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

Implementation of AES algorithm using Urdhwa Tiryakbhyam Sutra and Galois field

Kavuri Suresh
Assistant Professor
MallaReddy College of Engineering and Technology
Hyderabad, Telangana.

Jagadish Reddy
MTech Student
MallaReddy College of Engineering and Technology
Hyderabad, Telangana.

Abstract - ***With the ever increasing demand for secure transactions in banking and also in mail delivery systems, encryption and decryption using cryptography plays a very important role. Nowadays, with 90% of secure transactions occurring on smart phones and other hand-held devices, a low on chip area and a high speed algorithm to perform the same becomes the need for the day. In order to meet this requirement, several algorithms have been designed and implemented in the past, but each of these algorithms possess their own shortcomings with respect to an ASIC or an FPGA implementation. In this paper, we propose a novel and area efficient architecture for performing the mix columns & inverse mix columns operation, which is the major operation in the Advanced Encryption Standard (AES) method of cryptography. We perform the same using ancient Vedic Mathematics techniques. The cryptographic unit involving mix columns & inverse mix columns for AES was designed and implemented on a Xilinx Spartan 3e series of FPGA. A 100% area efficiency and a 2 times increase in speed was achieved by the novel algorithm, in comparison with two other popular implementations of the same.***

Key Words: ***encryption; decryption; AES; mixed column; inverse mix columns; Vedic maths; Urdhwa Tiryakbhyam Sutra; Galois field multiplication.***

1. Introduction

Ever since the evolution of wireless communication systems, encryption of data that is to be transmitted, is of a major concern. This is mainly due to the confidentiality of the data at the sender's end and that the risk of a brute force attack by an eavesdropper of such data could prove fatal not only to individuals but also to national security as well . In order to prevent leakage of such potential data, it is necessary to encrypt the same. Cryptography is one such domain which enables data communication with immense security. It involves encrypting the data to be transmitted or shared by means of unique keys, for encryption and decryption, which are known only to the authorized parties, thereby ensuring data security. This serves as a great boon to common man as well the military regime.

Over the past few years, several cryptographic algorithms have been discovered and researched upon, giving importance to the problem of vulnerability of the algorithms especially in applications which demand high security i.e. for smart cards, ATMs, WWW servers etc. Among these, the Advanced Encryption Standard (AES) algorithm is one of the highly preferred algorithms since it is immune to brute-force attack. However, in a hardware implementation perspective, the AES falls short, since it involves several complex operations implemented in the Galois Field (GF-28). Also, these complex operations are iterative in nature which in turn disturbs the speed of the encryption system and therefore increases the vulnerability. One of the key stages in AES architecture, which is prone to the aforementioned problem, is the mix columns stage which is quite circuitous and has been explained in detail along with the other stages involved in AES.

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

Two such popular and conventional methods have been elaborated in Section III of this paper along with their disadvantages. In this paper, we introduce a non-laborious, area efficient and computationally less complex methodology to implement the transformations, with the aid of ancient Vedic mathematics. Vedic Mathematics is an archaic style of mathematics which subsisted in India in 1500 B.C., and was later on brought to limelight by a famous scholar Sri Bharthi Krishna Tirthaji between 1911 and 1918. He systemized it into 16 simple sutras, which are used by most of the researchers and mathematicians due to its ease of use. Out of the 16 formulae available in Vedic Mathematics, the Urdhwa Tiryakbhyam Sutra was utilized in order to address the flaws observed in the conventional mix columns architecture utilized in AES.

2. Basic concept of Rijndael algorithm

Rijndael algorithm was developed by John Daeman and Vincent Rijnmen. It was announced by National Institute Of Standards and Technology (NIST) in 1997 as a criteria of security, performance efficiency, flexibility and implementability and published the specifications of standard in Federal Information Processing Standard (FIPS) publication 197.Before long time, Data Encryption Standard (DES) was considered as a standard for symmetric key encryption. Data Encryption standard has key length of 56 bits which is considered small and can be easily broken. Rijndael can be specified with key and block sizes in any multiple of 32 bits. It has fixed block size of 128 bits and key size of 128, 192 , 256 bits. This paper deals with FPGA implementation of Rijndael encryptor / decryptor using an iterative looping approach with block size 128 bits and key size of 192 bits in 12 rounds.

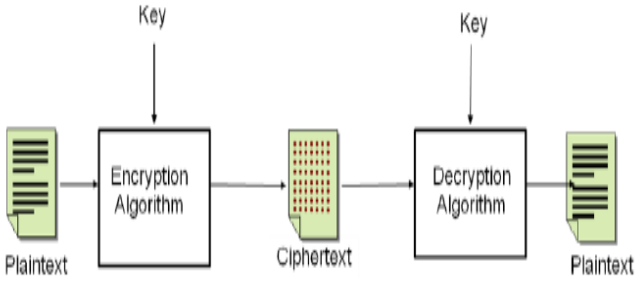

Fig. 2.1: Basic block diagram of Rijndael algorithm

The encryption process is achieved by processing plaintext. Key expansion generates a key schedule that is used in cipher and inverse Cipher procedure and is composed of specific number of rounds. Cipher text is a scrambled message produced as output. Decryption process is same but in reverse manner. The number of rounds to be performed during the execution of algorithm is dependent on the key length. Rijndael is composed of four high-level steps. These are:

- Key Expansion
- Initial Round
- Rounds
- Final Round

Key expansion is performed using key schedule. Initial round consists only an AddRound Key operation. The round step consists of a Subbytes, Shiftrows, Mix Columns and an Add Round key operation. The number of rounds in the rounds step varies from 10 to 14 depending on the key size. Finally, the Final Round performs a SubBytes, ShiftRows & an AddRoundKey operation. Decryption in Rijndael algorithm is done by performing the inverse operation of the simple operations in reverse order.

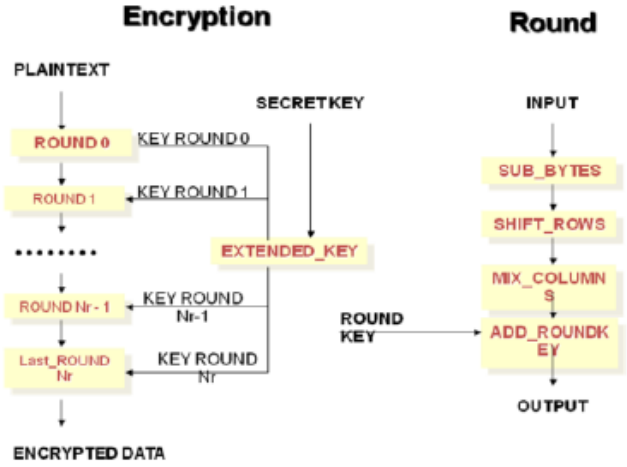

Fig. 2.2: Basic concept

3. Encryption Process

In the encryption of the Rijndael algorithm, each round performs four transformations namely SubBytes, ShiftRows, MixColumns and AddRoundKey, while the final round does not perform the MixColumns

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

transformation. The key used in each round which is called the round key, this is generated from the initial key by a separate key scheduling module of Rijndael . These can be proposed as:

- Substitution of Bytes using S-box
- Row shifting operation using different offset
- Mixing of data within each column of state array
- Adding a round key with State

3.1 Sub Byte Transformation: It is a non-linear byte substitution which operates independently on each byte of the state using the S-box table which contains 256 numbers.

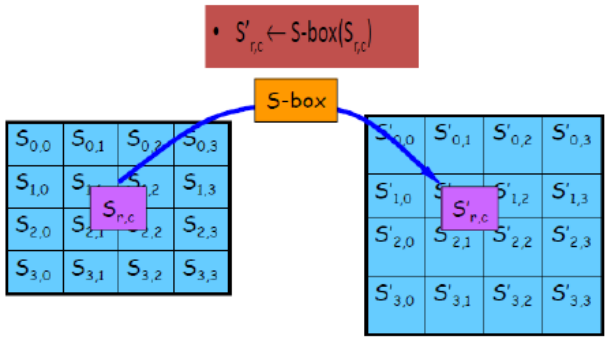

Fig. 3.1: Subbyte transformation

3.2 ShiftRows Transformation: In Shift Rows transformation , the rows of the state are cyclically left shifted over different offsets. Row 0 is not shifted; row 1 is shifted one byte to the left; row 2 is shifted two bytes to the left and row 3 is shifted three bytes to the left.

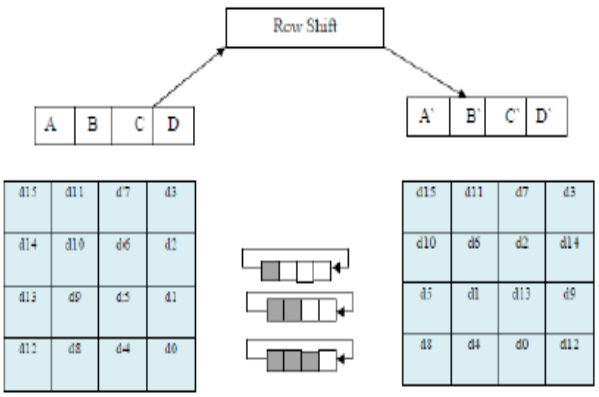

Fig.3.2: Shiftrows transformation

3.3 Mixcolumns Transformation: In MixColumns transformation, the columns of the state are considered as polynomials over GF (2*) and multiplied by modulo x4 + 1 with a fixed polynomial c(x), given by: c(x) = {03}x3 + {01}x2 + {01}x + {02}.

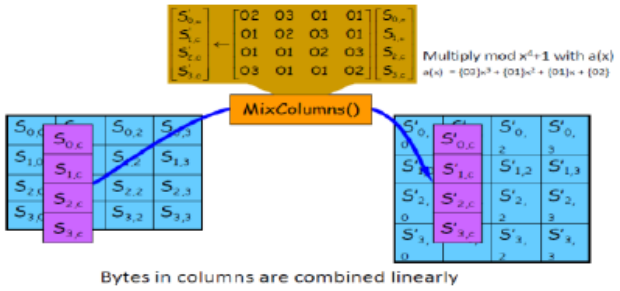

Fig.3.3: Mixcolumns transformation

3.4 AddRoundKey Transformation: In AddRoundKey transformation, a Round Key is added to the state by a simple bitwise XOR operation. Each round Key consists of Nb words from the Key expansion. These Nb words are added into the columns of the state.

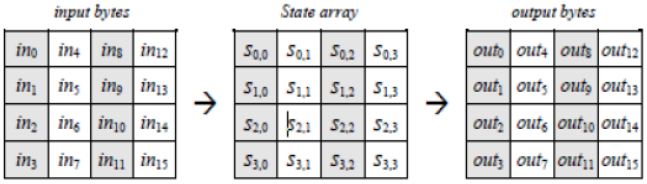

Fig.3.4: Addroundkey transformations

4. Decryption Process

For decryption, the same process occurs simply in reverse order- taking the 128-bit block of cipher text and converting it to plaintext by the application of the inverse of the four operations. AddRoundKey is the same for both encryption and decryption this process is inverse of encryption process. The last round values of both data and key are first round inputs for the decryption process and follows in decreasing order.

5. Proposed Method : Vedic Mathematics

Vedic mathematics is part of four Vedas (books of wisdom). It is part of Sthapatya- Veda (book on civil engineering and architecture), which is an upa-veda (supplement) of Atharva Veda.

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

It covers explanation of several modern mathematical terms including arithmetic, geometry (plane, co-ordinate), trigonometry, quadratic equations, factorization and even calculus. His Holiness Jagadguru Shankaracharya Bharati Krishna Teerthaji Maharaja (1884-1960) comprised all this work together and gave its mathematical explanation while discussing it for various applications. Swamhiji constructed 16 sutras (formulae) and 16 Upa sutras (sub formulae) after extensive research in Atharva Veda. The very word „Veda" has the derivational meaning i.e. the fountainhead and illimitable storehouse of all knowledge. Vedic mathematics is the name given to the ancient system of mathematics or, to be precise a unique technique of calculations based on simple rules and principles with which many mathematical problems can be solved, be it arithmetic, algebra, geometry or trigonometry. The system is based on 16 Vedic sutras or aphorisms, which are actually word formulae describing natural ways of solving a whole range of mathematical problems. The beauty of Vedic mathematics lies in the fact that it reduces the otherwise cumbersome-looking calculations in conventional mathematics to a very simple one. This is so because the Vedic formulae are claimed to be based on the natural principles on which the human mind works. This is a very interesting field and presents some effective algorithms which can be applied to various branches of engineering such as computing and digital signal processing.

A. Urdhva Tiryakbhyam Sutra

The proposed Vedic multiplier is based on the "Urdhva Tiryagbhyam" sutra (algorithm). These Sutras have been traditionally used for the multiplication of two numbers in the decimal number system. In this It is based on a novel concept through which the generation of all partial products can be done with the concurrent addition of these partial products. The algorithm can be generalized for n x n bit number. Since the partial products and their sums are calculated in parallel and the multiplier is independent of the clock frequency of the processor. Due to its regular structure, it can be easily layout in microprocessors and designers can easily circumvent these problems to avoid catastrophic device failures. The processing power of multiplier can easily be increased by increasing the input and output data bus widths since it has a quite a regular structure. Due to its regular structure, it can be easily layout in a silicon chip. The Multiplier based on this sutra has the advantage that as the number of bits increases, gate delay and area increases very slowly as compared to other conventional multipliers.

B. Multiplication of two decimal numbers 252 x 846

To illustrate this scheme, let us consider the multiplication of two decimal numbers 252 x 846 by Urdhva-Tiryakbhyam method as shown in Fig. 4.1. The digits on the both sides of the line are multiplied and added with the carry from the previous step. This generates one of the bits of the result and a carry. This carry is added in the next step and hence the process goes on. If more than one line are there in one step, all the results are added to the previous carry. In each step, least significant bit acts as the result bit and all other bits act as carry for the next step. Initially the carry is taken to be zero.

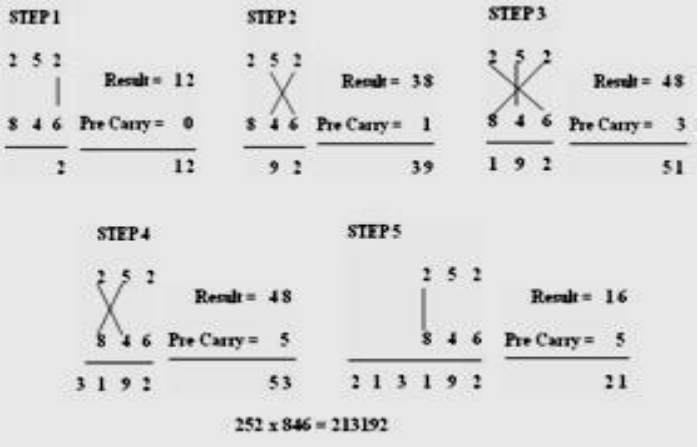

Fig.4.1: Example

6. SYNTHESIS AND SIMULATION RESULTS

A. Multiplier performance and comparison

For performance comparison, various multipliers- Urdhwa Tiryakbhyam based Vedic multiplier, conventional compressor based Vedic multiplier and the proposed higher order compressor based Vedic multiplier, have been implemented in Verilog. The design of 8x8 multiplier for each of these architectures was simulated and synthesized in Cadence RTL

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

Compiler for 180 nm technology. The unoptimized gate count, area and power have been compared. The standard cell based fast library was used for synthesis. Fig. 6.1, 6.2 and 6.3 illustrates the comparative performance of the three multipliers in terms of gate count, area and power dissipation. It can be observed that the proposed multiplier uses 6.3% and

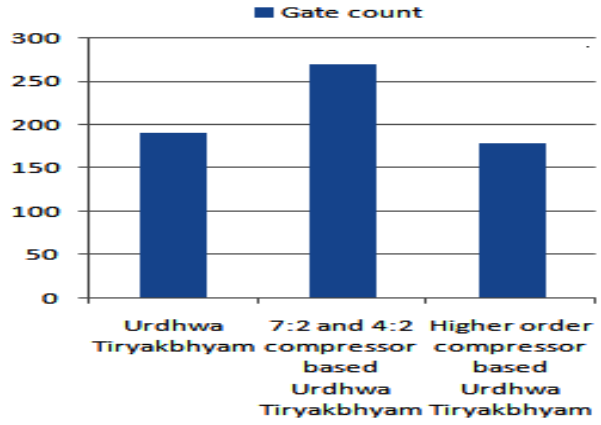

Fig 6.1 Gate count comparison for 8x8 multipliers

33.7% less gates and 12.9% and 25.8% less area in comparison to Urdhwa Tiryakbhyam based Vedic multiplier and conventional compressor based Vedic multiplier respectively.

The leakage power, dynamic power and total power dissipation for the three multipliers has been tabulated in Table III. It can be observed that the proposed multiplier dissipates 27.8% and 30.5% less total power than Urdhwa Tiryakbhyam based Vedic multiplier and conventional compressor based Vedic multiplier respectively as illustrated in Fig. 6.3. And the simulation results in figure 6.4 and 6.5

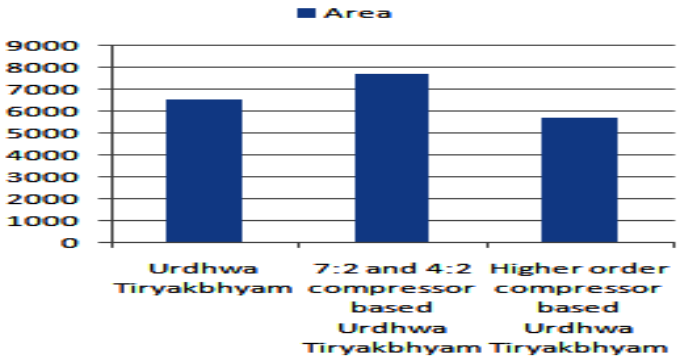

Fig 6.2 Area comparison for 8x8 multipliers

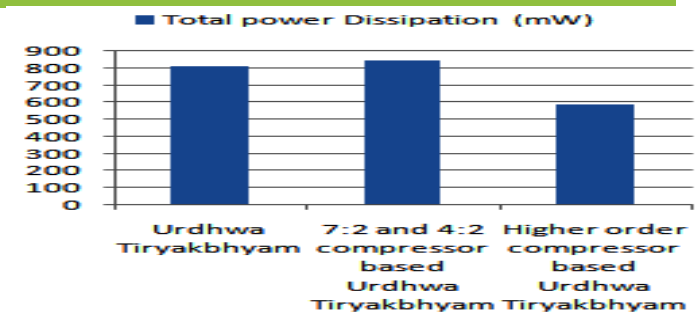

Fig 6.3 Power dissipation comparison for 8x8 multipliers

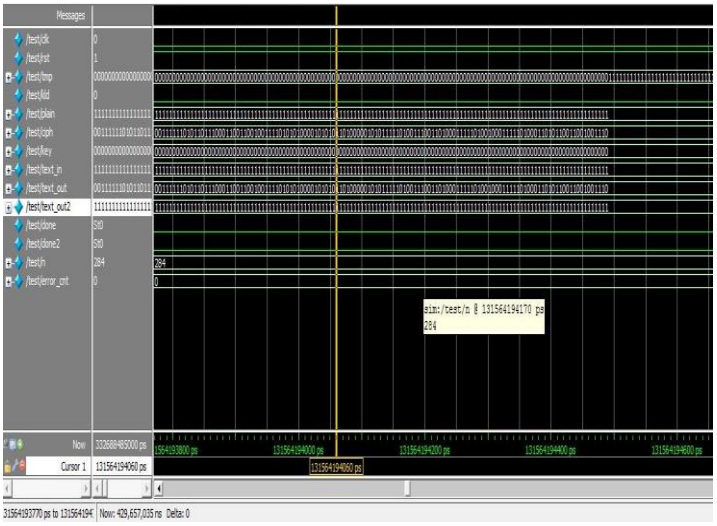

Fig 6.4 Waveform result of 128 bit AES algorithm at clock 0

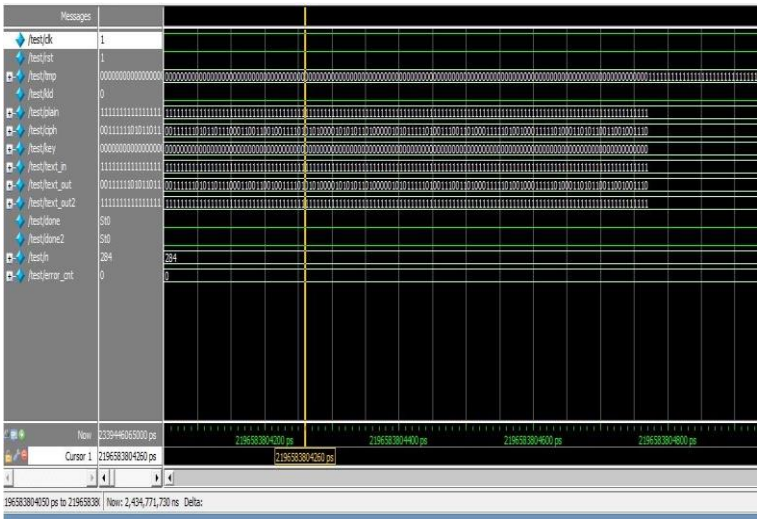

Fig 6.5 Waveform result of 128 bit AES algorithm at clock 1

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

7. CONCLUSION AND FUTURE SCOPE

In this paper, we introduce a non-laborious, area efficient and computationally less complex methodology to implement the transformations, with the aid of ancient Vedic mathematics. Out of the 16 formulae available in Vedic Mathematics, the Urdhwa Tiryakbhyam Sutra was utilized in order to address the flaws observed in the conventional mix columns architecture utilized in AES.

In this paper, we have proposed an architecture using Vedic mathematics for performing mix and inverse mix column computations. In a VLSI/FPGA perspective, the new architecture performs extremely well in terms of speed and occupies less area. The methodology introduced in this paper and other Vedic mathematics operations could be extended for other operations involved in AES as well. Exploration of the same for other encryption or decryption algorithms such as DES can also be researched upon in the future.

8. References

[1] *Advanced Encryption Standard (AES)*, FIPS PUB 197, Nov. 26, 2001, Federal Information Processing Standards publication 197. Federal Information Processing Standards Publication 197.

[2] X. Zhang and K. K Parhi, "High-speed VLSI Architecture for the AES Algorithm", *IEEE Transactions on Very Large Scale Integration (VLSI) System.*, vol.12, no. 9, pp. 957–967, Sep. 2004.

[3] Jose M. Granado-Criado , Miguel A.Vega-Rodrıguez, Juan M. Sanchez-Perez and Juan A. G´omez-Pulido, "A new methodology to implement the AES algorithm using partial and dynamic reconfiguration", Integration, the VLSI Journal 43 (2010) 72-80.

[4] Vincent Rijmen, "Efficient Implementation of the Rijndael S-box". Katholieke Universiteit Leuven, Dept. ESAT. Belgium".

[5] Jarvinen, K., Tommiska, M., and Skytta, "A Fully Pipelined Memoryless 17.8 Gbps AES-128 Encryptor". Proc. ACM/SIGDA 11th ACM Int. Symposium on Field-Programmable Gate Arrays, FPGA 2003, Monterey, CA, USA, February 2003, pp. 207–215.

[6] Hodjat A. and Verbauwhede. I, "A 21.54 Gbits/s fully pipelined AES processor on FPGA". Proc.12th Annual IEEE Symposium. Field Programmable Custom Computing Machines, FCCM‟04, Napa, CA, USA, April 2004, pp.308–309.

[7] I. Hammad, K. El-Sankary, and E. El-Masry, "High Speed AES Encryptor with Efficient Merging Techniques," *IEEE Embedded Systems letters*, vol. 2, no. 3, pp. 67-71, Sept. 2010.

[14] K. Gaj and P. Chodowiec. Very Compact FPGA Implementation of the AES Algorithm. In the proceedings of CHES 2003, Lecture Notes in Computer Science, vol 2779, pp. 319-333, Springer-Verlag.

[8] G. Rouvroy, F.-X. Standaert, J.-J. Quisquater and J.-D. Legat, "Compact and efficient encryption/decryption module for FPGA implementation of the AES Rijndael very well suited for small embedded applications", Information Technology Coding and Computing 2004.

Author Details:

Kaveri Suresh has completed his B.Tech from JNTU Hyderabad, Telengana and M.Tech from Sathyabama University, Chennai, Tamil Nadu.

Jagadish Reddy has completed his B.E in Instrumentation Technology from Poojya doddappa appa college of Engineering and Technology, VTU affiliated college, Karnataka .He is pursuing his M.Tech in VLSI and Embedded Systems from Malla Reddy College of Engineering and Technology, J.N.T.U.H affiliated college.