



Article Title: An Efficient of High-Speed and High precision Overlap-Free Karatsuba-Based Finite-Field Multiplier for FPGA Implementation

An Efficient of High-Speed and High precision Overlap-Free Karatsuba-Based Finite-Field Multiplier for FPGA Implementation

Arun Kumar R.E¹, S.Amutha²

¹PG student, Department of ECE, Marthandam College of Engineering and Technology

²Assistant Professor, Marthandam College of Engineering and Technology

¹arunmacet08@gmail.com

ABSTRACT

Finite field arithmetic is becoming increasingly a very prominent solution for calculations in many applications. In this paper, complexity and delay of six different multipliers (Mastrovito multiplier, Paar-Roelse multiplier, Massey- Omura multiplier, Hasan-Masoleh multiplier, Berlekamp multiplier and Karatsuba multiplier) are compared. Also this paper presents a modified multiplier based on Karatsuba multiplication algorithm. To optimize the Karatsuba multiplication algorithm, the product terms are splited into two alternative forms and all the terms are expressed in the repeated fashion. This Modified architecture saves the 14.9% computation time and it consumes 45.5% less slices than existing Karatsuba multiplier. The proposed architecture has been simulated and synthesized by Xilinx ISE design suite for Spartan & Vertex device family. The new architecture is Simple & easy. The proposed Modified Karatsuba Multiplier (MKM) is also applied to compute the circular convolution for DSP application. In Spartan3E FPGA device family, computation of 8-bit circular convolution using Modified Karatsuba Algorithm (MKA) is 26.5% faster than Karatsuba Algorithm (KA). It also consumes 61.7% less slices than existing KA based Convolution.

Keywords: Karatsuba Algorithm; Finite fields; FPGA; VLSI; polynomial multiplication; Circular Convolution;

1 Introduction

CRYPTOGRAPHY is used to provide confidentiality, data security, and authentication in many applications, such as communication devices [1], [2], autonomous vehicles [3], Internet of Things (IoT) [4], and healthcare [5]. There are generally two types of cryptography techniques: sym metric-key cryptography [6] and public-key cryptography [7]. The public-key cryptography allows all communication parties to safely communicate without any previously shared secret information among them. For this purpose, it is required to provide key establishment and digital signature for secure communications. Examples of public-key cryptography include Diffie–Hellman [8], RSA [9], ElGamal [10], and elliptic curve cryptography (ECC) [11]. Among these algorithms, the ECC has become the most dominant



Article Title: An Efficient of High-Speed and High precision Overlap-Free Karatsuba-Based Finite-Field Multiplier for FPGA Implementation

public key cryptosystem, mainly because of its relatively short key size with respect to the security strength and implementation effectiveness

2 Related Work

Compared to software implementations, hardware cryptography results in a higher speed and a lower cost while satisfying the efficiency and low-power requirements of electronic devices. There exists a number of VLSI [11], [14] and field programmable gate array (FPGA) [15], [16] implementation of ECC cryptography algorithms. The computation of the elliptic curve points can be broken down into finite-field operations in which the finite-field multiplier becomes the major component in the hardware implementation of ECC, where multiplier's size and delay dominate the overall area and throughput of the system. Accordingly, numerous research specifically targeted designing fast and efficient finite-field multipliers for cryptosystems

B. Niu, Q. Li et al., proposes Dummy-Location Selection (DLS) algorithm to achieve k-anonymity for users in LBS. Different from existing approaches, the DLS algorithm carefully selects dummy locations considering that side information may be exploited by adversaries. We first choose these dummy locations based on the entropy metric, and then propose an enhanced-DLS algorithm, to make sure that the selected dummy locations are spread as far as possible

3 Architectural Design

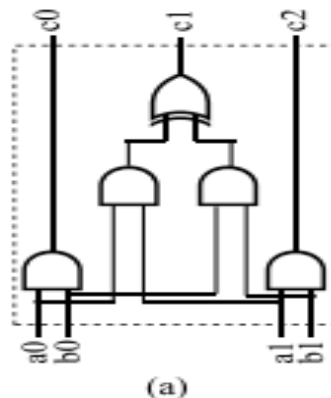


Figure 1: DFG for hardware implementation of (a)

Overlap-free Karatsuba algorithm (OKA) [33]–[35] in one of the algorithms proposed by researchers to reduce the combinational delay of the KA. This algorithm aims to remove an XOR gate in the critical path of KA. This change results in a considerable reduction in the theoretical delay in comparison to the original KA method. However, the CA still outperforms OKA in terms of delay due to its non-iterative structure



Article Title: An Efficient of High-Speed and High precision Overlap-Free Karatsuba-Based Finite-Field Multiplier for FPGA Implementation

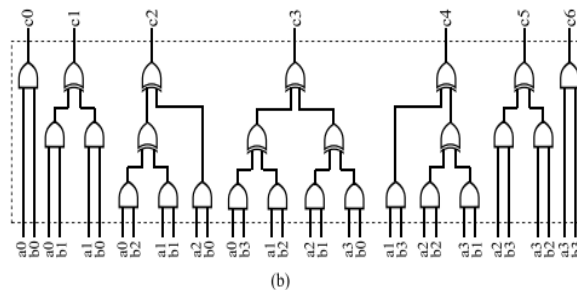


Figure 2: 4-bit conventional binary polynomial multiplier.

4 Background

Polynomial multiplication, followed by a modular reduction, is one of the widely used operations in $GF(2^n)$ whose efficiency greatly impacts the overall performance and cost of the system [30]. The multiplier's efficiency is evaluated according to its required space and the total delay of the multiplication based on the theoretical boundaries of these quantities using ideal two-input AND and XOR gates [30], [33], [36]. These works do not consider the limitations of the actual hardware; for example, the issue of limited gate fan-out in these devices is ignored in most instances. The multiplier's delay and space are simply calculated by the ideal configuration of the system and linear addition of the standard gate delays and area requirements. However, in actual implementation, there are other considerations, such as when the hardware restrictions dictate inserting of buffers, which in theoretical works are not of concern [37]. This article incorporates both theoretical and implementation results.

Karatsuba Multiplier

(KM) In this section, we introduce the fundamental Karatsuba algorithm which can successfully be applied to polynomial multiplication. The Karatsuba Algorithm was introduced by Karatsuba in 1962. The fundamental Karatsuba multiplication for polynomial in $GF(2^m)$ is a recursive divide-and-conquer technique. It is considered as one of the fastest way to multiply long numbers. For polynomial multiplication with original Karatsuba method both operands have to be divided into two equal parts. Then each sub operands is divided again into two parts. The process will continue until this become single. Figure1 shows the block diagram of Karatsuba multiplier for degree-3 polynomials. Then we get the followings by splitting the polynomials using KM: If $A(x)$ and $B(x)$ are field polynomials with degrees 3 over a field $GF(2^4)$. With the auxiliary variables

$$D0 = a0b0, D1 = a1b1$$

$$D2 = a2b2, D3 = a3b3$$

$$D0, 1 = (a0 + a1)(b0 + b1)$$

$$D0, 2 = (a0 + a2)(b0 + b2)$$



Article Title: An Efficient of High-Speed and High precision Overlap-Free Karatsuba-Based Finite-Field Multiplier for FPGA Implementation

$$D1, 3 = (a1 + a3) (b1 + b3)$$

$$D3, 2 = (a3 + a2) (b3 + b2)$$

$$D0, 1, 2, 3 = (a0 + a1 + a2 + a3) (b0 + b1 + b2 + b3)$$

Field multiplication can be performed into two steps. Firstly, we perform an ordinary polynomial multiplication of two field elements. Secondly, a reduction operation with an irreducible polynomial is need to be performed in order to obtain the $(m - 1)$ degree polynomial. It is noticed that once the irreducible polynomial $p(x) = x^4 + x + 1$ has been selected, the reduction step can be accomplished by using XOR gates only [9]. From the irreducible polynomial $p(x)$ we can replace $x^4 = x + 1$, $x^5 = x^2 + x$ and $x^6 = x^3 + x^2$ to obtain $C'(x)$ as follows:

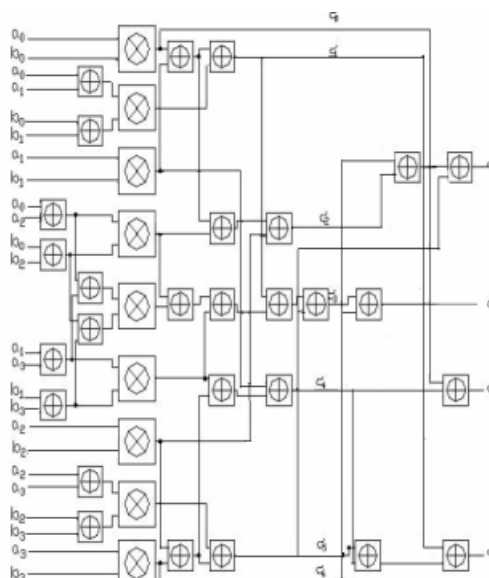
$$C'(x) = A(x) B(x) \text{ mod } p(x)$$

$$C'(x) = (D0, 1, 2, 3 - D1, 3 - D2, 0 - D3,$$

$$2 - D0, 1 + D0 + D1 + D2) x^3 + (D0, 2 + D3, 2 + D1 - D0) x^2 + (D0, 1 + D1, 3 + D3, 2 - D0) x + (D1, 3 - D1 - D3 + D2 + D0) \quad (5)$$

Modified Karatsuba Multiplier (MKM)

In this section our Modified Karatsuba Algorithm (MKA) has been discussed. In MKA all techniques are same as fundamental basic Karatsuba multiplier except the splitting techniques. To optimize the Karatsuba Multiplication Algorithm, the product terms are splitted into two alternative forms. This reduction technique requires small area and less delay than others existing multiplication algorithms. The results are compared by using Xilinx based synthesis tools on different FPGA device family like Spartan & Vertex. Our synthesis results are better than existing basic Karatsuba algorithm which is shown in the following section. Assume $A(x)$ and $B(x)$ are two field polynomials with degree 3 in $GF(2^4)$. $A(x) = a_3 x^3 + a_2 x^2 + a_1 x + a_0$ $B(x) = b_3 x^3 + b_2 x^2 + b_1 x + b_0$



Since the conventional multiplication method is not the most efficient method, other methods, such as KA [27] and its variations, have been developed. We will briefly review the original KA in the following. Obtain the validation process using their LBS and querying

[illegible]

In this paper, modified Karatsuba multipliers for degree 3 and 7 polynomials has been implemented on FPGA platform. The device utilization and combinational path delay of MKM have been compared with standard 8×8 KM. It has been observed that the proposed multiplier has better timing performance than standard KM. In Spartan3E FPGA device, proposed multiplier needs 14.9% lesser delay than KM, and it also consumes 45.5% lesser slices compared to KM. The new architecture is very simple and easy. This feature is advantageous to have a suitable trade-offs between area and speed for implementing circular convolution algorithm in VLSI. In FPGA device family, computation of 8- bit circular convolution using MKA is 26.5% faster than KA. It also consumes 61.7% less slices than existing KA based convolution. MKM may also be used to design cryptosystems. Proposed multiplier is faster and hardware efficient compared to existing Karatsuba multiplier.



Article Title: An Efficient of High-Speed and High precision Overlap-Free Karatsuba-Based Finite-Field Multiplier for FGPA Implementation

References

1. R. Abu-Salma; M. A. Sasse; J. Bonneau; A. Danilova; A. Naiakshina; M. Smith, Year:2017, "Obstacles to the adoption of secure communication tools," in Proc. IEEE Symp. Secur. Privacy (SP), May 2017, pp. 137–153.
2. B. Vembu; A. Navale; S. Sadhasivan, Year: 2017, "Creating secure communication channels between processing elements," U.S. Patent 9 589 159.
3. J. Yoo; J. H. Yi, Year: 2018, "Code-based authentication scheme for lightweight integrity checking of smart vehicles," IEEE Access, Vol: 6, pp. 46731–46741.
4. K. Shahbazi; S. B. Ko, Year: 2021, "Area-efficient nano-AES implementation for Internet-of-Things devices," IEEE Trans. Very Large Scale Integer. (VLSI) Syst., Vol: 29, no. 1, pp. 136–146.
5. P. Aparna; P. V. V. Kishore, Year: 2019, "Biometric-based efficient medical image watermarking in E-healthcare application," IET Image Process., Vol: 13, no. 3, pp. 421–428.
6. S. Raza; L. Seitz; D. Sitenkov; G. Selander, Year: 2016, "S3K: Scalable security with symmetric keys—DTLS key establishment for the Internet of Things," IEEE Trans. Autom. Sci. Eng., Vol: 13, no. 3, pp. 1270–1280.
7. X. Zhang; J. Long; Z. Wang; H. Cheng, Year: 2016, "Lossless and reversible data hiding in encrypted images with public-key cryptography," IEEE Trans. Circuits Syst. Video Technol., Vol: 26, no. 9, pp. 1622–1631.
8. A. Faz-Hernandez; F. Rodriguez-Henriquez; E. Ochoa-Jimenez; J. Lopez, Year:2018, "A faster software implementation of the supersingular isogeny Diffie-Hellman key exchange protocol," IEEE Trans. Comput., Vol: 67, no. 11, pp. 1622–1636.
9. X. Zhou; X. Tang, Year: 2011, "Research and implementation of RSA algorithm for encryption and decryption," in Proc. 6th Int. Forum Strategic Technol., Vol: 2, Aug. 2011, pp. 1118–1121.
10. F.-Y. Rao, Year: 2019, "On the security of a variant of ElGamal encryption scheme," IEEE Trans. Dependable Secure Comput., Vol: 16, no. 4, pp. 725–728.
11. Z. U. A. Khan; M. Benaissa, Year: 2017, "High-speed and low-latency ECC processor implementation over GF (2m) on FPGA," IEEE Trans. Very Large Scale Integration. (VLSI) Syst., Vol: 25, no. 1, pp. 165–176.
12. F. Mallouli; A. Hellal; N. S. Saeed; F. A. Alzahrani, Year: 2019, "A survey on cryptography: Comparative study between RSA vs ECC algorithms, and RSA vs El-Gamal algorithms," in Proc. 6th IEEE Int. Conf. Cyber Secur. Cloud Comput. (CSCloud)/ 5th IEEE Int. Conf. Edge Comput. Scalable Cloud (EdgeCom), pp. 173–176