개선된 하드웨어 산술연산기 구성

박춘명*

*한국교통대학교

A Construction of the Improved Hardware Arithmetic Operation Unit

Chun-Myoung Park*

*Korea National University of Transportation

E-mail : cmpark@ut.ac.kr

요 약

본 논문에서는 Galois체에 기초를 둔 고효율 산술연산기 구성에 관한 한가지 방법을 제안하였다. 제안한 연산기는 기존의 방법에 비해 좀 더 규칙적이고 확장성이 용이한 이점이 있으며, 또한, 각종 멀티미디어 하드웨어 구성시의 기본인 연산기로 적용 및 응용할 수 있다. 향 후 연구과제로는 좀 더 콤팩트하고 효과적인 산술연산 알고리즘의 도출이 필요하며, 이에 논리연산기를 접목하여 산술연산 및 논리연산을 수행하는 연산전용 프로세서의 개발이 필요하다.

ABSTRACT

This paper propose the method of constructing the improved hardware arithmetic operation unit over galois fields. The proposed the hardware arithmetic operation unit have advantage which is more regularity and extensibility compare with earlier method. Also it is able to apply to any multimedia hardware which is the basic arithmetic operation unit. For the future we will research the processor which is the processing arithmetic and logical operation.

키워드

Arithmetic operation, logical operation unit, regularity, extensibility etc.

Ⅰ. 서 론

최근에 멀티미디어 H/W와 S/W에 기반을 둔 여러 분야가 매우 급속도로 발전되고 있으며 21C에는 더욱 더 활용 및 요구될 것이다. 특히 멀티미디어 H/W는 지금까지의 각종 데이터 처리보다는 훨씬 방대한 데이터 량, 최적의 데이터 압축 및 복원, 초고속 전송 등의 복합적이고 고기능의 기술이 요구되고 있다.[1-8] 따라서 본 논문에서는 이를 해결할 수 있는 방법으로서 각종 멀티미디어 H/W 시스템에서 기본적으로 사용되는 산술연산을 효율적으로 수행할 수 있는 Galois체에 기초를 둔 연산기의 구성에 대한 한가지 방법을 제안한다.

Ⅱ. 연산 알고리즘

2-1. 가산 연산 알고리즘

피가산원소를 ei, 가산원소를 ej, 가산후원소를 ea라 하고 이들을 벡터공간으로 표현한 것을 각각 ei(aV), ej(bV), ea(AV)라 하면 두 원소 ei와 ej의 가산은 다음 식(2-1)과 같다.

$$ei \oplus ej = ei(aV) \oplus ej(bV) = ea(AV) \quad (2\text{-}1)$$

여기서 i,j=0,1,.....,2m-2,2m-1이고, aV,bV,AV∈GF(2) (V=0,1,.....,m-2,m-1)이고, ⊕는 modP 합이다.

2-2. 감산 연산 알고리즘

특히, P=2인 경우에는 Mod2의 수학적 성질에 의해 감산은 가산과 같다. 따라서 두원소간의 감산 알고리즘은 가산 알고리즘과 같다.

2-3. 승산 연산 알고리즘

피승산원소를 e_i, 승산원소를 e_j, 승산후원소를 e_M이라 하고 이들을 벡터공간으로 표현한 것을 각각 $\underline{e_i}(a_V)$, $\underline{e_i}(b_V)$, $\underline{e_M}(M_V)$라 하면 두 원소 e_i와 e_j의 승산은 다음 식(2-2)과 같다.

$$e_i \otimes e_i = \underline{e_i}(a_V) \otimes \underline{e_j}(b_V) = \underline{e_m}(M_V) \quad (2\text{-}2)$$

한편, 피승산원소, 승산원소, 승산후원소의 기약다항식을 각각 $F(\xi)=\sum_{i=0}^{m-1} a_i \xi^i$, $G(\xi)=\sum_{j=0}^{m-1} b_j \xi^j$와 $H(\xi)=\sum_{k=0}^{m-1} M_k \xi^k$라 하면 식(2-2)는 다음 식(2-3)과 같이 표현 할 수 있다.

$$\begin{aligned} F(\xi)\otimes G(\xi) &= \left(\sum_{i=0}^{m-1} a_i \xi^i\right)\otimes\left(\sum_{j=0}^{m-1} b_j \xi^j\right) \\ &= \sum_{i=0}^{m-1}\left(\sum_{j=0}^{m-1} b_j\right) a_i \xi^{i+j} \\ &= \sum_{i+j=0}^{2m-2} a_i b_j \xi^{i+j} \end{aligned} \quad (2\text{-}3)$$

여기서 a_i, $b_j \in GF(2)$이고 i,j=0,1,.....,m-2,m-1이다.

한편, 여기서 r=i+j라 하면 식(2-3)는 식(2-4)와 같고 이는 $H(\xi)$와 같아야 한다.

$$\begin{aligned} F(\xi)\otimes G(\xi) &= \sum_{r=0}^{2m-2} a_i b_j \xi^r \\ &= H(\xi)=\sum_{k=0}^{m-1} M_k \xi^k \end{aligned} \quad (2\text{-}4)$$

따라서 ξ^r의 r은 $m \le r1 \le 2m-2$ 부분과 $0 \le r2 \le m-1$ 부분으로 분할 할 수 있으며 ξ^{r1}항을 수학적 성질로부터 ξ^{r2}항으로 표현하여 ξ^k항과 일치시킬 수가 있다. 또한, 이들 ξ^{r2}항들이 승산기 모듈중 제어입력생성 모듈의 입력이 되고 이 제어입력 C_L에 의해 최종 승산후원소 $\underline{e_m}(M_V)$를 얻는다.

Ⅲ. 분배기 구성

본 장에서는 앞에서 논의한 연산을 P=2인 경우의 분배기 구성에 대해 논의한다. 피연산원소인 $\underline{e_i}(a_V)$는 가산일때는 가산기 모듈의 입력으로 승산일때는 ξ^r 생성 모듈의 입력으로 사용된다. 그러므로 이를 수행하기 위한 제어입력 T_1과 패스 트랜지스터 G_{D1i}(i=0,1)로 모듈 D_1의 기본 셀을 구성할 수 있다. 이를 식으로 표현하면 다음 식(3-1)과 같고 이를 토대로 D_1 모듈을 구성할 수 있으며 이에 대한 진리치표는 표3-1과 같다.

$a_i = y_{0i}$ if T1=0 ⇨ 가산기 모듈

y_{1i} if T1=1 ⇨ 승산기 모듈 (3-1)

여기서 i=0,1,.....,m-2,m-1이다.

표 3-1. 모듈 D1의 기본셀(D1-cell)에 대한 진리치표.

Table 3-1. Truth table for basic cell(D1-cell) of module D1.

a_i	T1	G_{D10}	Y_{0i}	G_{D11}	Y_{1i}
a_i	0	ON	a_i	OFF	×
a_i	1	OFF	×	ON	a_i

where, × means nonpass

Ⅳ. 결 론

본 논문에서는 최근에 그 활용과 향 후 21C에 많이 적용되는 멀티미디어 H/W 시스템에 반드시 필요한 고속 연산기 구성의 한가지 방법을 제안하였다. 제안한 고속 연산기는 Galois체 $GF(2^m)$상에서 구성하였다. 가산기 모듈은 가산, 감산, 승산, 제산의 어떤 연산을 하더라도 항상 사용된다. 또한, 제안한 고속 연산기는 모듈들의 합성으로 구성되므로 m의 확장에 따른 고속 연산기는 각 모듈을 m에 따라 확장만 하면 되며 최종 분배기 모듈로서 합성하여 용이하게 구성할 수 있다.

참 고 문 헌

[1] I. F. Blake, *Algebraic Coding Theory : History and Development*, Down, Hutchinson & Ross, Inc., Stroudburg, Pennsylvania, 2010.

[2] R. Lidi and G. Pilz, *Applied Abstract Algebra* ,Spring-Verlg, Inc., N.Y., 2011.

[3] R. E. Blahut, *Fast Algorithms for Digital Signal Processing*, Addison-Wesley Publishing Company, Inc., 2012.

[4] S. Y. Kung, *VLSI ARRAY PROCESSORS*, Prentice-hall, Inc., 2010.

[5] K, Bromley, Sun-Yuan Kang and E. Swartzlander, *International Confernece on SYSTOLIC Array*, Computers Society Press, N.Y., 2010.

[6] M. D. Ercegovac and T. Lang, *Digital Systems and Hardware/Firmware Algorithms*, John Wiley & Sons, Inc., Canadea, 2011.

[7] B. A. Laws and C. K. Rusforth,"A cellular-array multiplier for $GF(2^m)$,IEEE Trans.Compt., Short-notes, pp.1573-1578, Dec., 2010.

[8] C. S. Yeh, I. S. Reed and T. K. Trung,"Systolic multiplier for finite filds $GF(2^m)$,"IEEE Trans. Comput., vol.C-33, pp.357-360, Apr. 2010.