A Study on Constructing Inverse Element Generator over GF(3^m)

Chun-Myoung Park* and Hong-Bok Song**
*Dept. of Computer Engineering, School of Electrical Electronic and Information Engineering,
Chungju National University, Chungju City, 380-702, Korea
Tel : +82-41-841-5346 Fax : +82-41-841-5340 E-mail: cmpark@chungju.ac.kr
**Dept. of Electronic Engineering, Dongeui University, Busan City, Korea
Tel : +82-51-890-1679 E-mail: hbsong@deu.ac.kr

Abstract : This paper presents an algorithm generating inverse element over finite fields GF(3^m), and constructing method of inverse element generator based on inverse element generating algorithm.

A method computing inverse of an element over GF(3m) which corresponds to a polynomial over GF(3^m) with order less than equal to m-1.

Here, the computation is based on multiplication, square and cube method derived from the mathematics properties over finite fields.

Key Words : finite fields, irreducible polynomial, arithmetic operation, inverse element

1. INTRODUCTION

In recently, many digital logic system and computer systems are required to process more data and high efficiency.[1-4] Specially arithmetic relation is require more high speed and efficiency.

The advanced arithmetic operation is analyzed easily using by finite fields.[5-8]

In this paper, we propose the high efficiency inverse element generator over finite fields.

This paper's organization is as following.

In Section 2, we propose the multiplication, square and cube algorithm based on finite field mathematical properties.

In section 3, we discuss of inverse element generation algorithm over GF(3^m).

And in section 4, we construct an inverse element generator based on section 3's algorithm.

Finally in section 6, we summary our inverse element generator over finite fields, also we prospect future research fields.

2. MULTIPLICATION, SQUARE AND CUBE ALGORITHM OVER FINITE FIELDS

2.1. Mathematical Properties of Finite Fields

In this section, we review the important mathematical properties of finite fields, these mathematical properties used in build up this paper.

Any other mathematical properties except these mathematical properties refer to references[9-12].

Finite fields is defined by any prime number P and integer m, namely GF(P^m).

In generally finite fields is organized by 5-tuple {S,+,·,0,1}, where S is set of elements, + and · are binary operation over S, 0 and 1 are each identity element for addition and multiplication arithmetic operation.

Also finite fields are classified into ground fields GF(P) and extension fields GF(P^m). The number of elements over ground fields GF(P), P is the prime number more than 1, are {0,1,2,......,P-1}.

The important mathematical properties over finite fields are as following.

[Mp1] for $\alpha \in GF(P^m)$, $\alpha^{\psi}=\alpha$ and $\alpha^{\psi-1}=1$ ($\psi=p^m$) in case of $\alpha \neq 1$.
[MP2] for α, $\beta \in GF(P^m)$ and arbitrary integer m, $(\alpha \pm \beta)^{\mu}=\alpha^{\mu} \pm \beta^{\mu}$ ($\mu=p^m$)
[MP3] for $\alpha \in GF(P^m)$, $\alpha^i \bullet \alpha^j=\alpha^{i+j(\text{mod } \psi-1)}$ ($\psi=p^m$)
[MP4] The elements in GF(P^m) are represented by $F(\alpha)=\sum_{i=0}^{m-1} a_i\alpha^i$, where α is root, that is, α have coefficient that have element for integer field Z_P that have mod P, where m degree primitive irreducible polynomial $F(X)=X^m + f_{m-1}X^{m-1} + f_{m-2}X^{m-2} + \ldots\ldots + f_1X^1 + f_0$, $a_i \in Z_P(i=0,1,2, \ldots\ldots ,m-1)$ and $f_0 \neq 0$.

2.2. Algorithm of multiplication and square over GF(3^m)

GF(3^m) have 3m elements and we represented this with set is as following expression (2-1).

$$GF(3^m)=\{0, \alpha, \alpha^2, \ldots\ldots , \alpha^{\psi-2}=\alpha^{-1}, \alpha^{\psi-1}=1\} \qquad (2\text{-}1)$$

where, $\psi=3^m$.

Then, for more than m degree term, we can represent degree m-1 polynomial using mod F(X) arithmetic that used by primitive polynomial F(X).

Therefore arbitrary element over GF(3^m) can represent as following.

$$F(X)= a_{m-1}X^{m-1} + a_{m-2}X^{m-2} + \ldots + a_1X^1 + a_0 = \sum_{i=0}^{m-1} a_iX^i \qquad (2\text{-}2)$$

where, $a_i \in GF(3)$ and $i=0,1,2, \ldots\ldots, m-1$.

On the other hand, α is a root of primitive polynomial, therefore α^m is equal to following equation (2-3), where + is mod3.

$$\alpha^m + a_{m-1}\alpha^{m-1} + \ldots + a_1\alpha + a_0 = 0$$

$$\alpha^m = -a_{m-1}\alpha^{m-1} - a_{m-2}\alpha^{m-2} - \ldots - a_1\alpha - a_0$$

$$\alpha^m = (3 - a_{m-1})\alpha^{m-1} + (3 - a_{m-2})\alpha^{m-2} + \ldots + (3-a_1)\alpha + (3 - a_0) \qquad (2\text{-}3)$$

Therefore, we can represent element over $GF(3^m)$ into polynomial α of maximum m-1 degree, it is as following.

$$GF(3^m) = \{a_{m-1}\alpha^{m-1} + a_{m-2}\alpha^{m-2} + \ldots + a_1\alpha + a_0\}$$

where, $a_i \in GF(3)$ and $i=0,1,2, \ldots\ldots, m-1$.

For two elements over $GF(3^m)$, $e1 == \sum_{i=0}^{m-1} a_i\alpha^i$, $e2 == \sum_{i=0}^{m-1} b_j\alpha^j$, $e1 \bullet e2$ is represented as following.

$$e1 \bullet e2 = \sum_{i=m-1}^{0} P_i\alpha^i + \sum_{i=2m-2}^{m} P_i\alpha^I \qquad (2\text{-}4)$$

We consecutive multiply expression (2-4), generate maximum degree α^{2m-2}, we mod F(X) from α^{2m-2} to α^m, we can obtain following expression.

$$e1 \bullet e2 = \sum_{i=m-1}^{0} R_i\alpha^i \qquad (2\text{-}5)$$

where, multiplication and addition are mod3 product and mod3 addition.

In case of square, e1 equal e2, expression (2-5) is represented by expression (2-6).

$$e1^2 = e1 \bullet e1 = e1 \bullet e2 = \sum_{i=0}^{m-1} R_i\alpha^i \qquad (2\text{-}6)$$

Therefore, we can use construct the square generator using multiplier.

2.3. Method of cube multiplication over $GF(3^m)$

According to mathematical properties [MP2], $(a+b)^3=a^3+b^3$, and $a^3=a$, $b^3=b$.

Therefore, cube for anyone element e over $GF(3^m)$ is represented by equation (2-7)

$$e^3 = e = \sum_{j=3m-3}^{0} P_j\alpha^j \qquad (2\text{-}7)$$

where, $P_j=a_j$ in case of $j=3i$, $P_j=0$ in case of $j=3i+1$ and $j=3i+2$, and $i=0,1, \ldots\ldots, m-1$.

That reason is as following.

$$e^3 = e = (a_{m-1}\alpha^{m-1} + a_{m-2}\alpha^{m-2} + \ldots + a_1\alpha + a_0)^3$$

$$= a^3_{m-1}\alpha^{3m-3} + a^3_{m-2}\alpha^{3m-6} + \ldots + a_1^3\alpha^3 + a_0^3$$
$$= a_{m-1}\alpha^{3m-3} + a_{m-2}\alpha^{3m-6} + \ldots + a_1\alpha^3 + a_0$$

For example, we obtain cube for any element over $GF(3^4)$.

Let any element over $GF(3^4)$ is e, we can represent .t as following.

$$e = a_3\alpha^3 + a_2\alpha^2 + a_1\alpha + a_0$$

where, $a_i \in GF(3)$ and $i=0,1,2$

Next, we select $F(X)=X^4+X+2$ for polynomial F(X).

And, we obtain cube as following.

$$e = (a_3\alpha^3 + a_2\alpha^2 + a_1\alpha + a_0)^3$$
$$= a_3\alpha^9 + a_2\alpha^6 + a_1\alpha^3 + a_0$$

On the other hand, $F(\alpha)=\alpha^4+\alpha+2$, therefore $\alpha^4=\alpha+2$, then $\alpha^6=2\alpha^3+\alpha^2$ and $\alpha^9=\alpha^3+\alpha^2+\alpha$.

Finally, we represent e as following.

$$e = (a_3 + 2a_2 + a_1)\alpha^3 + (a_9 + a_2 + a_1)\alpha^2 + a_3\alpha + a_0$$

3. ALGORITHM OF INVERSE ELEMENT GENERATION

In this section, we discuss algorithm of inverse element generation over $GF(3^m)$.

Let e is any one element over $GF(3^m)$, then its inverse element e^{-1} is as following equation(3-1).

$$e^{-1} = e^{\Psi-2} \qquad (3\text{-}1)$$

where, $\Psi = a^m$

Also, above inverse element is represented by triple product of 3.

$$e^{-1} = e \bullet (e^2)^{\Omega 1} \bullet (e^2)^{\Omega 2} \bullet (e^2)^{\Omega 3} \bullet \ldots\ldots \bullet (e^2)^{\Omega m-1}$$

where, $\Omega 1=3$, $\Omega 2=3^2$, $\Omega 3=3^3$, ……, $\Omega m-1=3^{m-1}$

The following is algorithm of inverse element generation over $GF(3^m)$, and the fig. 3-1 depicted algorithm.

[Algorithm]

STEP 1 : Accept any element e over $GF(3^m)$.

STEP 2 : Obtain result for power of element e.

STEP 3 : Cube product result after Step2 or Step 5.

STEP 4 : If it (m-1) times triple product do, go to Step 6.

STEP 5 : Product result of Step 3 with e, then go to Step 3.

STEP 6 : Product result of Step 4 with e.

STEP 7 : The inverse element e^{-1} is result of Step 6.

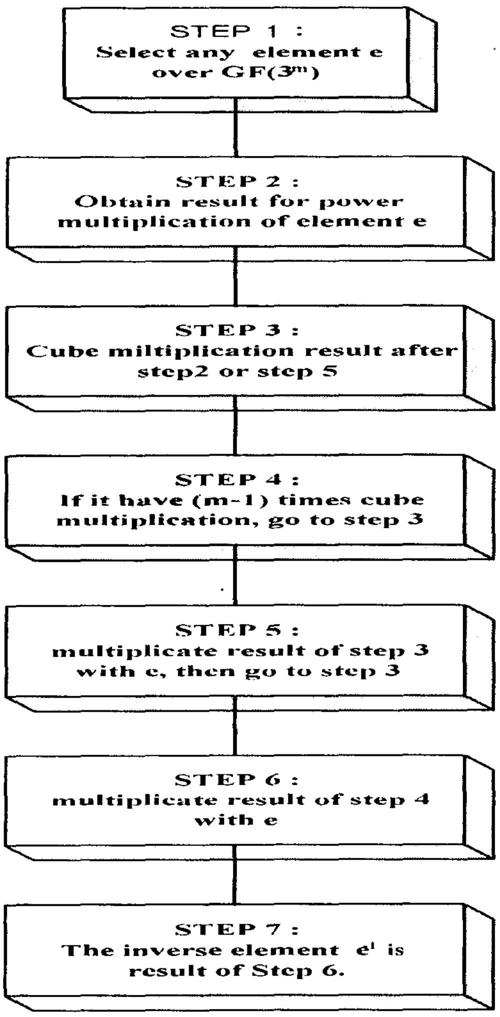

STEP 1 :
Select any element e over GF(3^m)
STEP 2 :
Obtain result for power multiplication of element e
STEP 3 :
Cube miltiplication result after step2 or step 5
STEP 4 :
If it have (m-1) times cube multiplication, go to step 3
STEP 5 :
multiplicate result of step 3 with e, then go to step 3
STEP 6 :
multiplicate result of step 4 with e
STEP 7 :
The inverse element e^{i} is result of Step 6.

Fig. 3-1. The block diagram of Inverse element generation over GF(3^m)

4. UNIVERSAL INVERSE ELEMENT GENERATOR

In this section, we discuss constructing the universal inverse element generator over GF(3^m).

The block diagram of universal inverse element generator over GF(3^m) is as following fig. 4-1.

As shown fig. 4-1, the universal inverse element generator was constructed with multiplier, square multiplier, cube multiplier.

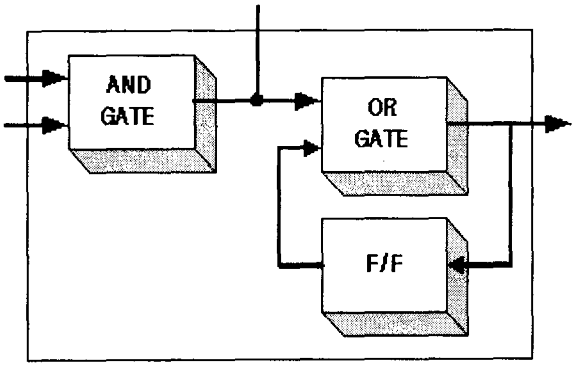
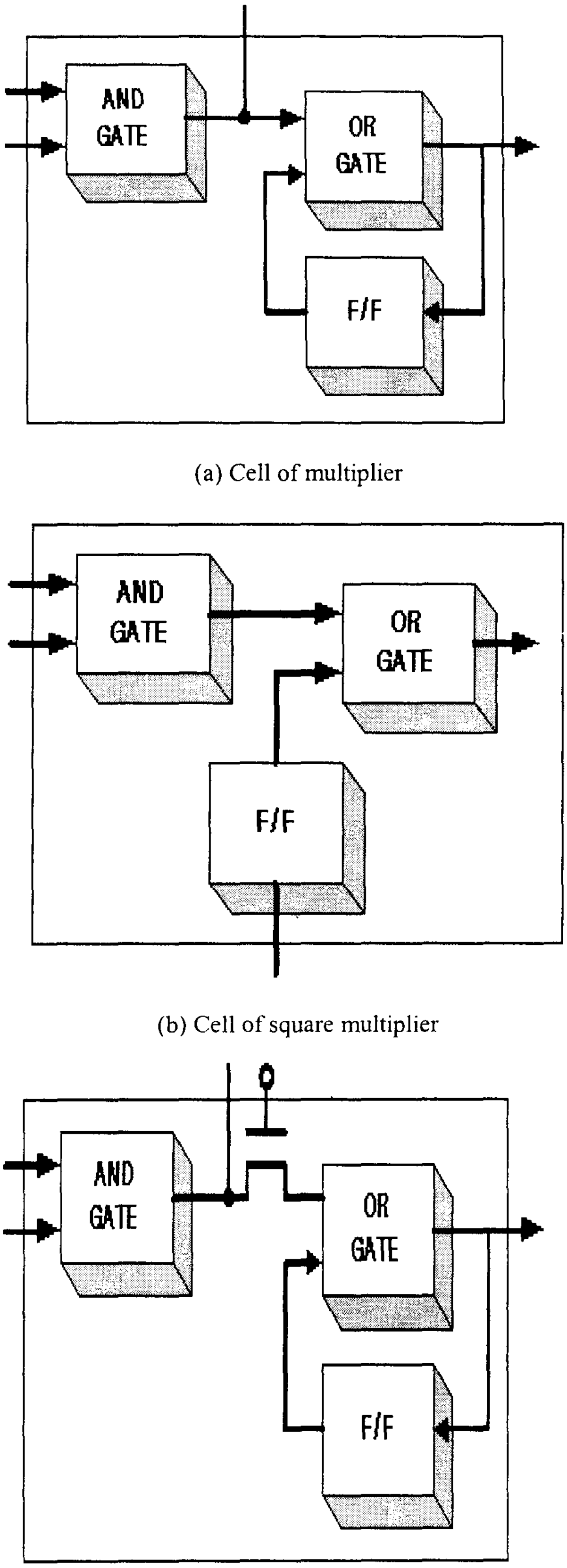


Fig. 4-1. Block diagram of universal inverse element generator over GF(3^m)

Aldo, the cell of multiplier, square multiplier and cube multiplier are fig.4-2 (a), fig.4-2 (b) and fig. 4-2 (c).



(a) Cell of multiplier

(b) Cell of square multiplier

(c) Cell of cube multiplier

Fig. 4-2. Each cell of universal inverse element generator over GF(3^m)

The multiplier over GF(3^m) is shown in fig. 4-3.

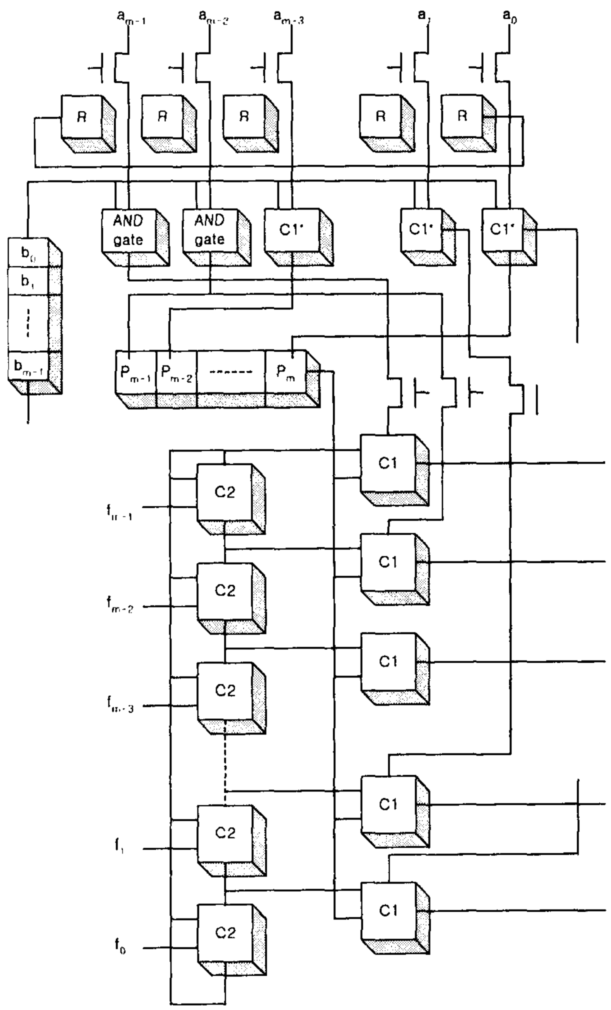


Fig. 4-3. Block diagram of multiplier over GF(3^m)

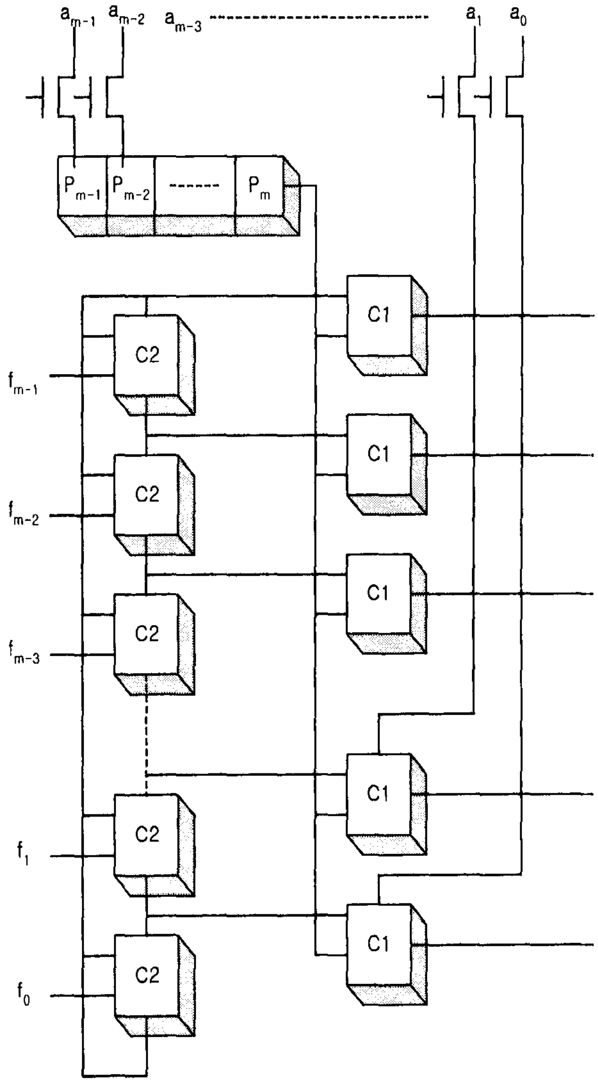


Fig. 4-4. Block diagram of cube multiplier over GF(3^m

The multiplier's characteristics don't change according to increase m or changing irreducible polynomial.

Also, if two input of multiplier, then multiplier is square multiplier.

Cube multiplier's basic type is same as multiplier, and then inputted element transferred into mod processing part without multiplication processing.

5. CONCLUSION

This paper propose a method of constructing the inverse element generation and inverse element generator over finite fields GF(3^m).

The proposed inverse element generator was constructed by serial processing multiplier type.

A method computing inverse of an element over GF(3^m) which corresponds to a polynomial over GF(3^m) with order less than equal to m-1.

Here, the computation is based on multiplication, square and cube method derived from the mathematics properties over finite fields.

Future research was required to advanced inverse element generator, then we are able to constructing the divider over finite fields. For the future, we prospect constructing four basic arithmetic operation unit systems(AOUS) over finite fields.

Then we obtain the arithmetic system that process the The other future research is construct the logical unit that process the shift, rotate and complement and so on If above future research is establish, we implement the arithmetic and logical unit(ALU) of high efficiency computer ALU architecture based on embedded system.

References

[1] D.L.Dietmeyer, *Logic Design of Digital Systems*, Allyn and Bacon, 1979.

[2] K. Hwang, *Comptuer Arithmetic principles, architecture, and design*, John Wiley & Sons,1979

[2] M.D.Ercegovac and T.Lang, *Digital Systems and* Hardware/ Firmware Algorithms, Wiley, 1985.

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

[4] D.Green, *Modern Logic Design*, Electronic Systems Engineering Series, 1986.

[5] G.Drolet,"A New Representation of Elements of Finite Fields GF(2^m) Yielding Small Complexity Arithmetic Circuits," IEEE Trans. Comput., vol. 47, no.9, pp.938-946, Sep. 1988.

[7] H.Wu and M.A.Hassn,"Low Complexity Bit-Parallel Multi-pliers for a Class of Finite Fields," IEEE Trans. Comput., vol.47, no.8, pp.883-887, Aug. 1988.

[8] C.Ling and J.Lung,"Systolic array implementation of multipliers for finite fields GF(2^m)", IEEE Trans. Cir. & Sys., vol.38,no.7,pp.796-800,Jul.1991.

[9] S.T.J.Fenn, M.Benaissa and D.Taylor, "GF(2^m) multiplication and Division over dual basis", IEEE Trans. Comput., vol.45,no.3,pp.319-327,Mar.1996.

[10] K.Z.Pekmastzi,"multiplxer-based array multipliers", IEEE Trans. Comput.,vol.48.no.1,pp.15-23,Jan.1999.

[11] G.Drolet,"A new representation of elements of finite fields GF(2^m) yields small complexity arithmetic circuits," IEEE Trans. Comput.,vol.47.no.9,pp.938-946,Sep.1998.

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