Industrial Engineering Journal
ISSN: 0970-2555
Volume : 53, Issue 4, April : 2024

A NOVEL HARDWARE ARCHITECTURE FOR EFFICIENT FIELD-PROGRAMMABLE GATE ARRAY (FPGA) IMPLEMENTATION

I. RAJYALAKSHMI M.Tech[1]

V. HARIPRIYA[2], R. SAHITHI REDDY[3],K. BHANU PRIYA[4], S. RAMYA[5]

[1]Assistant Professor, Department of ECE, ELURU COLLEGE OF ENGINEERING AND TECHNOLOGY, A.P., India.

2, 3,4, 5Student,Department of ECE, ELURU COLLEGE OF ENGINEERING AND TECHNOLOGY, A.P., India.

ABSTRACT: this research suggests an optimal polynomial multiplication method. This idea suggests brand-new optimisations for the polynomial multiplier, the component of lattice-based cryptography that requires the greatest computing, with a focus on high-speed hardware.Polynomial multiplication is regarded as the most time- and space-consuming operation in ECC systems. This article suggests a novel hardware design for effective Finitefield multiplier implementation for ECC using field-programmable gate arrays (FPGAs).The complexity (space) analysis and efficient FPGA implementation of bit parallel Karatsuba Multiplier over GF (2m) is presented. This is especially interesting for high performance systems because of its carry free property. Using Karatsuba multiplier we can improve the performance of the process. This paper proposes an optimised polynomial multiplication for compact digital architectures. This concept proposes novel optimisations for the most computationally intensive part of lattice-based cryptography constructions, i.e., the polynomial multiplier, targeting the high speed hardware platform.In ECC systems, polynomial multiplication is considered to be the most slow and area consuming operation. This article proposes a novel hardware architecture for efficient field-programmable gate array (FPGA) implementation of Finitefield multipliers for ECC.

INTRODUCTION

The two fundamental operations in the finite field GF are addition and multiplication. The bit parallel Karatsuba Multiplier over GF (2m)

complexity (space) study and effective FPGA implementation are provided. Due to its carry-free characteristic, this is extremely intriguing for high performance systems. By using a Karatsuba multiplier, we can increase the process' efficiency. For tiny digital systems, Due to its simplicity, itspolynomial version is widely adopted to design VLSI parallelmultipliers in GF(2n)- based cryptosystems [13]-[34]. Two parameters are often used tomeasure the performance of a GF(2n) parallel multiplier, namely, the space and time complexities. The space complexity is represented in terms of the total number of 2-input XORand AND gates used. The corresponding time complexity is given in terms of the maximum delay faced by a signal due to these XOR and AND gates. Symbols “TA” and “TX” are often used to represent the delays of one 2-input AND gate and one 2-input XOR gate, respectively.The existing bit parallel GF(2n) multipliers may be simply classified into the following three categories according to the asymptotic space complexity of the multiplication algorithm: quadratic, sub quadratic and hybrid multipliers. A number of quadratic multipliers have been proposed in the literature in which different basis representations of GF(2n) elements are used,e.g., polynomial, shifted polynomial, normal, dual, weakly dual, and triangular bases. Their time complexities are lower than those of sub quadratic multipliers.The main advantage of sub quadratic multipliers is that their low asymptotic space complexities make it possible to implement VLSI multipliers for large values of n. But when the size of operands is small, e.g., 32-bit, the space complexity may not remain as the critical factor considered by a cryptographic processor designer. Instead, the computational speed becomes the key factor. Based on this consideration, the hybrid approach is often used to designpractical multipliers [6] [18]

[21] [23] [32]. These multipliers first perform a few KOA iterations to reduce the whole space complexities, and then a quadratic multiplication algorithmon small input operands to achieve relatively high speed performance. By selecting different stop conditions for the KOA iterations, the hybrid approach can provide a trade-off between the time and space complexities. For the purpose of comparison, reference

[21] implemented four parallel GF(2233) multipliers on Xilinx FPGAs, namely classical, hybrid Karatsuba, Massey- Omura, and Sunar-Koc¸, and

analysed their time and space complexities in detail.

PROPOSEDSYSTEM

Existing system Drawbacks

- High complexity
- High propagation path delay
- Less efficient with huge reliability

Proposed System: The OKA is a speed-optimized version of the original Karatsuba. In this method, to improve the longest path delay, inputs are split into odd and even orders instead of the high and low parts. Once more, it is assumed that *A(x)* and *B(x)* are two polynomials in *GF(2n)* and$n = 2m$.

- low complexity
- low propagation path delay
- efficient with huge reliability

LITERATURE SURVEY

C. P. Rentería-Mejía et.al [1] proposed a Hardware Design of FFT Polynomial Multipliers. In this paper, they present the design of two FFT polynomial multipliers using parallel and sequential architectures. Parallel and sequential polynomial multipliers were optimized for throughput and area resources, respectively. The designs are described in genericstructural VHDL, synthesized on the Stratix EP4SGX230KF40C2 using Quartus II V. 13, andverified using Signal Tap. The hardware synthesis and performance results show that the designed multipliers present a good area throughput trade-off and they are suitable for high- performance scientific computing applications. Their work presents the design of two polynomial multipliers based on FFT. In this case, they used FFT based on complex fixed-point computations and R22SDF architecture.Parallel and sequential polynomial multipliers were optimized for throughput and area resources, respectively. Also, the designed multipliers were parameterized for polynomials of8, 16, 32, 64, 128, 256 and 512 coefficients. The synthesis results show that the designed polynomial multipliers use few area resources and have a good throughput. The parallel polynomial multiplier uses 53 % more resources and its throughput is in average 1.81 times bigger than the sequential polynomial multiplier. Also, the designed multipliers carry out the polynomial multiplication in less time than the corresponding software simulation in Maple 15,which was performed on an Intel Core i7- 3770 CPU @ 3.40 G taking into account the synthesisand hardware

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 53, Issue 4, April : 2024

verification results, they conclude that the designed multipliers are suitable for high-performance scientific computing applications [1].

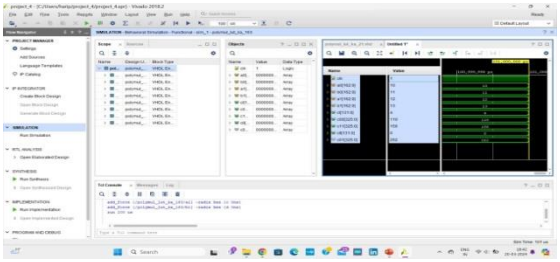
RELATED WORK

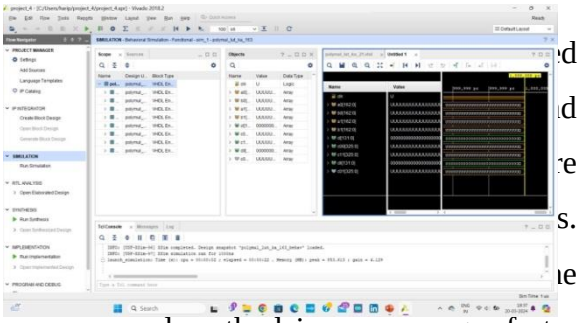
MOSFET consists of a MOS capacitor with two p-n junctions placed closed to the channel region and this region is controlled by gate voltage. To make both the p-n junction reverse biased, substrate potential is kept lower than the other three terminals potential. If the gate voltage will be increased beyond the threshold voltage (VGS>VTO), inversion layer will be established on the surface and n – type channel will be formed between the source and drain. This n – type channel will carry the drain current according to the VDS value. For different value of VDS, MOSFET can be operated in different regions as explained below.

Linear Region

At VDS = 0, thermal equilibrium exists in the inverted channel region and drain currentID = 0. Now if small drain voltage, VDS > 0 is applied, a drain current proportional to the VDS will start to flow from source to drain through the channel. The channel gives a continuous path for the flow of current from source to drain. This mode of operation is called **linear region**.

SAMPLE RESULTS





CONCLUSION

proposed method is, on average, faster than Karatsuba and faster than the OKA. Comparing with

state-of-the-art works also indicated that the design has higher speed and lower ADP, which demonstrates the efficiency of the design.

REFERENCES

- C. P. Rentería-Mejía, A. López- Parrado, J. VelascoMedina, “Hardware Design of FFT Polynomial Multipliers”, 978-1- 4799-2507-0/14/$31.00 ©2014 IEEE.
- Lo Sing Cheng, Ali Miri, Tet Hin Yeap, “EFFICIENT FPGA IMPLEMENTATION OF FFT

BASED MULTIPLIERS", 0-7803- 8886-0/05/$20.00 ©2005 IEEE.

- E. Theochari, K. Tatas, D. J. Soudris, K. Masselos, K. Potamianos, "A REUSABLE IP FFTCORE FOR DSP APPLICATIONS", 0-7803-8251- X/04/$17.00 © 2004 IEEE.
- A.Ronisha Prakash, S.Kirubaveni, "Performance Evaluation of FFT Processor Using Conventional and Vedic Algorithm", 2013 IEEE International Conference on Emerging Trendsin Computing, Communication and Nanotechnology (ICECCN 2013), 978-1-4673-5036-5/13/$31.00 © 2013 IEEE 89. [5] Ali Chamas Al Ghouwayel, Amin Haj-Ali and Zouhair El- Bazzal, "Towards a Triple Mode Common Operator FFT for SoftWare Radio Systems", 19th International Conference onTelecommunications (ICT 2012),

  978-1- 4673-0747-5/12/$31.00
- ©2012 IEEE.
- P. D. Chidgupkar and M. T. Karad, "The Implementation of Vedic Algorithms in Digital Signal Processing", Global J. of Engg. Edu., volume 8, Issue no. 2, Year 2004.
- M. Moreno and Y. Xie, "FFT- based dense polynomial arithmetic on multicores", 23rd Int.conf. on high perf. Comp. systems and applic., June Year 2009, p.p. 378- 399.
- Sushma R. Huddar and Sudhir Rao, Kalpana M., Surabhi Mohan, "Novel High Speed Vedic Mathematics Multiplier using Compressors", 978-1-4673-5090-7/13/$31.00 ©2013 IEEE.
- Laxman P. Thakre, Suresh Balpande, Umesh Akare, Sudhir Lande, "Performance Evaluation and Synthesis of Multiplier used in FFT operation using Conventional and Vedic algorithms", 978-0-7695-4246-1/10 $26.00 © 2010 IEEE.