FPGA Implementation of Point Multiplication on Koblitz Curves Using VHDL

Prof. Snehprabha Lad
[Guide]
E&TC Department
TIT Bhopal (MP) India

Prof. Vikas Gupta
[H.O.D]
E&TC Department
TIT Bhopal (MP) India.

Ms. Anupa S. Kalambe
ME-Digital Communication
E&TC Department
TIT Bhopal (MP) India.

Abstract

The point multiplication on Koblitz curves using multiple base expansions of the form $k=Sum pm tau^a(tau-1)^b$ & $k = Sum pm tau^a(tau-1)^b (tau^2 –tau – 1)^c.$. In this paper the number of terms in the second type is sublinear in the bit length of k, which lead to first provably sublinear point multiplication algorithm on Koblitz curves. Also present details of an innovative FPGA implementation of algorithm and performance data demonstrating the efficiency of method. Elliptic curve scalar multiplication is the essential operation in elliptic graph cryptography. But these paper presents to accelerate scalar multiplications on Koblitz curve. In this paper we are implementing to use low power technique on FPGA implementation.

Keywords *– Cryptoprocessor, elliptic curve cryptography (ECC), FPGA, Koblitz curve, Point multiplication.*

1. Introduction

These algorithms need to operate efficiently using minimal available resources. Binary Koblitz curves are special class of generic curves that point multiplication can be efficiently computed using their special properties. These curves employ Frobenius map (instead of doubling) and point addition operation for computing point multiplication. The Koblitz curves, or anomalous binary curves, are Ea: y2 + xy = x3 + ax2 + 1; defined over IF2. The major advantage of Koblitz curves is that the Frobenius automorphism of IF2 acts on points via τ(x,y)=(x2,y2). It has been claimed that the maximum number of the finite-field multipliers to get the highest parallelization in computing point multiplication on Koblitz curves is three parallel finite-field multipliers. This implementation proved to be competitive towards existing designs in terms of speed, low power but the additional area overhead was significant.

2. Literature Review

Fast and high-performance computation of finite field arithmetic is crucial for elliptic curve cryptography (ECC) over binary extension fields. Lastly worked on highly parallel scheme to speed up the point multiplication for high-speed hardware implementation of ECC cryptoprocessor on Koblitz curves. This slightly modify the addition formulation in order to employ four parallel finite-field multipliers in the data flow also reduces the latency of performing point addition and speeds up the overall point multiplication, which implemented our proposed architecture for point multiplication on an Altera Stratix II field-programmable gate array and obtained the results of timing and area.

3. Point Multiplication On Koblitz Curves Based Algorithms

Some of the existing point multiplications on Koblitz curves based algorithms are discussed in this section.

3.1 Point Multiplication On Koblitz Curves

Algorithm 1 Point multiplication on Koblitz curves using double-and-add-or-subtract algorithm .

Inputs: A point $P = (x, y) \in EK\ (GF(2\text{^}m))$ on curve and integer k, k ={ $\sum_{i=0}^{l-1} \boldsymbol{ki\ \tau\text{^}i}$ for $ki \in \{0,\tau\}1\}$.

Output: $Q = kP$.

```
1: initialize
        a: if $kl-1 = 1$ then $Q \leftarrow (x, y, 1)$
        b: if $kl-1 = -1$ then $Q \leftarrow (x, x + y, 1)$
2: for $i$ from $l - 2$ downto 0 do
        $Q \leftarrow \varphi(Q) = (X2, Y2, Z2)$
        if $ki$ _= 0 then
        $Q \leftarrow Q + kiP = (X, Y,Z)\ \tau\ \}\ (x, y)$
        end if
    end for
3: return $Q \leftarrow (X/Z, Y/Z2)$
```

The algorithm for computing point multiplication, i.e., $Q = kP$, on Koblitz curves ,where the scalar k is presented in τNAF.

3.2 High-Speed Parallelization Of Point Addition

Parallelization for hardware implementation of point addition on Koblitz curves has been considered recently employing different number of field multipliers in [4], [8], and [16]. In [4], it is shown that employing two finite-field multipliers reduces the number of multiplications.

Proposition: The point addition formulation and data dependence in computing by following

Z :
$A = Y1 + y2Z2,$
$B= X1 + x2Z1,$
$C = x2Z21 + X1Z1,$
$Z3 = C2.$

4. Proposed Work

The proposed cryptoprocessor architecture for point multiplication is given below.

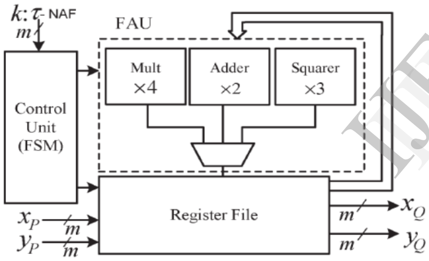

Fig. 2. Architecture of point multiplication cryptoprocessor.

4.1 Fau

The FAU performs three basic arithmetic operations employing four digit-level GNB multipliers, two $GF(2m)$ adders, and two squarers. Multiplication in $GF(2m)$ plays the main role in determining the efficiency of the point multiplication.

4.2 Control Unit And Register File

The control unit is designed with a finite-state machine (FSM) to perform the point multiplication with other units.

4.3 Fpga Implementation

The results of the area and maximum clock frequencies of the implementations after the place and route, which increasing the digit size results in the reduction of the latency of the point multiplication.

5. Result

It shows the result of two multiple binary number with its output in binary number and it shows its signal in wave form. In the previous paper for doing this multiplication technique needs 12 clock cycles but in this project we required only 1 clock cycle. Again this project required low power technique to run. and when we obtained the result in the binary form with its two binary input, to understand its value it can be converted in to decimal form with the help of IEEE 754 format as it is easy to understand. This Paper shows the FPGA implementation also.

6. Conclusion

It is easy to multiply two binary numbers But it is hard to multiply huge binary number. With the help of Koblitz curves it is easy to multiply huge binary numbers. With this paper we are showing point multiplication on Koblitz curve and FPGA implementation with low power technique.

7. Future Scope

From Improved low power technique with this Koblitz curves we also implement Point multiplication using high speed Hardware implementation.

8. Acknowledgement

I take this opportunity to express my sincere gratitude to Prof.Vikas Gupta (H.O.D) TIT College of engineering and technology, who have contributed in making the work towards great success. I express gratitude to my guide Prof. Snehprabha Lad & M.E. coordinators, whose constant help and encouragement supports me to complete my work.

9. References

[1] N. Koblitz,“Elliptic curve cryptosystems,”*Math. Comput.*, vol.48, pp.203–209, 1987.
[2] K. Järvinen,“Optimized FPGA-based elliptic curve cryptography processor for high-speed applications,” *Integr., VLSI J.*, vol. 44, no. 4, pp. 270– 279, Sep. 2011
[3] J. Adikari, V. S. Dimitrov, and R. J. Cintra, “A new algorithm for double scalar multiplication over Koblitz curves,” in *Proc. IEEE ISCAS*, 2011, pp. 709–712.
[4] B. B. Brumley and K. U. Järvinen, “Conversion algorithms and implementations for Koblitz curve

cryptography," *IEEE Trans. Comput.*, vol. 59, no. 1, pp. 81–92, Jan. 2010.

[5] J. Adikari, V. Dimitrov, and K. Jarvinen, "A fast hardware architecture for integer to τ-NAF conversion for Koblitz curves," *IEEE Trans. Comput.*, vol. 61, no. 5, pp. 732–737, May 2012
.