An Efficient Implementation of Montgomery Multiplication

Manasadevi R N[1] Ravindra P Rajput[2]
[1]M.Tech. Student [2]Associate Professor
[1,2]Department of Electronics & Communication Engineering
[1,2]UBDTCE Davangere

Abstract— To achieve information security in communication systems a method called Cryptography has been used. There are several methods in the cryptography and one among them is RSA algorithm. In this algorithm modular multiplication is used, which is the most critical part to implement either through software or hardware. Therefore, we proposed a1new method 0 for modular 7 multiplication called as Montgomery5multiplication. Simulation results show the reduction in the number of logic gates and improvement in the performance compared to modular multiplication.

Key words: Encryption, Decryption, RSA, Montgomery Multiplication

I. INTRODUCTION

With the development of communication, the security of the data or information which is being communicated between the sender and the intended receiver is necessary. In communication systems, the security means the information transmitted should be secured from the hackers or the unknown people or attackers. To secure the data from the hackers a method called Cryptography is developed. Montgomery multiplication is one of the modular multiplication method which is used in cryptography for encryption and decryption purpose. When implementing RSA encryption and decryption on hardware, this multiplication reduces the hardware size. Not only in RSA, but also in Elliptic Curve Cryptography (ECC) it is used.

A. Objectives of Cryptography

1) Confidentiality: Information cannot be hacked by other persons except the one for whom it is intended.
2) Integrity: Information cannot be changed during the communication of the encrypted message between sender and receiver.
3) Non-repudiation: Since the RSA provides digital signature facility, later the sender cannot deny the message which is sent by him.
4) Authentication: The sender and receiver must confirm each other's identity for the origin and destination of the information.

B. Applications of Encryption

1) Essential in e-commerce
2) Communications / Network processors
3) Smartcards / Digital cash
4) Military

Depending on the key used the methods are divided into 2 classes. They are:

- Symmetric cryptosystems.
- Public key cryptosystems.

1) Symmetric Cryptosystem

Block diagram for Symmetric cryptosystems is shown in the figure 1. In this method a single key is used for Encryption and Decryption. The source A uses encryption algorithm and a key to convert the message M to Ciphertext C and then transmit that C to the communication channel. The destination B uses decryption algorithm to convert the received Ciphertext C from the channel back to its original plain text format M. The decryption algorithm is completely reverse operation to encryption algorithm, but the same key is used in both the algorithms[1]. The problem with this method is to keep the key secret.

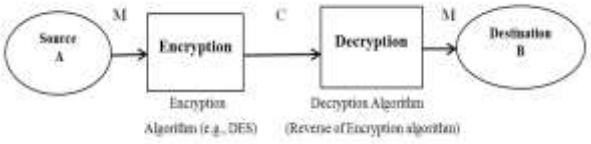

Fig. 1: Block diagram for Symmetric cryptosystems

2) Public Key Cryptosystem

Block diagram for Public Key cryptosystems is shown in the figure 2. In this method two different keys are used. One for Encryption and the other for Decryption. Each user will generate a pair of keys used for encryption and decryption of the information M. One of the two keys is published and it is called as public key, which can be accessed by the intended people. The other key is kept secret and it is called as private key, which is used by the generated party itself. The source A uses encryption key and encryption algorithm to convert the message M to Ciphertext C and then transmit that C to the channel[2]. The destination B uses decryption key and decryption algorithm to convert the received Ciphertext C from the channel back to its original plain text format M. The decryption algorithm is completely reverse operation to encryption algorithm, and the key used in decryption algorithm is different from the encryption key.

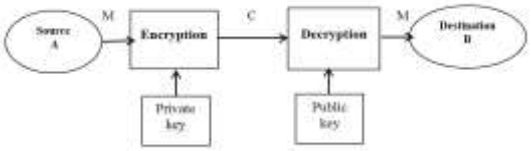

Fig. 2: Block diagram for Public key cryptosystems

II. RELATED WORK

Gaurav R Patil proposed a survey on various approaches applied on standard RSA algorithm in order to enhance it. The main disadvantage of RSA is its computation time; so many researchers applied various techniques to enhance the speed of an RSA algorithm [3]. B.Persis Urbana Ivy proposed a method to secure data or information by a modified n. It provided more efficiency and reliability over the networks [4]. Rajan.S.Jamgekar proposed a paper for secure RSA file transmission. Here some loopholes of RSA prevent a hacker from stealing and misuse of data [5].

A. RSA Algorithm

RSA algorithm was invented by Rivest, Shamir, Adleman in the year 1978. It is the most widely used security system.

This is the best method for Encryption and if the security of digital data is of importance. RSA provides secrecy and authenticity in email, credit card payments etc. As we can send the signed letter in the mail, emails can also be signed by using digital signatures[6]. RSA also provides data integrity and non-repudiation.

B. Key generation process in RSA algorithm

1) Randomly select any two large prime numbers: p and q
2) Compute n = p*q
3) Compute : phi(n) = (p-1)*(q-1)
4) Choose e, which is relatively prime to phi(n) i.e., gcd(phi(n), e) and which in the range 1 < e < phi(n).
5) Find d, so that d is multiplicative inverse of e
   e*d = 1 mod phi(n)
   (e*d mod phi(n) = 1
   i.e., [(e*d)/(phi(n))] remainder = 1)
6) Now the public key is (n, e)
7) The private key is (n, d).

RSA Encryption: Plain text: M < n
Cipher text : C= M^e mod (n)
RSA Decryption: Cipher text: C
Plain text: M= C^d mod (n)
Where C = cipher text, M = Message/ plain text,
e = encryption key, d = decryption key.

As we can see the encryption and decryption key in RSA algorithm it uses the modular multiplication. This modular multiplication is the most critical part to calculate in RSA algorithm [7]. Because this modular multiplication consumes more time and area while implemented through hardware. So to reduce this critical computation a multiplication called Montgomery multiplication is used. It reduces the computation complexity which was present in RSA algorithm.

III. MONTGOMERY MULTIPLICATION

Montgomery multiplication is one of the faster methods of performing modular multiplication. Here Division is replaced with a simple shift operation and add operation [8]. Montgomery multiplication consumes less area while implementing it through hardware and even it consumes less time. As we see in RSA it uses modular multiplication which consumes more area and time, so instead of modular multiplication, Montgomery multiplication is used.

A. Algorithm for Montgomery Multiplication

Input: n, A and B both with each of k bit length,
A_i and B_i represents the i_{th} bit of A and B respectively[9].
Output: M = A*B mod n
Initially assume M=0
For i = 0 to k
M = M + (A*B_i)
if M_0=0
M = M/2;
else
M = (M+n)/2;
return M;
followed by an extra multiplication of M with
2^k mod n.

B. Architecture of Montgomery Multiplication

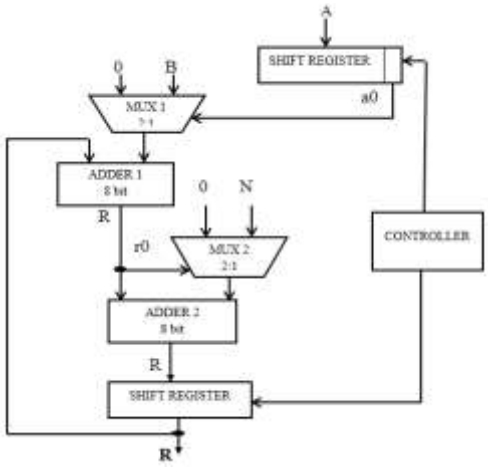

Fig. 3: Architecture of Montgomery Multiplication

IV. INTERACTION OF RSA ALGORITHM WITH MONTGOMERY MULTIPLICATION

Let us assume M = message, C = cipher text, n = p*q, e = encryption key, d = decryption key, e_k and d_k are the bit length of the encryption and decryption key[10].
RSA Encryption function: C= M^e mod (n).
RSA Decryption function: M= C^d mod (n).

A. Algorithm For Encryption

i.e.,for modular exponentiation (M, e ,n)
K= 2^{2k} (mod n);……..(computed externally)
X[0] = MontMul (K, M, n);
Y[0] = MontMul (K, 1, n);
for i in 0 to d_k
X[i+1] = MontMul (X[i], X[i] n);
if d[i] = 1 then
Y[i+1] = MontMul (Y[i], X[i] n);
end if;
end loop;
C = MontMul (1, Y[k], n);
return C;

B. Architecture of Montgomery Multiplication with RSA Algorithm

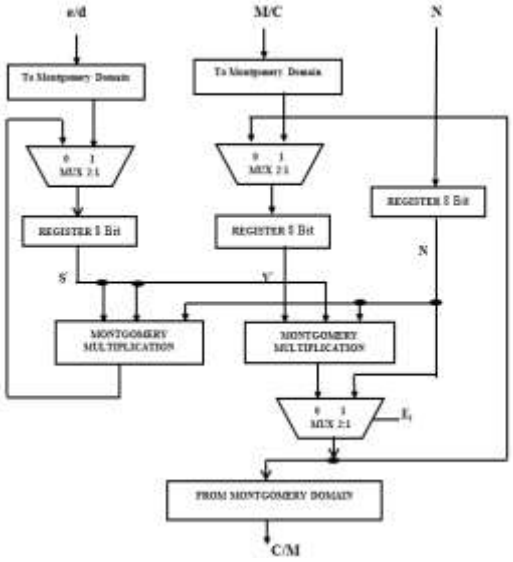

Fig. 4: Architecture for RSA with Montgomery Multiplication

V. RESULT & DISCUSSIONS

A. Simulation Results

1) RSA with Modular Multiplication

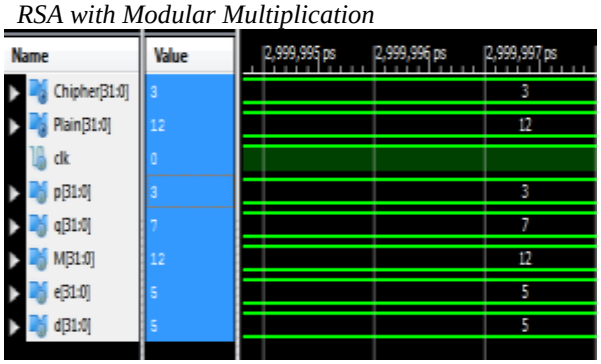

Fig. 5: Simulation result for RSA without Montgomery Multiplication

2) RSA with Montgomery Multiplication

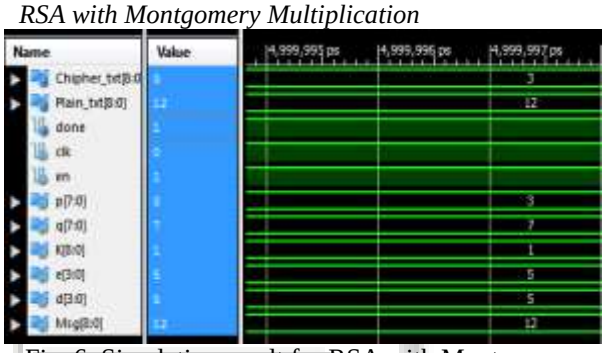

Fig. 6: Simulation result for RSA with Montgomery Multiplication

B. Area Utilization Summary

1) RSA with Modular Multiplication

Logic Utilization	Used	Available	Utilization
No. of Slice registers	660	28800	2%
No. of Slice LUTs	1020	28800	3%
No. of fully Utilized LUT-FF pairs	358	1322	27%
No. of Bonded IOBs	225	480	46%
No. of BUFG/ BUFGLTRLs	1	32	3%
No. of DSP 48Es	15	48	31%

Table 1: Area consumption summary for RSA without Montgomery Multiplication

2) RSA with Montgomery multiplication

Logic Utilization	Used	Available	Utilization
No. of Slice registers	225	28800	0.78%
No. of Slice LUTs	493	28800	1%
No. of fully Utilized LUT-FF pairs	202	1322	36%
No. of Bonded IOBs	63	480	13%
No. of BUFG/ BUFGLTRLs	1	32	3%
No. of DSP 48Es	1	48	2%

Table 2: Area consumption summary for RSA with Montgomery Multiplication

C. Time Consumed

The time used in simulating both the algorithms is necessary because the RSA algorithm is used for communication purpose.

Algorithm / Coding method		Time
RSA without Montgomery Multiplication	Encryption	2305ns
	Decryption	1615ns
RSA with Montgomery Multiplication	Encryption	1385ns
	Decryption	695ns

Table 3: Time consumed for RSA algorithm with and without Montgomery Multiplication

VI. CONCLUSION

The Montgomery Multiplication reduces the area and time which is shown in the tables. So this multiplication can be implemented in situation where the size and speed of the system plays an important role. It should be noted that an increase in bit length results in an increase in critical path delay. On the other hand, the area of the multiplier doubles as the bit length doubles. The area figures presented includes all the I/O registers required to store the bits.

REFERENCES

[1] Wiiliam Stallings, "Cryptography and Network Security" third edition.
[2] R.L Rivest, A.Shamir, and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems,"commun. ACM, vo l. 21, no.2, pp. 120-126, Feb 1978
[3] Gaurav R Patel, "A Comprehensive Study on Various Modifications in RSA Algorithm",International Journal
[4] B.Persis Urbana Ivy, P.Mandiwa,"A modified RSA cryptosystem based on 'n' Prime numbers", vol.1, issue 2, Nov. 2012.
[5] Rajan.S.Jamgekar,"File Encryption and Decryption using Secure RSA",vol 1, Issue 4, Feb 2013.
[6] Ritu tripati & Sanjay Agarwal"Critical Analysis of RSA Public Key Cryptosystem" vol. 4, issue 7, july 2014.
[7] Sushanth.K.S & Manoranjan Pradhan, 2011 International Conference on communication systems and network technologies "Implementation of modular multiplication for RSA algorithm".
[8] P. Montgomery, "Modular multiplication without trial division," Math comput., vol. 44, no. 170, pp. 519-521, 1985
[9] C. Mclvor, M. McLoone, and J. McCanny, "Fast Montgomery modular multiplication and RSA cryptographic processor architectures," in Proc.37th Asilomar Conf. Signals, Syst. Comput. Conf. Rec., vol. 1. Nov. 2004,pp. 379–384.
[10] Richa Garg & Renu Vig,"An efficient Montgomery Multiplication Algorithm and RSA Cryptography Processor", 2007, International Conference on Multimedia Application.

