Polynomial Basis Multiplier over GF(2^m) with Confusion and Error Detection Capabilities

(具混淆及錯誤偵測能力之有限場多項式基底乘法器設計)

Chia-Jen Wu(吳嘉仁)[1], Chi-Ting Ma(馬季廷)[2], , Che Wun Chiou(邱綺文)[3]

[1]Institute of Computer, Communication, and System Engineering
Ching Yun University, Chung-Li 320, Taiwan
Email: m9652004@cyu.edu.tw

[2]Department of Computer Science and Information Engineering
Ching Yun University, Chung-Li 320, Taiwan
Email: m9713007@cyu.edu.tw

[3]Department of Computer Science and Information Engineering
Ching Yun University, Chung-Li 320, Taiwan
Email: cwchiou@cyu.edu.tw

Abstract

The communication system, coding theory, and modern cryptosystems employ the Galois field multiplier widely. Especially, Galois field multiplier is the most important part for computing elliptic curve cryptosystem. Recently, the new developed fault based cryptanalysis would attack both symmetrical and asymmetrical cryptosystems effectively. Therefore, this paper proposes two methods, termed RESO and parity prediction methods, for fighting against such new cryptanalysis. The concurrent error detection polynomial basis GF(2^m) multiplier has advantage of low hardware cost. When the multiplier with error detection capability detects the fault, a random number would be added to its output for confusing hackers. Thus, the hacker will spend more huge time to decrypt the message and then the cryptosystems will become safe.

Keywords: Galois field multiplier, error detection, cryptography.

摘要

近年來通訊系統(Communication systems)、編碼理論(Coding theory)與密碼系統(Cryptosystems)皆廣泛地使用有限場(Galois field)乘法器，而在目前新穎的密碼系統中，橢圓曲線密碼系統(Elliptic Curve Cryptosystem)大量地使用有限場乘法器做為核心運算。針對抵抗植入錯誤式密碼破解法(Fault based cryptanalysis)在密碼系統乘法器中所造成的破壞，本文針對有限場多項式基底之心臟型乘法器提出錯誤偵測架構電路，使具有較節省電路成本之錯誤偵測能力和較快速的運算速度，並且為了提高破解密碼系統所需耗費的時間，當偵測到錯誤發生時加上一組亂數，使密碼系統具有混淆能力，使其有更高的安全性。

關鍵詞：有限場、錯誤偵測、多項式基底乘法器、密碼學。

一、簡介

在資訊發達的現今社會中，電腦與我們的生活息息相關，在周遭環境中皆可輕易看見電腦系統，尤其是大人或小孩都會接觸到的電子交易系統，只要坐在電腦前將我們具有 ATM 功能的銀行提款卡，透過 ATM 讀卡機即可在網路裡完成線上付款動作，但在做這動作同時我們的隱私可能已經遭受到駭客的威脅，而大多數的讀卡機廠商都宣稱其產品具有加密功能，為了保護我們的隱私，一個完善的密碼系統(Cryptosystem)此時就顯得相當重要。

常見的密碼系統可分為秘密金鑰密碼系統(Private Key Cryptosystem)與公開金鑰密碼系統(Public Key Cryptosystem)兩種，一般在做加密動作時通常會依據情況所需，利用秘密金鑰密碼系統進行資料的加密，與公開金鑰密碼系統來傳遞加解密所使用的金鑰以達到最佳效率。秘密金鑰密碼系統即對稱式加密系統，較著名的代表為 DES (Data Encryption Standard)與 AES(Advanced Encryption Standard)，此種密碼系統使用同一組金鑰來做加密與解密，這種方法只使用一組金鑰所以運算速度快，但也因為加密與解密是使用同一組金鑰，使得傳遞金鑰成為此種密碼系統的重要課題。現在常見的公開金鑰密碼系統(又稱非對稱式加密碼系統)有 RSA、橢圓曲線密碼系統(Elliptic Curve Cryptosystem, ECC)等加密系統，公開金鑰密碼系統使用公鑰及私鑰兩把金鑰來達到加密及解密，公鑰負責做加密動作，私鑰負責做解密動作，在公開金鑰密碼系統的機制裡，不會產生秘密金鑰密碼系統所面臨傳遞金鑰的問題。

在對稱式密碼系統跟非對稱式密碼系統中，後者具有比較高的安全性，在使用相同的加密位元數時，非對稱式密碼系統可以達到得到比較好的加密效果，能夠有力的防止密碼系統遭受破解，也就是非對稱式密碼系統會有比較高安全性的表現。

近年來有限場(Galois field,或 Finite field)乘法器的乘法運算、除法運算與反元素運算， 在編碼理論、加密系統、數位訊號處理、虛擬亂數產生器中都有頻繁地出現，於 AES、RSA 與 ECC 等密碼系統中也扮演相當重要的角色。因為在加密系統中的乘法器是最重要也最複雜的運算核心，且是最耗費時間的階段。在實現有限場乘法器中常用的基底有三種，分別為多項式基底

(Polynomial Basis, PB) [2,4,10,11,13,15,17,19,22,27,32,34] 、雙重基底(Dual Basis, DB) [7,12,14,16,20,24,25,29,30,31]、正規基底(Normal Basis, NB) [1,3,5,6,8,9,18,28]。

每一種基底代表著不同的特性及優點。多項式基底表示法的優點在於硬體架構的低複雜度設計、規則化、簡單性、和模組化，因此多項式基底乘法器適合用在 VLSI 乘法器設計。正規基底乘法的優點，是有限場中元素的平方運算可以利用循環的二進位位移達成，因此在執行平方運算、反元素運算和指數運算上有著非常高效率。而雙重基底因使用兩種基底來做轉換，因此其電路成本相對較低，在本文中所採用的多項式基底乘法器依據其特性，設計出適用於 FPGA 上實現的乘法器架構。

在文獻探討中，植入錯誤式攻擊法(Fault Based Cryptanalysis)在破解秘密金鑰與公開金鑰密碼系統的技術上，利用非經由加密演算法破解而以外在因素的方式進行攻擊，利用不同的電力頻率或電壓導致加密晶片產生錯誤的運算結果，且已經被證明是相當有效的技術，藉由所得到的錯誤之運算結果推測出加密時所使用的金鑰，進而達到破解的目的，Kelsey 等人已經證明使用差分攻擊法只需要 50 至 200 個密文就能破解對稱式加密系統的 DES(Data Encryption Standard)。為抵抗這種新型的植入錯誤式攻擊法，因此讓企圖破解密碼系統的駭客得不到其想要的輸出，是目前重要的課題之一。

目前已經有許多學者提出防止植入錯誤式攻擊法的方法，主要可分為重覆運算法[26,33]和同位元預測法[23,32]，在 Reyhani-Masoleh 和 Hasan [10]針對有限場多項式位元並列和串列乘法器，利用同位元預測法提出非線上即時錯誤偵測方法，這個方法只能做到非線上即時錯誤偵測方法，本文針點此一缺點進行改良，乘法器架構改採用運算速度較快速的心臟收縮型乘法器配合同位元預測法的演算法，達到降低電路成本特性的錯誤偵測乘法器，並且當偵測到錯誤時，會將輸出加上亂數，提高密碼系統之安全性。

二、有限場之多項式基底乘法器

多項式在數學上，是以項次(Terms)結構總合表示，每一項包含一變數或多變數上升冪次與係數相乘表示之。若以多項式基底運算，係數是模以一個質數，稱之為有限場多項式基底(Polynomial basis)，以 $GF(2^m)$ 表示。以下例子為一變數多項式：

$$A = a_{m-1}x^{m-1} + \cdots + a_2x^2 + a_1x^1 + a_0x^0 = \sum_{i=0}^{m-1} a_i x^i \qquad a_i, x^i \in \{0,1\}$$

使用有限場做模數運算運算的過程在下列方程式中表示：

$$\begin{aligned} &C(x) \\ &= A(x)B(x) \bmod P(x) \\ &= \left(a_0x^0 + a_1x^1 + a_2x^2 + ... + a_{m-1}x^{m-1}\right)B(x) \bmod P(x) \\ &= \begin{pmatrix} a_0x^0B(x) \bmod P(x) + a_1x^1B(x) \bmod P(x) + a_2x^2B(x) \bmod P(x) + ... \\ + a_{m-1}x^{m-1}B(x) \bmod P(x) \end{pmatrix} \\ &= c_0x^0 + c_1x^1 + c_2x^2 + ... + c_{m-1}x^{m-1}, \quad c_i \in \{0,1\}, 0 \le i \le m\text{-}1. \end{aligned} \tag{1}$$

$$B(x) = b_0 + b_1x + b_2x^2 + ... + b_{m-2}x^{m-2} + b_{m-1}x^{m-1}$$

$$\begin{aligned} &xB(x) \\ &= b_0x + b_1x^2 + b_2x^3 + ... + b_{m-2}x^{m-1} + b_{m-1}x^m \\ &= \begin{pmatrix} b_0x + b_1x^2 + b_2x^3 + ... + b_{m-2}x^{m-1} + \\ b_{m-1}\left(p_0 + p_1x + p_2x^2 + ... + p_{m-1}x^{m-1}\right) \end{pmatrix} \\ &= \begin{pmatrix} b_{m-1}p_0 + \left(b_0 + b_{m-1}p_1\right)x + \left(b_1 + b_{m-1}p_2\right)x^2 + ... + \\ \left(b_{m-2} + b_{m-1}p_{m-1}\right)x^{m-1} \end{pmatrix} \end{aligned} \tag{2}$$

模數運算實例部份當 $C(x)=A(x)\times B(x) \bmod P(x)$，其中假設 m=4 的情況下，$A(x)=0011$，$B(x)=0010$，$P(x)=10011$ 根據公式(1)與(2)可以求得 $C(x)=0110$。

因此可以發現在使用硬體設計乘法器中，多項式基底因具有規則性、簡單性、模組化和低複雜度，相當適合在 VLSI 的設計，圖 1 即為利用公式(1)與(2)所實現出，由 $m\times m$ 個 U 細胞電路所組成的有限場多項式基底心臟型乘法器。

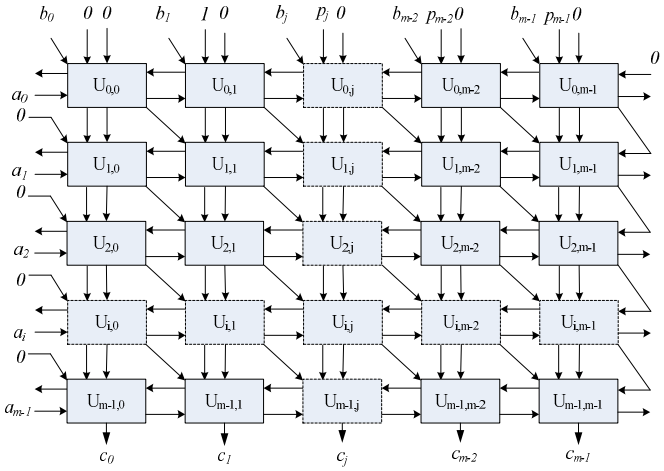

圖 1 有限場多項式基底之心臟型乘法器

其中每個 U 細胞電路的內部電路如圖 2 所示。

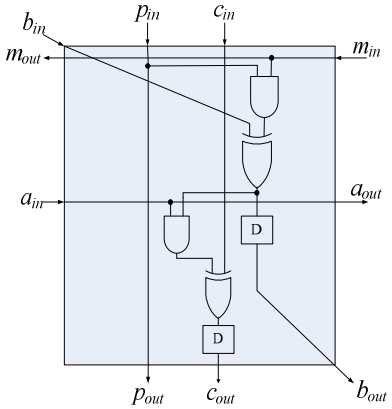

圖 2 U 細胞

三、錯誤偵測能力設計

在錯誤偵測階段所使用的是同位元預測方法(Parity prediction)，同位元預測法，是一個已經有明確概念的方法，能夠預測出運算結果的相同位元以用來和實際運算出來的結果做比對，由於要預測出運算結果必須使用另一個演算法，若是在設計時，將兩種演算法分離將會造成電路成本提高，所以最好的同位元預測法應該是依據原始演算法加以改良，使得只需增加少許的電路，就能將其合併成一個具有乘法功能且有同位元預測法的演算法。

文獻探討中，同位元預測法常設計於一般平行架構與串列式架構的多項式基底之有限場乘法器之中，在 Bayat Sarmadi 和 Hasan [27]中所討論的錯誤偵測架構，即為採用此種方法。主要的設計理念是將乘法器的運算過程中為成多個部份，針對多個部份各別做錯誤偵測，在乘法器運算結束之後將所得到的所有預測位元經過 XOR Tree，若結果為 0 即代表所得到運算結果並未遭受植入錯誤；而相反地，得到結果為 1 將得知此運算結果是遭受植入錯誤，是一錯誤的運算結果。

(一) 錯誤偵測架構

本節將對論文中所使用的同位元預測法做一說明，選用此方法的原因是參考了文獻中 Lee et al. [12]在 2005 年提出的論文，利用雙重基底(Dual basis)之有限場乘法器的特性做出具有預測同位元的功能並達到偵錯能力和 Bayat Sarmadi 和 Hasan [27]在 2007 所提出的平行計算有限場乘法器同位元預測法，並將這兩篇論文的想法做一結合應用在心臟型多項式基底乘法器。

有限場的乘法運算依據前面章節所介紹的有限場乘法器之公式 1 與公式 2 可以達成，為了實現同位元預測法，必須設計另一個運算法來追蹤原本的乘法運算，由於乘法器是使用心臟型具有規律性的硬體架構，因此可以將同位元預測法設計成每計算出一列結果做比對一次，以達到快速偵測錯誤之目的。

同位元預測法中所使用的各個乘法運算元，其同位元表示如下：

$$A_p = (a_0 + a_1 + a_2 + ... + a_{m-1}),$$

$$B_p = (b_0 + b_1 + b_2 + ... + b_{m-1}),$$

$$C_p = (c_{p,0} + c_{p,1} + c_{p,2} + ... + c_{p,m-1}), \text{和}$$

$$P_p = p_0 + p_1 + p_2 + ... + p_{m-1} + p_m = p_0 + p_1 + p_2 + ... + p_{m-1}$$

並將 B_p^i 設成 B^i 的同位元之係數，即 $B_p^i = \sum_{j=0}^{m-1} b_j^i$ ，推導過程如下：

$$B_p^0 = B_p$$

$$\begin{aligned} & B_p^{i+1} \\ &= (b_0^{i+1} + b_1^{i+1} + ... + b_{m-1}^{i+1}) \\ &= (b_0^i + b_1^i + ... + b_{m-1}^i) + b_{m-1}^i (p_1 + p_2 + ... + p_{m-1}) \\ &= B_p^i + b_{m-1}^i P_p \end{aligned} \quad (3)$$

依據公式 3，即可求得 C_p 表示如下：

$$\begin{aligned} & C_p \\ &= c_0 + c_1 + ... c_{m-1} \\ &= \sum_{i=0}^{m-1} a_i b_0^i + \sum_{i=0}^{m-1} a_i b_1^i + ... + \sum_{i=0}^{m-1} a_i b_{m-1}^i \\ &= \sum_{i=-0}^{m-1} a_i (b_0^i + b_1^i + ... + b_{m-1}^i) \\ &= \sum_{i=0}^{m-1} a_i B_p^i \end{aligned} \quad (4)$$

心臟型乘法器運算過程中配合公式3與公式4，即能在心臟型乘法器每一列運算結束後立即求出與該列運算結果相對應的預測位元，如果心臟型乘法器最終計算出的結果 C 值與同位元預測法求得的 C_p 值，經過 XOR Tree 產生的位元做比對，若兩者的數值不一致時，即代表乘法器遭受破壞。

圖 3 為此一方法的電路架構圖，圖中右半部為心臟型乘法器為了清楚表示此方法，因此將電路由每一列的方塊來表示，將乘法器運算的結果和同位元預測方法的結果，在運算完成後經由 XOR Tree 的電路求得該方法所欲比對的數值，由於其錯誤比對是於運算結束後才比對。

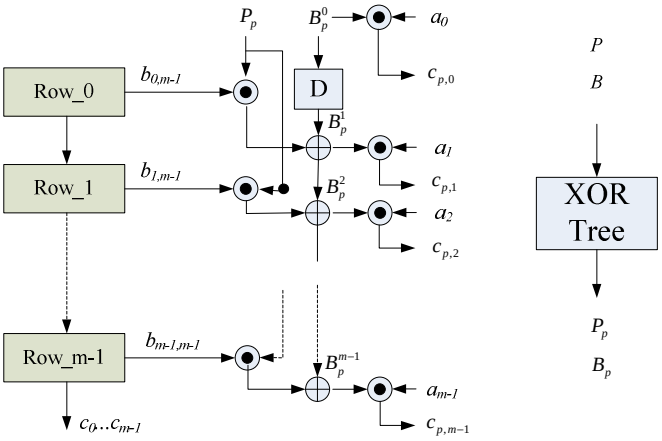

圖 3 同位元錯誤偵測法之有限場乘法器

(二) 同位元預測法乘法器比較

在文獻探討中將圖 3 與 Bayat Sarmadi 和 Hasan [27]在 2007 年所提出的方法做一比較，在相同的同位元錯誤偵測法之平行處理乘法器架構下，Bayat Sarmadi 和 Hasan [27]與圖 3 所使用的概念相當雷同，皆為利用有限場乘法器將每一列 c 值求出後，送進 XOR Tree 再與先前提出之方法得到預測的同位元做比對，然而圖 3 在偵錯方法所使用的電路成本較其來得低，而且在時間的延遲上也比較少，其中 Bayat Sarmadi 和 Hasan [27]中提到其預測同位元的所使用的位元數，設計時分別有 4、8、12、16、20 等五種情形，而這邊所選擇用來做比較的位元長度為 8 位元。

表 1 為圖 3 與 Bayat Sarmadi 和 Hasan [27]中的錯誤偵測方法所需電路與耗費的時間做一比較，當 m 值等於 163 的情況下，本文所提出的方法可以節省的時間約為 55%，電路成本節省約

50%。

表 1 同位元預測法乘法器比較

	Bayat Sarmadi 和 Hasan [27]	圖 3
Time Complexity		
Total delay (unit: ns)	124m	56m
Space Complexity		
2-input AND gates	m^2+8m	$2m^2 + 3m$
2-input XOR gates	$9m^2+7m$	$2m^2+85m$
Total transistor counts	$60m^2+31m$	$24m^2+528m$

四、混淆能力架構

在植入錯誤式攻擊法中，目的就是要使結果變成駭客所期望的數值，進而求得正確的密文，這邊針對此一議題提出提昇防止被破解的構想，即利用具錯誤偵測能力的乘法器，當其發現有錯誤出現時，就把該次運算的結果與一個亂數產生器所製造出來的亂數做相加，以增加破解密文所需要的時間。

(一) 亂數產生器

利用 Forney [21]於 1970 年提出使用移位暫存器之亂數產生器，加至前一章節中具偵測錯誤能力之乘法器的輸出端，在亂數產生器中依據參考的文獻中之數據，配合乘法器的位元長度來做調整，以得到較適合的亂數，依據位元長 m 可決定所產生出的亂數多寡，m 位元可求得 2^m-1 筆亂數，圖 4 為一長度 4 位元亂數產生器範例，產生出循環的亂數共 15 筆數值。

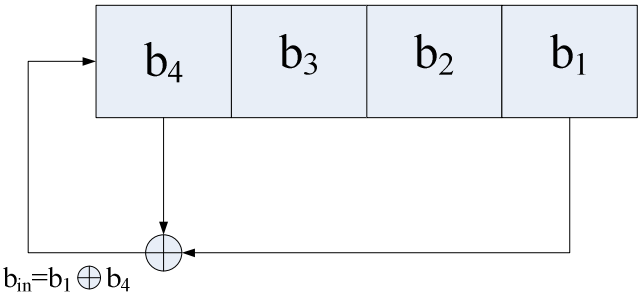

圖 4 位元移位暫存器

(二) 具亂數產生器之即時錯誤偵測能力乘法器

本節中將即時錯誤偵測電路配合移位暫存器，當電路偵測到錯誤時將與該時脈下所生成的亂數做互斥或運算再把結果送至輸出端，由於設計時移位暫存器是獨立運作，會隨著時脈不停的變換所產生的數值，可增加破解密所需的時間。圖 5 為該方法示意圖。

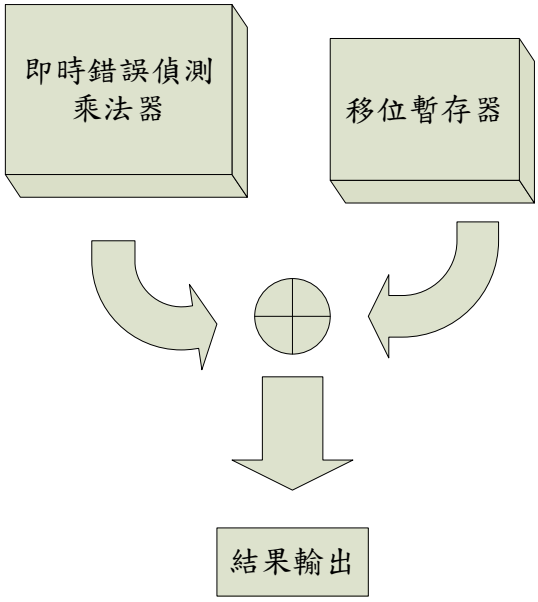

圖 5 具亂數產生器之即時錯誤偵測能力乘法器示意圖

(三) 告知機制

在有限場乘法器的錯誤偵測機制發現錯誤時，即會將移位暫存器所產生的亂數與已遭受破壞的運算結果做相加，此設計方法目的在於使欲破解密碼系統之駭客，其必須花費更多的時間來進行破解的動作，但是此方法在混淆駭客的同時已經將輸出結果改變，將導致最終的輸出結果為錯誤的資料，故需要在此方法設計一個告知的機制，確保密碼系統在計算時能夠得知密碼系統是否遭受破壞。

由於有限場乘法器中的錯誤偵測架構中，在發現錯誤發生時即會產生一條訊號線以便後續判斷是否需要將亂數加至運算結果中，於是在有限場乘法器的最終輸出時就利用這一條訊號線，在 m 位元的運算結果中加上額外的一位元來做為辨識用位元，預設情況中將辨識用的位元置於運算結果之最低位元處，為避免過於明顯故將辨識位元與運算結果中的任一位元做置換，密碼系統中的其它協定只需知道辨識用的位元與哪一位置的位元做置換，即可得到辨識用位元並得知此一運算結果是否遭受破壞並已經加上一組移位暫器所產生的亂數，圖 6 為其示意圖，此處使用與前一章節相同的電路成本比較方法，依據組成及閘(AND Gate)、或閘(OR Gate)、互斥或閘(XOR Gate)的電晶體數量來計算所需的電路成本，而在 m 值等於 163 的情況中加上亂數的架構，需額外增加具錯誤偵測乘法器電路的 0.26%。

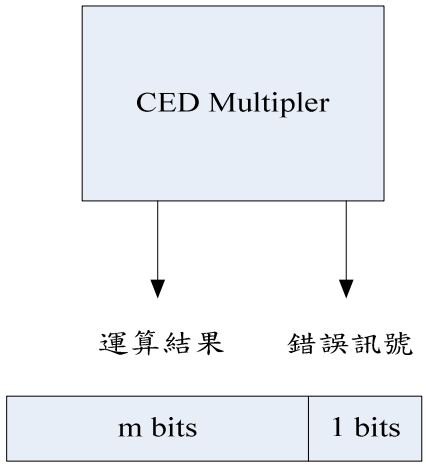

圖 6 初始狀態

五、結論與未來發展

為確保密碼系統的安全性，論文在第五章中提出一種混淆的方法，在具錯誤偵測之心臟型乘法器中增加混淆能力，即當發現錯誤存在時將運算結果加上移位暫存器所產生的亂數產生器，因此可以增加駭客破解密碼系統所需耗費的時間，達到提昇密碼系統之安全性之目的。

同位元預測法在加上移位暫存器之元件比較，由所使用的移位暫存器只需依照乘法器設計時的 m 值，額外增加 m 位元的暫存器，因此在 m 位元下的錯誤偵測心臟型乘法器，僅增加不到 1%的電路成本即可達到減低密碼系統遭受破解的可能性。

由於論文所提出的概念是密碼系統中的運算核心-乘法器加上錯誤偵測的機制，並於此架構在偵測到錯誤發生時將其錯誤的運算結果加上一筆亂數，使得想要破解的駭客無法得到其所預期的數值，所以論文中所提出的設計架構可能遭受到駭客不斷地企圖破解而無法正常進行加密計算，導致加密晶片整體無法正常動作，因此此一架構較適合在此一乘法器遭受

植入錯誤後，並無法還原至正常運算狀態中實現。

六、參考文獻

[1] A. Reyhani Masoleh and M. A. Hasan, “A New Construction of Massey-Omura Parallel Multiplier over GF(2^m)”, IEEE Transactions Computers, vol. 51, no. 5, pp. 511-520, May. 2002.

[2] A. Reyhani Masoleh and M. A. Hasan, “Error Detection in Polynomial Basis Multipliers Over Binary Extension Fields”, Proc. of Cryptographic Hardware and Embedded Systems-CHES, vol. LNCS 2523, vol. 2523, pp. 515-528, 2002.

[3] A. Reyhani Masoleh and M. A. Hasan, “Fast Normal Basis Multiplication Using General Purpose Processors”, IEEE Transactions Computers, vol. 52, no. 11, pp. 1379-1390, November. 2003.

[4] A. Reyhani Masoleh and M. A. Hasan, “Fault Detection Architectures for Field Multiplication Using Polynomial Bases”, IEEE Transactions Computers, vol. 55, no. 9, pp. 1089-1103, September 2006.

[5] A. Reyhani Masoleh, “Efficient Algorithms and Architectures for Field Multiplication Using Gaussian Normal Bases”, IEEE Transactions Computers, vol. 55, no. 1, pp. 34-47, January 2006.

[6] B. Sunar and C. K. Koc, “An efficient optimal normal basis type II multiplier” , IEEE Trans. Computers, Vol. 50, No.1, pp.83-87, January 2001.

[7] C. C. Wang, “An algorithm to design finite field multipliers using a self-dual normal basis”, IEEE Trans. Computers, Vol.38, No.10, pp.1547-1459, October 1989.

[8] C. C. Wang, T. K. Truong, H. M. Shao, “VLSI architectures for computing multiplications and inverses in GF(2^m)” , IEEE Transactions Computers, Vol,C-34, No.8, pp.709-717, August 1985.

[9] C. W. Chiou, C. Y. Lee, J. M. Lin, “Concurrent error detection and correction in dual basis multiplier over GF(2^m)”, IET Circuits, Devices & Systems, vol. 3, no. 3, pp. 22-40, February. 2009.

[10] C. W. Chiou, L. C. Lin, F. H. Chou, “Low complexity finite field multiplier using irreducible trinomials”, Electronics Letters, vol. 39, no. 24, pp. 1709-1711, November 2003.

[11] C. Y. Lee, “Low-complexity bit-parallel systolic multipliers over GF(2^m)”, Integration, the VLSI Journal, vol. 41, no. 1, pp. 106-112, January 2008.

[12] C. Y. Lee, C. W. Chiou and J. M. Lin, “ Concurrent Error Detection in a Bit-Parallel Systolic Multiplier for Dual Basis of GF(2^m)”, Journal of Electronic Testing: Theory and Application , vol. 21, No. 5, pp. 539-549, October 2005.

[13] C. Y. Lee, C. W. Chiou and J. M. Lin, “Concurrent Error Detection in a Polynomial Basis Multiplier over GF(2^m)”, Journal of Electronic Testing: Theory and Applications, vol. 22, no. 2, pp. 143-150,June. 2006.

[14] C.Y. Lee and C.W. Chiou, “Efficient design of low-complexity bit-parallel systolic Hankel multipliers to implement multiplication in normal and dual bases of GF(2^m) ”, IEICE Transactions on Fundamentals of Electronics, Communications and Computer Science, Vol.E88-A, No.11, pp.3169-3179, Nov. 2005.

[15] C.Y. Lee, E.H. Lu and J.Y. Lee, “ Bit-Parallel Systolic Multipliers for GF(2^m) Fields Defined by All-One and Equally Spaced Polynomials”, IEEE Trans. Computers, vol. 50, no. 5, pp. 385-393, May. 2001.

[16] E. R. Berlekamp, “Bit-serial Reed-Solomon encoder”, IEEE Transactions Information Theory, Vol. IT-28, no. 6, pp.869-874,

November 1982.

[17] G. Seroussi, “Table of Low-Weight Binary Irreducible Polynomials”, Visual Computing Dept., Hewlett Packard Laboratories, August 1998.

[18] H. Fan and Y. Dai, “Key function of normal basis multipliers in GF(2^m)” , Electronics Letters, Vol,38, No. 23, pp.1431-1432, 7th November 2002.

[19] H. Wu, “Bit-Parallel Polynomial Basis Multiplier for New Classes of Finite Fields”, IEEE Transactions Computers, vol. 57, No. 8, pp. 1023-1031, August 2008.

[20] H. Wu, M. A. Hasan and I. F. Blake, “New low-complexity bit-parallel finite field multipliers using weakly dual bases”, IEEE Transactions on Computers, vol. 47, No. 11, pp. 1223-1234, November 1998.

[21] J. G. Forney, “Coding and its application in space communications”, IEEE Spectrum, vol. 7, pp. 47-58, June 1970.

[22] J. L. Massey and J. K. Omura, “Computational method and apparatus for finite field arithmetic”, U.S. Patent no. 4587627, May. 1986.

[23] J.-S. Horng, C.Y. Lee, I.-C. Jou, “Fault-based triangular basis multiplication over GF(2^m) using bit-level parity prediction scheme”, Tencon 2007.Proceedings of the IEEE Region 10 Conference, pp. 1-4, October 2007.

[24] M. Morii, M. Kasahara, and D. L. Whiting, “Efficient bit-serial multiplication and the discrete-time Wiener-Hopf equation over finite fields”, IEEE Transactions Information Theory, Vol.35, No.6, pp.1177-1183, November 1989.

[25] M. Wang and I. F. Blake, “Bit serial multiplication in finite fields”, SIAM J. Disc. Math., Vol.3, No.1, pp.140-148, February 1990.

[26] R. Hughey, “Concurrent Error Detection on Programmable Systolic Arrays”, IEEE Transactions Computers, vol. 42, no. 6, pp752-756, June 1993.

[27] S. Bayat Sarmadi and M. A. Hasan, “On concurrent detection of errors in polynomial basis multiplication”, IEEE Transactions on Very Large Scale Integration Systems, vol. 15, no. 4, pp. 413-426, April, 2007.

[28] S. Oh, C. H. Kim, J. Lim, “Efficient normal basis multipliers in composite fields” , IEEE Trans. Computers, Vol.49, No.10, pp.1133-1138, October 2000.

[29] S. T. J. Fenn, D. Taylor and M. Benaissa, “A dual basis bit-serial systolic multiplier for GF(2^m)”, Integration, the VLSI Journal, vol. 18, no. 2, pp. 139, June. 1995.

[30] S. T. J. Fenn, M. Benaissa and D. Taylor, “ GF(2^m) Multiplication and Division Over the Dual Basis”, IEEE Transactions on Computer, vol. 45, no. 3, pp. 319-327, March. 1996.

[31] S. T. J. Fenn, M. Benaissa, and D. Taylor, “Dual basis systolic multipliers for GF(2^m)”, IEE Proc. Comput. Digit. Tech., vol.144, no.1, pp.43-46, January. 1997.

[32] S. T. J. Fenn, M. G. Parker, M. Benaissa, “Bit-serial multiplication in GF(2^m) using irreducible all-one polynomials”, IEE Proceedings: Computers and Digital Techniques, vol. 144, no. 6, pp. 391-393, November. 1998.

[33] S. Y. Kuo, S. C. Liang, “Concurrent Error Detection and Correction in Real-Time Systolic Sorting Arrays”, IEEE Transactions Computers, Vol. 41, No. 12, December 1992.

[34] S.T.J. Fenn, M. Gossel, M. Benaissa, “On-Line Error Detection for Bit-Serial Multipliers in GF(2^m)”, Journal of Electronic Testing: Theory and Application, vol. 13, no. 1, pp. 29-40, August 1998.