

論文98-35C-3-7

에지값 決定圖에 의한 多值論理函數構成과 全加算機設計에 관한 研究

(A Study on the Construction of Multiple-Valued Logic Functions and Full-Adders using by the Edge-Valued Decision Diagram)

韓 聖 一 * , 崔 在 碩 * , 朴 春 明 ** , 金 興 壽 *

(Sung-Il Han, Jai-Sock Choi, Chun-Myoung Park, and Heung-Soo Kim)

요 약

본 논문에서는 최근의 디지털논리시스템의 함수구성시에 도입되고 있는 그래프이론에 바탕을 둔 결정도로 부터 새로운 형태의 데이터구조 형태인 에지값 다치논리 결정도를 추출하는 알고리즘의 한가지 방법을 제안 하였다. 그리고 이를 기초로 임의의 m 치 n 변수의 축약된 함수구성을 도출하는 방법에 대해 논의하였다. 제안된 알고리즘의 규칙성을 고려하여 동일부분을 모듈화함으로써 일반성을 가짐을 보였다. 모듈계산을 통한 전가산기를 EMVDD로 도시하였고 적용예를 들어 제안한 알고리즘의 타당성을 검증하였다. 제안한 다치논리함수구성방법은 도식적이며 규칙적이고 정규성을 내포하고 있다.

Abstract

This paper presented a method of extracting algorithm for Edge Multiple-Valued Decision Diagrams(EMVDD), a new data structure, from Binary Decision Diagram(BDD) which is resently used in constructing the digital logic systems based on the graph theory. We discussed the function minimization method of the n -variables multiple-valued functions and showed that the algorithm had the regularity with module by which the same blocks were made concerning about the schematic property of the proposed algorithm. We showed the EMVDD of Full Adder by module construction and verified the proposed algorithm by examples. The proposed method has the visible, schematical and regular properties.

I. 서 론

현재 사용되고 있는 디지털논리시스템 및 이를 근간으로 하는 컴퓨터 하드웨어 분야는 2치논리(2진논리)에 기초를 두고 집적회로기술의 비약적인 발전으로 회로의 형태가 VLSI, ULSI화 되어 단일 칩상에 방대

한 양의 회로를 집적할 수 있게 되었다. 이렇게 회로가 대형화 될수록 상호결선은 더욱 복잡해지고, 연산 속도의 제한성, 정보전송량의 방대함에 따른 정

보전송시간 지연 및 외부 단자수의 증가로 인한 칩면적의 효율성 저하 및 단자수의 제한 등의 문제점들이 대두되기 시작하였다.^[1-4] 그 해결방안의 하나로 한 개의 단자 수와 결선으로 2치논리보다 더 많은 양의 정보를 처리할 수 있는 다치논리이론에 대한 연구가 1970년대 초부터 활발히 진행중이며 이중 일부는 점차 실용화 되고 있는 현실이다.^[1-7]

특히 최근에는 그래프이론^[8,9]에 바탕을 두고 디지

* 正會員, 仁荷大學校 電子工學科

(Dept. of Electronic Eng. In-Ha Univ.)

** 正會員, 忠州産業大學校 컴퓨터工學科

(Dept. of Computer Eng. Chung-Ju National Univ.)

接受日字:1998年1月12日, 수정완료일:1998年2月27日

탈논리시스템을 해석하려는 연구가 시도되고 있다.

S.B.Aker^[10]는 최초로 방향성 비순환 그래프(Directed Acyclic Graph: DAG) 형태의 데이터구조인 결정도(Decision Diagram: DD)의 개념을 정립하였으며 R.E.Bryant^[11]는 S.B.Aker가 제안한 개념으로부터 최초로 2치결정도(Binary Decision Diagram: BDD)를 사용하여 부울함수구성의 추출 및 축약에 적용하였다. 또한, D.M.Miller^[12]는 S.B.Aker가 제안한 개념을 확장하여 다치논리결정도(Multiple-Valued Logic Decision Diagram: MDD)의 개념을 정립하였으며 이로부터 다치논리함수구성의 추출 및 축약에 적용하였다. 앞에서 언급한 연구들은 각각 그래프에 바탕을 둔 2치논리시스템 및 다치논리시스템의 연구의 효시가 되고 있으며 그 후 다수의 연구들이 발표되고 있다.^[13-16]

한편, S.B.K.Vrudhula등^[17,18]은 기존의 BDD가 부울함수식만을 표현하는 단점을 보완하여 새로운 확장체인 에지값 2치결정도(Edge Valued BDD: EVBDD)를 제안하였는데, 이는 부울표현식의 벡터를 필요로 하는 대신 더 높은 레벨의 특성을 갖는 상태에서 도출될 수 있으며 계층적으로 검증할 수 있다는 장점이 있으므로 임의의 정수함수식에 대한 함축적이고 전형적(canonical)인 표현방식이다.

그러나 기존에 제안된 EVBDD는 2치논리상에서 정의되어 있으므로 정수입력을 갖는 입력변수의 범위가 큰 경우에는 변수를 2치 벡터로 표시하여야한다. 이러한 경우 다치논리를 적용하면 디지털 감소의 효과를 생각할 수 있다.

디지털의 감소는 노드의 감소를 의미하고 실제 회로 설계시 DD상의 노드는 게이트에 해당하므로 노드수의 감소를 통한 설계비용감소 및 처리속도의 감소 등의 성능개선을 볼 수 있다.

본 연구에서는 EVBDD의 장점을 유지하면서, 다치논리함수를 구현하는데 적합한 전류모드 CMOS를 이용한 Literal함수를 통해서 EVBDD를 EMVDD로 전환하는 알고리즘을 제안하고, 이를 바탕으로 임의의 m 치 n 변수의 다치논리함수구성 및 함수축약에 적용하는 한가지 방법을 제안하였다.

다치논리를 적용한 결과 2진논리의 경우에 비해서 같은 범위의 정수입력에 대해서 적은 디지털로 데이터를 처리할 수 있는 다치논리의 장점에 의해 노드 수가 감소하였다.

결정도에서는 노드의 개수는 실제 설계상에 있어서는 소자의 개수에 연관되어 있으므로 결정도 구성시 노드수를 줄이는 방법은 여러 가지 알고리즘에 의해 연구되어지고 있다.

본 논문에서는 노드수의 감소에 의해서 처리속도의 감소, 회로구성의 간략화 및 비용감소 등의 장점을 도출하였다.

노드별로 동일하게 적용되는 에지화 과정을 하나의 모듈로 구성하여 각각의 노드에 반복적용하여 제안된 알고리즘이 정규성이 있음을 보였고 모드계산을 이용한 다치논리 전가산기의 설계를 제안하고 이를 EMVDD로 도시하였다.

본 논문의 서술과정은 다음과 같다, II장에서는 EVBDD에 대하여 소개하였고 III장에서는 본 논문의 수식전개의 바탕이 되는 수학적 배경을 논의하였다. 그리고 IV장에서는 EMVDD를 정의하고 EVBDD로부터 EMVDD를 생성하는 알고리즘을 제안하였으며 V장에서는 EMVDD를 다치 다변수인 경우로 확장하여 일반식을 도출하였다. VI장에서는 제안된 알고리즘의 모듈구성을 통한 정규성에 대하여 논하였다. VII장에서는 다치논리 전가산기와 적용예를 통하여 제안한 알고리즘의 적용결과를 기존의 연구와 비교 및 검토 하였으며, 마지막으로 VIII장에서는 결론을 맺었다.

II. 에지값 결정도^[17,18]

EVBDD는 BDD의 확장으로 산술연산함수의 효과적인 처리를 수행한다. 변수의 브랜치를 에지화한다는 것은 2치결정도에 비해서 2치결정도가 부울연산 표현식의 벡터들을 필요로 하는 반면에 에지변수화 결정도는 좀 더 높은 레벨의 특성을 갖는 상태에서 유도될 수 있고 계층적 검증이 가능하다는 장점을 갖는다.

EVBDD는 부울연산식을 정수함수식으로 전환하여 결정도를 생성하는 과정에 중점을 두고 있기 때문에 2진논리상에서 나타나는 한계를 벗어나지 못하였다.

본 논문에서는 EVBDD의 장점을 유지하면서 2진에 있어서의 노드 에지화를 확장하여 다치논리를 적용하는 알고리즘에 대하여 논하였다.

1) 부울 표현식을 산술연산함수로의 전환
정수 상에서 0과 1이 부울대수의 참(true)과 거짓(false)을 각각 나타냄을 이용하여 부울연산을 산술연

산으로 바꾸는 기본 수식은 다음 식들과 같다.

$$x \wedge y = xy, x \oplus y = x + y - 2xy$$

$$x \vee y = x + y - xy, x' = 1 - x$$

따라서 부울함수는 정수함수의 특별한 경우로 볼 수 있고 BDD는 EVBDD의 특별한 경우로 볼 수 있다.

2) EVBDD의 변수확장

EVBDD의 노드에는 각각 변수가 할당되어 변수 값에 따라서 브랜치가 결정된다. 변수는 모두 $\{0, 1\}^n$ 의 값을 가지며 만일 함수 중의 변수가 2치값을 갖지 않고 정수의 값으로 입력된다면 해당 변수를 2치 벡터로 표현해야한다.

예를 들어, 함수 $f(x, y, z) = 3x + 2y - 9z$ 이 $x \in \{0,1,2,3,4,5\}$ 와 $y, z \in \{0,1\}$ 에서 정의되어있다면 그림1-a의 x노드는 $x = 4x_2 + 2x_1 + x_0$ 와 같이 전환하여 다시 표현되어야 한다. 이때 노드 x에 3치를 적용하면 $x = 2x_1 + x_0$ 로 전환되어 3개의 비트로 표현된 식이 2개의 3치 디지털로 표현이 가능하며 이는 노드수의 감소를 의미한다.

만일 세변수 모두에 정수입력이 정의되어 있다면 더 많은 노드수의 감소를 기대할 수 있다. 여기서 그림 1-c는 x변수에 대한 예지변수화 3치 결정도(Edge Ternary-Valued DD : ETVDD)이다. ETVDD는 EMVDD의 한가지 예로서 입력변수 처리를 0, 1, 2의 3치로 하는 경우이다.

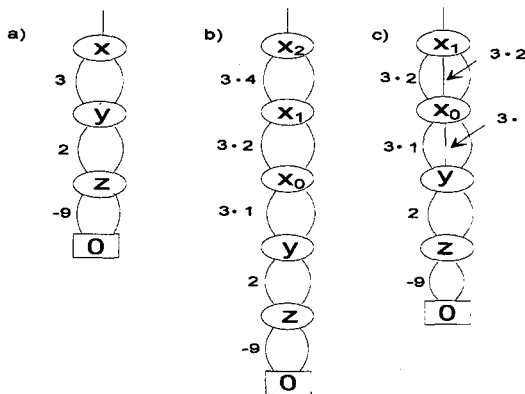


그림 1. (a) 함수 $f(x,y,z)=3x+2y-9z$ 에 대한 EVBDD (b) x변수의 정수입력에 대한 변환 EVBDD (c) x변수의 정수입력에 대한 변환 ETVDD
Fig. 1. (a) EVBDD for the given function (b) EVBDD for x integer input variable (c) ETVDD for x integer input variable

III. 수학적 배경 [19-21]

1. Literal 함수

BDD상의 2치를 다치로 확장하기 위한 기본개념이 되는 literal 함수를 다음과 같이 정의한다.

<정의 1>

X_i 를 집합 $P_i = \{0, 1, 2, \dots, (p-1)\}$ 중의 임의의 값을 취하는 다치변수라 한다. 임의의 부분집합 $S_i \subseteq P_i$ 에 대하여, $X_i^{S_i}$ 를 다음 식(1)의 내용을 표현하는 literal이라 한다.

$$X_i^{S_i} = \begin{cases} 1 & \text{if } \in S_i \\ 0 & \text{if otherwise} \end{cases} \quad (1)$$

위의 정의에 따라 예를 들어 $P = \{0,1,2,3\}$ 일 때 $X^{(1,2)}$ 는 $X = 1$ 또는 2 일 때 1 이고 $X = 0$ 또는 3 일 때 0이다. 따라서 $X^{(1,2)}$ 는 EXOR함수, $xy' \vee x'y = x \oplus y$ 를 나타낸다.

가령 3치 입력을 2치로 출력하는 literal 표현은 식 (1)에 의해 $P = \{0,1,2\}$ 일 때 $X^{(2)}$ 이고 이는 입력이 $\{0,1,2\}$ 일 때 출력은 $\{0,1\}$ 로 되는 함수이고 이를 회로 실현하면 다음의 그림2.(a)와 같고 입출력에 대한 회로 시뮬레이션 결과는 그림2.(b)와 같다.

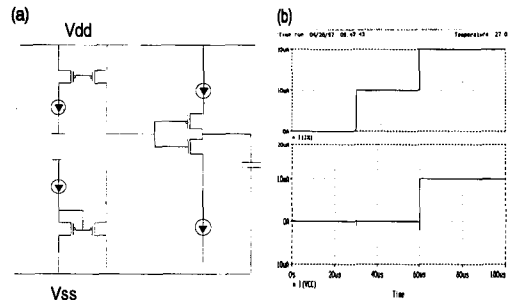


그림 2. Literal함수의 CMOS회로도와 입출력파형
Fig. 2. CMOS circuit and input-output simulation for the ternary input-binary output literal function.

2. Reed-Muller expansion

보다 일반적인 경우인 m 치 n 변수로 확장하는 방법은 부울연산식을 모듈연산에 의해서 확장시킨 Reed-Muller expansion으로 도출할 수 있으며 이로부터 다치 다변수 예지값 결정도의 일반식을 유도한다.

부울연산 내의 논리합과 논리곱은 모듈연산에 있어서의 모듈2합과 모듈2곱으로 대체될 수 있음을 이용하

여 논리곱과 논리적으로 구성되어 있는 부울연산식의 일반식을 모듈합과 모듈적으로 재배열한 것을 Reed-Muller expansion이라 하고 그 일반식은 다음식과 같다.

$$F(x_n, \dots, x_1) = \sum_{i=0}^{2^n-1} C_i x_n^{e_{i,n}} x_{n-1}^{e_{i,n-1}} \dots x_1^{e_{i,1}} \text{ over GF} \quad (2)$$

여기서, $e_{i,j}$ 는 0 또는 1 이고 C_0 는 상수항이다. 계수 선택은

$$T_1 = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}, \quad T_n = \begin{pmatrix} T_{n-1} & 0 \\ T_{n-1} & T_{n-1} \end{pmatrix}$$

로써 선택할 수 있다.

IV. 에지값 다치논리 결정도의 정의

1. EMVDD의 정의

본 절에서는 2치 상에서 정의된 EVBDD를 m치로 확장하기 위해서 각각의 노드에 m개의 브랜치를 할당하고 리터럴함수를 적용하여 변수를 에지화 하기위한 기본 개념을 정의한다.

<정의 2>

EMVDD는 $\langle c, f \rangle$ 의 쌍으로 구성되며 c는 상수 값이며 f는 다음 2가지 경우의 노드를 갖는 방향성 비순환 그래프이다.

2-1) 0에 의해서 표현되는 하나의 터미널 노드 0가 존재한다.

2-2) 터미널이 아닌 노드 v는 $\langle \text{variable}(v), \text{child}_1(v), \text{child}_{M_1}(v), \dots, \text{child}_{M_{m-2}}(v), \text{child}_r(v), \text{value} \rangle$ 의 m+2 개의 쌍으로 구성되며 variable(v)는 다치 변수이다.

<정의 3>

에지값 m치 결정도 $\langle c, f \rangle$ 는 산술 연산 함수 $c+f$ 를 나타내며 f 는 f 에 의해서 나타내지는 함수이다. 0 는 상수함수 0를 나타내며 $\langle x, l, M_1, \dots, M_{m-2}, r, v \rangle$ 는 산술연산함수 $x(1+v)+(x-1)M_1+\dots+(x-(m-2))M_{m-2}+(x-(m-1))r$ 이다. 각각의 노드에는 식(1)의 literal함수를 적용하여 변수입력이 M인 경우만을 에지화 시킨다.

2. EVBDD에서 EMVDD를 생성하는 알고리즘
정의된 EMVDD를 EVBDD로부터 생성하는 과정

을 에지화 되는 노드에 중점을 두고 브랜치 삽입, 리터럴함수 적용, 터미널 노드에 이를 때까지 반복적용 등의 단계별로 기술하면 다음과 같다.

단계1) 터미널 노드가 아닌 노드에서는 EMVDD의 산술연산함수를 나타내는 m개의 쌍 $\langle x, l, M_1, \dots, M_{m-2}, r, v \rangle$ 를 만족하는 노드로 만들기 위해서 중간 (middle) 브랜치를 삽입한다.

단계2) 중간 브랜치를 삽입한 후 각각의 노드의 산술 연산 함수는 $x(1+v) + (x-1)M_1 + \dots + [x - (m-2)]M_{m-2} + [x - (m-1)]r$ 로 전환 된다.

단계3) 각각의 노드에 스톱쉬홀드 검출기(threshold detector)인 literal 함수를 적용하면 다음식 (A)가 된다. 예를 들어 3치인 경우 $m=2$ 이므로 변수 x의 입력이 2이면 에지화되어 브랜치가 가중치를 갖고 그외에 0, 1이면 비에지화되어 브랜치가 가중치를 갖지않는다.

$$X = \begin{cases} 1 & \text{if } x = m \\ 0 & \text{otherwise} \end{cases} \quad (A)$$

위 단계2)의 논리식에 의해서 입력 변수의 값이 m 치일 때만 에지화되어 논리 1의 값을 갖게되고 나머지 브랜치는 논리 0인 비에지화 브랜치로 된다.

V. 에지값 다치결정도의 일반화된 함수식

III장 2절에서의 n변수 함수식인 전개식에서 각항의 계수의 개수는 다음과 같다.

$$F(x_1, \dots, x_n) \rightarrow \begin{aligned} & {}_n C_0 : \text{상수항 계수의 개수} \\ & {}_n C_1 : \text{1차항 계수의 개수} \\ & {}_n C_2 : \text{2차항 계수의 개수} \\ & \vdots \\ & {}_n C_{n-1} : \text{n-1차항 계수의 개수} \\ & {}_n C_n : \text{n차항 계수의 개수} \end{aligned}$$

여기서, ${}_n C_0 + \dots + {}_n C_n =$

$$\sum_{i=0}^n {}_n C_i = 2^n - 1$$

따라서, 일반식은 아래와 같이 표현할 수 있다.

$$F(x_1, \dots, x_n) = C_0 + \sum_{i=1}^{2^n-1} C_i x_1^{e_{i,1}} x_2^{e_{i,2}} \dots x_n^{e_{i,n}} \quad (2)$$

이제 위 식(2)에 리터럴 함수(A)를 적용하여 주어진 변수의 순서에 따라 예지화하는 과정은 다음과 같다.

- f_{x_1} : x_1 의 변수 예지화
- f_{x_2} : x_2 의 변수 예지화
- ⋮
- f_{x_n} : x_n 의 변수 예지화

위 과정으로부터 각각의 예지화 노드를 추출하기 위해 먼저 변수 x_1 에 대해 예지화하는 과정은 다음과 같다. 우선 2차인 경우의 예지화는 식(3)과 같고 m차로 확장하면 다음 식(4)와 같다.

$$f_{x_1} = C_0 + x_1 \cdot (C_1 + \sum_{i=2}^{2^n-1} C_i x_2^{e_{i,2}} x_3^{e_{i,3}} \dots x_n^{e_{i,n}} + \sum_{j=2}^{2^n-1} C_j x_2^{e_{j,2}} x_3^{e_{j,3}} \dots x_n^{e_{j,n}}) + (1 - x_1) \cdot (\sum_{j=2}^{2^n-1} C_j x_2^{e_{j,2}} x_3^{e_{j,3}} \dots x_n^{e_{j,n}}) \quad (3)$$

$$f_{x_1} = C_0 + x_1 \cdot (C_1 + \sum_{i=2}^{2^n-1} C_i x_2^{e_{i,2}} x_3^{e_{i,3}} \dots x_n^{e_{i,n}} + \sum_{j=2}^{2^n-1} C_j x_2^{e_{j,2}} x_3^{e_{j,3}} \dots x_n^{e_{j,n}}) + (x_1 - 1) \cdot (\sum_{j=2}^{2^n-1} C_j x_2^{e_{j,2}} x_3^{e_{j,3}} \dots x_n^{e_{j,n}}) \vdots \vdots + (x_1 - (m-1)) \cdot (\sum_{j=2}^{2^n-1} C_j x_2^{e_{j,2}} x_3^{e_{j,3}} \dots x_n^{e_{j,n}})$$

$$\equiv C_0 + x_1 \cdot (C_1 + \sum_{i=2}^{2^n-1} C_i x_2^{e_{i,2}} x_3^{e_{i,3}} \dots x_n^{e_{i,n}} + \sum_{j=2}^{2^n-1} C_j x_2^{e_{j,2}} x_3^{e_{j,3}} \dots x_n^{e_{j,n}}) \vdots \vdots + \sum_{k=1}^{m-1} (x_1 - k) \cdot (\sum_{j=2}^{2^n-1} C_j x_2^{e_{j,2}} x_3^{e_{j,3}} \dots x_n^{e_{j,n}}) \quad (4)$$

여기서, 상수항을 고려하기 위해서 $f_{x_0} = f_0 = C_0$ 라고 놓으면 주어진 n변수 일반식은 다음 식(5)와 같다.

$$F(x_n, \dots, x_1) = \sum_{i=0}^m f_{x_i} \quad (5)$$

여기서,

$$f_{x_0} = f_0 = C_0$$

$$f_{x_1} = \sum_{i=1}^{2^n-1} C_i x_1^{e_{i,1}} x_2^{e_{i,2}} \dots x_n^{e_{i,n}}$$

$$f_{x_2} = \sum_{i=2}^{2^n-1} C_i x_2^{e_{i,2}} x_3^{e_{i,3}} \dots x_n^{e_{i,n}}$$

$$\vdots \quad \vdots$$

$$f_{x_{n-1}} = \sum_{i=n-1}^{2^n-1} C_i x_{n-1}^{e_{i,n-1}} x_n^{e_{i,n}}$$

$$f_{x_n} = \sum_{i=n}^{2^n-1} C_i x_n^{e_{i,n}} \text{ 이다.}$$

위 내용에서 상수항인 f_{x_0} 를 제외한 각각의 변수에 대하여 m차로 예지화 시키면 다음 항이 앞의 항에서 반복하여 나오는 형태가 되므로 이를 정리하면 다음 식(6)과 같다.

$$f_{x_1} = x_1(C_1 + f_{x_2} + f_{x_2}') + \sum_{k=1}^{m-1} (x_1 - k) f_{x_2}'$$

$$f_{x_2} = x_2(C_2 + f_{x_3} + f_{x_3}') + \sum_{k=1}^{m-1} (x_2 - k) f_{x_3}'$$

$$\vdots \quad \vdots$$

$$f_{x_{n-1}} = x_{n-1}(C_{n-1} + f_{x_n} + f_{x_n}') + \sum_{k=1}^{m-1} (x_{n-1} - k) f_{x_n}' \quad (6)$$

따라서 m치 n변수 에지변수화 일반식은 다음과 같이 표현할 수 있다.

$$F(x_1, \dots, x_n) \equiv f_{x_i} = x_i(C_i + f_{x_{i+1}} + f_{x_{i+1}}') + \sum_{k=1}^{m-1} (x_i - k) f_{x_{i+1}}' \quad (i = 1, 2, \dots, n)$$

여기서 $f_{x_0} = f_0 = C_0$ 이고 f_{x_i}' 는 비에지화 브랜치가 갖는 함수식이다.

또한, 위의 일반식을 EMVDD로 그리면 그림3과 같다.

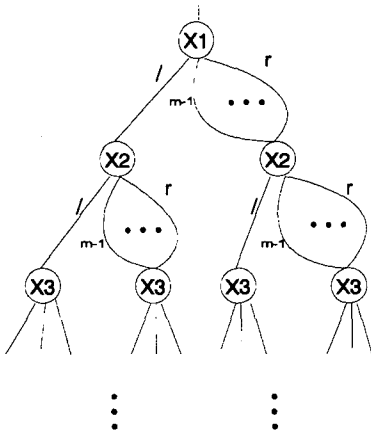


그림 3. m치 n변수의 일반식에 대한 EMVDD
Fig. 3. General structure of an EMVDD.

VI. 에지화 모듈 블럭도

각각의 변수에 대하여 리터럴 함수를 적용하여 에지화하고 다음단의 함수식을 찾아내는 과정을 하나의 모듈로 생각하여, 만일 n개의 변수입력이 있다면 한개의 모듈과 상수항으로 모듈화 해서 블럭도를 그릴 수 있다. 예를 들어 n개의 변수입력을 갖는 함수식에서 f_{x_i} 를 변수를 포함하는 함수식 중에서 x_i 변수를 제외한 함수식이라 하고 f_{x_i}' 을 x_i 변수를 포함하지 않는 함수식이라할 때 변수 x_i 에 대한 모듈구성은 다음 그림 4와 같고 여기서 모듈 i는 다음단 변수의

모듈 i+1의 입력 모듈이 된다.

변수의 개수가 n개인 경우는 상수항을 제외하고 각각의 변수에 동일한 모듈을 적용하여 반복함으로써 모든 변수를 에지화할 수 있다.

VII. 전가산기 구성 및 적용예

1. 전가산기 설계

1) 2치 전가산기

3변수의 전가산기를 구현할 경우 이를 수식으로 표현하면 다음 식(7)과 같고 EVBDD로 도시하면 그림 5와 같다.

모듈 i

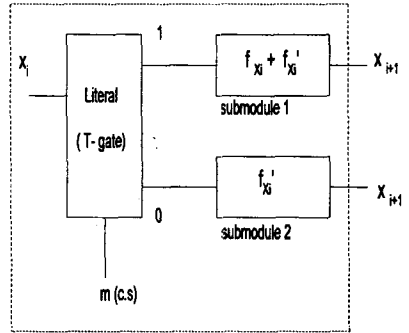


그림 4. 변수 x_i 에 대한 에지변수화 블럭도
Fig. 4. EMVDD block diagram for variable x_i .

sum (x, y, z) , carry (x, y, z)

sum (x, y, z) = $\omega_1 = f_a \text{ mod } 2$

carry (x, y, z) = $\omega_2 = (f_a - (f_a \text{ mod } 2)) / 2$

$f_a = x + y + z$

spec (x, y, z) = $\omega_1 + 2 \omega_2$

$x + y + z = \omega_1 + 2 \omega_2$

$= \text{sum} (x, y, z) + 2 \text{carry} (x, y, z) \quad (7)$

2) 3치 전가산기의 설계

기본개념은 2치의 경우와 같고 mod계산의 경우만 달리하면 쉽게 3치인 경우의 전가산기를 설계할 수 있다. 이를 수식으로 나타내면 식(8)과 같다.

sum (x, y, z) = $\omega_1 = f_a \text{ mod } 3$

carry (x, y, z) = $\omega_2 = (f_a - (f_a \text{ mod } 3)) / 3$

$$\begin{aligned}
 f_a &= x + y + z \\
 \text{spec} (x, y, z) &= \omega_1 + 3 \omega_2 \\
 x + y + z &= \omega_1 + 3 \omega_2 \\
 &= \text{sum} (x, y, z) \\
 &\quad + 3\text{carry} (x, y, z)
 \end{aligned} \tag{8}$$

$$\begin{aligned}
 \text{carry} (x, y, z) &= \omega_2 \\
 &= (f_a - (f_a \bmod m)) / m \\
 f_a &= x + y + z \\
 \text{spec} (x, y, z) &= \omega_1 + m \omega_2 \\
 x + y + z &= \omega_1 + m \omega_2 = \text{sum} (x, y, z) \\
 &\quad + m \text{carry} (x, y, z)
 \end{aligned} \tag{9}$$

입력변수의 범위가 커지는 경우에 다치논리를 적용하면 입력변수의 디지털을 줄일 수 있고 이는 처리속도의 감소를 의미하며 범위가 커질수록 더 큰 효과를 기대할 수 있다.

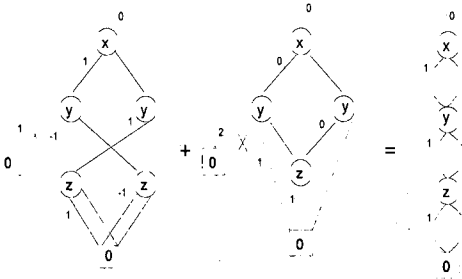


그림 5. EVBDD를 통한 전가산기 설계
Fig. 5. The construction of the full adder using EVBDD.

3) m치 전가산기의 구현
3치에서 m치로 확장하는 경우에도 3치와 마찬가지로 리터럴 함수를 적용하여 m치로 확장된 sum과 carry로 전가산기를 구현하는 식은 아래 식(9)와 같고 이를 EMVDD로 도식하면 그림 6과 같다. 변수마다 리터럴 함수의 적용이 반복되므로 모듈을 통한 구성에 적합함을 알 수 있다.

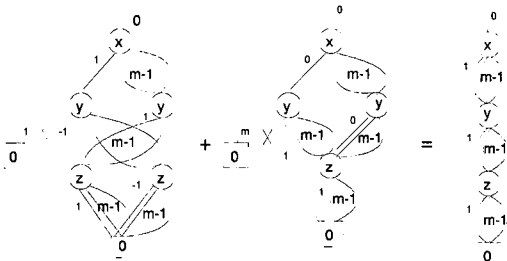


그림 6. EMVDD를 통한 전가산기의 설계
Fig. 6. The construction of the full adder using EMVDD.

$$\text{sum} (x, y, z) = \omega_1 = f_a \bmod m$$

2. 적용예를 통한 검증

다음 예제를 통해 IV장의 알고리즘을 적용하여 각각 EVBDD와 ETVDD를 표현하는 식을 구하고, 이의 결과를 비교 검토하면 다음 표2와 같고 이를 BDD, EVBDD 및 ETVDD로 도식하면 다음 그림 7과 같다.

예제) 함수 $f(x, y, z) = -2 + 5y + yz + 3xy + 4xyz - 2xz + z$ 의 ETVDD를 구성하면 다음과 같다. 우선 함수 f를 EVBDD로 표현하면 다음 식(10)과 같다.

$$\begin{aligned}
 f &= -2 + y(5 + 3x + 2xz + 2z) + (1-y)(-2xz + z) \\
 &= -2 + y[5 + x(3 + 4z) + (1-x)(2z)] \\
 &\quad + (1-y)[x(-z) + (1-x)(z)] \\
 &= -2 + y\{5 + x[3 + z(4)] + [1-x][z(2)]\} \\
 &\quad + (1-y)\{x[z(-1)] + [1-x][z(1)]\}
 \end{aligned} \tag{10}$$

여기에 IV장의 literal 함수 (A)를 적용시켜 ETVDD로 표현하면 다음 식(11)과 같다.

$$\begin{aligned}
 f &= -2 + y\{5 + x[3 + z(4)] + [x-1][z(2)] + [x-2][z(2)]\} \\
 &\quad + (y-1)\{x[z(-1)] + [x-1][z(1)] + [x-2][z(1)]\} \\
 &\quad + (y-2)\{x[z(-1)] + [x-1][z(1)] + [x-2][z(1)]\}
 \end{aligned} \tag{11}$$

3. 적용예를 통한 비교검토

그림 7의 a), b)와 c)를 살펴보면 모두 DD의 성질을 가지고 있으므로 가시적이고 각각의 변수에 대하여 같은 성질을 보이므로 규칙적이고 변수에 리터럴함수를 적용하는 과정을 모듈화하여 정규성을 갖는다. 만일 표2의 내용을 II-2-2)에서의 예의 그림1을 참조하면 세변수 x, y, z 모두에서 1개씩의 노드수가 감소되는 것을 알 수 있다.

위의 예에서 만일 변수의 개수가 많아지고 변수의 입력범위가 커지면 다치논리 적용으로 인한 노드수의

감소효과는 더욱더 커짐을 알 수 있다. 3변수인 경우는 EVBDD에서 ETVDD으로 전환할 경우 7개의 노드감소, 5변수일 경우에는 31개의 노드감소를 볼 수 있다.

EMVDD 상에서의 노드는 데이터처리를 다치논리에 근거를 둔다. 따라서, 그래프 상에서의 노드에 따르는 브랜치로 인한 구조에서의 복잡도는 실제 데이터 처리에서의 복잡도로 연관되어지지 않는다. 본 논문에서는 소자수의 감소를 통한 성능향상에 집중했다.

표 2. 예제2의 함수에 대한 결정도별 노드수 비교

Table 2. The comparison table for each DDs.

	BDD	EVBDD	정수입력 EVBDD	정수입력 ETVDD
노드수	$2^{n+1} - 1$	2^n	$n \cdot (2^n - 1)$	$(n-1) \cdot (2^n - 1)$

(단, 변수의 순서는 동일하다고 가정, n : 변수의 개수)

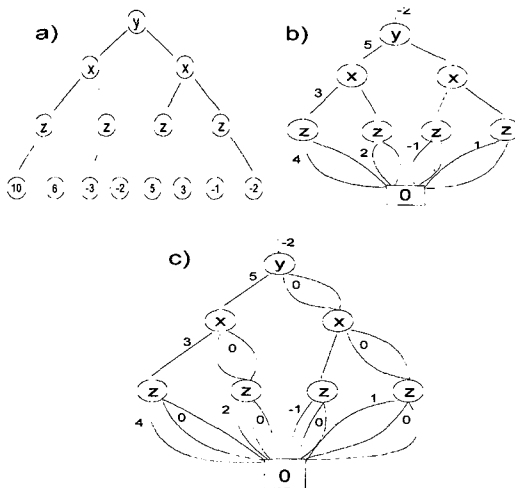


그림 7. (a) 예제의 함수에 대한 BDD^[18] (b) 예제의 함수에 대한 EVBDD^[18] (c) 예제의 함수에 대한 ETVDD

Fig. 7. (a) BDD for the function^[18] (b) EVBDD for the function^[18] (c) ETVDD for the function.

VIII. 결론

본 논문에서는 최근에 디지털논리시스템의 함수구성에 도입되고 있는 그래프이론에 바탕을 둔 BDD와 이의 새로운 데이터구조 형태인 EVBDD로부터 임의의 m차 n변수의 함수를 다룰 수 있는 EMVDD를 도

출하는 한 가지 방법에 대하여 논의하였으며, 예제를 통하여 타당성을 살펴보았다.

제안된 EMVDD를 사용하여 노드수를 줄일 수 있었으며, 이는 함수의 축약과 회로의 간략화를 수행할 수 있음을 의미하며 처리속도의 감소를 기대할 수 있다. 또한 알고리즘의 규칙성을 고려하여 모듈구성을 통한 정규성에 대한 기능성에 관하여 논하였다.

향후 연구과제로는 제안한 EMVDD를 다치논리의 기호화된 시뮬레이션, 조합논리시스템 및 순차논리시스템 등의 분야에 효과적으로 적용하는 부분의 연구가 요구된다.

참고 문헌

- [1] M. Kameyama, "Toward the Age of the Beyond-Binary Electronics and System," ISMVL 90, pp. 162-166, North Carolina U.S.A., May. 1990.
- [2] K. C. Smith and P. G. Gulak, "Prospects for multiple-valued intergrated circuits," Special issue on Multiple-Valued intergrated circuits IEICE TRANS. ELECTRON., vol. E76-C, no. 3, pp. 372-382, Mar. 1993.
- [3] S. L. Hurst, "Multiple-Valued logic - its status and its future," IEEE Trans. Comput., vol. C-33, pp. 1160-1179, Dec. 1984.
- [4] T. Hanyu, M. Kameyama, T. Higuchi, "Prospects of Multiple-Valued VLSI Processors," IEICE Trans. Electron, vol. E76-C, no. 3, pp. 383-392, March 1993.
- [5] David C. Rine, *Computer Science and Multiple-Valued Logic Theory and Applications*, North-Holland Publishing Company, 1977.
- [6] G. Epstein, *Multiple-Valued Logic: an Introduction*, Institute of physics publishing, 1993.
- [7] D. Etiemble, "On the Performance of Multivalued intergrated Circuits: Past, Present and Future," ISMVL 92, pp. 156-164, Sendai Japan, May. 1992.
- [8] R. Gould, *Graph Theory*, The Benjamin/Cummings Publishing Company, Inc.

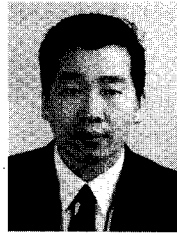
- 1988.
- [9] R. J. Wilson and J. J. Watkins, *Graph an Introductory Approach*, John Wiley & Sons, Inc. 1990.
- [10] S. B. Aker, "Binary Decision Diagrams," *IEEE Trans. Comput.*, vol. C-27, no. 6, pp. 509-516, Jun. 1978.
- [11] R. E. Bryant, "Graph-Based Algorithms for Boolean Function manipulations," *IEEE Trans. Comput.*, vol. C-35, no. 8, pp. 677-691, Aug. 1986.
- [12] D. M. Miller, "Multiple-Valued Logic Design Tools," *IEEE Proc. of Symposium on Multiple-Valued Logic*, Sacramto, California, pp. 2-11, May. 1993.
- [13] J. Gergoy and C. Meinel, "Efficient Boolean Manipulation with OBDD's can be Extended to FBDD's," *IEEE Trans. Comput.*, pp. 1197-1209, vol. 43, no. 10, Oct. 1994.
- [14] T. Sasao and J. T. Butler, "Planar Multiple-Valued Decision Diagrams," *ISMVL 95*, pp. 28-35, Bloomington, May 23-25, 1995.
- [15] 朴春明, 金興壽, "決定 다이어그램에 의한 多值組合論理 시스템 構成에 관한 研究," *電子工學會論文誌 第32卷 B編 第6號*, pp. 48-55, 1995
- [16] T. Sasao, Jon T. Butler, "A Method to Represent Multiple-Output Switching Functions by Using Multi-Valued Decision Diagrams," *IEEE 26th proceedings on ISMVL*, pp. 248-254, 1996.
- [17] Yung-Te Lai and Sarma Sastry, "Edge-Valued Binary Decision Diagrams for Multi-Level Hierarchical Verification" *29th ACM/IEEE Design Automation Conference*, pp. 608-613, 1992.
- [18] Yung-Te Lai, M. Pedram and S.B.K. Vrudhula, "Formal Verification Using Edge-Valued Binary Decision Diagrams," *IEEE Trans. on Computers*, vol. 45, no.2, February 1996.
- [19] D. Green, *Modern Logic Design*, Addison-Wesley Publishing Company, 1988.
- [20] M. Davio, Jean-Pierre, Deschamps and Andre Thayse, *Discrete and Switching Functions*, McGraw-Hill international Book company, 1978.
- [21] R. J. McEliece, *Finite Fields for Computer Science and Engineers*, Kluwer Academic Publishers, 1987.

저 자 소 개



韓 聖 一(正會員)

1995년 2월 인하대학교 전자공학과 졸업(공학사). 1995년 3월 인하대학교 대학원 전자공학과 석사학위과정 입학. 1998년 2월 인하대학교 대학원 전자공학과 졸업(공학석사). 1998년 3월 현재 대우통신연구소 근무. 주관심분야는 다치논리회로 구성, VLSI 설계, FUZZY control 등임



崔 在 碩(正會員)

1988년 2월 인하대학교 전자공학과 졸업(공학사). 1990년 2월 인하대학교 대학원 전자공학과 졸업(공학석사). 1990년 8월 인하대학교 대학원 전자공학과 졸업(공학박사). 1990년 1월 ~ 1995년 4월 (주)기아정보시스템개발연구소 근무. 주관심분야는 다치논리시스템, VLSI 설계, DSP & DIP 등임



朴 春 明(正會員)

1983년 2월 인하대학교 전자공학과 졸업(공학사). 1986년 2월 인하대학교 대학원 전자공학과 졸업(공학석사). 1994년 2월 인하대학교 대학원 전자공학과 졸업(공학박사). 1990년 3월 ~ 1995년 8월 마산전문대학 전자계산학과 조교수. 1995년 9월 ~ 1998년 3월 현재 충주산업대학교 컴퓨터공학과 조교수. 주관심분야는 다치논리이론 및 다치논리시스템설계, 디지털시스템설계, 컴퓨터구조, 마이크로프로세서응용 등임

金 興 壽(正會員) 第 32卷 B編 第 6號 參照

현재 인하대학교 전자공학과 교수