IJIRSET
International Journal of Innovative Research in
SCIENCE | ENGINEERING | TECHNOLOGY

e-ISSN: 2319-8753 | p-ISSN: 2347-6710

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

Volume 13, Issue 7, July 2024

ISSN INTERNATIONAL STANDARD SERIAL NUMBER INDIA

Impact Factor: 8.423

9940 572 462 | 6381 907 438 | ijirset@gmail.com | www.ijirset.com

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.423

|www.ijirset.com |A Monthly, Peer Reviewed & Referred Journal| e-ISSN: 2319-8753; p-ISSN: 2347-6710|

Volume 13, Issue 7, July 2024

|DOI: 10.15680/IJIRSET.2024.1307057|

COPMA: Compact and Optimized Polynomial Multiplier Accelerator for High-Performance Implementation LWR-based PQC

D.Anbarasu, A.Ramya

PG Student, Department of Electronics and Communication Engineering, Thiruninravur, India

Assistant Professor, Department of Electronics and Communication Engineering, Jaya Engineering College, Thiruninravur, India

ABSTRACT: Cryptographic computation exploits finite field arithmetic and, in particular, multiplication. Lightweight and fast implementations of such arithmetic are necessary for many sensitive applications. In this paper, we have shown that the AOP-based systolic multipliers can easily achieve low register-complexity implementations and the proposed architectures can be employed as computation cores to derive efficient implementations of systolic Montgomery multipliers based on trinomials. The rapid progress in quantum computing has initiated a new round of cryptographic innovation, that is, developing postquantum cryptography (PQC) to resist attacks from well-established quantum computers. In this brief, we propose a novel compact and optimized polynomial multiplier accelerator (COPMA) for First, we propose a novel data broadcasting scheme in which the register complexity involved within existing AOP-based systolic multipliers is significantly reduced. We have found out that the modified AOP-based structure can be packed as a standard computation core. Next, we propose a novel Montgomery multiplication algorithm that can fully employ the proposed AOP-based computation core. The proposed Montgomery algorithm employs a novel pre computed modular operation, and the systolic structures based on this algorithm fully inherit the advantages brought from the AOP-based core (low register complexity, low critical-path delay, and low latency) except some marginal hardware overhead brought by a pre computation unit. The proposed architectures are then implemented by Xilinx ISE 14.1 using Verilog HDL and it is shown that compared with the existing designs, the proposed designs achieve at less area-delay product and power delay product than the best of competing designs, respectively.

I. INTRODUCTION

Finite field multipliers over GF (2^m) have wide applications in elliptic curve cryptography (ECC) and error control coding systems. Polynomial basis multipliers are popularly used because they are relatively simple to design, and offer scalability for the fields of higher orders. Efficient hardware design for polynomial-based multiplication is therefore important for real-time applications. All-one polynomial (AOP) is one of the classes of polynomials considered suitable to be used as irreducible polynomial for efficient implementation of finite field multiplication.

Multipliers for the AOP-based binary fields are simple and regular, and therefore, a number of works have been explored on its efficient realization. In this paper, we have presented a novel register sharing technique to reduce the register requirement in the systolic structure. The proposed algorithm not only facilitates sharing of registers by the neighboring PEs to reduce the register complexity but also helps reducing the latency. Cut-set retiming allows introducing certain number of delays on all the edges in one direction of any cut-set of a signal flow-graph (SFG) by removing equal number of delays on all the edges in the reverse direction of the same cut-set. When all the edges are in a single direction, one can introduce any desired number of delays on all the edges of any cut-set of an SFG. Therefore, this technique is highly useful for pipelining digital circuits to reduce the critical path. Application such as error-correcting code switching theory, and cryptography. Important operations in are addition, multiplication, exponentiation, and division. Since addition in is bit independent XOR operation, it can be implemented in fast and inexpensive ways. The other operations are much more complex and expensive. This paper focuses on the hardware implementation of a fast and low-complexity digit-serial multiplier over, since computing exponentiation and division can be performed by repeated multiplications. Many approaches and architectures have been proposed to perform multiplication. In those implementations, many architectures applied systolic array concept. In general, a non systolic

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.423

|www.ijirset.com |A Monthly, Peer Reviewed & Referred Journal| e-ISSN: 2319-8753; p-ISSN: 2347-6710|

Volume 13, Issue 7, July 2024

|DOI: 10.15680/IJIRSET.2024.1307057|

architecture has global signals. Therefore, if becomes large, propagation delay also increases. A systolic architecture, however, does not suffer from the problem. This is because the systolic architecture consists of replicated basic cells and each basic cell is connected with its eighboring cells through pipelining,i.e., there are no global signals. Consequently, the systolic architecture is a better choice the non systolic architecture for a high-speed VLSI implementation The most commonly used basis representations are dual, normal, and standard basis. Multipliers using the dual and normal basis representations require a basis conversion, in which complexity heavily depends on the irreducible polynomial In contrast, multipliers that use the standard basis do not require a basis conversion; they are therefore more efficient from the point of view of irreducible polynomial selection and hardware optimization The array type multiplication algorithms, when the standard basis is used, are classified as least significant bit (LSB) first and most significant bit (MSB) first schemes The LSB first scheme processes the least significant bit of the multiplier operand first, while the MSB-first scheme processes its most significant bit first. For multiplication, the multiplier implemented using the LSB-first scheme has lower critical path delay compared to the multiplier based on the MSB-first scheme Additionally, these multipliers for can be further classified into four types: bit-serial, bit-parallel, hybrid, and digit-serial architecture. The bit-serial architecture, which processes one bit of input data per clock cycle, is area-efficient and suitable for low-speed applications. The bit-parallel architecture, which processes one whole word of input data per clock cycle, is ideal for high-speed applications when pipelined at the bit-level. These architectures are typical examples of the area–speed tradeoff paradigm. But these architectures may not be suitable for applications that require moderate sample rates. In other words, the bit-serial architecture may be too slow and the bit-parallel architecture with bit-level pipelining may be faster than necessary and require too much area. To avoid the extreme tradeoffs required by the bit-serial and bit-parallel architectures, hybrid and digit-serial architectures have recently been proposed.. Suppose that word size is -bits, digit size is -bits, and ; then the digit-serial system processes input data at a rate of -bits per clock cycle and yields output results at a rate of one every clock cycles. For the architecture proposed by Paar et al. the number of gates (AND, XOR) and registers can be drastically reduced by selecting special irreducible polynomials such as trinomials or pentanomials. The disadvantage of this architecture is that cryptosystems based on the discrete logarithm in might have weaknesses if is composite, making hybrid architectures unsuitable for this application. The architecture proposed by Song and Parhi has less computational delay time and power consumption than the other architectures. However, its disadvantage is that the range of the digit size depends on the selected irreducible polynomial. The architectures of Guo and Wang and Mekhallalati et al. The two most well known bit-serial architectures are the Berlekamp multiplier (BM) and the Massey-Omura multiplier (MOM). Of these, the BM has lower hardware requirements and an easy-to-derive structure based on the defining irreducible polynomial for the field fix). The BM is also particularly suited to applications where constant multiplication is required such as RS encoders. The only dk advantage of the BM is that basis The MOM operates over just one basis, the normal basis, which has the advantage that the square of a field element can be generated by a cyclic shift of the basis coefficients. However, the MOM requires more hardware than the BM, cannot efficiently carry out constant multiplication and, for a given normal basis, it is not obvious what the defining boolean function for the multiplier.

II. LOW-COMPLEXITY MULTIPLIER FOR GF (2^m) BASED ON ALL-ONE POLYNOMIALS

Finite field multipliers over GF (2^m) have wide applications in elliptic curve cryptography (ECC) and error control coding systems. Polynomial basis multipliers are popularly used because they are relatively simple to design, and offer scalability for the fields of higher orders. Efficient hardware design for polynomial-based multiplication is therefore important for real-time applications. All-one polynomial (AOP) is one of the classes of polynomials considered suitable to be used as irreducible polynomial for efficient implementation of finite field multiplication. Multipliers for the AOP-based binary fields are simple and regular, and therefore, a number of works have been explored on its efficient realization. In this paper, we have presented a novel register sharing technique to reduce the register requirement in the systolic structure. Then, two low-complexity bit parallel systolic multipliers are presented based on the algorithm. The first multiplier is composed of m. identical cells, each consisting of one 2-input AND gate, one 2-input XOR gate, and three 1-bit latches. The other multiplier is composed of identical cells and m XOR gates. Each cell consists of one 2-input AND gate, one 2-input XOR gate, and four 1-bit latches. Each multiplier is very fast because the propagation delay in each cell is very short. Moreover, the latency is only m 1 clock cycles. The architectures of these two new multipliers can also be applied in computing multiplications over the class of fields GF.2mThe polynomial (standard) basis (PB) multipliers are widely used and suited for hardware and software implementations. However, the normal basis multipliers are complex implementations in VLSI architectures; since a normal basis multiplication is determined by Inversion can be considered as a special case of exponentiation because they can be realized by the Fermat's theorem. The structure of PE [0] is shown in Fig. 1. It consists of an AND cell and a BSC. Each XOR cells and AND

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.423

|www.ijirset.com |A Monthly, Peer Reviewed & Referred Journal| e-ISSN: 2319-8753; p-ISSN: 2347-6710|

Volume 13, Issue 7, July 2024

|DOI: 10.15680/IJIRSET.2024.1307057|

cells in the PE consists of (m+1) number of gates working in parallel. The regular PE, as shown in Fig. 2, consists of three basic cells, e.g., the bit-shift cell (BSC), the AND cell, and the XOR cell. The PE[m+1] of the systolic structure in Fig.3 consists of only an XOR cell, which performs bit-by-bit XOR operations of its pair of m-bit inputs.

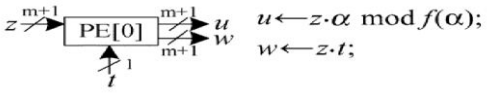


Fig .1 PE[0]

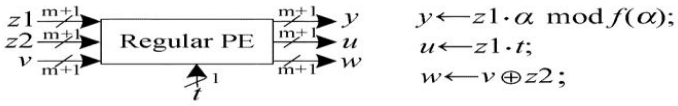
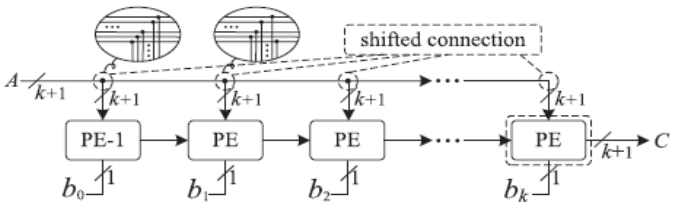


Fig .2 Regular PE

MS-I and MS-II, a shifted connection strategy is used in which the input A is directly fed to each PE and thus reduces the registers required. Moreover, the details of shifted connection are also shown in Figs. 3 and 4. To reduce the complexity further, we have used NAND and XNOR gates to replace the original AND and XOR gates, as depicted. It is noted that for AOP-based multiplication, the last PE (inside the dotted box) can be removed as bk = 0. The modified structures involve nearly the same time complexity as the previous ones, but the register complexity is significantly reduced.



Modified Scheme

III. SYSTOLIC STRUCTURE

A structure to realize symmetric functions in binary reversible and quantum logic has been presented by Perkowski and Kerntopf. the structure considered by them is not completely regular but semi-regular, therefore it is not systolizable. Here our motivation is twofold; first, develop the structure to have a regular structure, and then derive the modular and scalable reversible systolic structure which has a smaller number of garbage than that of Perkowski and Kerntopf. The proposed structure and corresponding systolic structure consisting of two planes to realize arbitrary reversible symmetric functions are shown respectively. The Feynman gates in the first plane are used to make the first plane has regular structure. The second plane is a plane of Feynman gates that uses their internal EXOR gates to realize every output function as an EXOR of symmetric functions. Horizontal outputs from the first plane are EXOR-ed using Feynman gates in the second plane to create arbitrary symmetric functions at the bottoms. Additional garbage outputs of Kerntopf gates must be forwarded to the primary outputs to satisfy reversibility. The programmability is obtained in the second plane in the form of interconnecting or disconnecting the Feynman gate to generate certain symmetric functions at the bottom of second plane.

IV. SOFTWARE REQUIREMENTS

- Verification Tool
- Modelsim 6.4c
- Synthesis Tool
- Xilinx ISE 9.1

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

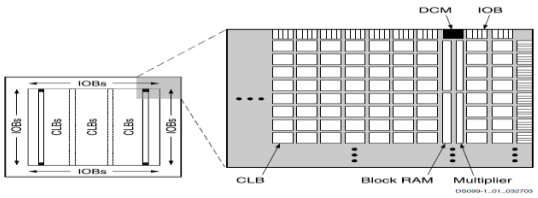
Impact Factor 8.423

|www.ijirset.com |A Monthly, Peer Reviewed & Referred Journal| e-ISSN: 2319-8753; p-ISSN: 2347-6710|

Volume 13, Issue 7, July 2024

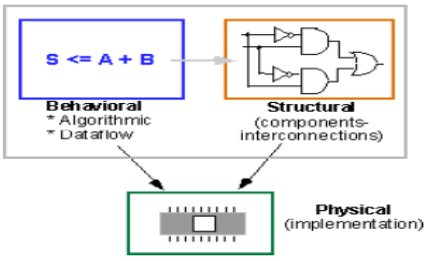
|DOI: 10.15680/IJIRSET.2024.1307057|

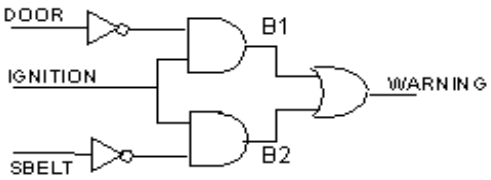
Digital Clock Manager (DCM) blocks provide self-calibrating, fully digital solutions for distributing, delaying, multiplying, dividing, and phase shifting clock signals. These elements are organized as shown in Figure. A ring of IOBs surrounds a regular array of CLBs. The XC3S50 has a single column of block RAM embedded in the array. Those devices ranging from the XC3S200 to the XC3S2000 have two columns of block RAM. The XC3S4000 and XC3S5000 devices have four RAM columns. The Spartan-3 family features a rich network of traces and switches that interconnect all five functional elements, transmitting signals among them. Each functional element has an associated switch matrix that permits multiple connections to the routing.



V. SIMULATION IMPLEMENTATION

The dramatic increase in the logic density of silicon chips has made it possible to implement digital systems with multimillion gates on a single chip. The complexity of such systems makes it impractical to use traditional design descriptions (e.g., logic schematics) to provide a complete and accurate description of a design. Currently, all complex digital designs are expressed using a hardware description language (HDL). An HDL, unlike traditional programming languages such as C or C++, can describe functions that are inherently parallel. A major advantage of an HDL is that it provides a better and more concise documentation of a design than gate-level schematics. Two very popular HDLs are VHDL and VERILOG. In this text we use VHDL. VHDL includes facilities for describing logical structure and function of digital systems at a number of levels of abstraction, from system level down to the gate level. It is intended, among other things, as a modeling language for specification and simulation.





RS ENCODER PROGRAM:

```
M_GF_Multiplier M10 (
 .A(A),
 .B(Code1),
 .C(Mul0)
 );
  M_GF_Multiplier M11 (
 .A(A),
 .B(Code2),
 .C(Mul1)
 );
M_GF_Multiplier M12 (
 .A(A),
 .B(Code3),
 .C(Mul2)
 );
 M_GF_Multiplier M13 (
```

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.423

|www.ijirset.com |A Monthly, Peer Reviewed & Referred Journal| e-ISSN: 2319-8753; p-ISSN: 2347-6710|

Volume 13, Issue 7, July 2024

|DOI: 10.15680/IJIRSET.2024.1307057|

```
 .A(A),
 .B(Code4),
 .C(Mul3)
 );
M_DFF M30 (
 .Clk(Clk),
 .Rst(Rst),
 .D(Mul0),
 .Q(Q0)
 );
M_GF_Adder M20 (
 .A(Q0),
 .B(Mul1),
 .C(Add1)
 );
M_DFF M31 (
 .Clk(Clk),
 .Rst(Rst),
 .D(Add1),
 .Q(Q1)
 );
M_GF_Adder M21 (
 .A(Q1),
 .B(Mul2),
 .C(Add2)
 );
M_DFF M32 (
 .Clk(Clk),
 .Rst(Rst),
 .D(Add2),
 .Q(Q2)
 );
 M_GF_Adder M22 (
 .A(Q2),
 .B(Mul3),
 .C(Add3)
 );
assign A=C23&Cont;
M_Mux M26 (
 .A(Q3),
 .B(In),
 .S(Cont),
 .Out(Out)
 );
```

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

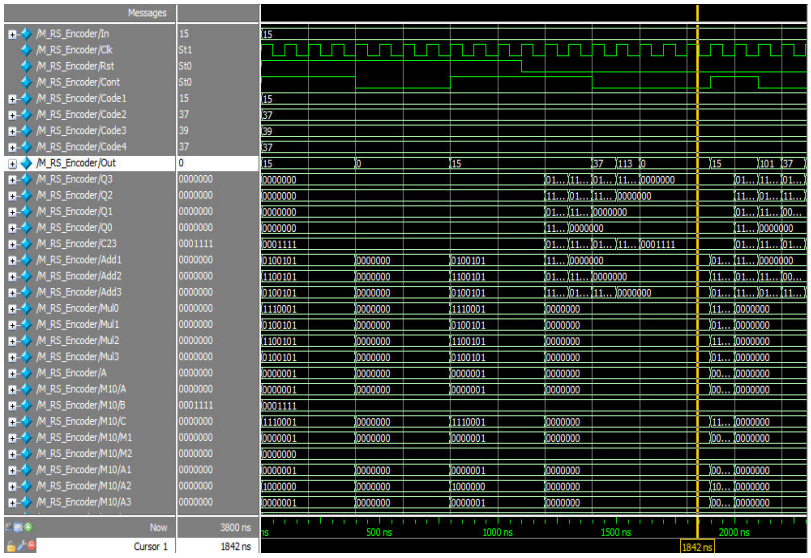
Impact Factor 8.423

|www.ijirset.com |A Monthly, Peer Reviewed & Referred Journal| e-ISSN: 2319-8753; p-ISSN: 2347-6710|

Volume 13, Issue 7, July 2024

|DOI: 10.15680/IJIRSET.2024.1307057|

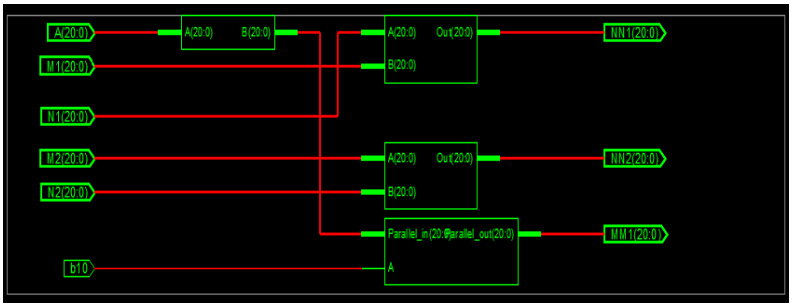
RS ENCODER USING OUR PROPOSED MULTIPLIER



DEVICE UTILIZATION SUMMARY:

Device Utilization Summary				
Logic Utilization	**Used**	**Available**	**Utilization**	**Note(s)**
Number of 4 input LUTs	399	7,168	5%	
Logic Distribution				
Number of occupied Slices	200	3,584	5%	
Number of Slices containing only related logic	200	200	100%	
Number of Slices containing unrelated logic	0	200	0%	
Total Number of 4 input LUTs	399	7,168	5%	
Number of bonded IOBs	63	141	44%	
Total equivalent gate count for design	2,835			
Additional JTAG gate count for IOBs	3,024			

RTL SCHEMATIC



International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

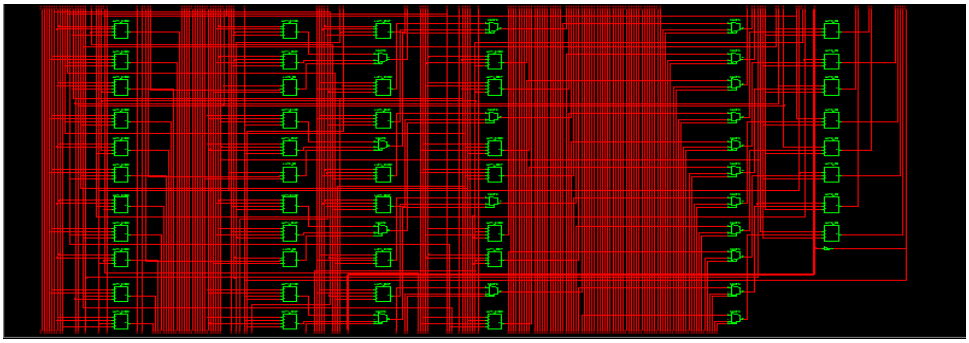
Impact Factor 8.423

|www.ijirset.com |A Monthly, Peer Reviewed & Referred Journal| e-ISSN: 2319-8753; p-ISSN: 2347-6710|

Volume 13, Issue 7, July 2024

|DOI: 10.15680/IJIRSET.2024.1307057|

TECHNOLOGY DIAGRAM



VI. CONCLUSIONS

An efficient, new scheme for low-complexity implementation of finite field multipliers over GF(2m) based on trinomials benchmarked on the FPGA platform has been proposed. We have proposed a modified data broadcasting technique to reduce the register complexity within the existing AOP-based multipliers. Then, the AOP-based multipliers have been packed as standard computation cores to be used for trinomial-based multipliers. Moreover, a novel low register-complexity Montgomery multiplication algorithm for systolic trinomial-based finite field multipliers is presented. The systolic multiplier based on the proposed algorithm can employ the AOP-based computation core to offer low register-complexity implementations. We have also introduced structures for low-latency and digit-parallel implementations. Both the theoretical analysis and the FPGA implementation results have confirmed the higher efficiency of the proposed architectures compared with the competing ones.

REFERENCES

[1] I. F. Blake, G. Seroussi, and N. Smart, Elliptic Curves in Cryptography (London Mathematical Society Lecture Note Series). Cambridge, U.K.: Cambridge Univ. Press, 1999.
[2] N. R. Murthy and M. N. S. Swamy, "Cryptographic applications of Brahmagupta–Bhˇaskara equation," IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 53, no. 7, pp. 1565–1571, Jul. 2006.
[3] M. Sun et al., "eButton: A wearable computer for health monitoring and personal assistance," in Proc. 51st Design Autom. Conf., 2014, pp. 1–6.
[4] D. Schinianakis and T. Stouraitis, "Multifunction residue architectures for cryptography," IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 61, no. 4, pp. 1156–1169, Apr. 2014.
[5] FIPS 186-2, Digital Signature Standard (DSS), Federal Information Processing Standards Publication 186-2, Nat. Inst. Standards Technol., Gaithersburg, MD, USA, 2000.
[6] K. K. Parhi, VLSI Digital Signal Processing Systems: Design and Implementation. New York, NY, USA: Wiley, 1999.
[7] C.-Y. Lee, J.-S. Horng, I.-C. Jou, and E.-H. Lu, "Low-complexity bitparallel systolic Montgomery multipliers for special classes of GF(2m)," IEEE Trans. Comput., vol. 54, no. 9, pp. 1061–1070, Sep. 2005.
[8] P. K. Meher, "Systolic and super-systolic multipliers for finite field GF(2m) based on irreducible trinomials," IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 55, no. 4, pp. 1031–1040, May 2008.
[9] J. Xie, P. K. Meher, and J. He, "Low-latency area-delay-efficient systolic multiplier over GF(2m) for a wider class of trinomials using parallel register sharing," in Proc. IEEE Int. Sym. Circuits Syst., May 2012, pp. 89–92.
[10] S. Bayat-Sarmadi and M. Farmani, "High-throughput low-complexity systolic Montgomery multiplication over GF(2m) based on trinomials," IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 62, no. 4, pp. 377–381, Apr. 2015.
[11] J. Xie, J. J. He, and P. K. Meher, "Low latency systolic Montgomery multiplier for finite field GF(2m) based on pentanomials," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 21, no. 2, pp. 385–389, Feb. 2013.
[12] P. L. Montgomery, "Modular multiplication without trial division," Math. Comput., vol. 44, no. 170, pp. 519–521, 1985.
[13] R. Azarderakhsh, K. Järvinen, and V. Dimitrov, "Fast inversion in GF(2m) with normal basis using hybrid-double multipliers," IEEE Trans. Comput., vol. 63, no. 4, pp. 1041–1047, Apr. 2014.
[14] R. Azarderakhsh, D. Jao, and H. Lee, "Common subexpression algorithms for space-complexity reduction of Gaussian normal basis multiplication," IEEE Trans. Inf. Theory, vol. 61, no. 5, pp. 2357–2369, May 2015.

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

INNO SPACE SJIF Scientific Journal Impact Factor

doi cross ref

निस्केयर NISCAIR

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

9940 572 462 6381 907 438 ijirset@gmail.com

www.ijirset.com

Scan to save the contact details