Industrial Engineering Journal
ISSN: 0970-2555
Volume : 51, Issue 8, August : 2022

DESIGN AND IMPLEMENTATION OF A HIGHLY SECURE MN-HOMOMORPHIC ENCRYPTION SYSTEM BASED ON VLSI

[1]M.Nikhil Sitharam, [2]V.Ganesh, [3]M.Harshitha Goud, [4]Bolgam Pooja
[1,2,3]Assistant Professor, [4]UG Student, [1,2,3,4]Department of Electronics and Communication Engineering,
Kasireddy Narayanareddy College of Engineering and Research, Hyderabad, Telangana

ABSTRACT
The traditional encryption solutions are not entirely secure from an intermediate service like cloud servers because of the privacy leakage of sensitive data. A unique type of encryption method that can address security and privacy concerns is homomorphic encryption. This contains three security steps, namely key creation, encryption, and decryption, as opposed to public key encryption. The design and implementation of highly secure MN-homomorphic encryption on a VLSI platform are done in this research. In comparison to current norms, this system will offer superior security and resource efficiency. Private information and data integrity are both guaranteed by fully homomorphic encryption and decoding. The major goal is to make operations go more quickly. S-Box is first provided with input bits and a key. Then, bits are replaced using S-Box. After shifting operation is performed to the substituted bits. Now these bits are encrypted using MM homomorphic encryption. Hence MM homomorphic encryption better security compared to exist one.
Key Words: Homomorphic encryption, Large Integer Multiplication, Operand Reduction, VLSI Architecture, S-Box.

INTRODUCTION

The majority of board systems' databases use fully homomorphic encryption (DMBS). The test of confirming and securely storing the legitimate treatment of classified information in the remote database is one of the current problems associated with the use of databases. Cryptography allows for the protection of the privacy of sensitive data. It's possible that using clever encryption techniques to store data in distant databases will significantly lessen how the framework is presented without interpretation. To solve the problem, MIT evaluates the cryptographic system that is on display. The server can perform SUM, AVG, and Count queries on encoded data thanks to the use of additively homomorphic cryptography; the other SQL queries make use of unique encryption calculations with crucial practicality. Theadjustment of completely homomorphic cryptosystem will keep the capacity to perform run of the mill database tasks on encoded information without decoding the information in a confided condition. In any case, such a cryptosystem must fulfill certain prerequisites for practical qualities and computational unpredictability, which is significant. Fully Homomorphic Encryption (FHE) is a huge achievement in cryptographic research in recent years. A FHE plan can be utilized to elective perform calculations on figure content without trading off the substance of relating the plain text [1]. Therefore, a practical FHE plan will open the way to various new security advances and protection related to the applications, for example, security safeguarding pursuit and cloud-based processing. For the most part, FHE can be ordered into three classifications: cross section based, number based, and learning with mistakes. One of the fundamental difficulties in the improvement of FHE applications is to moderate the amazingly high-computational intricacy and asset necessities [2]. For instance, programming usage of FHE in superior PCs still expend the critical calculation time, especially to achieve thevast whole number duplication which more often than not includes more than countless bits. For cross section based FHE, bitincrease the required for the little setting with a grid measurement. To quicken the FHE tasks, different effective plans have been proposed to

handle the extensive wholenumber duplication. The objective of this paper is to revive the encryption natives in entire number based FHE using FPGA advancement. This particular FHE count is picked because of the less unpredictable theory, humbler key size and equivalent execution. Also, the introduction of a grouped FHE plots over the entire numbers ensures further capability upgrades. Augmentation is a key segment in these FHE plans the features in the encryption, unscrambling and evaluation steps. Broad entire number FFT duplication has furthermore been used in the late of referenced gear and GPU use of other FHE plans. Future work will look into the impact of the gear multiplier on substitute walks inside the FHE plot. Specifically, presenting the primary gear execution of encryption rough required for FHE over the numbers.

ULLY homomorphic encryption (FHE) allows computations to be carried out directly on ciphertexts for ensuring data privacy on untrusted servers, thus attracting much attention for cloud computing applications. Generally, FHE can be classified into three categories: lattice-based, integer based [3], and (ring) learning with errors. One of the main challenges in the development of practical FHE applications is to mitigate the extremely high- computational complexity and resource requirements. For example, software implementations of FHE in high- performance computers [4], [5] still consume significant computation time, particularly for accomplishing large integer multiplication which usually involves more than hundreds of thousands of bits. For lattice-based FHE, 785 006- bit multiplication is required for the smallsetting with a lattice dimension of 2048.

RELATED WORK

In Gentry's game plan to go from to some degree homomorphic encryption plan to an absolutely homomorphic encryption plot is utilized bootstrapping. Precisely when an understand substance goes to be pointlessly monster or excessively clamorous expansions is illegal, the encoder can utilize the some degree homomorphic encryption plan to assess as far as possible on the figure content, utilizing the blended private key that is a touch of open key. So this encryption strategy encodes plaintext once more, that isn't so great deal of uproarious yet rather dynamically irrelevant. So as to stay an astounding course of action, it is basic to nearly homomorphic plan could safely scramble your private key and attest the precision as far as possible. For this the fairly homomorphic cryptosystem necessities are safely scramble its private key and arranged for assessing the unscramble work. In this way the Gentry utilizes squashing of the disentangling that awards get unscramble fill in the breaking point that to some degree cryptosystem can homomorphically assess. High society's homomorphic encryption plot is dependent on the perfect cross fragments and two assignments must be process able over the rings for homomorphicity these activities. The deterrents of Gentry's absolutely homomorphic encryption plot is nonsensical (it understands a tiny bit at a time and keys and figure structures is colossal), reality it depends upon the new and unassumingly untested cryptographic local people. One year after the age of first absolutely homomorphic encryption plot Dijk, Gentry, Halevi, Vaikuntanathan proposed absolutely homomorphic encryption conspire that utilizing basic isolated number juggling (it works over the Integers) and utilize Gentry's methodologies to change over to some degree homomorphic cryptosystem to absolutely homomorphic encryption plot. Security is the most significant issue in the communication networks to protect private data of every individual. Many types of cryptography techniques have been proposed each one is remarkably suitable for specific applications. Another type of cryptography technique discussed earlier is hash functions do not use any keys for performing encryption of data. It is not suitable for the applications where security is primitive. The purpose of security can be obtained using public key encryption because it uses two keys for scrambling and unravelling. One key is used for validation of the user and other is used for deciphering of text.The intersecting network sender initiates transfer of data. The public key is used to verify the message whether it is encoded or not. In case of unscrambled message, it stops sending to the other client. The protection of data deserves some changes to data. The public key encryption is responsible for secured transmission, user authentication, traffic checking, non-repudiation and investigation of unauthorised users.

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 51, Issue 8, August : 2022

Encryption using computers is most powerful technique among many discovered algorithms on data systems. A cryptography algorithm is said to be the most powerful algorithm only when it has evident proof of adverse attack and essential changes it have been made to act against that kind of attacks. A method introduce for providing utmost security is key schedule. Inthis schedule, different keys are extractedfrom private key, which are used for encryption in each round the order to conceal information from interpreting and changing. The computation of different keysfor various stages can be done using computers. There is a chance of disclosing data by trespassers to other association who may reveal mystery data or alter it accordingto their wish. When we want to send data using particular encryption technique we should first aware of total structure used init. Then only we can block intruders from attacking our info systems. Cryptography gives assurance the data could not beinterpreted or analyzed by any unauthorized persons except the user destined for that. It has the ability to block the trespasser from striking data which is protected.

EXISTED SYSTEM

The below figure (1) shows the architecture of existed system. In this system mainly, two NTT units, a controller unit, an AGU, and several memory units are used. ROM main intent is to store the twiddle factors. There are mainly two single ports of SRAM in NTT block. Here firstly two inputs are computed at same time by using the twoNTT data there are NTT1 and NTT2. For thepurpose of multiplication the NTT is used asinverse NTT and because of R input data is processed. Addition and subtraction operations areperformed in the Mul Mod unit. The result of this unit is processed to the buffer unit. Now the values are saved in ROM. Herepoint wise multiplication process is performed in the NTT block and bits arecomputed depends on the current status of operation.

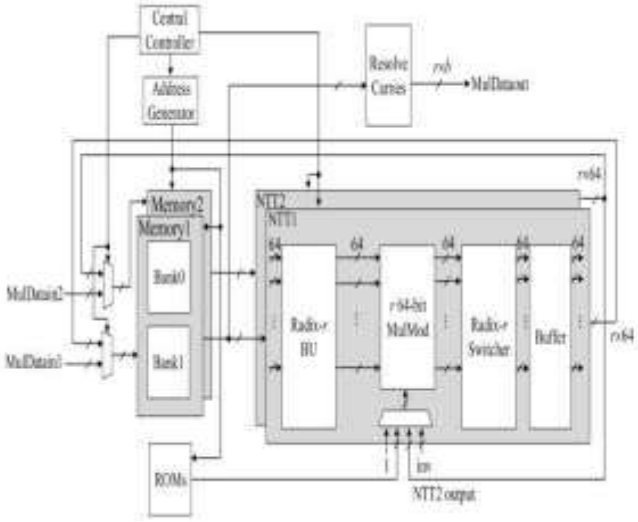

Fig. 1: Existed System

To relocate the data radix r is used and this will saves the memory temporarily. Basically there are four pipelined stages in the MulMod unit. To get conflict freeaddress in the system buffer is used. But thissystem does not give effective results in terms of delay and time. Hence to overcome this, a new system is introduced which is discussed in below section.

PROPOSED SYSTEM

The below figure (2) shows the blockdiagram of proposed system. This systemwill provide better security and resource efficiency compared to existing standards. fully homomorphic encryption and decryption technique guarantee both privacyand integrity. The main intent is to increase the speed of operation. Initally, input bitsand

key is given to S-Box. Next, bits are subtitited using S-Box. After NTT is appliedto the substituted bits. Now these bits are encrypted using fully homomorphic encryption. Similarly, decryption process is performed in reverse operation. The description of each block is given in detail manner.

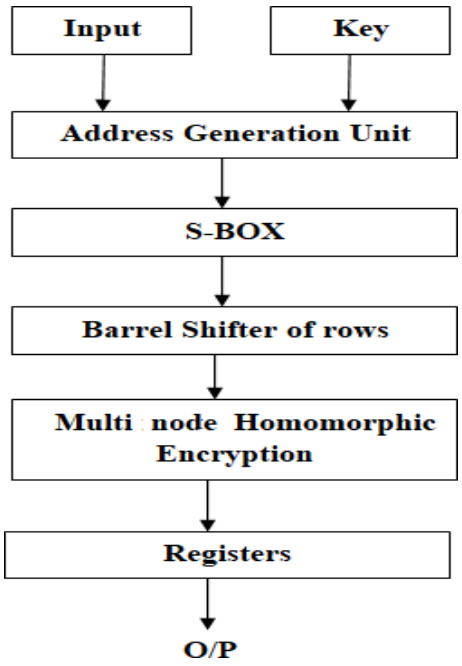

Fig. 2: PROPOSED SYSTEM

SUBSTITUTE BYTESTRANSFORMATION (S-BOX)

The modified structure starts with changes in the Sub bytes step. The function of this step is to substitute data present in the S-boxmemory unit within the state by diverse datapresent in other memory unit. The dispersion of data in memory units creates the confusion. The main purpose of this Shannon's contents for scientific restraint arrangement is to stimulate security. The basic purpose of substitution of bytes is to secure information.

ENCRYPTION

Encryption algorithm is a combination of complex mathematical functions which are used to encrypt the confidential information. Encryption key is a secret values that the sender utilizes as one of the inputs to the encryption algorithm in conjunction with plain text to generate a cipher text.

Shift Rows Transformation

Shift row transformation is followed by substitution of bytes step. This step works on shifting of bytes present in each row. Commonly the shifting done either to left side or right side. The shifting employed in the row transformation is circular shift. In this step first row is moved one byte to the left side as it is circular shift the left most byte comes right side of the row. In the sameway the second row is moved two byte positions left and third row shifts three positions left. Consequently the size of the output state matrix of this step does not change but, the byte positions will change.

RESULTS

The below figure (3) & (4) shows the RTL schematic and technology schematic of proposed system.

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 51, Issue 8, August : 2022

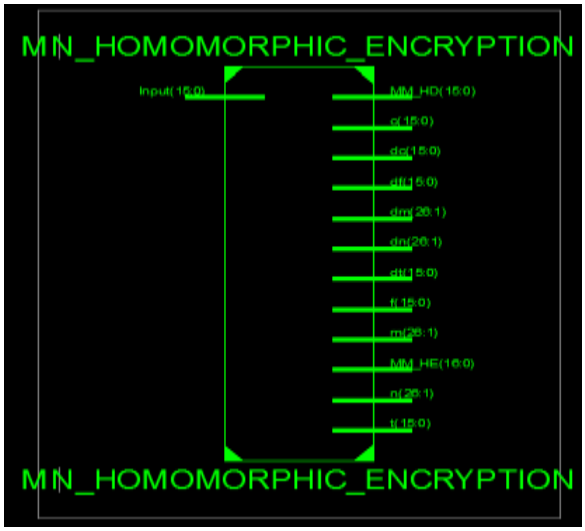

Fig. 3: RTL SCHEMATIC OF PROPOSEDSYSTEM

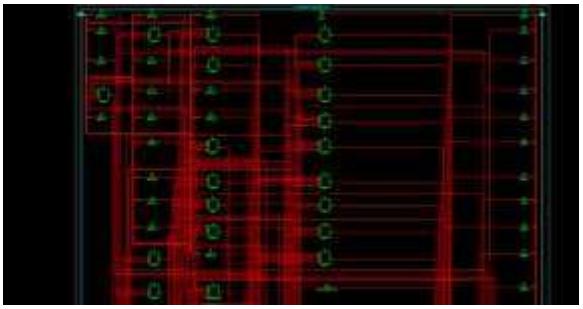

Fig. 4: TECHNOLOGY SCHEMATIC OFPROPOSED SYSTEM

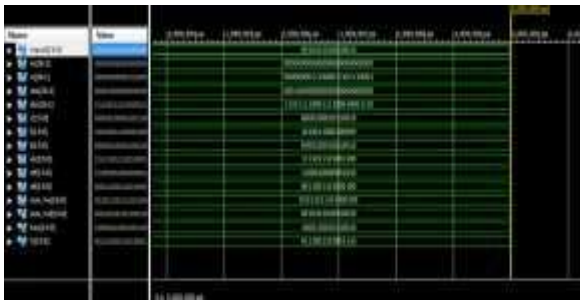

Fig. 5: OUTPUT WAVEFORM OF PROPOSEDSYSTEM

S.NO	Parameters	Existing System	Proposed System
1	Memory Used	3698723 Kilo Bytes	3702060 Kilo Bytes
2	Total Delay	4.103 ns	3.806 ns
3	Logic Delay	0.210 ns	0.301 ns
4	Route Delay	3.893 ns	3.505 ns

Table. 1: Comparison Table

CONCLUSION

In this design and implementation of high secure VLSI based MN homomorphic encryption was implemented. The proposed system was synthesized with an estimated core area. MM homomorphic encryption performs the operation depend on the homomorphic conditions. The public and private key will shift the bits in single clock cycle. From Experimental results it can observe that the proposed system is faster than CPU and provides security in efficient way.

REFERENCES

1. Jheng-Hao Ye and Ming-Der Shieh, "Low-Complexity VLSI Design of Large Integer Multipliers for Fully Homomorphic Encryption", 1063-8210 © 2018 IEEE.
2. S. Koteshwara and A. Das, "Comparative study of authenticated encryption targeting lightweight IoT applications," IEEE Design Test, vol. 34, no.4, pp. 26–33, Aug. 2017.
3. C. Dobraunig, M. Eichlseder, S. Mangard, F. Mendel, and T. Unterluggauer, "ISAP–towards side-channel secureauthenticated encryption," IACR Trans. Symmetric Cryptol., vol. 2017, no. 1, pp.80–105, 2017.
4. H. Böck, A. Zauner, S. Devlin, J. Somorovsky, and P. Jovanovic, "Nonce- disrespecting adversaries: Practical forgeryattacks on GCM in TLS," in Proc. USENIX WOOT, 2016, pp. 1–11.
5. P. G. Lopez et al., "Edge-centric computing: Vision and challenges," ACMSIGCOMM Comput. Commun. Rev., vol.45, no. 5, pp. 37–42, Oct. 2015
6. F. Abed, C. Forler, and S. Lucks, "General overview of the firstround CAESAR candidates for authenticated encryption," IACR Cryptol. ePrint, Tech. Rep. 2014/792, 2014.
7. Nitesh Aggarwal, Cp Gupta, and Iti Sharma. 2014. Fully Homomorphic symmetric scheme without boot strapping. In Cloud Computing and Internet of Things (CCIOT), 2014 International Conference on.IEEE, 14–17.
8. S Sobitha Ahila and KL Shunmuganathan. 2014. State Of Art in Homomorphic Encryption Schemes. International Journal of Engineering Research and Applications 4, 2 (2014), 37– 43.
9. D. McGrew and D. Bailey, AES-CCM Cipher Suites for Transport Layer Security (TLS), document RFC 6655, 2012.
10. H. Handschuh and B. Preneel, "Key- recovery attacks on universal hash function based MAC algorithms," in Proc. Annu. Int. Cryptol. Conf. Berlin, Germany: Springer,2008, pp. 144–161.