Area Efficient Implementation of GF (2m)Multipliers For Finite Fields.

B. Dhivya[1], N. Divya[2], R.Anisha[3], S. Dhivya[4] S.M.Swaminathan[5]
[1 2 3 4] UG student, ECE department, SNS college of technology, Coimbatore-641034.
[5] Assistant professor, ECE department, SNS college of technology, Coimbatore-641-34
{divyabalachander1902 , divyanallathambi.5.10 , anisharavi101 , dhivyashankar14 , mmsuresh6 }@gmail.com

Abstract:**This paper presents an irreducible all-one polynomial (AOP) based on area-time-efficient systolic structure for multiplication over *GF* (2^m).A novel cut-set retiming to minimize the duration of the criticalpath to one XOR gate delay is used. The systolic structure can be decomposed into two or more parallel systolic branches, where the pair of parallel systolic branches has the same input operand, and they can share the same input operand registers is further shown. Specific integrated circuit and field-programmable gate array (FPGA) synthesis results from the application. The proposed design provides significantly less area-delay and power-delay complexities over the best of the existing designs has been found.**

Keywords: All-One Polynomial, Systolic Design, finite field, FPGA.

I. INTRODUCTION

The wide applications are elliptic curve cryptography (ECC) and error control coding systems in finite field multipliers over *GF* (2*m*)[2], [3]. 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 [4]–[6]. All-one polynomial (AOP) is one of the classes of polynomials to be used as irreducible polynomial for efficient implementation of finite field multiplication. Multipliers for the AOP-based binary fields are easy and regular and therefore, on its efficient realization a number of works have been explored [7]. Irreducible AOPs are not in huge number. One has to make careful choice of the field order to use irreducible AOPs for cryptographic application and they are very often not preferred in cryptosystems for security reasons [2], [10]. The AOPbased multipliers can be used for the NAOP (Nearly AOP) which could be used for efficient realization of ECC systems. AOP-based fields can also be used for efficient implementation of Reed-Solomon encoders. Besides, the AOP-based architecture can be used as a kernel circuit for field exponentiation, inversion and division architecture.

Systolic design is a widely chosen type of specialized hardware solution due to its high-level of pipeline ability, local connectivity and many other profitable features. In a bitparallel AOP-based systolic multiplier has been suggested by Lee etal. Another efficient systolic design is presented. In a recent paper, a low-complexity bit-parallel systolic Montgomery multiplier has been recommended. Lately, an efficient digit-serial systolic Montgomery multiplier for AOP-based binary extension field is presented. The systolic structure for field multiplication have two major problems. Intially, the registers in the systolic structures usually consume large area and power. Second, the systolic structures usually have a latency of nearly *m* cycles, which is very often not suitable for real-time applications.Therefore, in this paper, we have presented a novel register- sharing technique to minimize the register requirement in the systolic structure. The proposed algorithm not only facilitates sharing of registers by the neighboring PEs to minimize the register complexity but also helps minimize the latency. Cut-set retiming allows introducing certain number of delays on every edges in one direction of any cut-set of a signal flowgraph (SFG) by avoiding equal number of delays on all the edges in the opposite direction of the same cut-set. When all the edges are in a one direction, one can introduce any desired number of delays on every edges of any cut-set of an SFG.

Therefore, this technique is highly useful for pipelining digital circuits to minimize the complexed path. In this paper, we have proposed a novel cut-set retiming approach to minimize the clock-period. The proposed structure is found to involve significantly low area-time-power complexity compared with the already existing designs. The rest of this paper is organized as follows. In Section II .And III we have Proposed Structure and Sequential Polynomial Multiplier.In section IV, we have listed the complexities and compared them with those of the already existing structures. At last the conclusion is given in Section V.

II .PROPOSED STRUCTURE

In this section, we derive a basic systolic design followed by the recommended register sharing structure.

A. Elementary Systolic Design

The operations can be performed recursivelyfor systolic implementation of multiplication over GF (2^m). Each recursion is consists of three steps i.e., modular reduction of bit-multiplication of bit-addition. Equation represented by the SFG (shown in Fig. 1) consisting of m modular reduction nodes R (i) and m addition nodes A (i) for $1 \le i \le m$, and (m+1) multiplication nodes M (i) for $1 \le i \le m+1$. The functions of these nodes are shown in Fig. 1(b)– (d). Node R (i) performs the modular reduction of degree by one according to node M (i) performs an AND operation of a bit of operand B with a minimized form of operand A, according. Node A (i) performs the bit-addition operation according to, as shown in Fig. 1(d), where *Ci* is the partial result available to the node.

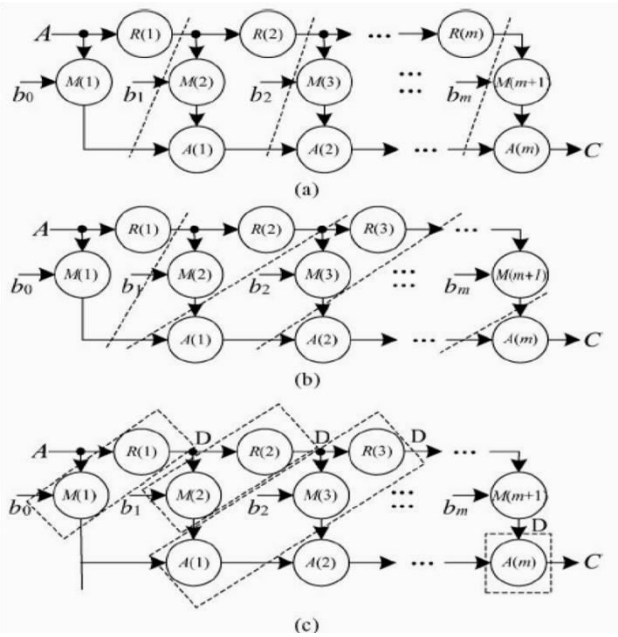

Fig.1. SFG of the algorithm (a) The SFG (b) Function of node R (i) (c) Function of node M (i). (d) Function of node A(i).

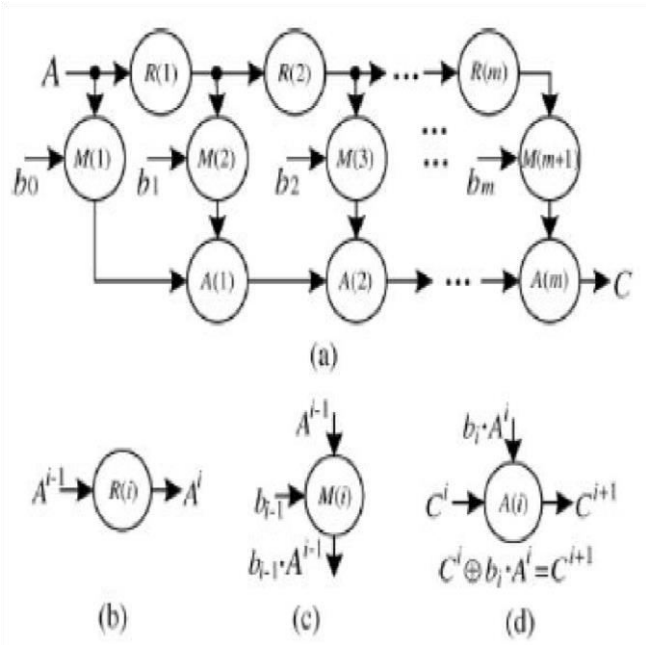

Fig.2. Cut-set retiming of the SFG (a) Cut-set retiming in a general way (b) Proposed cut-set retiming. (c) Formation of PE. "D" denotes unit delay.

Commonly, we can introduce a delay between the reduction node and its corresponding bit-multiplication and bit-addition nodes, as shown in Fig. 2(a), such that the critical-path is not larger than ($T_A + T_X$), where the $T_A$refer the propagation delay of AND gate and T_X refer the propagation delay of the XOR gate. In this section, a novel cut-set retiming to minimize the critical-path of a PE to $T_X$is introduced. The node R(i) performs only the bit-shift operation is observed and so it does not involve any time consumption. Hence, a critical-path which is not larger than $T_X$is introduced as shown in Fig. 2(b). To derive the basic design of a systolic multiplier, the formation of PE of the retimed SFG is shown in Fig. 2(c). It can be examined that the cut-set retiming allows to perform a reduction operations, bit-addition and bit-multiplication simultaneously, so that the critical- path is minimized to max $\{T_A, T_M, T_R\}$, where $T_A$is computation times of bit-addition nodes $T_M$is the bit multiplication nodes and $T_R$is the reduction nodes.In Fig.3 the basic design of systolic multiplier. It consists of (m+2) PEs and the functions of the PEs are shown in Fig. 3. During each cycle period, the regular PE (from PE to PE[m-1]) not only perform the modular reduction operation and it also perform the bit-multiplication and bit-addition operations simultaneously. The detail circuit of a regular PE is shown in Fig. 4.

As shown in Fig. 4(a),the regular PE consists of three basic cells, e.g., the bit-shift cell (BSC), the AND cell, and the XOR cell. The AND cell correspond to the node M (i), and the XOR cell corresponds to node A (i) of the SFG of Fig. 1, respectively. The structure of PE of Fig. 3 is shown in Fig. 4(b). It comprises of an AND cell and a bit-shift cell(BSC). Each XOR cells and AND cells in the PE consists of (m+1) number of gates working in parallel. An example of AND cell for m = 4 is shown in Fig. 4(c).In Fig. 3 the PE [m+1] of the systolic structure consists of only an XOR cell, as shown in Fig. 4(d), which executes bit-by-bit XOR operations of its pair of m-bit inputs. According to (11) the BSC in the PE performs the bit-shift

operation. An example of the structure of BSC (of PE of Fig. 4) is shown in Fig. 4(e) for m = 4. One can obtain Ai directly from A0 for $1 \leq i \leq m$, i.e., every PE of the structure of Fig. 3 can have the same input operand A0, and Ai can be obtained from the BSC after A0 is fed as input. Hence, we can change the circuit-designs of Fig. 4(a) and (b) into the form of Fig. 4(f) and (g) respectively. Moreover, the operation of node R (i) does not involve any area and time-consumption. Hence, the minimum duration of clock-period of a regular PE amounts to max $\{T_A, T_X\} = T_X$.

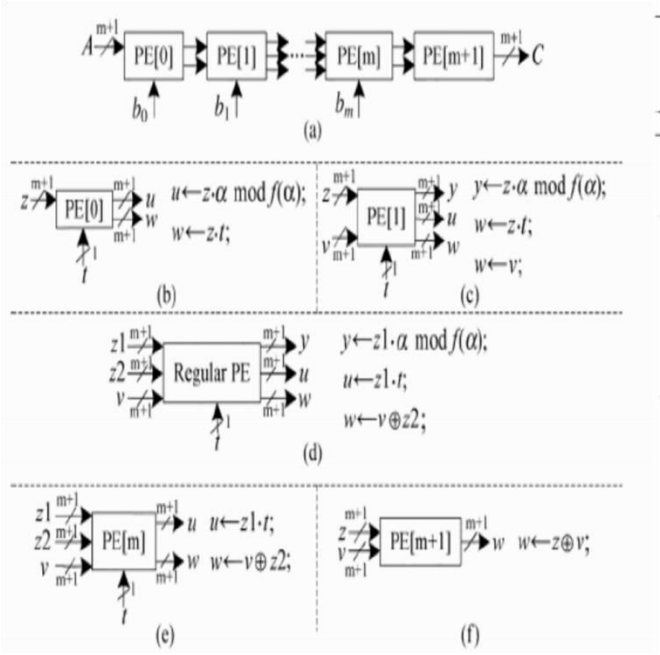

Fig.3. Proposed systolic structure (a) Systolic design (b) Function of PE. (c) Function of PE. (d) Function of regular PE (from PE to PE [m-1]). (e) Function of PE[m]. (f) Function of [m+1].

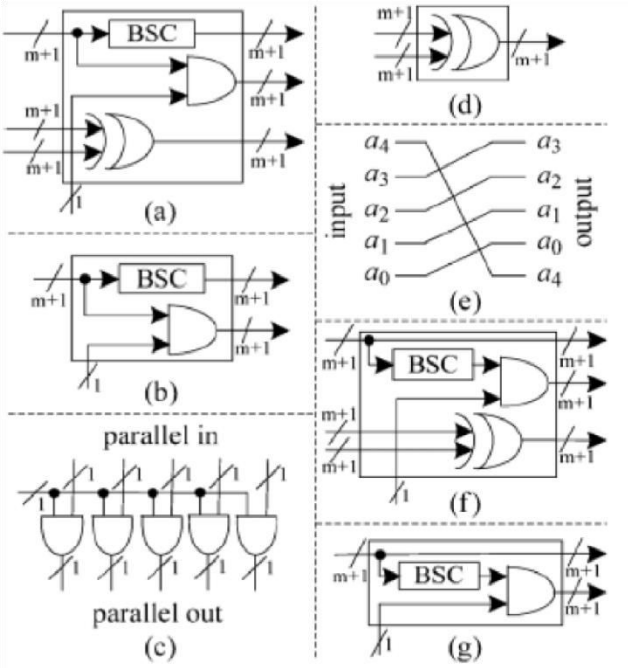

Fig.4. Structure of PEs (a) internal structure of a regular PE. (b) Internal structure of PE [0] of Fig. (c) An example of AND cell for m = 4. (d) Structure of the AC. (e) Structure of BSC where m = 4. (f) Alternate structure of a regular PE. (g) Alternate structure of PE [0].

III. SEQUENTIAL POLYNOMIAL MULTIPLIER

Even with the hRAIK method combinatorial multiplier with long bit sizes are still quite slow and not as small as it is required for mobile devices or even wireless sensor nodes. Common approaches use lesser combinatorial multiplication units and serialize the multiplication. The original iterative Karatsuba multiplier (IKM) approach was presented as solution for this purpose. It uses lesser combinatorial multiplication blocks and applies them repetitively following the Karatsuba method. In order to perform a larger polynomial multiplication the IKM design for a 233 bit multiplication unit presented in the starting point for the investigation concerning improved IKM design. It consists of three main parts: The selection logic selects and combines the factors of the partial multiplication, within one clock cycle the partial multiplier performs the partial multiplication and the accumulation logic computes the final product by accumulating the partial products. The amount of clock cycles depends on the size of the segmentation. For our 233 bit ECC(elliptic curve cryptography) design we are considering IKM configurations Area and timing of combinatorial multipliers in 0.25µm CMOS128, 64, and 32 bit partial combinatorial multiplier. The 3, 9, and 27 clock cycles are required for thesedesign respectively.

For the hardware-IKM which is described the selection and accumulation blocks of the multiplier becomes large with higher segmentation. This is due to a complicated data path that indeed minimizes the total number of executed XOR operations, but leads to an irregular data path structure. We solved the issue by implementing a data path that does not minimize the number of operations but has a much more regular structure and therefore requires less silicon area.The seven different positions are

possible in the accumulation of the four segments IKM. The positions can be denoted by a seven bit command word which is generated by a small XOR operations that have to be performed. The result for the new selection and accumulation method are listed in Table 2 and are compared to the original method.Tailored ECC core multiplier with purpose to apply the multiplier in a particular ECC design. We made a further modification of the accumulation logic: we integrated the reduction inside the multiplier. The reduction must be performed to transform the long product to an equivalent m bit element inside the field GF (2^m) the reduction must be performed which corresponds to modulo operation in prime fields. Usually, the reduction is done after the multiplication is finished, i.e. after the nine partial multiplication steps were performed. Instead, after every iteration step the reduction is performed. Thus in Fig.5, the partial results c4, c5, c6, and c7, is shown which do not need to be stored. In case of a 256 bit multiplier it saves 255 flip-flops. For the B-233 curve with four-segment multiplier, which requires nine clock cycles for the polynomial multiplication in GF(2233) the silicon area is 0.62mm2 measured for the 0.25 μm CMOS technology.

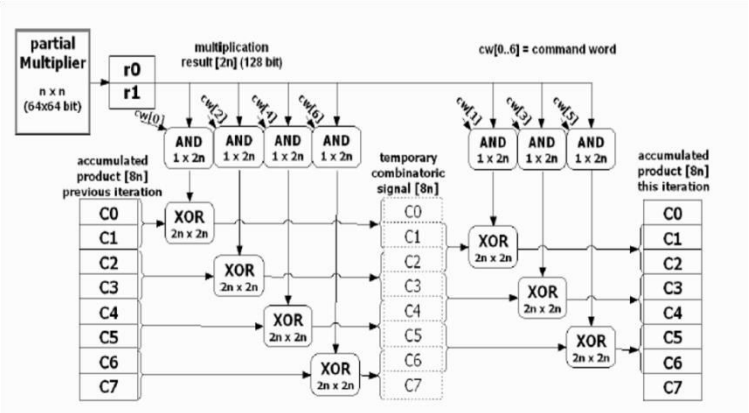

Fig.5. Configurable structure for the accumulation of partial products in the IKM process with embedded reduction operation this design allows only storing four registers (c0 to c3) in flip-flops instead of eight.

The results clearly show that the impact of the number of segments on area consumption is no longer that significant.

TABLE1. AREA AND TIMING OF COMBINATIONAL MULTIPLIERS IN 0.25 μm CMOS

m	CPM			hCKM			hRAIK		
	[mm^2]	[ns]	[nWs]	[mm^2]	[ns]	[nWs]	[mm^2]	[ns]	[nWs]
64	0.477	3.3	3.36	0.176	6.2	0.84	0.170	6.0	0.78
128	2.070	3.9	18.64	0.555	7.9	3.52	0.537	7.6	3.31
256	9.000	4.6	105.1	1.714	9.8	14.01	1.636	9.1	12.51

TABLE2.AREA CONSUMPTIIN IN mm^2 OF SELECTION AND ACCUMULATION TASKS FOR 233 BIT IKM COMPARED TO THE ORIGINAL METHOD.

	Selection	Accumulation	Summation sel. +acc	Orignal Method
2 Segment	0.05	0.08	0.13	0.15
4 Segment	0.05	0.09	0.14	0.39
8 Segment	0.06	0.10	0.16	

TABLE 3: GF(2^m) MULTIPLIERS TAILORED FOR B-163,B-233 AND B-271.

Size[bit]	Segments	Size core mul	Cycles	Area[m^{2}]	Power [mW]	Energy [nWs]
163	2	96	3	0.79	47.9	4.31
163	4	48	9	0.45	31.6	8.53
163	8	24	27	0.39	18.5	14.99
233	2	128	3	1.17	64.5	5.80
233	4	64	9	0.62	42.9	11.58
233	8	32	27	0.44	22.8	18.47
571	2	320	3	4.35	277.6	25.0
571	4	160	9	2.10	141.8	38.5
571	8	80	27	1.31	82.9	67.18

IV. HARDWARE AND TIME COMPLEXITY

The proposed structure shown in Fig.6needs [(*m*/2) +2] PEs and one AC. Each of the regular PEs contains 2(*m*+1) XOR gates in a pair of XOR cells and 2(*m*+1) AND gates in a pair of AND cells. Also, the AC requires (*m*+1) XOR gates. Moreover, (2.5m^2+6.5*m*+4) bit-registers are essential for transferring data to the nearby PE. The latency of the design is [(*m*/2) +3] cycles, where the length of the clockperiod is T_X. The structure of Fig.7 requires nearly the same gate-counts as that of Fig. 6. But its latency is [(*m*/4) +4] cycles. The proposed design outperforms the existing designs can be seen. Even though slightly more registers than that in [11] are used, the proposed design requires smaller latency and lesser critical-path than the other as well as the MUX gates. The digit-serial structures of and yield one product word in *m/l* and (*2m/l*-1) clock-periods respectively, one product word in every clock-period which is produced in the proposed structure. The proposed design can be extended further to obtain a more efficient design for high-speed implementation, especially when *m* is a large number as shown in the Fig.7.

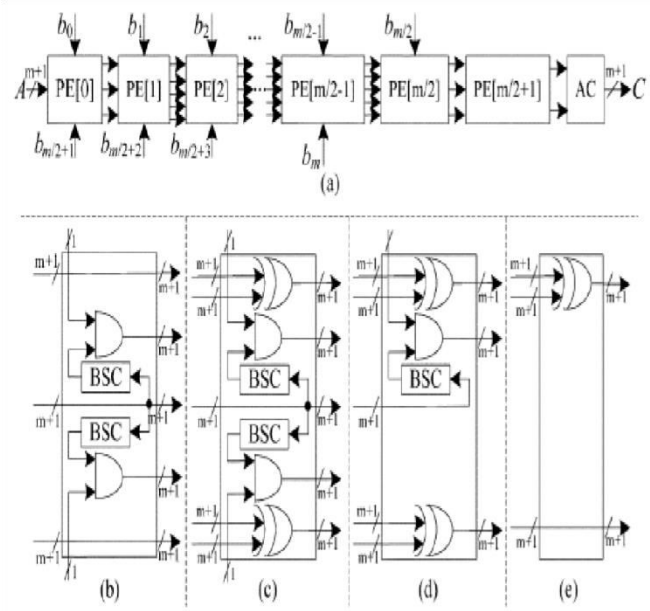

Fig.6. Low-latency register-sharing systolic structure (a) the systolic structure (b) Structure of PE [1] (c) Structure of a regular PE (from PE [2] to PE [*m*/2-1]). (d) Structure of PE [*m*/2]. (e) Structure of PE [*m*/2+1].

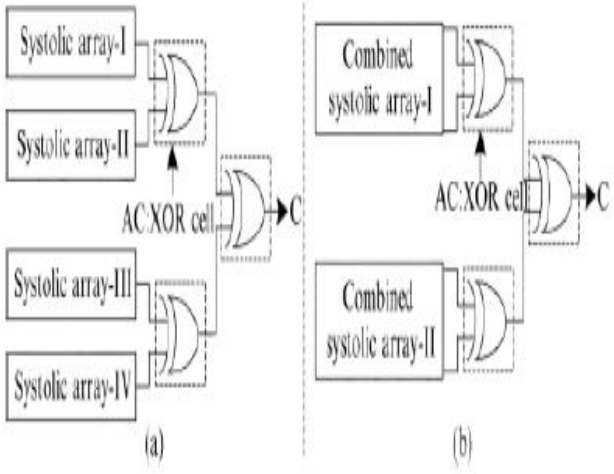

Fig.7. Improved low-latency systolic structure (a) The proposed systolic array merging (b) Improved systolic structure.

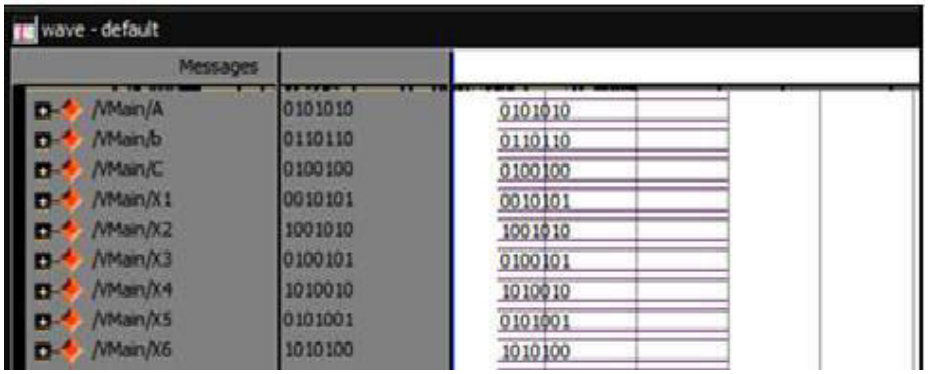

Fig.8Simulation result for m=6

TABLE 4:
Comparison of area and time complexity for m = 6

Design	LUTs	Delay (ns)	Critical path duration (ns)
Exsisting bit parallel design	42	10.083	7.055
Proposed Fig.4 (systolic)	35	9.841	6.503
Proposed Fig.4 (low latency)	35	8.815	6.503
Proposed Fig.5 (register sharing)	21	7.963	6.503

The comparison table II gives the comparison between area and time complexity of the existing and proposed designs for m value of 6.Table III gives the comparison between the area and time complexities of the existing and proposed designs for the value of m equal to 20.

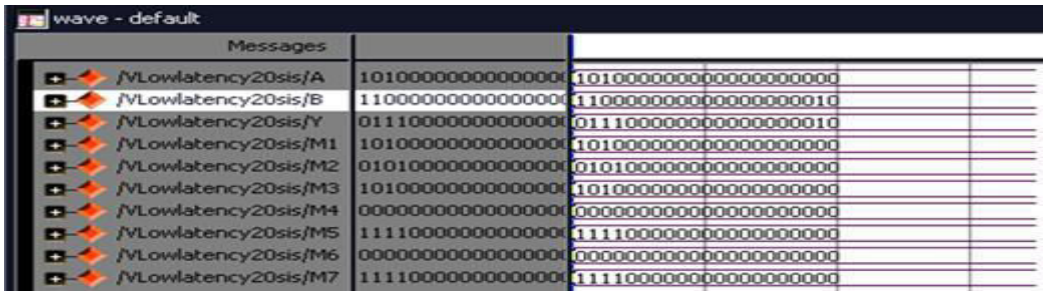

Fig.9Simulation result for m=20

TABLE 5 Comparison of area and timecomplexity for m=20

Design	LUTs	Delay (ns)	Critical path duration (ns)
Exsisting bit parallel design	420	17.428	6.389
Proposed Fig.4 (low latency)	399 (2,835)	14.344	5.753
Proposed Fig.5 (register sharing)	399 (2,772)	14.344	5.753
Proposed Fig.6 (improved low latency register sharing low latency)	378	10.399	5.753
Proposed Fig.7 (further improved register sharing)	312	10.116	5.753

V. CONCLUSION

An irreducible all-one polynomial (AOP) based on area-time-efficient systolic structure for multiplication over *GF* (2^m) is proposed. We have been able to minimize the critical path to one XOR gate delay by novel cut-set retiming and we have derived a low-latency bitparallel systolic multiplierby sharing of registers for the input operands in the PEs. The proposed one is found to involve less area, shorter critical-path and lower latency by comparing with the existing systolic structures for bit-parallel realization of multiplication over GF (2^m).We find that the proposed design involves significantly less ADP and PDP than the existing designs from ASIC and FPGA synthesis. Additionally, our proposeddesign can be extended to further minimize the latency.

REFERENCES

1. Jiafeng Xie, Pramod Kumar Meher, and Jianjun He, **"**Low-Complexity Multiplier for *GF* (2^m) Based on All-One Polynomials", IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Volume. 21, Number. 1, January 2013.
2. M. Ciet, J. J. Quisquater, and F. Sica, "A secure family of composite finite fields suitable for fast implementation of elliptic curve cryptography," in Proc. International Conference Cryptol. India, 2001, pp. 108–116.
3. H. Fan and M. A. Hasan, "Relationship between *GF* (2^m) Montgomery and shifted polynomial basis multiplication algorithms," IEEE Transaction on Computers, volume. 55, number. 9, pp. 1202–1206, September. 2006.
4. C.-L.Wang and J-L. Lin, "Systolic array implementation of multipliers for finite fields *GF* (2^m)," IEEE Transactions on Circuits Systems, vol. 38, no. 7, pp. 796–800, Jul. 1991. [5] B. Sunar and C. K. Koc, "Mastrovito multiplier for all trinomials," IEEE Transaction. Comput., volume 48, number. 5, pp. 522– 527, May 1999.
5. C. H. Kim, C.-P. Hong, and S. Kwon, "A digit-serial multiplier for finite field *GF* (2^m)," IEEE Transactions. Very Large Scale Integration. (VLSI) Systems., volume 13, number 4, pp. 476–483, 2005.
6. C. Paar, "Low complexity parallel multipliers for Galois fields *GF* ($(2^m)^4$) based on special types of primitive polynomials," in Proc. IEEE International Symp. Information. Theory, 1994, p. 98.
7. H. Wu, "Bit-parallel polynomial basis multiplier for new classes of finite fields," IEEE Transactions on Computers, volume. 57, number 8, pp. 1023–1031, August. 2008.
8. S. Fenn, M.G. Parker, M. Benaissa, and D. Taylor, "Bitserial multiplication in *GF* (2^m) using all-one
9. complexity bit-parallel multiplier for *GF* (2^m) defined by allone polynomials using redundant representation," IEEE Transactions on Computers, volume. 54, number. 12, pp. 1628–1629, December 2005.
10. H.-S. Kim and S.-W. Lee, "LFSR multipliers over *GF* (2^m) defined by all-one polynomial,Integr" VLSI J., volume. 40, number. 4, pp. 571–578, 2007.

11 .P. K. Meher, Y. Ha, and C.-Y. Lee, "An optimized design of serial-parallel finite field multiplier for $GF(2^m)$ based on all-one polynomials," in Proc. ASP-DAC, 2009, pp. 210–215.