Volume 5, Issue 5 OCT 2016

DESIGN AND IMPLEMENTATION OF REDUNDANT BASIS HIGH SPEED FINITE FIELD MULTIPLIERS

Vakkalakula Bharathsreenivasulu[1] G.Divya Praneetha[2]
[1]PG Scholar, Dept of VLSI & ES, G.Pullareddy Eng College,kurnool
[2]Assistant professor, Dept of VLSI & ES, G.Pullareddy Eng College, kurnool

Abstract: Redundant Based Multiplier Over Galois Field(GF(2^m)) have gained huge popularity in elliptic curve cryptography (ECC) mainly because of their negligible hardware cost for squaring and modular reduction. In this paper, we have proposed a novel recursive decomposition algorithm for RB multiplication to obtain high-throughput digit-serial implementation. Through efficient projection of signal- flow graph (SFG) of the proposed algorithm, a highly regular processor-space flow-graph (PSFG) is derived. By identifying suitable cut-sets, we have modified the PSFG suitably and performed efficient feed-forward cut-set retiming to derive three novel multipliers which not only involve significantly less time-complexity than the existing ones but also require less area and less power consumption compared with the others.

Both theoretical analysis and synthesis results con firm the efficiency of proposed multipliers over the existing ones. The synthesis results for field programmable gate array (FPGA) and application specific integrated circuit (ASIC) realization of the proposed designs and competing existing designs are compared. It is shown that the proposed high-throughput structures are the best among the corresponding designs, for FPGA and ASIC implementation. It is shown that the proposed designs can achieve up to 94% and 60% savings of area-delay-power product (ADPP) on FPGA and ASIC implementation over the best of the existing designs, respectively.

I .INTRODUCTION

Finite field is a basic operation frequently ITE FIELD multiplication over Galois Field (GF(2^m)) (encountered in modern cryptographic systems such as the elliptic curve cryptography (ECC) and error control coding . Moreover, multiplication over a finite field can be used further to perform other field operations, e.g., division, exponentiation, and inversion. Multiplication over can be implemented on a general purpose machine, but it is expensive to use a general purpose machine to implement cryptographic systems in cost-sensitive consumer products. Besides, a low-end coprocessor cannot meet the real-time requirement of different applications since word length of these processors is too small compared with the order of typical finite fields used in cryptographic systems. Most of the real-time applications, therefore, need hardware implementation of finite field arithmetic operations for the benefits like low-cost and high-throughput rate.

In this paper, we aim at presenting efficient digit-level serial/parallel designs for high-throughput finite field multiplication over based on RB. We have proposed an efficient recursive decomposition scheme for digit-level RB multiplication, and based on that we have derived parallel algorithms for high throughput digit-serial multiplication. We have mapped the algorithm to three different high-speed architectures by mapping the parallel algorithm to a regular 2-dimensional signal- flow graph (SFG) array, followed by suitable projection of SFG to 1-dimensional processor-space flow graph (PSFG), and the choice of feed-forward cut-set to enhance the throughput rate. Our proposed digit-serial multipliers involve significantly less area-time-power complexities than the corresponding existing designs. Field programmable gate array (FPGA) has evolved as a mainstream dedicated computing platform.

II Derivation of Proposed High throughput Structures For RB Multipliers

A.Proposed Structure I

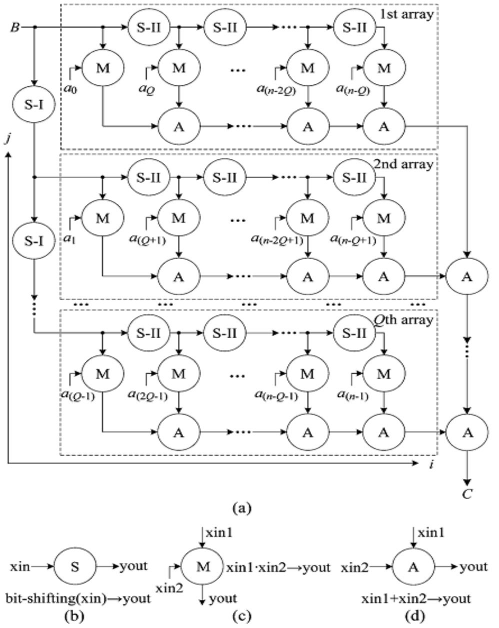


Fig. 1. Signal- flow graph (SFG) for parallel realization of RB multiplication. (a) The proposed SFG. (b) Functional description of S node, where S-I node performs circular bit-shifting of one position and S-II node performs circular bit-shifting by positions. (c) Functional description of M node. (d) Functional description of A node.

The RB multiplication can be represented by the 2-dimensional SFG (shown in Fig. 1) consisting of parallel arrays, where each array consists of bit-shifting nodes (S node), multiplication nodes (M nodes) and addition nodes (A nodes). There are two types of S nodes (S-I node and S-II node). Function of S nodes is depicted in Fig. 1(b), where S-I node performs circular bit-shifting by one position and S-II node performs circular bit-shifting by positions for the degree reduction requirement. Functions of M nodes and A nodes are depicted in Fig. 1(c) and 1(d), respectively. Each of the M nodes performs an AND operation of a bit of serial-input operand with bit-shifted form of operand , while each of the A nodes performs an XOR operation. The final addition of the output of arrays of Fig. 1 can be performed by bit-by-bit XOR of the operands in number of A nodes as depicted in Fig. 1. The desired product word is obtained after the addition of parallel output of the arrays

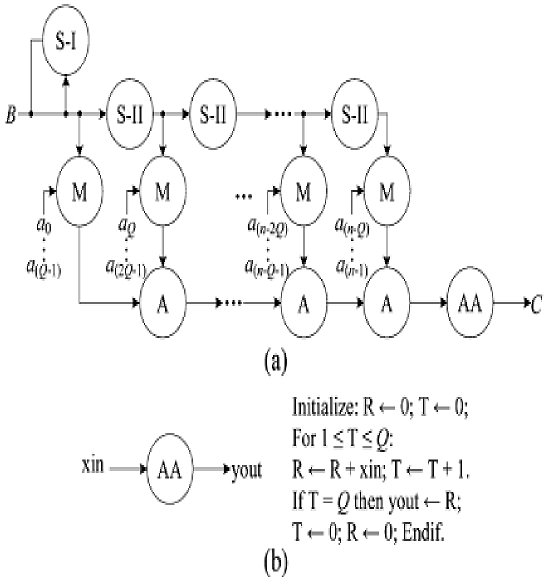


Fig. 2. Processor-space flow graph (PSFG) of digit-serial realization of finite field RB multiplication over . (a) The proposed PSFG. (b) Functional description of add-accumulation (AA) node.

For digit-serial realization of RB multiplier, the SFG of Fig. 1 can be projected along -direction to obtain a PSFG as shown in Fig. 2, where input bits are loaded in parallel to multiplication nodes during each cycle period. The functions of nodes of PSFG are the same as those of corresponding nodes in the SFG of Fig. 1 except an extra add-accumulation (AA) node.

The function of the AA node is, as described in Fig. 2(b), to execute the accumulation operation for cycles to yield the desired result thereafter.

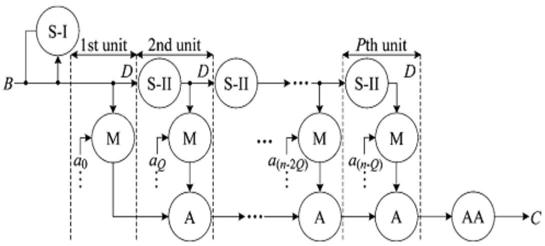


Fig. 3. Cut-set retiming of PSFG of finite field RB multiplication over (GF(2^m)) Where "D " denotes delay

For efficient realization of a digit-serial RB multiplier, we can perform feed-forward cut-set retiming in a regular interval in the PSFG as shown in Fig. 3. As a result of cut-set retiming of the Fig. 3, the minimum duration of each clock period is reduced to(T_A+T_B) , where and denote the delay of an AND gate and an XOR gate, respectively

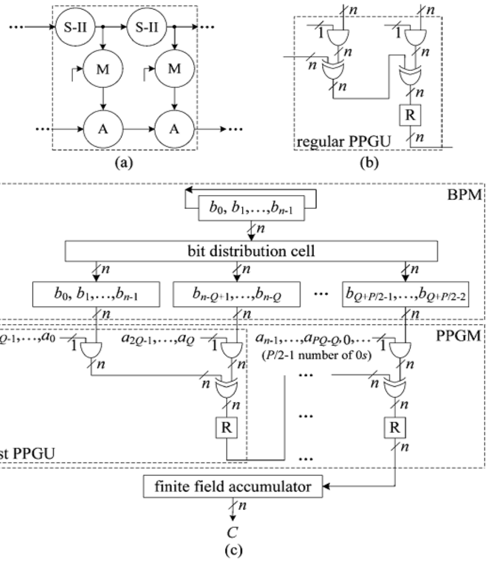


Fig. 5. PS-I for RB multiplier when . (a) Proposed cut-set retiming of PSFG when . (b) Detailed internal structure of merged regular PPGU. (c) Corresponding PS-I for the case .

For example, to obtain the proposed structure for d=2 , a pair of S nodes, a pair of M nodes and a pair of A nodes of the PSFG of Fig. 3 can be merged to form a macro-node as shown within the dashed-lines in Fig. 5. Each of these macronodes can be implemented by a new PPGU to obtain a PPGM of p/2 PPGUs as shown in Fig. 5(b), which consists of two AND cells and two XOR cells (the first PPGU requires only one XOR cell). The functions of AND cell, XOR cell and register cell The critical path of the structure of Fig. 5(c) amounts to . The first output of desired product is available from this structure after a latency of cycles, while the successive outputs are available thereafter in each cycles of duration . The technique used to derive the structure for may be extended for any value of , to obtain a structure consisting of PPGUs.

Proposed Structure-II

we can further transform the PSFG of Fig. 3 to reduce the latency and hardware complexity of PS-I. To obtain the proposed structure, serially-connected A nodes of the PSFG of Fig. 3 are merged into a pipeline form of A nodes as shown within the dashed-box in Fig. 6(a). These pipelined A nodes can be implemented by a pipelined XOR tree, as shown in Fig. 6(b). Since all the AND cells can be processed in parallel, there is no need of using extra "0"s on the input path to meet the timing requirement in systolic pipeline. The critical path and throughput of PS-II are the same as those of PS-I. Similarly, PS-II can be easily extended to larger values of to have low register-complexity structures.

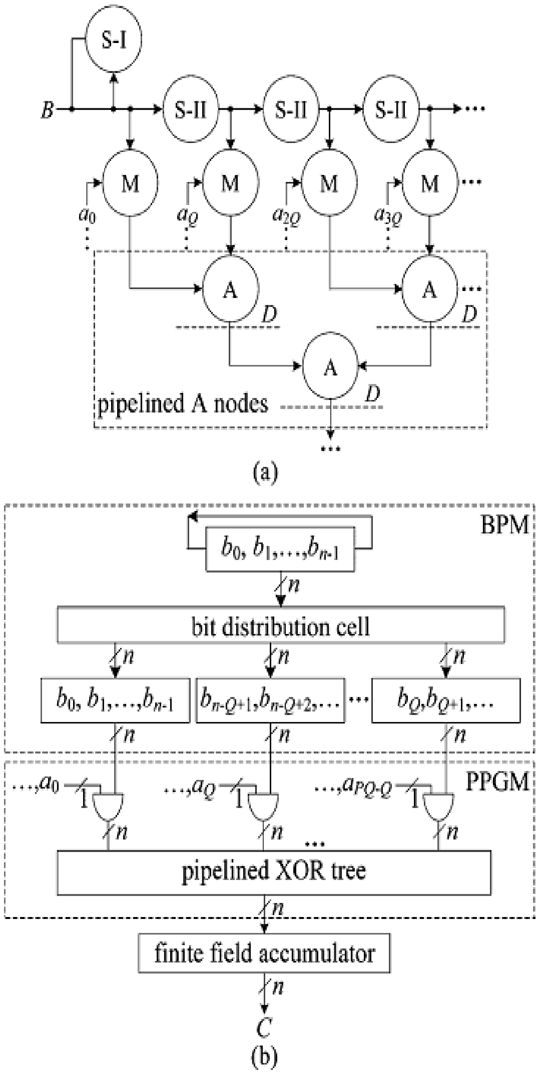


Fig. 6. Proposed structure-II (PS-II) for RB multiplier, where “R” denotes a register cell. (a) Modified PSFG. (b) Structure of RB multiplier

Proposed Structure-III

Since the S nodes of Fig. 3 perform only the bit-shifting operations they do not involve any time consumption. Therefore, we can introduce a novel cut-set retiming to reduce the critical path further, as shown in Fig. 7(a). It can be observed that the cut-set retiming allows to perform the bit-addition and bit-multiplication concurrently, so that the critical-path is reduced to max{T_A+T_X}=T_X,i.e.,the throughput of the design Is increased.

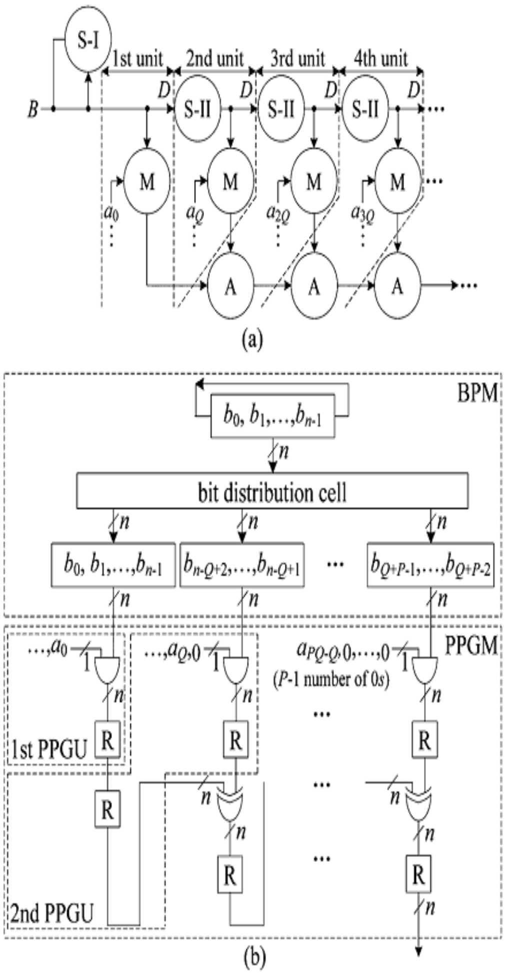
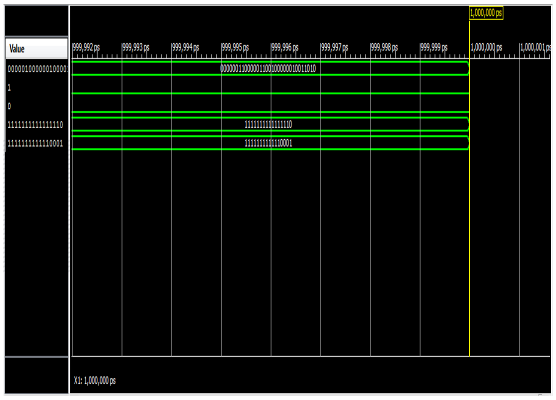


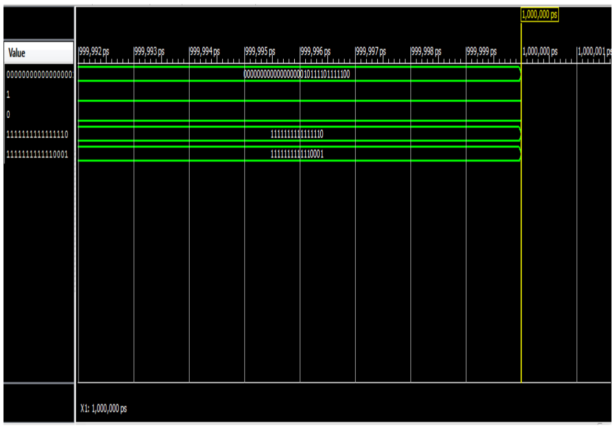
Fig. 7. Novel cut-set retiming of PSFG and its corresponding structure: PS-III. (a) Cut-set retiming. (b) BPM and PPGM of PS-III.

III.Simulation Results:

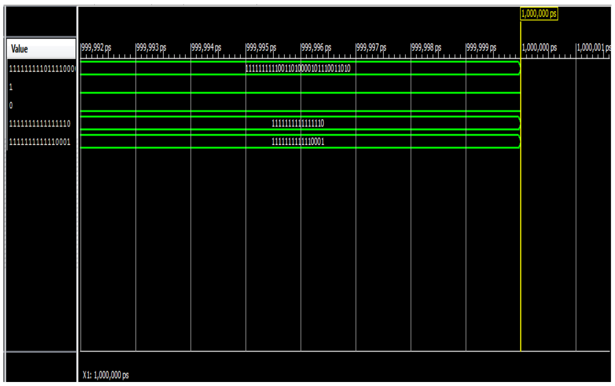
Arc1



Arc2



Arc3



IV. CONCLUSION

We have proposed a novel recursive decomposition algorithm for RB multiplication to derive high-throughput digit-serial multipliers. By suitable projection of SFG of proposed algorithm and identifying suitable cut-sets for feed-forward cut-set retiming, three novel high-throughput digit-serial RB multipliers are derived to achieve significantly less area-time-power complexities than the existing ones. Moreover, efficient structures with low register-count have been derived for area-constrained implementation; and particularly for implementation in FPGA platform where registers are not abundant. The results of synthesis show that proposed structures can achieve saving of up to 94% and 60%, respectively, of ADPP for FPGA and ASIC implementation, respectively, over the best of the existing designs. The proposed structures have different area-time-power trade-off behavior. Therefore, one out of the three proposed structures can be chosen depending on the requirement of the application environments.

REFERENCES

[1] I. Blake, G. Seroussi, and N. P. Smart, Elliptic Curves in Cryptography, ser. 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 brahmaqupta-bha skara equation,” IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 53, no. 7, pp. 1565–1571, 2006.

[3] L. Song and K. K. Parhi, “Low-energy digit-serial/parallel finite field multipliers,” J. VLSI Digit. Process., vol. 19, pp. 149–C166, 1998.

[4] P. K. Meher, “On efficient implementation of accumulation in finite field over and its applications,” IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 1 7, no. 4, pp. 541–550, 2009.

[5] L. Song, K. K. Parhi, I. Kuroda, and T. Nishitani, “Hardware/software codesign of finite field datapath for low-energy Reed-Solomn codecs,” IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 8, no. 2, pp. 160–172, Apr. 2000

[6] G. Drolet, “A new representation of elements of finite fields yielding small complexity arithmetic circuits,” *IEEE Trans. Comput.*, vol. 47, no. 9, pp. 938–946, 1998.

[7] C.-Y. Lee, J.-S. Horng, I.-C. Jou, and E.-H. Lu, “Low-complexity bit-parallel systolic montgomery multipliers for special classes of

,” *IEEE Trans. Comput.*, vol. 54, no. 9, pp. 1061–1070, Sep. 2005.

[8] P. K. Meher, “Systolic and super-systolic multipliers for finite field based on irreducible trinomials,” *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 55, no. 4, pp. 1031–1040, May 2008.