IJISET - International Journal of Innovative Science, Engineering & Technology, Vol. 4 Issue 4, April 2017
ISSN (Online) 2348 – 7968 | Impact Factor (2016) – 5.264
www.ijiset.com

Design and Implementation of Efficient Architecture for High Speed Convolution and Deconvolution Process

Ms.Priya R.Jain[1], Prof. Dr.Ujwalla A . Bellorkar [2]

[1]PG Student ,ENTC Department ,HVPM'S College of Engineering and Technology, Amravati ,India

[2]Head & Asst. Professor, ENTC Department ,HVPM'S College of Engineering and Technology, Amravati ,India

Abstract

In Digital Signal Processing, the convolution and deconvolution with a very long sequence is ubiquitous in many application areas. They consume much of time. This paper presents a direct method of computing the discrete linear convolution, circular convolution and deconvolution. The most significant aspect of the proposed method is the development of a multiplier and divider architecture based on high speed algorithm. It shows that the implementation of linear convolution and circular convolution is efficient in terms of area and speed compared to their implementation using conventional multiplier & divider architectures.

This paper describes implementation of modified convolution and deconvolution using Booth multiplier and Vedic multiplier respectively.

This is implemented on FPGA platform using Xilinx software for synthesis and simulated on Modelsim.

.

Keywords: *multiplier and divider architecture based on high speed algorithm, Radix-2 Booth Multiplier, radix-4 Modified Booth Multiplier.*

1. Introduction

Convolution and deconvolution is the most important and fundamental concept in signal processing and analysis. However, beginners often struggle with convolution and decovolution because the concept and computation requires a number of steps that are tedious and slow to perform. Therefore many of researchers have been trying to improve performance parameters of convolution and deconvolution system using new algorithms and hardware. Complexity and excess time consumption are always the major concern of engineers which motivates them to focus on more advance and simpler techniques. Pierre and John have implemented a fast method for computing linear convolution, circular convolution and deconvolution .This method is similar to the multiplication of two decimal numbers and this similarity makes this method easy to learn and quick to compute. Also to compute deconvolution of two finite length sequences, a novel method is used. This method is similar to computing long-hand division and polynomial division .Following diagram shows the overall process of high speed convolution and deconvolution process. With the latest advancement of VLSI technology, digital signal processing plays a pivotal role in many areas of electrical engineering. of Digital Signal Processing and Image Processing. It is used for designing of digital filter and correlation application. Discrete convolution is central to many applications. The most commonly taught approach is a graphical method because of the visual insight into the convolution mechanism. The principal components required for implementation of convolution calculation are adder and multiplier for partial multiplication. Therefore the partial multiplication and addition are bottleneck in deciding the overall speed of the convolution implementation technique. Complexity and excess time consumption are always the major concern of engineers which motivates them to focus on more advance and simpler techniques.

.

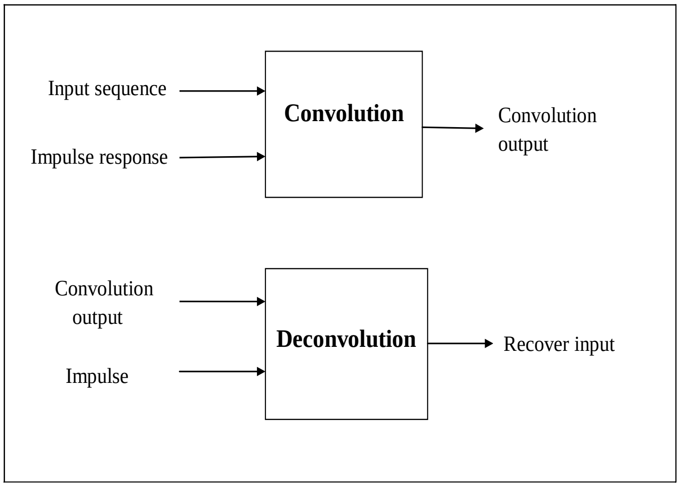


fig 1.1 Block diagram of Convolution and Deconvolution process

2. Literature review

Surabhi Jain & Sandeep Saini [1] Presented a direct method of computing the discrete linear convolution, circular convolution and deconvolution. The approach is easy to learn because of the similarities to computing the multiplication of two numbers. The most significant aspect of the proposed method is the development of a multiplier and divider architecture based on Ancient Indian Vedic Mathematics sutras Urdhvatriyagbhyam and Nikhilam algorithm. The results show that the implementation of linear convolution and circular convolution using vedic mathematics is efficient in terms of area and speed compared to their implementation using conventional multiplier & divider architectures.

G.Ramanjaneya Reddy, A. Srinivasulu [2] Presented an on the spot methodology of reducing convolution processing time using hardware computing and implementations of discrete linear convolution of two finite length sequences (NXN). This implementation method isrealized by simplifying the convolution building blocks. The purpose of this analysis is to prove the feasibility of an FPGAthat performs a convolution on an acquired image in real time. . In addition, the presented circuit uses less power consumption and delay from input to output. It additionally provides the required modularity, expandability, and regularity to form different convolutions for any variety of bits.-+.

Sukhmeet Kaur, Suman and Manpreet Signh Manna [3] Describes implementation of radix-4 Modified Booth Multiplier and this implementation is compared with Radix-2 Booth Multiplier. Modified Booth's algorithm employs both addition and subtraction and also treats positive and negative operands uniformly .No special actions are required for negative numbers. The Speed and Circuit Complexity is compared, Radix-4 Booth Multiplier is giving higher speed as compared to Radix-2 Booth Multiplier and Circuit Complexity is also less as compared to it.

Madhura Tilak [4] Presented a novel method of implementing linear convolution of two proposed method uses modified design approach by replacing the conventional multiplier by Vedic multiplier internally in the implementations. The proposed method is efficient in terms of computational speed, hardware resources and area significantly. The efficiency of the proposed algorithm is tested by simulations and comparisons with different design approaches.The proposed circuit is also modular, expandable and regular which provides flexibility.

Asmita Haveliya [5] Describes a block convolution process which is proposed using a multiplier architecture based on vertical and crosswise algorithm of Ancient Indian Vedic Mathematics and embedding it in OLA method for reducing calculations. Found on embedding Vedic multiplication for OLA, there is a considerable improvement in their performance Overlap-Add method (OLA) and Overlap-Save method (OLS) methods are employed.

Rashmi . Lomte [6] Describes a method of two finite length sequences (NXM), is implemented using direct method to reduce deconvolution processing time.The perfomance of the circuit has a delay of 79.595 ns from input to output using 90nm process.The outcome of research is high speed deconvolver implementation is achieved. Since 4×4 bitmultiplier is need of this project, different 4×4 bit multipliersare studied and Urdhava Triyakbhyam algorithm which gives lowest delay among remaining all multipliers is used .

Honey Durga Tiwari, Ganzorig Gankhuyag ,Chan Mo Kim, Yong Beom Cho [7] Describes New multiplier and square architecture is proposed based on algorithm of ancient Indian Vedic Mathematics, for low power and high speed applications. It is based on generating all partialproducts and their sums in one step. The design implementationon ALTERA Cyclone –II FPGA shows that the proposed Vedic multiplier and square are faster than array multiplier and Booth multiplier.

3.Proposedwork

Convolution is an important mathematical tool in both fields of signal and image processing. It is employed in filtering , denoising,edge detection, correlation, compression, deconvolution,simulation, and in many other applications Deconvolution is a computationally intensive digital signal processing (DSP) function widely used in applications such as imaging, wireless communication, and seismology.

3.1 Convolution

Convolution is considered to be heart of the digital signal processing. It is the mathematical way of combining twosignals to obtain a third signal. Convolution helps to estimate the output of a system with arbitrary input, withknowledge of impulse response of the system. Linear systems characteristics are completely specified by the systems impulse response, as governed by the mathematics of convolution. Convolution is an operation which takes two functions as input, and produces a single function output (much like addition or multiplication of functions).Consider two finite length sequences f(n) and h(n) on which the convolution operation is to be performed with lengths l and m respectively. The output of convolution operation y(n) contains l+m-1 number of samples.

The linear convolution of f(n) and h(n) is given by :

$$y(n) = (\ n) *h(n) \qquad (1)$$

$$y[n] = \sum_{k=-\infty}^{\infty} x(k)h(n-k) \qquad (2)$$

3.2 Deconvolution

If the impulse response and the output of a system are known, then the procedure to obtain the unknown input is referred to as deconvolution.The concept of deconvolution is also widely used in the techniques of signal processing and image processing.In general, the object of deconvolution is to find thesolution of a convolution equation of the form:

$$x * h = y \qquad (1)$$

Usually, yis some recorded signal, and xis some signalthat wish to recover, but has been convolved with someother signal hbefore get recorded. The function hmightrepresent the transfer function of an instrument or a drivingforce that was applied to a physical system.
Know h or at least form of h,then one can perform deterministicdeconvolution.

If the two sequences x(n) and h (n) are causal, then the convolution sum is

$$y[n] = \sum_{k=0}^{n} x(k)h(n-k) \qquad (2)$$

Therefore, solving for x(n) given h(n) and y(n) results in

$$x(n) = \frac{y(n) - \sum_{k=0}^{n-1} x(k)h(n-k)}{h(n)}$$

Where

$$x(0) = \frac{y(0)}{h(0)}$$

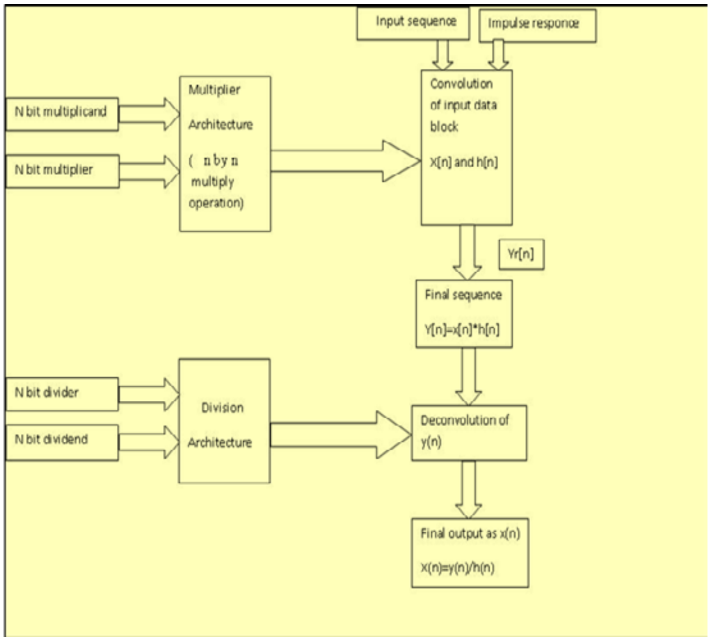

Input sequence
Impulse responce
N bit multiplicand
N bit multiplier
Multiplier Architecture (n by n multiply operation)
Convolution of input data block X[n] and h[n]
Yr[n]
Final sequence Y[n]=x[n]*h[n]
N bit divider
N bit dividend
Division Architecture
Deconvolution of y(n)
Final output as x(n) x(n)=y(n)/h(n)

Fig 3.1 block diagram of proposed system

3.3Conventional multiplier:

An array multiplier is a digital combinational circuit that is used for the multiplication of two binary numbers by employing an array of full adders and half adders.. To form the various product terms, an array of AND gates is used before the Adder array. To clarify more on the concept, let us consider a 4×4 bit multiplication with A and B being the multiplicand and the multiplier respectively. Assuming A = (1 0 0 1) and B= (1 0 1 0), the various bits of the final product can be written as:-

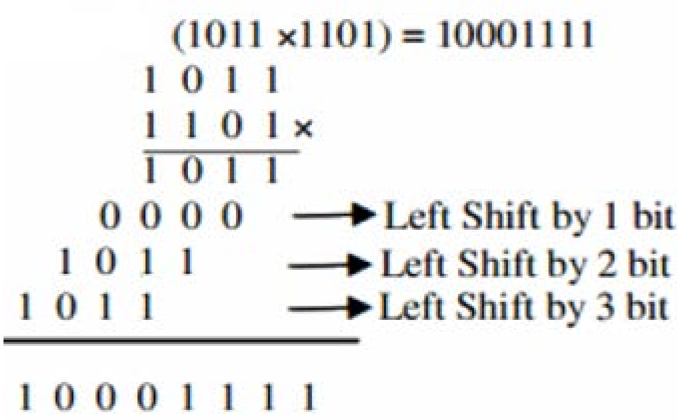


Fig.3.3 Example of conventional multiplier

3.4 Conventional divider:

Division is most complex and very time consuming if it is done straight forwardly, because we need to compare the remainder with the divisor after every subtraction Basicaly division algorithm is classified as multiplicative and subtractive approaches. Multiplicative division algorithms do not compute the quotient directly, but use successive approximations to converge to the quotient. Normally, such algorithms only yield a quotient, but with an additional step the final remainder can be computed, if needed. Consider the following example, Assuming A=(11100110) and B=(110) Example:

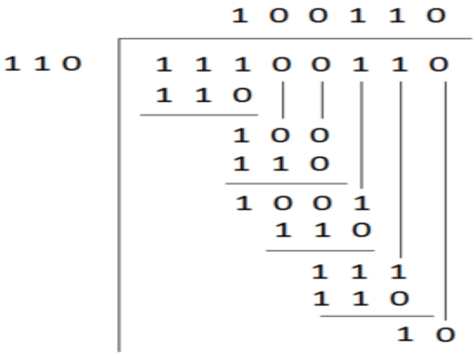


Fig 3.4 Example of conventional divider

3.5 Booth's multiplication algorithm

Booth's multiplication algorithm is a multiplication algorithm that multiplies two signed binary numbers in two's complement notation. The algorithm was invented by Andrew Donald Booth in 1950while doing research on crystallography at Birkbeck College in Bloomsbury, London. Booth used deskcalculators that were faster at shifting than adding and created the algorithm to increase their speed.Booth's algorithm is of interest in the study of computer architecture.

For each multiplier bit, also examine bit to its right
- **00**: middle of a run of 0s, do nothing
- **10**: beginning of a run of 1s, subtract multiplicand
- **11**: middle of a run of 1s, do nothing
- **01**: end of a run of 1s, add multiplicand

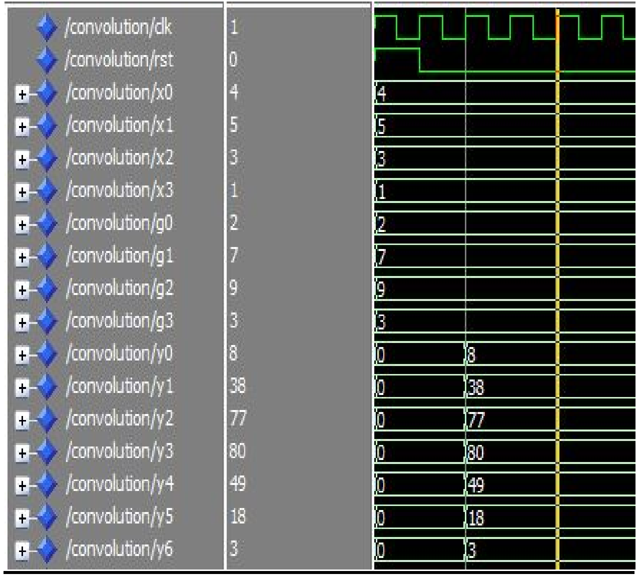
```
   43 = 00000101011
*  12 = 00000001100
    0 = 00000000000        multiplier bits 0
  + 0 = 00000000000        multiplier bits 00
- 172 = 11101010100        multiplier bits 10
  + 0 = 00000000000        multiplier bits 11
+ 688 = 01010110000        multiplier bits 01
  516   01000000100
```

4.Results

Using Modelsim, simulation is done for convolution process ,

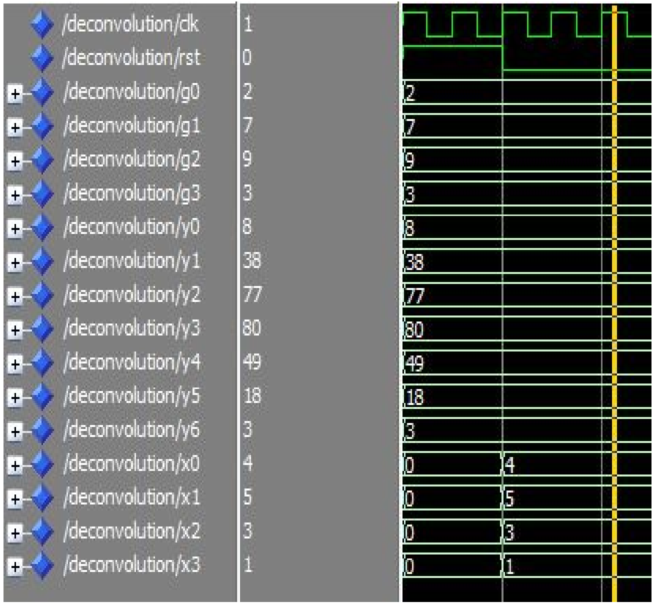
$$y[n] = \sum_{k=-\infty}^{\infty} x(k)h(n-k)$$

Output of convolution process



```
/convolution/clk   1
/convolution/rst   0
/convolution/x0    4
/convolution/x1    5
/convolution/x2    3
/convolution/x3    1
/convolution/g0    2
/convolution/g1    7
/convolution/g2    9
/convolution/g3    3
/convolution/y0    8    0  8
/convolution/y1    38   0  38
/convolution/y2    77   0  77
/convolution/y3    80   0  80
/convolution/y4    49   0  49
/convolution/y5    18   0  18
/convolution/y6    3    0  3
```

same for deconvolution process



```
/deconvolution/clk   1
/deconvolution/rst   0
/deconvolution/g0    2
/deconvolution/g1    7
/deconvolution/g2    9
/deconvolution/g3    3
/deconvolution/y0    8
/deconvolution/y1    38
/deconvolution/y2    77
/deconvolution/y3    80
/deconvolution/y4    49
/deconvolution/y5    18
/deconvolution/y6    3
/deconvolution/x0    4    0  4
/deconvolution/x1    5    0  5
/deconvolution/x2    3    0  3
/deconvolution/x3    1    0  1
```

Using Xilinx ,
Select the following values
♦ Product Category: **All**

♦ Family: **Spartan 6**
♦ Device: **XC6SLX16**
♦ Package: **CSG324**
♦ Speed: **-2**
We get proper outputs.

5. Conclusion

The execution time and area of the proposed method for convolution will be compared with that of convolution with the simple multiplication should be less.

An extension of the proposed linear convolution approach to circular convolution will also be introduced which should have less delay and area than the conventional method also introduced a straightforward approach to performing the deconvolution .

Radix-2, Booth Multipliers are implemented here; the complete process of the implementation is giving higher speed of operation.

The Speed and Circuit Complexity is compared.

Implementation is done of modified convolution and deconvolution using Booth multiplier and Vedic multiplier respectively.

This is implemented on FPGA platform using Xilinx software for synthesis and simulated on Modelsim.

Acknowledgments

We are very grateful to our college of HVPM College of Engineering and Technology to support and other faculty and associates of ENTC department who are directly & indirectly helped me for these paper.

References

[1] Surabhi Jain & Sandeep Saini “High Speed Convolution and Deconvolution Algorithm (Based on Ancient Indian Vedic Mathematics)” IEEE transaction on education ,ECTI-Conference, May 2014.

[2] G.Ramanjaneya Reddy, A. Srinivasulu “An Efficient Method for Implementation of Convolution” *International Archive of Applied Sciences and Technology* IAAST; Vol 4 [2] June 2013.

[3] Sukhmeet Kaur, Suman and Manpreet Signh Manna,”Implementation of Modified Booth Algorithm (Radix 4) and its Comparison with Booth Algorithm (Radix-2)”, *Advanced in Electronic & Electric Engineering* Vol 3, No.6 ,2013.

[4] Madhura Tilak “An Area Efficient, High Speed Novel VHDL Implementation of Linear Convolution of Two Finite Length Sequences Using Vedic Mathematics”, IJITEE Volume-2, Issue-1, December 2012.

[5] Asmita Haveliya “FPGA Implementation of a Vedic Convolution Algorithm“ ,*International Journal of Engineering Research and Applications (IJERA)* Vol. 2, Issue 1,Jan-Feb 2012.

[6] Rashmi Lomte and Bhaskar P.C., "High Speed Convolution and Deconvolution using Urdhva Triyagbhyam " ,2011 IEEE Computer Society Annual Symposium on VLSI ,p.323 ,July 2011.

[7] Honey Durga Tiwari, Ganzorig Gankhuyag, Chan Mo Kim, Yong Beom Cho “Multiplier design based on ancient Indian Vedic Mathematics”IEEE,SOC Design conference, Volume-2,NOV-2008.

[8] Very High Speed Integrated Circuit Hardware Description Language.
URL: http://electrosofts.com/vhdl/.

[9] John W. Pierre, “A Novel Method for Calculating the Convolution Sum of Two Finite Length Sequences”, IEEE transaction on education,VOL.39, NO. 1, 1996.

[10] J. G. Proakis and D. G. Manolakis, ”Digital Signal Processing: Principles Algorithm, and Applications,” 2nd Edition. NewYork Macmillan, 1992.

[11] Hsin-Lei Lin, Robert C. Chang, Ming-Tsai Chan (2004), Design of a Novel Radix-4 Booth Multiplier, IEEE Asia-Pacific Conference on Circuits and Systems, Vol.2, pp. 837-840.

[12] H. Lee, “A High-Speed Booth Multiplier”, ICCS, (2002).