International Journal of Research
Available at https://edupediapublications.org/journals
e-ISSN: 2348-6848
p-ISSN: 2348-795X
Volume 05 Issue 7
March 2018

A Memory-Based FFT Architecture Design Using Montgomery Multiplication

[1]DURGA PRASAD RAJULAPATI, [2]MEDIBOYINA SUPARNA
[1]Assistant professor, Dept of ECE, Ramachandra College of engineering,Eluru,A.P,India.
[2]M.Tech-Scholar, Dept of ECE, Ramachandra College of engineering, Eluru, A.P,India

ABSTRACT: Basically, in number-theoretic cryptographic algorithm, the modular multiplication operation consumes more time. So there are fast multiplier architectures to minimize the delay and increase the throughput using parallelism and pipelining. However such designs are large in terms of area and gives less efficiency. An improved FFT-based Montgomery modular multiplication (MMM) algorithm is proposed. The FFT algorithm is used to provide fast convolution computation. We also introduce a general method for efficientparameter selection for the proposed algorithm. Architectures with single and double butterfly structures are designed obtaining low area-latency solutions, which we implemented on Xilinx Virtex-6 FPGAs. From results it can observe that it produces better results compared to existed system.

KEY WORDS: Montgomery modular multiplication, Fast Fourier transform (FFT), field-programmable gate array (FPGA).

I.INTRODUCTION

The subject of the paper is hardware implementations of the RSA algorithm with larger than 1,024-bit modulus length. In particular, our objective is to create implementations that achieve high area-time efficiency, rather than creating very low area or ultra high speed implementations at the high cost of the other. The RSA algorithm, being the very first public-key encryption and digital signature algorithm since 1978, is ubiquitously deployed and used, from smart cards to cell phones and SSL boxes. Its security depends on the difficulty of factoring a modulus n to find its two prime factors p and q. The very first implementations of the RSA algorithm in Early 1980s assumed 512-bit modulus (and thus, two 256-bit primes) would be sufficient, but within a decade, advances in factorization methods increased the This has been the case for almost 2 decades, but now, as recently as 2010s, the security of 1,024-bitwas questioned. Many implementations now use 2,048-bitmodulus, while the National Institute of Standard and Technology (NIST) recommended 3,072-bit or 4,096-bit modulus size for the near future in order to maintain RSA secure. Needless to say, larger key sizes lead to longer processing time and more hardware resource when computing, due tothe fact the RSA computation requires the modular exponentiation (xm modN), which is computed by repeated modular multiplications. Therefore, the performance of modular multiplication has a direct impact on the efficiency of RSA computation, and therefore, high performance modular multipliers supporting 3,072-bit or higher operand size are required.

Montgomery modular multiplication (MMM) is an efficient method to compute modular multiplication. Due to this fact, integer multiplication has been extensively studied in order to improve MMM. Existing multiplication methods can be classified into two groups. Methods of the first group are performed only in time domain, including the school book method, the Karatsuba method and the Toom-Cook method. Methods of the second group are performed in both time and spectral domains.Since the fast Fourier transform (FFT) based algorithm is applied to the second group, a lower asymptotic complexity can beachieved compared to

International Journal of Research
Available at https://edupediapublications.org/journals
e-ISSN: 2348-6848
p-ISSN: 2348-795X
Volume 05 Issue 7
March 2018

the methods in the first ones. So the proposed system gives better results.

II.EXISTED SYSTEM

The below figure (1) shows the architecture of existed system.Operationsof the FMLM, are computed sequentially, while pipelined architectures are designed inside each unit. The below architecture consists of multiply adder, Fast Fourier Transform, Ripple Carry Adder, Subtracter, Shift Module, RAM Sets. Let us discuss each component in detail manner.

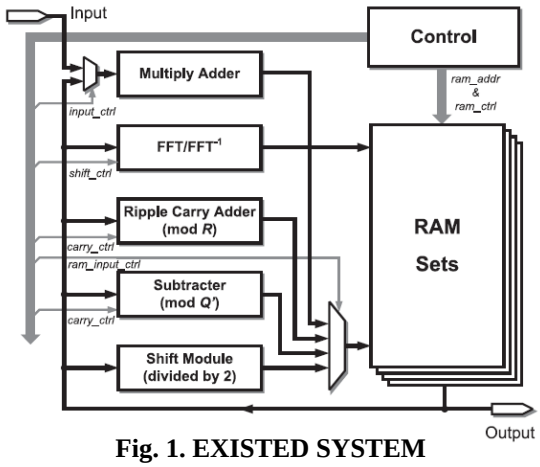

Fig. 1. EXISTED SYSTEM

The first and main important component in the architecture is multiply and adder unit. This unit implements the component wise multiplication and addition of FFT-RAM. To realize the component wise multiplication when operand size is not larger than few hundred bits then karatsuba method is used. The multiplier and adder units works with pipeline of 3 bit inputs and one bit output. At last to enhance the performance of multiplication, karatsuba method is applied recursively. This is about multiply and adder unit and let us discuss about FFT unit.

Forward and reverse networks are formed in FFT unit. Basically, this unit is targeted on high clock frequency and small resource cost. Here constant geometry FFT is applied to FFT computation. The main comparison of in-place and constant geometry FFT is it has same connection network between every adjacent stages. The FFT is designed with six inputs. In this the four inputs forward the digits into BFSs for FFT computation and the other two inputs forward the pre-computed upper bound constraints into FSO.

Next one is RAM unit. RAM unit consists of several RAM sets which stores the pre-computed data, the intermediate results, and the final modular product. In RAM the data storage requirement during FFT-RAM computation is not trivial. Now to well manage the input and output of data and to reduce the wiring workload, RAM unit is built. The remaining are the Ripple Carry Adder (RCA), the Subtracter and the Shift Module units are responsible for the time domain operations, such as modulo R and Q0 reductions, conditional selections. Now to generate all control signals of entire system, control unit is designed. But this system does not gives better results, so a new system is proposed which is discussed in below section.

III.PROPOSED SYSTEM

The below figure (2) shows the architecture of proposed system. It consists of the following main parts: an input and an output index vector generator, a computation address generator, three different memory bank groups, a PE unit applied with the HRSB scheme, some commutators between the memory and the PE, some pre stored twiddle factors, and FFT size parameters, and an exponent scaling unit.The dashed lines represent the Control signals while the real lines denote the flowing data. Let us discuss each component in detail manner.The input index vector generator distributes the input data to different memory banks without

International Journal of Research
Available at https://edupediapublications.org/journals
e-ISSN: 2348-6848
p-ISSN: 2348-795X
Volume 05 Issue 7
March 2018

data conflicts, and the output one reorders the output data to a natural sequence. The computation address generator obtains all the concurrentdata of each cycle and stores back the intermediate results.Memory groups 1 and 2 are in a ping pong mode to hold two continuous data symbols in input sampling, and memory group 3 is used to output the computed data in right order.

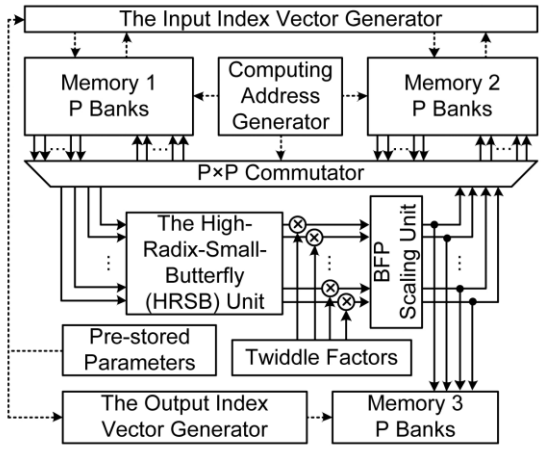

Fig. 2. PROPOSED SYSTEM

Three memory groups only exist in NSPP FFTs. For SPP FFTs, as in-place strategy is available, only two memory groups are enough. The HRSB unit is the kernel processing engine. The commutators located between the memories and HRSB unit provide efficient data routing mechanism which is controlled by the computation address generator. The twiddle factors and the FFT size parameters are all pre stored in register files, which are used to configure the FFT working modes. The exponent scaling unit controls scaling operations for block floating-point, which can increase the signal-to-quantization noise ratio and reduce the memory storage.

The computation can be completed within 128 clock cycles. The input and output index vector generators in the 2*n*-point FFT are merged to one. The only difference is that the binary representation of the index for the input is in a forward manner, while it is in a reversed manner for the output. In this paper, we present a new factorization method that will make the continuous-flow mode completely available and simultaneously ensure the effectiveness of the conflict free address scheme. So it gives efficient results compared to existed system.

IV. RESULTS

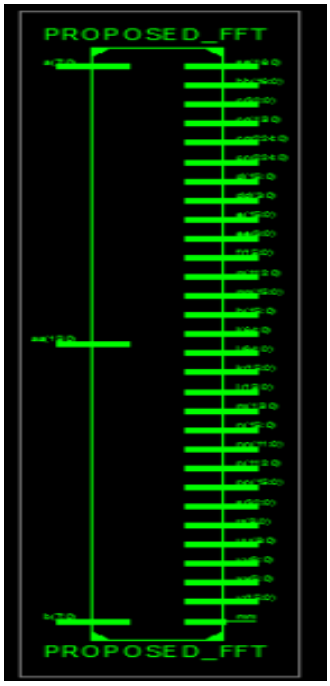

FIG. 3. RTL SCHEMATIC

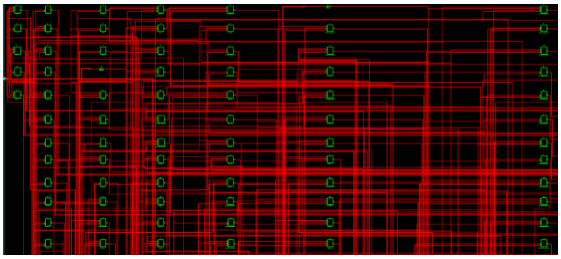

FIG. 4. TECHNOLOGY SCHEMATIC

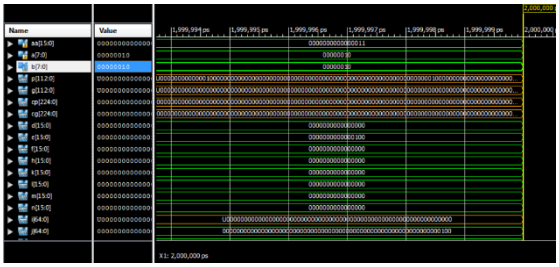

FIG. 5. INPUT WAVEFORM

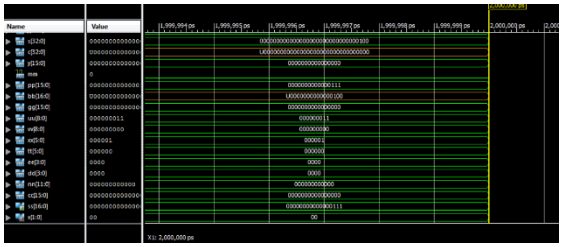

Fig. 6. OUTPUT WAVEFORM

V.CONCLUSION

In this paper we propose a modified version of FFT. FFT based Montgomery modular multiplication algorithm under McLaughlin's framework (FMLM3). The zero padding occurred in existed system is avoided by computing the modular multiplication steps. So in proposed system,the memory bank and address can be generated by modulo and multiplicationoperations of the decomposition digits. Compared to exist one the proposed system gives effective efficiency.

VI.REFERENCES

[1] 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, 1978.

[2] R. L. Rivest, "A description of a single-chip implementation of the RSA cipher," Lambda, vol. 1, no. Oct.–Dec., pp. 14–18, 1980.

[3] "Recommendation for key management," NIST, Tech. Rep. Special Publication 800-57, Part-1, Rev.-3, 2012.

[4] P. L. Montgomery, "Modular multiplication without trial division," Mathematics Comput., vol. 44, no. 170, pp. 519–521, 1985.

[5] A. Karatsuba and Y. Ofman, "Multiplication of multidigit numbers on automata," Soviet Physics Doklady, vol. 7, 1963, Art. no. 595.

[6] S. A. Cook and S. O. Aanderaa, "On the minimum computation time of functions," Trans.Amer.Math. Soc., vol. 142, pp. 291–314, 1969.

[7] A. Sch€onhageand V. Strassen, "Schnelle multiplikation Großer Zahlen," Computing, vol. 7, no. 3/4, pp. 281–292, 1971.

[8] M. F€urer, "Faster integer multiplication," SIAM J. Comput., vol. 39, no. 3, pp. 979–1005, 2009.

[9] D. Harvey, J. van der Hoeven, and G. Lecerf, "Even faster integer multiplication," CoRR, vol. abs/1407.3360, 2014. [Online]. Available: http://arxiv.org/abs/1407.3360.

DURGA PRASAD RAJULAPATI present working as assistant professor in Ramachandra College of engineering,Eluru,A.P,India.

MEDIBOYINA SUPARNA M.Tech scholar in Ramachandra College of engineering, Eluru, A.P, India.