International journal of basic and applied research

www.pragatipublication.com

ISSN 2249-3352 (P) 2278-0505 (E)

Cosmos Impact Factor-5.86

Area And Power Efficient Modified Karatsuba Multiplier for Error-Free Cryptography Algorithms

[1]Alladurgam Gowtham,[2]Chellem Ramu,[3]Dasari Sampath,[4]Deshetti Harathi,[5]G.Sujana,

[1,2,3,4] Student,Department of ECE,Narsimha Reddy Engineering College,Misammaguda(V),Kompally-500100,Telangana State,India.

[5] Assistant Professor,Department of ECE,Narsimha Reddy Engineering College,Misammaguda(V),Kompally-500100,Telangana State,India.

Abstract—

Using efficient finite field multipliers becomes vital in elliptic curve cryptography (ECC), where data security and authentication are critical. These multipliers do affect performance, however, because they use quite a lot of hardware resources. In order to enhance hardware efficiency on FPGA devices, this study explores the Karatsuba algorithm and its modifications. Although performance is improved with the overlap-free Karatsuba algorithm. Because of problems with recombining intermediate findings, they contribute 20% inaccuracies. In order to solve this problem, we provide a modified Karatsuba method that can compute four key outputs from 2-bit inputs without making any mistakes. Compared to the original Karatsuba algorithm, the improved design, which was tested on Artix-7 FPGA and implemented in Verilog HDL, decreases space utilization by 95% and power consumption by 73.95%. Its overall efficiency is much improved while accuracy is guaranteed, despite a 3.22% increase in area and an 11.12% increase in power compared to the overlap-free version.

Keywords—artix FPGA board, cryptography, karatsuba algorithm, polynomial, xilinx vivado tool.

INTRODUCTION

Numerous fields rely on cryptography for data security and authentication purposes; they include communication devices, autonomous vehicles, the IoT, healthcare, etc. Since public-key and symmetric-key cryptography are two types of cryptography. Secure information exchange is made possible via public-key cryptography, which relies on digital signatures and key setup. Algorithms such as Diffie-Hellman, RSA, ElGamal, ECC, and others fall within the category of public-key cryptography. Because of its compact key sizes and robust security features, ECC is easy to deploy. Faster, cheaper, more efficient, and better suited to the low-power requirements of electronic devices is hardware-based encryption as compared to software-based cryptography. As a result, ECC cryptographic methods are implemented utilizing FPGA (field-programmable gate array) technology. In order to calculate the points of an elliptic curve, finite-field operations must be performed. Because of its larger footprint, the finite-field multiplier has an impact on the algorithm's efficiency. A space-efficient multiplier based on the Karatsuba algorithm (KA) uses fewer multiplications and more addition operations.

With KA, you only need n1.58 bits if the n-bit conventional multiplier can handle n2 single-digit multiplications. Time complexity, or performance, is sacrificed due to KA's iterative nature. There are major restrictions on output, area, and power when using the Karatsuba algorithm in hardware implementation. Its recursive structure reduces the

number of multiplications used compared to naïve techniques, but it increases the number of intermediate additions and subtractions, which might lead to incorrect output in some designs. Because of the complexity of managing intermediate results and recombinations, the method takes up more space, making it less efficient for systems with limited resources, such as FPGAs. Additionally, the performance benefits may be nullified in some applications due to the increased power consumption caused by the increased usage of memory and extra processes. When it comes to systems that prioritize energy economy and compactness, these drawbacks stand out even more. The spark has been lit, and now we can work on creating the Overlap free Karatsuba algorithm. By removing overlapping terms from intermediate calculations, the overlap-free Karatsuba algorithm achieves better output accuracy than its overlap counterpart. Applications requiring precision, like encryption, are well-suited to this method because it enhances dependability. Still, space and power use are somewhat increased due to the increased number of processes caused by removing overlaps. It improves accuracy, but it comes at the cost of hardware resources and energy efficiency, which means it may not be the best choice for systems with limited resources or power. Improving upon the conventional Karatsuba method, the improved algorithm makes it more efficient and accurate. By guaranteeing accurate computations for critical outputs, it fixes mistakes made during intermediate result recombination. By meticulously managing the intermediary stages, the improved version of the multiplication process eliminates the errors seen in the overlap-free variation, particularly for tiny inputs. Even with 2-bit inputs, this ensures error-free outputs without sacrificing the algorithm's efficiency. When compared to older Karatsuba algorithms, the updated version achieves a better balance between speed and accuracy while also making better use of hardware resources, using less space and power, and generally boosting performance.

All of the prior literature articles' extensive introductory material is here. The paper's primary emphasis is on multipliers' areas and powers at the moment. Because of its recursive nature and the need to store intermediate results, the Karatsuba algorithm consumes a lot of hardware room and power, although it is efficient at decreasing multiplication operations. Due to its increased complexity, the overlap-free Karatsuba method uses somewhat more space and power than the normal Karatsuba algorithm, but it increases efficiency by removing unnecessary intermediary stages. However, the updated Karatsuba algorithm is better in terms of power consumption and area optimization. It guarantees error-free outputs while reducing space utilization and power consumption compared to the original Karatsuba algorithm. This improved technique is ideal for hardware implementations with limited resources because, despite a little increase in area and power compared to the overlap-free version, it provides a better balance between efficiency, accuracy, and resource utilization. Encryption, digital signatures, and ECC are some of the modern cryptographic techniques used to protect data. Technologies like secure communications and blockchain are supported by it. Emergence of quantum computing has prompted the creation of new algorithms to guarantee the safety of data in the future. The physical space needed to construct a circuit is called area in hardware design, whereas power is the amount of energy used by the circuit while it is operating. In embedded systems and field-programmable gate arrays (FPGAs), when resources are limited, both are critical to hardware optimization. Minimizing area makes gadgets smaller and cheaper, which in turn makes them more compact. Minimizing power consumption makes them more energy efficient and extends the life of their batteries. The goal of space-and power-efficient design is to provide maximum performance with little waste of either.

This is what drives the effort. Improving the algorithm's performance is a practical usage of pipelines. In prose, several algorithms and methods are suggested. Fast carry chains with look-up tables implemented as multistage ring oscillators have been investigated in [1]. With predictable routing and a high level of frequency sensitivity and PVT (process, voltage, temperature) tolerance, system administrators may be certain that their routing schemes will remain constant, even when hardware slice occupancy reaches 50% and energy use reaches 44%. For ECC asymmetric digit multipliers, the Karatsuba method is formulated and developed in [2]. Hardware efficiency may be enhanced by the use of parallel processing and pipelining techniques. All things considered, the speed is enhanced by the interleaved quick reduction and pipelined modular multiplication algorithms. Implementing countermeasures such as scalar blindness and base-point randomization strengthens resistance against side channel attacks. In [3], we look into Parallel Decimal Multiplier and how symmetric and asymmetric partitioning techniques impact the

International journal of basic and applied research

www.pragatipublication.com

ISSN 2249-3352 (P) 2278-0505 (E)

Cosmos Impact Factor-5.86

Karatsuba algorithm. These methods minimize power consumption by 25% while optimizing the Karatsuba algorithm, however they come at the expense of area. Their ability to localize switching activity also helps to decrease dynamic power dissipation. Therefore, these algorithms are well-suited for portable devices that are power-sensitive due to this characteristic.

In order to efficiently perform point multiplication (PM) and double point multiplication (DPM) for signature and verification operations in ECDSA, the Differential Addition Chain (DAC) was developed in [4]. This design is ideal for time-sensitive 5G applications because it lowers the area-time product and boosts throughput efficiency. In order to decrease the processing time of quantum cryptoanalysis, namely for Shor's solution for the elliptic curve discrete algorithm problem (ECDLP) and binary point addition, Binary Elliptic Curves are used to execute depth optimization in [5]. Optimizing the FLT-based inversion and Karatsuba multiplier's current circuit implementation in the Qiskit quantum computer simulator is what makes it better. Describing a shallower circuit and using fewer CNOT gates improves efficiency. Additionally, this approach removes 90% of the Toffoli gates needed for a one-step point addition. The Urdhva Triyagbhyam Sutra provides the basis for the high-speed Vedic multiplier that is invented in [6] and [7]. But this technique was applied to create a FIR filter, and the suggested filter cuts down on power dissipation by 19.49%, latency by 45.83%, and area by 51.11% [6]. As the number of taps is doubled for precise filtering, the suggested design outperforms integer-based FIR filters. In particular, the delay goes down from 58.04% to 46.95% and the area goes down from 58.72% to 49.77%. Nevertheless, a cost-benefit analysis reveals that power dissipation rises from 45.41% to 47.50% [7]. Rivest, Shamir, Adleman, and other ECC methods are evaluated in [8] for their better encryption and decryption capabilities. The article [9] offers various ways for implementing constrained-resource designs in multiple coordinate systems, such as point multiplication in encryption. Using Python, we evaluate the performance and efficiency of the Montgomery ladder point multiplication algorithm.

The paper suggests a hardware architecture for variable-sized large integers called a flexible Karatsuba multiplier [10]. Optimizing and accommodating operands of various sizes in hardware implementation is achieved via the use of the recursive breakdown technique. When applied to a Xilinx Zynq MP FPGA, this method provides a 9.2× performance boost compared to well optimized software libraries. In order to broaden the scope of cloud data security utilizing ECC, an advanced approach is investigated in [11]. To avoid using the same database transmitter and receiver, we mix the raw data with the CII characters and send it into ECC as a source. In [12], a dual-field architecture for elliptic curve cryptography (ECC) point multiplication (PM) is proposed, which is based on the Montgomery Ladder method and combines 6CC-6CC and 6CC-4CC designs. By making the most of hardware resources, ECC processes are very efficient and compatible with a variety of binary fields, including GF(2^283) and GF(2^571). improved velocity and use of resources. There are several FPGA systems that do the single PM operation in 17.44 µs and 12.55 µs, respectively. In [13], RSA cryptography is used to solve the discrete logarithm issue in small fields and the factorization decomposition problem for huge numbers. A comprehensive analysis of the ECC strategy parameters for the system's critical internals, the steady elliptic curve collecting process, is investigated. The CASIA-IrisV3 database was used to assess the accuracy, security, and privacy of a new ECC-based iris identification system that was created in [14].

Each of the original IrisCode's binary shards is represented by a point. The Toom-Cook 8-way multiplier algorithm, described in [15], is an advanced method for performing polynomial multiplication that has been optimized for quantum multiplication in order to achieve space and time efficiency. Toom-Cook 8-way iterative optimization is efficient with a Toffoli depth of O(n1.0569) and a lower asymptotic qubit count of around O(n1.245). This makes it more resistant to multiplication-based side-channel assaults like Correlation Power Analysis (CPA). The characteristics of Hyper Elliptic Curve Cryptography (HECC), such as its compressibility, untraceability, finite condition, and key size, are studied and enhanced in [16]. The classic KA is made more efficient by the overlap-free Karatsuba algorithm (OKA), which eliminates an XOR gate from the critical path. Nevertheless, this multiplier is slower than the standard one, and the trade-off involves mistakes. In order to enhance speed and decrease calculation

International journal of basic and applied research

www.pragatipublication.com

ISSN 2249-3352 (P) 2278-0505 (E)

Cosmos Impact Factor-5.86

mistakes, this work proposes a modified Karatsuba algorithm that strikes a compromise between area and speed. It takes a lower-level base unit like KA and mixes it with a technique like OKA. For varying operand sizes, all three solutions use FPGA polynomial multiplications. The Karatsuba algorithm, also known as the overlap-free Karatsuba algorithm, is reviewed in Section II. Section III explains the revised Karatsuba algorithm. Section IV covers the discussion of the simulation and implementation findings, followed by the conclusion.

KARATSUBA ALGORITHMS

To improve efficiency over conventional multipliers Karatsuba Algorithm and its variations are developed. Both the 2-bit and 4-bit KA structures are shown in fig.1. One of the most important features of public-key cryptosystems such as RSA, ECC, and Diffie-Hellman is the ability to quickly multiply large integers. The Karatsuba method is used in cryptography for this purpose. One process with extremely huge numbers that these systems depend on is modular exponentiation, which comprises repeated multiplications. The divide-and-conquer technique developed by Karatsuba makes the naïve multiplication algorithm more efficient for large inputs by reducing its temporal complexity to O(n2). Especially in contexts requiring secure key generation and modular arithmetic, cryptographic methods benefit from faster multiplication.

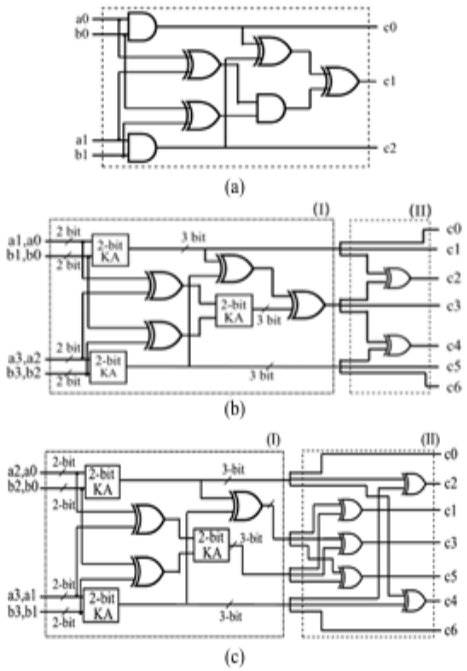


Fig. 1. Hardware implementation of (a) 2-bit KA and (b) 4-bit KA (c) 4-bit OKA

Here is how the calculation is done for two polynomials of degree one: (=2), ()= +, and ()= +. It is shown in figure 1. You may think of two n-term polynomials in (2) as an n-bit multiplier: and (). The following polynomials with a degree of -1 are given in equations (1) through (12).

International journal of basic and applied research

www.pragatipublication.com

ISSN 2249-3352 (P) 2278-0505 (E)

Cosmos Impact Factor-5.86

$$CA_{XOR}(n) = (n-1)^2 \quad (1)$$

$$CA_{AND}(n) = (n)^2 \quad (2)$$

$$T_{CA}(2) = T_x + T_a \quad (3)$$

$$T_{CA}(n) = T_a + log_2(n)\, T_x \quad (4)$$

$$A(x) = \sum_{i=0}^{n-1} a_i x^i \quad (5)$$

$$B(x) = \sum_{i=0}^{n-1} b_i x^i \quad (6)$$

$$A(x) = x^m \sum_{i=0}^{m-1} a_{m+i}\, x^i + \sum_{i=0}^{m-1} a_m\, x^i = A_H\, x^m + A_L \quad (7)$$

$$B(x) = x^m \sum_{i=0}^{m-1} b_{m+i}\, x^i + \sum_{i=0}^{m-1} b_m\, x^i = B_H\, x^m + B_L \quad (8)$$

$$A(x)B(x) = (A_H\, x^m + A_L)(\, B_H\, x^m + B_L) \quad (9)$$

$$= P_2(x)x^{2m} + [P_1(x) - P_2(x) - P_0(x)]x^m + P_0(x) \quad (10)$$

$$P_2 = A_H B_H \quad (11)$$

$$P_1 = (A_H + A_L)(B_H + B_L) \quad (12)$$

The karatsuba algorithm multiplier for 2-bit input produces the equations (11) through (13) as its outputs. The space complexity and area of the KA method are then determined using the following equations: (13)– (19).

$$P_0 = A_L B_L \quad (13)$$

$$KA_{XOR}(n) = 3\, KA_{XOR}\left(\frac{n}{2}\right) + 4n - 4 \quad (14)$$

$$KA_{AND}(n) = 3\, KA_{AND}\left(\frac{n}{2}\right) \quad (15)$$

$$T_{KA}(n) = 3\, T_X + T_{KA}\left(\frac{n}{2}\right) \quad (16)$$

$$KA_{XOR}(n) = 6\, n^{log_2(3)} - 8n + 2 \quad (17)$$

$$KA_{AND}(n) = n^{log_2(3)} \quad (18)$$

$$T_{KA}(n) = T_a + (3\, log_2(n) - 1)T_x \quad (19)$$

International journal of basic and applied research

www.pragatipublication.com

ISSN 2249-3352 (P) 2278-0505 (E)

Cosmos Impact Factor-5.86

The conventional algorithm's quadratic space complexity is reduced to KA's subquadratic space complexity, which is 2(3) = 1.58. Alternatively, the temporal complexity shows an increase from 2( ) 3 2( ). Finally, the area of the multipliers is reduced by KA, but the price is moving at a slower pace. Consequently, the original Karatsuba is optimized for speed using the overlap free KA (OKA), as illustrated in figure 1(c), in order to overcome problems. By dividing the inputs into odd and even orders rather than high and low, we can improve the longest route latency. The assumption that and are two polynomials in (2) and = 2 is restated. Here is the equation (20)-(27) shown.

$$A(x) = \sum_{i=0}^{m-1} a_{2i}\, x^{2i} + \sum_{i=0}^{m-1} a_{2i+1}\, x^{2i+1} \qquad (20)$$

$$B(x) = \sum_{i=0}^{m-1} b_{2i}\, x^{2i} + \sum_{i=0}^{m-1} b_{2i+1}\, x^{2i+1} \qquad (21)$$

$$A(x) = \sum_{i=0}^{m-1} a_{2i}\, y^{i} + x \sum_{i=0}^{m-1} a_{2i+1}\, y^{i} = A_e(y) + xA_O(y) \qquad (22)$$

$$B(x) = \sum_{i=0}^{m-1} b_{2i}\, y^{i} + x \sum_{i=0}^{m-1} b_{2i+1}\, y^{i} = B_e(y) + xB_O(y) \qquad (23)$$

$$4(x)B(x) = \left(A_e(y) + x\, A_o(y)\right) \times \left(B_e(y) + x\, B_o(y)\right) = ;_2(y)y + [G_1(y) - G_2(y) - G_0(y)]x + G_0(y) \qquad (24)$$

$$G_0 = A_e B_e \qquad (25)$$

$$G_1 = (A_o + A_e)(B_o + B_e) \qquad (26)$$

$$G_2 = A_O B_O \qquad (27)$$

In this case, the components of the odd and even terms do not overlap, allowing an OR gate to be removed from the critical path of the Karatsuba multiplier. Equations (25), (26), and (27) provide the results of the 2-bit input multiplier algorithm's overlap freeness. Then, using equation (28) (33), we can determine the OKA algorithm's space and area complexity.

$$OKA_{XOR}(n) = 3\, OKA_{XOR}\left(\frac{n}{2}\right) + 4n - 4 \qquad (28)$$

$$OKA_{AND}(n) = 3\, OKA_{AND}\left(\frac{n}{2}\right) \qquad (29)$$

$$T_{OKA}(n) = 2\, T_X + T_{OKA}\left(\frac{n}{2}\right) \qquad (30)$$

$$OKA_{XOR}(n) = 6\, n^{log_2(3)} - 8n + 2 \qquad (31)$$

$$OKA_{AND}(n) = n^{log_2(3)} \qquad (32)$$

$$T_{OKA}(n) = T_a + (2\, log_2(n) - 1)T_x \qquad (33)$$

International journal of basic and applied research

www.pragatipublication.com

ISSN 2249-3352 (P) 2278-0505 (E)

Cosmos Impact Factor-5.86

We find that the 2-bit KA, according to our calculations, should provide four outputs, but there are only three accessible, which creates an error by definition. Therefore, we use the updated KA to create the KA that is free of errors.

MODIFIED KARATSUBA ALGORITHMS

The original Karatsuba multiplication technique has been refined and improved upon to create the modified karatsuba algorithm. By dividing two big integers into smaller ones and then performing three recursive multiplications instead of four in classical multiplication, the time complexity is reduced from () to (), which is about the same as (). This is the typical Karatsuba method. Possible enhancements to the updated version include decreasing the amount of recursive calls, improving the base cases, or adding more heuristics for number splitting. Overall efficiency for big number multiplication may be improved using these improvements, which can decrease the number of operations, optimize memory use, or tailor the algorithm to certain kinds of numbers or hardware architectures. Fig.2 shows the use of the modified KA for multiplication due to the fact that error-free smaller blocks are necessary for efficient and safe ECC. If we take the same polynomial form into consideration, then

$$C0 = a0 \,.\, b0 \tag{34}$$

$$C1 = (a0 \,.\, b1) + (a1 \,.\, b0) \tag{35}$$

$$C2 = a1 \,.\, b1 \,.\, (a0.\, b0) \tag{36}$$

$$C3 = a1 \,.\, b1 \tag{37}$$

This modified karatsuba method multiplier takes a 2-bit input and produces the following equations: (34), (35), (36), and (37). Then, using the formulas in equations (38)–(43), we can determine the OKA algorithm's area and space complexity.

$$KA_{XOR}(n) = 3\, KA_{XOR}\left(\frac{n}{2}\right) + 4n - 4 \tag{38}$$

$$KA_{AND}(n) = 5\, KA_{AND}\left(\frac{n}{2}\right) \tag{39}$$

$$T_{KA}(n) = 3\, T_X + T_{KA}\left(\frac{n}{2}\right) + 1\, KA_{NOT} \tag{40}$$

$$KA_{XOR}(n) = 6\, n^{log_2(3)} - 8n + 2 \tag{41}$$

$$KA_{AND}(n) = n^{log_2(5)} \tag{42}$$

$$T_{KA}(n) = T_a + (3\, log_2(n) - 1)T_x. \tag{43}$$

International journal of basic and applied research

www.pragatipublication.com

ISSN 2249-3352 (P) 2278-0505 (E)

Cosmos Impact Factor-5.86

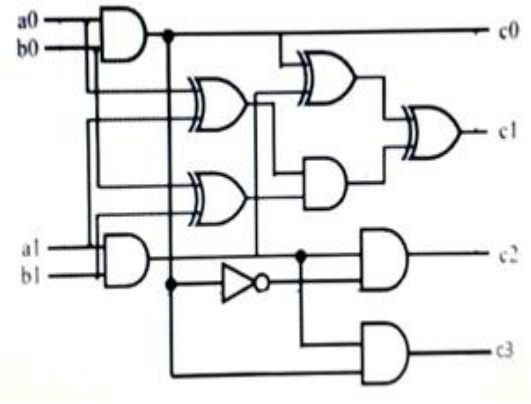


Fig. 2. Hardware implementation of 2-bit Modified KA

RESULTS AND DISCUSSION

Xilinx Vivado is used to design the multipliers for the Artix-7 FPGA board, which is based on 28nm CMOS technology (XC7A200TLFFG1156-2L). Windows 10, 4 GB of RAM, and Xilinx Vivado 2023 are the minimum system requirements.2. The developed multipliers' simulation results are given in figure 3, with a and b serving as inputs and c as the output.

Fig. 3. Simulation Result of 4-bit KA

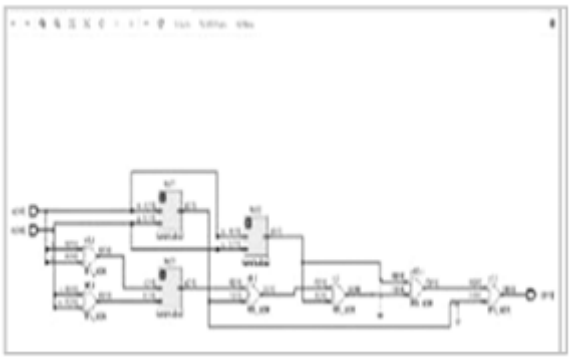
Figure 4 shows the RTL and technological schematics of 4-bit KA, OKA, and MKA. Despite having the simplest RTL schematic, the OKA makes repeated use of the same block. Data lookup tables and input/output buffers are used in the technical diagram. For MKA, the technological diagram is the most basic.

International journal of basic and applied research

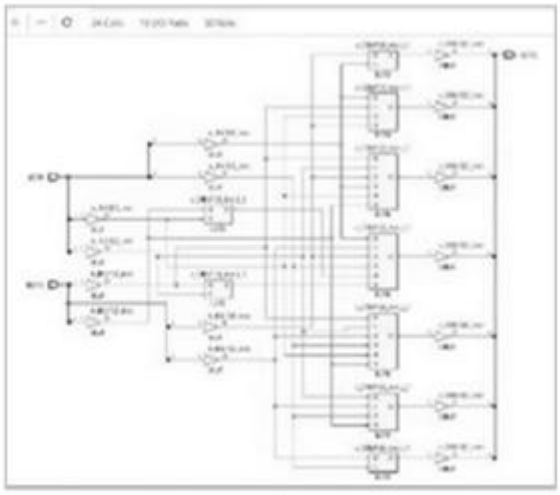
www.pragatipublication.com

ISSN 2249-3352 (P) 2278-0505 (E)

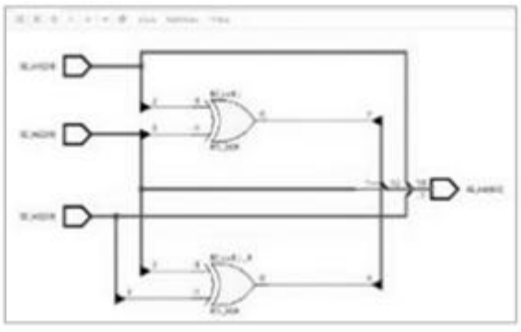
Cosmos Impact Factor-5.86



(a)



(b)



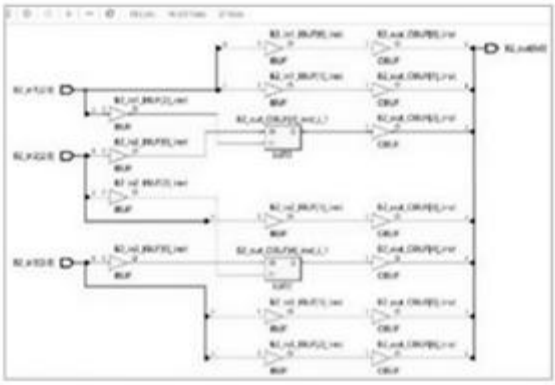
(c)

International journal of basic and applied research

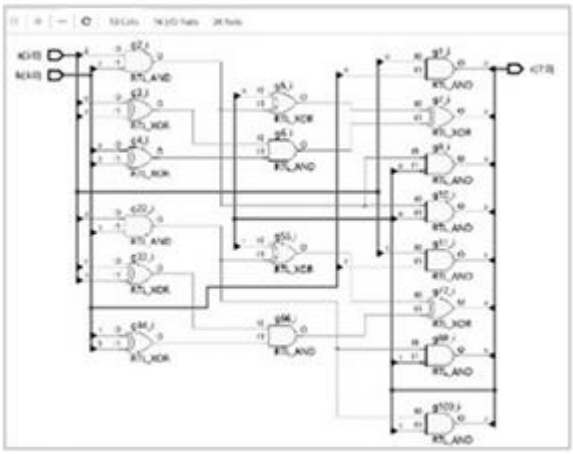
www.pragatipublication.com

ISSN 2249-3352 (P) 2278-0505 (E)

Cosmos Impact Factor-5.86



(d)



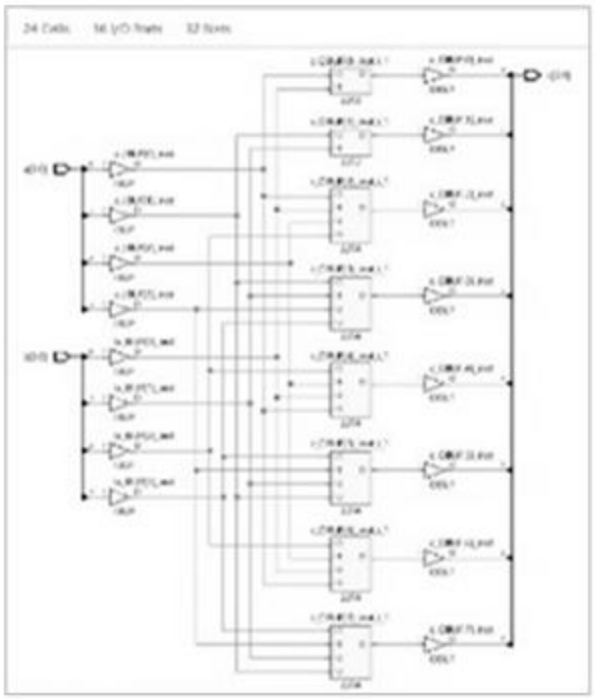
(e)

International journal of basic and applied research

www.pragatipublication.com

ISSN 2249-3352 (P) 2278-0505 (E)

Cosmos Impact Factor-5.86



(f)

Fig. 4. RTL and Technology Schematics of 4-bit KA, OKA and MKA (a) RTL Schematic of 4-bit KA (b) Technology Schematic of 4-bit KA (c) RTL Schematic of 4-bit OKA (d) Technology Schematic of 4-bit OKA (e) RTL Schematic of 4-bit MKA (f) Technology Schematic of 4-bitM KA The synthesis and implementation results are as tabulated in Table I.

As the number of bits used for input grows, so do the Slice LUTs. When comparing the MKA to the KA and OKA, the slice LUTs are reduced by 95.35% and -3.22%, respectively, as shown in Table I. When contrasted with KA and OKA, the MKA significantly lowers the on-chip power consumption by 83.88% and 24.89%, respectively. The decline, not the betterment, is shown by the negative sign. The KA, OKA, and MKA algorithms undergo thorough testing to ensure they do not introduce any problems. The results show that KA mistakes account for 20% of the total and OKA errors for 60%. No errors were detected during the functioning of the MKA. The recursive nature of the Modified Karatsuba Algorithm (MKA) causes its dynamic power consumption to grow with both the input bit width and the polynomial degree (see Table I). There are more sub-problems and operations (such addition, subtraction, and multiplication) that result in more recursive calls as the input bit-width rises. Because the method is based on reducing huge numbers to smaller ones, increasing the bit-width leads to more recursive levels, which raises the power consumption because each level requires more operations. A similar trend holds true for polynomials: the complexity and computing resources required to solve them rise in direct proportion to the degree of the polynomial. More clock cycles, more switching activity, and more dynamic power consumption are the outcomes of this. A substantial increase in power consumption is caused by both of these variables as the input size rises.

TABLE I COMPARISON IN TERMS OF AREA AND POWER DISSIPATION

International journal of basic and applied research

www.pragatipublication.com

ISSN 2249-3352 (P) 2278-0505 (E)

Cosmos Impact Factor-5.86

Parameter	4-bit		8-bit				16-bit			32-bit			64-bit		
Algorithms	*KA*	*OKA*	*MKA*	*KA*	*OKA*	*MKA*	*KA*	*OKA*	*MKA*	*KA*	*OKA*	*MKA*	*KA*	*OKA*	*MKA*
Slice LUT's (out of 134600)	7	2	4	31	6	8	117	14	16	405	30	32	1377	62	64
On chip power (mW)	3.29	2.21	2.057	10.863	4.943	4.223	28.922	0.111	8.226	74.303	21.89	16.55	205.007	43.981	33.032
Dynamic Power (mW)	3.166	2.087	1.935	10.721	4.815	4.097	28.681	0.000	8.090	73.068	21.70	16.38	203.772	43.540	32.750
Static Power (mW)	0.125	0.123	0.123	0.142	0.128	0.127	0.241	0.111	0.135	1.235	0.190	0.163	1.235	0.440	0.282
Logic Power (mW)	0.022	0.003	0.017	0.173	0.009	0.033	0.914	0.000	0.058	5.435	0.056	0.131	24.621	0.124	0.262
I/O power (mW)	2.955	2.051	1.876	9.981	4.707	3.800	25.201	0.000	7.600	55.619	20.64	15.20	116.649	42.450	30.377
Signal Power (mW)	0.189	0.032	0.041	0.567	0.099	0.264	2.566	0.000	0.432	12.013	1.003	1.056	62.502	0.966	2.111
Total Delay	3.118	7.410	7.880	8.453	7.711	5.725	2.608	2.608	2.600	1.939	10.73	5.725	4.498	5.700	5.725
Efficiency	60% to 70%	70% to 85%	80% To 90%	60% to 70%	70% to 85%	80% To 90%	60% to 70%	70% to 85%	80% To 90%	60% to 70%	70% to 85%	80% To 90%	60% to 70%	70% to 85%	80% To 90%

CONCLUSION

Elliptic curve cryptography uses finite field multipliers, which increase the algorithm's footprint and slow it down. To get around this, hardware efficiency is improved using the Karatsuba algorithm and variations. Three algorithms—Karataba, overlap free Karatsuba, and modified Karatsuba—are tested on an Artix-7 FPGA using Verilog HDL models. Although it rises by 3.22% and 11.12% when compared to the overlap free Karatsuba algorithm, the findings show that the area occupied by the Modified Karatsuba Algorithm is at least 95% and 73.95% smaller than that of the Karatsuba algorithm, respectively, when considering slice LUTs and power dissipation. Additional optimization of the suggested method for enhanced accuracy and reduced area is possible in the future.

REFERENCES

[1]. F. Spagnolo, S. Perri, F. Frustaci, F. Crupi, M. Vatalaro, and P. In IEEE Transactions on Very Large Scale Integration (VLSI) Systems,volume, Corsonello explains "Exploring the Usage of Fast Carry Chains to Implement Multistage Ring Oscillators on FPGAs: Design and Characterisation." August 2024; 32, no. 8, pp. 1472-1484; doi: 10.1109/TVLSI.2024.3395302.

[2]. A. J. Park, M. Awaludin, R. W. Wardhani, and H. Kim, 2009. "A High Performance ECC Processor Over Curve448 Based on a Novel Variant of the Karatsuba Formula for Asymmetric Digit Multiplier," in the IEEE Access, vol. 10, 20, 22; doi: 10.1109/ACCESS.2022.3184786, pp. 67470-67481.

[3]. S. Gorgin, Nejad, H. Z., and J. "A Practical Energy/Power Reduction Approach for Parallel Decimal Multiplier," by A. Lee, IEEE Access, vol. doi: 10.1109/ACCESS.2022.3145001. 10, pp. 11372-11381, 2022.

[4]. X. He and colleagues, "A Universal Architecture for Single and Double Point Multiplications for ECDSA Based on Differential Addition Chains," in IEEE Access, vol. Publication number: 10.1109/ACCESS.2024.3390244, pages. 55434-55447, 2024.

[5]. D. H. T. Larasati, J. Ji, H. Putranto, R. W. Wardhani, and H. Kim "Depth-Optimization of Binary Elliptic Curves for Quantum Cryptanalysis," in IEEE Access, vol. 11, 2023; doi: 10.1109/ACCESS.2023.3273601. Pages. 45083-45097.

[6]. Satyam, Neelima K, M. Sandhiya, C. Padma, Shaik Jaffar Ali and Kumar Raja Meruva, "High Speed Single Precision 64-Tap FIR Filter Using Urdhva Tiryagbhyam Sutra," 2024 IEEE Students Conference on Engineering and Systems (SCES), Prayagraj, India, 2024, pp. 1-5, doi: 10.1109/SCES61914.2024.10652418.

[7]. Neelima K, H. Yogananda Reddy, G. Bhaskar, N. Mani Teja and N. K. Priya, "Design and Evaluation of 32-Bit N-Tap FIR Filter for Audio Processing Applications," 2024 1st International Conference on Innovative Sustainable Technologies for Energy, Mechatronics, and Smart Systems (ISTEMS), Dehradun, India, 2024, pp. 1-6, doi: 10.1109/ISTEMS60181.2024.10560173.

[8]. K. Shah, A. Bhadauria, P. Thakkar, J. Shah and H. Kaur, "Advancements in Elliptic Curve Cryptography: A Review of Theory and Applications," 2024 Parul International Conference on Engineering and Technology (PICET), Vadodara, India, 2024, pp. 1-6, doi: 10.1109/PICET60765.2024.10716041.

[9]. N. H. Sabbry and A. Levina, "Elliptic Curve Cryptography on Constrained Devices: A Comparative Study of Point Multiplication Methods," 2024 13th Mediterranean Conference on Embedded Computing (MECO), Budva, Montenegro, 2024, pp. 1-5, doi: 10.1109/MECO62516.2024.10577953.

[10]. B. H and Kang. Huang, "FlexKA: A Flexible Karatsuba Multiplier Hardware Architecture for Variable-Sized Large Integers," in IEEE Access, volume Publication 10.1109/ACCESS.2023.3282646, pages 55212–55222, 2023. Nu

[11]. V. Srinadh, B. Maram and T. Daniya, "Data Security And Recovery Approach Using Elliptic Curve Cryptography," 2021 IEEE International Conference on Computation System and Information Technology for Sustainable Solutions (CSITSS), Bangalore, India, 2021, pp. 1-6, doi: 10.1109/CSITSS54238.2021.9683473.

[12]. J. Li, Y. Luo, W. Wang, J. Zhang, and S. Chen, "Innovative Dual Binary-Field Architecture for Point Multiplication of Elliptic Curve Cryptography," in IEEE Access, volume In 2021, volume 9, pages 12405–12419, doi: 10.1109/ACCESS.2021.3051282.

[13]. J. VenkataGiri and A. Murty, "Elliptical Curve Cryptography Design Principles," 2021 International Conference on Recent Trends on Electronics, Information, Communication & Technology (RTEICT), Bangalore, India, 2021, 10.1109/RTEICT52294.2021.9573662. pp. 889-893, doi:

[14]. A. A. Asaker, Z. F. Elsharkawy, S. Nassar, N. Ayad, O. Zahran and F. E. Abd El-Samie, "A Novel Iris Cryptosystem Using Elliptic Curve Cryptography," 2021 9th International Japan-Africa Conference on Electronics, Communications, and Computations (JAC-ECC), Alexandria, Egypt, 2021, pp. 155-158, doi: 10.1109/JAC ECC54461.2021.9691307.

[15]. D. H. T. Larasati, H. C. Putranto, R. W. Wardhani, and H. "Space and Time-Efficient Quantum Multiplier in Post Quantum Cryptography Era," by Kim, IEEE Access, vol. doi: 10.1109/ACCESS.2023.3252504; 11, pp. 21848-21862, 2023.

[16]. A. Yadav, P. Sharma and Y. Gigras, "A Comparative Study of Elliptic curve and Hyperelliptic Curve Cryptography Methods and an Overview of Their Applications," 2024 International Conference on Intelligent Systems for Cybersecurity (ISCS), Gurugram, India, 2024, pp. 01-06, doi: 10.1109/ISCS61804.2024.10581015.