DESIGN OF OPTIMIZED SYMMETRIC KEY ENCRYPTION ALGORITHM FOR INCREASING SECURITY

ApoorvaHora*

Prof. NamrataSahayam**

Abstract-

Design of optimized symmetric key encryption algorithm for increasing security had been covered in this paper using International Data Encryption Algorithm (IDEA). The current era has seen an explosive growth in communication. Applications like online banking, personal digital assistants, mobile communication, smartcards, etc. have emphasized the need for security in resource constrained environment. International Data Encryption Algorithm (IDEA) cryptography serves as a perfect encryption tool because of its 128 bits key sizes and high security comparable to that of other algorithms. However, to match the ever increasing requirement for speed in today's applications, hardware acceleration of the cryptographic algorithms is a necessity. This study presents an efficient hardware structure for the modulo (2^n+ 1) Multiplier and modulo(2^n) adder which is the most time and space consuming operation in IDEA. The proposed design saves more time, area and cost. The block size considered here is same as of traditional IDEA encryption algorithm which is of 64 bits with 16 bit sub-blocks.

Keywords- IDEA, Crptography, Mobile communication, Modulo (2^n+ 1) Multiplier,Modulo(2^n) adder

* Department of Electronics & Communication Engineering, Jabalpur Engineering College, Jabalpur

** Professor, Department of Electronics & Communication Engineering, Jabalpur Engineering College, Jabalpur

I. INTRODUCTION

Data security is an important issue in today's computer networks. This paper presents the algorithm,which implements a new version of the cryptography[1]. To understand the theory of cryptography one has to understand the knowledge of different threats in the network. Cryptography involves the fulfillment of the security goals in the network which are Data Confidentiality, Integrity, Authentication and Non-Repudiation. Data confidentiality is achieved by means of Cryptography.

The proposed Encryption Standard (PES) is a block cipher introduced by Lai and Massey. It was then improved bythe Lai, Massey and Murphy in 1991. This version, with stronger security against differential analysis and truncated differentials, was called the Improved PES (IPES). IPES was renamed to be the International Data Encryption Algorithm (IDEA) in 1992. Claims have been made that the algorithm is the most secure block encryption algorithm in the public domain.

IDEA was to develop a strong encryption algorithm, which would replace the DES procedure developed in the U.S.A. in the seventies. It is also interesting in that it entirely avoids the use of any lookup tables or S-boxes. When the famous PGP email and file encryption product was designed by Phil Zimmermann, the developers were looking for maximum security. IDEA was their first choice for data encryption based on its proven design and its great reputation.

The IDEA encryption algorithm

- provides high level security not based on keeping the algorithm a secret, but rather upon ignorance of the secret key.
- is fully specified and easily understood.
- is available to everybody.
- is suitable for use in a wide range of applications.
- can be economically implemented in electronic components .
- can be used efficiently.
- may be exported worldwide.
- is patent protected to prevent fraud and piracy.

In this paper, the cipherused is a symmetric key block cipher. It takes input as 64 bit plain text and gives a 64 bit cipher text as output using a 128 bit key. While working on plain text, it divides the input data in to 16 bit sub-blocks and operates on each block. It is described as one of the more secure block algorithm due to its high immunity to attacks. In spite of the fact thatData Encryption standard (DES) is another popular symmetric block cipher which is used in several financial and business applications and its drawback is the short key word length.Moreover unlike DES,IDEA doesn't need any S-box or P-box is required for implementing this cipher. The most crucial module part of this algorithm is the design of the multiplier modulo a Fermat prime, which is one of the algebraic group operation used and entire speed of IDEA depends on this module. So designing the multiplier is a major during the hardware or software implementation of IDEA because its speed is a big issue when hardware implemented IDEA is used in real time application. The overall objective for hardware implementation of IDEA is to minimize the hardware requirements which result in efficient use of area and at the same time improve the processing speed and high throughput of data. As the performance of IDEA cipher depends entirely on the modulo(2^n+1) multiplier design, the main objective is to design an efficient andfast modulo multiplier which is to be used in the entire IDEA algorithm.

The organization of the rest of the paper is as follows. The previous hardware and software implementations are covered in section II. Section III describes the IDEA cipher and its detailed operations as well as modules. Section IV describes the general architecture of the cryptosystem to be implemented and the proposed modulo multiplier architecture. Section V discusses the performance reviews and comparisons with previous schemes and section VI finally concludes the paper.

II. PREVIOUS WORK

In spite of the fact that IDEA works with 16 bit word blocks, software implemented IDEA cannot reach the speed that is required for online encryption in high speed networks. IDEA was implemented in software by Ascom, the patent holder of IDEA, and it achieved an encryption rate of 23.53 Mbps.It has been discussed as:

(a) HarivansPratapSingh[2] have proposed There are many security algorithms that are used for security purpose. IDEA is one of them. The block cipher IDEA operates with 64-bit plaintext and cipher text blocks and is controlled by a 128-bit key. The fundamental innovation in the design of this algorithm is the use of operations from three different algebraic groups. The algorithmstructure has been chosen such that, with the exception that different key sub-blocks are used, the encryption process isidentical to the decryption process. The drawback of IDEA is that the large numbers of weak keys were found in IDEA(International Data Encryption Algorithm). the size of the key has been increased from 128 bits to 256 bits. This increased key size will increase the complexity of the algorithm.

(b) Modugu.R et al. [3] have discussed Cryptographic algorithms such as International Data Encryption Algorithm (IDEA). Which have found various applications in secure transmission of the data in networked instrumentation and distributed measurement systems. Modulo 2n +1 multiplier and squarer play a pivotal role in the implementation of such crypto-algorithms. In this work, an efficient hardware design of the IDEA (International Data Encryption Algorithm) using novel modulo 2n + 1 multiplier and squarer as the basic modules is proposed for faster, smaller and low-power IDEA hardware circuits.

Many security algorithms were proposed regarding security purpose. IDEA uses 128 bit key; hence for decoding the transmitting data one has to achieve 2^{128} possible combination.
The task is to remove the linear intrusion attack.

III. THE IDEA ALGORITHM

International Data Encryption Algorithm (IDEA) is a block cipher designed by Xuejia Lai and James L. Massey of ETH-Zürich and was first described in 1991. It is a minor revision of an earlier cipher, PES (Proposed Encryption Standard); IDEA was originally called IPES (Improved PES). IDEA was used as the symmetric cipher in early versions of the Pretty Good Privacy cryptosystem. IDEA is a symmetric, secret-key block cipher. The keys for both encryption and decryption must be kept secret from unauthorized persons. Since the two keys are symmetric, one can divide the decryption key from the encryption one or vice versa. The size of the key is fixed to be 128 bits and the size of the data block which can be handled in one

encryption/decryption process is fixed to 64 bits.

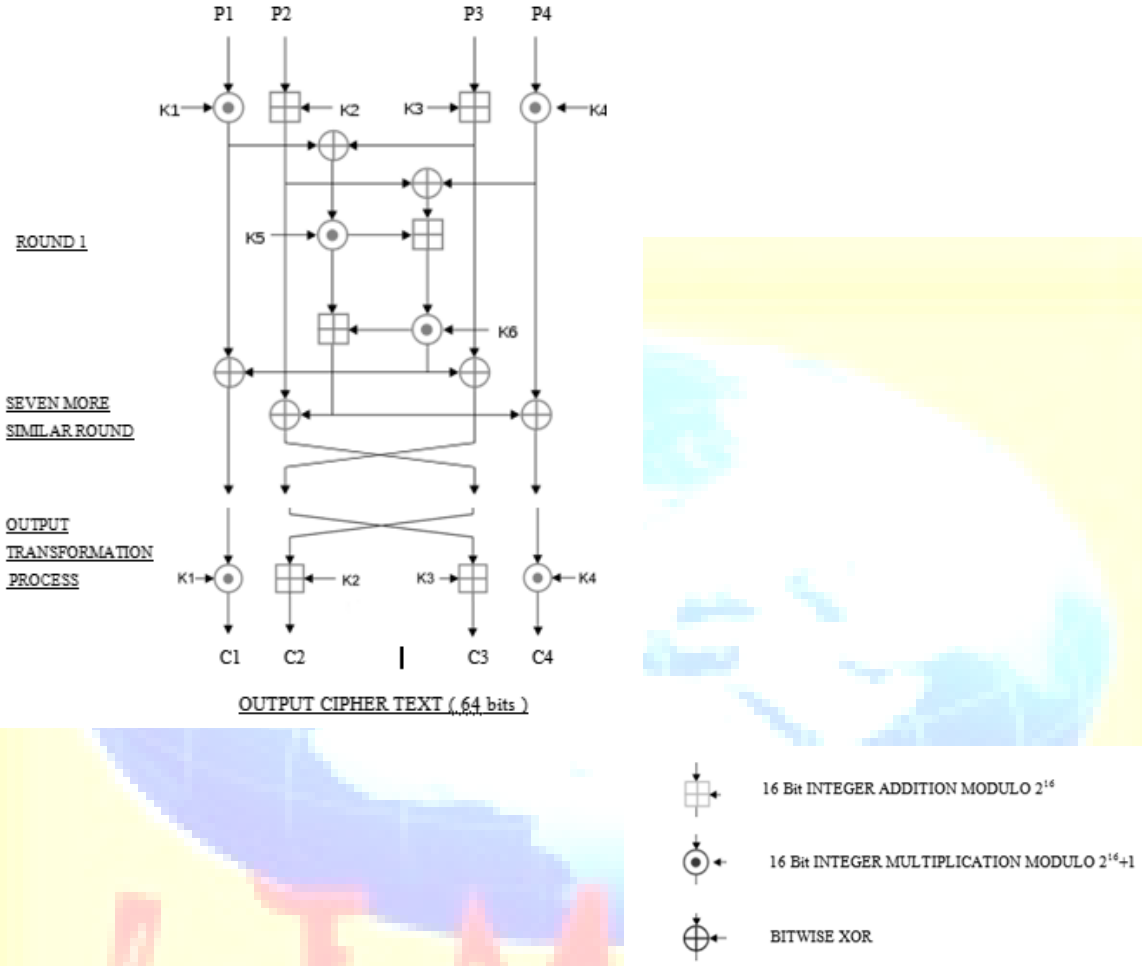

Figure1. Basic structure of IDEA Algorithm

All data operations in the IDEA cipher are in 16-bit unsigned integers. When processing data which is not an integer multiple of 64-bit block, padding is required. The security of IDEA algorithm is based on the mixing of three different kinds of algebraic operations: EX-OR, addition and modular multiplication. IDEA is based upon a basic function, which is iterated eight times. The first iteration operates on the input 64-bit plain text block and the successive iterations operate on the 64-bit block from the previous iteration. After the last iteration, a final transform step produces the 64-bit cipher block. The decryption phase of IDEA is identical to that of the

encryption phase. It uses the same sequence of operations as in the encryption phase. The only change is that the sub-keys are reversed and are slightly different. That means the sub-keys which are used in round 1 during encryption phase are manipulated during last round of decryption phase. The subkeys used in decryption are either additive or multiplicative inverse of the sub-keys used in the encryption phase.

A. Key Generation

The 64-bit plaintext block is partitioned into four 16-bit sub-blocks, since all the algebraic operations used in the encryption process operate on 16-bit numbers. Another process produces for each of the encryption rounds, six 16-bit key sub-blocks from the 128-bit key. Since a further four 16-bit key-sub-blocks are required for the subsequent output transformation, a total of 52(= 8 x 6 + 4) different 16-bit sub-blocks have to be generated from the 128-bit key.

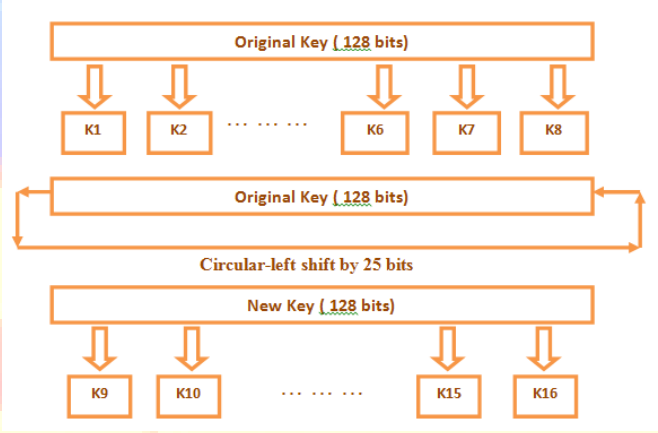

Figure 2. IDEA Algorithm Key generation

The 52 16-bit key sub-blocks which are generated from the 128-bit key are produced as follows:

- First, the 128-bit key is partitioned into eight 16-bit sub-blocks which are then directly used as the first eight key sub-blocks.

- The 128-bit key is then cyclically shifted to the left by 25 positions, after which the resulting 128-bit block is again partitioned into eight 16-bit sub-blocks to be directly used as the next eight key sub-blocks.
- The cyclic shift procedure described above is repeated until all of the required 52 16-bit key sub-blocks have been generated.

IV. DESIGN AND IMPLEMENTATION

The main objective of implementing a cryptosystem is to increase the data encryption and decryption rate so as to increase the throughput. MATLAB implementations of cryptosystem offer adequate speed so that it can be easily used in real time applications. To implement an algorithm in hardware, we have to first realize the architecture of the entire system. The general architecture of hardware implementation of a cryptosystem is shown in Figure 3.

A. Hardware Implementation of IDEA

The IDEA cipher has three separate units other than key generation module. The performance and speed of hardwareimplemented IDEA depends on these three modules. These modules are:

- Multiplier Module.
- Addition unit.
- Ex-or logic.

In each round of the 8 rounds of algorithm, the following sequences of events are performed:

1. Multiply* P1 and K1
2. Add* P2 and K2
3. Add* P3 and K3
4. Multiply* P4 and K4
5. XOR the results of step 1 and step 3
6. XOR the results of step 2 and step 4
7. Multiply* the results of step 5 with K5
8. Add* the results of step 6 and step 7

9. Multiply* the results of step 8 with K6
10. Add* the results of step 7 and step 9
11. XOR the results of step 1 and step 9
12. XOR the results of step 3 and step 9
13. XOR the results of step 2 and step 10
14. XOR the results of step 4 and step 10

Sequence of events followed in the output transformation round:-

1. Multiply* R1 and K1
2. Add* R2 and K2
3. Add* R3 and K3
4. Multiply* R4 and K4

Among these modules, the main component whichcontrols the speed and performance of IDEA is the modulo(2^n + 1) multiplier module. It consumes the major portionof the clock cycles required by the entire algorithm. Themodulus used in the multiplication is a Fermat prime which is (2^n + 1). One important thing here is that theoperand 0 is treated as 2^n. The implementation of thismultiplier in hardware is the most difficult task becausethe word length of the operands is comparatively large andimplementing the multiplication sequentially is really timeconsuming.

B. Multiplication Modulo (2^n +1)

Multiplication modulo a Fermat prime (p) is used as an important operation in many algorithms and it is crucial in various applications like pseudorandom number generation,Arithmetic processing and Cryptography. In IDEA algorithm, this modulo multiplier plays a very important role in the throughput and speed. In general, amodulo multiplier consists of two stages, Multiplier module and modulo reduction. This paper mainly deals with the various multiplication schemes that have been implemented along with some newly proposed schemes.

The multiplier module is the stage where two binary n bit numbers are multiplied to form a product of 2n bits. Themodulo reduction stage produce the product modulo the Fermatprime

number and the final output becomes an n bit number.Various implementations have been done on this multipliermodule so as to improve the efficiency of the cryptosystem.The most crucial part of multiplying two binary numbers is thegeneration of partial products. A lot of problems arise whentwo numbers are multiplied in a straightforward approach inhardware. As per human nature of calculation, when anyexpression is given as

$$xy \bmod(2^n + 1) = z' \bmod(2^n + 1) = z$$

At first the product is calculated by traditional method and thenthe modulo reduction is done by iterative subtraction methoduntil the value falls under the range of 0 and 2 .But thedrawbacks of this implementation is inefficient use of area which increases the hardware cost and it is timeconsuming.

C. Proposed design

There are several methods existing for designing the idea. Here we are presenting a new kind of MATLAB design which is highly optimized as compared to previous one. The description for proposed design are as follows. In general if we multiply the two n bit number ,then we get 2n bit number. Store this 2n bit number(result) in to temporary register ie.t1,then make the length of modulo(2^n+1) equal to the length of 2n bit number ie.t1, by consented zeroes('0's) after the LSB bit of the modulo(2^n+1) and store this value into t2. Here we define the MATLAB multiplier which is easy to understand.

Let the 64 bit data be-

Data4=3345,

Data3=5112,

Data2=9341,

Data1=7245,

In IDEA the 64 bit data is divided into four 16 bit plaintext data. Let the 128 bit key be-

Key8=9345,

Key7=9678,

Key6=9345,

Key5=5345,

Key4=9331,

Key3=1821,

Key2=3029,

Key1=5521,

The 128 bit key is divided into eight 16 bit key resulting in cipher generation as follws-

c1=FFFF,

c2=22C1,

c3=C6F0,

c4=FFFF.

The cipher generated is- FFFFC6F022C1FFFF.

V. RESULTS AND OBSERVATIONS

The performance parameter which is to be accountedfor implementing IDEA in hardware is:

- **Area Requirements**: It can be reported either in termsof space required for each round of IDEA. The area consumed was reduced as compare to previous work.

The profile summary of MATLAB code is-

Function Name	Calls	Total Time	Self Time*	Total Time Plot (dark band = self time)
idea	1	0.116 s	0.021 s	
roundidea	8	0.043 s	0.011 s	
hex2dec	136	0.042 s	0.036 s	
keygen	6	0.028 s	0.003 s	
dec2hex	84	0.025 s	0.025 s	
subroundidea	1	0.010 s	0.002 s	
dec2bin	48	0.006 s	0.006 s	
circshift	6	0.005 s	0.003 s	
iscellstr	136	0.004 s	0.004 s	
fliplr	136	0.002 s	0.002 s	
circshift>ParseInputs	6	0.002 s	0.002 s	
facto	12	0.001 s	0.001 s	

VI. CONCLUSION

RSA Security goes on to say that IDEA was analyzed to measure its strength against differential cryptanalysis. The analysis concluded that IDEA is immune to that technique. In fact, there are no linear cryptanalytic attacks on IDEA, and there are no known algebraic weaknesses in IDEA. The only weakness of note was discovered by Daemen: using any of a class of 2^{56} weak keys during encryption results in easy detection and recovery of the key. However, since there are 2^{128} possible keys, this result has no impact on the practical security of the cipher for encryption provided the encryption keys are chosen at random. IDEA is generally considered to be a very secure cipher and both the cipher development and its theoretical basis have been openly and widely discussed.After implementing on MATLAB it has been observed that the required area was reduced.

REFERENCES

[1] ApoorvaHora, NamrataSahayam, "Implementation Of Efficient Probabilistic Symmetric Key Encryption Method"Asian Journal of Current Engineering and Maths 4:3 May - June (2015) Page 33 – 36.

[2] HarivansPratap Singh, ShwetaVerma, Shailendra Mishra, "Secure-International Data Encryption Algorithm" International Journal of Advanced Research in lectrical, Electronics and Instrumentation Engineering Vol. 2, Issue 2, February 2013,

[3] Modugu.R, Yong-Bin Kim, MinsuChoi,"Design and performance measurement of efficient IDEA crypto-hardware using novel modular arithmetic components", Instrumentation and Measurement Technology Conference (I2MTC), 2010 IEEE, 3-6 May2010,pp1222-1227.

[4] HarivansPratap Singh, SwetaVerma, Shailender Mishra, "Design Implementation of IDEA to S-IDEA", International Journal of Engineering and Innovative Technology (IJEIT) Volume 1, Issue 6, June 2012.

[5] SomayehTimarchi, KeivanNavi, "Improved Modulo 2^n+1 Adder Design", International Journal of Computer and Information Engineering 2:7 2008.

[6] X.Lai and J.L Massey "A Proposal for a New Block Encryption Standard," in advances in Cryptology – EUROCRYPT 90,Berlia,Germany: Springer Verlag pp. 389-404, 1990.

[7] AnttiH¨am¨al¨ainen, MattiTommiska, and JormaSkytt¨, "6.78 Gigabits per Second Implementation of the IDEA Cryptographic Algorithm",2002 Springer-Verlag, pages 760-769.

[8] M.P. Leong, O.Y.H. Cheung, K.H.Tsoi and P.H.W.Leong "ABit Serial Implementation of the International Data Encryption Algorithm IDEA" ©IEEE 2000.

[9] P. Kitsos , N. Sklavos, M.D. Galanis, O. Koufopavlou , "64 Bit Blockciphers: Hardware Implementations and Comparison analysis",593-604,3rd November,2004,Elsevier

[10] R. Zimmermann, A. Curiger, H. Bonnenberg, H. Kaeslin, N. Felber, and W. Fichtner,"A 177mb/s VLSI implementation of the international data encryption algorithm,"IEEE Journal of Solid-State Circuits, Vol. 29, 1994, pp. 303-307.

[11] Rahul Ranjan and I. Poonguzhali, "VLSI Implementation of IDEA Encryption Algorithm", Mobile and Pervasive Computing (CoMPC–2008).

[12] Thaduri,M.,Yoo,S. and Gaede,R, " An Efficient Implementation ofIDEA encryption algorithm using VHDL", ©2004 Elsevier.

[13] Allen Michalski1, Kris Gaj, Tarek El-Ghazawi, "An Implementation Comparison of an IDEA Encryption Cryptosystemon Two General-Purpose Reconfigurable Computers"

[14] SarangDharmapurikar and John Lockwood, "Fast and Scalable Pattern Matching for Network Intrusion Detection Systems"IEEE Journal on Selected Areas in Communications: Oct. 2006, Volume: 24, pp. 1781- 1792 .

[15] Chiranth E, Chakravarthy H.V.A, Nagamohanareddy P, Umesh T.H, Chethan Kumar M., "Implementation of RSA Cryptosystem Using Verilog" International Journal of Scientific &Engineering Research Volume 2, Issue 5, May-2011.

[16] RajashekharModugu, Yong-Bin Kim and Minsu Choi, "A Fast Low-Power Modulo 2n + 1 Multiplier", Journal of IET Computers & Digital Techniques Jan-2011.