순차논리시스템을 이용한 제산기 구성에 관한 연구

박춘명*

A Study on Constructing the Divider using Sequential Logic Systems

Chun-Myoung Park*

요 약

본 논문에서는 유한체상의 순차논리시스템을 이용하여 제산기를 구성하는 방법을 제안하였다.

제안한 방법은 먼저 유한체와 순차논리시스템의 수학적 성질을 기반으로 현재상태와 차순상태 사이의 선형성질을 도출하였다. 그리고, 선형성질과 행렬로 표현한 특성다항식을 사용하여 유한체상의 순차논리시스템을 구현하였으며, 이를 이용하여 제산기를 구현하였다. 제안한 방법은 기존의 방법에 비해 규칙적이고 좀 더 효율적으로 순차논리시스템을 구현할 수 있었으며, 이를 이용하여 효과적인 제산기를 구현할 수 있었다.

ABSTRACT

This paper presents a method of constructing the divider using sequential logic systems over finite fields(or galois fields). The proposed the sequential logic systems is constructed by as following. First of all, we obtain the linear characteristics between present state and next state based on mathematical properties of finite fields and sequential logic systems. Next, we realize the sequential logic systems over finite fields using above linear characteristics and characteristic polynomial which is expressed using by matrix. Also, we apply to implement divider using the proposed sequential logic systems over finite fields.

키워드

유한체, 특성다항식, 순차논리시스템, 선형성질, 현재 및 차순 상태

Key word

Finite fields, characteristics polynomial, sequential logic systems, linear characteristics, present and next state.

* 충주대학교 컴퓨터공학과 (교신저자, cmpark@cjnu.ac.kr)

접수일자 : 2010. 02. 09
심사완료일자 : 2010. 03. 02

Ⅰ. 서 론

유한체인 Galois체를 근간으로 여러 분야들이 발전되어 왔다. 특히, 이중에서 P=2인 경우에는 현존의 2진 논리에 해당하는 디지털시스템의 근간인 부울대수에 귀착되며 오진정정코드 분야를 비롯하여 디지털스위칭이론[1-3], 디지털신호처리 및 디지털정보처리[4], 디지털통신에서의 디지털정보에 대한 암호화(Encryption)와 복호화[5], 고속산술연산기구성[6-7] 등과 같은 여러 분야에 걸쳐 적용 및 응용되고 있다. 일반적으로 유한체[8-9]는 기초체 GF(P)와 이의 확대인 확대체 $GF(P^m)$으로 분류되며, 본 논문에서는 기초체인 GF(P) 상에서의 순차논리시스템을 구성하는 방법을 제안하였다.

본 논문의 서술과정은 다음과 같다.

Ⅱ장에서는 유한체상에서의 순차논리시스템의 수학적 성질을 논의하였으며, Ⅲ장에서는 유한체상의 순차논리시스템의 구성에 사용되는 선형궤환시프트레지스터와 이에 대한 행렬표현에 논의하였다. 그리고 Ⅳ장에서는 Ⅲ장에서 구성한 유한체상의 순차논리시스템을 제산연산을 수행하는 제산기 구성에 적용하는 예를 들었다. 마지막 Ⅴ장에서는 본 논문에서 제안한 유한체상의 순차논리시스템 구성의 특징을 요약하였다.

Ⅱ. 유한체상의 순차논리시스템의 수학적 성질

유한체 GF(P)상에는 P개의 원소가 존재하고 이들 원소를 ei(i=0,1,2, ... ,P-1)로 표시한다. 이외의 유한체에 대한 수학적 성질은 참고문헌[8-9]을 참조하였다. 한편, 순차논리시스템의 출력은 조합논리시스템과는 달리 현재입력 뿐만 아니라 과거의 입력에 의해서도 결정되는 특징을 갖고 있다. 따라서, 순차논리시스템에서 현재의 출력은 지연소자 또는 기억소자에 의해서 그 정보가 입력으로 궤환되어져야 한다.

일반적으로 순차논리시스템인 M은 다음 식(1)과 같이 5-tuple로 표현된다.

$$M=(S, I, Z, \delta, \lambda) \quad (1)$$

$$S, I, Z=e_i \in GF(P)(i=0,1,2, ... ,P-1)$$

여기서, S는 상태, I는 입력, Z는 출력, δ는 차순상태함수, λ는 출력함수를 각각 의미한다.

또한, 위 식(1)은 다음 식(2)와 같은 사상(mapping) 관계를 갖는다.

$$S_t \times I_t \xrightarrow{\delta} S_{t+1} \quad (2)$$

여기서, St는 현재상태이며 St+1은 차순상태이다.

한편, λ는 Mealy model과 Moore model에 따라 다음 식(3)과 식(4)와 같다.

• Mealy model :

$$S_t \times It \xrightarrow{\lambda} S_{t+1} \quad (3)$$

• Moore model :

$$S_t \xrightarrow{\lambda} S_{t+1} \quad (4)$$

즉, Moore model에서의 출력은 오직 현재상태의 함수로만 이루어진다.

한편, 유한체 GF(P)상의 행렬(Matrix)을 각각 **A**, **B**, **C**, **D** 라 하면 위 식(2), (3), (4)는 각각 다음 식(5), (6), (7)로 표현 할 수 있다.

$$\delta(S_t, I_t) = S_{t+1} = \mathbf{A} \bullet S_t + \mathbf{B} \bullet I_t \quad (5)$$

$$Z_t = \mathbf{C} \bullet S_t + \mathbf{D} \bullet I_t \quad (6)$$

$$Z_t = \mathbf{C} \bullet S_t \quad (7)$$

Ⅲ. 순차논리시스템구성

3-1. 선형궤환시프트레지스터구성

Ⅱ장의 식(5)에서 살펴 본봐와 같이 시간 t에서의 현재상태 S_t와 시간 (t+1)에서의 차순상태 S_{t+1}은 선형관계가 존재한다. 따라서, 위 내용과 다음의 그림1, 2, 3의 회로소자들을 사용하여 선형궤환시프트레지스터를 구성하면 다음 그림4와 같다.

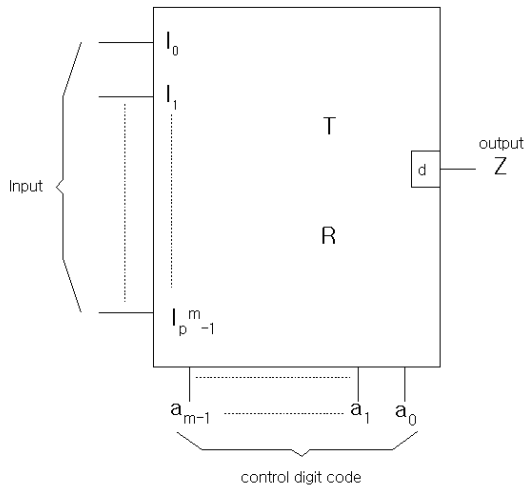

그림 1. modP 스칼라 곱 회로 심볼
Fig. 1. The circuit symbol modP scalar product.

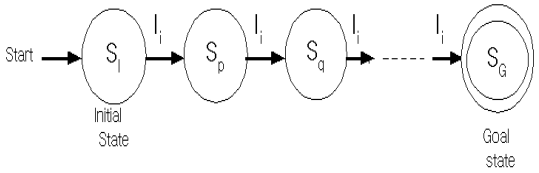

그림 2. modP 가산연산 심볼
Fig. 2. The circuit symbol of modP addition.

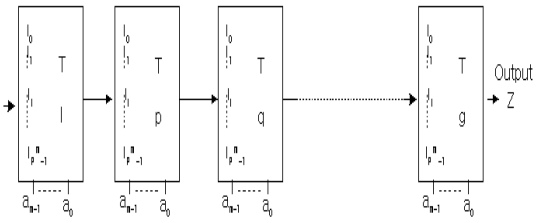

그림 3. modP 승산연산 심볼
Fig. 3. The circuit symbol of modP multiplication

특히, P=2인 경우는 ⊙는 AND 게이트로 대체된다.

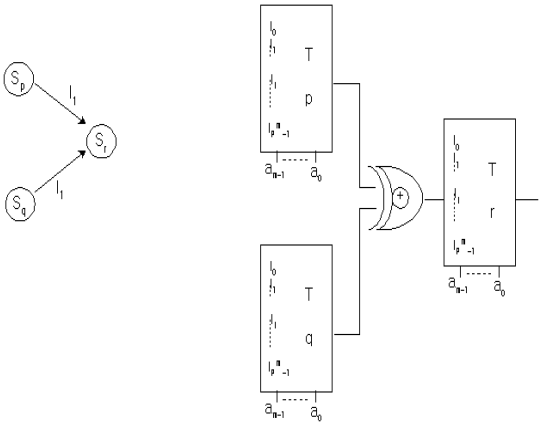

그림. 4. 선형궤환레지스터
Fig. 4. Linear feed-back shift register.

여기서, $S_0 = a_n \bullet S_{n-1}$

$S_t = a_{n-1} \bullet S_{n-1} + S_{t-1} (0 < t < n)$

$a_i \in GF(P)(i=0,1, ... , n-1)$

3-2. 선형궤환레지스터의 행렬 표현

선형궤환레지스터의 행렬 **A**에 대한 특성다항식(Characteristics polynomial)은 다름 식(8)로부터 구할 수 있다.

$$\det(\mathbf{X} \bullet \mathbf{I} - \mathbf{A}) \tag{8}$$

여기서 **I**는 Identity 행렬이다.

이제 식(8)로부터 선형궤환시프트레지스터의 특성다항식은 다음 식(9)에 의해 구할 수 있다.

$$P(X)=Xn - a_1 \bullet X_{n-1} - a_2 \bullet X_{n-2} - ..... - a_n \tag{9}$$

여기서 $a_i(i=1,2, ... , n) \in GF(P)$

Ⅳ. 적용 예

이 장에서는 Ⅲ장의 내용이 어떻게 적용되는지 GF(P) 상의 제산연산 처리에 적용하여 각각 몫과 나머지를 구하는 과정을 살펴보기로 한다.

[예1] GF(3)상의 순차논리머시인 구성

다음 그림5와 같은 GF(3)상의 임의의 상태천이도(State-transition diagram)가 주어졌다고 하면, 각각 차순 상태함수는 다음 식(10), (11), (12)와 같다.

$$[S_0]_{t+1} = [S_2]t + \text{Input} \quad (10)$$

$$[S_1]_{t+1} = [S_0]t + 2 \cdot [S_2]t \quad (11)$$

$$[S_2]_{t+1} = [S_1]_t \quad (12)$$

따라서 행렬 **A**는 다음 식(13)과 같다.

$$\mathbf{A} = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 2 \\ 0 & 1 & 0 \end{bmatrix} \quad (13)$$

한편, 특성다항식을 구하면 다음 식(14)와 같다.

$$\det(\mathbf{X} \cdot \mathbf{I} - \mathbf{A}) = X_3 + X + 2 = (1012) \quad (4\text{-}5)$$

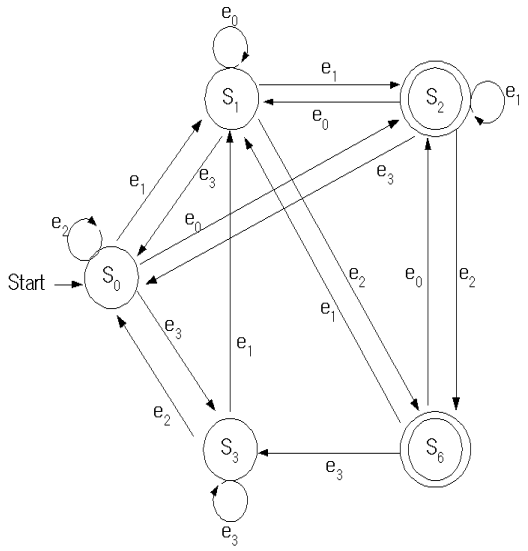

그림. 5. GF(3)상의 상태천이도
Fig. 5. State-transition diagram over GF(3)

이제 식(14)와 Ⅲ장의 식(9)를 토대로 선형궤환시프트레지스터를 사용한 순차논리머시인을 구성하면 다음 그림6과 같다.

여기서, 입력 데이터 열이 $X_7 + 2 \cdot X_6 + X_4 + 2 \cdot X_2 + X$ = (12110210)이라면, 이는 $(X_7 + 2 \cdot X_6 + X_4 + 2 \cdot X_2 + X)/(X_3 + X + 2)$와 같고 이를 수행하는 과정은 다음 표1과 같다.

표 1. 입력 데이터 (12110210)에 대한 수행과정
Table 1. Procedure of input data (12110210)

		S_0 S_1 S_2	Output	
	Initial	0 0 0		
	1	1 0 0	0	
	2	2 1 0	0	
Input	1	1 2 1	1	
data	1	2 0 2	2	Quotient
stream	0	2 0 0	0	
	2	2 2 2	0	
	1	1 2 2	2	
	0	2 2 1		
		Remainder		

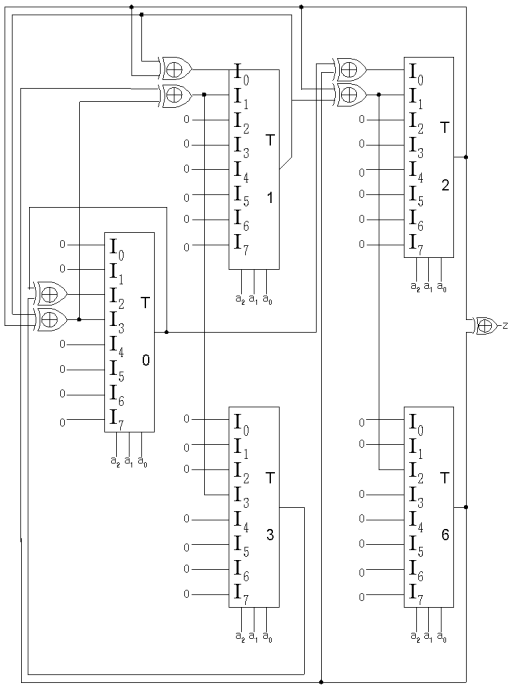

그림 6. 그림 4-1의 순차논리머시인
Fig. 6. Sequential logic systems of fig.5.

[예2] GF(2)상의 순차논리머시인 구성

다음 그림4-3과 같은 GF(2)상의 임의의 상태천이도가 주어졌다고 하면, 각각 cktnstkdxogkantn는 다음 식(15), (16), (17)과 같다.

$$[S_0]_{t+1} = [S_2]_t + \text{Input} \quad (15)$$

$$[S_1]_{t+1} = [S_0]_t \quad (16)$$

$$[S_2]_{t+1} = [S_1]_t + [S_2]_t \quad (17)$$

따라서, 행렬 **A**는 다음 식(18)과 같고 위 식(15), (16), (17)은 다음 식(19)와 같이 표현 할 수 있다.

$$\mathbf{A} = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 1 \end{bmatrix} \quad (18)$$

$$[S_0\ S_1\ S_2]T_{t+1} = \mathbf{A} \cdot [S_0\ S_1\ S_2]T_t \quad (19)$$

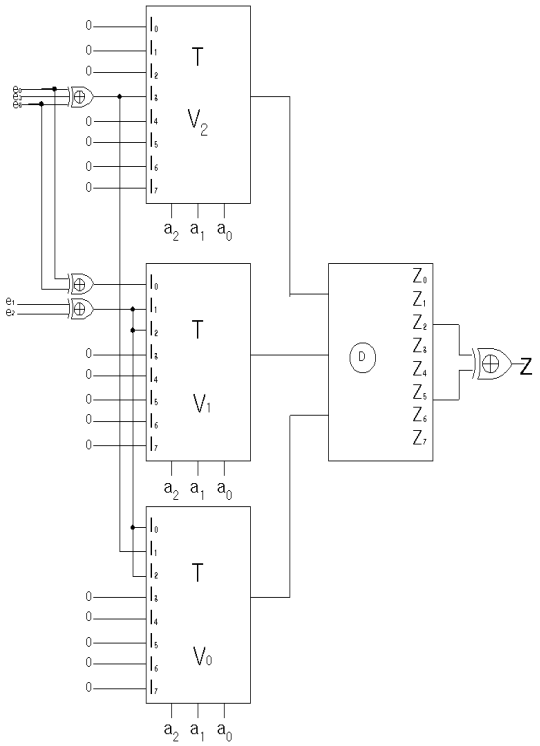

그림. 7. GF(2)상의 상태천이도
Fig. 7. State-transition diagram over GF(2)

한편, Ⅲ장의 식(9)에 의해 특성다항식을 구하면 다음 식(20)과 같다.

$$\det(\mathbf{X} \cdot \mathbf{I}\text{-}\mathbf{A}) = X_3 + X_2 + 1 = (1101) \quad (20)$$

이제 식(15)와 Ⅲ장의 식(9)를 토대로 선형궤환시프트레지스터를 사용한 순차논리머시인을 구성하면 다음 그림8과 같다.

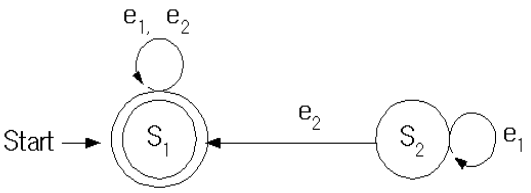

그림 8. 그림7의 순차논리머시인
Fig. 8. Sequential logic systems of fig.7.

여기서, 입력 데이터 열이 $X_5 + X_3 + X = (101010)$이라면, 이는 $(X_5 + X_3 + X)/(X_3 + X_2 + X)$와 같고 이를 수행하는 과정은 다음 표2와 같다. 표2에서 몫은 그림8의 출력이고, 나머지는 시프트레지스터에 남아있는 값이다.

표 2. 입력 데이터 (101010)에 대한 수행과정
Table 2. Procedure of input data (101010)

		$S_0\ S_1\ S_2$	Output	
	Initial	0 0 0		
	1	1 0 0	0	
Input	0	0 1 0	0	
data	1	1 0 1	1	Quotient
stream	0	1 1 1	1	
	1	0 1 0	0	
	0	0 0 1		
		Remainder		

V. 결 론

본 논문에서는 유한체상의 순차논리시스템을 구성하는 방법을 제안하였다. 제안한 방법은 먼저 유한체 GF(P)상에서의 순차논리시스템의 수학적 성질을 논의하였으며, 순차논리시스템의 구성을 위하여 기본 3가지 회로소자를 사용하여 선형궤환시프트레지스터와 이에 대한 행렬표현에 대해 논의하였다. 그리고, 제안한 방법을 제산연산처리에 적용하였다. 제안한 방법은 기존의 방법에 비해 연산속도와 그 효율성이 개선됨을 알 수 있었다. 향 후 연구과제로는 제산연산처리 이외의 분야에 적용하는 연구가 요구되며, 또한 GF(P)의 확대체인 $GF(P^m)$(여기서, P는 소수, m은 양의 정수)상에서 대해서도 본 논문에서 제안한 순차논리머시인을 확장할 수 있으리라 사료되며, 각종 디지털시스템에 응용 할 수 있으리라 전망된다.

참고문헌

[1] David Green, Modern Logic Design, Addison-Wesley Publishing Company, 1986.

[2] E.J.McClusky, Logic Design Principles, Prentice-Hall, 1986.

[3] D.Lee, A.Abuda Gaffar, O.Mencer, and W.Luk, "Optimizing hardware function evaluation," IEEE Trans. Comput. pp.1520-1531, vol.54, Dec. 2005.

[4] R.E.Blahut, Fast Algorithms for Digital Signal Processing, Addison-Wesley Publishing Company, Inc., 1985.

[5] I.F.Blake,Algebraic Coding Theory:History and Development, Down, Hutchinson & Ross, Inc., Stroudburg, Pennsylvania,1973.

[6] M.E.Kaihara and N.Takagi,"A hardware algorithm for modular multiplication/division," IEEE Trans. Comput. pp.12-21, vol.54, Jan. 2005.

[7] H.Wu,"Bit-parallel polynomial basis multiplier for new classes of finite fields," IEEE Trans. Comput. pp.1023-1031, vol.57, Aug. 2008.

[8] E.Artin, Galois Theory, NAPCO Graphic arts, Inc., Wisconsin.1971.

[9] R.J.McEliece, Finite Fields for Computer Science and Engineers, Kluer Academic Publishers, 1987.

저자소개

박춘명(Chun-Myoung Park)

한국해양정보통신학회 논문집
제12권 제12호 참조