International Journal of Engineering & Technology, *7 (4.6) (2018) 586-589*

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper

Implementation of Operand Decomposition and Carry Approximation for ECG Signal Denoising

Mr.T.R.Dinesh Kumar*[1],Dr.KMohanasundaram[2], Dr.M.Anto Bennet[3] , S.Kishore Kumar[4], M.Thangavel[5], M.Venkatesh[6],

[1]Assistant professor,[3]Professor,Electronics and Communication Engineering, Vel tech, Chennai-600062
[2]Professor,Electrical and Electronics Engineering,Vel Tech Multitech Dr. Rangarajan Dr. Sakunthala Engineering College,,Chennai,Tamilnadu,India
[4,5,.6]UG students, Department of Electronics and Communication Engineering,
VEL TECH Chennai-600 062.
**Email: thangavel1652@gmail.com*

Abstract

A novel RNA based architecture topology that resulted more effective than prefix topology. Here, A tendancy to describe the stages in RNA prefix adders is divided and presents a modulo residue detection network that reduces error chances compared to previous approaches. Finally through FPGA based hardware synthesis performance is evaluated and latency variation is proved through exhaustive test bench simulation. For high speed design we carried multiply and Accumulate (MAC) using proposed adder and inner products are calculated through sequent addition.

Keywords: *Multiply and Accumulate (MAC), Residue Number Algorithm (RNA), FIR filter, Fast Fourier Transform (FFT), Computation Intensive Arithmetic Functions(CIAF), MATLAB, FPGA.*

1. Introduction

Multiplication is a very important elementary function in arithmetic operations. Multiplication-based operations like Multiply and Accumulate(MAC) and real number are among a number of the often used Computation Intensive Arithmetic Functions(CIAF) presently enforced in several Digital Signal process (DSP)applications like convolution, fast Fourier Transform(FFT), filtering and in microprocessors in its arithmetic and logic unit. Since multiplication dominates the execution time of most DSP algorithms, therefore there's a requirement of high speed number. Currently, multiplication time continues to be the dominant factor in deciding the instruction cycle time of a DSP chip. The demand for prime speed process has been increasing as a results of increasing deskt op and signal process applications. Higher output arithmetic operations are vital to realize the required performance in several time period signal and image process applications.

One among the key arithmetic operations in such applications is multiplication and therefore the development of fast multiplier circuit has been a topic of interest over decades. Reducing the time delay and power consumption are terribly essential needs for several applications. In recent years, RNS computations have attracted robust interest for reducing vital path delays by exploiting the tradeoffs between reliability and performance. RNA modulo adders, that combine to attain high performance for low area overhead over ancient adders.

2. Proposed System:

In order to propose RNS arithmetic for any DSP applications both RNS addition/ multiplication will be carried out. To prove the overall RNS system efficiency over conventional MAC units in terms of delay /cost/power.To carry out fully RNS arithmetic based digital FIR filter.

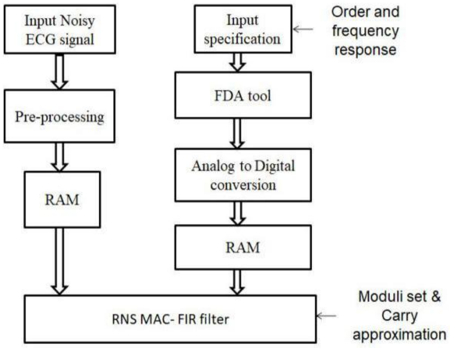

Input Noisy ECG signal
Pre-processing
RAM
Input specification
Order and frequency response
FDA tool
Analog to Digital conversion
RAM
RNS MAC- FIR filter
Moduli set & Carry approximation

Fig 1: Outline of ECG Signal Processing

2.1. Residue Number System

In fig. 1, The primary is absence of carry-propagation in addition and multiplication, carry-propagation being the foremost important speed-limiting consider these operations. The second is that as a result of the residue representations carry no weight-information, a slip-up in any digit-position in a given illustration doesn't have an effect on alternative digit-positions. and also the third is that there's no significance-ordering of digits in an RNS illustration, which implies that faulty digit-positions is also discarded with no impact aside from reduction in dynamic range. The new interest in RNS wasn't durable, for three main reasons: One, a whole arithmetic unit ought to be capable of a minimum of addition, multiplication, division, square-root, and comparisons, however implementing the last three in RNS is not easy; 2, technology became a lot of reliable; and, three, changing from RNS notation to traditional notation, for “human consumption", is difficult. all the same, in recent years there has been revived interest in RNS. There are many reasons for this new interest, together with the subsequent. an excellent deal of computing currently takes place in embedded processors, like those found in mobile devices, and for these high speed and low-power consumption are critical, the absence of carry-propagation facilitates the conclusion of high-speed, low-power arithmetic. Also, computer chips are currently going to be therefore dense that full testing cannot be possible; thus fault-tolerance and also the general space of process integrity have again become more necessary. Lastly, there has been progress within the implementation of the tough arithmetic operations.

2.2. RNS Representation:

High Speed: The absence of carry propagation between the arithmetic blocks leads to high speed process.Reduced Power: using small arithmetic units in realizing th e RNS processor reduces the change activities in every channel. This leads to reduction within the dynamic power, since the dy namic power is directly proportional to change activities.

Reduced Complexity: As a result of the RNS illustration encodes massive numbers into little residues, the complexity of the arithmetic units in every modulo channel is reduced. This facilitates and simplifies the design.

Error Detection and Correction: The RNS could be a non-positional system with no dependence between its channels. Thus, a mistake in one channel doesn't propagate to alternative channels. Therefore, isolation of the faulty residues permits fault tolerance and facilitates error detection and correction

2.3. FPGA Architecture:

The most common FPGA design consists of an array of Configurable Logic Blocks (CLBs), I/O pads, and routing channels. Generally, all the routing channels have an equivalent width (number of wires). Multiple I/O pads might work into the peak of one row or the width of one column within the array. An application circuit should be mapped into an FPGA with adequate resources. whereas the amount of CLBs and I/Os needed is well determined from the look, quantity of routing tracks required might vary significantly even among designs with an equivalent amount of logic. (For example, a crossbar switch needs much more routing than a pulsation array with an equivalent gate count.) Since unused routing tracks increase the price (and decrease the performance) of the part without providing any profit, FPGA makers attempt to give only enough tracks in order that most styles which will fit in terms of LUT’s and IO’s may be routed. This is often determined by estimates like those derived from Rent's rule or by experiments with existing designs.

2.4. FDA Tool:

The Filter design and Analysis Tool (FDA Tool) could be a powerful interface for coming up with and analyzing filters quickly as shown in fig. 1. FDA Tool allows you to design digital FIR or IIR filters by setting filter specifications, by importation of filters from your MATLAB space, or by adding, moving or deleting poles and zeros. FDA Tool additionally provides tools for analyzing filters, like magnitude and section response and pole-zero plots.

3. Filter Design

FDA Tool is interactive GUI model that enables to get frequency responses of any given filter specifications for any order. Magnitude response of filter is verified by the time of coefficients generation itself. Frequency responses of the accurate ECG elements are predicted to set FIR design.

3.1. RNS- MAC Filter:

Encoding sizable amount into a bunch of small numbers leads to important speed from the data processing. An RNS is outlined by a group of moduli. every whole number will be described as a collection of smaller integers known as the residues. RNS has long been applied in implementations of digital signal processing hardware like Digital-to-Analog converters, Finite Impulse Response (FIR) filters, Infinite Impulse Response (IIR) Filters, 2-D FIR filters.RNS has also been used in the design of 1-dimensional and 2-dimensional Discrete Wavelet Transform architectures.RNS has also been used to speed up RSA encryption and decryption and for high-speed Elliptic Curve Cryptography (ECC).

3.2. Forward Conversion:

The process of encryption the input file into RNS illustration is termed as Forward Conversion. This method are often done by dividing the given typical range by all the moduli and finding the remainders of the divisions

3.3. Reverse Conversion:

In fig. 2, The process of conversion from RNS representation to conventional representation is called Reverse Conversion. The reverse conversion process is more difficult and introduces more overhead in terms of speed and complexity .To overcome this problem, we are using operand decomposition and carry approximation.

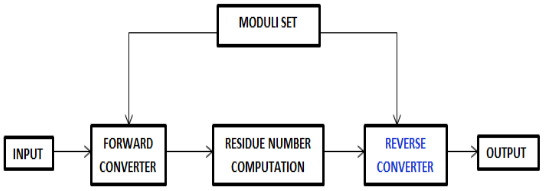


Fig. 2: Residue Conversion model

The basic concept adopted in most previous general modular adders was the use of two different n-bit binary adders; one to compute A + B and one to compute A + B − m. Input, output, and internal sub module interconnections were routed in two distant groups, where each group dealt with one adder. Additionally, a gate-based multiplexing stage was used to select one of the two outputs as shown in fig. 2. Different alterations to this capacious concept were introduced here to reduce either area, time, or both.

3.4. Multiplexing and Sum Computation Stage:

Based on the output carry bit cout, one of the two vectors H and H′, produced in the pre-processing stage, will be selected. If cout = 0, H is selected. Otherwise, H′ is selected. Similarly, one of the two vectors C and C′, produced in the parallel-prefix stage, will be selected at this stage, where C is selected if cout = 0, and C′ is selected otherwise. The two selected vectors are then exclusive OR. The proposed multiplexing and sum computation circuit for n = 7. Multiplexing is realised using unidirectional tristate buffers.

3.5. Modelsim

Supporting standard HDLs, ModelSim increases design quality and debug productivity.The graphical user interface is powerful, consistent, and intuitive. All windows update automatically following activity in any other window. For example, selecting a design region within the Structure window mechanically updates the supply, Signals, Process, and Variables windows. Edit, recompile, and re-simulate without going away the ModelSim setting.

4. Model Sim Output

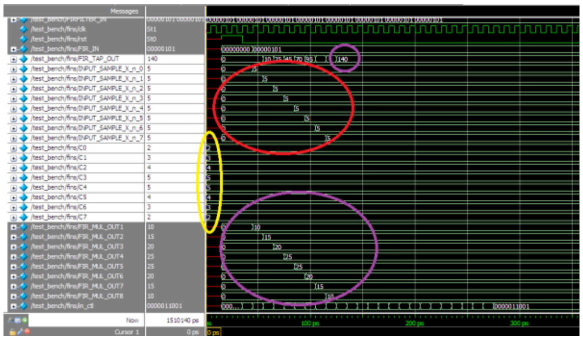


Fig 3: Simulation output – FIR convolution

The input ECG signal is added with the gaussian white noise to obtain the noisy ecg signal. The noise signal is then pre-processed to obtain the signal coefficients as shown in fig.3. The pre- processing consists of finding the signal coefficients by using quantizer. The coefficients of the noisy ECG signal is then stored in a RAM.On the other hand, the order and frequency response are obtained to define the filter coefficients. The filter coefficients are then converted to digital representation by the use of analog to digital converter and then finally stored in the RAM. The moduli set for finding the residue has to be declared.

In fig. 4, It has been shown that frequency range of interest, the selection of the actual coefficient subset and the number of FIR that are employed in the classification.The simulation result of the noisy ECG signal is obtained from MODELSIM shows that the range of clarification enhanced than the previous simulation techniques.

The RNS MAC Multiplier has less number of gates required for given 8x8 bits Multiplier so its power dissipation is very small as compared to other multiplier architecture. As shown in fig. 5, It has less switching activity as compared to other architecture for same operation. Due to parallel computation method, it has to process the less number of multiplication cycle. Thus, It requires only less amount of power than the other computational methods.

From fig. 6,It has been proven that the area needed for RNS-MAC is very small as compared to other multiplier architectures i.e. the number of devices used in RNS-MAC multiplier are 259.while Booth and Array Multiplier is 592 and 495 respectively for 16 x 16 bit number when implemented on Spartan FPGA.As shown in fig. 6,Thus the result shows that the RNS-MAC multiplier is smallest and the fastest of the reviewed architectures. Due to its parallel and regular structure, this architecture can be easily realized on silicon and can work at high speed without increasing the clock frequency. Speed improvements are gained by parallelizing the generation of partial products with their concurrent summations. It is demonstrated that this design is quite efficient in terms of silicon area. Such a design should enable substantial savings of resources in the FPGA when used for signal processing application.

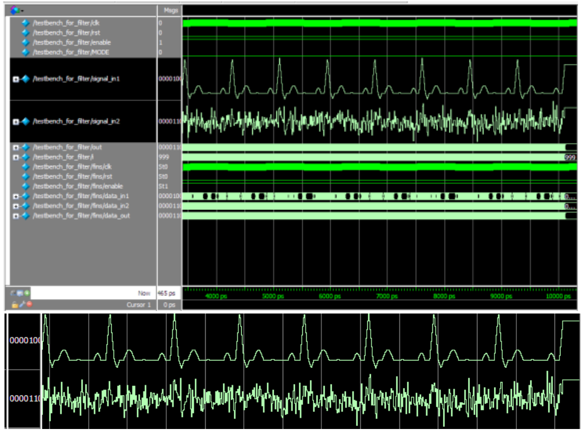
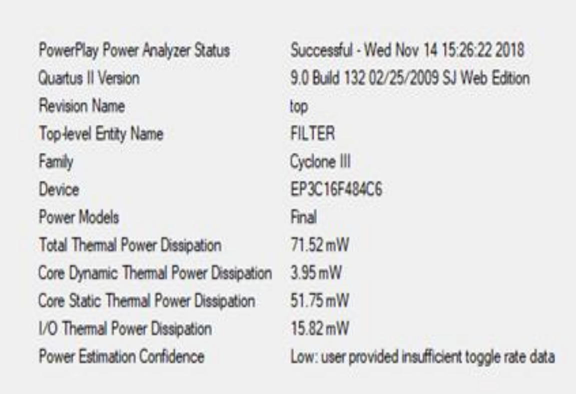
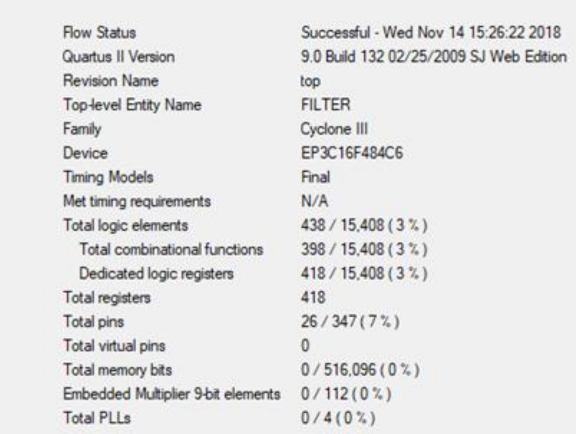


Fig. 4: ECG de-noising



PowerPlay Power Analyzer Status	Successful - Wed Nov 14 15:26:22 2018
Quartus II Version	9.0 Build 132 02/25/2009 SJ Web Edition
Revision Name	top
Top-level Entity Name	FILTER
Family	Cyclone III
Device	EP3C16F484C6
Power Models	Final
Total Thermal Power Dissipation	71.52 mW
Core Dynamic Thermal Power Dissipation	3.95 mW
Core Static Thermal Power Dissipation	51.75 mW
I/O Thermal Power Dissipation	15.82 mW
Power Estimation Confidence	Low: user provided insufficient toggle rate data

Fig. 5: Power analysis Report



Flow Status	Successful - Wed Nov 14 15:26:22 2018
Quartus II Version	9.0 Build 132 02/25/2009 SJ Web Edition
Revision Name	top
Top-level Entity Name	FILTER
Family	Cyclone III
Device	EP3C16F484C6
Timing Models	Final
Met timing requirements	N/A
Total logic elements	438 / 15,408 (3 %)
Total combinational functions	398 / 15,408 (3 %)
Dedicated logic registers	418 / 15,408 (3 %)
Total registers	418
Total pins	26 / 347 (7 %)
Total virtual pins	0
Total memory bits	0 / 516,096 (0 %)
Embedded Multiplier 9-bit elements	0 / 112 (0 %)
Total PLLs	0 / 4 (0 %)

Fig. 6: Area utilization Report

5. Performance Comparison

Table.1; Performance Comparison Table

Multiplier Model	Design complexity (LEs)	Speed (MHz)
Conventional MAC	287	155.64 MHz
RNS MAC	164	234.03 MHz
RNS using carry approximated model	125	366.17 MHz

The above table shows the design complexity and speed in MHz. It can be seen that the RNS-MAC computation has higher speed and also a decrease in the design complexity.to increase the speed further, applying carry approximation in the reverse conversion.Since, In RNS computation, reverse conversion takes up more time due to accumulation of larger numbers. Here, Comparison of various computational methods shows the significance of RNS computation in signal processing.

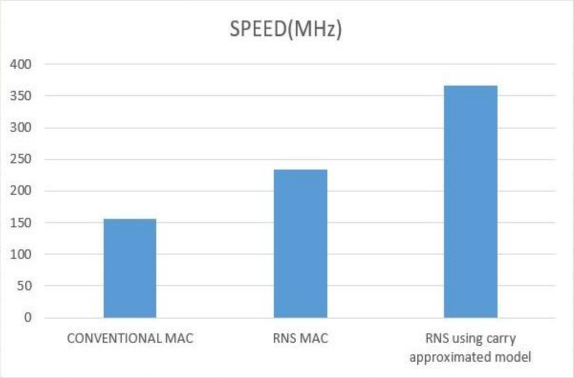


Fig 7: Speed Comparison

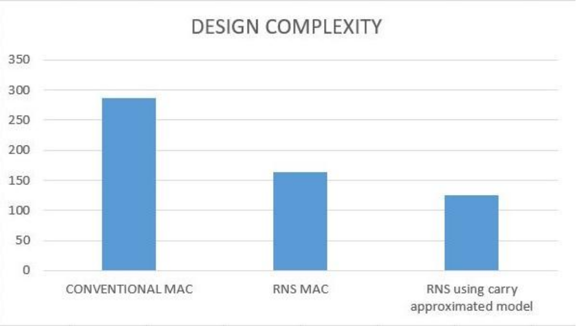


Fig 8: Design Complexity Comparison

Both fig. 7 and fig. 8 gives the pictorial representation of performance comparison among conventional MAC, RNS-MAC and RNS using carry approximation model. It clearly shows that the RNS using carry approximation model gives enhanced results with less design complexity and increased speed execution.

6. Conclusion

In general both multiplier and Adder component is a crucial component in any DSP and DIP applications; here RNS based computation is validated which leads merits over high speed and low complexity MAC unit while carry approximated model helps with the reduction of the maximum adder depth followed by the prefix based accumulation of FIR coefficients. Here we intend to perform modulo adder design for high-speed hardware RNS structure using MUX based post computations. To extend the proposed RNS based model that adding techniques to carried out FIR filtering in ECG signal de-noising applications. The efficiency of improved RNS coded MAC is verified using FIR implemented filter unit. And finally the complete trade of metrics of RNS encoded multiplier unit with prefix topology based accumulation unit is validated using hardware synthesis.

References:

[1] Omondi and B. Premkumar (2007),‘Residue Number Systems: Theory and Implementations’. Imperial College Press, London, U.K.

[2] Parhmi (2010),’Computer Arithmetic: Algorithms and Hardware Designs’, 2nd ed., Oxford Univ. Press, New York, NY, USA

[3] J. Chen and J. Hu (Jul. 2013), ‘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.

[4] C. H. Vun, A. B. Premkumar, and W. Zhang (Aug. 2013), ‘A new RNS based DA approach for Proc. IEEE Int. Conf. Comput. Sci. Educ., pp. 1658–1661.

[5] S. Antao and L. Sousa (Jan. 2013),’The CRNS framework and its application to programmable and reconfigurable cryptography’, ACM Trans. Archit. Code Optim., vol. 9, no. 4, p. 33.

[6] A. S. Molahosseini, S. Sorouri, and A. A. E. Zarandi (Jul. 2012),’Research challenges in next generation residue number system architectures’, in inner product computation’, IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 60, no. 8, pp. 2139–2152.

[7] K..Navi , A. S. Molahosseini, and M. Esmaeildoust (Feb. 2011),’How to teach residue number system to computer scientists and engineers’, IEEE Trans. Educ., vol. 54, no. 1, pp. 156–163.

[8] Y. Wang, X. Song, M. Aboulhamid, and H. Shen (Jul. 2002),’Adder based residue to binary numbers converters for (2n − 1, 2n, 2n + 1)’, IEEE Trans. Signal Process., vol. 50, no. 7, pp. 1772–1777.

[9] Dr. AntoBennet, M, Sankar Babu G, Suresh R, Mohammed Sulaiman S, Sheriff M, Janakiraman G ,Natarajan S, “Design & Testing of Tcam Faults Using TH Algorithm”, Middle-East Journal of Scientific Research 23(08): 1921-1929, August 2015 .

[10] Dr. AntoBennet, M “Power Optimization Techniques for sequential elements using pulse triggered flipflops”, International Journal of Computer & Modern Technology , Issue 01 ,Volume01 ,pp 29-40, June 2015.

[11] Dr. AntoBennet, M,Manimaraboopathy M,P. Maragathavalli P,Dinesh Kumar T R, “Low Complexity Multiplier For Gf(2m) Based All One Polynomial”, Middle-East Journal of Scientific Research 21 (11): 2064-2071, October 2014.