VLSI IMPLEMENTATION OF AREA EFFICIENT 2-PARALLEL FIR DIGITAL FILTER

L kholee phimu and Manoj kumar

Department of Electronic and Communication Engineering, NIT Manipur, India

ABSTRACT

This paper aims to implement an area efficient 2-parallel FIR digital filter. Xilinx 14.2 is used for synthesis and simulation. Parallel filters are designed by using VHDL. Comparison among primary 2–parallel FIR digital filter and area efficient 2-parallel FIR digital filter has been done. Since adders are less weight in term of silicon area, compare to multipliers. Therefore multipliers are replaced with adders for reducing area and speed of the filter. 2-parallel FIR filter is used in digital signal processing (DSP) application.

KEYWORDS

Finite impulse response (FIR), Booth multiplier, Carry-look-ahead adder (CLA), Digital Signal Processing (DSP), Parallel FIR, Very Large Scale Integration (VLSI).

1. INTRODUCTION

A finite impulse response (FIR) digital filter has no feedback and is used in DSP application, which starts ranging from wireless mobile communications to video and image processing [2]. However, an area efficient 2-parallel FIR filter uses booth multiplier, carry look-ahead adder, and a carry-look-ahead subtractor for the design of the filter. Booth multiplier only multiplies in two signed binary numbers in two's complement and had high performance, consume low power and it does not have weak regularity. Let's take for an 8-bit binary number, in which the number may be either positive or negative and will be shown in two's complement format, i.e. the value is from -128 to +127 [4]. Traditional hardware multiplication is presented similarly as multiplication is done by hand:

a. Partial products are computed,
b. shifted appropriately, and
c. Summed.

This booth algorithm can be increased if we reduced the number of the partial product (i.e. fewer bits) because output will wait only for few sum to perform [7]. Figure.1 represent the basic design of booth algorithm.

DOI : 10.5121/vlsic.2016.7602

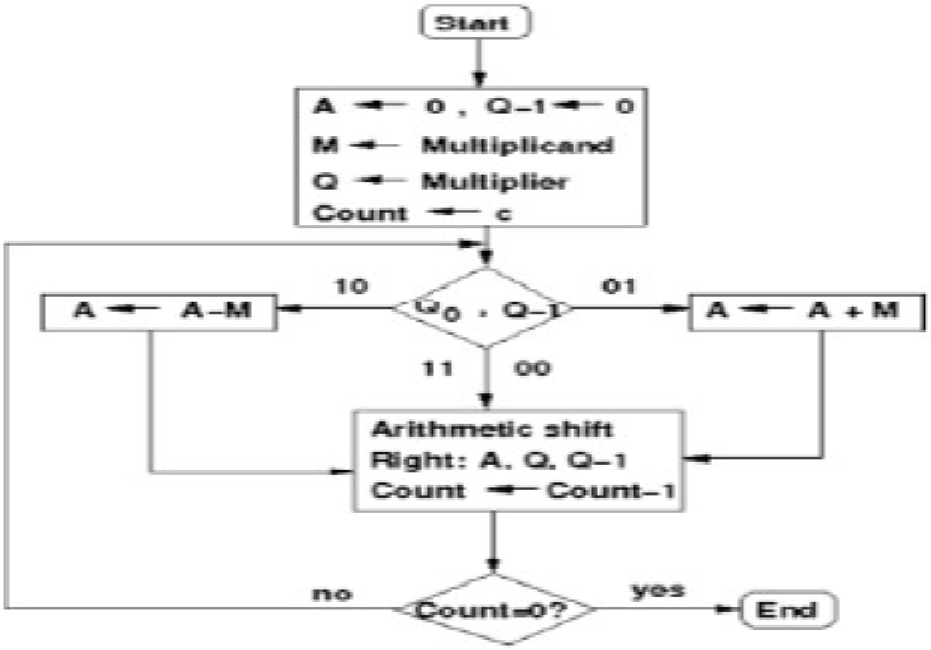


Figure. 1 Design flow of Booth Algorithm

Carry look-ahead adder is used in a digital logic circuit. The advantage of using carry lookahead adder is that it speeds up the bits and reduces area, and also it reduces the time required to examine the carry bits [3].Since adders are less weight in term of silicon area, compare to multipliers. Therefore multipliers are replaced with adders for reducing area and speed of the filter [5]. The carry look ahead subtractor is a fast subtractor which is designed to reduce the delay. If utilize the fact that, at each point of the bit position, whether it should carry with a generated at that bit or it can carry with a propagated at that bit.

In this paper, we are implementing the area-efficient 2-paralel FIR digital filter using VHDL. Integrated circuit (IC) which is designed in VLSI has become a drawback regarding area and speed. Our project is about improving the drawback which makes the area less that is storage resource of memory becomes small, and the speed of the operator becomes faster [6].

2. PARALLEL PROCESSING

Parallel processing and pipelining system are similar with one another. Independent sets of computations are computed and inserted in a pipelined technique, whereas a duplicate hardware computation is calculated and added in parallel processing [1].

To obtain a parallel processing system, we should convert the SISO (single- input- single -output) system into a MIMO (multiple-input-multiple-output) system. For example, the given below expression shows three inputs parallel system per clock cycle (i.e., a level of parallel processing L=3) [1].

y(3k) = ax(3k) + bx(3k-1) + cx(3k-2)

y(3k+1) = ax(3k+1) + bx(3k) + cx(3k-1)

y(3k+2) = ax(3k+2) + bx(3k+1) + cx(3k)

Where k represents the clock cycle. We know that at the kth clock cycle all the 3 inputs x (3k), x (3k+1) and x(3k+2) are further processed and 3 samples are generated at the output. Because of the MIMO system, a latch has been placed which is also known as block delay (or L-slow). For example, if we delay the signal x(3k) by 1 clock cycle it will give us output as x(3k-3) instead of x(3k-1), which has been input in the different input line. The block architecture for a 3-parallel FIR filter is shown in figure.2 [1].

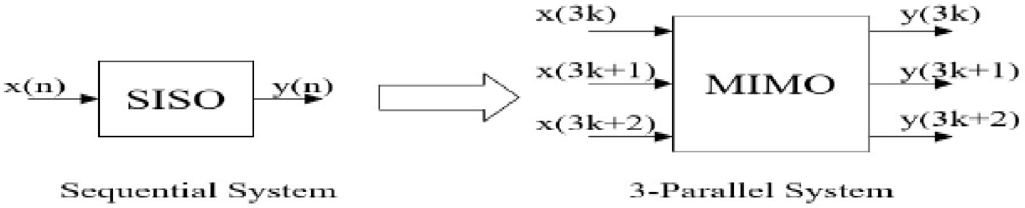


Figure. 2 A block processing example

2.1 Parallel processing FIR filters for High Speed or Low power

Consider a 2-parallel FIR digital filter shown in figure.3[1]. The 2-parallel FIR filter has exactly two copies of the primary 4-tap FIR filter. The dashed line in fig.3 indicates the critical path.16-bit Binary adder and the 16 bit binary multiplier are used for 2-parallel FIR filter designing. We consider the input x(2k) and x(2k+1) as even and odd respectively. Here h0,h1,h2,h3 indicates the filter coefficient of 2-parallel filter. D means delay. Delay of one clock cycle, which means the value, has to be stored for one clock cycle.

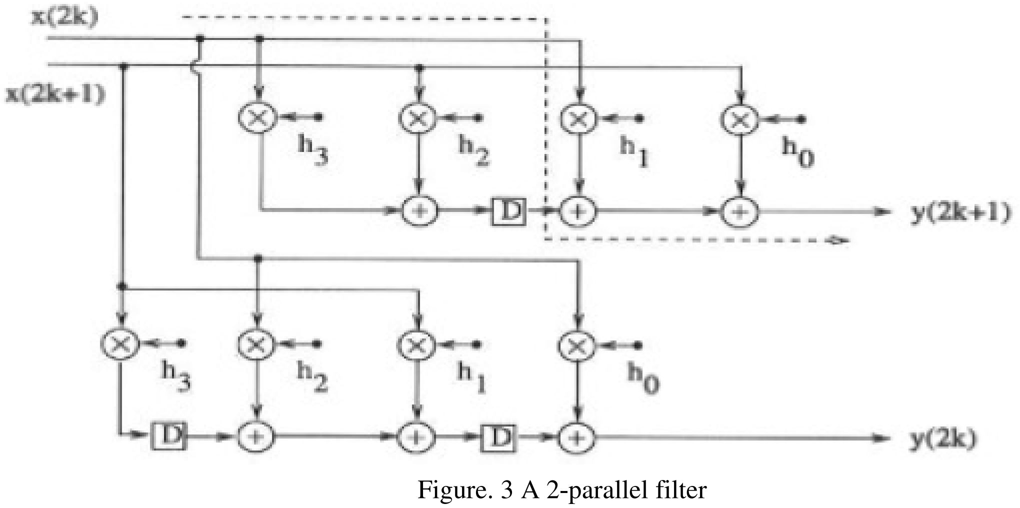


Figure. 3 A 2-parallel filter

From fig.2, the equation for 2-parallel FIR filter is expressed as

Y(2K) = h_0X(2K) + h_1X(2K-1) + h_2X(2K-2) + h_3X(2K-3)

$Y(2K+1) = h_0X(2K+1) + h_1x(2K) + h_2X(2K-1) + h_3X(2K-2)$

Similarly, consider the area efficient 2-parallel FIR filter in figure.4 [1]. The area efficient 2-parallel filter shown in figure.4 is more efficient in term of area and speed when compared with a basic 2-parallel FIR filter shown in figure.3. In figure.4, we consider the input x (2k) and x (2k+1) as even and odd respectively. D indicates the delay. Delay of one clock cycle, which means the value has to be stored for one clock cycle. The dashed line shows the critical line. h0, h1, h2, h3 are the filter coefficients.

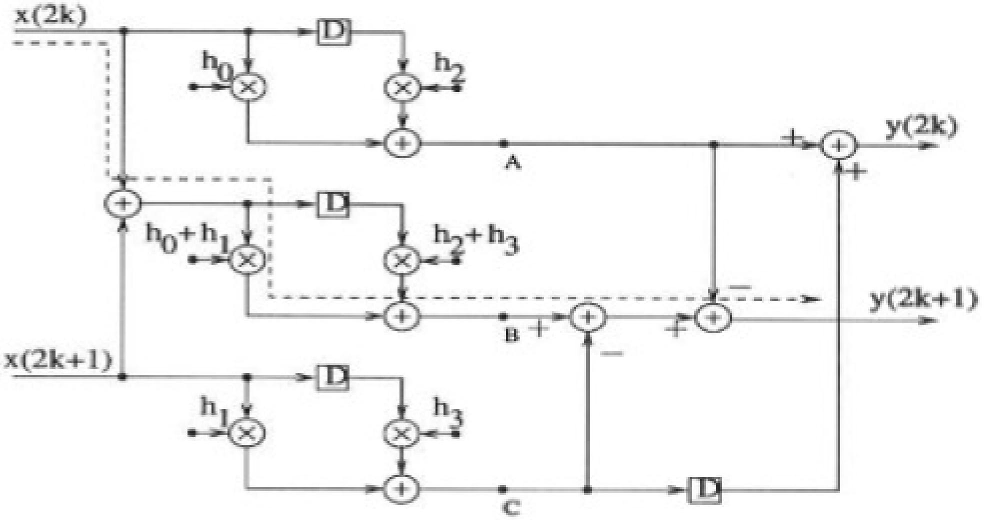


Figure. 4 An area efficient 2-parallel filter

The system equation for the given area efficient 2-parallel FIR filter is as follows:

$y(n) = h_0(n) + h_1x(n-1) + h_2x(n-2) + h_3x(n-3)$

Defining the outputs at node A, B and C as yA, yB and yC respectively from fig.5, we have

$y_A = h_0x(2k) + h_2x(2k-2)$

$y_B = (h_0+h_1) (x(2k) + x(2k + 1)) + (h_2+h_3) (x(2k-2) + x(2k-1))$

$y_c = h_1x(2k+1) + h_3x(2k-1)$

Then

$y(2k) = y_A +$ [y_C after 1 block delay]

$= h_0x(2k) + h_1x(2k-1) + h_2x(2k-2) + h_3x(2k-3)$

$y(2k+1) = y_B – y_A - y_C$

$= h_0x(2k+1) + h_1x(2k) + h_2x(2k-1) + h_3x(2k-2)$

3. SIMULATION AND RESULT

3.1 Results and Discussion

Clk_2 in figure. 6 and figure. 8 is an 8 bit up counter. From clk_2 we are generating x_even and x_odd signals.y1 [15:0] and y2 [15:0] in the figure. 6 and figure. 8 represents y (2k) and y (2k+1) signals. From synthesis report, it is found that memory usage for area efficient 2 parallel FIR filter is 301308 kb and 2-parallel FIR filter is 306920 kb and delay for area efficient is found to be 10.880ns and delay for 2-parallel is found to be 11.905ns. Thus, it is clear that the area efficient 2-parallel FIR filter has less area and more speed when compare with the existing 2-parallel FIR filter. Even the number of a slice, flip-flop, input LUTs are improved for area efficient 2-parallel FIR filter when compare to 2-parallel FIR filter.

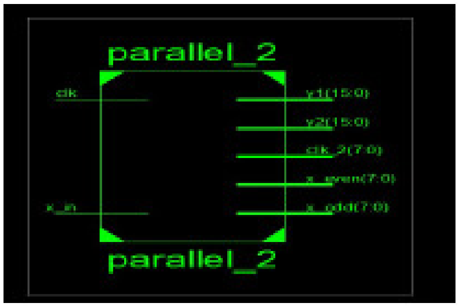


Figure. 5 RTL diagram for the 2-parallel FIR filter.

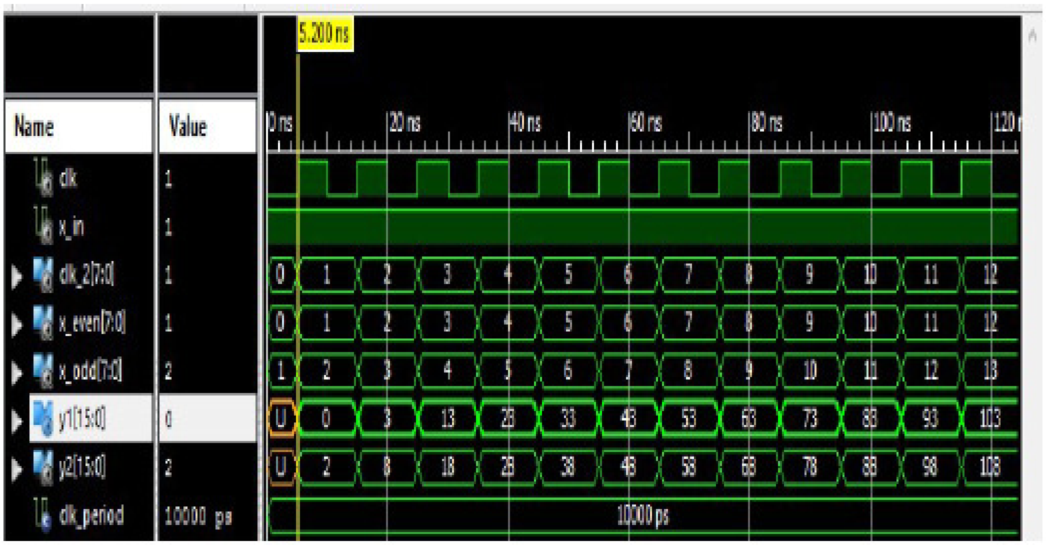
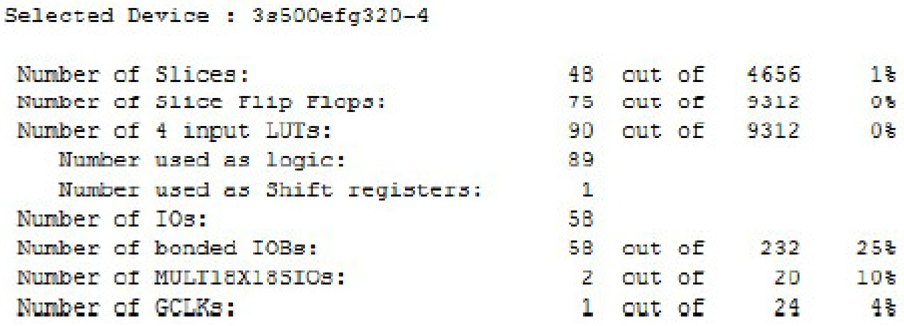


Figure.6 Simulation result for 2-parallel filter

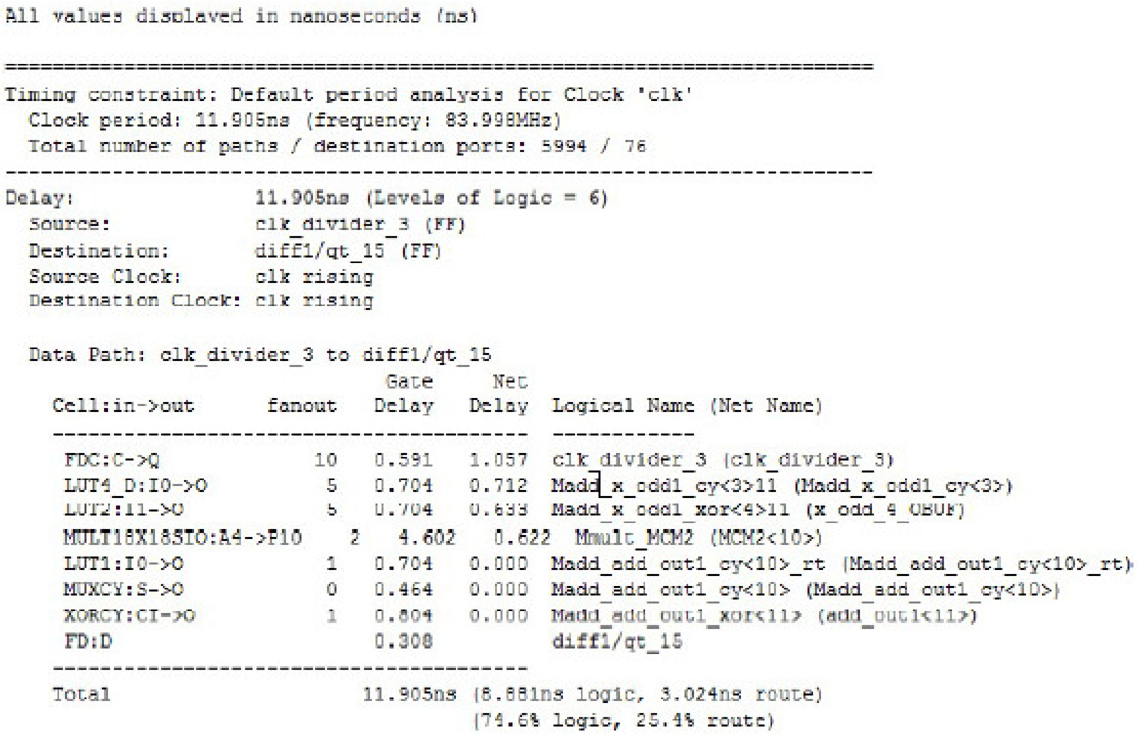
A. Device utilization summary for 2-parallel filter:



```
Selected Device : 3s500efg320-4

 Number of Slices:                        48  out of   4656     1%
 Number of Slice Flip Flops:              75  out of   9312     0%
 Number of 4 input LUTs:                  90  out of   9312     0%
    Number used as logic:                 89
    Number used as Shift registers:        1
 Number of IOs:                           58
 Number of bonded IOBs:                   58  out of    232    25%
 Number of MULT18X18SIOs:                  2  out of     20    10%
 Number of GCLKs:                          1  out of     24     4%
```

B. Timing Detail for 2-parallel:



```
All values displayed in nanoseconds (ns)

=========================================================================
Timing constraint: Default period analysis for Clock 'clk'
  Clock period: 11.905ns (frequency: 83.998MHz)
  Total number of paths / destination ports: 5994 / 76
-------------------------------------------------------------------------
Delay:               11.905ns (Levels of Logic = 6)
  Source:            clk_divider_3 (FF)
  Destination:       diff1/qt_15 (FF)
  Source Clock:      clk rising
  Destination Clock: clk rising

  Data Path: clk_divider_3 to diff1/qt_15
                                Gate     Net
    Cell:in->out      fanout   Delay   Delay  Logical Name (Net Name)
    ----------------------------------------  ------------
     FDC:C->Q             10   0.591   1.057  clk_divider_3 (clk_divider_3)
     LUT4_D:I0->O          5   0.704   0.712  Madd_x_odd1_cy<3>11 (Madd_x_odd1_cy<3>)
     LUT2:I1->O            5   0.704   0.633  Madd_x_odd1_xor<4>11 (x_odd_4_OBUF)
     MULT18X18SIO:A4->P10    2   4.602   0.622  Mmult_MCM2 (MCM2<10>)
     LUT1:I0->O            1   0.704   0.000  Madd_add_out1_cy<10>_rt (Madd_add_out1_cy<10>_rt)
     MUXCY:S->O            0   0.464   0.000  Madd_add_out1_cy<10> (Madd_add_out1_cy<10>)
     XORCY:CI->O           1   0.804   0.000  Madd_add_out1_xor<11> (add_out1<11>)
     FD:D                      0.308          diff1/qt_15
    ----------------------------------------
    Total                     11.905ns (8.881ns logic, 3.024ns route)
                                       (74.6% logic, 25.4% route)
```

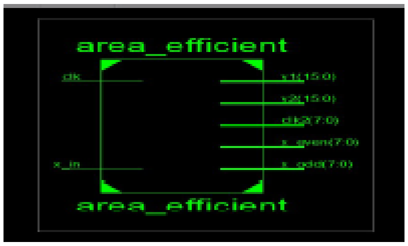


Figure. 7 RTL diagram for area-efficient 2-parallel FIR filter.

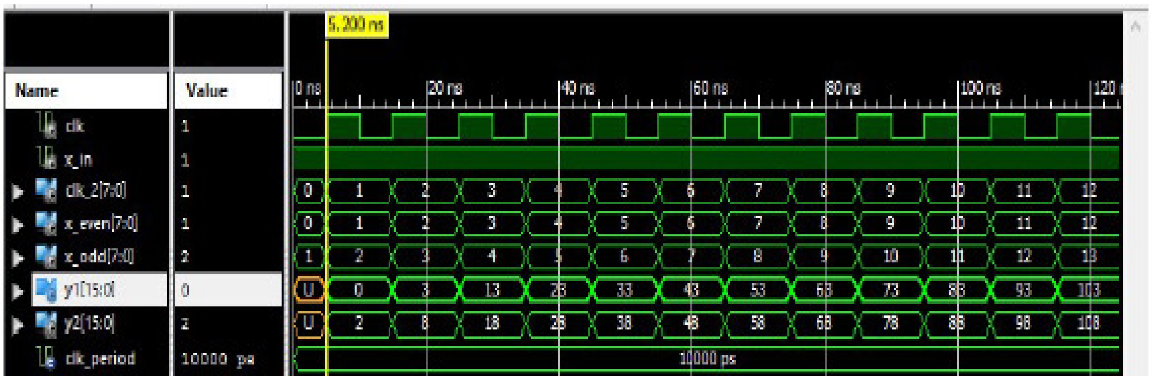
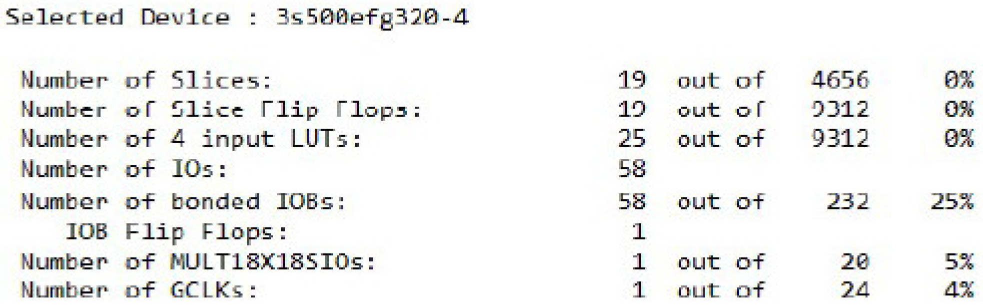


Figure. 8 An area efficient 2-parallel filter output waveform.

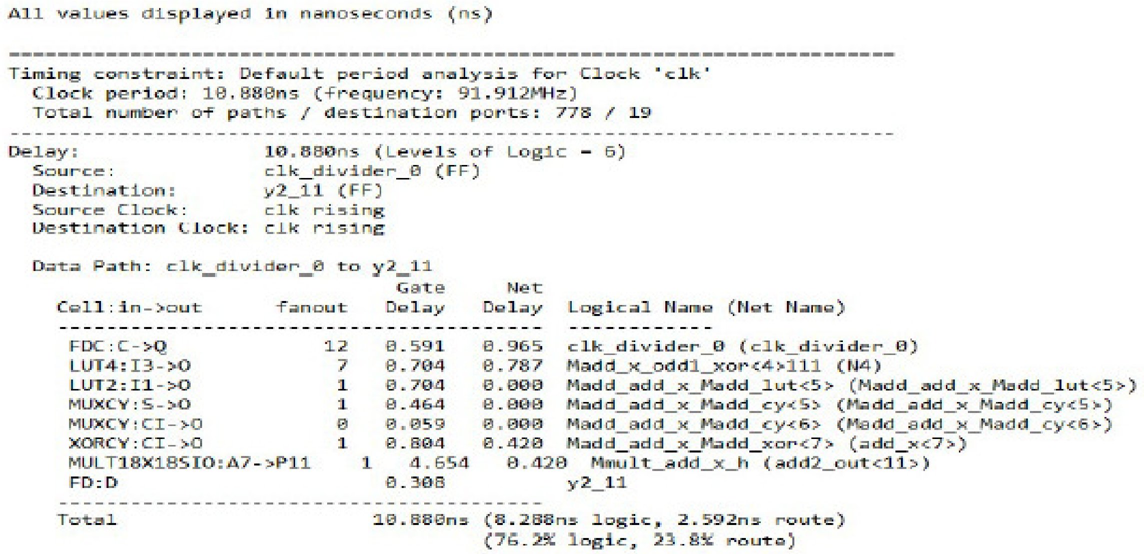
A. Device utilization summary of area efficient 2 parallel filter:



```
Selected Device : 3s500efg320-4

 Number of Slices:                      19  out of   4656     0%
 Number of Slice Flip Flops:            19  out of   9312     0%
 Number of 4 input LUTs:                25  out of   9312     0%
 Number of IOs:                         58
 Number of bonded IOBs:                 58  out of    232    25%
    IOB Flip Flops:                      1
 Number of MULT18X18SIOs:                1  out of     20     5%
 Number of GCLKs:                        1  out of     24     4%
```

B. Timing Details of area efficient 2 parallel filter:



```
All values displayed in nanoseconds (ns)

=========================================================================
Timing constraint: Default period analysis for Clock 'clk'
  Clock period: 10.880ns (frequency: 91.912MHz)
  Total number of paths / destination ports: 778 / 19
-------------------------------------------------------------------------
Delay:               10.880ns (Levels of Logic = 6)
  Source:            clk_divider_0 (FF)
  Destination:       y2_11 (FF)
  Source Clock:      clk rising
  Destination Clock: clk rising

  Data Path: clk_divider_0 to y2_11
                                Gate     Net
    Cell:in->out      fanout   Delay   Delay  Logical Name (Net Name)
    ----------------------------------------  ------------
     FDC:C->Q             12   0.591   0.965  clk_divider_0 (clk_divider_0)
     LUT4:I3->O            7   0.704   0.787  Madd_x_odd1_xor<4>111 (N4)
     LUT2:I1->O            1   0.704   0.000  Madd_add_x_Madd_lut<5> (Madd_add_x_Madd_lut<5>)
     MUXCY:S->O            1   0.464   0.000  Madd_add_x_Madd_cy<5> (Madd_add_x_Madd_cy<5>)
     MUXCY:CI->O           0   0.059   0.000  Madd_add_x_Madd_cy<6> (Madd_add_x_Madd_cy<6>)
     XORCY:CI->O           1   0.804   0.420  Madd_add_x_Madd_xor<7> (add_x<7>)
     MULT18X18SIO:A7->P11     1   4.654   0.420  Mmult_add_x_h (add2_out<11>)
     FD:D                      0.308          y2_11
    ----------------------------------------
    Total                     10.880ns (8.288ns logic, 2.592ns route)
                                       (76.2% logic, 23.8% route)
```

4. CONCLUSIONS

In this paper, area-efficient 2-parallel FIR filter is designed and compared with a primary 2-parallel filter. Area and speed of area-efficient 2-parallel filter are improved. Carry-look-ahead adder and subtractor are used in an area-efficient 2-parallel filter. For multiplication, booth multiplier is used in an area-efficient 2-parallel filter. All the simulated waveforms are discussed.

ACKNOWLEDGEMENTS

I would like to thank Manoj Kumar Assistant Professor in the Department of ECE, NIT Manipur for his support and valuable guidance.

REFERENCES

[1] K. K. Parhi, VLSI Digital Signal Processing Systems: Design and Implementation. New York: Wiley, 1999.

[2] R. Tamilarsi, “An Efficient Area and Low Power FIR Digital Filter Structure Implemented By Fast FIR Algorithm Utilizes the Symmetric Convolution” ISSN:2454-4116, Volume-2, Issue-4, April 2016 page 37-40.

[3] Amita and Mrs. Nitin Sachdeva, “Design and Analysis of Carry Look Ahead Adder Using CMOS Technique” (IOSR-JECE) e-ISSN, P-ISSN: 2278-8735. Volume 9, Issue 2, Ver.VII(Mar-Apr. 2014).

[4] V.V. Haibatpure, P.S. Kasliwal and B.P. Patil, “Performance Evaluation of Proposed Vedic Multiplier In Micro wind,” International Journal of Communication Engineering Applications, vol. 03, pp. 498-502, Jul. 2012.

[5] Yu-Chi Tsao and Ken Choi, “Area-Efficient Parallel FIR Digital Filter Structures for Symmetric Convolutions Based on Fast FIR Algorithm” IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 20, no. 2, pp. 366– 371, Feb. 2012.

[6] Lavina Magdalene Mary “Area Efficient Parallel Fir Digital Filter Structures Based On Fast Fir Algorithm” (IJERA) ISSN:2248-9622 Vol. 3, Issue 1, January -February 2013, pp.2042-2046.

[7] Neha Goyal, Khushboo Gupta, Renu Singla, “Study of Combinational and Booth Multiplier”International Journal of Scientific and Research Publications, Volume 4, Issue 5, May 2014.

AUTHORS

L kholee phimu was born in 1990. He received the B.E. degree in Electronic and Communication Engineering from Vinayaka Mission University, Salem in 2012. He is currently perusing his M.Tech Degree in VLSI and Embedded system in NIT Manipur, India.

Mr. Manoj Kumar is currently working as assistant professor in ECE department at NIT Manipur.