus c_3)x_1) \oplus ((c_2 \oplus c_3) \oplus c_3 x_1) x_2$$

극수 3의 계수 ; $c_0 \oplus c_1 \oplus c_2 \oplus c_3, c_1 \oplus c_3, c_2 \oplus c_3, c_3$ ($g_0^1 \oplus g_1^1, g_1^1$ 의 삼각형)

위에서 전개한 2변수에서의 GRM 함수식을 삼각 셀을 이용하여 도시하면 그림 4와 같다. 그림 4에서 $c_{0.1} = c_0 \oplus c_1, c_{2.3} = c_2 \oplus c_3, c_{0.1.2.3} = c_0 \oplus c_1 \oplus c_2 \oplus c_3$ 을 의미하며, 그림 2에서 표현한 단일변수 삼각 셀의 확장 형태를 유지하고 있음을 알 수 있다. 극수에 대한 계수들은 고정된 자리를 갖고 있으므로 단일변수 삼각 셀에 포함되지 않는 $c_{1.2}$ 는 실제로 연산에 영향을 주지 않게 된다.

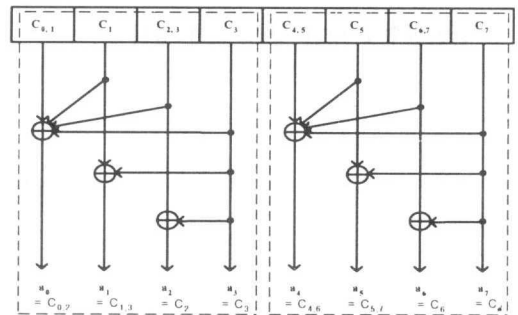


그림 5. 2변수 삼각 셀 확장에 대한 변환회로.
Fig. 5. The transforms circuit for Triangle cell expansion of two-variable.

그림 5는 단일변수의 삼각 셀(극수 1)에서 2변수 삼각 셀(극수 2)로의 확장을 할 때 적용되는 변환회로이다. 즉, 극수 1(01)에서의 계수들 $c_0 \oplus c_1, c_1, c_2 \oplus c_3, c_3$ (g_0^1, g_1^1 의 삼각형)을 이용해서 극수가 2인 계수 $c_0 \oplus c_2, c_1 \oplus c_3, c_2, c_3$ ($g_0^1 \oplus g_1^1, g_1^1$ 의 삼각형)을 얻어내기 위한 연관관계 변환회로이다.

3. 3변수에 대한 삼각 셀 적용

3변수에 대한 RM함수는 식 (13)을 변수의 차 순으로 확장하여 정리하면 식(17)과 같이 정리할 수 있다.

$$f(x_3, x_2, x_1) = c_0 \oplus c_1 x_1 \oplus c_2 x_2 \oplus c_3 x_1 x_2 \oplus c_4 x_3 \oplus c_5 x_3 x_1 \oplus c_6 x_3 x_2 \oplus c_7 x_3 x_2 x_1 \quad (17)$$

위의 식 (17)은 변수의 순서대로 다시 정리하면 식 (18)과 같이 나타낼 수 있다.

$$f(x_3, x_2, x_1) = [(c_0 \oplus c_1 x_1) \oplus (c_2 \oplus c_3 x_1) x_2] \oplus [(c_4 \oplus c_5 x_1) \oplus (c_6 \oplus c_7 x_1) x_2] x_3 = g_0^2 \oplus g_1^2 x_3 \quad (18)$$

식 (18)에서 보면 g_0^2, g_1^2 은 각각 단일변수에 대한 RM 함수식과 같은 형태를 갖고 있으며(위첨자 2는 2변수) 단일변수일 때 적용되어진 방법을 3변수에서도 동일하게 확장 적용할 수 있음을 보여준다.

3변수에서의 GRM 함수식을 전개하면 모든 극수에 대하여 다음과 같이 나타낼 수 있다.

- ① 극수 0, 1, 2, 3의 GRM 함수식이

$$f(x_3) = g_0^2 \oplus g_1^2 x_3$$

이므로 각 극수에 대한 모든 계수들은 g_0^2 와 g_1^2 에 해당하는 삼각 셀 안에서 단일변수 삼각 셀과 2변수 삼각 셀의 적용에 의하여 모두 구해낼 수 있다.

극수 0의 계수 : $c_0, c_1, c_2, c_3, c_4, c_5, c_6, c_7$ (RM의 계수와 동일)

극수 1의 계수 :

$$c_0 \oplus c_1, c_1, c_2 \oplus c_3, c_3, c_4 \oplus c_5, c_5, c_6 \oplus c_7, c_7$$

극수 2의 계수 :

$$c_0 \oplus c_2, c_1 \oplus c_3, c_1, c_3, c_4 \oplus c_6, c_5 \oplus c_7, c_6, c_7$$

극수 3의 계수 :

$$c_0 \oplus c_1 \oplus c_2 \oplus c_3, c_1 \oplus c_3, c_2 \oplus c_3, c_3, c_4 \oplus c_5 \oplus c_6 \oplus c_7, c_5 \oplus c_7, c_6 \oplus c_7, c_7$$

- ② 극수 4, 5, 6, 7의 GRM 함수식이

$$f(x_3) = g_0^2 \oplus g_1^2 (x_3 \oplus 1)$$

$$= (g_0^2 \oplus g_1^2) \oplus g_1^2 x_3$$

이므로 각 극수에 대한 모든 계수들은 $g_0^2 \oplus g_1^2$ 와 g_1^2 에 해당하는 삼각 셀 안에서 단일변수 삼각 셀과 2변수 삼각 셀의 적용에 의하여 모두 구해낼 수 있다.

- 극수 4의 계수 : $c_0 \oplus c_4, c_1 \oplus c_5, c_2 \oplus c_6, c_3 \oplus c_7, c_4, c_5, c_6, c_7$
- 극수 5의 계수 : $c_0 \oplus c_1 \oplus c_4 \oplus c_5, c_1 \oplus c_5, c_2 \oplus c_3 \oplus c_6 \oplus c_7, c_3 \oplus c_7, c_4 \oplus c_5, c_5, c_6 \oplus c_7, c_7$
- 극수 6의 계수 : $c_0 \oplus c_2 \oplus c_4 \oplus c_6, c_1 \oplus c_3 \oplus c_5 \oplus c_7, c_2 \oplus c_6, c_3 \oplus c_7, c_4 \oplus c_6, c_5 \oplus c_7, c_6, c_7$
- 극수 7의 계수 : $c_0 \oplus c_1 \oplus c_2 \oplus c_3 \oplus c_4 \oplus c_5 \oplus c_6 \oplus c_7 \oplus c_7, c_1 \oplus c_3 \oplus c_6 \oplus c_7, c_2 \oplus c_3 \oplus c_6 \oplus c_7, c_3 \oplus c_7, c_4 \oplus c_5 \oplus c_6 \oplus c_7, c_5 \oplus c_7, c_6 \oplus c_7, c_7$

위에서 전개한 3변수에서의 GRM 함수식을 삼각 셀을 이용하여 도시하면 그림 6과 같다. 그림 6에서

$c_{0-3} = c_0 \oplus c_1 \oplus c_2 \oplus c_3$,
 $c_{0-7} = c_0 \oplus c_1 \oplus c_2 \oplus c_3 \oplus c_4 \oplus c_5 \oplus c_6 \oplus c_7$ 을 의미하며, 앞서 설명한 그림 2의 단일변수에 대한 삼각 셀과 이의 확장 형태인 그림 4의 2변수 삼각 셀의 형태를 순환적으로 반복하여 확장된 형태로 구성하였음을 알 수 있다.

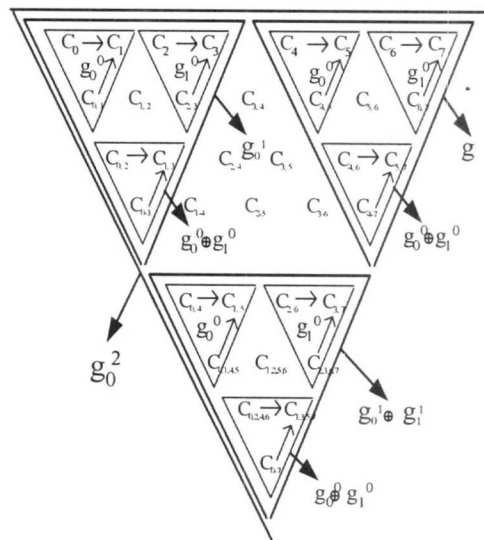


그림 6. 3변수 삼각 셀. Fig. 6. Triangle cell for 3-variable.

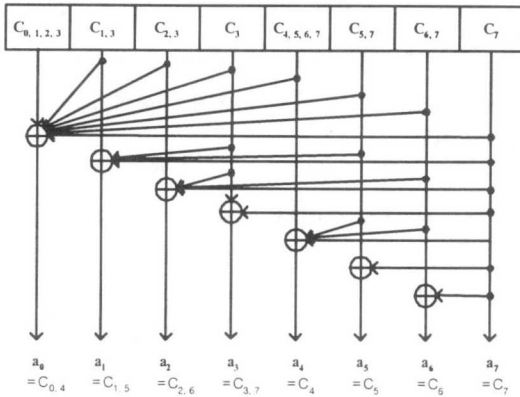


그림7. 3변수 삼각 셀 확장에 대한 변환회로
 Fig. 7. The transforms circuit for Triangle cell expansion of three-variable.

그림 7은 2변수의 삼각 셀(극수 3)에서 3변수 삼각 셀(극수 4)로의 확장을 할 때 적용되는 변환회로로 극수 3(011)의 계수들

$$c_0 \oplus c_1 \oplus c_2 \oplus c_3, c_1 \oplus c_3, c_2 \oplus c_3, c_3, c_4 \oplus c_5 \oplus c_6 \oplus c_7, c_5 \oplus c_7, c_6 \oplus c_7, c_7$$

이용해서 극수가 4인 계수 $c_0 \oplus c_4, c_1 \oplus c_5, c_2 \oplus c_6, c_3 \oplus c_7, c_4, c_5, c_6, c_7$ 을 얻어내기 위한 회로이다. 이 회로는 3변수 삼각 셀에서 가장 복잡한 형태를 가지지만 실제 연산에서는 계수들 간에 이미 고정된 연산을 수행하므로 극수들의 순차적인 연관관계를 의미하는 회로로 해석할 수 있다.

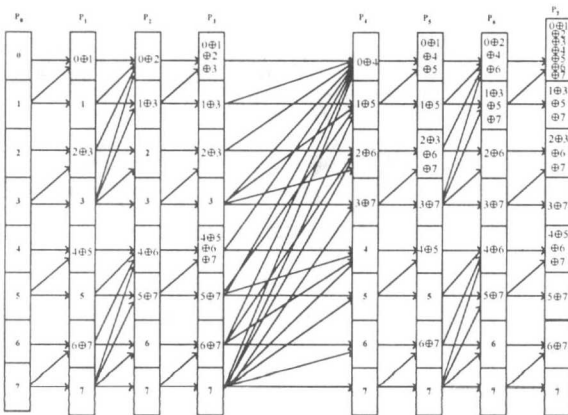


그림 8. 3변수 GRM 계수에 대한 연속적인 생성
 Fig. 8. Consecutive generation of GRM Coefficients for 3-variable.

그림 8은 3변수일 때 본 논문에서 제안할 삼각 셀을 이용한 GRM 계수들의 관계가 순차적인 연관 관계가 있음을 보여주며 또한, 병렬적 구조로 모든 극수의 계수들을 연속적이며 순차적으로 구해낼 수 있음을 보여준다. 이 회로구성에서 각각 극수의 계수는 모두 고정된 연산식을 가지므로 쉽게 구성할 수 있고 빠른 속도로 연산을 수행할 수 있다.

3. n변수에 대한 삼각 셀 적용

앞장에서 설명한 3변수에 대한 삼각 셀을 n변수에 대해서 확장 적용할 수 있다.

n변수로 확장했을 때 RM 함수의 일반식은 식 (15)과 같으며 이것을 정리해보면 식 (19)로 나타낼 수가 있다.

$$f(x_1, x_2, \dots, x_n) = g_0^{n-1} \oplus g_1^{n-1} x_n \quad (19)$$

식 (19)는 단일변수 RM함수 일반식인 식 (15)와 형태가 같고 g_0^{n-1}, g_1^{n-1} 은 각각 n-1변수의 RM 일반식을 의미한다.

단일변수의 식 (8)과 식 (10)을 식 (19)에 적용하면

$$\textcircled{1} x_n \text{에 변화를 주지 않은 경우(극수 } 0, 1, 2, \dots, 2^{n-1}-1)$$

GRM 계수는 g_0^{n-1}, g_1^{n-1} 가 되고 함수식은 아래와 같다.

$$f(x_n') = f(x_n) = g_0^{n-1} \oplus g_1^{n-1} x_n \quad (20)$$

$$\textcircled{2} x_n = x_n \oplus 1 \text{인 경우(극수 } 2^{n-1}, 2^{n-1}+1, \dots, 2^n-1)$$

GRM 계수는 $g_0^{n-1} \oplus g_1^{n-1}, g_1^{n-1}$ 이 되고 함수식은 아래와 같다.

$$f(x_n') = f(x_n) = g_0^{n-1} \oplus g_1^{n-1} (x_n \oplus 1) = (g_0^{n-1} \oplus g_1^{n-1}) \oplus g_1^{n-1} x_n \quad (21)$$

위에서 유도한 식 (20)과 식 (21)을 삼각 셀로 도시하면 그림 8과 같다.

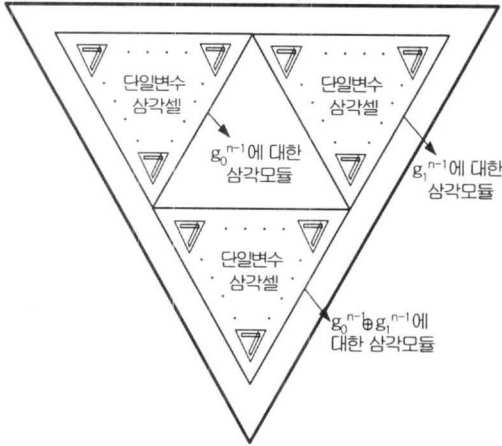


그림 9. n 변수 삼각 셀.
Fig. 9. Triangle cell for n-variable.

그림 9을 통해서 n변수에서의 각각의 극수에 대한 GRM 상수를 한 번에 구할 수가 있다. 즉, 단일 변수의 Triangle을 확장하여 n 변수의 GRM 상수를 Triangle을 사용해서 쉽게 구할 수 있다.

이것을 바탕으로 해서 **Triangle 방법을 이용한 n 변수 GRM 상수 생성 알고리즘**을 제안한다.

Triangle 방법을 이용한 n 변수 GRM 상수 생성 알고리즘

[단계 1] 삼각 셀을 그림 9과 같이 크게 3그룹 (g_0^{n-1} , g_1^{n-1} , $g_0^{n-1} \oplus g_1^{n-1}$)의

삼각 셀로 나눈다.

[단계 2] 나누어진 셀을 다시 그림 2와 같은 단일 변수 셀이 될 때까지 계속해서 나눈다.

[단계 3] 주어진 RM 계수들을 삼각 셀의 첫 행에 기입한다.

[단계 4] n차에 대하여 모든 계수들의 자리에 연산식이 고정되어 있으므로 주어진 RM 계수들을 사용하여 삼각 셀의 모든 계수들을 순차적으로 구한다.

[단계 5] 나열된 계수들 맨 왼쪽에 극수 0부터 $2^n - 1$ 까지 행 별로 차례차례 적는다.

극수가 적힌 행에서 가장 근접한 단일변수 삼각 셀이 바로 그 극수의 GRM 계수의 시작이 된다.

[단계 6] 단일변수 셀부터 사각기호로 연결된 2개

의 GRM 계수를 찾는다.

[단계 7] 적용된 단일변수 삼각 셀을 포함하는 확장된 2변수 삼각 셀에서 우측이나 우측 상단에 또 다른 단일변수 삼각 셀이 있는지 확인하여 만일, 단일변수 삼각 셀이 존재하면 단계 6을 반복하고 없으면 단계 9로 간다.

[단계 8] 단계 7을 n 변수까지 확장하여 반복 실행한다.

[단계 9] 더 이상 적용되지 않은 단일변수 삼각 셀이 없으므로 구해진 GRM 상수들을 순서대로 나열

하고 알고리즘을 끝낸다.

예제를 통하여 주어진 RM 함수식에서 [Triangle 방법을 이용한 n 변수 GRM 상수 생성 알고리즘]을 이용하여 모든 극수에서의 GRM 계수를 생성하는 과정을 살펴본다.

[예제] 제안된 알고리즘을 사용하여 다음 RM함수에서 P=1과 P=6의 GRM 상수를 구하라.

$$f(x_3, x_2, x_1) = 1 \oplus x_1 \oplus x_2 \oplus x_3 x_1 \oplus x_3 x_2 x_1$$

주어진 함수식으로부터 RM계수(P = 0)들은 다음과 같이 구해진다.

$$p = 0 : [1 \ 1 \ 1 \ 0 \ 0 \ 1 \ 0 \ 1]$$

제안한 알고리즘 단계중 [단계 1]부터 [단계 5]까지를 적용하면 그림 10과 같다.

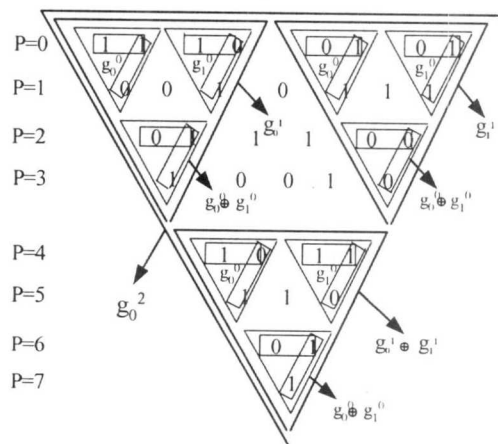


그림 10. 3변수에 대한 삼각 셀 방법 적용 예.

Fig. 10. An Example using Triangle cell for 3-variable.

[단계 6] P=0와 P=6에 적힌 행에서 가장 근접한 단일변수 셀부터 사각기호로 연결된 2개의 GRM 계수를 찾는다.

P = 1 : 가장 근접한 단일변수 삼각 셀 g_0^0 에서 사각 기호로 연결된 2개의 GRM 계수는 [0 1]

P = 6 : 가장 근접한 단일변수 삼각 셀 $g_0^0 \oplus g_1^0$ 에서 사각 기호로 연결된 2개의 GRM 계수는 [0 1]

[단계 7-1] 적용된 단일변수 삼각 셀(P = 0 : g_0^0 , P = 6 : $g_0^0 \oplus g_1^0$)을 포함하는 확장된 2변수 삼각 셀(P = 0 : g_0^1 , P = 6 : $g_0^1 \oplus g_1^1$)에서 알고리즘을 적용한 단일변수 삼각 셀의 우측(P = 1 : g_1^0)이나 우측 상단(P = 6 : g_1^0)에 또 다른 단일변수 삼각 셀이 존재하므로 단계 6을 반복한다.

P = 1 : 단일변수 삼각 셀 g_1^0 에서 사각 기호로 연결된 2개의 GRM 계수는 [1 0]

P = 6 : 단일변수 삼각 셀 g_1^0 에서 사각 기호로 연결된 2개의 GRM 계수는 [1 1]

[단계 8] 단계 7을 3 변수까지 확장하여 반복 실행한다.

[단계 7-2] 적용된 2변수 삼각 셀(P = 0 : g_0^1 , P = 6 : $g_0^1 \oplus g_1^1$)을 포함하는 확장된 3변수 삼각 셀(P = 0, 6 : g_2^0)에서 알고리즘을 적용한 2변수 삼각 셀의 우측(P = 1 : g_1^1)이나 우측 상단(P = 6 : g_1^1)에 또 다른 단일변수 삼각 셀이 존재하므로 단계 6과 단계 7-1을 반복한다.

P = 1 : 단일변수 삼각 셀 g_0^0 에서 사각 기호로 연결된 2개의 GRM 계수는 [1 1], 단일변수 삼각 셀 g_1^0 에서 사각 기호로 연결된 2개의 GRM 계수는 [1 1]

P = 6 : 단일변수 삼각 셀 $g_0^0 \oplus g_1^0$ 에서 사각 기호로 연결된 2개의 GRM 계수는 [0 0], 단일변수 삼각 셀 g_1^0 에서 사각 기호로 연결된 2개의 GRM 계수는 [0 1]

[단계 9] P1 = [0 1 1 0 1 1 1 1], P6 = [0 1 1 1 0 0 0 1]

IV. 기존 방법과 제안한 방법과의 비교

본 장에서는 주어진 RM 함수에서 최적의 GRM 함수를 찾아내기 위하여 제안된 GREEN 등의 방법과 Besslich 등의 방법을 본 논문에서 제안한 방법에 대하여 그 특성 및 설계 방법, 설계된 회로 구성에 대한 결과들을 비교 및 분석한다.

1. 구조적인 알고리즘의 비교

GRM 계수를 구하는 목적은 최적의 극수를 선정하여 최적화된 GRM 함수를 구성하기 위한 것이며, 최적화된 GRM 함수는 보다 간단한 회로 형태로 표현된다. 이러한 간략화된 회로 함수를 도출하기 위하여 최적화된 GRM 함수를 찾는 방법으로써 단일변수 전달행렬을 이용하여 n 변수 전달행렬로 확장해서 모든 극수들의 전달 행렬을 구하는 GREEN 등의 방법과 전달 행렬의 특성을 이용하여 GREEN 등의 방법을 개선시킨 Besslich 등의 직렬형 방법 등이 있다.

최적화 함수를 생성하기 위한 기존의 GREEN 등의 방법은 단일변수에 대한 계수들을 전달행렬로 표현하고 이들 간에 Kronecker 곱을 해서 다 변수의 전달행렬로 확장시켜 행렬연산을 이용한 최적화 GRM 함수를 결정하였다. 이러한 고전적인 방법은 변수의 증가에 따른 행렬의 차수가 선형적으로 증가하게 되고 연산자의 개수가 많아지는 문제점이 있다. Besslich 등의 방법은 전달 행렬의 특성을 이용하여 기본 전달 행렬들을 정하고 변수의 개수에 따라 정해진 기본 전달 행렬들을 반복 적용하여 다른 극수들을 구하는 직렬형 방법이다. 따라서 Besslich 등의 방법이 Green의 방법에 비해 소자의 개수와 도식적인 면에서 개선된 부분이 많지만 Besslich 등의 방법 역시 변수가 증가함에 따라서 필요로 하는 연산자의 개수가 많아지는 단점이 있다. 이러한 기존의 방법들과는 달리 본 논문에서 제안한 알고리즘은 도식적으로 GRM 계수들을 구할 수 있는 부분에 있어서는 Besslich 등의 방법과 같은 측면이 있으나 단일변수 삼각 셀에서 n 변수 삼각 셀로의 반복적인 확장을 통하여 병렬형으로 GRM 계수를 구해나감으로써 적은 수의 연산자를

표1. 세 가지 방법의 시스템 복잡도 비교
System comparison of three algorithmic complexities.

	변수	Green 등의 방법		Besslich 등의 방법	본 논문에서 제안한 방법
		가산기 (Ex-OR)	승산기 (Multiplier)	가산기 (Ex-OR)	가산기 (Ex-OR)
회로 복잡도	1	1	4	1	1
	2	4	32	6	5
	3	64	256	28	19
	4	128	2048	120	65
	5	256	16384	496	211
	6	1024	131072	2016	665
	n	$2^n \times 2^{n-2}$	$4^n \times 2^{n-1}$	$2^{n-1} \times (2^n - 1)$	$2 \times (n-1 \text{ 변수에서의 가산기수}) + 3^{n-1}$
지연시간				$(\log_2 n) T_X$	$(\log_2 n) T_X$

가지고서 효율적으로 모든 극수에 해당하는 GRM 계수들을 순차적으로 구할 수 있고 최적의 함수를 도출해낼 수 있다.

2. 기존 방법과의 시스템 복잡도 비교

본 논문에서 제안한 방법은 최적화된 GRM 함수를 생성하는데 있어서 계산된 일부 극수의 계산치를 다른 극수에서 반복적으로 사용하기 때문에 필요한 연산자의 수를 최소화 할 수 있다. Green 등이 제안한 알고리즘은 모든 극수의 GRM 계수를 구하기 위하여 n 변수일 경우 $2^n \times 2^{n-2}$ 개의 가산기(Ex-OR)와 $4^n \times 2^{n-1}$ 개의 승산기(Multiplier)가 요구된다. 또한, Besslich 등이 제안한 알고리즘은 승산기는 요구되어지지 않지만 $2^{n-1} \times (2^n - 1)$ 개의 가산기가 필요하다. 그러나 본 논문에서 제안한 알고리즘은 변수에 따라 계수들의 위치에 연산식이 고정되어 있어서 삼각 셀을 구성하기 위하여 모든 계수들 간의 승산이 필요하지 않고 Besslich 등과 같은 지연시간 $(\log_2 n) T_X$ 으로 $2 \times (n-1 \text{ 변수에서의 가산기수}) + 3^{n-1}$ 개의 가산기만 요구되어진다. 표 1은 본 논문에서 제안한 방법과 Green 등과 Besslich 등의 방법을 변수가 1에서 n 일 때 까지 모든 극수에 대하여 GRM 계수를 구할 때 필요한 시스템 복잡도를 비교하였다. 분명하게 본 논문에서 제안한 알고리즘이

기준에 제안된 Green과 Besslich 등의 방법보다 더 많은 장점이 있다는 것을 확인할 수 있다.

IV. 결론

본 논문은 단일변수 삼각 셀을 순환적으로 사용하여 2^n 개의 서로 다른 극수(polarity)를 갖는 새로운 GRM(Generalized Reed-Muller) 계수를 생성하는 방법에 대하여 논의 하였다.

본 논문에서 제안한 알고리즘은 기준에 제안되어진 알고리즘들이 모든 극수의 GRM 계수들을 생성하는데 있어서 많은 수의 연산자들이 필요하다는 단점을 개선하기 위하여 주어진 $p=0$ 의 RM상수를 고정된 연산순서를 갖는 삼각 셀 형태로 변환하고 단일변수 삼각 셀을 n 변수 삼각 셀까지 순환적으로 사용하여 적은 연산자를 가지고 순차적이고 병렬구조로 모든 극수에 대하여 GRM 계수를 생성하는 방법을 유도하였다.

알고리즘의 관점에서 Green 등의 알고리즘은 행렬식구조이고 Besslich 등의 알고리즘은 직렬형 구조를 갖는 반면 본 논문에서 제안한 알고리즘이 고속 병렬형 방법임을 보였으며, 같은 변수에서 모든 극수에서의 GRM 계수를 구하기 위하여 요구되어지는 시스템 복잡도를 비교함으로써 본 논문에서 제안한 방법이 더 효율적임을 증명하였다.

본 논문은 GF(2)일 경우에 한하여 연구가 진행되었다. 보다 일반적인 GRM 해석을 위하여 GF(2^m)

에서의 연구가 진행되어야 한다고 사료된다.

참 고 문 헌

[1] M. Kameyama, "Toward The Age of Beyond-Binary Electronics and Systems," IEEE Proc. 20th Int. Symposium on Multiple-Valued Logic, pp.162-166, May 1990.

[2] K. C. Smith and P. G. Gulak, "Prospects for multiple-valued integrated circuits," Special issue on Multiple-Valued Integrated Circuits IEICE Trans. Electron. vol. E76-C, no. 3, pp.372-382, Mar. 1993.

[3] T. Hanyu, M. Kameyama and T. Higuchi, "Prospects of Multiple-Valued VLSI Processors," Special issue on Multiple-Valued integrated Circuits IEICE Trans. Electron, vol. E76-C, no. 3, pp. 383-391, Mar. 1993.

[4] S. M. Reddy. "Easily testable realization for logic functions," IEEE Trans. on Computer, vol 11(11), pp. 1183-1188, Nov. 1972

[5] J. Saul. "An Algorithm for the multi-level Minimization of Reed-Muller Representation," IEEE Proc. of ICCD, pp 634-637, 1991

[6] M. Helliwell and M. Perkowski. "A Fast Algorithm to Minimized Multi-Output Mixed-Polarity Generalized Reed-Muller Forms," Design Automation Conference, pp. 427-432, 1988

[7] T. R. Damarla, M. Karpovsky, "Fault detection in combinational network by Reed-Muller transform, IEEE Trans. Computers," Vol. 38, No. 6, pp 788-797, June 1989

[8] Q. Hong, B. Fei, H. Wu, M. Perkowski, N. Zhuang, "Fast Synthesis for Ternary Reed-Muller Expansion," IEEE Proc. of Symposium on Multiple-Valued Logic, pp. 14-16, May 1993

[9] W. Besslich, "Efficient computer method for Ex-OR logic design," IEE Proc., vol 130, no. 6, pp. 203-206, Nov. 1983

[10] T. Sasao and P. Besslich, "On the complexity of mod-2 sum PLAs," IEEE Transactions on Computer, vol. 39, no. 2,

pp. 262-266, Feb 1990

[11] X. Wu, X. Chen, "The synthesis of ternary funtions under fixed polarities and ternary I²L circuits," IEEE Proc. of Symposium on Multiple-Valued Logic, pp. 424-429, May 1983

[12] David Green, Modern Logic Design, Addison-Wesley Publishing company, Inc.1986 pp. 131-164

[13] D. H. Green, "Reed-Muller canonical forms with mixed polarity and their manipulations," IEE Proc., vol. 137, Pt. E, no. 1, pp. 103-113, Jan. 1990

나 기 수(Gi-Su Na)

정회원



1997년 2월 : 건양대학교
컴퓨터공학과 (공학사)
1999년 2월 : 인하대학교 대학원
전자공학과 (공학석사)
1999년 3월 - 현재 : 인하대학교
대학원 전자공학과 박사과정
<주관심 분야> 디지털 로직, 오
류정정부호 설계, 퍼지회로 설계

윤 병 희(Byoung-Hee Yoon)

정회원



1997년 2월 : 원광대학교
전자공학과 (공학사)
1999년 2월 인하대학교 대학원
전자공학과 (공학석사)
1999년 3월 ~ 현재 :
인하대학교 대학원 전자공학과
박사과정

<주관심분야> 다치 프로세서, 다치 저장 소자 설계, VLSI 설계

변 기 영(Gi-Young Byun)

정회원



1994년 2월 : 인하대학교
전자공학과 (공학사)
1998년 8월 : 인하대학교 대학원
전자공학과 (공학석사)
2003년 2월 : 인하대학교 대학원
전자공학과 (공학박사)
1994년 1월 ~ 1996년 8월 :

(주)LG전자 VCR사업부 회로설계연구원
2003년 3월 ~ 현재 : 가톨릭대학교 정보통신
전자공학부 강의전담교수
현재 IEK, KICS 정회원, IEICE 해외회원
<주관심분야> 정보 및 부호이론, 논리시스템설계,
컴퓨터 구조, 유한체 이론의 응용 및 VLSI 회로
구현 등