IOSR Journal of VLSI and Signal Processing (IOSR-JVSP)
Volume 6, Issue 5, Ver. I (Sep. - Oct. 2016), PP 48-52
e-ISSN: 2319 – 4200, p-ISSN No. : 2319 – 4197
www.iosrjournals.org

High-Performance VLSI Architecture for SCS Based Montgomery Modular Multiplication

B.Vaisalini[1], M.Pradeep[2]
[1]*PG Scholar, Dept of ECE, SVECW, Bhimavaram, AP, India*
[2]*Associative Professor, Dept of ECE, SVECW, Bhimavaram, AP, India*

Abstract: *In present generation Cryptography plays a crucial role in security purpose. Security comes mostly with three parameters Confidentiality, Integrity and authentication. All these terms are important for a data to be secured. For hardware implementation of this process Montgomery Modular Multiplication is used, for encryption process in public key cryptography. This paper is discussing about the Semi Carry Save based Montgomery Modular Multiplication (SCS-MM2), with high speed performance. In this Paper, we propose a modified SCS based Montgomery modular multiplication (SCS-MM2) with a Reversible Carry Save Adder (RCSA) using peres gates, so that the performance can be increased, and its simulation and synthesis results are presented. Previously, the radix-2 Montgomery modular Multiplication (MM) architecture was implemented for Basic MM, Full Carry Save Montgomery Modular multiplication (FCS-MM) and the Basic SCS-MM1. The proposed Radix-2 modified SCS-MM2 describes high performance architecture and its results are shown for 128bit length. The resultant architecture is simulated using Modelsim, design verification and synthesis results are done using Xilinx ISE. The proposed architecture is compared with the existing SCS-MM2 it can achieve high performance.*

I. Introduction

Cryptography is a method of storing and transmitting data in a particular form, so that those whom it is intended only can read and process the data. Cryptography is very essential for security purpose in data transmission. For its hardware implementation Montgomery modular Multiplication algorithms is used. Mostly in Public Key Cryptography this logic is used in data encryption process. Montgomery algorithm can be classified into two types based on its operation. They are Full-Carry-Save Montgomery modular Multiplication (FCS-MM) and Semi-Carry-Save Montgomery modular multiplication (SCS-MM1) forms. In FCS-MM both the obtained carry and sum are considered as outputs. In SCS-MM only the sum which was obtained is considered as output. When compared to FCS-MM, SCS-MM is having a low area because of less number of adder levels in the basic algorithm.

In this paper we discuss about the Modified SCS-MM2 architecture and analyse it for 128-bit inputs. In SCS-MM algorithm it has three input A, B, N, and S as sum output .A is a Multiplicand, B is multiplier and N is modulus. There are some rules for considering the inputs. They are length of the inputs should be same. Modulus value should be always greater than the multiplicand and multiplier.

II. SCS-MM2 Algorithm

The modified SCS-MM2 algorithm is shown in fig.1. Initially we make the carry and sum values as the sum of multiplier and the modulus this is pre-computation step. The steps from 3 to 4 iterates for K times. Here K represents the number of bits and i represents i^{th} bit. In fig.1 suffix 0 represents the least significant bit.

```
Algorithm MM;
Modified SCS-MM2 algorithm
Input : A, B, N ( modulus )
Output : S[ k ]
1. ( SS [0], SC [ 0 ] ) = (B + N + 0)
2. For ( i = 0 to k - 1 )
3. {   q [ i ] = ( SS [ i ] ) + A [ i ] * B ) mod2;
       If ( A [ i ] = 0  and  q [ i ]0 =  0 )  X = 0;
       If ( A [ i ] = 0  and  q [ i ]0 =  1 )  X = N;
       If ( A [ i ] = 1  and  q [ i ]0 =  0 )  X = B;
       If ( A [ i ] = 1  and  q [ i ]0 =  1 )  X = B + N;
4.     SS[ i+1 ] , SC[ i + 1]= ( SC [ i ] + SS [ i ] + X ) / 2;
   }
5. If ( SS[ k ] >= N)   then
6.              S[ k ] =SS[ k ] – N;
7. else        return S[ k ];
```

Fig1. Modified SCS-MM2 algorithm

Adders are of many types. Out of those carry save adder is efficient because it is having less propagation delay. Carry Save adder for n-bit means it is having n-parallel adders, which produce n-bit sums and n-bit carry's. The inputs for carry save adder are SS,SC and mux output. Mux output depends up on "aa" and "qa" of a single bit. Here we considered "aa" as A[i] * B and gate Least Significant Bit. "qa" represents the sum of SS and "aa".

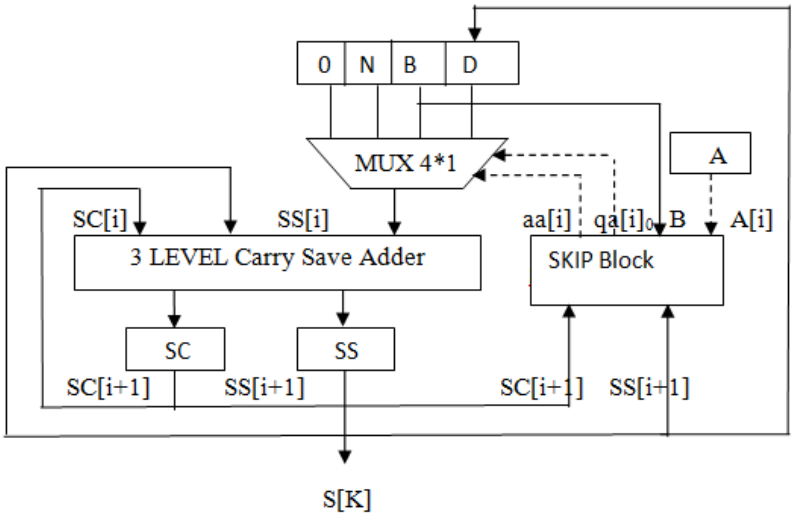


Fig.1(a) Block diagram of SCS-MM2 algorithm.

In fig2 depending upon "aa" and "qa" values the third input for the CSA varies. This loop iterates for n times. The final stage sum is considered as the final output. The CSA block internally consists of full adders.

III. Proposed System

The main advantage of proposed system is to increase the speed of algorithm. It achieved by implementing a full adder using two peres gates. Fig.2(a) represents a full adder logic by using two peres gates. Peres gates are called reversible gates. The first peres gate resembles a half adder and the second also the same. These gates produce three garbage outputs, which we don't require. Here we neglected it. The performance of RCSA is higher than CSA.

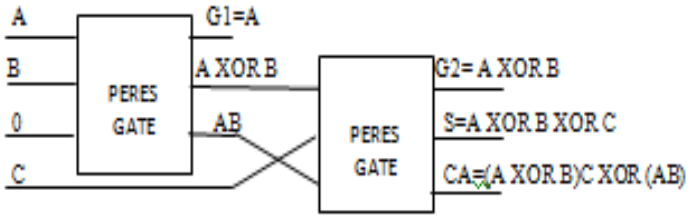


Fig.2(a) Block diagram of Reversible Full adder.

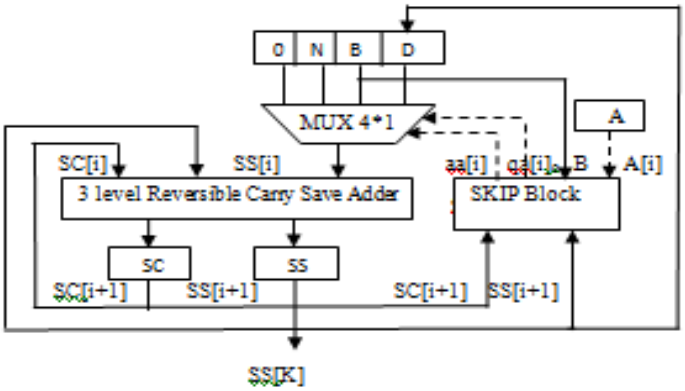


Fig.2(b) Block diagram of Modified SCS-MM2.

The remaining modified SCS-MM2 algorithm process is same as existing system. The internal operation of existing and proposed system is same. The only difference is adder block. RCSA of n-bit consists

of n full adders which are made of peres gates. so that its performance increases. So that overall performance of SCS-MM2 algorithm was also increased.

IV. Results and Comparison

The design of SCS-MM2 and Modified SCS-MM2 has been made by using Verilog Hardware Description Language (Verilog HDL). The simulation results has been evaluated by using Modelsim 6.3c and synthesis Performances are estimated by using Xilinx 10.1 for 16-bit.

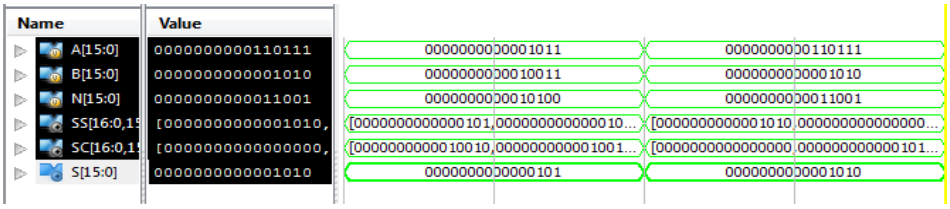


Fig. 3(a). Simulation Waveform of SCS-MM2 algorithm

In Fig.3(a). A, B, N are inputs and S is final output all the inputs and outputs are of same size. SS, SC are internal registers. All the inputs and outputs are of 16-bit. Finally obtained sum value is from the multiplication of A and B and modulus for the resultant.

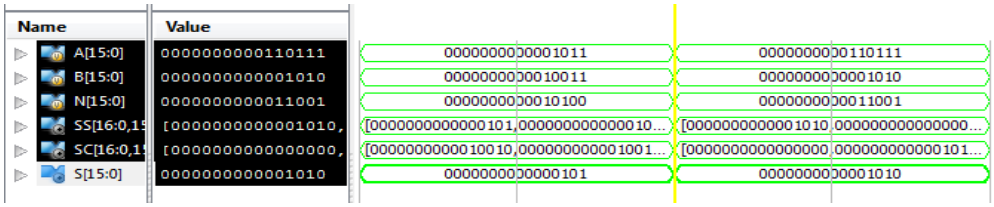


Fig. 3(b). Simulation Waveform of Modified SCS-MM2 algorithm.

In Fig.3(b). A, B, N are inputs and S is final output all the inputs and outputs are of same size. SS, SC are internal registers. Finally obtained sum value is from the multiplication of A and B and modulus for the resultant. which is same as the SCS-MM2 value.bit size is of 16bit.

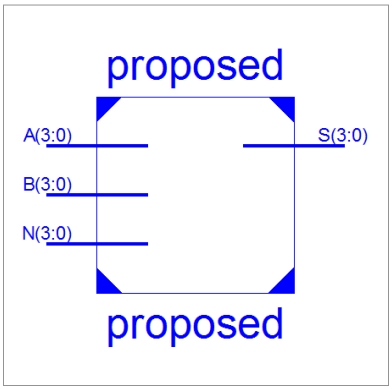


Fig. 4(a) RTL Schematic of modified SCS-MM2 of 4-bit.

In Fig.4 RTL Schematic shows input and output signals. A (multiplicand), B (multiplier), N (modulus), S (final sum) of 4 bit range.

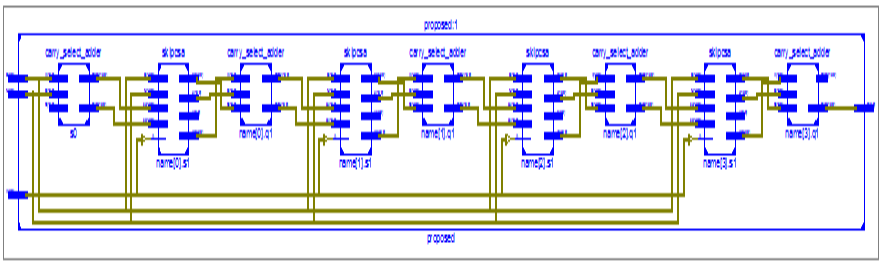
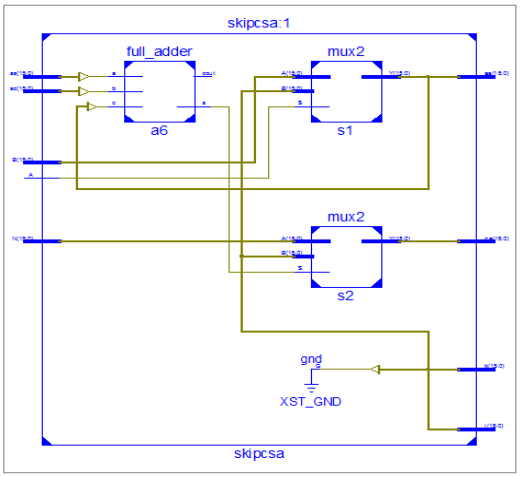


Fig. 4(a): RTL Schematic detailed view of SCS-MM2

In Fig.4(a) RTL Schematic shows RCSA and skip logic. Bit by bit operation is done. This is represented in the schematics. It consists of 5 compressed CSA's and 4 Skip blocks.



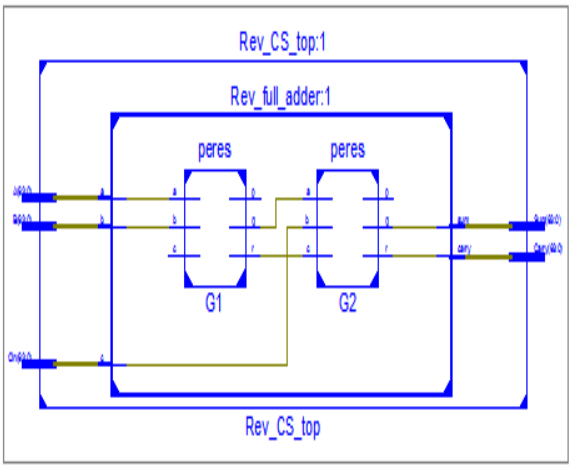


Fig. 4(b): RTL Schematic of Skip block **Fig. 4(c):** RTL Schematic of RCSA

In Fig. 4(b) RTL Schematic of Skip block, consists of a single bit adder and two mux's. Two mux's are used to calculate "aa" and "qa" values. In Fig. 4(c) RTL Schematic consists of two peres gates which implement a full adder. RCSA consists of two n-full adders. So that speed of operation increased.

Table1. Parametric analysis of SCS-MM2 and modified SCS-MM2 for critical path

No of bits	SCS-MM2	Modified SCS-MM2
4	13.630ns	11.539ns
8	23.381ns	16.864ns
16	40.505ns	28.182ns
32	82.471ns	51.183ns
64	165.309ns	99.238ns
128	344.134ns	209.491ns

Table.1 represents the timing report for SCS-MM2 and Modified SCS-MM2 architectures, using virtex-2 FPGA using Xilinx10.1 tool. So that the performance of proposed system increases.

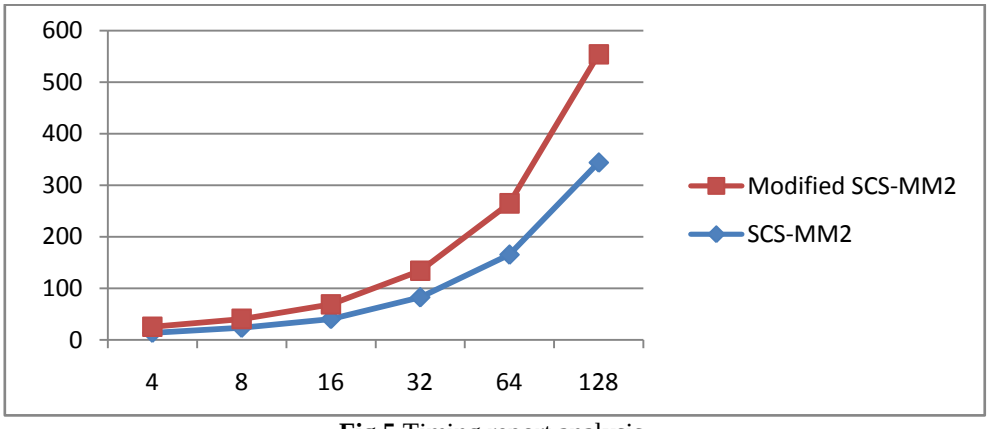


Fig.5 Timing report analysis.

Fig.5 represents a graphical representation of modified SCS-MM2 and the proposed system. The critical path of existing system is more than the proposed system as number of bits increased

V. Conclusion

The proposed SCS-MM2 (Semi Carry Save Montgomery modular multiplication) for radix-2 architecture reduces number of critical path delay when compared to the existing logic. The SCS-MM2 (Semi Carry Save Montgomery modular multiplication) architecture is simulated using Modelsim and design verification, area timing report is done using Xilinx ISE 10.1. Finally, the proposed architecture can achieve reduced critical path, and increases the speed of operation.

References

[1]. KUANG et.al."Low Cost High Performance VLSI Architecture for Montgomery MM", IEEE Trans. Very Large Scale Inegr. (VLSI) Syst., vol. 24, issue: 2,pp434-445 , Feb.2016.

[2]. R. L. Rivest, A. Shamir and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems", Commun. ACM, vol. 21, no. 2, pp. 120-126, Feb. 1978.

[3]. Chandra.K and Kumar.P, "Optimization of RSA Processors Using Multiplier", International Journal of Computer Trends and Technology (IJCTT) - Vol-4, Issue-5-May 2013.

[4]. S.Ashwini, P.Thirapaiah "A High-Speed Montgomery Modular Multiplication Algorithm To Reduce The Energy Consumption Based On RSA Cryptosystem", Proc. International Journal of Engineering and Computer Science (IJECS)- Vol-4, Issue-10 Oct 2015,pg-14653-14658.

[5]. Alan Daly and William Marnane "Fast Montgomery Modular Multiplication and RSA Cryptographic Processor Architecture".

[6]. Santharubavagini.K , Abirami.C and Jegadeeshwari,"MBRFA Circuits for High-Throughput and Low Latency Montgomery Modular Multipliers ", International journal of Communication and Computer Technologies (IJCCTS), vol-02,No-09,Issue-02 March 2014

[7]. Jhing-Fa Wang, Po-Chuan Lin and Ping-Kun Chiu "A Staged Carry-Save-Adder Array for Montgomery Modular Multiplication"

[8]. Dr.Deepa Jose , Nathimugil.J and Abida Begum ,"Implementation of Optimized Montgomery modular Multiplier on FPGA", International Journal of Advanced Research in Biology Engineering Science and Technology (IJARBEST) Vol. 2, Issue 4, April 2016.

[9]. J. Han, S. Wang, W. Huang, Z. Yu, and X. Zeng, "Parallelization of radix-2 Montgomery multiplication on multicore platform," IEEE Trans.Very Large Scale Integr. (VLSI) Syst., vol. 21, no. 12, pp. 2325–2330,Dec. 2013.

[10]. David Narh Amanor, Christof Paar, Jan Pelzl and Viktor Bunimov, Manfred Schimmler ,"Efficient Hardware Architectures For Modular Multiplication On Fpgas".

[11]. Harmeet Kaur1, Mrs.Charu Madhu, "Montgomery Multiplication Methods - A Review" International Journal Of Application Or Innovation In Engineering & Management (Ijaiem) Volume 2, Issue 2, February 2013.