Print ISSN: 2249 – 3492, Online ISSN: 2249 – 3506

International Journal of Research in Management, Science and Technology

Issue - 13, Vol-07, pp. 52-56, Jan-Jun 2017
CLEAR International Journal of Research in Management, Science and Technology

RESEARCH ARTICLE

HIGH PERFORMANCE VLSI ARCHITECTURE FOR MONTGOMERY MODULAR MULTIPLIER

M.S.Shobana Priya[1], Y.V.Ramana Rao[2]
[1]Final year M.E (Applied Electronics), Dept of ECE, College of Engineering, Guindy
[2]Professor, Dept of ECE, College of Engineering, Guindy

Article History:

Received 22nd May 2017

Received in revised form 27th May 2017

Accepted 15.06.2017

Published on 30.06.2017

Keywords: MMM- SCS- CMOS- CSA

Corresponding Author:

M.S. Shobana Priya

ABSTRACT

The paper proposes implementation of Montgomery Modular Multiplier (MMM) using hybrid full adders instead of Conventional CMOS Full adders in the carry save adder(CSA) adder. The hybrid full adder is designed using a conventional CMOS and transmission gate logic. There is about 54% and 55% reduction of area (no. of components) in Radix 2 MMM and Semi-Carry-Save (SCS) based MMM with hybrid full adders respectively compared to Conventional CMOS designs. There is significant reduction in the power dissipation of 52% for Radix 2 MMM and 46% for SCS based MMM when hybrid adders are used instead of Conventional CMOS Full Adders. The implementations are carried out using Mentor Graphics Pyxis Schematic in 180-nm technology.

Introduction

Traditionally, cryptography is used for military and diplomatic services to provide secure communication, in which two communicating parties share a secret key in secured ways. All cryptographic operations are made in finite fields, which map to modular multiplication in the digital world. The Montgomery multiplication, is used to build cryptography applications. This multiplication is usedto perform fast modular multiplication.

Young Sae Kim et al. (2000) implemented 1024-bit modular processor for RSA Cryptosystem [2]. They proposed an implementation method to optimize a 1024-bit RSA processor in which hardware resources are reduced. This architecture is compatible for effective input output interface. In these cryptosystem design modified Montgomery algorithm is used which makes the computation simple which requires one additional multiplication. The proposed architecture provides a better solution to the practical single chip implementation for large bit size RSA processor for cryptosystem design.

Viktor Bunimov et al. (2002) implemented a modified Montgomery algorithm [3] that makes efficient use of the Carry Save Adder by reusing one level CSA. This meets the predominant requirement of most hardware's and the area and power optimization is being achieved.

M. Zhang et al. (2003) proposed an architecture [4] with hybrid pass logic with static CMOS output. The CMOS stage here provides full swing and balanced output eventually adder cells can be cascaded without buffer.

Print ISSN: 2249 – 3492, Online ISSN: 2249 – 3506

International Journal of Research in Management, Science and Technology

C. Mclvor et al. (2004) proposed an algorithm [5] that makes use of Carry Save Adders and the output obtained in each stage are given is Carry Save formats so that the process is much easier to be solved. Two algorithmic variants one based on a five-to-two CSA and the other on a four-to-two CSA plus multiplexer are presented. The practical application of the approach has been demonstrated to design special purpose RSA processing units with 512-bit and 1024-bit key sizes. The resulting RSA units exhibit the highest data rates reported in the literature, reflecting the achievement of the very low and word length independent critical path delay.

Yuan –Yang Zhang et al. (2007) proposed an efficient CSA architecture for Montgomery modular multiplication[6]. This proposes a method to reuse the CSA architecture to perform the result format conversion, which leads to small area and fast speed as CSA involved in Montgomery modular multiplication require a full addition to convert the carry save representation of the result into a conventional form. Experimental results show better performance on the critical delay of this architecture and decrease in clock cycles. With the usage of only 2 CSA's, the area reduction achieved is very significant.

S-R Kuang et al. (2013) developed an architecture [7] to speed up the process of encryption and decryption by employing carry save addition to avoid carry propagation process. This algorithm reduces the energy consumption and increases the throughput by superfluous carry save addition and register write operation. Gated clock is applied by bringing change to the BRFA structure and this reciprocates optimizing the energy consumption of the overall system.

P. Bhattacharyya et al. (2015) proposed a hybrid 1-bit full adder [8] which overcomes the disadvantage in area, power and delay with the introduction of more than one logic style in the design. Optimizing of XNOR module also leads to the reduction in the overall power dissipation of the system.

Shiann-RongKuangetal.(2016) proposed fast modular multiplication [9] with the help of CSA. The levels of CSA used is also reduced here and the modification is done in the CSA as Configurable CSA and due to this the highly efficient Montgomery Modular Multiplier is obtained.

Montgomery Multiplication Algorithm has the advantage of replacing division operations by bit shift operations. If the least significant bits to be shifted out are not zero, Montgomery's algorithm adds multiples of modulus to clear these bits before shifting them out. In regular modular multiplication, after all bits of the multiplicand are processed, modulus is repeatedly subtracted from the result unless the result is less than the modulus. In Montgomery multiplication, bits are shifted out as each bit of the multiplicand is processed, leaving no need for the subtractions.Thus, reducing the overall execution time when there are many multiplications to be done with the same modulus and with the same number of multipliers is achieved using Montgomery Modular Multiplier.

In this paper, conventional CMOS full adder in radix 2 MMM and SCS based MMM [9] are replaced using Hybrid Full adders. The performance of these modified implementations are compared with those containing Conventional CMOS Full adders about 50% reduction in area and power dissipation is achieved in the proposed modification

I. Montgomery Modular Multiplication

Modular multiplication with large integers is a time-consuming operation and considered difficult. Therefore, many algorithms were derived in-order to make this process easier. One of these techniques is Montgomery Modular Multiplication (MMM). The existing systems that are being considered for using hybrid full adders. The radix 2 Montgomery Modular and Semi-Carry-Save Based Montgomery modular multiplications. The architectures and algorithms of Radix 2 Based MMM and SCS based MMM are as follows.

II. Radix 2 based Montgomery Modular Multiplier [9]

A, B, N are considered as inputs, S(k) is considered as output**,** k is the number of input bits which determines the number of iterations and i is the iteration value. Initially for 0^{th} iteration k=0, S(k) is considered as 0. The input Ai and B gets multiplied. Once when this multiplication is over the initial carry and sum value are added with the multiplied value. Here q_i is the last bit of sum obtained from carry save adder 1. The sum and carry obtained from carry and sum is denoted as SS(Sum) and SC(Carry). The value of qi when it is said to be 1, gets multiplied with N and added with the already multiplied value of Ai and B. this process takes place until the number of iterations is k-1 and the result S[k] is obtained. The block diagram is shown in figure 1.

Algorithm MM:
Radix-2 based Montgomery Modular Multiplier

Inputs: A,B,N (modulus)

Print ISSN: 2249 – 3492, Online ISSN: 2249 – 3506

International Journal of Research in Management, Science and Technology

output: S[k]
1. S[0] = 0;
2. for i = 0 to k – 1 {
3. qi = (S[i] + Ai * B) mod 2;
4. S[i+1] = (S[i]+ * B + qi * N)/2;
5. }
6. if (S[k] > N) S[k} = S[k] – N;
7. return S[k];

III. SCS based Montgomery Modular Multiplier [9]

The SCS based MMM gives results in Semi Carry Save format. SS and SC are the corresponding outputs that are been obtained. The main advantage of SCS based MMM over radix 2 MMM is that the usage of subtractor is avoided. Instead of subtractor the number of iterations are increased by 2 that is from k to k+2 so that the final comparing and subtraction can be fully avoided. Due to this the area optimization, can be done and the power dissipation is also reduced with reduction in the overall delay. The algorithm of SCS based MMM is given as follows and the block diagram is given in figure 1.

Algorithm MM:
SCS based Montgomery Modular Multiplier

Inputs: A,B,N (modulus)
output: S[k+2]

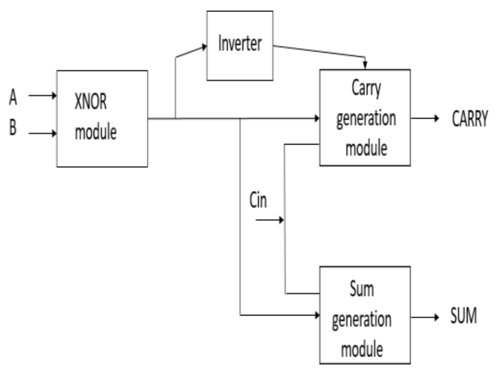


Figure 3. Hybrid Full Adder.

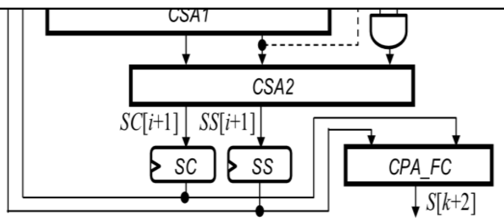


Figure 2. SCS based Montgomery Modular Multiplier.

IV. Hybrid Full Adder [8]

The hybrid full adderconsists of modified XNOR Sum and Carry generation modules. In hybrid logic conventional CMOS(C-CMOS) and transmission gates are coupled together to achieve low power and high-speed operation when compared to other conventional full adder design logic styles.

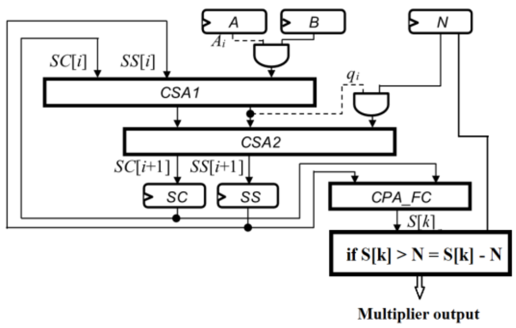


Figure 1. Radix-2 based Montgomery Modular Multiplier.

The XNOR module is designed using C-CMOS and the carry generation module is designed using transmission gates. Figure 3 shows the block diagram of Hybrid Full adder.

Figure 4 shows the transistor level diagram of Hybrid Full adder.

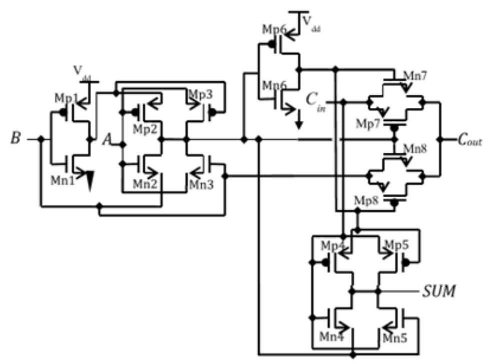


Figure 4. Hybrid Full Adder (Transistor level).

In the place of conventional full adders in CSA block, these Hybrid Full-Adders are used in Radix 2 and SCS based MMMs.

Print ISSN: 2249 – 3492, Online ISSN: 2249 – 3506

International Journal of Research in Management, Science and Technology

V. Results and Discussion

The implementation of radix 2 and SCS based MMMs using C-CMOS and Hybrid Full adders in CSA block is carried out in Mentor Graphics Pyxis Schematic tool. The Area and Power calculations are done in Mentor graphics Pyxis Schematic Tool in 180nm Technology. Figures 5 and 6 shows the schematics of Radix 2 and SCS based MMMs.

The performance comparison of Radix 2 based MMM with SCS based MMM when executed with Conventional full adder with Hybrid full adder has optimizes Area, Power Dissipation and Delay. The bit size occupied are 4-bit. Table 1 shows the area and power consumption of Radix 2 and SCS based MMMs with C-CMOS and Hybrid full adders. Where in 4-bit CSA is considered

Table 1. Area and power of Radix 2 and SCS based MMMs with C-CMOS and Hybrid Full adders in 4-bit CSA

Factors	Radix 2 MMM		SCS based MMM	
	C-CMOS FA	Hybrid FA	C-CMOS FA	Hybrid FA
Area	2682	1458	1866	1040
Power Dissipated	66.461 n watts	35.166 n watts	45.609 n watts	21.064 n watts

VI. Graphical Representation

Figures 7 and 8 show the graphical representation of the result obtained in table 1

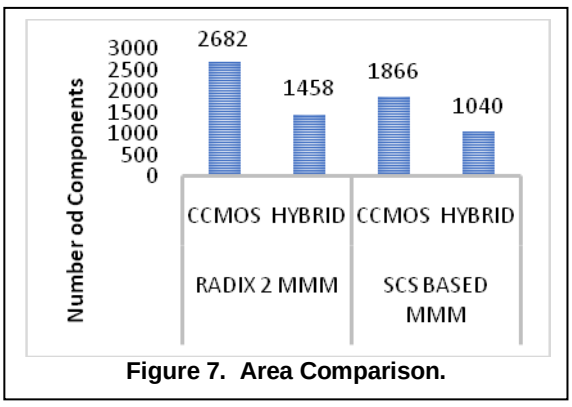


Figure 7. Area Comparison.

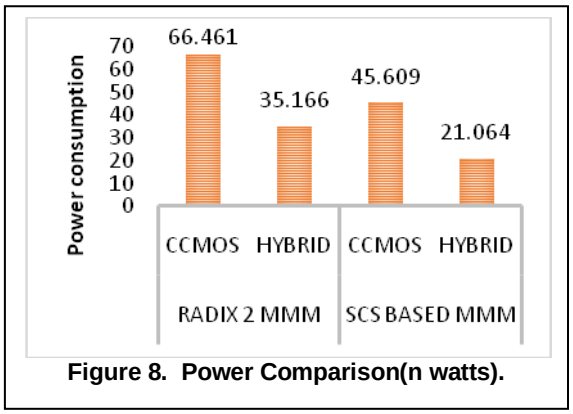
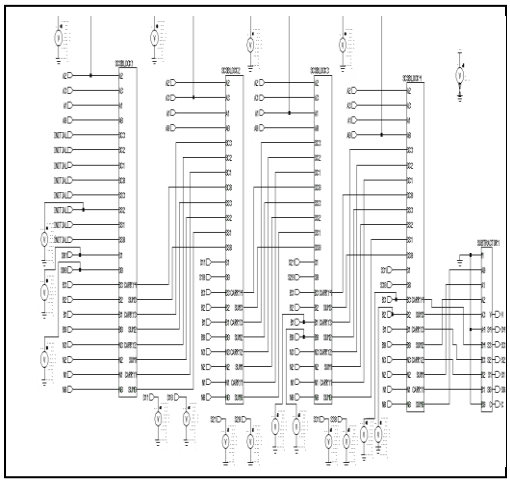


Figure 8. Power Comparison(n watts).



VII. Conclusion

Thus, a Montgomery Modular Multiplier architecture is designed with Hybrid Full Adders. This reduces the overall power dissipation and area. The efficiency of the multiplier is found to improve by replacing normal C-CMOS adder with Hybrid Full Adder. The significance of using Hybrid Full Adder in place of C-CMOS Full Adder is the reduction of overall area of about 54% and 55% for Radix 2 MMM and SCS based MMM respectively. The optimization obtained in Power dissipation is 52% and 46% for Radix 2 MMM and SCS based MMM when Hybrid Full Adder is used in place of C-CMOS Full Adders.

VIII. Acknowledgements

The authors would like to thank College of Engineering, Guindy, Chennai, India, for providing the simulation facility.

IX. References

[1]C. D. Walter, "Montgomery exponentiation needs no final subtractions," Electron. Lett., vol. 35, no. 21, pp. 1831–1832, Oct. 1999.

[2]Y. S. Kim, W. S. Kang, and J. R. Choi, "Asynchronous implementation of 1024-bit modular processor for RSA cryptosystem," in Proc. 2nd IEEE Asia-Pacific Conf. ASIC, Aug. 2000, pp. 187–190.

[3]V. Bunimov, M. Schimmler, and B. Tolg, "A complexity-effective version of Montgomery's algorihm," in Proc. Workshop Complex. Effective Designs, May 2002.

[4]M. Zhang, J. Gu, and C.-H. Chang, "A novel hybrid pass logic with static CMOS output drive full-adder cell," in Proc. Int. Symp. Circuits Syst., May 2003, pp. 317–320.

[5]C. McIvor, M. McLoone, and J. V. McCanny, "Modified Montgomery modular multiplication and RSA exponentiation techniques," IEE Proc.Comput. Digit. Techn., vol. 151, no. 6, pp. 402–408, Nov. 2004.

[6]Y.-Y. Zhang, Z. Li, L. Yang, and S.-W. Zhang, "An efficient CSA architecture for Montgomery modular multiplication," Microprocessors Microsyst., vol. 31, no. 7, pp. 456–459, Nov. 2007.

[7]S.-R. Kuang, J.-P. Wang, K.-C. Chang, and H.-W. Hsu, "Energy-efficient high-throughput Montgomery modular multipliers for RSA cryptosystems," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 21, no. 11, pp. 1999–2009, Nov. 2013.

[8]P. Bhattacharyya, B. Kundu, S. Ghosh, V. Kumar and A. Dandapat, "Performance Analysis of a Low-Power High-Speed Hybrid 1-bit Full-Adder Circuit," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 23, no. 10, pp. 2001–2008, Oct. 2015.

[9]Shiann-RongKuang, Member, IEEE, Kun-Yi Wu, and Ren-Yao Lu, "Low-Cost High-Performance VLSI Architecture for Montgomery Modular Multiplication," in IEEE Transactions On Very Large Scale Integration (VLSI) Systems, Vol. 24, No. 2, Feb 2016.