A Fast Elliptic Curve Cryptosystem

G.B. Agnew R.C. Mullin S.A. Vanstone
University of Waterloo

Introduction

In the fall of 1986, the authors developed a prototype of a fast GF(2^{593}) multiplier/exponentiator. This device was based on the discovery of optimal normal basis structures in fields of characteristic two [1][2]. In the ensuing years, much effort has gone into fabricating this structure as a VLSI device. In the early months of 1989, the first functioning VLSI devices were fabricated [3]. These devices, which implement a cryptographic system based on discrete exponentiation [4][5], have throughput rates of up to 300 Kbps.. Many cryptographic applications based on discrete exponentiation have been implemented or proposed [6][7], and several new applications of the normal basis multiplier have recently been considered.

In 1985, Miller [8], presented a method of implementing a cryptographic system based on elliptic curves. The advantage of such cryptosystems is that, unlike RSA or discrete exponentiation, no sub-exponential method is known for attacking elliptic curves [9]. Thus, smaller block sizes could be used to implement a computationally secure system.

The problem with elliptic curve implementations is the complexity (and thus computation time) of calculating points on the curve. In this paper, we will examine an implementation (not optimized) of an elliptic curve system using the fast normal basis multiplier structure.

Elliptic Curve Calculations

Koblitz [10], lists several forms of elliptic curves which form groups. For curves of characteristic two, the curve

$$y^2 + y = x^3$$

was chosen. In this system, the calculation of a new point KP from a point P on the curve and value K ($0 < K \leq 2^n - 1$) can be realized as

$$KP = k_0P + k_1 2P + k_2 4P + \ldots + k_{n-1}2^{n-1}P$$

or

$$KP = (\ldots((k_0P + k_1 2P) + k_2 4P) + \ldots + k_{n-1}2^{n-1}P) .$$

J.J. Quisquater and J. Vandewalle (Eds.): Advances in Cryptology - EUROCRYPT '89, LNCS 434, pp. 706-708, 1990.

This requires the calculation of the intermediate values of 2^iP where,

$$2P = (x^4, y^4 + 1)$$

In integer fields, this involves successive squaring and would be costly in terms of computation time. Using the normal basis representation though, squaring becomes a cyclic shift of the representation of x and y [1] (a shift of two positions for x^4 and y^4 respectively).

The addition of two points

$$(x_3, y_3) = (x_1, y_1) + (x_2, y_2)$$

is calculated as

$$x_3 = A^2 + x_1 + x_2$$

$$y_3 = 1 + y_1 + A\,(x_1 + x_3)$$

where

$$A = (x_1 + x_2)^{-1}\,(y_1 + y_2)\,.$$

In the system of characteristic 2, the addition of co-ordinates (e.g., $x_i + x_j$) is performed by XOR'ing corresponding terms. Division (inversion) and multiplication of coordinates is performed using successive exponentiation using the normal basis multiplier.

A Hybrid Implementation

Our earlier investigations of applications of the normal basis multiplier led us to develop a PC-based board to implement various forms of public key cryptographic functions based on discrete exponentiation. This board incorporates a Motorola M68008 processor to control the operation of the encryption processor (this board was originally designed to implement a virus protection scheme and to provide secure file management). To implement the elliptic curve system, the board was reprogrammed to allow the M68008 to perform the manipulation and storage of point co-ordinates while the normal basis multiplier was used to calculate inverses and the products of point co-ordinates. In our observations, the hybrid system can calculate about 9 elliptic curve points per second for K with Hamming weight of 30 (this is equivalent to about 5 Kbps.).

Conclusions

While no attempt was made to optimize the performance of the hybrid system, we believe the system is the fastest implementation of an elliptic curve cryptosystem to date. Our studies have also shown that a modified version of the current architecture could produce an elliptic curve processor many times faster than this hybrid implementation.

References

1. Agnew, G., R. Mullin, S. Vanstone, "Fast Exponentiation in $GF(2^n)$", Proceeding of Eurocrypt'88, Springer-Verlag Lecture notes in Computer Science, May 1988.
2. "Computational Method and apparatus for finite field multiplication", U.S. patent #4,745,568.
3. "CA34C168 Data encryption processor data sheet", CALMOS Semiconductor, Kanata, Ontario, Canada, Oct. 1988.
4. Diffie, W., M. Hellman, "New directions in cryptography", IEEE Trans. on Info. Theory, Vol. IT-22, pp.472-492, 1976.
5. ElGamal, T., "A public key cryptosystem and a signature scheme based on discrete logarithms", IEEE Trans. on Infor. Theory, Vol. IT-31, pp. 469-472, 1985.
6. Agnew, G., R. Mullin, S. Vanstone, "An interactive data exchange protocol based on discrete exponentiation", Proceeding of Eurocrypt'88, Springer-Verlag Lecture notes in Computer Science, May 1988.
7. Boyd, N., "CRYPNET: A public key cryptographic network testbed: Design document", CCNG Technical Report E-172, University of Waterloo, Nov. 1988.
8. Miller, V., "Use of elliptic curves in cryptography", Lecture Notes in Computer Science #218 - Proceedings of CRYPTO '85, Springer Verlag, pp. 417-426, Aug. 1985.
9. Coppersmith, D., "Cryptography", IBM Journal of Res. and Development, Vol. 31, No. 2, Mar. 1987, pp. 244-248.
10. Koblitz, N., "A course in number theory and cryptography", Springer Verlag, 1987.