IJARSCT

ISSN (Online) 2581-9429

International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)

Impact Factor: 6.252

Volume 2, Issue 2, July 2022

Design of Low Power Versatile Bit-Serial Multiplier in Finite Field GF (2^m)

Niveditha SR[1], Brunda BS[2], Rohith K[3], Nithyashree R[4], Mrs. Asha R[5]

Students, Department of Electronics and Communication Engineering[1,2,3,4]

Faculty, Department of Electronics and Communication Engineering[5]

Vidya Vikas Institute of Engineering & Technology, Mysuru, Karnataka, India

Abstract: *Finite field arithmetic is the most important component in applications like cryptography, computer algebra and error correcting codes. Versatility is an important property the hardware industry lacks and trying to establish as much as possible. To survive in this booming technological word, the new designs should be of an adjustable one, which processes the versatile property. In this we have proposed an efficient VLSI design for versatile bit-serial multiplier in finite fields GF (2m). The versatile multiplier designed here modifications done by reducing the unwanted switching activity removed by clock gating scheme. Our design provides a solution to the power reduction. The introduced multiplier operates over a variety of binary fields up to an order of 2m. The value of m can vary up to 264 bits. Multiplier is designed using Verilog HDL.*

Keywords: Low power, Versatile, Bit-serial multiplier, Finite field, Galois field

I. INTRODUCTION

For the majority of cutting-edge technology in fields like computer algebra, error correction, and cryptography, finitefield arithmetic has emerged as a fundamental building block. The most important finite field operation is finite field multiplication. The point multiplication operation, which is entirely based on the modified finite field multiplication operation, is the lowest rung in the hierarchy of cryptography processing.

Due to their effective hardware implementations, the binary extension fields, abbreviated as GF (2^m), have drawn a lot of attention. A GF (2^m) consists of 2^m distinct elements, each of which is represented by a distinct set of m binary coordinates (0 or 1) with respect to a basis. A collection of m linearly independent field elements is used to buildthe basis. The representations for field elements that are most frequently used are polynomial basis (PB) and normal basis (NB). These representations allow for the straightforward bit-wise addition (Exclusive OR, or XOR) of the appropriate coordinates to combine fields of two elements. On the other hand, field multiplication is more difficult than addition, and the complexity of a field multiplication directly depends on the representation of the underlying fields. Field multiplication also serves as the foundation for more complex operations like field exponentiation,division, and inversion. These operations are frequently utilized in symmetric key cryptography methods and are implemented as iterations of multiplications. The multiplier holds the most notable place in design, and choosing a multiplier is an important responsibility while creating a processor. Therefore, a significant contribution to the evolution of the cryptosystem can be made by changing the fundamental structure of the multiplier design. This characteristic draws a lot of attention to this area, where efficient finite field architecture can be implemented. Finite field GF(2^m) multipliers come in three different varieties. They combine partial bit serial and partial bit parallel features to be bit serial, bit parallel, and a mix of the two. Although the hybrid multiplier has a faster processing speed than the bit serial multiplier, it uses more resources than the bit parallel multiplier. The bit parallel has a complexity of O, while the bit serial has an O(m) complexity(m^2). The irreducible polynomial has a requirement that is directlyproportional to the multiplier. As a result, this can be simplified by using an All-One Polynomial (AOP), a trinomial, or a redundant field representation.

II. LITERATURE SURVEY

Review of the literature is essential to the project's success. It mostly assists in learning in-depth about the fundamental concepts to pay attention to and in gathering data from many angles. We can learn how to priorities the job and finish it as intended by conducting a literature review. We can weigh the benefits and drawbacks of adopting a methodology,

DOI: 10.48175/IJARSCT-5848

which greatly facilitates decision-making and increases efficiency. Finally, a literature review helps us finish the assignment more effectively.

In [1] in order to improve point addition and point doubling for speed, low power, and less- Area applications, elliptic Curve arithmetic structures were presented. According to the comparison, our suggested designs outperform other relevant, well-known works significantly. The development and testing of a fully parameterized processor for VHDL Elliptic Curve point multiplication was the last step. It is designed to aid in the development of cryptographic fundamentals including key agreement methods and digital signatures. The fundamental shortcomings of these multipliers are the need for operand conversions from integer toMontgomery domain representation prior to and following the multiplication, respectively. This, however, is a flaw in Montgomery's approach generally compared to other methods. Additionally, this is no longer significant once a sufficient number of consecutive modular multiplications are required.

In [2] it is suggested to use a multiplication algorithm with interleaved modular reduction that multiplies any two field elements for any field defining an irreducible polynomial over GF (2^m). The suggested approach results in a flexible sequential polynomial basis multiplier architecture over GF(2^m). The proposed multiplier's area and delay complexity are computed, and its performance is evaluated in comparison to that of current sequential polynomial basis multipliers on the market. Comparing sequential multipliers to existing multipliers, they produce better area-delay and power-delay products. The area estimate was presented in a rather complicated manner. Since this involved using the challenging field programmable gate array (FPGA) and application specific integrated circuit (ASIC) implementation of the suggested multiplier.

In [3] over GF, a modified interleaving multiplication technique was suggested that multiplies any two arbitrary field elements for a general irreducible polynomial (2^m). When compared to equivalent multipliers, the multiplier achieves low hardware complexity and comparable latency, critical path, and total delay. There was an ideal cut-set retiming. A systolic multiplier that multiplies finite-field elements over universal irreducible polynomials has been presented as a result of the retimed SFG. The proposed systolic multiplier and other systolic multipliers' hardware complexity and delay are estimated.

In [4] a new bit-level serial multiplication scheme for the elements of GF (2^m), based on the PB representation. The proposed formulation for the bit-level PB multiplication is based on a recursive definition of the field elements. The recursive definition constructs an element bit-by-bit, one bit per a clock cycle, starting from the most significant bit. The polynomial equation are very complicated to solve for hardware components and the power consumption is high when compared to present existing designs.

In [5] an FPGA implementation of a new multiplier for the finite field GF(2^n) using a polynomial basis representation. It includes two parts, classical and Montgomery, each of which is a systolic array. The multiplier is designed for polynomial bases which allow more flexibility in hardware and is dedicated to efficient implementations of elliptic curve cryptography. The area consumption is large and when executed in hardware components the complexity increases.

In [6] a flexible architecture for a polynomial basis multiplier over GF (2^m) is studied. Series of tri-state buffers and a set of control signals are within created to achieve a versatile property together with a lower power dissipation property. The key advantages of the architecture are its compatibility with arbitrary Galois Field size and its hardware easiness which results in lesser area. In this there is an increase in the critical path delay and the power consumption is high when compared to others. The outcome for the literature survey is as follows:

- From referring to different paper published in various journals, it can be concluded that many surveys are conducted on multipliers.
- The implementation of Galois field over different multipliers.

How power can be reduced using gated clock techniques.

II. PROBLEM STATEMENT

Design and verification of low power versatile bit-serial multiplier in finite field GF(2^m) to reduce power, area and time delay**.**

DOI: 10.48175/IJARSCT-5848

III. OBJECTIVES

- To have an insight into commonly used classed of finite field in cryptography (Galois Field).
- Study of different types of algorithms used in multipliers in finite field.
- Implementing and differentiating with different types of multipliers.
- Implementing the extensions of Karatsuba algorithm in finite field.

IV. METHODOLOGY

A finite field, also known as a Galois field, is made up of a limited set of elements and the descriptions of the addition and multiplication operations that can be carried out on pairs of field elements. The distributive law, the existence of an additive identity and a multiplicative identity, the existence of an additive inverse and a multiplicative inverse, and associativity and commutativity of both addition and multiplication are some of the features that these operations must have. The smallest finite field is GF(2), which has only the numbers 0 and 1 as field elements. Since addition and multiplication are carried out modulo 2, the addition and multiplication are equivalent to the logical XOR and AND, respectively.

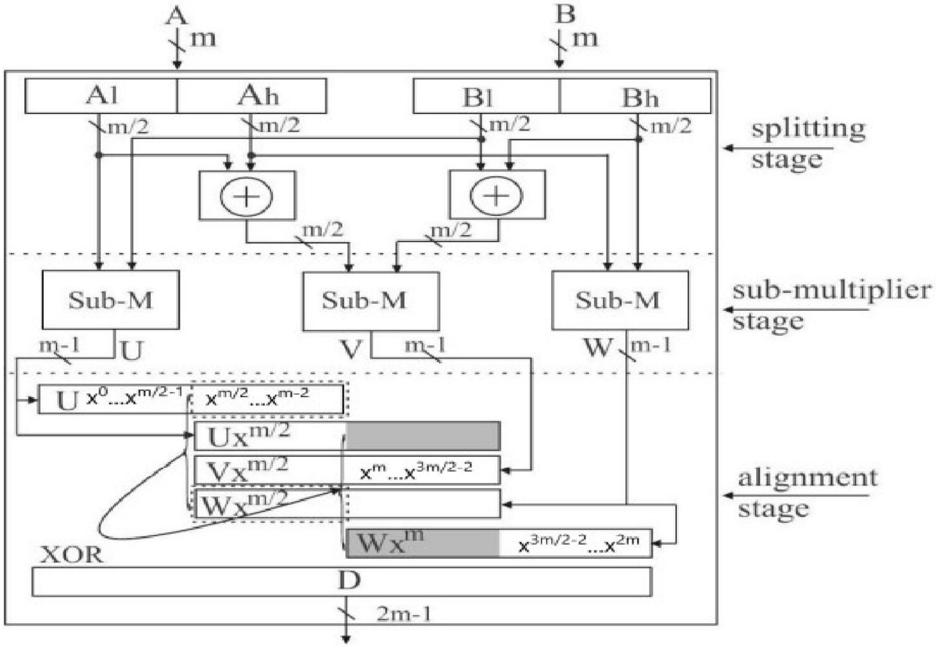

Fig. 1. Multiplier Architecture by applying Overlap-free KA

The Combining the Multiplier architecture with the proposed modified Overlap-free KA algorithm and completing the implementation of the proposed modified Overlap-free KA algorithm for GF (2^n). Make the module when n = 2 as an example to help with comprehension. It has been noted that no overlap will take place at this moment due to the value of n.

Then, as demonstrated in Verilog HDL, the value of n can be expanded from 2 to 4. To complete the module, we employ nested and transferred statements because value 4 is precisely twice the size of 2. In this section, we use the suggested algorithm since for this value, overlap happens during the alignment stage. If m = n is even, Figure.1 depicts the multiplier architecture using the one-step Overlap-free KA algorithm as an example. Three phases make up the multiplier: splitting, sub-multiplier, and alignment, where three sub- multipliers work in parallel. We can effectively specify which component's function in this design. The common sub-expressions are found when calculating D m/2….m-2 and Dm... 3m/2 −2.

$$\begin{cases} D_{\frac{m}{2}\ldots m-2} = [U_{\frac{m}{2}\ldots m-2} + W_{0\ldots\frac{m}{2}-2}] + U_{0\ldots\frac{m}{2}-2} + V_{0\ldots\frac{m}{2}-2} \\ D_{m\ldots\frac{3m}{2}-2} = [U_{\frac{m}{2}\ldots m-2} + W_{0\ldots\frac{m}{2}-2}] + W_{\frac{m}{2}\ldots m-2} + V_{\frac{m}{2}\ldots m-2} \end{cases}$$

DOI: 10.48175/IJARSCT-5848

Using this architecture and proposed modified Overlap-free Karatsuba algorithm, we can implement it and analyses its features.

4.1 Execution of a Multiplication

In order to describe how the offered bit-serial multiplier performs a multiplication, Let's assume that, at the start of the multiplication, the multiplier-polynomial a(t) is placed in register Multiplier and the multiplicand-polynomial b(t) is kept in register Multiplicand, as illustrated in Figure 2.

The multiplier-bits, or coefficients ai, are delivered to the GF(2^m) arithmetic unit, starting with a m1, afterthe register Result has been cleared and the register Multiplier has been changed bit by bit in MSB direction. As was previously discussed in this section, the intermediate result is calculated as r(t). The outcome of the multiplicationis stored in register Result after the last coefficient, a0, has been evaluated. Then it can either be sent to the outside world from register Multiplicand or loaded there to serve as a multiplicand- polynomial for the subsequent multiplication.

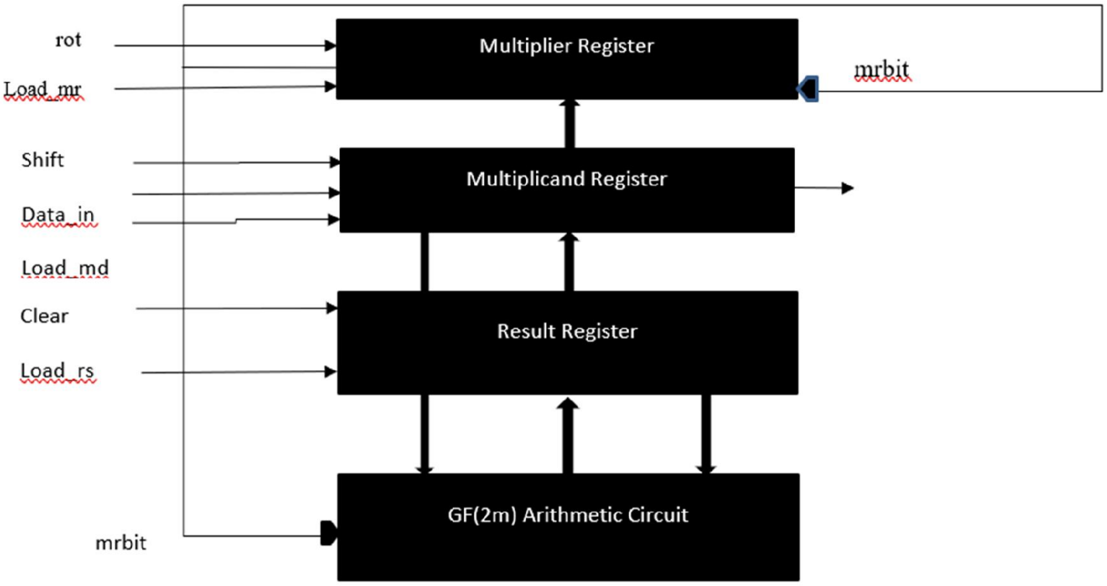

Fig. 2. Block diagram of the proposed GF(2^m) multiplier

4.2 Software Requirement

The simulation is completed using the Xilinx ISE simulator program. For the synthesis and analysis of HDL design, Xilinx created the Xilinx ISE (Integrated Synthesis Environment). The entire design flow is managed by the ISE software.

- Synthesis or compile its design.
- Perform timing analysis.
- Examine RTL diagrams.
- Simulate a design's reaction to different stimuli.
- Configure the target device with the programmer.

4.3 Cadence Tool

Cadence is a platform for electronic design automation that enables many tools and applications to integrate into a single framework, covering all stages of integrated circuit design and verification from a single environment. These instruments are totally versatile and facilitate various construction A series of configuration and technology-related files are used to tailor the cadence environment when a specific technology is used. A collection of software program, including schematic editors, simulators, and layout editors, make up the cadence Development system. This software controls the creation of mixed-mode, digital, and analogue circuits.

DOI: 10.48175/IJARSCT-5848

V. EXPECTED OUTCOME

When compared to existing multipliers, the suggested project should have low power consumption, small area usage,and finite field parameters, and the serial bit-multiplier can be modified depending on the application environment. The ability to use the same multiplier to more applications should become more flexible as a result.

5.1 Comparison Table

Table 1: Comparison result with different multipliers in Finite field

Different GF multiplier (8-bits)	Area (incell)	Power (nW)	Delay (nsec)
Array multiplier	4396	170216.191	28.75
Booth multiplier	13401	12281168.007	16.077
Modified Overlap-free Karatsuba	914	56286.633	12.451

Table 2: Result of device utilization that is area in cells, power in Watts and combinational path delay of different bits of GF (2^m) multipliers.

Multipliers in different bits	Area (in cells)	Power (nW)	Delay (nsec)
2-bit	49	1875.490	7.824
4-bit	237	10609.802	9.077
8-bit	914	56286.633	12.451
16-bit	3171	260598.298	16.294
32-bit	10396	1190587.160	19.438

5.2 Waveforms

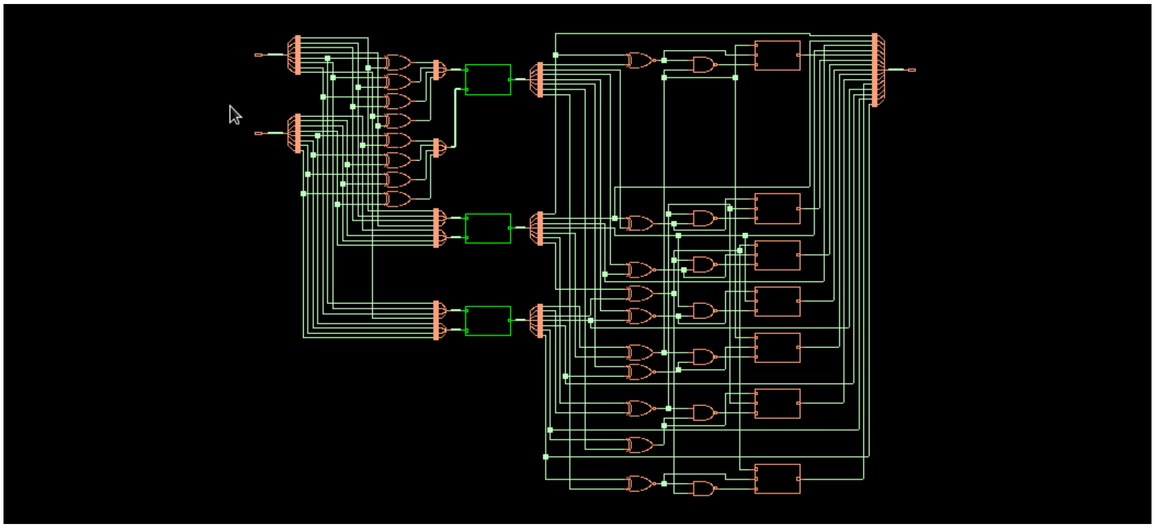

Fig. 3. 8-bit schematic diagram

DOI: 10.48175/IJARSCT-5848

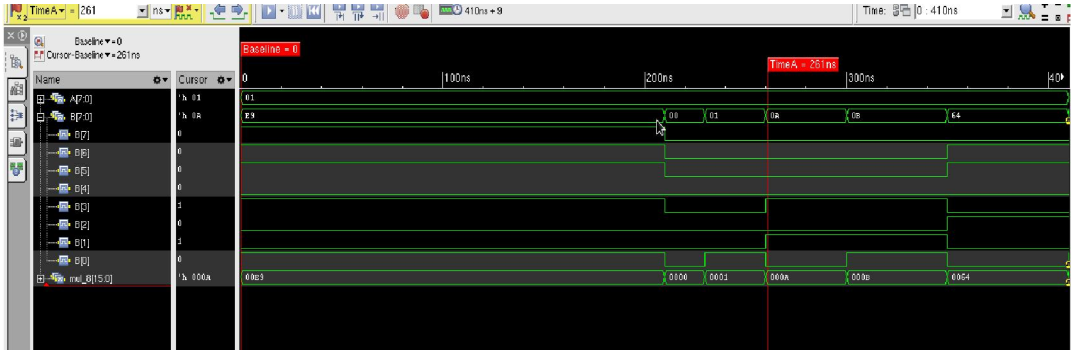

Fig. 4. Simulation result of proposed modified module for GF(2^8)

VI. CONCLUSION

In this study, bit-serial multiplication using a modified Overlap-free Karatsuba algorithm has been studied. and lists the shortcomings of the finite field flexible multiplier designs that have already been done. The problem found in the related existing work was addressed in a changed architecture that was put forth. The current design's clock gating method does not reduce unauthorized switching inside the feedback loop. The alterations made in this work result in a loss in power. An Elliptic curve cryptography processor will be used to analyze the suggested flexible bit serial multiplier's area and performance efficiency. The following succinctly summaries our key contribution:

In comparison to other well-known algorithms already in use, such as the Karatsuba algorithm, the reconstruction Karatsuba method, and the improved reconstruction by Bernstein, the suggested approach outperforms them all in terms of maximum combined path latency, area, and power .For GF(24), GF(28), and GF(232), the suggested modified Overlap-free Karatsuba algorithm multiplication was compared with specifically published multiplications (Karatsuba and Modified Karatsuba multiplications) (216). The comparison's findings support the idea that the proposed modified Overlap-free Karatsuba algorithm multiplication significantly reduces the maximum combinational path time, area, and power.

REFERENCES

[1]. El Hadj Youssef Wajih, Guitouni Zied, Machhout Mohsen & Tourki Rached, "Design and Implementation of Elliptic Curve Point Multiplication Processor over GF (2m)", Electronics and Micro-Electronic Laboratory (LEME), Monastir, Tunisia Monastir, 5000, Tunisia, 2018

[2]. Lakshmi Boppanna, "Design and Implementation of a Sequential Polynomial Basis Multiplier over GF(2m)"- Department of Electronics and Communication Engineering, National Institute of Technology-Warangal Warangal, Telangana, 2017

[3]. Sudha Ellison Mathe, "Low-Power and Low-Hardware Bit-Parallel Polynomial Basis Systolic Multiplier over GF(2m) for Irreducible Polynomials"- ETRI Journal, 2017

[4]. Hayssam El- Razouk and Arash Reyhani-Masoleh, "New Bit-Level Serial GF (2m) Multiplication Using Polynomial Basis", Department of Electrical and Computer Engineering Western University London, Canada, 2015

[5]. Lejla Batina, Nele Mentens, Sıddıka Berna Ors, Bart Preneel Katholieke Universiteit Leuven, "Serial Multiplier Architectures over GF(2n) for Elliptic Curve Cryptosystems", ESAT/SCD-COSIC Kasteelpark Arenberg 10 B-3001 Leuven-Heverlee, Belgium Lejl, 2014

[6]. Riddhish Shukla, Kulin Shah, Raj Chaurasia, Sivanatham S, "Study of Bit-Serial Multiplier in Finite Fields GF (2m)", National Conference on Innovative Trends in Science andEngineering (NC-ITSE'16),2014

[7]. Jeng-Shyang Pan, Reza Azarderakhsh, Mehran Mozaffari Kermani, Chiou-Yng Lee, Wen-Yo Lee, Che Wun Chiou, "Low-Latency Digit-Serial Systolic Double Basis Multiplier over GF(2m) Using Sub quadratic, MAY 2014

[8]. I Grasschadi, "A Low power Bit-Serial multiplier for finite field GF(2m)"- Graz University of Technology Institute for Applied Information Processing and Communications Inffeldgasse 16a, A–8010 Graz, Austria,2010

DOI: 10.48175/IJARSCT-5848

[9]. P. Kitsos,G. Theodoridis, O. Koufopavlou, "An efficient reconfigurable multiplier architecture for Galois field GF(2m)"- VLSI Design Laboratory, Department of Electrical and Computer Engineering, University of Patras, Rio, Patras 26500, Greece,2003