

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 6, Issue 10, October 2017

High Speed 17-tap Finite Impulse Response Filter based on Distributive Arithmetic Technique

Seema Patidar and Prof. Alok Kumar

M. Tech. Scholar, Dept. of Electronics and Communication, Sagar Institute of Research Technology & Science
(SIRTS), Bhopal, India

Assistant Professor, Dept. of Electronics and Communication, Sagar Institute of Research Technology & Science
(SIRTS), Bhopal, India

ABSTRACT: Low complexity and configurability is the key features in emerging communication applications in order to support multi-standards and operation modes. To obtain these features, efficient implementations of finite impulse response (FIR) filter with multiplier-less distributive arithmetic technique is proposed in this paper. In this technique consist of Look Up Table (LUT), shift register and accumulator. Based on this technique multipliers in FIR filter are removed. Multiplication is performed through shift and addition operations. The LUT can be subdivided into a number of LUT to reduce the size of the LUT for higher order filter. Each LUT operates on a different set of filter taps. Analysis on the performance of various filter orders with different address length are done using Xilinx synthesis tool. The proposed architecture provides less latency and less area compared with existing structure of FIR filter.

KEYWORDS: Finite Impulse Response (FIR), Look Up Table (LUT), Distributive Arithmetic Technique

I. INTRODUCTION

A digital filter is a system that performs mathematical operations on a sampled or discrete time signal to reduce or enhance certain aspects of that signal. One type of digital filter is FIR filter. It is a stable filter. It gives linear phase response. Pipelining and parallel processing technique is used in FIR filter. Pipelining operation takes place in an interleaved manner. Pipelining done by inserting latches (delay element) in the system. It increases the overall speed of the architecture but the hardware structure and system latency will increase. Hardware structures increase due to inserting pipelining latches. For M level pipelining M-1 delay elements required. Latency is the difference between the availability of first output in the sequential system and the pipeline system [1]. At every clock cycle it will operate multiple inputs and produced multiple outputs is called parallel processing. It required extra hardware. Both pipelining and parallel processing has disadvantages. For FIR filters, output is a linear convolution of weights and inputs. For an Nth-order FIR filter, the generation of each output sample takes N+1 multiply accumulate (MAC) operations. Multiplication is strongest operation because it is repeated addition. It require large portion of chip area. Power consumption is more. Memory-based structures are more regular compared with the multiply accumulate structures; and have many other advantages, e.g., greater potential for high throughput and reduced-latency implementation and are expected to have less dynamic power consumption due to less switching activities for memory-read operations compared to the conventional multipliers. Memory based structures are well-suited for many digital signal processing (DSP) algorithms, which involve multiplication with a fixed set of coefficients. For this Distributed Arithmetic architecture used in FIR filter.

Distributed arithmetic is one way to implement convolution with multiplier less unit, where the MAC operations are replaced by a series of LUT access and summations. Distributed Arithmetic is a different approach for implementing

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

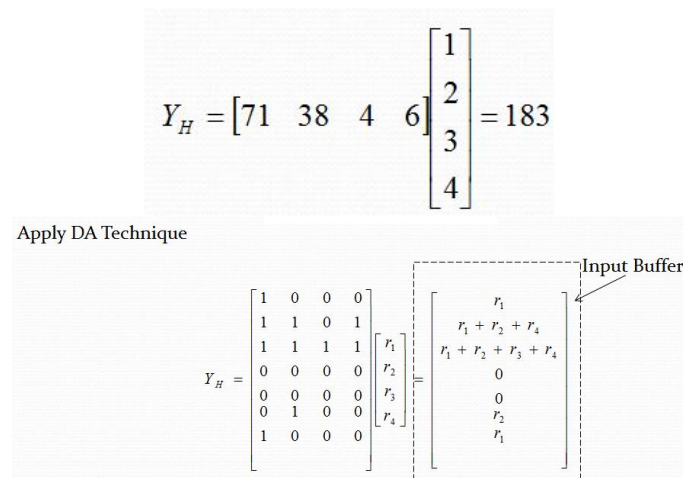
Visit: www.ijirset.com

Vol. 6, Issue 10, October 2017

digital filters. The basic idea is to replace all multiplications and additions by a table and a shifter-accumulator. LUT are the kind of logic that used in SRAM based FPGAs. Basically each look table is a bunch of single bit memory cells storing individual bit values in each of the cells. Memory access time is less in SRAM, so speed of the static RAM is high. Distributed Arithmetic provides cost-effective and area-time efficient computing structures. Digital Finite Impulse Response (FIR) filters are essential building blocks in most Digital Signal Processing (DSP) systems. A large application area is telecommunication, where filters are needed in receivers and transmitters, and an increasing portion of the signal processing is done digitally [2, 3]. However, power dissipation of the digital parts can be a limiting factor, especially in portable, battery-operated devices. Scaling of the feature sizes and supply voltages naturally helps to reduce power. For a certain technology, there are still many kinds of architectural and implementation approaches available to the designer.

II. DISTRIBUTIVE ARITHMETIC TECHNIQUE

Distributed arithmetic is one way to implement convolution with multiplier less unit, where the multiplication operations are replaced by a series of LUT access and summation.



step-1 all the input converts' binary number, Step-2 all the binary input applied to a input buffer array so,

$$\begin{array}{ll}
 P_1 = 0001 & P_2 = 0111 \\
 P_3 = 1010 & P_4 = 0000 \\
 P_5 = 0000 & P_6 = 0010 \\
 P_7 = 0001 &
 \end{array}$$

The entire adder array input applied to MUX so, the entire adder array input

$$\text{MUX}(1) = 0001 = Y_p(0)$$

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 6, Issue 10, October 2017

$$\begin{aligned} \text{MUX (1) add MUX (2)} &= Y_p (1) \\ &= 00001 \\ &= 01110 \\ &+ 01111 \end{aligned}$$

$$\begin{aligned} \text{Output of the } Y_p (1) \text{ again right shift 1-bit and adds MUX (3) so} \\ &= 001111 \\ &= 101000 \\ &+ 110111 \end{aligned}$$

$$Y_p (1) + \text{MUX (3)} = Y_p (2)$$

$$\begin{aligned} \text{Output of the } Y_p (2) \text{ again right shift 1-bit and adds MUX (6) so} \\ &= 000110111 \\ &= 001000000 \\ &+ 001110111 \end{aligned}$$

$$Y_p (2) + \text{MUX (6)} = Y_p (3)$$

Output of the $Y_p (3)$ again right shift 1-bit and adds MUX (7) so

$$\begin{aligned} &= 001110111 \\ &= 001000000 \\ &+ 010110111 \end{aligned}$$

$$\begin{aligned} Y_p (3) + \text{MUX (6)} &= Y_p (4) \\ \text{Total output } Y_p (4) &= 010110111 = 183 \end{aligned}$$

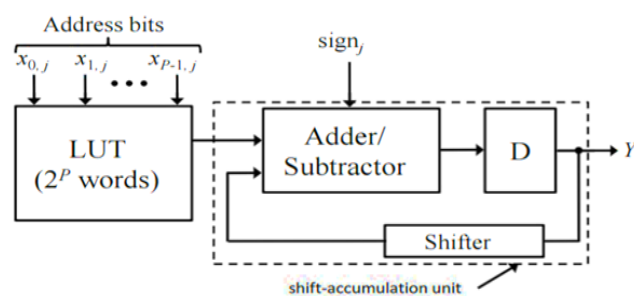


Figure 1: The block diagram of DA computation

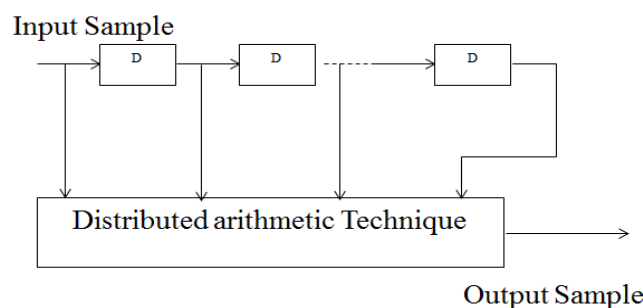


Figure 2: Multiplier-less Distributive Arithmetic Based FIR Filter

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 6, Issue 10, October 2017

III. PROPOSED METHODOLOGY

The above technique holds good only when we go for lower order filters. For higher order filters, the size of the LUT also increases exponentially with the order of the filter. For a filter with N coefficients, the LUT have $2N$ values. This in turn reduces the performance.

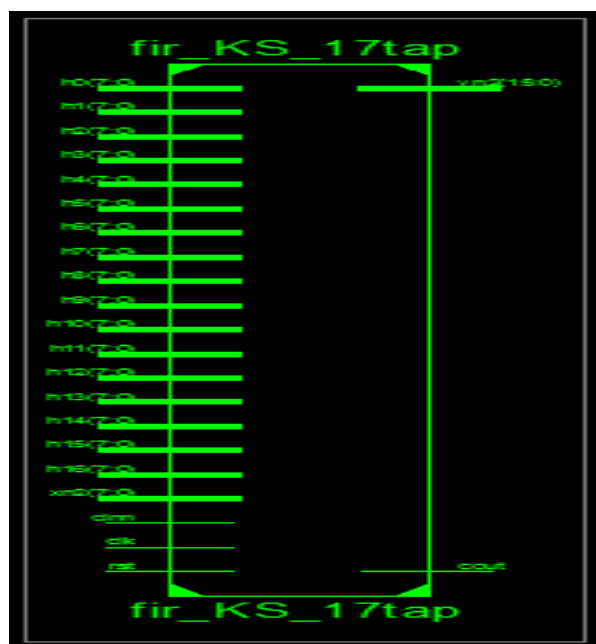


Figure 3: View Technology of 17-tap FIR Filter

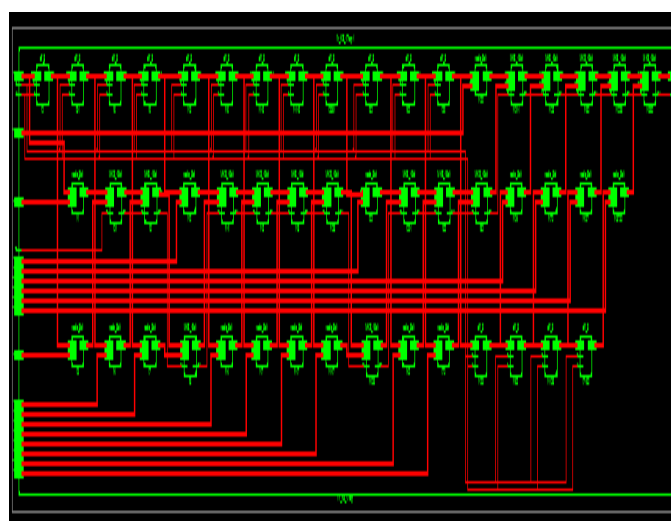


Figure 4: RTL View of 17-tap FIR Filter

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 6, Issue 10, October 2017

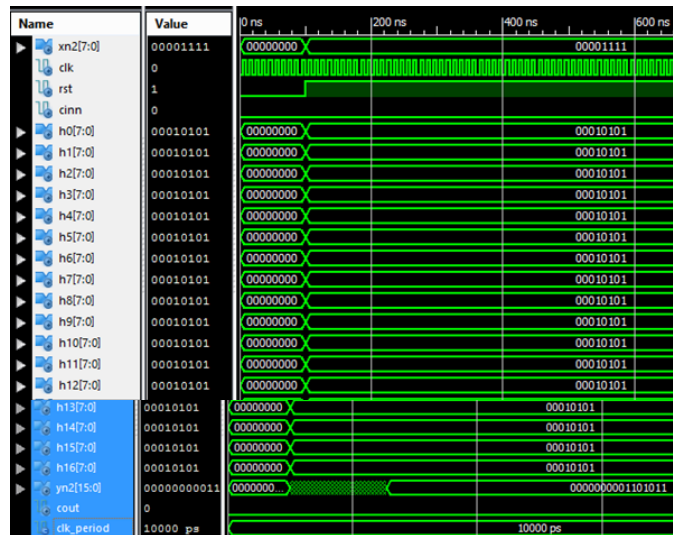


Figure 5: Output Waveform of 17-tap FIR Filter

```

Device utilization summary:
-----
Selected Device : 7vh290thcgl1155-2

Slice Logic Utilization:
Number of Slice Registers:      188 out of 437600  0%
Number of Slice LUTs:          1843 out of 218800  0%
    Number used as Logic:      1843 out of 218800  0%

Slice Logic Distribution:
Number of LUT Flip Flop pairs used: 1942
Number with an unused Flip Flop: 1754 out of 1942  90%
Number with an unused LUT:      99 out of 1942  5%
Number of fully used LUT-FF pairs: 89 out of 1942  4%
Number of unique control sets: 1

IO Utilization:
Number of IOs:                  164
Number of bonded IOBs:          164 out of 300  54%

Specific Feature Utilization:
Number of BUFG/BUFGCTRLs:      1 out of 32  3%
    
```

Figure 6: Device Utilization Summary of 17-tap FIR Filter

```

Timing Summary:
-----
Speed Grade: -2

Minimum period: 0.703ns (Maximum Frequency: 1421.666MHz)
Minimum input arrival time before clock: 1.183ns
Maximum output required time after clock: 14.882ns
Maximum combinational path delay: 14.659ns
    
```

Figure 7: Timing Summary of 17-tap FIR Filter

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 6, Issue 10, October 2017

IV. CONCLUSION

Finite Impulse Response filter plays an important role in many Digital Signal Processing applications. In this method, the multiplier less FIR filter is implemented using Distributed Arithmetic which consists of Look Up Table and then partitioning is involved. Memory access time is less than multiplication time. LUT partition reduces memory requirements. This technique reduces the delay, area, power consumption. The performance can be further improved by pipelining all the partial tables. This architecture provides an efficient area-time power implementation which involves significantly less latency and less area-delay complexity when compared with existing structures for FIR Filter.

REFERENCES

- [1] Indranil Hatai, Indrajit Chakrabarti, and Swapna Banerjee, "An Efficient VLSI Architecture of a Reconfigurable Pulse-Shaping FIR Interpolation Filter for Multi-standard DUC", IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Vol. 23, No. 6, June 2015.
- [1] G. Gokhale and P. D. Bahirgonde, "Design of Vedic Multiplier using Area-Efficient Carry Select Adder", 4th IEEE International Conference on Advances in Computing, Communications and Informatics (ICACCI-2015), Kochi, August 10-13, 2015, India.
- [2] G. Gokhale and Mr. S. R. Gokhale, "Design of Area and Delay Efficient Vedic Multiplier Using Carry Select Adder", 4th IEEE International Conference on Advances in Computing, Communications and Informatics (ICACCI-2015), Kochi, August 10-13, 2015, India.
- [3] Pavan Kumar, Saiprasad Goud A, and A Radhika had published their research with the title "FPGA Implementation of high speed 8-bit Vedic multiplier using barrel shifter", 978-1-4673-6150-7/13 IEEE.
- [4] B.Madhu Latha1, B. Nageswar Rao, published their research with title "Design and Implementation of High Speed 8-Bit Vedic Multiplier on FPGA" International Journal of Advanced Research in Electrical ,Electronics and Instrumentation Engineering, Vol. 3, Issue 8, August 2014.
- [5] A Murali, G Vijaya Padma , T Saritha, published their research with title "An Optimized Implementation of Vedic Multiplier Using Barrel Shifter in FPGA Technology", Journal of Innovative Engineering 2014, 2(2).
- [6] Sweta Khatri , Ghanshyam Jangid, "FPGA Implementation of 64-bit fast multiplier using barrel shifter" Vol. 2 Issue VII, July 2014 ISSN: 2321-9653.
- [7] Toni J.Billore, D.R.Rotake, "FPGA implementation of high speed 8 bit Vedic Multiplier using Fast adders" Journal of VLSI and Signal Processing, Volume 4, Issue 3, Ver. II (May-Jun. 2014), PP 54-59 e-ISSN: 2319 – 4200, p-ISSN No. : 2319 – 4197.
- [8] S. S. Kerur, Prakash Narchi, Jayashree C N, Harish M Kittur and Girish V A, "Implementation of Vedic Multiplier for Digital Signal processing" International Conference on VLSI, Communication & Instrumentation (ICVCI) 2011.
- [9] Vaibhav Jindal, Mr. Navaid Zafar Rizvi, Dinesh Kumar Singh "VHDL Code of Vedic Multiplierwith Minimum Delay Architecture" National Conference on Synergetic Trends in engineering and Technology (STET-2014) International Journal of Engineering and Technical Research ISSN: 2321-0869, Special Issue.
- [10] Bhavin D Marul, Altaf Darvadiya "VHDL Implementation of 8-Bit Vedic Multiplier Using Barrel Shifter" International Journal for Scientific Research & Development| Vol. 2, Issue 01, 2014 | ISSN (online): 2321-0613.