ISSN(Online) : 2319 - 8753
ISSN (Print) : 2347 - 6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Special Issue 6, May 2015

Design of Finite Impulse Response Filter Using Distributed Arithmetic of Lookup Table

Manoshrudhy S [1], Maragatharaj S [2], Gayathri D [3]

P.G. Student, Department of Electronics and Communication Engineering, Knowledge Institute of Technology, Salem, India[1,2]

Assistant Professor, Department of Electronics and Communication Engineering, Knowledge Institute of Technology, Salem, India [3]

ABSTRACT: In digital signal processing filter plays a vital role in all applications. Generally FIR filter consists of multipliers and adders which require large space and high resource usage when implemented in FPGA. The usage of multipliers causes increase in area and increase in delay which ultimately reduce the performance of the filter. This problem can be overcome by a method of designing the filter using distributed arithmetic of three dimensional look up table is implemented. The filter designed by this method consists of look up table, shifter and accumulator instead of multiplier. Addition and shift operation is used for multiplication. As the order of the filter increases the look up table size also increased, to avoid this problem modified distributed arithmetic method is implemented in which the look up table is subdivided to reduce the memory size. In addition Distributed arithmetic allows pipelining of data which increases speed. To operate with various filter coefficient dynamic distributed arithmetic is used where as in distributed and modified distributed arithmetic only constant coefficient are applied. The filter design is implemented in Xilinx ISE Design suite 12.1. This is implemented with Spartan 3E FPGA device.

KEYWORDS: FIR filter, Distributed Arithmetic, Lookup Table, FPGA.

I. INTRODUCTION

In signal processing and telecommunication filters are the basic component and they are widely used in applications such as noise reduction, channel equalization, audio and video processing, biomedical signal processing, image processing, radar etc., Filters are used for frequency selection of a network. Based on the input and the output of the filter they are classified as analog and digital filter. And based on their impulse response they are classified as finite impulse response (FIR) and infinite impulse response (IIR) filter [1]. The major difference between FIR and IIR filter is that FIR filter has linear phase it is always stable and it has no analog history whereas IIR filter does not have linear phase and it is difficult to control, IIR is unstable and is derived from analog filter. IIR depends on both input and output whereas FIR depends only on present inputs. IIR consists of zero's and poles and require less memory whereas FIR consists of only zero's. With advances in digital technology they are rapidly replacing analog filters in which RLC components are used. For digital signal processing operation primary filters are digital filters. With the use of special DSP devices they are implemented as multiply and accumulate (MAC) algorithm [2]. The DSP devices are based on RISC architecture and consisting of fast array multipliers. Generally the multipliers are in the range of one to four and data is sequenced by the microprocessor to pass through it for multiply and other functions and intermediate result is stored in the accumulator. Performance is improved by increasing the clock speed used for multiplication. The strongest operation multiplication is performed by repeated addition which require portion hardware portion and chip area and power consumption is also high. When compared with multiply accumulate structure Memory-based structure are more regular and having greater potential for high throughput and has less latency with less power consumption [3]. For many Digital signal Processing (DSP) algorithms, memory-based structures are well suited for multiplication with fixed set of coefficients. For this FIR filter is designed using distributed Arithmetic which reduces resource usage in FPGA implementation.

ISSN(Online) : 2319 - 8753
ISSN (Print) : 2347 - 6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Special Issue 6, May 2015

II. FIR FILTER WITH MULTIPLIER

Finite impulse response (FIR) filter is a filter response to any finite length input is finite duration, because it settles to zero in finite time. Generally FIR filter consists of

1. Multipliers
2. Adders
3. Delay elements.

The following figure shows the structure of FIR filter.

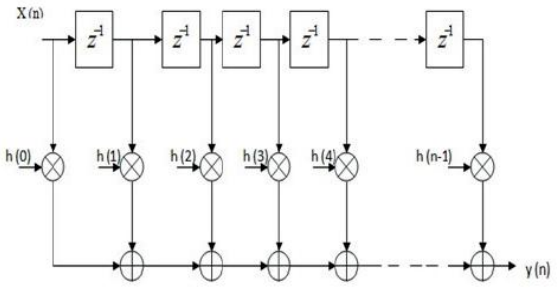

Fig. 1 FIR filter structure

The upper part of filter shows delay line with N+1 taps [4]. The output of the filter is determined by Convoluting its input signal x with its impulse response h and it is represented as

$$Y(n)= \sum_{K=0}^{N-1} C[K]\, X[n—K] \quad (1)$$

Where

X(n) is the input signal
Y(n) is the output signal
C(K) the filter coefficients which are also known as taps.

III. DISTRIBUTED ARITHMETIC

Digital filters can be implemented using distributed arithmetic. It replaces multiplications and addition by

1. Look up table
2. Adders
3. Shifters

It is used to compute the sum of product [5]. DA is bit-serial method of processing the data. The multiplication operation is replaced by a mechanism that generates partial products and then sums the products together. The partial products generation and their addition is the key difference between distributed arithmetic and standard multiplication. Multiplication and accumulation operation is represented as

$$Y= A1 \times X1 + A2 \times X2 + A3 \times X3 + \ldots . + Ak \times Xk$$

$$Y= \sum_{k=1}^{K} Ak \times Xk \quad (2)$$

where,
y - Output response
Ak - Constant filter coefficients
Xk - Input data
Let Xk be a N-bits and can be expressed in scaled two's complement number as

$$Xk = -bk0b + \sum_{}^{N-1} bkn2\text{-}n \quad (3)$$

ISSN(Online) : 2319 - 8753
ISSN (Print) : 2347 - 6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Special Issue 6, May 2015

n=1

Substituting the above equation in y, we get

$$Y = \sum_{k=1}^{K} Ak \left[-bk0 + \sum_{n=1}^{N-1} bkn2\text{-}n\right] \quad (4)$$

$$y = -\sum_{k=1}^{k} bk0.Ak + \sum_{k-1}^{k} \left[\sum_{n=1}^{N-1} (bkn.\ Ak)2\text{-}n\right] \quad (5)$$

Rearranging the power terms and grouping the sum of product terms we get the final result as

$$Y = -\sum_{k=1}^{k} b_{k0}.A_k + \sum_{n=1}^{N-1} \left(\sum_{k=1}^{k} b_{kn}.\ A_k\right)2^{-n} \quad (6)$$

IV. BLOCK DIAGRAM

The bock diagram which describes the operation of distributed arithmetic is shown below.

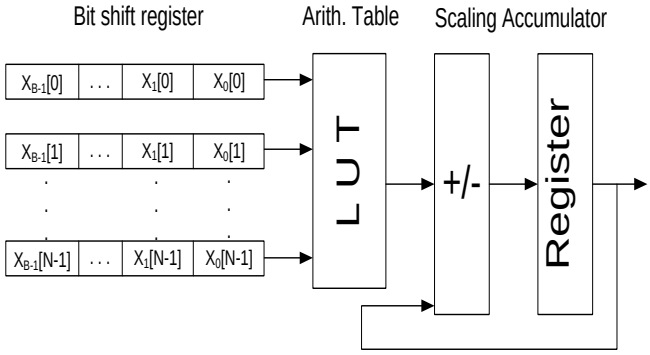

Fig.2. Block diagram of distributed arithmetic

LUT stores all the possible combinations of the sums of the coefficients in three dimensional order. Addition operation is to implement the Filter functionality by using LUT contents [6]. The register is used to shift the values again to arithmetic operation block. The major drawback of this method is that as the order of filter increased then look up table size also increased. Since we use three dimensional lookup table large amount of values are stored when compared to conventional lookup table.

Lookup table and DA technique for 3rd order filter

A lookup table is nothing but memory which stores the precomputed values for the possible combinations of input values. Here X is the input and A is the fixed coefficient.

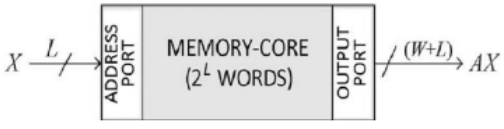

Fig.3. Conventional LUT-based multiplier

ISSN(Online) : 2319 - 8753
ISSN (Print) : 2347 - 6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Special Issue 6, May 2015

Let us consider the third order filter, if the number of coefficient is N the memory size required is 2N.
If the number of coefficients = 4
Number of inputs = 4
Then LUL size = 24 = 16 memory locations.
In this method possible outputs are pre computed and stored in LUT [7]. LUT addressed through input of the filter. For 4 tap filter, 4 tap represents the number of co-efficient of the filter as well as it represents the no. of inputs to the filter and address bit for the LUT.

Each location has different output for the corresponding inputs. The possible inputs for this filter is 0(0000) - 15(1111).For each input the computation of output is easy by using this technique.

If Input = 1011 means then
Output = 1.h0 + 0. h1 + 1. h2 + 1.h3 = h0+h2+h3

If Input = 1111 means then
Output = h0+ h1+h2+h3

If Input = 0101 means then
Output = h1+h3

If Input = 1010 means then
Output = h0+h2

It represents the addition of high level input co-efficient. We can easily find 16 output for corresponding input without any mathematical calculation. Table 1 shows the content of the LUT for 3rd order filter.

Address	Data
0000	0
0001	h_3
0010	h_2
0011	$h_2 + h_3$
0100	h_1
0101	$h_1 + h_3$
0110	$h_1 + h_2$
0111	$h_1 + h_2 + h_3$
1000	h_0
1001	$h_0 + h_3$
1010	$h_0 + h_2$
1011	$h_0 + h_2 + h_3$
1100	$h_0 + h_1$
1101	$h_0 + h_1 + h_3$
1110	$h_0 + h_1 + h_2$
1111	$h_0 + h_1 + h_2 + h_3$

Table.1. LUT for 3rd order filter

ISSN(Online) : 2319 - 8753
ISSN (Print) : 2347 - 6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Special Issue 6, May 2015

For higher order filter LUT size will increase, it required more memory space. For this the look up table is partitioned into smaller sizes.

V. LUT PARTITIONING

Size of look up table increases exponentially with the order of filter [8]. Therefore reduction of LUT into reasonable size for higher order filter is essential by dividing the look up table into smaller portions is called LUT partitioning [9]. Different sets of coefficients are operated on each partitioned LUT's.

For 3rd order filter

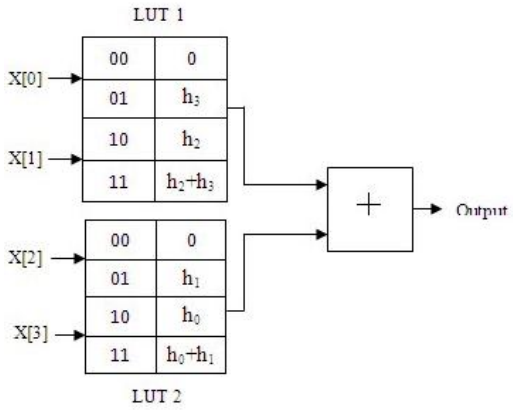

Fig.4. 3rd order FIR filter with partition method

Number of partition = 2
2 LUT tables are used. Each has 2 inputs [10]

Memory location = no .of partition * 2n
= 2*22
= 8 location
n = number of inputs of LUT.

LUT partitioning of 16 tab FIR filter

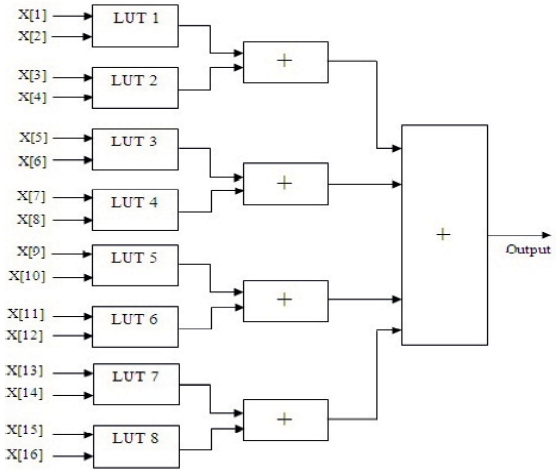

Fig.5. 16 tab FIR filter LUT partition

ISSN(Online) : 2319 - 8753
ISSN (Print) : 2347 - 6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Special Issue 6, May 2015

VI. COMPARISON

For a 3rd order FIR filters, Implementation of FIR filter by traditional method require
4 multiplier
3 adders

Distributed Arithmetic	18 memory locations
LUT partitioning Method	6 memory locations

Table.2. comparison results

From the above comparison it is well known that LUT partitioning method requires less memory than other methods.

VII. EXPERIMENTAL RESULTS

The following figures shows the RTL schematic of lookup table and result of lowpass FIR filter using Distributed Arithmetic.

Fig.6. RTL schematic for lookup table

The above block represents the schematic of 8- bit filter of input and output.

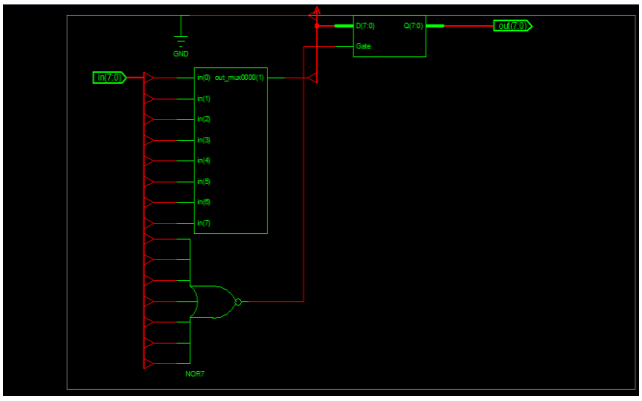

Fig.7. technology schematic for lookup table

ISSN(Online) : 2319 - 8753
ISSN (Print) : 2347 - 6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Special Issue 6, May 2015

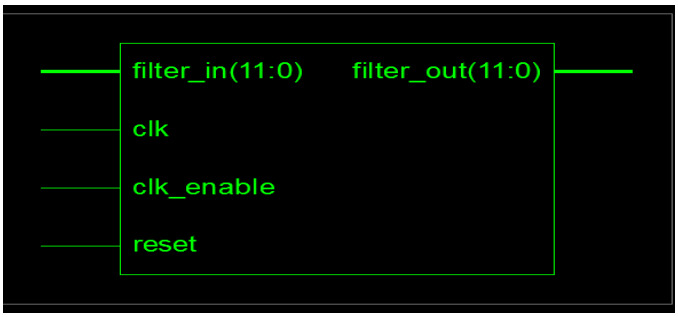

Fig.8. RTL schematic of lowpass FIR filter

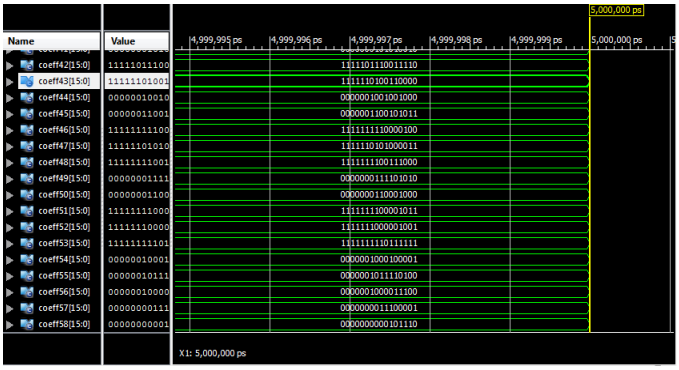

Fig.9. Simulation result of FIR filter

VIII. CONCLUSION

By using Distributed Arithmetic of three dimensional, FIR filter is designed and implemented. The filter is designed without multiplier by using LUT, shifter and adder. LUT is used for storing precomputed values so the memory access time is reduced. For higher order filter LUT size is reduced by LUT partitioning. Further this will be implemented using multidimensional lookup table.

REFERENCES

[1] ArathyIyer, Krishnapriya P.N, "Power And Area Efficient Implementation For Parallel FIR Filters Using FFAs And DA", International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering,Vol. 2, Special Issue 1, December 2013.

[2] Sirisha.A, Balanagu.P& Suresh Babu.N , "Design Of Fir Filter Using Look Up Table And MemoryBasedApproach", International Journal of Computer & Communication Technology ISSN (PRINT): 0975 - 7449, Volume-3, Issue-5, 2012.

[3] Chitra E, Vijay Kumar Ch, Leelakrishna Muthyala, "Design Of A High Speed FIR Filter On FPGA By Using DA-OBC Algorithm", International Journal of Engineering Research and General Science Volume 2, Issue 4, June-July, 2014K. Elissa, "Title of paper if known," unpublished.

[4] Jeevan Reddy K Mr. Shrikant Patel, " FIR Filter Implementation Using DA With LUT Less Design Structure Based On FPGA",International Journal of Modern Communication Technologies & Research (IJMCTR) ISSN: 2321-0850, Volume-2, Issue-3, March 2014.

[5] Kalaiyarasi.D, KalpalathaReddy.T, " Area Efficient Implementation Of FIR Filter Using Distributed Arithmetic With Offset Binary Coding",IOSR Journal of VLSI and Signal Processing (IOSR-JVSP) Volume 4, Issue 3, Ver. III (May-Jun. 2014), PP 01-09.

[6] KavyaJyothi.B, Dr. K.B.Shivakumar, Dr.M.Z.Kurian, Prof. Imran Rasheed, "Bp FIR Filter Implementation On FPGA Using Fully Parallel And Da Architecture",International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering An ISO 3297: 2007 Certified OrganizationVol. 3, Issue 4, April 2014.

[7] Nagakishore, Bhavanam, M. Keerthi , Vasujadevi Midasala , Jeevan Reddy K "Fpga Implementation Of Distributed Arithmetic For FIR Filter",International Journal of Engineering Research & Technology (IJERT) Vol. 1 Issue 9, November- 2012 ISSN: 2278-0181.

[8] RanjithKumar.T, "Design and implementation of DDA architecture for FIR Filters", International Journal of Engineering Trends and Technology (IJETT) – Volume 4 Issue 9- Sep 2013.

[9] Shrikant Patel, "Design And Implementation Of 31-Order Fir Low-Pass Filter Using Modified Distributed Arithmetic Based On FPGA", International Journal of Advanced Research in Electrical, Electronics and Instrumentation EngineeringVol. 2, Issue 10, October 2013.
Yazhini.M and Ramesh, "FIR Filter Implementation using Modified Distributed Arithmetic Architecture", Indian Journal of Science and Technology Print ISSN: 0974-6846 Vol 6 (5)ISSN: 0974-5645, May 2013.