

이산제어시스템 설계를 위한 확장된 마크흐름선도의 동작해석

여 정 모[†]

요 약

확장된 마크흐름선도(EMFG: Extended Mark Flow Graph)는 기존의 MFG를 개선한 그래프로서, 회로변환수식에 의해 실제회로로 쉽게 구현가능하므로 이산제어시스템의 모델링과 설계 및 구현의 강력한 도구로 사용될 수 있다. 본 논문은 EMFG의 트랜지션들이 점화하는 과정 및 트랜지션들이 점화완료하였을 때 각 박스들의 마크수 변화를 부울함수식과 벡터를 사용하여 표현하였다. 또한 시스템의 상태변화를 쉽게 판단할 수 있게 하는 EMFG의 동작알고리즘을 제안하였으며, 제안된 알고리즘은 3-비트 증가계수기를 설계한 EMFG와 시간트랜지션이 포함된 가상의 EMFG에서 잘 수행되었다. EMFG의 동작이 부울함수로 해석가능해짐으로 인해 시스템의 분석 및 설계가 용이하며 컴퓨터를 이용한 자동화된 시스템의 분석과 설계가 가능하다.

The Analysis of an Extended Mark Flow Graph's Operation for Design of the Discrete-event Control System

Jeong Mo Yeo[†]

ABSTRACT

The EMFG(Extended Mark Flow Graph) can be easily translated to a physical circuit through a conversion formula and then has been used as the powerful tool for modeling, design and implementation of discrete-event control systems. In this paper, we present an analysis method of an EMFG using boolean functions and vectors that describe the firing process of the transitions and calculate the number of marks in each box. We also present an EMFG running algorithm that a designer can recognize the state variation of the system easily. The proposed algorithm works well on two analysis examples: an EMFG representing a 3-bit up counter and an EMFG including time transitions. Due to the analysis possibility of an EMFG with boolean functions, the operations of the system can be understood and analyzed easily, then the automatic analysis and design of the system is possible with the computer simulation.

1. 서 론

마크흐름선도(Mark Flow Graph)가 Hasegawa에 의하여 제안[5]되었고 공장자동화[4]와 전기스위치 시

스템[6] 및 생산시스템[7, 8]에 적용되었다. Petri Net도 마찬가지로 논리제어기(logic controller)[9, 11]나 릴레이 라다(ladder)[10]에 적용되었으며, 공장자동화[12-16]에 이용되었다. 공장자동화나 논리제어기에 적용된 연구는 수없이 많다.

그리고 순서제어시스템의 모델링과 설계 및 구현에 적합하도록 마크흐름선도를 확장하여 확장된 마크흐름

[†] 정 회 원 : 부경대학교 자연과학대학 전자계산학과
논문접수 : 1998년 1월 17일, 심사완료 : 1998년 6월 22일

선도(EMFG : Extended Mark Flow Graph)가 [1]에서 제안되었다. EMFG는 안전(safe)한 그래프이므로 상태를 나타내는 박스의 마크수가 0이나 1이어서 순서제어시스템의 상태를 표현하기에 적합하고 마크선도에서 정의되지 않은 펄스박스나 게이트박스, 보존아크나 역아크 등이 도입되어 시스템의 제어가 용이할 뿐만 아니라 모델링된 EMFG를 일대일로 변환할 수 있어 구현이 쉽다. EMFG 사용예로 [1]에서는 4층 엘리베이터 제어회로가 설계되어 직접 구현되었다. EMFG는 Petri Net의 성질 대부분을 그대로 가지고 있으므로 EMFG로서 시스템을 모델링하면 Petri Net나 마크선도로서 시스템을 설계하는 것과 마찬가지로 시스템의 동시성이나 병렬성을 잘 표현할 수 있다. 또한 EMFG는 동기 또는 비동기 순서제어회로에도 그 적용[2]이 쉬우며 설계된 EMFG 회로도 간략화[3]될 수 있다.

또한 [17]의 EMFG에서는 박스들의 표현이 하나의 박스로 통일되고 보존아크를 사용하지 않았으며 출력역아크를 새로이 도입함으로써 EMFG가 재구성되었다. 이로 인해 EMFG의 기본동작이 명확하게 되었으며 그 이론이 진보되고 체계화되었다. 또한 [17]에서는 EMFG의 회로변환을 회로변환수식으로 표현하여 행함으로써 더욱 쉽게 EMFG를 회로구현할 수 있었다.

하지만 시스템을 표현한 EMFG의 각 박스들의 상태변화를 수동적으로 관측함으로써 규모가 큰 시스템[1]인 경우 정상동작의 판단에 시간을 요하며 설계에 반영하는데도 어려움이 있었다.

만약 EMFG를 부울함수로 해석하여 시스템의 상태변화를 쉽게 관측할 수 있다면 이산제어시스템의 모델링이나 설계가 쉬워질 뿐만 아니라 설계된 EMFG의 동작 즉 시스템의 동작분석에도 쉽게 적용될 수 있고 실제에 반영하는 것도 용이해져 시스템의 성능향상에 도움을 줄 것이다.

따라서 본 논문에서는 우선 각 트랜지션들의 점화조건식을 부울함수식으로 표현하고 점화조건 벡터로서 트랜지션들의 점화가능유무를 판단하였다. 그리고 EMFG가 동작하는 단위시간을 마킹시간으로 정의하고, 각 트랜지션들의 완료비(점화가 완료되는 시간에 대한 마킹시간의 비)로서 예상완료비, 현완료비, 경과완료비 등을 정의하고 이들의 완료비 벡터들을 도입하여 트랜지션들의 동작을 해석하였으며, 각 박스들의 입출력마크식을 부울함수식으로 표현하여 박스들의 마크수를 계산하였다. 또한 박스들의 상태변화를 쉽게 판단할 수 있도록

하는 EMFG의 동작 알고리즘을 제시하였다.

2. EMFG의 표현

이진값으로 동작하는 이산시스템의 제어회로를 모델링하고 설계하는데 적합한 Petri Net의 한 부류인 확장된 마크흐름선도(EMFG : Extended Mark Flow Graph)[17] G가 다음과 같이 정의된다.

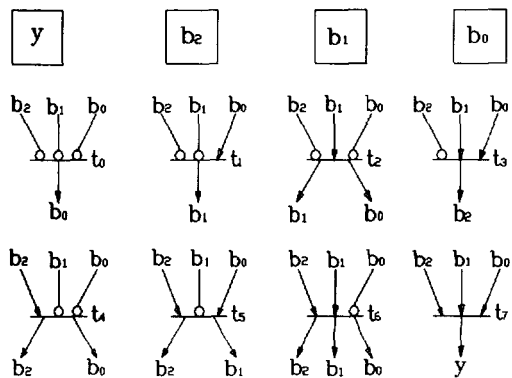
$$G = (B, T, I, O, M)$$

여기서 B는 박스(box)의 집합이고 T는 트랜지션(transition)의 집합이다. I는 트랜지션에 대한 입력함수이고 O는 트랜지션에 대한 출력함수이며 M은 각 박스의 마크상태를 나타내는 마크벡터이다.

예를 들어 그림 1은 동기 3비트 증가계수기를 EMFG로 표현한 것으로 $b_2b_1b_0$ 는 계수기의 출력을 나타내는 박스이며 y는 캐리를 나타내는 박스이다.

그림 1의 EMFG에서 B, T, I, O, M을 구하면 다음과 같다.

$$\begin{aligned} B &= \{y, b_2, b_1, b_0\} \\ T &= \{t_0, t_1, t_2, t_3, t_4, t_5, t_6, t_7\} \\ I(t_0) &= \{b_2', b_1', b_0'\} & O(t_0) &= \{b_0\} \\ I(t_1) &= \{b_2', b_1', b_0\} & O(t_1) &= \{b_1\} \end{aligned}$$



(그림 1) 동기 3-비트 증가계수기의 EMFG
(Fig. 1) An EMFG representing a synchronous 3-bit up counter

$$\begin{aligned}
 I(t_2) &= \{b_2', b_1, b_0'\} & O(t_2) &= \{b_1, b_0\} \\
 I(t_3) &= \{b_2', b_1, b_0\} & O(t_3) &= \{b_2\} \\
 I(t_4) &= \{b_2, b_1', b_0'\} & O(t_4) &= \{b_2, b_0\} \\
 I(t_5) &= \{b_2, b_1', b_0\} & O(t_5) &= \{b_2, b_1\} \\
 I(t_6) &= \{b_2, b_1, b_0'\} & O(t_6) &= \{b_2, b_1, b_0\} \\
 I(t_7) &= \{b_2, b_1, b_0\} & O(t_7) &= \{y\} \\
 \mathbf{M} &= (0, 0, 0, 0)
 \end{aligned}$$

트랜지션에 일반아크(화살표로 표시된 아크)로 입력된 박스에는 마크가 있어야, 역아크(선과 원으로 표시된 아크)로 입력된 박스에는 마크가 없어야 트랜지션의 점화조건을 만족시킬 수 있다. 점화조건이 만족된 트랜지션은 점화개시되어 점화시간이 경과한 후 점화완료된다. 물론 점화도중에 트랜지션의 점화조건이 만족되지 않으면 점화는 중지된다. 트랜지션의 점화가 완료되면 트랜지션에 일반아크로 입력된 박스의 마크는 소멸되고, 역아크로 입력된 박스의 마크는 변화가 없으며, 트랜지션에서 일반아크로 출력된 박스에는 마크를 생성시키고, 역아크로 출력된 박스에는 마크를 소멸시킨다. 만약에 일반/역아크로 출력된 박스에 마크가 있/없으면 마크가 그대로 유지될 것이다.

EMFG에서 트랜지션(들)의 점화조건이 만족되면 점화개시하여 박스(들)의 마크상태를 변화시키고, 다시 다른 트랜지션들의 점화조건이 만족되어 점화개시함으로써 계속적으로 박스(들)의 마크상태가 변화되어 EMFG가 동작한다.

예를 들어 그림 1에서 어떤 박스에도 마크가 없으므로 트랜지션 t_0 만 점화조건이 만족되어 점화될 것이고, t_0 가 점화완료하면 박스 b_0 에 마크가 생성되어 마크벡터는 (0, 0, 0, 1)이 된다. 즉 000에서 001로 계수가 된다. 다시 트랜지션 t_1 만이 점화조건이 만족되어 점화완료되면 b_0 의 마크가 소멸되고 b_1 에 마크가 생성되어 마크벡터는 (0, 0, 1, 0)이 됨으로써 계수가 된다. 같은 방법으로 계수기는 동작될 것이다.

그리고 EMFG를 회로변환식으로 표현하여 그대로 회로로 변환할 수 있다. [17] 예를 들어 그림 1의 EMFG의 회로변환식을 표 1에 나타내었다. 표 1에서 t_i 의 회로변환식은 EMFG에서 트랜지션을 변환한 식이며, G_i 와 D_i 는 박스를 GD 플립플롭 [17]으로 변환하였을 때의 플립플롭 입력이다. 그리고 GD 플립플롭의 진리표를 표 2에 나타내었다. 표 2에서 $Q(t)$ 는 GD 플립플롭의 현재상태 출력이며 $Q(t+1)$ 은 GD 플립플롭의

다음상태 출력이다.

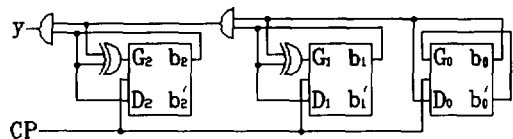
<표 1> 그림 1의 회로변환식들
<Table 1> Circuit translation formulas of Fig. 1

	회로변환식	간소화된 식
t_0	$b_2' b_1' b_0'$	$b_2' b_1' b_0'$
t_1	$b_2' b_1' b_0$	$b_2' b_1' b_0$
t_2	$b_2' b_1 b_0'$	$b_2' b_1 b_0'$
t_3	$b_2' b_1 b_0$	$b_2' b_1 b_0$
t_4	$b_2 b_1' b_0'$	$b_2 b_1' b_0'$
t_5	$b_2 b_1' b_0$	$b_2 b_1' b_0$
t_6	$b_2 b_1 b_0'$	$b_2 b_1 b_0'$
t_7	$b_2 b_1 b_0$	$b_2 b_1 b_0$
G_0	$t_0 + t_2 + t_4 + t_6$	b_0'
D_0	$t_1 + t_3 + t_5 + t_7$	b_0
G_1	$t_1 + t_2 + t_5 + t_6$	$b_1 \bullet b_0$
D_1	$t_2 + t_3 + t_6 + t_7$	b_1
G_2	$t_3 + t_4 + t_5 + t_6$	$b_2 \bullet (b_1 b_0)$
D_2	$t_4 + t_5 + t_6 + t_7$	b_2
y	t_7	$b_2 b_1 b_0$

<표 2> GD 플립플롭의 진리표
<Table 2> A truth table for a GD Flip Flop

G	D	$Q(t+1)$	$Q'(t+1)$
0	0	$Q(t)$	$Q'(t)$
0	1	0	1
1	0	1	0
1	1	1	0

표 1의 회로변환식을 회로로 그리면 그림 2와 같이 된다. 그림 2에서 CP는 클럭펄스 입력단자이다.



(그림 2) 그림 1의 변환회로
(Fig. 2) A circuit representing Fig. 1

3. EMFG의 동작

EMFG의 동작을 부울함수로 해석할 수 있다면 설계된 EMFG를 쉽게 분석할 수 있을 뿐만 아니라 컴퓨터 시뮬레이션을 이용한다면 복잡하고 큰 EMFG도 착오 없이 쉽게 분석하여 문제점을 발견하고 개선할 수 있을 것이다. 즉 시스템의 설계와 구현이 용이할 것이다.

따라서 트랜지션의 동작 및 박스의 마크수가 결정되는 과정을 부울함수를 사용하여 해석한다. 여기서 사용된 예는 그림 1의 EMFG이다.

3.1 트랜지션의 점화조건식과 벡터

트랜지션의 점화조건은 입력박스의 마크상태에 의존하고 이 마크상태들은 이진상태로 표현할 수 있으므로 점화조건을 부울함수로 표현할 수 있다.

정의 1) 어떤 트랜지션 t 의 점화조건(the fire condition)을 부울함수식으로 표현하였을 때, 이 부울함수식을 t 에 대한 점화조건식(the fire condition equation)이라 정의한다.

정리 1) EMFG의 어떤 트랜지션 t_i 에 대한 점화조건식 C_i 를 AND 함수로 표현하면 다음과 같다.

$$C_i = \text{AND}(b_j, b_k)$$

여기서 b_j/b_k 는 일반/역아크로 연결된 t_i 의 입력박스의 마크존재여부이고, j 와 k 는 양의 정수이다.

증명) 트랜지션 t_i 에 연결된 모든 입력박스의 마크상태가 동시에 만족되어야 t_i 의 점화조건이 만족된다. 즉 이는 t_i 에 연결된 각 입력박스의 마크상태가 AND되어 만족되어야 한다는 뜻이므로 정리는 성립한다.

EMFG에서 점화조건식은 트랜지션의 개수만큼 존재하고 트랜지션의 점화시간과는 무관하다. 점화조건식의 값이 0이면 점화조건이 만족되지 못하며, 1이면 점화조건이 만족된 상태이다.

정의 2) EMFG에서 점화조건 벡터 F 를 다음과 같이 정의한다.

$$F = (f_1, f_2, \dots, f_i, \dots, f_n)$$

여기서 f_i 는 트랜지션 t_i 의 점화조건식 C_i 의 값이며 n 은 트랜지션의 수이다.

EMFG의 점화조건 벡터에서 각 원소들은 해당 트랜지션의 점화조건 만족여부를 나타낸다. 예를 들어 그림 1의 EMFG에서 각 트랜지션의 점화조건식과 점화조건 벡터는 다음과 같다. 따라서 트랜지션 t_0 만이 점화조건이 만족되어 있으며 나머지 트랜지션들은 점화조건이 만족되어 있지 않다.

$$\begin{aligned} C_0 &= b_2' \cdot b_1' \cdot b_0' & C_1 &= b_2' \cdot b_1' \cdot b_0 \\ C_2 &= b_2' \cdot b_1 \cdot b_0' & C_3 &= b_2' \cdot b_1 \cdot b_0 \\ C_4 &= b_2 \cdot b_1' \cdot b_0' & C_5 &= b_2 \cdot b_1' \cdot b_0 \\ C_6 &= b_2 \cdot b_1 \cdot b_0' & C_7 &= b_2 \cdot b_1 \cdot b_0 \\ F &= (1, 0, 0, 0, 0, 0, 0, 0) \end{aligned}$$

3.2 마킹평가와 마킹시간

EMFG에서 점화조건이 만족된 트랜지션(들)은 점화를 개시하여 점화가 완료되고 각 박스의 마크상태는 변화한다. 이와같은 과정이 반복되어 EMFG의 각 박스의 마크상태는 계속적으로 변화하게 된다. 이는 EMFG로 모델링된 시스템이 동작되는 과정을 보여주는 것과 동일하다.

정의 3) EMFG에서 어떤 박스에 연결된 트랜지션들의 점화완료를 검사하여 마크수를 결정해 주는 것을 박스의 마킹평가(the marking evaluation)라 정의하고, EMFG내의 모든 박스들에 대하여 마킹평가하는 과정을 EMFG의 마킹평가라 정의한다.

EMFG에 초기마크가 부여되면 EMFG는 동작을 개시한다. 하지만 EMFG에는 일반트랜지션과 시간트랜지션이 혼재해 있으므로 각 트랜지션들이 동시에 점화조건이 만족된다 하더라도 점화가 완료되는 시간은 트랜지션마다 다를 것이다. 이 경우 점화시간이 가장 적은 트랜지션이 가장 먼저 점화완료하게 될 것이고 점화시간에 따라 순차적으로 트랜지션은 점화하게 될 것이다. 때로는 점화완료전에 만족되었던 점화조건이 만족되지 못함으로 인해 어떤 트랜지션은 점화를 중지해 버리는 경우도 있을 것이다. 따라서 EMFG에서 트랜지션의 점화동작을 어떤 규칙에 의해 동작시킬 필요가 있다.

정의 4) EMFG를 마킹평가하는 기준 단위시간을 EMFG의 마킹시간(the marking time)이라 정의한다.

정리 2) EMFG의 마킹시간 t_m 은 다음과 같다.

$$t_m = G.C.M.(s_1, s_2, \dots, s_i, \dots, s_n)$$

여기서 s_i 는 트랜지션 t_i 의 점화시간이며 n 은 트랜지션의 수이다.

증명) EMFG의 마킹평가는 마킹시간마다 일어나고, EMFG에 있는 모든 트랜지션들이 점화완료되는 각 시점에서 EMFG의 마킹평가가 일어나야 한다. 정리가 성립한다고 가정하면 마킹시간 t_m 은 각 트랜지션들의 점화시간의 최대공약수이므로 임의의 트랜지션 t_i 의 점화시간 s_i 는 항상 t_m 의 배수가 되어 t_i 는 t_m 의 배수가 되는 시점에서 t_i 에 연결된 박스들이 마킹평가된다. 즉 각 트랜지션들은 t_m 의 배수가 되는 시점에서 박스들이 마킹평가되므로 가정은 타당하여 정리는 성립한다.

일반트랜지션의 점화시간 t_g 가 마킹시간 t_m 과 다를 때는 t_g 와 t_m 을 동시에 따로 명기하여야 한다. 만약 점화시간을 고려하지 않는 일반트랜지션만 존재하는 EMFG인 경우에는 t_g 및 t_m 을 생략할 수 있다. 이때는 마킹시간을 단위 1로 하여 EMFG를 마킹평가한다. 시간트랜지션이 없는 시스템 즉 유한상태기계 등 타이머를 사용하지 않는 시스템에서는 마킹시간을 1로 하여 평가하면 될 것이다. 예를 들어 그림 1에서는 마킹시간 t_m 을 따로 명기되지 않았으므로 마킹시간이 1이 된다.

3.3 트랜지션의 완료비와 벡터

EMFG에는 일반트랜지션과 시간트랜지션이 동시에 점화조건이 만족되어도 각각 점화완료되는 시점이 다르다. 따라서 시간에 의존하여 트랜지션의 점화를 분석할 필요가 있다.

정의 5) EMFG에서 어떤 트랜지션 t_i 가 점화개시되어 점화중에 있을 때 점화완료까지 남아있는 시간을 t_i 의 점화완료시간(the fire complete time)이라 정의하고, t_i 의 점화완료시간에 대한 마킹시간의 비를 t_i 의 완료비(the complete ratio)라 정의한다.

특히 t_i 의 점화시간에서 마킹시간을 나눈 값을 t_i 의

초기완료비(the initial complete ratio)라 한다.

각 트랜지션들의 점화완료시간으로서 각 트랜지션들이 어느 정도 점화가 진행되었는지 알 수 있고, 각 트랜지션들의 완료비로서 각 트랜지션들의 점화완료정도를 판단할 수 있다. 이는 시스템의 동작진행정도를 판단하는데 그대로 이용될 수 있을 것이다.

정의 6) 트랜지션의 점화조건이 만족되었다고 가정하였을 때의 완료비를 예상완료비(the expected complete ratio)라 하고, 점화조건이 적용된 후의 완료비를 현완료비(the present complete ratio)라 하며, 현완료비보다 1 적은 완료비를 경과완료비(the lapsed complete ratio)라 한다.

예상완료비는 양의 정수로서 트랜지션의 점화조건이 만족되는 경우 점화완료까지의 시간정도를 나타낸다. 즉 시스템에서 상태변화조건이 만족되는 경우 상태변화가 일어나서 상태변화가 완료하기까지 필요한 시간의 정도를 의미한다.

현완료비는 항상 0 이상의 양수로서, 0이면 해당하는 트랜지션의 점화조건이 만족되지 않아 점화를 개시하지 않았음을 나타내며, 양의 정수이면 해당하는 트랜지션의 점화조건이 만족되어 점화를 처음 개시하였거나 계속 점화중으로 점화완료되기까지의 시간정도를 나타낸다. 즉 시스템에서 상태변화조건이 만족된 상태(상태변화조건이 처음 만족되어 상태변화를 시작했거나 이전에 상태변화조건이 만족되어 상태변화를 일으키고 있는 중)에서 상태변화를 완료할 때까지 소요되는 시간의 정도를 뜻한다.

경과완료비는 현완료비를 구한 시점에서 마킹시간이 경과한 시점에서의 현완료비이며, 점화완료유무를 뜻한다. 경과완료비가 -1이면 점화조건이 만족되지 않아 점화되지 않았음을 나타내고, 0이면 점화완료되었음을 나타내며, 양의 정수이면 계속 점화중으로 점화완료되기까지의 시간정도를 나타낸다. 경과완료비가 구해진 시점에서 마킹평가가 행해져야 한다. 마킹평가가 행해지면 박스들의 마크수는 변할 것이므로 점화조건들은 다시 검사되어야 한다. 즉 시스템에서 어떤 상태들이 상태변화를 완료하면 이 상태들은 다른 상태들에 다시 영향을 줄 것이므로 다시 상태변화조건들이 검사되어져 만족된 상태들만이 변화를 시작할 것이다.

마킹시간과 동일한 점화시간을 가지는 트랜지션의

경우, 점화조건이 만족되면 예상완료비가 현완료비로 되고, 마킹시간만큼 시간이 경과하면 현완료비가 경과 완료비로 되어 점화완료됨으로써 다시 원래의 예상완료비로 변한다. 하지만 마킹시간보다 큰 점화시간을 가지는 트랜지션의 경우에는 점화조건이 만족되면 예상완료비가 현완료비로 되고, 마킹시간만큼 시간이 경과하면 현완료비가 경과완료비로 되지만 점화가 완료되지 않았으므로 경과완료비는 예상완료비로 되어 다시 점화조건이 검사된다. 만족되지 않으면 점화중지할 것이고 계속 점화조건이 만족되면 예상완료비가 현완료비로 되어 시간이 경과할 것이다. 이와같은 과정을 점화완료될 때까지 반복한다. 즉 예상완료비와 현완료비는 점화가 진행됨에 따라 점점 감소하여 0에 도달함으로써 점화완료될 것이다.

정의 7) 예상완료비 벡터(the expected complete ratio vector) V , 현완료비 벡터(the present complete ratio vector) W 및 경과완료비 벡터(the lapsed complete ratio vector) X 를 다음과 같이 정의한다.

$$\begin{aligned} V &= (v_1, v_2, \dots, v_i, \dots, v_n) \\ W &= (w_1, w_2, \dots, w_i, \dots, w_n) \\ X &= (x_1, x_2, \dots, x_i, \dots, x_n) \end{aligned}$$

여기서 v_i 는 t_i 의 예상완료비, w_i 는 t_i 의 현완료비, x_i 는 t_i 의 경과완료비, n 은 트랜지션의 수이다.

정리 3) 트랜지션 t_i 의 예상완료비를 v_i 라 하고 점화조건식을 f_i 라 하면, 트랜지션 t_i 의 현완료비 w_i 는 다음 식으로 표현된다.

$$w_i = v_i \cdot f_i$$

증명) 예상완료비에서 점화조건식이 적용되면 현완료비가 되므로 정리는 자명하다.

보조정리 1) EMFG에서 예상완료비 벡터를 V 라 하고 점화조건 벡터를 F 라 하면, 현완료비 벡터 W 는 다음 식으로 표현된다.

$$W = V \star F$$

여기서 \star 는 두 벡터에서 동일한 순서에 해당하는 원소

끼리 곱하는 연산자로 정의된다.

증명) 정리 3에서 자명하다.

보조정리 2) EMFG에서 현완료비 벡터를 W 라 했을 때, 경과완료비 벡터 X 는 다음 식으로 표현된다.

$$X = (W - 1)$$

증명) 정의 6에서 자명하다.

예를 들어 그림 1에서 초기완료비 벡터는 (1, 1, 1, 1, 1, 1, 1)이므로 초기에 예상완료비 벡터 V_0 는 초기완료비 벡터가 된다. 그리고 그림 1에서는 현재 t_0 만이 점화조건이 만족되어 있으므로 점화조건 벡터 F_0 가 (1, 0, 0, 0, 0, 0, 0)이 된다. 따라서 현완료비 벡터 W_0 는 ($V_0 \star F_0$)가 되어 (1, 0, 0, 0, 0, 0, 0)이 된다. 현재상태에서 1의 마킹시간이 경과하면 W_0 는 경과완료비 벡터 X_1 이 된다. 즉 X_1 은 (0, -1, -1, -1, -1, -1, -1)이 될 것이다. 이는 트랜지션 t_0 가 점화개시하여 점화시간이 경과되어 점화완료시점에 도달했다는 뜻이 된다.

3.4 박스의 마킹평가

트랜지션의 점화조건이 만족되어 점화개시되고 점화시간이 경과되어 점화완료되면 트랜지션에 연결된 각 박스의 마크수는 0이나 1이 된다. 따라서 각 박스의 마크상태도 이진값으로 표현할 수 있으므로 점화완료후 박스의 마크수는 점화완료전 박스의 마크수와 점화조건 식으로 표현할 수 있다.

정의 8) EMFG에서 박스 b_x 에 연결된 트랜지션(들)이 점화한 후 박스 b_x 의 마크수를 표현한 식을 박스 b_x 의 마크식(a mark equation)이라 하고, 박스 b_x 에 입/출력으로 연결된 트랜지션 t_i 가 점화하였을 때의 점화식을 트랜지션 t_i 에 대한 박스 b_x 의 입/출력마크식(an input/output mark equation)이라 하고 $b_x(i)/B_x(i)$ 로 표기한다.

정리 4) 트랜지션 t_i 가 박스 b_x 에 연결되어 있을 때, t_i 의 점화조건식이 C_i 이고 t_i 의 점화완료전 b_x 의 마크수를 b_x 라 하면, 연결된 아크의 종류에 따라 박스 b_x 의

입출력마크식은 다음과 같다.

1) 일반아크로 연결된 경우

$$b_x(i) = C_i + b_x \quad (1)$$

$$B_x(i) = C_i' \cdot b_x \quad (2)$$

2) 역아크로 연결된 경우

$$b_x(i) = C_i' \cdot b_x \quad (3)$$

$$B_x(i) = b_x \quad (4)$$

증명) (1)식의 증명 : t_i 가 점화완료됨으로 인해 박스 b_x 의 마크수에 영향을 주는 요소는 C_i 와 b_x 뿐이다. 이 요소의 조합은 4 가지 경우가 있을 수 있으며, 각 경우에 대한 $b_x(i)$ 를 표 3에 나타내었다. 표 3에서 $b_x(i)$ 를 구하면 $C_i + b_x$ 이 된다.

<표 3> C_i 와 b_x 에 따른 $b_x(i)$ 상태
 <Table 3> The $b_x(i)$ state according to C_i and b_x

C_i	b_x	$b_x(i)$
0	0	0
0	1	1
1	0	1
1	1	1

(2)식 및 (3)식의 증명 : (1)식의 증명과 마찬가지로 방법으로 증명된다.

(4)식의 증명 : 박스 b_x 의 마크수 b_x 는 t_i 의 점화조건에만 영향을 주고 점화완료여부에는 어떠한 영향도 받지 않는다. 따라서 점화완료될 때 마크변화는 일어나지 않으므로 마크수 b_x 는 그대로이다.

그림 1에서 트랜지션 t_i 의 점화조건식을 C_i 라 하고 박스 b_x 의 마크수를 b_x 라 했을 때, 각 박스들의 입/출력마크식을 구하면 다음과 같다.

$$\begin{aligned}
 y(7) &= C_7 + y \\
 b_0(0) &= C_0 + b_0 & b_0(2) &= C_2 + b_0 \\
 b_0(4) &= C_4 + b_0 & b_0(6) &= C_6 + b_0 \\
 B_0(0) &= b_0 & B_0(1) &= C_1' \cdot b_0 \\
 B_0(2) &= b_0 & B_0(3) &= C_3' \cdot b_0 \\
 B_0(4) &= b_0 & B_0(5) &= C_5' \cdot b_0
 \end{aligned}$$

$$\begin{aligned}
 B_0(6) &= b_0 & B_0(7) &= C_7' \cdot b_0 \\
 b_1(1) &= C_1 + b_1 & b_1(2) &= C_2 + b_1 \\
 b_1(5) &= C_5 + b_1 & b_1(6) &= C_6 + b_1 \\
 B_1(0) &= b_1 & B_1(1) &= b_1 \\
 B_1(2) &= C_2' \cdot b_1 & B_1(3) &= C_3' \cdot b_1 \\
 B_1(4) &= b_1 & B_1(5) &= b_1 \\
 B_1(6) &= C_6' \cdot b_1 & B_1(7) &= C_7' \cdot b_1 \\
 b_2(3) &= C_3 + b_2 & b_2(4) &= C_4 + b_2 \\
 b_2(5) &= C_5 + b_2 & b_2(6) &= C_6 + b_2 \\
 B_2(0) &= b_2 & B_2(1) &= b_2 \\
 B_2(2) &= b_2 & B_2(3) &= b_2 \\
 B_2(4) &= C_4' \cdot b_2 & B_2(5) &= C_5' \cdot b_2 \\
 B_2(6) &= C_6' \cdot b_2 & B_2(7) &= C_7' \cdot b_2
 \end{aligned}$$

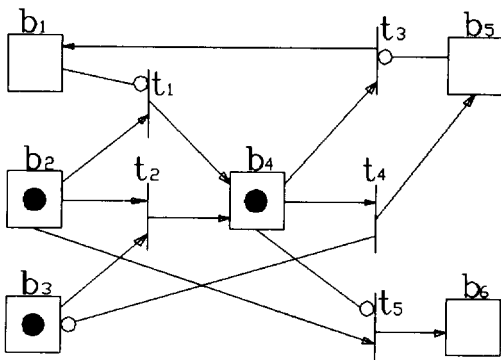
하나의 박스에 입력되는 트랜지션은 다수개 존재할 수 있고 출력되는 트랜지션도 다수개 존재할 수 있으며, 이 트랜지션들도 서로 점화시간이 같은 것이 있고 점화시간이 다른 것도 있을 수 있다. 점화시간이 같은 트랜지션들이라도 점화중에 점화조건이 만족되지 않아 점화가 중지될 수도 있다. 따라서 각 트랜지션들의 점화완료시점이 서로 다르므로 하나의 박스에 대한 입력 점화식들과 출력점화식은 따로따로 구성되어야 한다. 단지 점화시간이 마킹시간과 같은 트랜지션들의 점화식들은 마킹시간마다 평가되므로 하나의 점화식으로 표현될 수 있다.

정리 5) 어떤 박스 b 에 입력으로 연결된 트랜지션들중 경과완료비가 0인 트랜지션들이 $t_{i1}, t_{i2}, \dots, t_{in}, \dots, t_{im}$ 이고 출력으로 연결된 트랜지션들중 경과완료비가 0인 트랜지션들이 $t_{o1}, t_{o2}, \dots, t_{oj}, \dots, t_{om}$ 일 때, 박스 b 를 마킹평가하는 마크식은 다음식으로 표현된다.

$$\begin{aligned}
 b &= \text{OR}(b(i1), b(i2), \dots, b(ii), \dots, b(in)) \\
 &+ \text{AND}(B(o1), B(o2), \dots, B(oj), \dots, B(om))
 \end{aligned}$$

증명) 경과완료비가 0인 트랜지션은 점화완료된 트랜지션이다. 그리고 박스 b 의 입력트랜지션들중 어떤 한 트랜지션이라도 점화완료되어 박스에 마크를 생성시킨다면, 박스 b 에는 마크가 존재하게 된다. 즉 입력트랜지션들이 점화완료되는 경우의 입력마크식들은 논리 OR되어 영향을 주게 된다. 그리고 박스 b 의 출력트랜지션들중 어느 한 트랜지션이라도 점화완료되어 박스의

마크를 소멸시킨다면, 출력트랜지션의 점화완료에 의한 박스의 마크는 존재하지 않을 것이다. 즉 출력트랜지션들이 점화완료되는 경우의 출력마크식은 논리 AND 되어 영향을 주게 된다. 또한 박스 b의 마크수는 점화완료된 입력트랜지션들에 의해 마크가 생성하여도 박스 b에는 마크가 존재하게 되고, 점화완료된 출력트랜지션들에 의해 마크가 존재하게 되어도 역시 박스 b에는 마크가 존재하게 된다. 따라서 박스 b의 마크수가 점화완료된 입력트랜지션의 영향과 점화완료된 출력트랜지션의 영향이 논리 OR되어 결정되므로 정리는 타당하다.



(그림 3) 점화가능한 다수의 트랜지션들이 있는 EMFG의 예
 (Fig. 3) An EMFG example with multiple transitions being able to fired

예를 들어 그림 3의 EMFG에서 박스 b4를 마킹평가해보자. 현재 박스 b4에 연결된 트랜지션들중 점화가능한 트랜지션들은 t1, t2, t3, t4 들이고, 점화조건이 만족된 후 마킹시간이 경과되면 이 네 트랜지션들의 경과완료비가 0이 될 것이다. 따라서 박스 b4의 마크수를 결정하는 마크식은 정리 5에 의하여 다음과 같이 되고 계산된다.

$$\begin{aligned}
 b &= \text{OR}(b_4(1), b_4(2)) + \text{AND}(B_4(3), B_4(4)) \\
 &= \text{OR}(C_1 + b_4, C_2 + b_4) + \text{AND}(C_3' \cdot b_4, C_4' \cdot b_4) \\
 &= \text{OR}(b_1' \cdot b_2 + b_4, b_2 \cdot b_3 + b_4) \\
 &\quad + \text{AND}((b_4 \cdot b_5)' \cdot b_4, (b_4)' \cdot b_4) \\
 &= \text{OR}(0' \cdot 1 + 1, 1 \cdot 1 + 1) + \text{AND}((1 \cdot 0)' \cdot 1, 1' \cdot 1) \\
 &= \text{OR}(1, 1) + \text{AND}(0, 0) = 1 + 0 = 1
 \end{aligned}$$

따라서 박스 b4의 마크는 트랜지션 t1과 t2의 점화완

료에 의하여 두 번 생성되어 하나가 존재하고 트랜지션 t3과 t4의 점화완료에 의하여 두 번 소멸되어 하나가 소멸되지만 이들이 논리 OR로 작용하여 마크는 존재하게 된다.

보조정리 3) EMFG에서 경과완료비 벡터 X에서 0 이하의 원소들만 초기완료비로 대체시키면 예상완료비 벡터 V가 된다.

증명) 경과완료비가 양인 트랜지션은 계속 점화증명을 나타내므로 다시 점화조건이 적용되어 점화조건이 만족되더라도 현 경과완료비에서 점화를 계속하여야 하므로 초기완료비로 대체시켜서는 안된다. 하지만 경과완료비가 0이거나 -1인 트랜지션은 점화가 완료되었거나 점화되지 않았던 것이므로 점화조건이 적용되어 점화조건이 만족될 때는 처음부터 점화가 개시되어야 하므로 초기완료비로 대체되어야 한다. 따라서 정리는 타당하다.

예를 들어 그림 1에서 경과완료비 벡터 X1이 (0, -1, -1, -1, -1, -1, -1, -1)인 경우 마킹평가가 행해진 후 예상완료비 벡터 V1은 (1, 1, 1, 1, 1, 1, 1, 1)으로 된다. 즉 모든 트랜지션들의 점화조건이 만족되는 경우 처음부터 다시 점화가 시작된다.

4. EMFG의 동작 알고리즘

EMFG가 동작한다는 것은 이에 대응되는 시스템이 동작한다는 뜻이다. 즉 시간경과에 따라 각 박스들의 마크상태가 변화하는 것을 의미하고 시스템의 상태가 변화함을 의미한다. 이러한 마크의 상태변화를 기존에는 점화가능한 트랜지션들을 검사하고 점화완료되는 트랜지션들을 검사하여 마크상태를 변화시켰으므로 마크 상태변화를 체계적으로 판단할 수 있는 방법이 없었다. 이로 인해 엘리베이터와 같은 규모가 큰 시스템[1]인 경우 설계된 EMFG가 제대로 동작하는지 검사하는데 많은 시간을 요했으며 이를 설계에 반영하는데도 어려움이 있었다.

따라서 본 연구에서는 마크 상태변화를 체계적으로 판단할 수 있는 방법을 알고리즘으로 제시한다. 이는 시스템의 상태변화 즉 EMFG의 마크 상태변화를 쉽게 판단할 수 있을 것이므로 해석이 편리하고 설계에 반영하기가 쉬워 전체적인 설계시간의 단축뿐만 아니라 성

능개선에 많은 도움을 줄 것이다.

정리 6) 다음은 EMFG에 초기마크가 부여된 후 EMFG가 동작하는 알고리즘이다.

(1) 초기화 과정 : 정리 2를 사용하여 마킹시간 t_m 을 구하고, 각 트랜지션들의 초기완료비들을 구하여 예상 완료비 벡터 V_0 를 형성한다. 그리고 정리 1과 정리 4를 사용하여 각 트랜지션들의 점화조건식과 각 박스들의 마크식들을 구한다. 또한 초기마크벡터 M_0 을 형성한다.

(2) 점화조건 벡터 구함 : 점화조건식에서 점화조건을 구하고 점화조건 벡터 F_1 를 구한다. 이때 점화조건 벡터가 0이면 EMFG의 동작을 끝낸다.

(3) 현완료비 벡터 구함 : 보조정리 1을 사용하여 현 완료비 벡터 W_1 를 구한다.

(4) 경과완료비 벡터 구함 : 시간이 마킹시간만큼 경과하면 보조정리 2를 사용하여 경과완료비 벡터 X_1 를 구한다.

(5) 박스들의 마킹평가 : 경과완료비 벡터의 원소들중 0에 해당하는 트랜지션들에 연결된 박스들을 정리 5를 사용하여 마킹평가한다. 이에 해당되지 않은 박스들은 마킹평가하지 않는다.

(6) 마크벡터 구함 : (5)에서 마킹평가된 내용으로 마크벡터 M_1 를 구한다.

(7) 예상완료비 벡터 구함 : 보조정리 3을 사용하여 예상완료비 벡터 V_1 를 구한다.

(8) (2)에서 (7)의 과정을 반복한다.

증명) 정리 1부터 정리 5까지와 보조정리 1부터 보조정리 3까지에 의해 자명하다.

예를 들어 정리 6의 알고리즘에 따라 그림 1의 EMFG 동작결과를 표 4에 나타내었다. 표 4에서 보면 어떤 트랜지션들이 점화조건이 만족되고 점화완료되는지 쉽게 판단할 수 있으며, 박스들의 마크수 변화도 한눈에 확인할 수 있다. 또한 표 4의 결과로써 3-비트 증가 계수기를 설계한 그림 1의 EMFG가 올바르게 동작

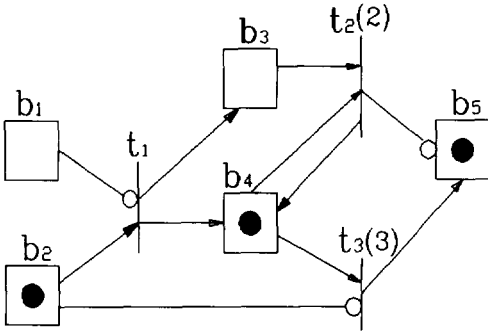
〈표 4〉 그림 1의 EMFG에 정리 6의 알고리즘을 적용시킨 결과표
 〈Table 4〉 A table applying an EMFG of Fig. 1 with an algorithm in theorem 6

시간 (i)	X_i	마킹평가에 사용된 마크식들	M_i	V_i	F_i	W_i
0	-	-	(0,0,0,0)	(1,1,1,1,1,1,1,1)	(1,0,0,0,0,0,0,0)	(1,0,0,0,0,0,0,0)
1	(0,-1,-1,-1,-1,-1,-1,-1)	$b_0 = b_0(0) + B_0(0)$, $b_1 = B_1(0)$, $b_2 = B_2(0)$	(0,0,0,1)	(1,1,1,1,1,1,1,1)	(0,1,0,0,0,0,0,0)	(0,1,0,0,0,0,0,0)
2	(-1,0,-1,-1,-1,-1,-1,-1)	$b_0 = B_0(1)$, $b_1 = b_1(1) + B_1(1)$, $b_2 = B_2(0)$	(0,0,1,0)	(1,1,1,1,1,1,1,1)	(0,0,1,0,0,0,0,0)	(0,0,1,0,0,0,0,0)
3	(-1,-1,0,-1,-1,-1,-1,-1)	$b_0 = b_0(2) + B_0(2)$, $b_1 = b_1(2) + B_1(2)$, $b_2 = B_2(2)$	(0,0,1,1)	(1,1,1,1,1,1,1,1)	(0,0,0,1,0,0,0,0)	(0,0,0,1,0,0,0,0)
4	(-1,-1,-1,0,-1,-1,-1,-1)	$b_0 = B_0(3)$, $b_1 = B_1(3)$, $b_2 = b_2(3) + B_2(3)$	(0,1,0,0)	(1,1,1,1,1,1,1,1)	(0,0,0,0,1,0,0,0)	(0,0,0,0,1,0,0,0)
5	(-1,-1,-1,-1,0,-1,-1,-1)	$b_0 = b_0(4) + B_0(4)$, $b_1 = B_1(4)$, $b_2 = b_2(4) + B_2(4)$	(0,1,0,1)	(1,1,1,1,1,1,1,1)	(0,0,0,0,0,1,0,0)	(0,0,0,0,0,1,0,0)
6	(-1,-1,-1,-1,-1,0,-1,-1)	$b_0 = B_0(5)$, $b_1 = b_1(5) + B_1(5)$, $b_2 = b_2(5) + B_2(5)$	(0,1,1,0)	(1,1,1,1,1,1,1,1)	(0,0,0,0,0,0,1,0)	(0,0,0,0,0,0,1,0)
7	(-1,-1,-1,-1,-1,-1,0,-1)	$b_0 = b_0(6) + B_0(6)$, $b_1 = b_1(6) + B_1(6)$, $b_2 = b_2(6) + B_2(6)$	(0,1,1,1)	(1,1,1,1,1,1,1,1)	(0,0,0,0,0,0,0,1)	(0,0,0,0,0,0,0,1)
8	(-1,-1,-1,-1,-1,-1,-1,0)	$b_0 = B_0(7)$, $b_1 = B_1(7)$, $b_2 = B_2(7)$	(0,0,0,0)	(1,1,1,1,1,1,1,1)	(1,0,0,0,0,0,0,0)	(1,0,0,0,0,0,0,0)

되고 있는 것을 확인할 수 있다.

정리 6의 알고리즘은 시간트랜지션들이 포함된 EMFG에도 그대로 적용될 수 있다. 따라서 시간트랜지션이 포함된 가상의 EMFG를 그림 4에 나타내어 이를 정리 6의 알고리즘에 따라 해석하기로 한다.

우선 그림 4에서 마킹시간 t_m , 초기완료비 벡터 즉 예상완료비벡터 V_0 , 각 트랜지션들의 점화조건식과 각 박스들의 마크식 및 마크벡터 M_0 을 구한다.



(그림 4) 시간트랜지션이 포함된 가상의 EMFG
(Fig. 4) An assumed EMFG including time transitions

$$\begin{aligned}
 t_m &= \text{G.C.M.}(1, 2, 3) = 1 \\
 V_0 &= (1/1, 2/1, 3/1) = (1, 2, 3) \\
 C_1 &= b_1 \cdot b_2 \quad C_2 = b_3 \cdot b_4 \quad C_3 = b_2 \cdot b_4 \\
 B_1(1) &= b_1 \\
 B_2(1) &= C_1 \cdot b_2 \quad B_2(3) = b_2 \\
 b_3(1) &= C_1 + b_3 \quad B_3(2) = C_2 \cdot b_3 \\
 b_4(1) &= C_1 + b_4 \quad b_4(2) = C_2 + b_4 \\
 B_4(2) &= C_2 \cdot b_4 \quad B_4(3) = C_3 \cdot b_4
 \end{aligned}$$

<표 5> 그림 3의 EMFG에 정리 6의 알고리즘을 적용시킨 결과표
<Table 5> A table applying an EMFG of Fig. 3 with an algorithm in theorem 6

시간(i)	X_i	마킹평가에 사용된 마크식들	M_i	V_i	F_i	W_i
0	-	-	(0, 1, 0, 1, 1)	(1, 2, 3)	(1, 0, 0)	(1, 0, 0)
1	(0, -1, -1)	$b_1 = B_1(1), b_2 = B_2(1),$ $b_3 = b_3(1), b_4 = b_4(1)$	(0, 0, 1, 1, 1)	(1, 2, 3)	(0, 1, 1)	(0, 2, 3)
2	(-1, 1, 2)	-	(0, 0, 1, 1, 1)	(1, 1, 2)	(0, 1, 1)	(0, 1, 2)
3	(-1, 0, 1)	$b_3 = B_3(2),$ $b_4 = b_4(2) + B_4(2),$ $b_5 = b_5(2)$	(0, 0, 0, 1, 0)	(1, 2, 1)	(0, 0, 1)	(0, 0, 1)
4	(-1, -1, 0)	$b_2 = B_2(3), b_4 = B_4(3),$ $b_5 = b_5(3)$	(0, 0, 0, 0, 1)	(1, 2, 3)	(0, 0, 0)	-

$$\begin{aligned}
 b_5(2) &= C_2' \cdot b_5 \quad b_5(3) = C_3' \cdot b_5 \\
 M_0 &= (0, 1, 0, 1, 1)
 \end{aligned}$$

정리 6의 알고리즘 (2)에서 (8)까지를 그림 3의 EMFG에 수행시킨 결과를 표 5에 나타내었다. 표 5에서 보면 역시 어떤 트랜지션들이 점화되어 점화완료되는지 시간에 따라 판단할 수 있으며, 각 박스들의 마크수가 변화해가는 모습을 쉽게 파악할 수 있으므로 EMFG의 동작을 쉽게 해석할 수 있다.

따라서 시스템을 모델링한 EMFG를 제안된 알고리즘에 따라 해석하게 되면 그 동작을 쉽게 분석할 수 있어 시스템의 동작해석 및 설계에 상당히 도움이 될 것이다.

5. 결 론

EMFG는 Petri Net의 성질 대부분을 그대로 가지고 있으므로 시스템의 동시성이나 병렬성을 잘 표현할 수 있으며 동기 및 비동기 시스템에도 그 적용이 쉽다. 뿐만 아니라 시스템을 EMFG를 사용하여 설계하면 설계회로로 쉽게 변환될 수 있으므로 이산제어시스템 및 공장자동화의 제어회로 등을 모델링하고 설계할 때 강력한 도구로 사용될 수 있을 것이다.

본 연구에서 얻은 결론은 다음과 같다.

(1) EMFG에서 각 트랜지션의 점화조건을 부울함수식으로 표현하여 점화조건 만족여부를 쉽게 판단하게 하였고, 점화개시되어 점화완료되기까지의 과정을 각종 완료비를 도입하여 쉽게 해석하였다.

(2) EMFG에서 각 박스들의 마크수를 계산하는 식을 부울함수식으로 표현함으로써 여러 트랜지션들이 동

시에 점화완료되었을 때 각 박스들의 마크수가 변화해 가는 과정 즉 EMFG가 동작되는 과정을 쉽게 해석할 수 있게 하였다.

(3) 기존에 수동으로 행하던 EMFG의 각 박스들의 상태변화를 쉽게 관측할 수 있도록 EMFG의 동작알고리즘을 제안하였고, 3-비트 증가 계수기를 설계한 EMFG 및 시간 트랜지션이 포함된 가상의 EMFG에 알고리즘을 적용하여 알고리즘이 올바르게 동작되는 것을 확인하였다.

따라서 시스템을 EMFG로 설계하였을 때, 설계된 EMFG를 제안된 알고리즘에 따라 해석하게 되면 시스템의 동작을 쉽게 분석할 수 있으므로 설계에 재반영하거나 시스템의 성능향상에 크게 도움이 될 것이다. 또한 컴퓨터 시뮬레이션으로 제안된 알고리즘을 설계된 EMFG에 적용하게 되면 시스템의 설계도 자동화될 수 있을 것으로 본다. 뿐만 아니라 설계된 EMFG가 올바르게 동작되는 것이 확인되면 쉽게 실제회로로 변환될 수 있으므로 시스템의 설계 및 구현에 시간을 단축할 수 있어 더욱 유리할 것이다.

참 고 문 헌

- [1] 여정모, "마크흐름선도의 확장," 부산대학교 대학원 석사학위 논문, 1982. 2.
- [2] 여정모, 황창선, "확장된 마크흐름선도와 시퀀셜 제어시스템에의 응용," 부산대학교 공과대학 연구보고, 제25집, pp.209-219, 1983. 6.
- [3] 여정모, "EMFG 회로의 간략화에 관한 연구," 부산개방대학 연구보고, 제29집, pp.741-760, 1987. 12.
- [4] K. Hasegawa, "Mark Flow Graph and Its Application to FA," Journal of the SICE, Vol.22, No.11, pp.946-951, Nov. 1983.
- [5] K. Hasegawa, K. Takahashi, R. Masuda and H. Ohno, "Proposal of Mark Flow Graph for Discrete System Control," Trans. of SICE, Vol.20, No.2, pp.122-129, Feb. 1984.
- [6] J. Ayache, M. Diaz and R. Valette, "A Methodology for Specifying Control in Electronic Switching System," Proceedings of the International Switching Symposium, Paris, pp.1049-1056, May 1979.
- [7] K. Hasegawa, K. Takahashi and P. E. Miyagi, "Application of the Mark Flow Graph to Represent Discrete Event Production Systems and System Control," Trans. of SICE, Vol.24, No.1, pp.69-75, Jan. 1988.
- [8] P. E. Miyagi, K. Hasegawa and K. Takahashi, "A Programming Language for Discrete Event Production Systems Based on Production Flow Schema and Mark Flow Graph," Trans. of SICE, Vol.24, No.2, pp.183-190, Feb. 1988.
- [9] Ferrarini, L., "An incremental approach to logic controller design with Petri Nets," IEEE Transaction on System, Man, and Cybernetics, Vol.22, No.3, pp.461-473, 1992.
- [10] A. Falcione and B. Krogh, "Design Recovery for Relay Ladder Logic," IEEE Control Systems Magazine, pp.90-98, April 1993.
- [11] R. Willson and B. Krogh, "Petri net tools for the specification and anaysis of discrete controllers," IEEE transaction on Software Engineering, Vol.16, No.1, pp.39-50, Jan. 1990.
- [12] M. Jeng and F. DiCesare, "A review of syn- thesis techniques for Petri Nets with applications to automated manufacturing systems," IEEE transaction on Systems, Man and Cybernetics, Vol.23, No.1, pp.301-312, Jan. 1993.
- [13] A. Merabet, "Synchronization of operations in a flexible manufacturing cell: the Petri Net approach," Journal of Manufacturing Systems, Vol.5, No.3, pp.161-169, 1986.
- [14] D. Y. Lee and F. DiCesare, "Integrated models for scheduling flexible manufacturing systems," Proceeding of the 1993 IEEE International Conference on Robotics and Automation, Atlanta, GA, pp.827-832, May 1993.
- [15] T. Murata, N. Komoda, K. Matsumoto and

K. Haruna, "A Petri Net-based Controller for Flexible and Maintainable Sequence Control and its Applications in Factory Automation," IEEE Transaction on Industrial Electronics, Vol.33, No.1, pp. 1-8, Jan. 1986

[16] M. Zhou, DiCesare and F. Desrochers, "A Hybrid Methodology for Synthesis of Petri Net Models for Manufacturing Systems," IEEE Trans. on Robotics and Automation, Vol.8, No.3, pp.350- 361, 1992.

[17] 여정모, 하재목, "확장된 마크흐름선도의 재구성 및 회로변환," 한국멀티미디어학회, 1998년도 춘계학술발표논문집, pp.423-431, 1998. 6.



여 정 모

1980년 동아대학교 전자공학과 졸업(공학사)

1982년 부산대학교 대학원 전자공학과(공학석사)

1993년 울산대학교 대학원 전자및 전산기공학과(공학박사)

1986년 - 현재 부경대학교 전자계산학과 부교수

관심분야 : 마이크로프로세서, CAD