High Throughput, low cost, Fully Pipelined Architecture for AES Crypto Chip

Nalini C. Iyer, Anandmohan P.V., *Fellow IEEE,* Poornaiah D.V, *and* V.D. Kulkarni, *Member, IEEE*

***Abstract*--Reprogrammable devices such as Field Programmable Gate Arrays (FPGA's) are highly attractive options for hardware implementations of encryption algorithms. Several papers described efficient architectures for ASICs and FPGAs.**

Various approaches for efficient hardware implementation of the Advanced Encryption Standard algorithm based on architectural optimization and algorithmic optimizationfor different modes are discussed and implemented. This paper proposes Compact,Memory less, high-speed hardware architectures for the Rijndael AES Encryptor/Decryptor, with combined data path ,resource sharing and logic optimization for novel networking applications. Architectural optimization exploits the strength of pipelining, loop unrolling and sub-pipelining. Speed is Increased by processing multiple rounds simultaneously at the cost of increased area. Algorithmic optimization exploits algorithmic strength inside each round unit.

Various methods such as Resource sharing and Common sub expression elimination method for realizing various transformations in each round unit are presented to reduce the critical path and area issues between encryptor and decryptor. advantage of sub-pipelining can be further explored by eliminating the unbreakable delay incurred by look-up tables in the conventional approaches, the widely used implementation of S-box, which uses combinational logic only .We explore the use of subfield arithmetic for efficient implementations of Galois Field arithmetic such as multiplication and inversion.. Our technique involves mapping field elements to a composite field representation and a representation technique which minimizes the computation cost of the relevant arithmetic. Our method results in a very compact and fast gate circuit for Rijndael encryption and decryption The pipelined architecture can be made to toggle between the encryption and decryption modes without the presence of any dead cycle The performance results for various architectural designs such as Iterative, pipelining and sub pipelining and various approaches for S-box, of fly computation such as Look up table, Use of Block RAM's and On fly computation using Composite field arithmetic in terms of throughput and area are presented and compared with previous reported designs. Using the proposed architecture, a fully sub-pipelined AES-128 core with both inner and outer round pipelining and a 5 sub-stages in each round unit implemented using Virtex -E devices can achieve a throughput of 26.64Gbps at 206.84 MHz and 11720 CLB Slices in non-feedback modes with reduction of reconfigurable logic area of the complete cipher by up to 30%., and S-box with 64% reduction in area ,which is faster and more efficient than the fastest previous FPGA implementation known to date.

***Index Terms*--AES, S-box, GF, FPGA**

I. Introduction

SOFTWARE-based implementations of cryptographic algorithms fall short of the required performance, as the transmission speeds of core networks reach the gigabits per second (Gbps) range. The significance and applicability of hardware-based implementations of cryptographic algorithms is therefore of interest also to the Field Programmable Gate Array (FPGA) design community. FPGAs are nearly ideal candidates for high-speed cryptography for several reasons. The target market is generally low- to medium sized, which makes the usage of Application Specific Integrated Circuits (ASIC) less attractive because of the large initial costs included in starting a ASIC manufacturing process. FPGA-designs also have a quicker time-to-market cycle than ASICs. In autumn 2000 the Rijndael algorithm, developed by Joan Daemen and Vincent Rijmen [1], was selected as the AES algorithm by The National Institute of Standards and Technology (NIST) of the United States and was formally published on November 26 2001 in Federal Information Processing Standards' (FIPS) publication FIPS-PUB 197 [2]. The standard became effective on May 26, 2002. The implementation of fully unrolled secret-key cryptographic algorithms with inner and outer pipelining is feasible on million-gate FPGAs. A typical feature of modern FPGAs is the inclusion of embedded internal memory within the device, for example BlockRAMs in Xilinx' Virtex devices [19, 20] and Embedded System Blocks (ESBs) in Altera's Apex devices [1]. This has several benefits, since lookup tables and

Nalini C. Iyer is with the Dept. of E&C, BVBCET, Hubli (e-mail: nalinic@bvb.edu).

Dr. Anandmohan P.V, is with *ECIL, Bangalore (e-mail: anandmohanpv@hotmail.com.)*

Poornaiah D.V, is with *ITI, Bangalore (e-mail:*poorna_dv@yahoo.com).

V.D. kulkarni is with ,*CG-Coreel Bangalore (e-mail:*vdk@cg-coreel.com).

1-4244-0370-7/06/$20.00 ©2006 IEEE

conversion functions can be easily implemented as small RAMs within the device.

The rest of the paper is organized as follows. Section 2 describes briefly the AES cryptographic algorithm and previous works. Section 3 explains the details of our design on the AES cryptographic chip. Section 4 compares the performance of our implementation to earlier ones. Finally, Section 5 concludes the paper.

II. AES ALGORITHM AND PREVIOUS WORK

A..AES Algorithm

The AES algorithm is a symmetric block cipher that processes data blocks of 128 bits using a cipher key of length 128-, 192-, or 256-bits. Each data block consists of a 4 array of bytes called the *state*, on which the basic operations of the AES algorithm are performed. After an initial round key addition, a round function consisting of four different transformations—sub-bytes, shift-rows, mix-columns, and add-round-key—is applied to the data block in the encryption procedure and in reverse order with inverse transformations in Decryption procedure . Round function in last round of Encryption and first round of decryption does not contain Mix/Inv. Mix column transformation. The round function is performed iteratively 10, 12, or 14 times, (Nr)depending on the key length. Sub-bytes operation is a nonlinear byte substitution that operates independently on each byte of the state using a substitution table (S-Box). Shift-rows operation is a circular shifting on the rows of the state with different numbers of bytes (offsets). Mix-columns operation mixes the bytes in each column by the multiplication of the state with a fixed polynomial modulo $x4 + 1$. Add-round-key operation is an XOR that adds a round key to the state in each iteration, where the round keys are generated during the key expansion phase.

B. *Previous Work*

There exist many presentations of hardware implementations of Rijndael AES algorithms in literature.. In 2001, Elbirt et al. [6] compared five candidate algorithms (including Rijndael algorithm) for AES block cipher using FPGA implementations. Here, the throughputs of Rijndael algorithm were in 187.8 Mb/s ~ 1.94 Gb/s. Verbauwhede et al. [25] presented an ASIC implementation under the throughput of 2.29 Gb/s. Su et al. [2] with reduced hardware overhead McLoone and McCanny [10] utilized look-up tables to implement the entire Rijndael round function under the throughput of 12 Gb/s using FPGAs. In 2004, Hodjat and Verbauwhede [8]'s FPGA implementation showed a high throughput of 21.54 Gb/s using a fully pipelined approach with inner-round pipelining and outer-round pipelining.

III. PROPOSED AES-128 ARCHITECTURE

One of the main goals of this work is to build a high-performance fully integrated and synthesizable Rijndael core as a piece of soft *intellectual property* (IP). In this section, we present a Rijndael cipher system architecture and discuss the design trade-offs among the various design choices to determine the architectures of the entire cipher system and functional units .

1) Architectural optimization:

Types of architectures that can be used to increase the speed of encryptor/decryptor are based on pipelining, sub-pipelining, and loop unrolling. Processing multiple rounds simultaneously at the cost of increased area increases speed. Architectural optimization is not an effective solution in feedback mode. Loop unrolling is the only architecture that can achieve a slight speedup with significantly increased area. Sub pipelining can achieve maximum speedup and the best speed/area ratio in non-feedback mode.

2) Algorithmic Optimization

Various methods such as Resource sharing and Common sub expression elimination method for realizing various transformations such as in each round unit are presented to reduce the critical path and area issues between encryptor and decryptor In this section, we present detailed architectures for each of the non-trivial transformations in the AES encryption process.

A. Pipelining

.

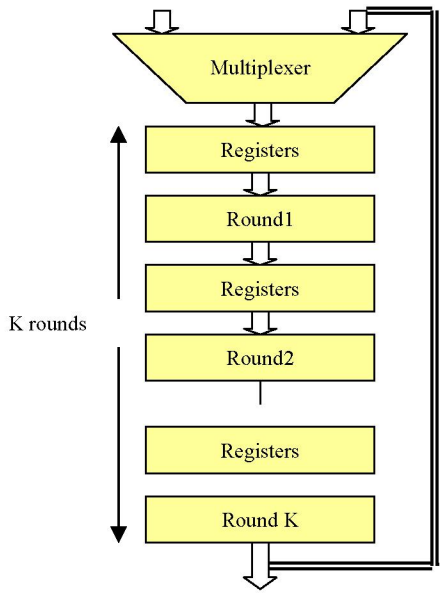

Fig. 1. Pipelined Architecture.

The pipelined architecture to increase the speed of algorithm by processing multiple blocks of data simultaneously is realized by inserting rows of registers among combinational logic [12]. Each pipeline stage is one round unit in this case. Pipelining introduced due to insertion of registers in between rounds is outer round pipelining as in fig 1.and within each

round is inner round pipelining. For a k-round pipelined architecture, after the pipeline reaches its full depth, k blocks of data are processed simultaneously in different stages with area and the latency proportional to k.

B. Sub-Pipelining

Similar to the pipelining, sub pipelining also inserts rows of registers among combinational logic, but in this case, registers are inserted both between and inside each round unit (both outer and inner pipelining) and is carried out on a partially pipelined design when the round is complex as in fig 2. If each round unit has n stages with equal delay, then a k-round sub-pipelined architecture can achieve approximately n times the speed of a k-round pipelined architecture with a slight increase of area caused by additional registers and control logic.

However, arbitrary number of stages in each round unit does not always bring speedup. The minimum clock period is decided by the combinational element with the longest delay. The overall speed does not improve despite increased area caused by the additional registers. It decreases the pipeline delay between stages but increases the number of clock cycles required to perform encryption.

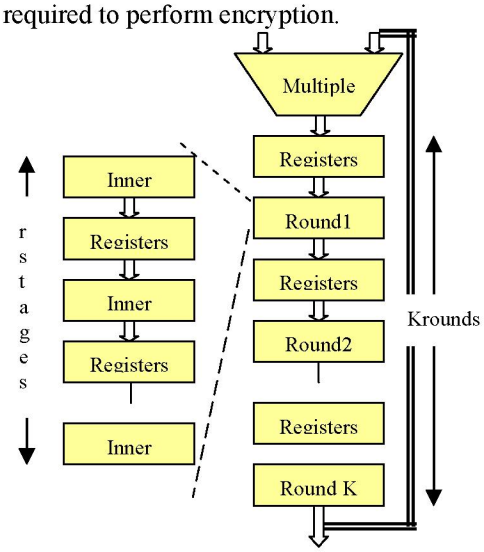

Fig. 2. Sub pipelined Architecture.

1) Loop Unrolling

Loop unrolled or unfolded architectures can process only one block of data at a time, but multiple rounds are performed in each clock cycle[12] as in fig 3. The unrolling or unfolding factor, k, is usually chosen as a divisor of Nr and the maximum value of k is Nr.

The number of cycles to process one block of data is Nr / k in this case. Meanwhile, the clock period of k-round loop unrolled architecture is increased to slightly smaller than k times the clock period of a pipelined architecture because of the setup time and propagation delay of registers.

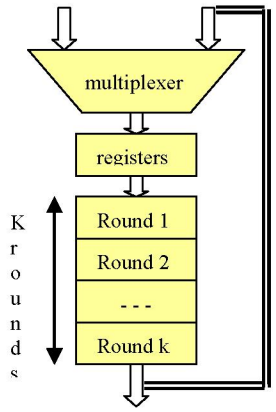

Fig. 3. Loop Unrolled Architecture.

The area of this architecture is also proportional to the number of rounds in each loop. Optimization for maximum speed can be realized by a fully sub-pipelined architecture [26].

C. AES-128 data path design:

Our implementation combines encryption/decryption by empowering resource sharing, as encryption/decryption uses the same operations (in different order) in each round to the maximum extent. Data path Modules present in the proposed design are, Add Key, Substitute byte +shift row including inverse sub. and inv.shift operation, mix/Inv Mix column, Selector and controller as shown in fig.4.

The dual functionality of the AES data path of 128 bit wide not only adds value to the design of being dual functional, but also results in a compact design due to high degree of resource sharing for both Iterative and Pipelined designs. A simple controller with a 6-bit counter attached to it, extracts timing information for issuing the control signals.

IV. Design of S-Box/Inv S-box Module

Designing a compact S-Box is one of the most critical problems for reducing the total circuit size of the Rijndael hardware. Most of the earlier designs implemented byte transformation using look up table techniques, which not only use more hardware (in terms of memory) but also limit maximum operable clock frequency to that of the memory access time of the device [10, 11]. In this section, we propose optimization of S-Box by introducing a new composite field [23]. Design of *S-Box* and Inverse S-box involves multiplicative inverse and affine transform and their inverses. Implementation of multiplicative inverse in GF (2^8) is not practical in hardware and also speed efficient [28]. To overcome this problem, the composite field mapping technique is used which provides an attractive way of multiplicative inverse computation involving mapping from the GF (2^8) to GF ($((2^2)^2)^2$) [24] reducing the hardware complexity To break the critical path delay, we insert registers at appropriate points such the the delay is equally distributed among the registers to realize 2 stage and 3

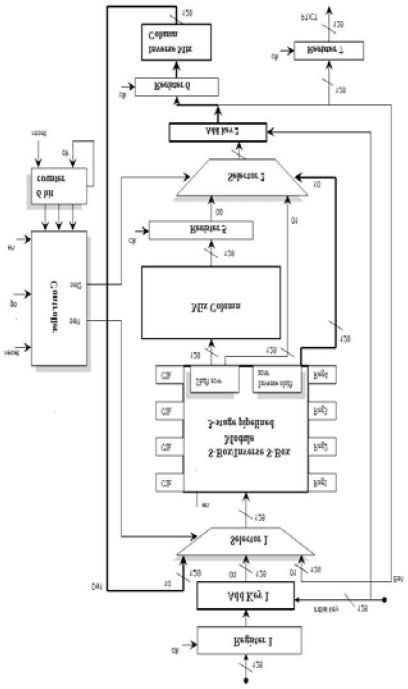

Fig. 4. Proposed AES Datapath Design with controller module.

stage pipelining. In AES128 we have combined the 3-stage pipelined S-box/Inv S-box with Shift row/InvShiftrow. The most costly operation in the S-Box is the multiplicative inversion over a field A. where A is an extension field over $GF(2)$ with the irreducible polynomial $m(x)$. the 3 steps involved in computation are mapping of all elements of the field A to a composite field B, using an Isomorphism function δ, Compute the multiplicative inverses over the field B and Re-map the computation results to A, using the function δ^{-1}.Isomorphism functions required in this method are merged with the affine transformations.. To reduce the cost as much as possible, we built the composite field B by repeating degree-2 extensions under a polynomial basis using these irreducible polynomials:

GF (2^2): $x^2 + x + 1$,GF $((2^2)^2)$: x2 + x + φ

GF $(((2^2)^2)^2)$: x^2 + x + λ, Where $\varphi = \{10\}_2$, $\lambda = \{1100\}_2$.

Two basic blocks are Inverter (GF (2^8) and Multiplier GF (2^4) are **as** shown in fig. 4 and fig 5. respectively.

Various architectures have been proposed for the implementation of the Mix Columns transformation [17, 18]. Applying substructure sharing to both the computation of a byte and between the computations of the four bytes in a column of the State, an efficient Mix Columns implementation architecture can be derived the two new optimization methods both in bit level are presented and implemented.

In Mix Column Operation Common sub expressions elimination method leads to common sub expressions $x_0 \oplus x_7 = x8$ and $x_3 \oplus x_7 = x_9$ for substitution, which requires 144 bit level XOR's with critical path delay of 5XOR's. Inverse Mix Column takes 264 bit level XOR's with critical path delay of 6XOR using following common sub expressions.

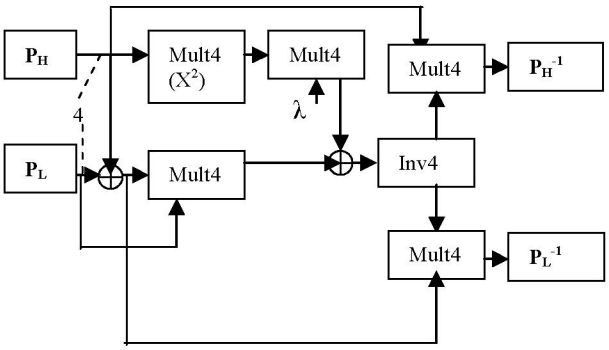

Fig. 5. Inverter over a composite field GF($(2^2)^2)^2$).

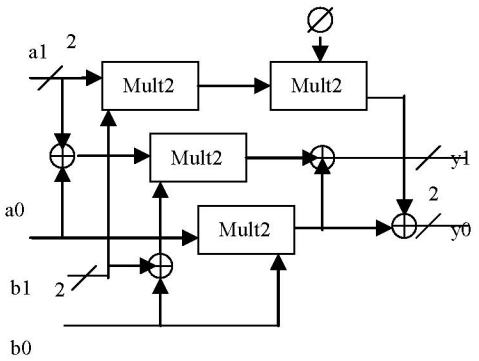

Fig. 6. Multiplier over GF(2^4).

$$x_8 = x_5 \oplus x_6 \qquad x_9 = x_4 \oplus x_7$$
$$x_{10} = x_1 \oplus x_8 \qquad x_{11} = x_0 \oplus x_5$$
$$x_{12} = x_2 \oplus x_3 \qquad x_{13} = x_3 \oplus x_7$$
$$x_{14} = x_0 \oplus x_{10} \qquad x_{15} = x_1 \oplus x_5$$
$$x_{16} = x_2 \oplus x_6 \qquad x_{17} = x_3 \oplus x_9$$
$$x_{18} = x_2 \oplus x_8 \qquad x_{19} = x_7 \oplus x_{16}$$

V. Performance Results and Related Work

The generic AES architecture described has been captured using VHDL and numerous designs instantiated using a Virtex-E XCV2000E- Xilinx Foundation Series 7.1i software were used in the synthesis of these designs. The parameters to evaluate the proposed implementation of AES are Throughput(maximum Encryption rate), Area cost in terms of number of Virtex-E slices and BRAM's and the throughput per area, which measures the contribution of each slice to the throughput and hence the Efficiency of the implementation. The results are expressed in terms of architecture, Maximum clock frequency, the Throughput, Area(Number of Slices) and Throghput per Area(T/A).

TABLE I
OVERALL COMPARISONS OF 128-BIT KEY AES WITH ON FLY S-BOX, FPGA IMPLEMENTATIONS

Arch	D N	Device	T(mbps)	CLB Slices	T/ S
Standaert et a(4) (Pipeline)	E	XCV1000	22016	17984	1.22
Standaert et a(4) (Iterative)	E	XCV1000	1563	2257	0.69
Wang(6)	E	XVC1000	1.64GB PS	1857	0.88
Wang(6)	E/ D	XCV1000	9728	5150	1.88
Sklavas(2 3)	E/ D	XCV1000	3650	17314	0.21
Hodjat et al[]	E	XCV2000	21.54gb ps	5177(84 BRAM)	
Ours(Iter ative)	E/ D	XCV1000	196mb ps	1700	0.12
Ours(out er 2 stage)	E/ D	XCV1000	392mb ps	3200	0.12
Ours(out er 5 stage)	E/ D	XCV1000	980	7500	0.13
0urs (Outer 10 stage , inner 5 stage)	E/ D	XCV1000 XCV2000 XC2VP30	8.26gb ps 11.87g bps 26.47g bps	12326 12026 11720	0.67 0.85 2.26

TABLE II
COMPARISON OF OUR AES DESIGN WITH LUT BASED S-BOX & ON THE FLY S-BOX FOR FULLY PIPELINED ARCHITECTURE

Desi gn	device	Slices	fmax	Throu ghput
On fly S-box	XC2vp 30 XCV20 00	11720 12026	206.84M Hz 92.7	26.47 gbps 11.87 gbps
LUT base d S-box	XCV20 00	19200 +160R OMs	130.242M Hz	16.67 gbps

Tables I,II,III show the implementation results of our AES Rijndael design and comparison with the other AES approaches. Table IV gives the synthesis of Optimised mix/Inv mix column approach. Our fully unrolled architecture with inner and outer pipelining provides the highest encryption throughput of 26.47gbps,highest clock frequency of 206.84 Mhz ,11720 slices and best T/A, among the reported implementations.

TABLE IV
SYNTHESIS OF MIX/INV MIX COLUMN TRANSFORMATION

Modules	Target Device	No. of Slices	No. of 4 input LUT	Estimated path delay
Mix Column	XCV1000-6	37	64	10.841n
Inv- Mix Column	XCV1000-6	76	132	14.678n

TABLE III
COMPARISON OF ITERATIVE AES DESIGN ON VARIOUS DEVICES.

Devices	Slices	Fmax(MHz)	Throughput(Mbps)
XCV1000	1572	76.237	195.16
XC2Vp20	1729	164.564	421.28
XC2v2000	1729	132.424	399
XC3s2000	1766	91.293	233.71
XC4vlx40	1725	194.250	497.28

VI. CONCLUSION

In this paper, efficient sub-pipelined architectures of the AES algorithm, with unrolling and pipelining to explore the design space is presented to tailor the performance and area requirements. Offline key Expansion is used in order to reduce memory requirements and save power. Inner-round pipelining and thorough scheduling allow high frequencies to be achieved and efficient usage of resources. In order to explore the advantage of sub-pipelining further, the SubBytes/InvSubBytes is implemented by combinational logic to avoid the unbreakable delay of LUTs in the traditional designs.

The resulting implementation has moderate area demands in terms of CLB slices, low memory requirements and achieves throughputs in the range of 26 Gbps. Compared to other academic and commercial implementations, the presented design demonstrated the highest throughput and one

of the smallest memory/area to performance ratios. Optimization approaches for the implementations supporting multiple key lengths and modes of operation require further study.

VII. REFERENCES

[1] J. Daemen and V. Rijmen, AES submission document on Rijndael, Version 2, September 1999. (http:// csrc.nist.gov/ CryptoToolkit/aes/rijndael/Rijndael.pdf)

[2] "Advanced Encryption Standard (AES)," Federal Information Processing Standards Publication 197, Nov. 26, 2001.

[3] A. J. Menezes, P. C. van Oorschot and S. A. Vanstone, Handbook of Applied Cryptography. CRC Press, 1996.

[4] FX. Standaert, G. Rouvoy, JJ. Quisquater, JD. Legat. Efficient Implementation of Rijndael Encryption in Reconfigurable Hardware: Improvements and Design Tradeoffs. In the proceedings of CHES 2003, Lecture Notes in Computer Science, vol 2779, pp. 334-350, Springer-Verlag

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

[6] A. Elbirt, W. Yip, B. Chetwynd, and C. Paar. An FPGA-based performance evaluation of the AES block cipher candidate algorithm finalists. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 9:545–557, August 2001.

[7] A.Dandalis et al, A Comparative Study of Performance of AES Candidates Using FPGA's, The Third Advanced Encryption Standard (AES3) Candidate Conference, April 13-14 2000, New York, USA.

[8] V. Fischer and M. Drutarovsky, "Two Methods of Rijndael Implementation in Reconfigurable Hardware", Proceedings CHES 2001, pp. 77–92, Paris, France, May2001

[9] K. Gaj and P. Chodowiec, "Comparison of the Hardware Performance of the AES Candidates Using Reconfigurable Hardware", The Third AES Conference (AES3), New York, April 2000..

[10] M. McLoone and J. V. McCanny,"Rijndael FPGA Implementation Utilizing Look-Up Tables", IEEE Workshop on Signal Processing Systems, pp. 349–360,September 2001.

[11] M. McLoone and J.V.McCanny, "High Performance Single-Chip FPGA Rijndael Algorithm Implementation", Proceedings CHES 2001, pp. 65–76, Paris, France,May2001.

[12] G. P. Saggese, A. Mazzeo, N. Mazocca, and A. G. M. Strollo, "An FPGA based performance analysis of the unrolling, tiling and pipelining of the AES algorithm," in Proc. FPL 2003, Portugal, Sept. 2003.

[13] T. F. Lin, C. P. Su, C. T. Huang and C. W. Wu, "A High-throughput low-cost AES cipher chip," in Proc. IEEE Asia-Pacific Conference on ASIC, pp. 85 88, 2002.

[14] A. Hodjat, W. Ingrid, "A 21.54 Gbits/s fully pipelined processor on FPGA," 12th Annual IEEE Symposium on Field-Programmable Custom Computing Machines, pp. 308 - 309, April 2004.

[15] C.C. Lu and S.Y. Tseng, "Integrated design of AES (Advanced Encryption Standard) encrypter and decrypter Application -Specific Systems,"*The IEEE International* Conference on Application-Specific Systems, Architectures and Processors, *2002*, pp. 277 –285, 2002.

[16] N.Sklavos and O.Koulopavlou,"Architecture and VLSI Implemenation of the AES-Proposal Rijindael" in IEEE Transactions on computers,Vol.51, No.12,December 2002,pp 1454-1459.

[17.] S.-F Hsiao and M.-C. Chen, "Efficient substructure sharing methods for optimizing the inner-product operations in Rijndael advanced encryption standard", IEE Proc.-comput. Digit. Tech., Vol. 152, No. 5, September 2005, IEE Proceedings online no. 2004, 5152.

[18] Shen-Fu Hsiao, Ming-Chih Chen, and Chia-Shin Tu, "Memory-Free Low-Cost Designs of Advanced Encryption Standard Using Common Subexpression Elimination for Subfunctions in Transformations", IEEE Transactions on circuits and systems-I: Regular Papers, Vol. 53, No.3, March 2006.

[19] Altera. APEX II Programmable Logic Device Family Data Sheet. www.altera.com/literature/ds/ds ap2.pdf.

[20]Virtex-E. Xilinx' Virtex-E Datasheet. www.xilinx.com/partinfo/ds022.pdf.

[21] Xilinx: Virtex 2.5V Field Programmable Gate Arrays Data Sheet, http://www.xilinx.com.

[22] Implementation Approaches for the Advanced Encryption Standard Algorithm by Xinmiao Zhang and Keshab K. Parhi

[23] Efficient Implementation of the Rijndael S-box, Vincent Rijmen,Katholieke Universiteit Leuven, Dept. ESAT,Kard. Mercierlaan 94,B{3001 Heverlee,Belgiumvincent.rijmen@esat.kuleuven.ac.be

[24] A Compact Rijndael Hardware Architecturewith S-Box Optimization,Akashi Satoh, Sumio Morioka, Kohji Takano, and Seiji Munetoh IBM Research, Tokyo Research Laboratory, IBM Japan Ltd., 1623-14,Shimotsuruma, Yamato-shi, Kanagawa 242-8502, Japan {akashi,e02716,chano,munetoh}@jp.ibm.com.

[25] H. Kuo and I. Verbauwhede, "Architectural Optimization for a 1.82Gbits/sec VLSI Implementation of the AES Rijndael Algorithm," *Proc. CHES 2001*, pp. 51–64, Paris, France, May 2001.

[26] K. U. Jarvinen, M. T. Tommiska and J. O. Skytta, "A fully pipelined memoryless 17.8 Gbps AES-128 encryptor," *Proc. International Symposium on Field-Programmable Gate Arrays (FPGA 2003)*, pp. 207-215, Monterey, CA, Feb. 2003.

[27] C. C. Lu and S. Y. Tseng, "Integrated Design of AES (Advanced Encryption Standard) Encrypter and Decrypter," *Proc. The IEEE International Conference on Application Specific Systems, Architectures and Processors*, pp. 277-285, 2002.

[28] X. Zhang and K. K. Parhi, "Implementation Approaches for the Advanced Encryption Standard Algorithm," *IEEE Circuits and Systems Magazine*, vol.2, Issue.4, pp. 24-46, Fourth Quarter 2002.

VII. BIOGRAPHIES

Nalinil C Iyer, has completed her M.Tech from KREC Surathkal, India and is now pursuing her PhD in the area of Cryptography. She is presently working as Assistant Professor dept. of Electronics and communication, BVB Engg. College, Hubli, Karnataka. Her areas of interests are IP for Network security, advanced communication.