Journal of Information Systems Engineering and Management
2025, 10(13s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Security Key generation using RSA Algorithm

S.M.Jagdale[1], S.A. Dhole[2], P. D. Kale[3], S. A. Itkarkar[4], R. A. Chavan[5], M. Kamate[6]

[1] *Bharati Vidyapeeth's College of Engineering for Women, Pune. (sumati.jagdale@bharatividyapeeth.edu)*

[2] *Bharati Vidyapeeth's College of Engineering for Women, Pune. (sampada.dhole@bharatividyapeeth.edu)*

[3] *Bharati Vidyapeeth's College of Engineering for Women, Pune. (pranoti.kale@bharatividyapeeth.edu)*

[4] *Bharati Vidyapeeth's College of Engineering for Women, Pune. (savita.itkarkar@bharatividyapeeth.edu)*

[5] *Vishwakarma Institute of Information Technology, Pune.(rohini.chavan@viit.ac.in)*

[6] *Bharati Vidyapeeth's College of Engineering for Women, Pune. (Manisha_kamate@rediffmail.com)*

ARTICLE INFO

Received: 30 Nov 2024
Revised: 16 Jan 2025
Accepted: 30 Jan 2025

ABSTRACT

Prime numbers are fundamental parts of cryptography and communication systems in general. Prime numbers are being used more and more for encryption and decryption purposes. RSA keys use prime numbers. Many algorithms and techniques are available and implemented to generate prime numbers efficiently. The length of prime numbers extends up to 200 bits in cryptography for security concerns. There is a need for electronic hardware that can detect and tell if a number given to it as input is prime or not in minimum time and using minimum hardware for faster transfer of data in the communication system. A simple algorithm for detecting prime numbers is presented in this paper which takes 53.023ns and 9143 LUTs for detecting a prime number whereas papers considered for literature were consuming 5.5ms for generating 128-bit prime numbers using minimum process time algorithm.

Keywords: Prime Number, Cryptography, Encryption, Verilog, Blockchain.

INTRODUCTION

In Mathematics number plays a big role to solve problems. Numbers are classified into various types. Apart from mathematics, there are numbers that are used for security purposes, these numbers are prime numbers.Prime numbers are the numbers which have no factors other than 1 and the number itself. Prime numbers start from *2,3,5……,* all the way to *n*, so the number gets bigger and bigger. Since the prime numbers only have two factors it is very difficult(for bigger primes) to check whether the number is a prime number, to obtain a prime number it should be ensured that the particular number has only two factors and there are various ways to obtain them. The research papers considered in this paper address numerous algorithms, methods, and tests to check the number's primality. Prime numbers have a wide range of applications in information technology, cryptography and blockchain. In a number of conference and journal papers considered in this paper use some sort of genetic algorithms that are implemented in c,c++, and Python languages, authors have introduced stochastic process, Packlington's theorem, Wheel Sieve, Boneh-Franklin scheme, Miller-Rabin test, Message Passing Interface, Trial Division, Sieve of Eratosthenes, all these algorithms were performed on software and results were obtained. Though there is no hardware implementation in any of the literature. In this paper, it is addressed to check the primality test of the input number. The primary objective is to design hardware to detect prime numbers up to numbers that are 34 digits long. This design was implemented in Vivado Xilinx. The algorithm was implemented using Verilog language which is hardware description language(HDL).Further the actual RSA algorithm was implemented.RSA algorithm is an asymmetric cryptography algorithm.Asymmetric sincerely method that it really works on unique keys i.e. Public Key and Private Key. As the call describes that the Public Key is given to everybody and the Private key's saved private.The concept of RSA is primarily based totally at the reality that it key is kept private.The idea of RSA is based on the fact that it is difficult to factorize a large integer. The public key includes numbers wherein one quantity is a multiplication of big high numbers.So if someone can factorize the massive number, the personal secret is compromised.Therefore encryption electricity

Copyright © 2024 by Author/s and Licensed by JISEM. This is an open access article distributed under the Creative Commons Attribution License which permitsunrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

absolutely lies on the important thing length and if we double or triple the important thing length, the electricity of encryption will increase exponentially.

LITERATURE REVIEW

Research tells that the Prime number can be obtained from a dynamical process in which a stochastic process is introduced, author G. Iovane did hypothesis testing to obtain prime numbers[1] also there were some selection rules for determining the prime numbers. Alexandros Papanikolaou and Song Y. Yan have come up with POCKLINGTON'S THEOREM. In this theorem there exists a natural number. Using this theorem the prime numbers can be obtained and this theorem does operate with 100% certainty only drawback is it takes a longer time for generating prime numbers[2]. Wheel Sieve was implemented to achieve prime numbers up to some limit. The wheel sieve is an optimization of the classical Sieve of Eratosthenes and is used to efficiently identify prime numbers[3]. Mit Patel et al. covered twenty types of prime numbers and generated those prime numbers on different platforms such as MacOS , Windows, and Linux. These prime numbers were obtained by prime factorization[4]. Authors Herv´e Chabanne et al. introduced a secure method for generating prime numbers, making it suitable for on-chip applications in insecure environments, with a particular focus on smart cards. It builds on shared key generation techniques and employs masking to protect against side-channel attacks[5]. The paper introduces two algorithms for efficiently generating ring signature keys, considering both memory and time constraints. The first algorithm is memory-efficient, while the second offers a significant improvement in computational complexity, making it suitable for use in mobile devices where optimizing both time and space resources is essential for implementing secure cryptographic operations[6].Ramon Carbó-Dorca introduced a novel approach to generating prime numbers using a recursive algorithm based on powers of 2 and a set of prime natural numbers. It suggests that prime numbers can be constructed by adding these elements, and it conjectures that this method can potentially generate all prime numbers. The paper explores a unique perspective on prime number generation, which may have implications for understanding the relationships between prime numbers and sets of natural numbers[7]. The paper presents a parallel processing algorithm designed to generate prime numbers on a cluster architecture. The primary motivation is to decrease computational costs and accelerate the prime number generation process. The experimental results provided in the paper showcase the practicality and efficiency of this approach, making it a promising solution for large-scale prime number generation tasks[8]. The paper focuses on improving the security of the RSA cryptographic method by generating large prime numbers with a bit size greater than or equal to 512. The Sieve of Eratosthenes is used to obtain small prime numbers, which are then combined to create larger prime numbers within a specified range. The Rabin-Miller primality test is employed to ensure the primality of these generated numbers, contributing to enhanced data security through the RSA method[9]. A new method to implement an S-box is proposed in a paper, as S-box is a key component of many encryption algorithms.S-Box is a table that is used to substitute one byte of data for another. This substitution makes the data more difficult to decrypt without the key.This new method is more efficient than earlier method.This makes data more difficult to be attacked.This method is very useful in applications where speed is a critical issue, example: real time encryption[10].The paper states different theorems to get prime numbers. For example theorem 1: that states that all odd non-primes can be generated using the equation 2(2ab+a+b) + 1, where a and b are natural numbers. The equation jumps over all odd primes, so the odd primes are the "holes" in the set of numbers generated by the equation. Theorem- 2 that states that we can use the equation 6n+1 to generate all odd non-primes, but we need to skip over some values of n in order to avoid generating odd primes. The specific values of n that we need to skip over are given by the equations (a, b) = (3k1, 3k2) or (a, b) = (3k1-1, 3k2-1) [11]. Research in the paper tells us about the Sieve of Eratosthenes and the Sieve of Sundaram Algorithm and shows how these 2 algorithms can be used to generate Prime numbers of randomly generated or sequential numbered random numbers.The paper compares these 2 algorithms and the conclusion is that for small prime numbers Sieve of Sundaram is better than Sieve of Eratosthenes, but to display a large prime number Sieve of Eratosthenes is better, experiment with the help of applications built using Java with the optimization of code and use buffer memory to optimize process prime generator[14]. The dynamics of the Stochastic algorithm that acts as a prime-number generator give rise to a continuous phase transition, which separates a phase where the algorithm is able to reduce a whole set of integers into primes.The paper presents both numerical simulations and an analytical approach in terms of an annealed approximation, by means of which the data are collapsed.This algorithm can be used to generate prime numbers efficiently and accurately, even for numbers that are too large to be factored using traditional

methods[13]. A paper proposes a modification to the RSA cryptosystem that uses multiple prime numbers instead of just two. This makes the system more secure because it is more difficult to factor a large number with multiple prime factors.It shows that the modified RSA system is more secure than the traditional RSA system. This is because it is more difficult to factor a large number with multiple prime factors. The paper also shows that the modified RSA system is more efficient than the traditional RSA system. This is because the encryption and decryption operations can be parallelized[14]. Mersenne numbers can be used to generate all natural numbers in a recursive way. Research in paper proposes a new method for generating natural numbers and counting prime numbers using Mersenne numbers.The paper also shows that Mersenne numbers can be used to count the number of prime numbers in a given interval.Mersenne numbers are very large numbers.This means that a small number of Mersenne numbers can be used to generate a large number of natural numbers and count a large number of prime numbers.And states that this method can be used to generate prime numbers more quickly[15].

METHODOLOGY

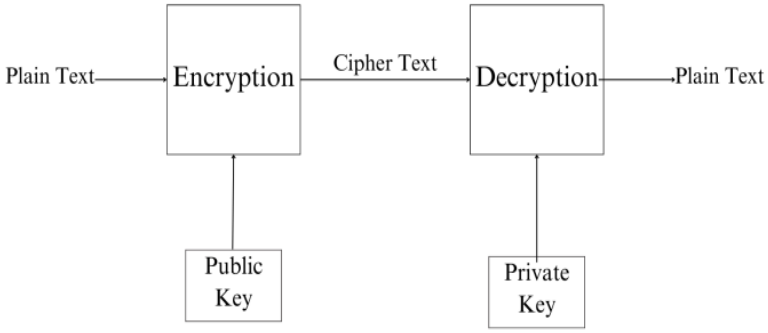

Fig. 1 : Procedural Flow of the System

The system has different phases.The program was coded in Verilog, a Hardware Description Language(HDL). RSA cryptography operates through a series of steps beginning with key generation. Two distinct prime numbers, (p) and (q), are chosen and multiplied to form the modulus (n). The totient function (phi(n)) is computed, and a public exponent (e) is selected, which, alongside (n), forms the public key.

The private exponent (d) is then calculated using the Extended Euclidean Algorithm, completing the private key. Encryption involves raising the plaintext message (M) to the power of (e) modulo (n), yielding the ciphertext (C).

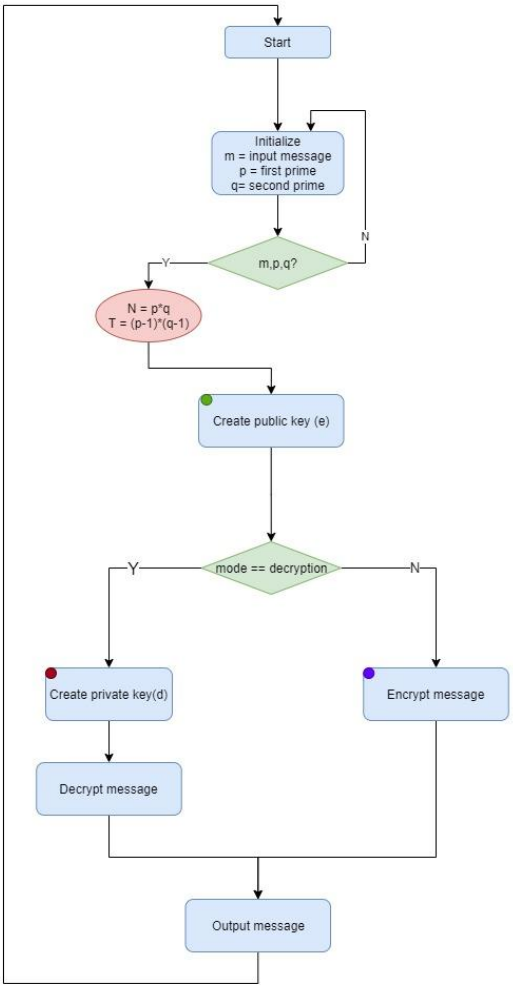

Fig. 2: Algorithm for RSA

Decryption employs the private exponent (d) to recover (M) from (C). RSA's security hinges on the difficulty of factoring large numbers, with longer key sizes offering enhanced protection. Practical implementation considerations, such as key size and padding schemes, are crucial for robust security. Careful management of keys, particularly the private key, is imperative to safeguard encrypted communication.

IV NOVELTY

There are many systems designed for RSA, but very few give accurate results and are actually implemented . This paper takes inspiration to produce a system or an algorithm to implement cryptographic applications. The algorithm is implemented in Verilog hardware description language in Xilinx software. The objective is to reduce the time and hardware taken in encryption and decryption process using up to 128 bit input.

V RESULT AND DISCUSSION

Using Xilinx Software 14.7, the circuit was designed. The program was coded in Verilog HDL language. After Synthesizing the RTL(Register Transfer Logic) schematic(shown in figure 3) for RSA implementation was obtained in which various internal circuits were made. The circuit comprises combinational blocks.

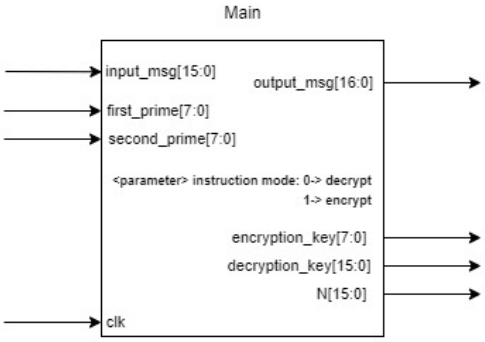

Fig. 3: Block diagram for RSA

In the same software the simulation was done. Various numbers were taken for the testing purpose and it was obtained that the circuit showed accurate result. Simulation end result is proven in fig 4.

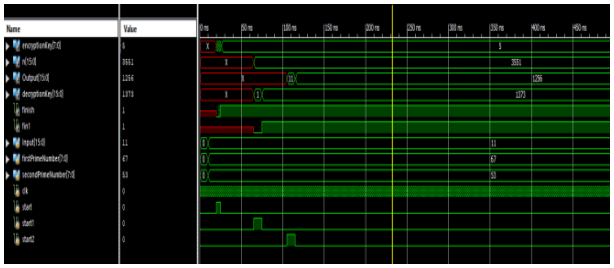

Fig. 4: Encryption Simulation

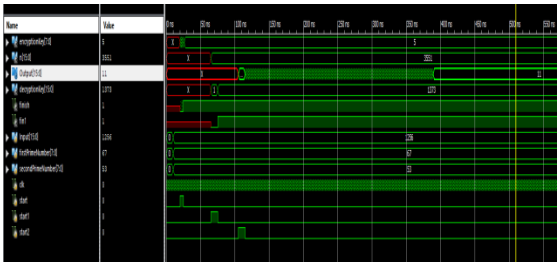

Fig. 5: Decryption Simulation

VII. CONCLUSION

The aim of the project was to build a system to generate keys for Encryption and Decryption in RSA algorithm implemented using Verilog hardware description language and therefore can be implemented on different hardware platforms like FPGA, which makes this project versatile and efficient. System developed in this project is a fast and effective way and can be used in a variety of applications such as Cryptography and Embedded systems.Additionally the system performance can be improved in many ways, expanding it to support larger numbers

REFERENCES

[1] Iovane, Gerardo. "The distribution of prime numbers: The solution comes from dynamical processes and genetic algorithms." Chaos, Solitons & Fractals 37, no. 1 (2008): 23-42

[2] Papanikolaou, Alexandros, and Song Y. Yan. "Prime number generation based on Pocklington's theorem." International journal of computer mathematics 79, no. 10 (2002): 1049-1056.

[3] Paillard, Gabriel Antoine Louis. "A Fully Distributed Prime Numbers Generation using the Wheel Sieve." In Parallel and Distributed Computing and networks, pp. 651-656. 2005.

[4] Patel, Mit, Alok M. Patel, and R. B. Gandhi. "Prime numbers and their analysis." Journal of Emerging Technologies and Innovative Research 7, no. 2 (2020): 1-5.

[5] Chabanne, Hervé, Emmanuelle Dottax, and Laurent Ramsamy. "Masked prime number generation." In First Benelux Workshop on Information and System Security. 2006.

[6] Salazar, José Luis, José Luis Tornos, and Joan Josep Piles. "Efficient ways of prime number generation for ring signatures." IET Information Security 10, no. 1 (2016): 33-36.

[7] Carbó-Dorca, Ramon. "On Prime Numbers Generation and Pairing." International Journal of Innovative Research in Sciences and Engineering Studies (IJIRSES) 3 (2023): 12-17.

[8] K. Wanjale, A. A. Deshmukh, J. C. Vanikar, A. P. Adsul and S. P. Bendale, "Detecting Human Eye Blinks through OpenCV," 2024 IEEE 9th International Conference for Convergence in Technology (I2CT), Pune, India, 2024, pp. 1-5.

[9] Apdillah, Dicky, Muhammad Khoiruddin Harahap, and Nurul Khairina. "Generating Mersenne Prime Number Using Rabin Miller Primality Probability Test to Get Big Prime Number in RSA Cryptography." Int. J. Inf. Syst. Technol 1, no. 1 (2017): 1-7.

[10] Abuelyman, Eltayeb Salih, and A-A. Sultan Alsehibani. "An optimized implementation of the S-Box using residue of prime numbers." International Journal of Computer Science and Network Security 8, no. 4 (2008): 304-309.

[11] Kristyan, Sandor. "On the statistical distribution of prime numbers: A view from where the distribution of prime numbers are not erratic." In AIP Conference Proceedings, vol. 1863, no. 1. AIP Publishing, 2017.

[12] Abdullah, D., R. Rahim, D. Apdilah, S. Efendi, T. Tulus, and S. Suwilo. "Prime numbers comparison using sieve of eratosthenes and Sieve of Sundaram Algorithm." In Journal of Physics: Conference Series, vol. 978, p. 012123. IOP Publishing, 2018.

[13] Luque, Bartolo, Lucas Lacasa, and Octavio Miramontes. "Phase transition in a stochastic prime-number generator." Physical Review E 76, no. 1 (2007): 010103.

[14] P. Allirani, R. Jain, S. Shankar Prasd, K. H. Wanjale, A. Amudha and A. Faiz, "Real-Time Depth Map Upsampling for High-Quality Stereoscopic Video Display," 2024 15th International Conference on Computing Communication and Networking Technologies (ICCCNT), Kamand, India, 2024, pp. 1-5.

[15] Carbó-Dorca, Ramon. "Mersenne numbers, recursive generation of natural numbers, and counting the number of prime numbers." Applied Mathematics 13, no. 06 (2022): 538-543