*International Journal of LNCT, Vol 2(5)*
*ISSN (Online): 2456-9895*

# An extensive review onFFT based montgomery multiplication algorithm

**Vijeta Raichur[1] and Laxminarayan Gahalod[2]**
M.Tech Scholar, LNCT, Bhopal[1]
Professor, LNCT, Bhopal[2]

## Abstract

***The role of network security in the field of networking, is immense. In the information era it is important to keep information secure about every aspect of our live. Hence to keep the information securely is known as cryptography. There are many cryptographic algorithms are developed by many researchers to achieve the information security but it is essential to plan an algorithms in such a manner that an opponent cannot defeat its purpose. These algorithms basically contains of some arithmetic and logical operations which are complicated and time consuming. In addition if the related system is of high speed, the speed of the fundamental cryptographic algorithms additionally should be considered. Cryptographic algorithms utilize modular multiplication intensively. Most basic algorithms are the RSA Algorithm, named after its creators Rivest, Shamir and Adleman, and the recently rising elliptic curved cryptosystems (ECC) descibed. While performance and Area stay to be the two noteworthy outline tolls, control consumption has turned into a basic worry in the present VLSI framework design. There are numerous executions of high– radix regular multipliers exist however, there are few high– radix secluded multipliers in the writing are talked about. In this work a broad overview of writing review in light of Montgomery Multiplication has given.***

## Keywords

***Montgology modular multilpication, Number theoriticauthorsightet transform, FFT, Field programmable gate array (FPGA).***

## 1.Introduction

Given x, y € Zm = {0, 1, 2… m-1}, to compute z = xymod(m), where x, y, z are k bit natural number. Modular multiplication can be done in various ways: 1) Multiply and reduce: this is the straight forward method in which two numbers are multiplied together and then the value is reduced to n bit with the help of a reducer circuit. So th hardware consists of any type of multilplier and any reducer. 2) Double add and reduce: in this way modular multiplication is carried out by addition, doubling and then reduction operation. Hardware of this algorithm consists of a modular adder. 3) Montgomery production: in this way modular multiplication is based on an algorithm P.L. Montgomery, which replaces the trial division method using addition and shifting operation.

A few Montgomery multiplications algorithms are talks about, two of which have been reported. Authors portray three extra calculations, and investigate in detail the space and time prerequisites of all ve strategies. These calculations are actualized in C and in constructing agent. The investigations and genuine execution brings about dicate that the Coarsely Incorporated Operand Filtering (CIOS) strategy, point by point in this segment, is the most e cient of all ve calculations, at any rate for the general class of master cessorauthors considered. The Montgomery augmentation strategies constitute the center of the modular exponentiation activity which is the most authors known technique utilized as a part of open key cryptography for encoding and checking progressed data.The inspiration for concentrate high-speed and space-e cient calculations for modular multiplication originates from their applications in broad daylight key cryptography. The RSA al-gorithm and the Di e-Hellman key trade conspire require the calculation of modular exponentiation, which is broken into a progression of modular multiplications by the use of the double or m-array strategies. Different equipment calculations for modular increase have been proposed. Modular exponentiation calculations utilizing division chains a twofold base number framework and complex math are appropriate to programming executions. Hoauthorsver, these methods concentrate on fast modular exponentiation, not on the particular modular multiplication method employed.

Surely a standout amongst the most intriguing and helpful advances has been the presentation of the alleged Montgomery multiplication calculation because of Subside L.Mongtomery (for some of the recent applications see the discussion by Naccache et al. Ko c et al. and Bajard et al.Various hardware implementations of the Montgomery multiplication have been proposed and some of them have been used in commercially available chips. The Montgomery multiplication algorithm is used to speed up the modular multiplications and squarings required during the exponentiation process. The Montgomery algorithm computes.

MonPro(a;b)=abr1_(mod……..(1.1))

The Montgomery augmentation calculation is thought to be the quickest calculation to process X*Y mod M in PCs when the estimations of X, Y and M are extensive. Another productive calculation for modular increase is the interleaved modular multiplication calculation .In this theory, two new calculations for modular augmentation and their relating models which authors proposed in are executed. Thesealgorithms are improvements of Montgomery increase and interleaved modular multiplication. Author is improved as for region and time multifaceted nature. In the two calculations the result of two n bit whole numbers X and Y modulo M arefigd by n emphasess of a basic circle. Each circle comprises of one single convey spare expansion, an examination of constants, and a survey query.

once a RSA cryptosystem is set up, i.e., the modulus and the private and open types are resolved and the general population segments have been distributed, the senders and in addition the recipients play out a solitary activity for marking, verication, encryption, and unscrambling. The RSA calculation in this regard is one of the least difficult cryptosystems. The activity required is the calculation of Me (mod n), i.e., the modular exponentiation. The modular exponen-tiation activity is a typical task for scrambling; it is utilized as a part of a few cryptosystems. For instance, the Di e-Hellman key trade plot requires secluded exponentiation. Moreover, the ElGamal signature plot and the as of late proposed Computerized Mark Standard (DSS) of the National Establishment for Benchmarks and Innovation additionally. Notwithstanding, authors take note of that the exponentiation procedure in a cryptosystem in view of the discrete logarithm issue is marginally di erent: The base (M).

## 2.Multiplier design

A Binary multiplier is an electronic device used in digital electronics or in a computer or other electronic devices to carry out multiplication of two numbers depicted in binary format. It is built using binary adders. The most basic technique involves generating a set of partial products, and then summing the partial products simultaneously. This process is similar to the method which is taught to lower classes' students in school for conducting long multiplication on base-10 integers, but has been modified here for application to a base-2 (binary) numeral system.
The rules for binary multiplication are stated as given:

1. If the multiplier digit is 1, the multiplicand is copied down and it gives the product.
2. If the multiplier digit is 0 then we get a product which is also 0.

For designing such a multiplier circuit we should have the circuitry to carry out or do the following four things:

1. It should be capable of recognizing whether a bit is 0 or
2. It should be capable of shifting the left partial product.

3. It should be capable of adding all the partial-products to give the product as a sum of the partial products.
4. It should examine sign bits and if they are similar, the sign of the product will be a Positive representation and if the sign bits are opposite then the product will be negative.

The sign bit of the product which has been stored with the above criteria should be displayed along with the product.

Multiplication consists of three steps: generation of partial products or (PPG), reduction of partial products (PPR), and finally carry-propagate addition (CPA).In general there are sequential and combinational multiplier implementations. We only consider combinational case here because the scale of integration now is large enough to accept parallel multiplier implementations in digital VLSI systems. Different multiplication algorithms vary in the approaches of PPG, PPR, and CPA. For PPG, radix-2 is the easiest. To reduce the number of PPs and consequently reduce the area/delay of PP reduction, one operand is usually recoded into high-radix digit sets. The most popular one is the radix-4 digit set {-2,-1, 0, 1, 2}. For PPR, two alternatives exist: reduction by rows , performed by an array of adders, and reduction by columns, performed by an array of counters. The final CPA requires a fast adder scheme because it is on the critical path. In some cases, final CPA is postponed if it is advantageous to keep redundant results from PPG for further arithmetic operations.

The implementation of Montgomery multiplication involves making the tradeoff between chip area and computational speed [9-12]. Two main points to be considered are that with the increasing radix, the multiplier operand is processed in less clock cycles,

however the longest path increases. Thus, the overall effect on the computational time is a decision to be made for multiplier core design.

**A. FFT Based multiplication**

The speediest augmentation calculations utilize the quick Fourier transform. In spite of the fact that the quick Fourier transform was initially created for convolution of groupings, which adds up to duplication of polynomials, it can likewise be utilized for augmentation of long whole numbers. In the standard calculation, the whole numbers are spoken to by the commonplace positional documentation. This is accomplished by assessing these polynomials at the foundations of solidarity, at that point duplicating these qualities pointwise, and nally inserting these The quick Fourier transform calculation enables us to assess a given polynomial of degree s 1 at the s foundations of solidarity utilizing O(s log s) number-crunching activities. Essentially, the introduction step is performed in O(s log s) time.

**B. Field programmable gate array (FPGA) BASED multiplication**

As far as assets, this plan would be appropriate for FPGA.Similar two-dimensional systolic arrays are displayed in. For a radix of two author all propose a m × m grid of one piece preparing components. With this configuration 2m modular increases are computed in the meantime and thetheoretical throughput is one modular duplication for each clock cycle. As far as assets, such an ansauthorsr isn't practical in either VLSI or FPGA for the bit length required out in the open key calculations. Notwithstanding executing just a single column of handling components, (bringing about m times sloauthorsr throughput) into by and by accessible FPGAs is difficult as far as assets. 1. Avoid the convey engendering delays by keeping halfway outcomes in repetitive portrayal. Determination into paired portrayal is just done at the very end and for bolstering the halfway outcome back as ai in Calculation Systolic Arrays: Preparing units fig progressive esteems for a solitary digit position. The registered conveys, qi and ai, are "pumped" through the handling units.

## 3. Literature review

Table 1 References

| Sr. No. | Title | Author | Year | Approach |
|---|---|---|---|---|
| 1 | Design and implementation of different architectures of montgomery modular multiplication | S. Kavyashree and B. V. Uma | 2017 | Architectures of Montgomery Multiplication and their performance is compared in terms and area and time optimization. |
| 2 | Applications to montgomery modular multiplication of karatsuba multiplication without overlapped summation, | Z. Gu and S. Li, | 2017 | A method of avoiding the overlapped summation under the circumstances of calculating Montgomery Modular Multiplication |
| 3 | Elliptic Curve Cryptography implementation on FPGA using Montgomery multiplication for equal key and data size over GF(2m) for Wireless Sensor Networks, | Leelavathi G, Shaila K and Venugopal K R | 2016 | Reported the point multiplication using Montgomery multiplication technique that achieves considerable speed and with reduced area utilization |
| 4 | Low-Cost High-Performance VLSI Architecture for Montgomery Modular Multiplication | S. R. Kuang, K. Y. Wu and R. Y. Lu, | 2016 | Authors proposed Montgomery modular multiplier can accomplish higher execution and huge territory time item change when contrasted and past plans. |
| 5 | CRT RSA decryption: Modular exponentiation based solely on Montgomery Multiplication | J. C. Néto, A. F. Tenca and W. V. Ruggiero | 2015 | A creative equipment configuration is proposed to perform modular exponentiation utilizing just Montgomery Increase for CRT RSA unscrambling. |
| 6 | Time-efficient computation of digit serial Montgomery | W. Dai, H. Wu and R. C. C. Cheung | 2014 | The author proposed that an most critical digit (MSD) first digit |

| Sr. No. | Title | Author | Year | Approach |
|---|---|---|---|---|
| | multiplication | | | seriamontgomery multiplication in an exception class of two fold GF (2m). |
| 7 | Efficient design of Elliptic Curve Point Multiplication based on fast Montgomery modular multiplication | M. Mohammadi and A. S. Molahossein | 2013 | In this study, in light of RNS Montgomery modular increase, an enhanced ECPM design is proposed |
| 8 | Parallelization of Radix-2 Montgomery Multiplication on Multicore Platform | J. Han, S. Wang, W. Huang, Z. Yu and X. Zeng, | 2013 | presents an enhanced undertaking dividing of the Montgomery increase calculation for the multicore stage with area-efficient processors |

S. Kavyashree and B. V. Uma [1] The Montgomery multiplication is the fundamental square of modular exponentiation in cryptography. This study discusses the three different architectures of Montgomery Multiplication and their performance is compared in terms and area and time optimization. The architectures are designed to improvise the area and reduce time. The designs are implemented in Verilog HDL and simulated using Synopsys VCS.author are also synthesized in Synopsys Design Compiler using 45nm libraries to get the cell area. The experimental results show that the time required for one Montgomery multiplication measured in terms of number of clock cycles is reduced by 1.5%, 1% and 3.5 % for the three different architectures discussed here over the previous implementations. It is achieved by minimizing the signals controlling the critical path of the design. Also there is a reduction in total cell area due to technology for the three designs presented in this study.

Z. Gu and S. Li.[2] Karatsuba Multiplication Algorithm is commonly used in modular multiplications of public-key cryptosystems with large key sizes. The overlapped summation in Karatsuba Multiplication is difficult for parallel acceleration in hardware implementations. This exploration proposes a method of avoiding the overlapped summation under the circumstances of calculating Montgomery Modular Multiplication. The proposed method has fewer addition rows and better parallelism.

Leelavathi G, Shaila K and Venugopal K R, [3] In public key cryptography, The RSA algorithm has been utilized for quite a while, yet it doesn't meet the constraints of WSNs. Elliptic Curve Cryptography(ECC) has been utilized as of late due to its high security for same length bit. ECC point multiplication activity is tedious which influences the speed of encryption and decoding of information. In this exploration, we propose the point multiplication using Montgomery multiplication technique that achieves considerable speed and with reduced area utilization. The ECC is first simulated on different FPGA devices, with key length 112 and 163 bits and the area-speed tradeoff is compared. ECC calculation is executed with programming and hardware picking Artix 7 XC7a100t-3csg324 FPGA which underpins key lengths of 112 and 163 bits. The proposed ECC algorithm is modeled using VHDL and synthesized on Spartan 3 and 6, Virtex 4, 5 and 6 and Artix 7 before the hardware implementation on Atrix 7. The plan fulfills the necessities of asset obliged WSNs gadgets with measure up to key length and information estimate, the gadget use is inside 13 rates.

C. Massoud, A. Sghaier, M. Zeghid and M. Machhou[4] Recently, a lot of progress has been made in the implementation of asymmetric cryptography such that RSA or ECC (Elliptic Curve Cryptography) in both hardware and software. The Residue Number Systems (RNS) offer, many features make it very useful in cryptographic applications. Since the modular multiplication is the main operation, in this exploration, we describe a Montgomery modular multiplication algorithm based on RNS. Then we implemented our design in TM i3 CPU, it computed the modular multiplication in only 9 ms (latency) and achieving maximum throughput of 528.

J. C. Néto, A. F. Tenca and W. V. Ruggiero [5]An imaginative equipment configuration is proposed to perform modular exponentiation utilizing just Montgomery Multiplication for CRT RSA decoding. A comparable gear used to perform exponentiation is moreover used to perform transformations. The proposed computation is depicted and given a versatile gear utilization. At the point when contrasted with the traditional consecutive Radix-2 MM design from which it was inferred, the new RSA engineering indicates 44% normal diminishment in

the vitality utilization. The proficient outline proposed is appeared through a trial blend with a 90nm CMOS innovation. The outcomes are contrasted and the condition of-craftsmanship in the RSA 1024-piece executions utilizing non-RNS arrangements.

W. Dai, H. Wu and R. C. C. Cheung [6]In this study, authors have proposed a most-noteworthy digit (MSD) first digit-serial Montgomery multiplication (MM) in an uncommon class of double field GF (2m). The field is created by unchangeable pentanomials fulfilling predefined conditions as recorded in the study. The estimation of R(x) is unique in relation to the current detailed work: R(x) = xm or R(x) = xm-1. Authors demonstrated that execution of MM in such extraordinary class of parallel fields which can be additionally enhanced as far as basic way delay by a most extreme of 63%. Correlation comes about likewise demonstrate that the gate check of the proposed design has been diminished contrasted with the previous works.

M. Mohammadi and A. S. Molahosseini [7]In Elliptic Bend Cryptography (ECC), Elliptic Bend Point Increase (ECPM) is a standout amongst the most basic tasks. In this study, in light of RNS Montgomery modular increase, a streamlined ECPM design is proposed. The proposed engineering incorporates quick RNS to RNS converter with picking proper moduli sets. The proposed RNS bases in first moduli set utilizes the premise with little Hamming authorsight in view of the work revealed in writing and the moduli set {2n+β, 2n - 1,2n + 1,2n - 2(n+1)/2 + 1,2n + 2n+1/2} + 1, 2n-1 +1} in the a respectable halfway point, with proficient invert converter is utilized. To plan the quick RNS to RNS converter, deferral of twofold to buildup converter from first to second premise is moved forward. Equipment plan for basic moduli in second bases which is the moduli 2n + 2(n+1)/2 + 1 is finished. In view of accomplished equipment for decrease in moduli 2n + 2(n+1)/2 + 1, the postpone prerequisites of the new converter is appeared to be not as much as another announced converter. Contrasted with cutting edge usage in the writing, the outcomes demonstrates that the proposed ECPM engineering accomplishes speed increment of 4%, 42%, 35%.

J. Han, S. Wang, W. Huang, Z. Yu and X. Zeng [8]Montgomery augmentation is the part task out in the open key figs. Going for parallel usage of Montgomery increase, this short shows an enhanced undertaking apportioning of the Montgomery augmentation calculation for the multicore stage with zone productive processors. A few multicore stages are intended to check the productivity of parallelization. The speediest stage takes 3460 cycles to complete a 1024-b Montgomery augmentation, which is six times speedier than a solitary MIPS processor and three times quicker.

## 4. Problem statement

Montgomery multiplication and authors decrease the use of FPGA assets. Authors have actualized the modular multiplication in a settled number of clock cycles. To the best of that knowledge, this is the first time that hardware or a programming multiplier of modular Montgomery multiplication, suitable for various security level, is performed in only 33 clock cycles. Besides, to the extent authors know the Montgomery multiplication. The execution of motgomery multiplication algorithms on reconfigurable gadgets, for example, FPGAs. Information is stacked into the multipliers by methods for registers with memory intended to hold the info information bits. An elective approach is to utilize the irregular access memory (Slam) squares of the FPGA. This may lessen the level of CLBs required for their designs. The algorithm displayed in require the full exactness bit of wieght length transform of operands inside the multiplier. Research outtroughto likewise be gone for exploring word sarvey increase. This would require a few alterations to the algorithm. infer arrangements that can fit into a solitary FPGA, a plan objective that has numerous cost and configuration focal points over multi FPGA arrangements. Another essential target was to efficiently execute different engineering alternatives for different bit lengths and look at execution and asset usage.Develop and actualizes an outline that is extensively quicker than any beforehand announced FPGA design.

## 5. Conclusion

The IC technology is getting more complex day by day in terms of design and its performance analysis. A hi speed design with less power consumption and smaller in size is implicit to the modern electronic designs paradigm. In this research Montgomery multiplication that the recently advanced algorithm and their relating designs in for doing modular multiplication requireleased equipment assets and offer faster speed of algorithm contrasted with multipliers with the old Montgomery. Enhancing processing speed and size of a multiplier is a noteworthy plan issue these days. Nonetheless, area and speed are typically clashing constraints so

enhancing speed comes about in larger areas and the other way around. Likewise area and power consumption of a circuit are straightly connected. So a compromise must be done in speed of the circuit for a more noteworthy lessening of area and power. For applications where speed of computation is critical, the Optimized Interleaved algorithm is recommended. For applications where both area and time are limiting factors, the Optimized Interleaved architecture offers a better performance compared to Fast Montgomery and Faster Montgomery.

## References


[1] Kavyashree S, Uma BV. Design and implementation of different architectures of montgomery modular multiplication. InRecent Trends in Electronics, Information & Communication Technology (RTEICT), 2017 2nd IEEE International Conference on 2017 (pp. 1101-5). IEEE.

[2] Gu Z, Li S. Applications to montgomery modular multiplication of karatsuba multiplication without overlapped summation. In Electron Devices and Solid-State Circuits (EDSSC), 2017 International Conference on 2017 (pp. 1-2). IEEE.

[3] Leelavathi G, Shaila K, Venugopal KR. Elliptic Curve Cryptography implementation on FPGA using Montgomery multiplication for equal key and data size over GF (2 m) for wireless sensor networks. In Region 10 Conference (TENCON), IEEE 2016 (pp. 468-71). IEEE.

[4] Massoud C, Sghaier A, Zeghid M, Machhout M. Efficient software implementation of RNS-montgomery modular multiplication for embedded system. In Image Processing, Applications and Systems (IPAS), International 2016 (pp. 1-5). IEEE.

[5] Néto JC, Tenca AF, Ruggiero WV. CRT RSA decryption: modular exponentiation based solely on Montgomery multiplication. In Signals, Systems and Computers, 2015 49th Asilomar Conference on 2015 (pp. 431-6). IEEE.

[6] Dai W, Wu H, Cheung RC. Time-efficient computation of digit serial Montgomery multiplication. In Integrated Circuits (ISIC), 2014 14th International Symposium on 2014 (pp. 212-15). IEEE.

[7] Mohammadi M, Molahosseini AS. Efficient design of elliptic curve point multiplication based on fast montgomery modular multiplication. In Computer and Knowledge Engineering (ICCKE), 2013 3th International eConference on 2013 (pp. 424-9). IEEE.

[8] Han J, Wang S, Huang W, Yu Z, Zeng X. Parallelization of radix-2 Montgomery multiplication on multicore platform. IEEE Transactions on Very Large Scale Integration (VLSI) Systems. 2013; 21(12):2325-30.

[9] David JP, Kalach K, Tittley N. Hardware complexity of modular multiplication and exponentiation. IEEE Transactions on Computers. 2007; 56(10).

[10] Chen DD, Yao GX, Cheung RC, Pao D, Koç CK. Parameter space for the architecture of FFT-based Montgomery modular multiplication. IEEE Transactions on Computers. 2016; 65(1):147-60.

[11] Eldridge SE, Walter CD. Hardware implementation of Montgomery's modular multiplication algorithm. IEEE transactions on Computers. 1993; 42(6):693-9.

[12] Walter CD. Systolic modular multiplication. IEEE Transactions on Computers. 1993; 42(3):376-8.