

논문 2009-1-12

다이렉트사이클릭그래프에 기초한 디지털논리시스템 설계

Digital Logic System Design based on Directed Cyclic graph

박춘명*

Chun-Myoung Park

요 약 본 논문에서는 경로수 ζ 로 주어진 DCG(Directed Cyclic Graph)의 입출력간의 연관관계를 고효율디지털논리 회로로 설계하는 알고리즘과 DCG의 각 노드들에 코드를 할당하는 알고리즘을 제안하였다. 본 논문에서는 기존 알고리즘의 문제점을 도출한 후, 다른 접근방법으로써 DCG의 경로수로 부터 행렬방정식을 유도한 후 이를 통해 DCG의 경로수에 따른 회로설계 알고리즘을 제안하였으며, 설계된 회로와 함께 DCG의 특성을 만족하도록 노드들에 대한 코드를 할당하는 알고리즘을 제안하였다. 본 논문에서 제안한 고효율디지털논리회로설계 알고리즘은 기존의 알고리즘으로는 가능하지 않았던 경로수의 DCG에 대하여 회로설계가 가능하게 되었고, 보다 최적화된 디지털논리회로를 구현할 수 있음을 확인하였다. 본 논문에서 제안한 회로설계 알고리즘을 통해 임의의 자연수를 경로수로 갖는 DCG에 대한 설계가 가능하며, 입출력단자 수의 감소, 회로구성의 간략화, 연산속도의 향상과 비용감소 등의 잇점이 있고, 예제를 통해 본 논문에서 제안한 알고리즘의 적합성과 타당성을 검증하였다.

Abstract This paper proposes the algorithms that design the highly digital logic circuit and assign the code to each node of DCG(Directed Cyclic Graph) of length ζ . The conventional algorithm have some problems, so this paper introduce the matrix equation from DCG of length ζ and proposes highly digital logic circuit design algorithms according to the DCG of length ζ . Using the proposed circuit design algorithms in this paper, it become realized that was able to design from former algorithm. Also, making a comparison between the circuit using former algorithm and this paper's, we testify that proposed paper's algorithm is able to realize more optimized circuit design. According to proposed circuit design algorithm in this paper, it is possible to design current that DCG have natural number, so it have the following advantages, reduction of the circuit input/output digits, simplification of circuit composition, reduction of computation time and cost. And we show comparability and verification about this paper's algorithm.

Key Words : graph theory, galois fields, highly digital logic circuit design.

I. 서 론

임의의 디지털논리시스템에서 노드들 간의 관계가 순환특성을 갖는 방향성 그래프(Directed Cyclic Graph: DCG)의 형태로 주어졌을 때, 입출력간의 연관관계를 유한체상에서 정의된 덧셈기와 곱셈기를 통해 디지털논리회로로 구현하면 효과적으로 디지털논리회로를

구현할 수 있다. 기존의 연구[1-2]는 주어진 DCG의 경로의 수를 소수들의 곱으로 인수분해하여 해당 소수에 해당하는 DCG에 대하여 디지털논리회로를 구성한 후, 이들을 선형결합하여 주어진 DCG의 특성을 만족하는 디지털논리회로를 설계하는 기법이다. 그러나 이러한 기법은 임의의 경로의 수를 갖는 DCG에 대한 해석과 그에 따른 디지털논리회로구현에 제약을 가지고 있다. 이러한 문제점을 해결하기 위해, 본 논문에서는 DCG의 경로의 수로부터 입출력간의 연관관계를 나타내는 전달행렬의 방정

*정회원, 충주대학교 컴퓨터공학과
접수일자 2008.12.3, 수정완료일자 2009.2.3

식을 유도하여 이로부터 주어진 DCG특성을 만족하는 수식을 도출한 후, 디지털논리회로를 설계하는 새로운 알고리즘을 제안하였다. 본 논문에서 제안한 알고리즘은 기존의 알고리즘으로는 설계가 불가능했던 DCG의 경우에도 이의 해석과 설계가 가능하며 또한 종래의 연구내용을 통해 설계된 디지털논리회로보다 더욱 최적화된 디지털논리회로[3-5]를 설계할 수 있었다. 또한, 구현된 회로와 함께 주어진 DCG의 특성을 만족시키는 각 상태들의 값을 코드로 할당하는 코드할당 알고리즘도 제안함으로써 DCG에 대한 디지털논리회로설계를 보다 구체화하고 일반화시킬 수 있었다. 본 논문의 구성과 서술과정은 다음과 같다. II장에서는 본 논문을 전개하는데 필요한 그래프성질 및 수학적 배경에 대하여 논의하였으며, III장에서는 DCG에 의한 고효율디지털논리회로를 설계하기 위하여 본 논문에서 제안한 행렬방정식을 유도하는 기법과 이 행렬방정식으로부터 고효율디지털논리회로설계알고리즘과 DCG의 각 노드들에 코드를 할당하는 코드할당 알고리즘에 대하여 논의하였다. IV장에서는 제안한 방법을 적용예를 들었으며, V장의 결론에서는 본 논문에서 다룬 내용의 특징을 요약하였다.

II. 그래프 성질 및 수학적 배경

1. 방향성그래프(DG:Directed Graph)

일반적으로, 그래프는 방향성그래프(DG: Directed Graph)와 비방향성 그래프(Undirected Graph: UG)로 구분되며 이들의 계층적구조에 따라 각각 사이클(cycle)과 트리(tree)로 세분된다. 그래프는 노드(node)들과 이들을 연결한 브랜치(branch)들의 집합으로 정의되며, 특히 브랜치들의 방향이 정의되어 있는 그래프를 방향성 그래프라 한다. 방향성그래프의 몇가지 예를 그림1에 도시하였다.

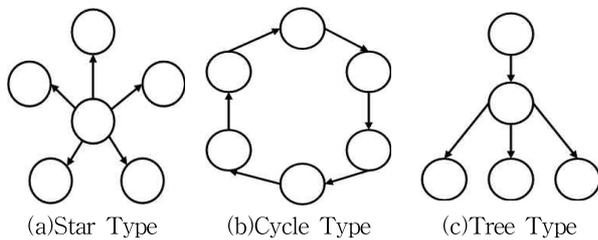


그림 1. 방향성그래프의 예
Fig 1. Examples of Directed Graph

위 그림 1의 (b)와 같이 그래프의 형태가 방향을 가지며 사이클을 이루는 구조를 DCG라 하고 이의 몇가지 성질을 정리하면 다음과 같다.

[성질 1] $s_2 = \Psi s_1$ 의 관계를 갖는 두 노드 s_1, s_2 에 대하여 s_1 을 s_2 의 조상(원인 또는 입력), s_2 를 s_1 에 대한 자손(결과 또는 출력)이라 한다. 이때 s_1 과 s_2 의 관계를 그림으로 나타내면 s_1 에서 출발한 화살표가 s_2 에 도착한 형태로 표현되며, 이때 화살표는 두 노드들 간의 관계를 설명해주는 전달함수(transfer function) Ψ 가 된다.

[성질 2] $s_k = \Psi^m s_1$ 의 관계를 갖는 두 개의 노드 s_k 와 s_1 에 대하여 s_1 은 s_k 의 m번째 조상이 되며, s_k 는 s_1 의 m번째 자손이 된다. 즉, s_1 에서 출발하여 s_k 에 도착하기 위해서는 m번의 Ψ 를 취해야 한다.

[성질 3] $s_1 \neq s_2 \neq \dots \neq s_{i+1}$ 이며 경로 1을 형성하는 1+1개의 노드들 s_1, s_2, \dots, s_{i+1} 에 대하여 $s_{i+1} = s_1$ 의 관계가 성립할 때 이러한 관계를 경로 1의 사이클이라 한다.

이외의 유용한 그래프의 성질은 참고문헌 [6-7]을 참조하였다.

2. 유한체 GF(p)의 성질

프랑스의 수학자 Galois에 의해 발견된 유한체는 일명 Galois체라고도 하며 오류정정부호 및 디지털스위칭이론 등에 광범위하게 적용되는 대수학의 일부분이다. 유한체 GF(p)는 p가 1보다 큰 소수이고 $\{0, 1, 2, \dots, p-1\}$ 를 그 원소로 갖는 기초체이다. 유한체 GF(p)상의 임의의 원소 α, β, γ 는 다음과 같은 성질을 만족한다.

[성질 1] 유한체 GF(p)상의 원소들의 연산에 대하여 가산(\oplus)과 승산(\odot)이 정의되고, 그 연산 결과는 유한체 GF(p)에 대하여 닫혀 있다.

[성질 2] 유한체 GF(p)상의 원소들의 연산에 대하여 교환, 결합, 분배법칙이 성립한다.

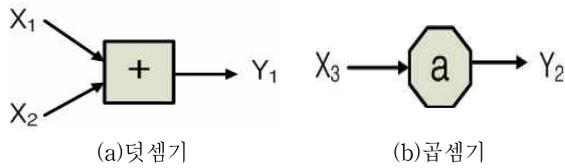
[성질 3] 유한체 GF(p)상의 원소들의 연산에 대하여 가산과 승산의 항등원과 역원이 각각 존재한다.

이외의 유용한 유한체 GF(p)의 성질은 참고문헌[8-9]을 참조하였다.

3. 선형회로의 구성과 코드

선형특성을 만족하는 디지털논리회로를 설계하기 위해, 기본게이트로써 유한체 GF(p)내에서 가산기와 곱셈기를 다음 그림2와 같이 정의한다.

위 그림2의 (a)의 소자는 두개의 입력 x_1, x_2 를 GF(p)상의 덧셈연산을 수행한 후 그 결과를 출력시키는 게이트이며, (b)의 소자는 입력 x_3 를 GF(p)



where, $Y_1 = X_1 \oplus X_2$ and $Y_2 = a \odot X_3$

그림 2. 유한체 GF(p)상의 선형디지털논리게이트
Fig 2. Linear digital logic gates over GF(p)

상에서 a배 만큼 스칼라 곱한 후 그 결과를 출력시키는 게이트이다.

4. 회로의 분할 연산

두 개의 정방행렬 Ψ 와 Ψ' 가 유한체 GF(p)상에서 정의될 때 $\Psi' = \Phi \Psi \Phi^{-1}$ 를 만족하는 가역행렬 Φ 가 존재하면, Ψ' 는 Ψ 와 상사(similar)라 한다.

$n \times n$ 행렬 Ψ 가 존재할 때 Ψ 의 특성다항식의 차수는 n이며, 특성다항식의 최고차항의 계수는 1이다.(이러한 성질을 갖는 다항식을 monic다항식이라 한다.)

$$d(x) = a_0 + a_1 x + \dots + a_{n-1} x_{n-1} + x_n \tag{1}$$

$$\Psi = \begin{pmatrix} 0 & 0 & 0 & \dots & -a_0 \\ 1 & 0 & 0 & \dots & -a_1 \\ 0 & 1 & 0 & \dots & -a_2 \\ & & \vdots & & \\ 0 & 0 & 0 & \dots & -a_{n-1} \end{pmatrix} \tag{2}$$

식(1)의 특성다항식을 갖는 행렬은 식(2)와 같은 행렬로 표현되며, 이때 식(2)의 Ψ 를 특성다항식 $d(x)$ 의 동반행렬(Companion matrix)이라 한다. 그리고 동반행렬들은 선형특성에 의해 식(3)과 같이 선형결합이 가능하다.

$$\Psi = \begin{pmatrix} \Psi_1 & & & \\ & \Psi_2 & & \\ & & \ddots & \\ & & & \Psi_s \end{pmatrix} \tag{3}$$

이외의 회로의 분할연산 및 행렬구성기법에 대한 내용은 참고문헌[14]를 참조하였다.

III. 디지털논리회로의 설계

1. 행렬방정식을 통한 DCG의 해석과 설계

본 절에서는 입출력간의 연관관계(DCG)를 디지털논리회로로 설계하기 위하여 행렬의 방정식을 유도해내고 이로부터 회로를 구현하는 과정에 대하여 논의한다. 이후 『m개의 노드들 간의 입출력 연관관계』 또는 『m개의 노드를 갖는 DCG』를 편의상 『m-노드 DCG』 또는 『 DCG_{m-m} 』으로 나타낸다. 예를 들어 다음 식(4)와 같이 표현된 GF(3)상의DCG 노드에 대해, 밑줄 친 관계식으로부터 다음과 같은 행렬방정식을 유도할 수 있다.

$$\begin{aligned} \Psi \underline{\Xi} = \underline{\Omega}, \quad \Psi \underline{\Omega} = \underline{\Phi} = \Psi \underline{2} \underline{\Xi}, \quad \Psi \underline{\Phi} = \underline{\Xi} = \Psi \underline{3} \underline{\Xi} \\ \Psi^2 \underline{\Xi} = \underline{\Xi} \Leftrightarrow (\Psi^2 - \Theta) \underline{\Xi} = \underline{\Xi} \Leftrightarrow (\Psi - \Theta)(\Psi^2 + \Psi + \Theta) \underline{\Xi} = \underline{\Xi} \end{aligned} \tag{4}$$

(여기서, Θ 는 단위행렬을 나타낸다.)

여기서, DCG를 만족하기 위해 3개의 $\underline{\Xi}, \underline{\Omega}, \underline{\Phi}$ 는 서로 구분되어야 하며($\underline{\Xi} \neq \underline{\Omega} \neq \underline{\Phi}$), 전달행렬 Ψ 는 $\Psi^2 + \Psi + \Theta = \underline{\Xi}$ 의 관계식을 만족해야 한다. 따라서, 전달행렬 Ψ 는 식(5)와 같이 얻을 수 있다.

$$\Psi = \begin{pmatrix} 0 & 2 \\ 1 & 2 \end{pmatrix} \tag{5}$$

식(5)의 전달행렬 Ψ 를 디지털논리회로로 구성하면 그림3과 같다.

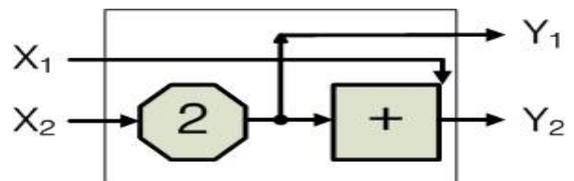


그림 3. GF(3)상의 디지털논리회로
Fig 3. Digital logic circuit over GF(3)

2. 회로설계 알고리즘1과 코드할당 알고리즘

III장의 1절에서 논의한 내용을 체계화하여 디지털논리시스템의 입출력 노드들간의 관계가 DCG의 형태로 주어졌을 때, 이를 디지털논리회로로 설계하고 설계된 디지털논리회로와 함께 주어진 DCG의 특성을 만족하도록 노드들에 코드를 할당해주는 알고리즘은 다음과 같다.

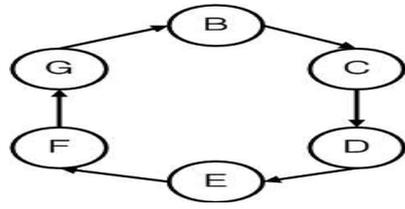


그림 4. 『6-노드 DCG』
Fig 4. 『6-노드 DCG』

[고효율 디지털논리회로설계 알고리즘]

- step 1 : DCG로 부터 경로의 수를 파악한다.
- step 2 : 싸이클의 경로수를 ζ 이라고 하면, 다음과 같이 행렬 ζ 방정식이 구성된다.
$$\Psi^{(\zeta-1)} + \Psi^{(\zeta-2)} + \dots + \Psi^2 - \Theta = \emptyset$$
- step 3 : step2에서 구한 행렬식으로부터 이를 만족하는 행렬을 구한다. 이 행렬을 주어진 DCG의 특성을 만족하는 전달행렬 $\Psi_{\zeta=6}$ 라 한다.
- step 4. 전달행렬 $\Psi_{\zeta=6}$ 를 디지털논리회로로 구현하다. 여기서, 아래첨자는 DCG의 경로의 수를 의미한다.

[고효율 디지털논리회로설계 알고리즘]

- Step 1 : 경로 6의 싸이클
- Step 2 : $\Psi^5 + \Psi^4 + \Psi^3 + \Psi^2 + \Psi^1 - \Theta = \emptyset$

Step 3 : $\Psi_{\zeta=6} =$

0	0	0	0	2
1	0	0	0	2
0	1	0	0	2
0	0	1	0	2
0	0	0	1	2

(6)

Step 4 : 전달행렬 $\Psi_{\zeta=6}$ 를 디지털논리회로로 구현한다.

[코드할당 알고리즘]

- step 1. 위 회로설계 알고리즘을 통해 구한 전달행렬 Ψ 에서 임의의 한 칼럼을 DCG의 노드들 중 임의의 한 노드의 코드로 할당해 준다. 일반적으로, 전달행렬 Ψ 의 첫 번째 열을 DCG의 첫 번째 노드의 코드로 할당해 준다.
- step 2. step 1에서 구한 첫 번째 코드값을 전달행렬 Ψ 와 연산하여 두 번째 노드의 코드값을 구한다.
- step 3. step 2와 같은 방법으로 DCG의 모든 노드의 코드값이 할당될 때까지 이 과정을 반복수행한다.
- step 4. DCG의 모든 노드들의 코드값이 할당되면 이를 표로 정리한다.

[코드할당 알고리즘]

Step 1 : 식(6)의 전달행렬 $\Psi_{\zeta=6}$ 의 1번째 열을 그림4의 『6-노드 DCG』의 노드 B의 코드로 할당한다.

$$E = [01000]^{TR}$$

여기서, TR은 transpose를 의미한다.

Step 2 :

0	0	0	0	2	0	0
1	0	0	0	2	1	0
0	1	0	0	2	0	1
0	0	1	0	2	0	0
0	0	0	1	2	0	0

$\Psi_{\zeta=6} \quad E \quad \Omega$

Step 3 : Step2와 같은 방법으로 나머지 노드들의 코드값을 구하면 다음과 같다.

또한, 위 과정을 통해 구한 그림4의 『6-노드 DCG』의 각 노드들의 코드값을 정리하면 다음 표1과 같다.

IV. 적용 예

본 장에서는 선 앞에서 서술한 내용들이 어떻게 적용되는 지 예를 들어 설명한다. 다음 그림4의 『6-노드 DCG』를 고효율디지털논리회로설계하고 각 노드들에 코드를 할당하면 다음과 같다.

0	0	0	0	2	1
1	0	0	0	2	0
0	1	0	0	2	0
0	0	1	0	2	0
0	0	0	1	2	0
E	Ω	Φ	Θ	Λ	Δ

여기서, $\Xi, \Omega, \Phi, \Theta, \Lambda$ 까지의 드값이 전달행렬 $\Psi_{\zeta=6}$ 의 5개 열벡터들과 같다.

표 1. 『6-노드 DCG』의 각 노드들에 대한 코드할당표
Table 1. Code assignment table for each nodes of 『DCG $_{\zeta=6}$ 』

	Ξ	Ω	Φ	Θ	Λ	Δ
X_1	0	0	0	0	2	1
X_2	1	0	0	0	2	0
X_3	0	1	0	0	2	0
X_4	0	0	1	0	2	0
X_5	0	0	0	1	2	0

Step 4 : 전달행렬 $\Psi_{\zeta=6}$ 를 디지털논리회로로 구현하면 다음 그림4와 같다.

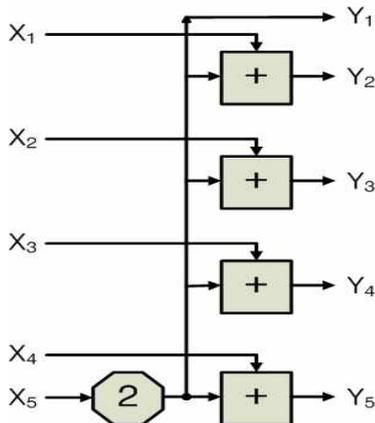


그림 5. 전달행렬 $\Psi_{\zeta=6}$ 를 통해 구현한 $DCG_{\zeta=6}$ 에 대한 GF(3) 상의 디지털논리회로
Fig 5. digital logic circuit over GF(3) for $DCG_{\zeta=6}$ from transfer matrix $\Psi_{\zeta=6}$

V. 결 론

본 논문에서는 최근에 활발히 연구되고 있는 그래프에 기초한 유한체상의 디지털논리시스템 구성 및 회로설계에 관한 연구 중, 노드들의 관계가 사이클의 특성을 갖는 DCG의 경우에 이 노드들간의 관계를 회로로 구현하는 회로설계 알고리즘과 각 노드들에 코드를 할당하는 알고리즘을 제안하였다. 기존의 DCG에 대한 회로설계 알고리즘은 노드들의 개수를 소수들의 곱으로 인수분해한 후, 인수분해된 소수들에 대하여 각각의 회로를 구현한 후 이들을 선형결합하는 방식을 제안하였다. 그러나, 이와 같은 방식은 매우 제한적으로 적용되며 그 해석에

있어 일반성이 결여된 단점이 있다. 예를 들어 노드들의 개수가 소수들의 곱의 수($4=2 \times 2, 8=2 \times 2 \times 2, 9=3 \times 3$ 등)로 주어질 경우, 이에 대한 충분한 해석과 회로설계방법이 제안되지 않았다. 이와 같은 문제점을 해결하기 위해, 본 논문에서는 DCG의 경로수로부터 행렬방정식을 유도하여 DCG의 특성을 만족하는 수식을 도출하였고, 이 수식을 정리함으로써 DCG의 특성을 만족하는 회로를 설계하였다. 또한, 주어진 DCG의 입출력관계를 구현한 회로와 함께 DCG를 구성하고 있는 노드들에 적합한 코드를 할당해 줌으로써 회로의 동작을 보다 분명히 정의할 수 있는 장점이 있다.

참 고 문 헌

- [1] M.Nakajima and M.Kameyama, "Design of highly parallel linear digital circuits based on symbol-level redundancy", IEEE Proc. 26th Int. symposium on MVL, pp.104-109, May, 1996.
- [2] M.Nakajima and M.Kameyama, "Design of highly parallel linear digital system for ULSI processors" IEICE Trans., vol.E76-C, no.7, pp.1119-1125, 1993.
- [3] S.Mitra, N.R.Saxena, and E.J.McCluskey, "Efficient Design diversity Estimation for Combinational Circuits," IEEE Trans. on Computers, pp.1483-1492, Vol.53, No.11, Nov. 2004.
- [4] M.davio, J.Deschamps and A.Thayse, Discrete and Switching Functions, McGraw-Hill Inc. 1978.
- [5] D.Lee, A.A.Gaffar, O.Mencer, and W.Luk, "Optimization hardware Function Evaluation," IEEE Trans., Comput., Vol.54, No.12, pp.1520-1531, Dec., 2005.
- [6] R.J.Wilson and J.J.Watkins, GRAPH An Introductory Approach, John Wiley & Sons, Inc., 1990.
- [7] D.B.west, Introduction to Graph Theory, Prentice-Hall, Inc., 1996.
- [8] E.Arın, Galois Theory, NAPCO Graphic arts, Inc., Wilconsin, 1971.
- [9] R.J.McEliece, Finite Fields for Computer Scientists and Engineers, Kluwer Academic Publisher, 1978.

※ 이 논문은 충주대학교 대학구조개혁지원사업비(교육과학기술부 지원)의 지원을 받아 수행한 연구임

저자 소개

박 춘 명(정회원)

제8권 제2호 참조