ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

Double Key Encryption RSA Algorithm Using FFT in Homomorphic Methodology

Pottendla Anjali Devi
PG Scholar,
Dept of ECE,
Global College of Engineering & Technology,
Kadapa, YSR (Dt), AP, India.

S K Imam Basha
Assistant Professor,
Dept of ECE,
Global College of Engineering & Technology,
Kadapa, YSR (Dt), AP, India.

ABSTRACT:

Fully homomorphic encryption (FHE) can perform arbitrarily on the given user data using a secondary key which is unknown to the user but the algorithm contains a large number of multipliers which is a bottle neck for the vlsi design .so the large number of multipliers are replaced with the Fast Fourier transform without any loss in performance of the algorithm and low complexity in hardware design .the paper focus on the hardware design for the existing schemes to make the existing FHE schemes more efficient and practical towards real-life applications. The integer-FFT multiplication algorithm is adopted for the implementation of Gentry-Halevi's FHE scheme. As the Moore law continues driving the computer technology, the key size of the Rivest-Shamir-Adelman (RSA) encryption is necessary to be upgraded

KEYWORDS:

Fully homomorphic encryption, Fast Fourier transform, RSA algorithm.

I.INTRODUCTION:

Fully homomorphic encryption can be considered as ring homomorphism. In mathematics, a ring is a set R equipped with two operations + and * satisfying the following eight axioms, called the ring axioms. R is an abelian group under addition. Craig Gentry [6, 7], using lattice-based cryptography, showed the first fully homomorphic encryption scheme as announced by IBM on 25 June 2009. His scheme supports evaluations of arbitrary depth circuits. His construction starts from a somewhat homomorphic encryption scheme using ideal latticesthat is limited to evaluating low-degree polynomials over encrypted data. It is limited because each cipher text is noisy in some sense, and this noise grows as one adds and multiplies ciphertexts, until ultimately the noise makes the resulting ciphertext indecipherable.

He then shows how to modify this scheme to make it bootstrappable—in particular, he shows that by modifying the somewhat homomorphic scheme slightly, it can actually evaluate its own decryption circuit, a self-referential property. Finally, he shows that any bootstrappable somewhat homomorphic encryption scheme can be converted into a fully homomorphic encryption through a recursive self-embedding. In the particular case of Gentry's ideal-lattice-based somewhat homomorphic scheme, this bootstrapping procedure effectively "refreshes" the cipher text by reducing its associated noise so that it can be used thereafter in more additions and multiplications without resulting in an indecipherable ciphertext.

II.RELATED WORK:

Previously general-purpose GPU has also been used for acceleration of security algorithms such as elliptic curve cryptography [54]. But the GPU architecture was originally geared for graphics operations and later has been extended for general purpose computations. It is not the most power efficient architecture for a specific algorithm or applications. One approach is to attach an Application Specific Integrated Circuit (ASIC) to the CPU which is dedicated to encryption/decryption operations. At micro-architectural level, it can be implemented as an extension of instruction set. Previously customized ASIC or IP blocks has been designed to accelerate the well-known encryption schemes such as Advanced Encryption Standard (AES) and RSA [1] [2]. Today many embedded processors have AES or RSA cores included. This work is aimed to take a similar approach and to design a specific hardware or IP blocks for accelerating the core computations in FHE. There are some works tackling the problem of hardware acceleration of fully homomorphic encryption. In [55], an FPGA implementation draft for improving the speed of FHE primitives was proposed. However, no implementation results were presented. [56] Presents a first custom hardware architecture supporting

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

encryption, decryption and recryption primitives for the lowest security setting with a dimension 2,048 for the Gentry-Halevi scheme. A number theoretical transform based fast million-bit multiplier is the heart of all the primitives as claimed in [3]. Large integer multiplication is by far the most time consuming operation in the FHE scheme. Therefore we have selected it as the first block for hardware acceleration. Because multiplication is the dominating component of FHE operations, it will be a significant step toward practical application of FHE if a high performance, low power, area efficient, high precision, integer multiplier architecture can be developed. Therefore, our initial attempt is to tackle the design of a large-number multiplier thatcan handle 768K bits, in support of the 2048 dimension FHE scheme demonstrated by Gentry and Halevi.

III.IMPLEMENTATION:

The finite-field FFT and IFFT units are the main components of the large-number multiplier. For high-throughput applications, a pipeline FFT architecture is often used [17]. However, the pipeline design requires a local memory buffer at every stage [17] which often results larger chip area and more power consumption. In contrast, a memory-based inplace FFT architecture stores the intermediate results at the same memory where the input data are read from. As a result, it minimizes the memory usage but still can produces high throughput [15]. Thus, the memory-based strategy is adopted in this design. The memory-based FFT architecture mainly consists of a butterfly unit, a data memory, a ROM storing the twiddle factors, and a control logic unit. The butterfly unit needs to read γ input data from the memory then write back γ output results back to the memory. Hereby, we adopt a conflict memory access approach [15] [16] that partitions the memory into γ banks for concurrent conflict-free read and write access. More specifically, the input data with indexes D = [dn−1, dn−1, ..., d0]γ, where n = logK γ , are stored at address = [dn−1, dn−2, ..., d1]r and bank = (Pn−1 i=0 di) mod r. In this design, γ is 4 and K is 40. In this architecture, we need to design a 1,024-point finite- field FFT processor. For hardware efficiency, we choose to use two stages of 32-point FFTs to implement the 1,024-point FFT. A radix-4 butterfly unit can be used recursively for four times to compute one radix-4 FFT. For 40-point FFT. From equations (1-2), it is obvious that only shift operations and modulo additions are needed to compute the FFT.

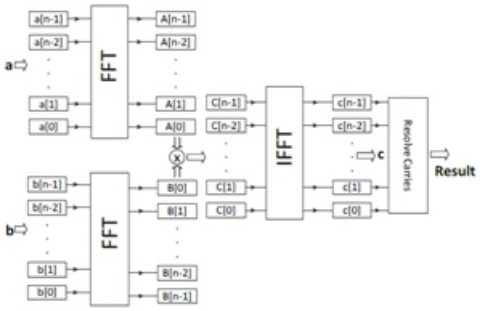

Fig .1 FFT-based multiplication algorithm.

Carry Save Adder:

The limitation of speed of a modern computer in performing the arithmetic operations where addition is the basic units of arithmetic machines leads to slow operating devices. A fast arithmetic unit allows the extension of the application domain of fast addition in digital image processing, signal processing, inversion of matrices; computation of Eigen values, digital filters etc.The adders are important building blocks of any digital processors and their design have been of great concern to engineers and professionals in the area of microprocessors and special signal processors. This is because most of the As a result fast adder designs have been of continued interest in order to reduce the addition time constraints in sophisticated processing applications. Redundant number system may not be convenient for manual computations but they are useful in designing high speed arithmetic machines. CSA number representation systems posses sufficient redundancy to allow the annihilation of carry or borrow chains and hence result in fast, propagation free addition and subtraction. Additionally, Adder designed with CSA number system has a regular layout which is suitable for implementations. The redundant binary representation (RBR) is a numeral system that uses A signed digit number is represented by m+n+1 digits x' Formula having algebraic value as;

$$X_v = \sum_{i=-n}^{m} x_i r^{-i} \qquad (1)$$

Where the value of r and x' are such that the following conditions satisfied: The 'r' radix is a positive integer Algebraic value Formula has unique representation Totally parallel addition and subtraction is possible for all digits in corresponding position of two's representation.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

If Formula, then Formula number representation is known as Redundant Binary Signed Digit Number System.Multiplication requires the addition of several summands. This technique of adding the summands is called carry save addition. In this type of addition, addition of two numbers is done and carry is saved and the carry is added to the final sum. We can understand it easily by an example, suppose the addition of n-bit numbers 'a', 'b', 'c' to produce a sum 'z'. Now first we add 'a' and 'b' to produce an intermediate sum 'u' and now we add 'u' to 'c' to produce a final sum 'z'.

IV.RESULTS

TRANSMITTER:

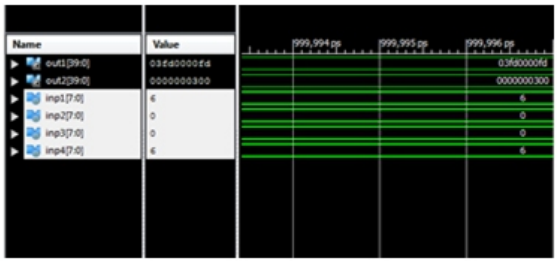

Fig 2 .Transmitter resultant

The above resultant is the four input port of 8 bit each 6 0 0 6 is given as the data and key and the two resultants out1 and out2 are the chipper text with proposed encryption

RECIEVER:

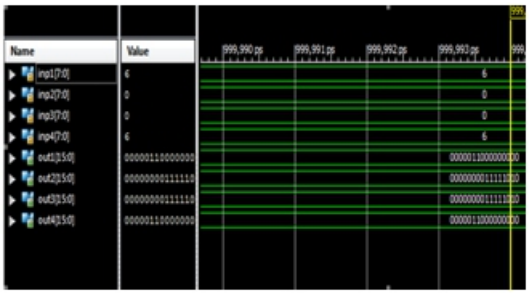

Fig 3 Reciever resultant

The above resultant is the two input port of 16 bit each .the chipper text generated is given as the data and the four resultants out1, out2, out3 and out4 are the original text with proposed decryption.

V.CONCLUSION AND FUTURE SCOPE:

The large-number multiplier is using Strassen's FFT-based multiplication algorithm.

The memory-based, in-place FFT architecture was used for the FFT processor to reduce the memory usage. We use a number of design optimization strategies to improve the performance and reduce the area of the Radix-16 unit. For RADIX FFTs, coding gain is only achieved if it is concatenated with an outer code, such as a TCM code or a Turbo TCM Code. This was mentioned as an ongoing, exciting area of research.

REFERENCES:

[1]J. D. Cohen and M. J. Fischer, "A robust and veriable cryptographically secure election scheme,in FOCS, vol. 85, 1985, pp. 37282.

[2]E.Kushilevitz and R. Ostrovsky, Replication is not needed: Single database, computationally-private information retrieval, in Foundations of Computer Sci- ence, 1997. Proceedings., 38th Annual Symposium on. IEEE, 1997, pp. 364-373.

[3]M. Naehrig, K. Lauter, and V. Vaikuntanathan, Can homomorphic encryption be practical?in Proceedings of the 3rd ACM workshop on Cloud computing security workshop. ACM, 2011, pp. 113-124.

[4]R. L. Rivest, L.Adleman, and M. L. Dertouzos, On data banks and privacy homomorphisms, Foundations of secure computation, vol. 32, no. 4, pp. 169- 178, 1978.

[5]S.Goldwasser and S. Micali, Probabilistic encryption,Journal of computer and system sciences, vol. 28, no. 2, pp. 270-299, 1984.

[6]R. L. Rivest, A. Shamir, and L. Adleman, A method for obtaining digital signatures and public-key cryptosystems, Communications of the ACM, vol. 21, no. 2, pp. 120-126, 1978.

Author's Profile:

Pottendula Anjali Devi, Pursuing M.Tech (VLSI), Global College of Engineering & Technology, Kadapa, Andhra Pradesh, India.

S K Imam Basha, M.Tech, Assistant Professor at Global College of Engineering & Technology, Kadapa, Andhra Pradesh, India.