International Journal of advancement in electronics and computer engineering (IJAECE)
Volume 06, Issue 5, June 2017, pp.514-518, ISSN: 2274-1412

Result Analysis of Low power and area efficient Reverse Converter Design via Parallel-Prefix Adders

Mukesh Kumar Pandey, Suresh Gawande
M.Tech Scholar, Dept. of EC, Bhabha Engineering Research Institute, Bhopal, mukeshpandey1990@gmail.com, India;
Professor, Dept. of EC, Bhabha Engineering Research Institute, Bhopal,India;

Abstract *– The implementation of residue number system reverse converters based on well-known regular and modular parallel prefix adders is analyzed. The VLSI implementation results show a significant delay reduction and area × time2 improvements, all this at the cost of higher power consumption, which is the main reason preventing the use of parallel-prefix adders to achieve high-speed reverse converters in nowadays systems. Hence, to solve the high power consumption problem, novel specific hybrid parallel-prefix-based adder components those provide better tradeoff between delay and power consumption. The power, area and delay of the proposed system are analysis using Xilinx 14.2.*

Keywords*: parallel-prefix adder, residue number system (RNS), reverse converter.*

I. Introduction

Power dissipation has become one of the major limiting factors in the design of digital ASICs. Low power dissipation will increase the mobility of the ASIC by reducing the system cost, size and weight. DSP blocks are a major source of power dissipation in modern ASICs. The residue number system (RNS) has, for a long time, been proposed as an alternative to the regular two's complement number system (TCS) in DSP applications to reduce the power dissipation. Some research have shown that implementing FIR filters in residue number system (RNS) instead of two's complement number system (TCS) can give a reduction in power dissipation. FIR filters are among the less complex DSP blocks. A general sketch of how RNS computations can be performed is shown in figure 1.

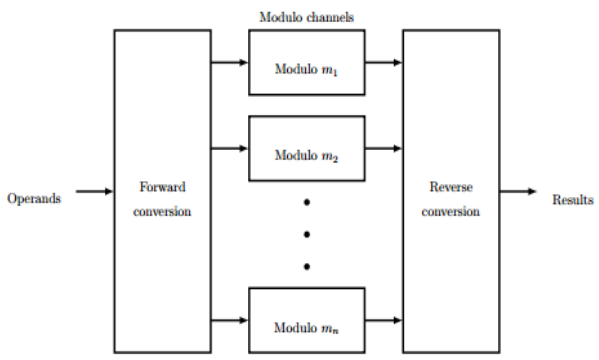

Fig. 1: The basic principle of RNS

The Residue Number System plays a significant role in the battery based and portable devices because of its low power features and its competitive delay. The Residue number system reverse converter is designed with parallel prefix addition by using new components methodology for higher speed operation[1].The RNS consists of two main components forward and the reverse converter that are integrated with the existing digital system. The forward converter performs the operation of converting the binary number to the modulo number whereas the reverse converter performs the operation of reverse converting the modulo number to the binary number which is the hard and time consuming process compared with the forward converter. The fundamental RNS concepts such as 1)RNS definition with properties and their applications,2)consideration of modulo set selection,3)design of forward converter,4)modulo arithmetic units,4)design of reverse converter are discussed[2]. The voltage over scaling (VOS) technique is applied to the residue number system to achieve high energy efficiency. The VOS technique introduces soft errors which degrades the performance of the system. To overcome these soft errors a new technique is implemented called joint RNS-RPR (JRR) which is the combination of RNS and the reduced precision redundancy. This method provides the advantage of satisfying the basic properties of RNS includes shorter critical path, reduced complexity and low power [3].New Architectures are presented for the module sets (2n-1, 2n,

2n+1) for the conversion from the residue to the binary equivalents [4]. Here the speed and the cost are major concern. Distributed arithmetic principles are used to perform the inner product computation in [5].The input data which are in the residue domain which are encoded using the Thermometer code format and the outputs are encoded using the One hot code format. Compared to the conventional method which used Binary code format, the proposed system which achieves higher operating speed. The residue number system which provides carries free addition and fully arithmetic operation [6], for several applications such as digital signal processing and cryptography [7]-[11]. In paper present a comprehensive method which uses the parallel prefix adder in selected position, thereby using the shift operation on one bit left to design a multiplier on the same design module to achieve a fast reverse converter design. The usage on parallel prefix structure in the design leads to higher speed in operation meanwhile it increases the area and power consumption. In order to compensate the tradeoff between the speed, area and power consumption, a novel specific hybrid parallel prefix based adder components are used to design the reverse converter. This hybrid design which provides the significant reduction in the power delay product (PDP) metric and leads to considerable improvements in the area time² product (AT^2) in comparison with the traditional converters without using parallel prefix adders.

II. Theory

II.1. Field-Programmable Gate Array

A field-programmable gate array (FPGA) is a semiconductor device that can be configured by the customer or designer after manufacturing—hence the name "field-programmable". To program an FPGA one must specify how they want the chip to work with a logic circuit diagram or a source code in a hardware description language (HDL). FPGAs can be used to implement any logical function that an application-specific integrated circuit (ASIC) could perform, but the ability to update the functionality after shipping offers advantages for many applications.

FPGAs contain programmable logic components called "logic blocks", and a hierarchy of reconfigurable interconnects that allow the blocks to be "wired together"—somewhat like a one-chip programmable breadboard. Logic blocks can be configured to perform complex combinational functions, or merely simple logic gates like AND and XOR. In most FPGAs, the logic blocks also include memory elements, which may be simple flip-flops or more complete blocks of memory.

For any given semiconductor method, FPGAs are sometimes slower than their fixed ASIC counterparts. They additionally draw a lot of power, and usually achieve less functionality using a given quantity of circuit complexity. However their benefits include a shorter time to market, ability to re-program within the field to fix bugs, and lower non-recurring engineering prices. Vendors may take a middle road by developing their hardware on ordinary FPGAs, however manufacture their final version thus it will no longer be changed once the design has been committed.

Field Programmable Gate Array (FPGA) devices were introduced by Xilinx within the mid-1980s. They differ from CPLDs in design, storage technology, variety of inbuilt options, and cost, and are aimed toward the implementation of high performance, large-size circuits.

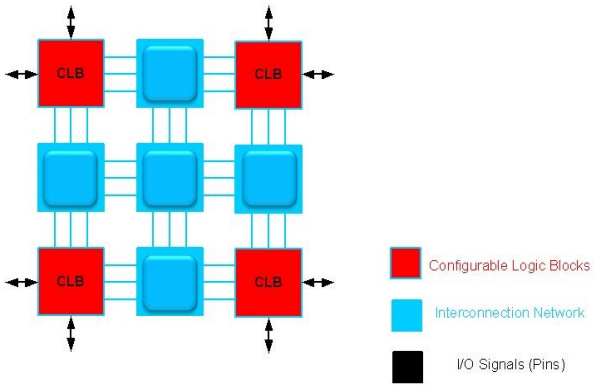

Fig.2 FPGA Architecture

The basic design of an FPGA is illustrated in figure 2. It consists of a matrix of CLBs (Configurable Logic Blocks), interconnected by an array of switch matrices.

The internal design of a CLB is totally different from that of a PLD 1st, rather than implementing SOP expressions with AND gates followed by OR gates (like in SPLDs), its operation is generally based on a LUT (lookup table). Moreover, during an FPGA the number of flip-flops is far additional abundant than in a CPLD, therefore allowing the construction of additional refined sequential circuits. Besides JTAG support and interface to numerous logic levels, different further options are included in FPGA chips, like SRAM memory, clock multiplication (PLL or DLL), PCI interface, etc. Some chips additionally include dedicated blocks, like multipliers, DSPs, and microprocessors.

Another basic difference between an FPGA and a CPLD refers to the storage of the interconnects. Whereas CPLDs are non-volatile (that is, they make use of antifuse, EEPROM, Flash, etc.), most FPGAs use SRAM, and are thus volatile. This approach saves area and lowers the value of the chip because FPGAs present a very large number of programmable interconnections, however needs an external ROM. There are, however, non-volatile FPGAs (with antifuse), which could be advantageous once reprogramming isn't necessary.

FPGAs are very sophisticated. Chips manufactured with state-of-the-art0.09mmCMOS technology, with 9

copper layers and over 1,000 I/O pins, are currently offered.

III. Method

The main reason for the high power consumption and space overhead of those adders is that the recursive impact of generating and propagating signals at every prefix level. However, this technique suffers from high fan-out, which can build it usable just for small width operands. However, we tend to may address this drawback by eliminating the additional prefix level and using a changed excess-one unit instead. In contrast to the BEC, this modified unit is ready to perform a conditional increment supported control signals as shown in Fig. 3, and also the resulted hybrid modular parallel-prefix excess-one (HMPE) adder is pictured in Fig.4. The HMPE consists of 2 parts:

1)A regular prefix adder
2)A modified excess-one unit.

First, 2 operands are added using the prefix adder, and therefore the result's conditionally incremented afterward based on control signals generated by the prefix section therefore on assure the single zero illustration.

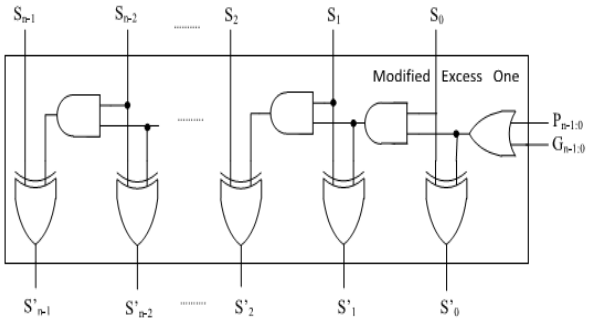

Fig.3 Modified excess-one unit

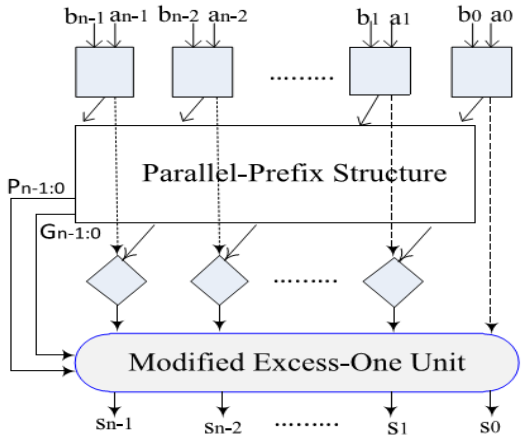

Fig.4 HMPE structure

In this style, the adder style is implemented by using the berkelium adder parallel prefix structure. Here the primary 2 operands are added by using the prefix adder preprocessing stage thereby generating the propagate and generate equation. the primary stage processed signal get passed to subsequent stage known as the prefix carry tree, this stage once more computes, generate and propagate equation by using the previous output and every one the logic cells used within the berkelium adder network. These processed signals are passed to the post processing block.

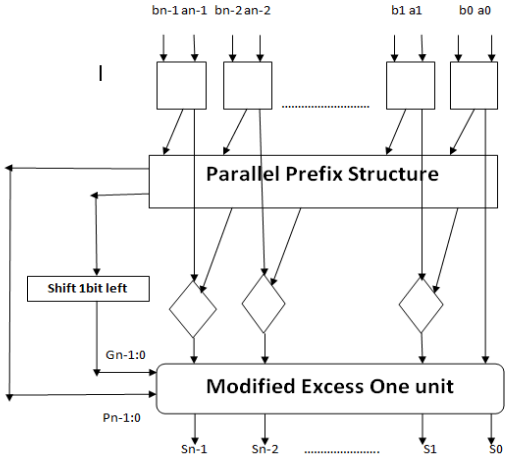

Fig.5 Parallel prefix adder based multiplier design

The generated carry bits in initial 2 stages of the parallel prefix network get passed to the last stage. Again the generated and prorogated signal within the second stage get passed to the last stage referred to as the post process stage, this stage computes the total and also the carryout signal by using the processed generate and propagate equation to style the adder for (4n+1) modulo addition for n=5.In that style the prorogated signal or the generated signal get left shifted to 1-bit position and so the sum get obtained for planning the multiplier factor.

IV. Result

Proposed system simulation results are as follows:

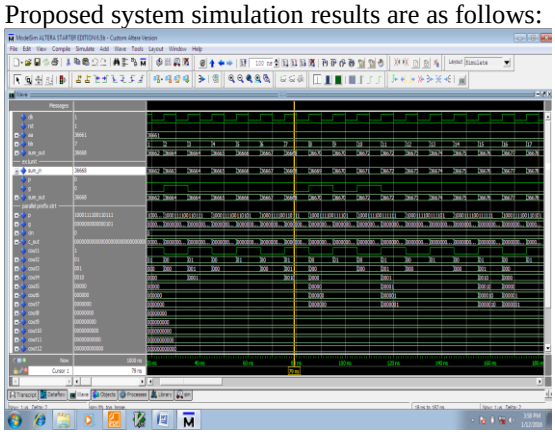

Fig.6: simulation output

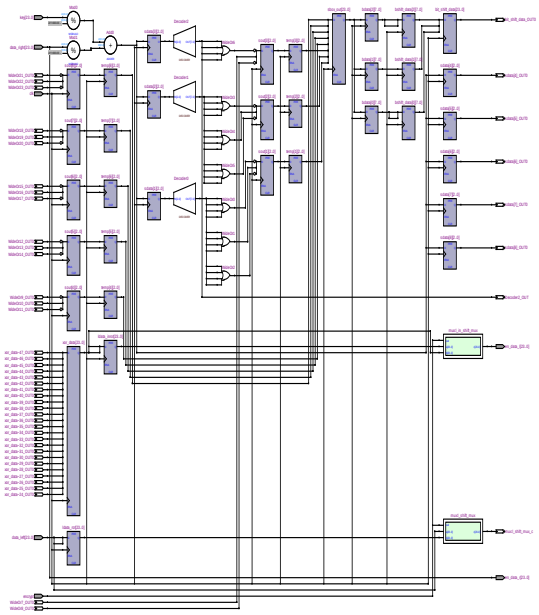

Fig.7 Scalable Encryption algorithm Top Module

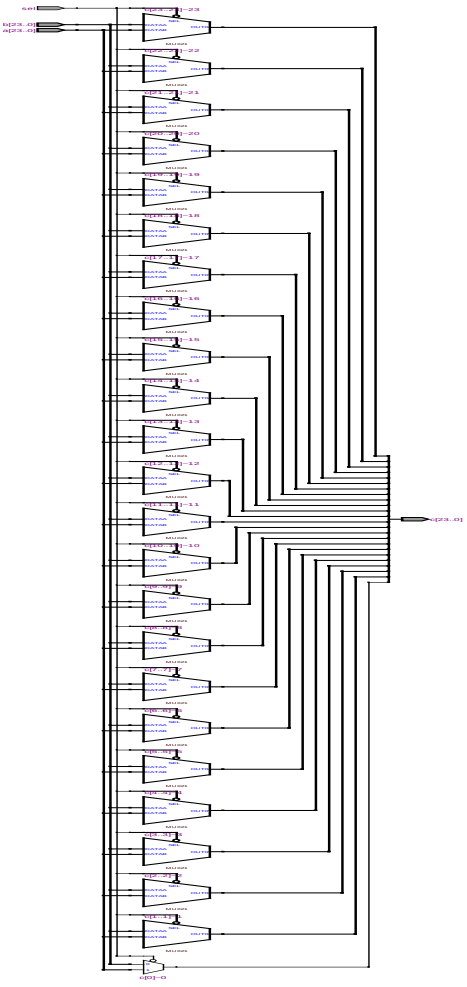

Fig.8 Scalable Encryption Algorithm Multiplexer

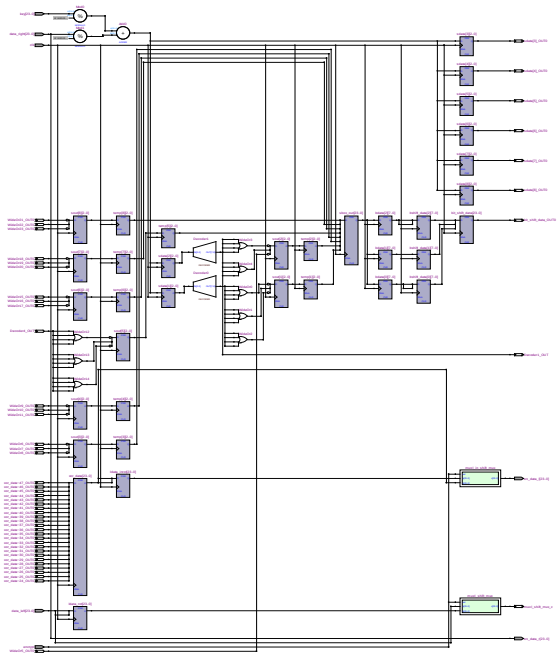

Fig.9 Scalable encryption algorithm Decryption

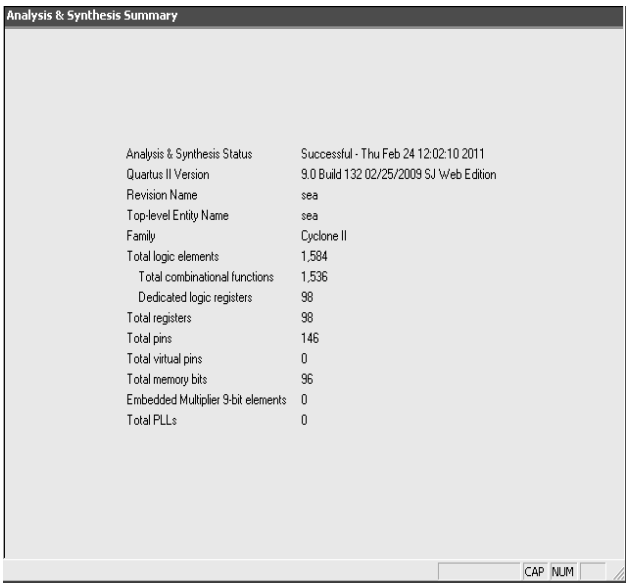

Fig.10 Analysis and Synthesis

V. Conclusion

This paper presents parallel-prefix-based adder elements that provide higher tradeoff in area and delay are therefore exhibited to design reverse converters. a technique is represented to style reverse converters depending on numerous styles of prefix adders. This brief presents a technique which will be applied to most of the present reverse converter architectures to enhance their Performance and adjust the cost/performance to the application specifications. he use of modular and regular parallel-prefix adders projected during this brief in reverse converters highly decrease the delay at the expense of significantly additional power AND circuit

area, whereas the projected prefix-based adder elements permits one to achieve appropriate tradeoffs between speed and price by selecting the right adders for the elements of the circuits which will benefit from them the most.

References

[1] A.A. E. Zarandi, A. S. Molahosseini, M. Hosseinzadeh, S. Sorouri, S. Antão, and Leonel Sousa "Reverse Converter Design via Parallel-Prefix Adders: Novel Components, Methodology, and Implementations" IEEE Tra. on Very Large Slale Int. Sys. Pp. 1-5 2015.

[2] Somayeh Timarchi, Mahmood Fazlali1, and Sorin D.Cotofana, "Unified Addition Structure for Moduli Set {2n-1, 2n,2n+1} Based on a Novel RNS Representation" IEEE pp.247-252 2010.

[3] Ghassem Jaberipur "On Building General Modular Adders from Standard Binary Arithmetic Components" The CSI Journal on Computer Science and Engineering Vol. 4, No. 2&4, Pp. 10-16, 2006.

[4] Saeid Banhanfar and Nadali Zarei "Reverse Converter for the Moduli Set {2n-1, 2n, 2n+1} Base on Grouping Number", IJCSI International Journal of Computer Science Issues, Vol. 10, Issue 6, No 1, November 2013.

[5] Chan Hua Vun, Senior Member, IEEE, Annamalai Benjamin Premkumar, Senior Member, IEEE, and Wei Zhang, Member, IEEE, "A New RNS based DA Approach for Inner Product Computation", IEEE Trans.Circuits And Systems—I: Regular Papers, vol. 60, no. 8, AugusT 2013.

[6] A. Omondi and B. Premkumar, Residue Number Systems: Theory and Implementations. London, U.K.: Imperial College Press, 2007.

[7] Rajalingam, M. Kuttimani, A. Muthumanicckam, and R. Sornalatha. "Design and Implementation of RNS Reverse Converter using Parallel Prefix Adders." International Journal of Computer Applications 117.6 (2015).

[8] Devi, J. Brindha, and G. Rohinipriya. "Design of Reverse Converter Using Parallel Prefix Adders and CRT." International Journal of Engineering and Applied Sciences (IJEAS) ISSN: 2394-3661, Volume-2, Issue-3, March 2015.

[9] Vinod Kumar PS, Mr. Sudhakar Reddy N "AN EFFICIENT REVERSE CONVERTER DESIGN VIA PARALLEL PREFIX ADDER", IJESRT August, 2015.

[10] Yezerla, Sudheer Kumar, and B. Rajendra Naik. "Design and Estimation of delay, power and area for Parallel prefix adders." Engineering and Computational Sciences (RAECS), 2014 Recent Advances in. IEEE, 2014.

[11] B. Parhami, Computer Arithmetic: Algorithms and Hardware Designs,2nd ed., New York, NY, USA: Oxford Univ. Press, 2010.

[12] J.Chen and J. Hu, "Energy-efficient digital signal processing via voltageover scaling-based residue number system," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 21, no. 7, pp. 1322–1332, Jul. 2013.

[13] C. H. Vun, A. B. Premkumar, and W. Zhang, "A new RNS based DA approach for inner product computation," IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 60, no. 8, pp. 2139–2152, Aug. 2013.

[14] S. Antão and L. Sousa, "The CRNS framework and its application to programmable and reconfigurable cryptography," ACM Trans. Archit. Code Optim., vol. 9, no. 4, p. 33, Jan. 2013.

[15] A. S. Molahosseini, S. Sorouri, and A. A. E arandi,"Researchchallenges in next-generation residue number system architectures," in Proc. IEEE Int. Conf. Comput. Sci. Educ., Jul. 2012, pp. 1658–1661.