Design and Implementation of High speed counters using "MUX based Full Adder (MFA)"

K.V.Jyothi[1], G.P.S. Prashanti[2],
1Student, 2Assistant Professor
1,2,(Department of Electronics and Communication Engineering,
Gayatri Vidya Parishad College of Engineering For Women,Visakhapatnam, Andhra Pradesh, India)
Corresponding Author: K.V.Jyothi

Abstract: *In this brief, a new binary counter design is proposed. Counters are used to determine how many number of inputs are active (in the logic ONE state) for multi input circuits. In the existing systems 6:3 and 7:3 Counters are designed with full and half adders, parallel counters, stacking circuits. It uses 3-bit stacking and 6-bit stacking circuits which group all the "1" bits together and then stacks are converted into binary counts. This leads to increase in delay and area. To overcome this problem "MUX based Full adder (MFA)" 6:3 and 7:3 counters are proposed. The backend counter simulations are achieved by using MENTOR GRAPHICS in 130nm technology and frontend simulations are done by using XILINX. This MFA counter is faster than existing stacking counters and also consumesless area. Additionally, using this counters in Wallace tree multiplier architectures reduces latency for 64 and 128-bit multipliers.*

Keywords: *stacking circuits, parallel counters,High speed counters, MUX based full adder (MFA) counter,Mentor graphics,Xilinx,SPARTAN-6 FPGA.*

Date of Submission: 28-07-2019 Date of acceptance: 13-08-2019

I. Introduction

High speed, an efficient addition of multiple operands is an essential operation in any computational unit. With the ever-increasing applications in mobile communications and portable equipment, the demand for low-power and high speed VLSI systems is steadily increasing. Multiplier circuits are an essential part of an arithmetic logic unit (ALU), or a digital signal processor (DSP) system for performing filtering and convolution. Counters applications include in the realization of multipliers, computer arithmetic units, multiple input adders and associative processors. Counters are also used in digital neural networks. Counters play an important role in multipliers. The binary multiplication of integers or fixed-point numbers results in partial products that must be added to produce the final product. The addition of these partial products dominates the latency and power consumption of the multiplier. In order to combine the partial products efficiently, column compression is commonly used. Many methods have been presented to improving the performance of the partial product summation, such as the well-known row compression techniques in the Wallace tree [1] or Dadda tree [2], or the other improved architecture [3]. Through several layers of reduction, the number of summands is reduced to two, which are then added using a conventional adder circuit. To achieve higher efficiency, larger numbers of bits of equal weight can be considered. In the existing method i.e stacking uses 3- bit stacking circuits followed by a novel method of combining two small stacks to form larger stacks. While combining stacks delay and complexity of the circuit is increasing. To overcome this, a new method has been proposed named as "MUX Based Full Adder (MFA)" counter. These are faster than existing counter designs and also area is reduced. From this basis, a 3-2 and 4-2 counters are designed, which are the basic components in the partial product summation tree of a parallel array multiplier [5]. This MFA involve using MUX based full adders functioning as counters to reduce groups of 3 bits of the same weight to 2 bits of the different weight.

II. Literature Review

There are many binary counters existed as full adders and half adders that build from counters, parallel counters [7], multiplexer based counters[8], stacking circuits[6]. Compared with all of these, stacking counter has an improvement in delay. The 6:3 and 7:3 counters are realized by first stacking all of the input bits such that all of the "1" bits are grouped together. Small 3-bit stacking circuits are first used to form 3-bit stacks. These 3-bit stacks are then combined to make a 6-bit stack using a symmetric technique that adds one extra layer of logic[6].

2.1 Three-Bit Stacking Circuit:

Here inputs are X0, X1, and X2, a 3-bit stacker circuit will have three outputs Y0, Y1, and Y2 such that the number of "1" bits in the outputs is the same as the number of "1" bits in the inputs. First "1" bits occupies the left,next followed by "0". The outputs are formed by

Y0 = X0 + X1 + X2
Y1 = X0X1 + X0X2 + X1X2
Y2 = X0X1X2.

The first output will be "1" if any of the inputs is one, the second output will be "1" if any two of the inputs are one, and the last output will be one if all three of the inputs are "1." The Y1 output is a majority function and can be implemented using one complex CMOS gate. The 3-bit stacking circuit is shown in Fig. 1.

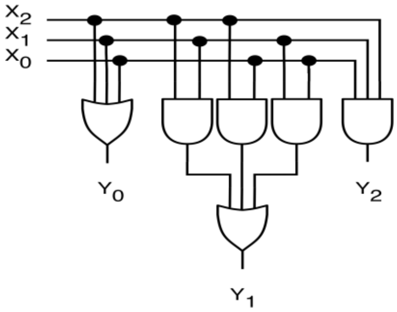


Fig. 1. Three-bit stacker circuit.

2.2 6:3 stacking counter:

Using this 3-bit stacking circuit, 6:3 and 7:3 stacking counters are formed. To form 6:3 and 7:3 counters, the 3-bit stacker circuits are to be merged. Then a bit stack is formed. This stack is then converted to be a binary number. That number indicates the binary representation of number of one's in the input.

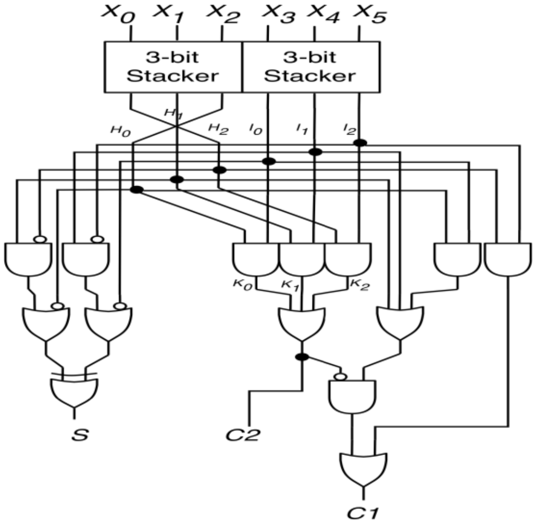


Fig. 2. A 6:3 counter based on symmetric stacking.

Here H and I vectors are the intermediate values. For a fast and more efficient count, an intermediate values can be used, whereH, I, and K to quickly compute each output bit without needing thebottom layer of stackers. Call the output bits C2, C1, and S in which C2, C1, S is the binary representation of the number of "1" input bits. To get sum,

He = H0 + H1H2
Ie = I0 + I1 I2.
S = He ⊕Ie.

To compute sum, the intermediate values H and I are related by ex-or operation. C1 and C2 are computed by,
C1 = (H1 + I1 + H0 I0)(K0 + K1' + K2) + H2 I2.
C2 =K0 + K1 + K2.
C2 as it should be set, whenever we have at least 4-bits.

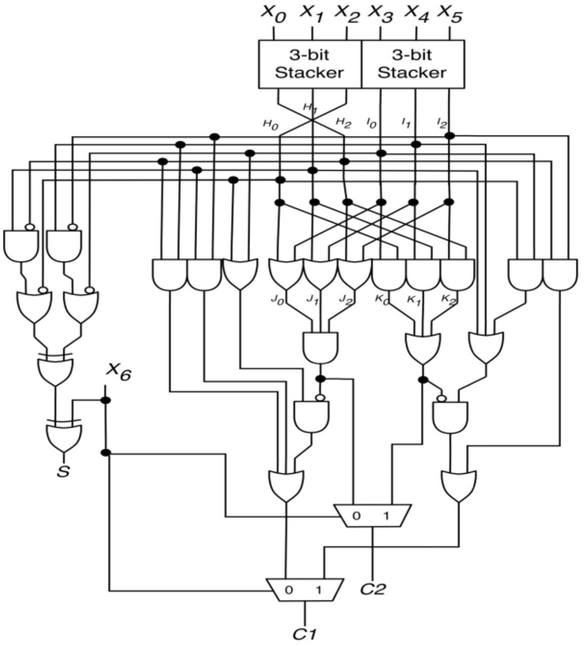
2.3 7:3 stacking counter:

The 7:3 counters are desirable as they provide a higher compression ratio. The design of the 7:3 counter involves computing outputs for C1 and C2 assuming both X6 = 0 and assuming X6 = 1.S output can be computed by adding one additional XOR gate. If X6 = 1, then C1 = 1 if the count of X0 to X5 is at least 1 but less than 3 or 5, which can be computed as

$C1 = (H0 + I0)J0\ J1'\ J2 + H2\ I1 + H1\ I2.$

Also, C2 = 1 if the count of X0 to X5 is at least 3

$C2 = J0\ J1\ J2.$



X_0 X_1 X_2 X_3 X_4 X_5
3-bit Stacker
3-bit Stacker
H_0 H_1 H_2 I_0 I_1 I_2
J_0 J_1 J_2 K_0 K_1 K_2
X_6
S
0 1
C2
0 1
C1

Fig. 3. A 7:3 counter based on symmetric stacking.

Both versions of C1 and C2 are computed and a mux is used to select the correct version based on X6. Note that this design therefore has multiplexers on the critical path. Here while concentrating on delay, the power consumption is increasing. To reduce the power the proposed counter i.e MFA counter is being introduced.

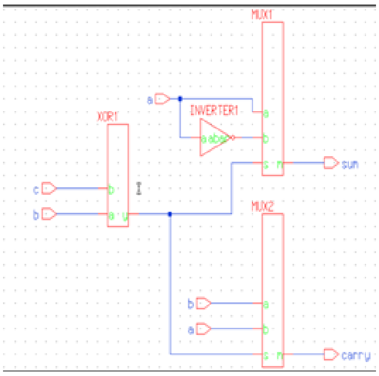
III. System Overview

Here, in this method a binary counter design is proposed which is designed with the MUX based Full adders. In order to reduce the power consumption and delay of the existing counter design this method is proposed.

3.1 MFA counter:

These methods involve using MUX based full adders functioning as counters to reduce groups of 3 bits of the same weight to 2 bits of the different weight.The proposed MFA counter is shown in Fig 4.Which consists of two 2:1 MUX and an XOR gate. In the proposed circuit, one XOR block in the conventional Full Adder is replaced by a multiplexer block so that the critical path delay is minimized. The critical path delay is given by,

Delay = XOR + MUX



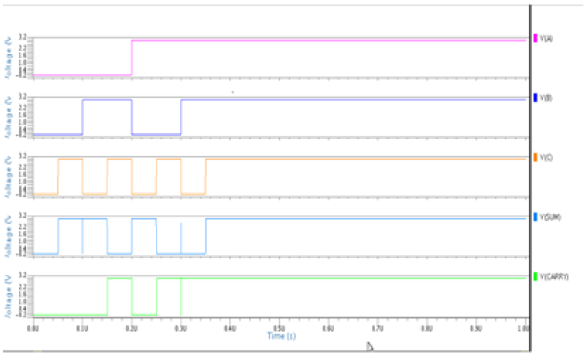


Fig 4. MUX based Full Adder **Fig 5.** MFA simulation results

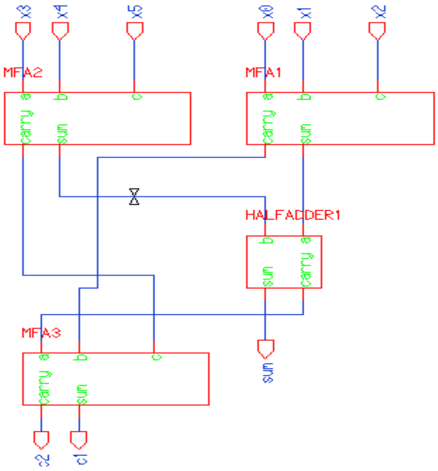
The proposed Full adder consists of three bits as inputs. Out of the three inputs, one input and its complement are provided as inputs to the first multiplexer. The other two inputs are given to the XOR gate, the output of which will act as a select line to both the multiplexers. The inputs of the second multiplexer are the bits other than the carry bit. This operation can be explained by

Y = B XOR C
Y = 0
Sum = A
Carry = B

Y = 1
Sum = A
Carry = A

3.2 6:3 MFA counter:

This 6:3 MFA counter is implemented by using three mux based full adders and one half adder. By using this method delay and power are reduced. The inputs are given as X0 to X5. The first three inputs X0, X1 and X2 undergo the XOR operation. The result of the XOR operation is denoted as"y0". The t0 is given as the select lines for the two multiplexers. If the value of t0 is 0 then the sum output value is X0, carry value is given by X1. If the value of t0 is 1 then for the sum output, the value is X0. The carry value is given by X0. Similarly, the next three inputs X4, X5 and X6 is given to the next MFA. The first two inputs undergo XOR operation and the sum and carry are determined. The sum output from the first MFA and second MFA are given to the half adders and the operation is performed which gives the sum and carry output.



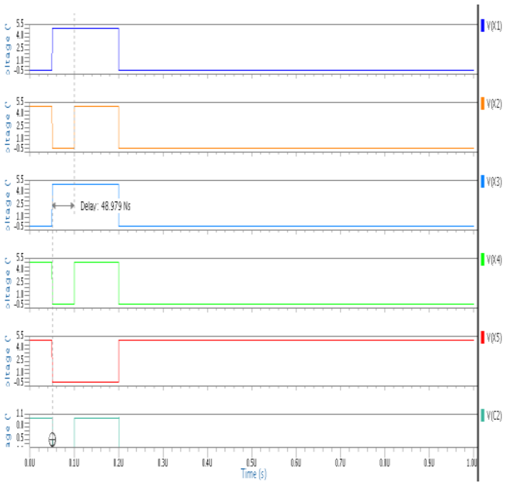


Fig 6. MFA 6:3 Counter schematic **Fig 7**. MFA 6:3 Counter delay

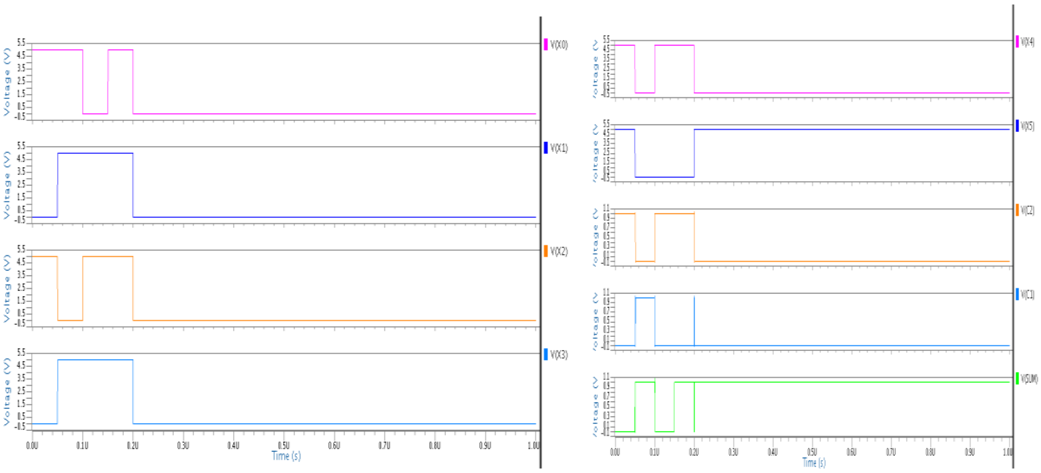
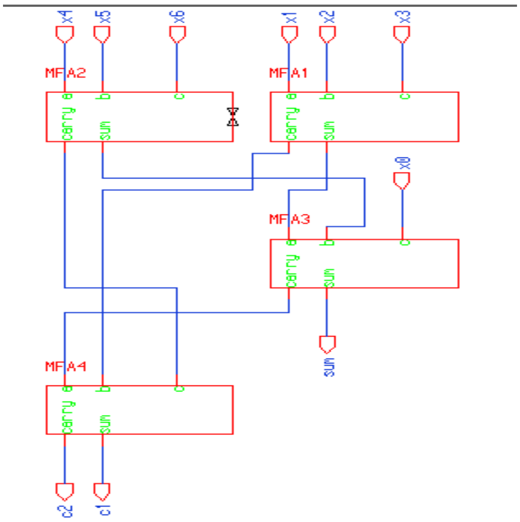


Fig 8. 6:3 MFA simulation results

For the third MFA all the carry outputs are given as inputs. The sum, carry1 and carry2 are generated at the third MFA.

3.3 7:3 MFA counter:

This 7:3 MFA counter is implemented by using four mux based full adders. The inputs to the counter are given as X0 to X6. The inputs X1, X2 and X3 are given to the first MFA. The X2 and X3 undergo XOR operation. The output of the XOR operation is denoted as "t0". The "t0" is given to the select lines to the multiplexer. If "t0" is 0 then sum is X1 and carry is X2. If "t0" is 1 then sum is X1 and carry is X1. Similarly, for all the Full adders the input is given and the output is obtained using the select lines. The fourth MFA gives the sum, carry1 and carry2 outputs, Which results the binary count of the given input.



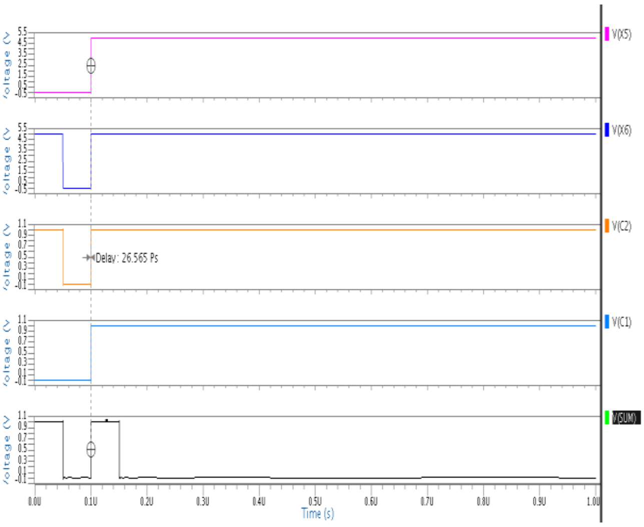


Fig 9. MFA 7:3 Counter schematic **Fig 10.** MFA 7:3 Counter delay

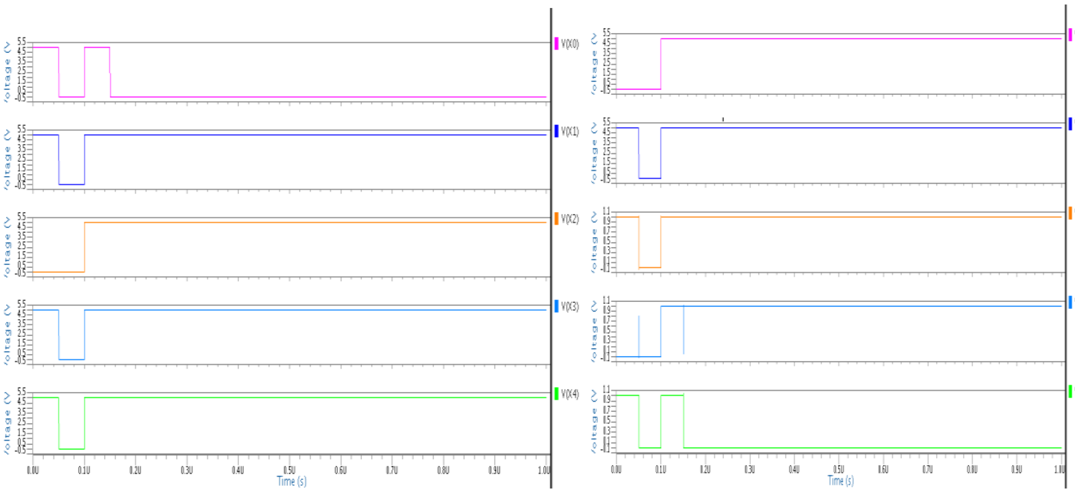


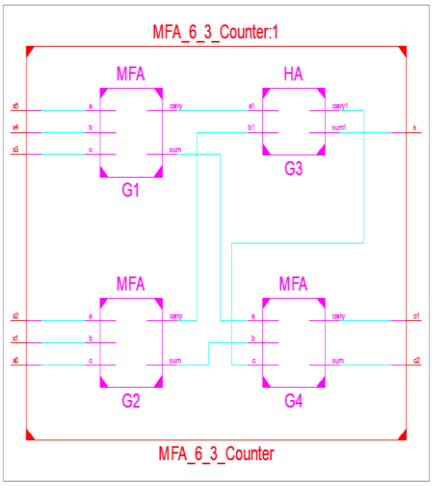
Fig 11. 7:3 MFA simulation results

IV. Simulation Results

Delay Comparision table

Technology	6:3 DELAY(NS)	7:3 DELAY(PS)
Full adder[6]	149.26	208.2
Stacking counter[6]	100.17	99.1
MFA counter	48.91	26.5

Xilinx results:
6:3 MFA counter:



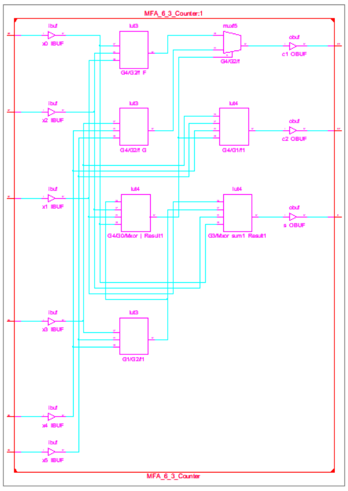
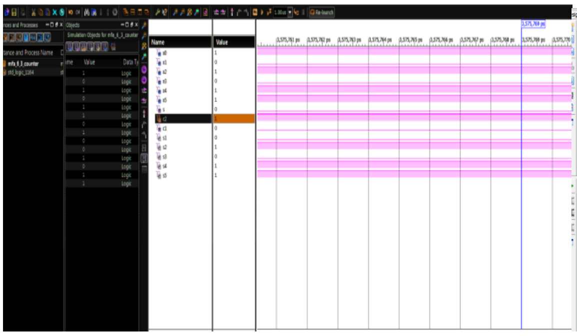
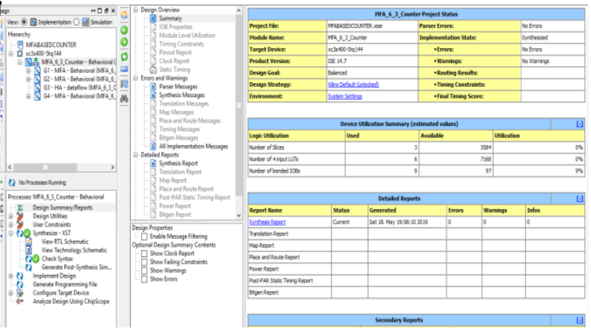


Fig 12.RTL schematic **Fig 13.**Technology view





MFA_6_3_Counter Project Status			
Project File:	MFABASEDCOUNTER.xise	Parser Errors:	No Errors
Module Name:	MFA_6_3_Counter	Implementation State:	Synthesized
Target Device:	xc3s400-5tq144	•Errors:	No Errors
Product Version:	ISE 14.7	•Warnings:	No Warnings
Design Goal:	Balanced	•Routing Results:	
Design Strategy:	Xilinx Default (unlocked)	•Timing Constraints:	
Environment:	System Settings	•Final Timing Score:	

Device Utilization Summary (estimated values)			
Logic Utilization	Used	Available	Utilization
Number of Slices	3	3584	0%
Number of 4 input LUTs	6	7168	0%
Number of bonded IOBs	9	97	9%

Detailed Reports					
Report Name	Status	Generated	Errors	Warnings	Infos
Synthesis Report	Current	Sat 18. May 19:58:10 2019	0	0	0
Translation Report					
Map Report					
Place and Route Report					
Power Report					
Post-PAR Static Timing Report					
Bitgen Report					

Secondary Reports

Fig 14. Timing diagram **Fig 15.** Summary report

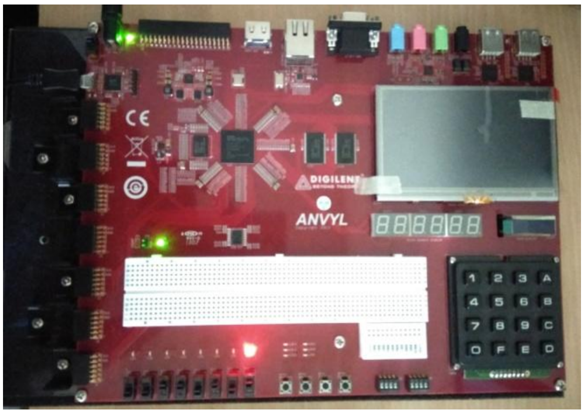
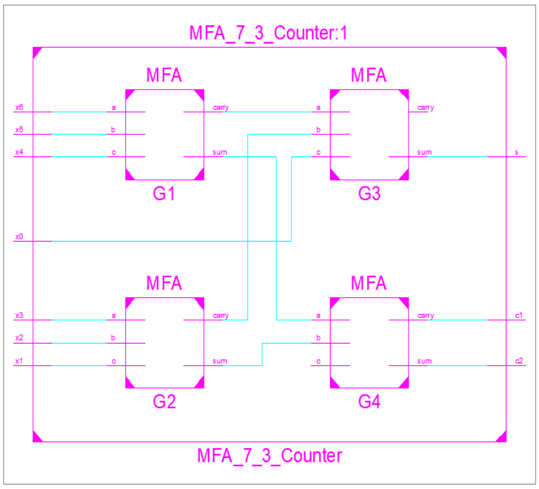


Fig 16. FPGA result

7:3 MFA counter:



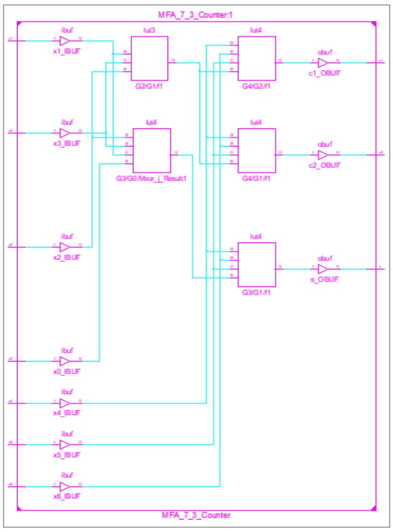
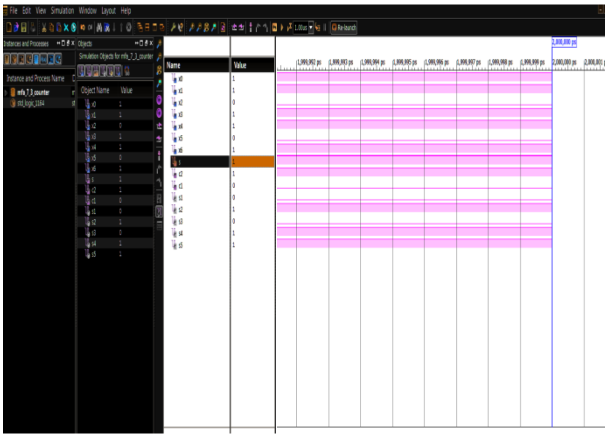
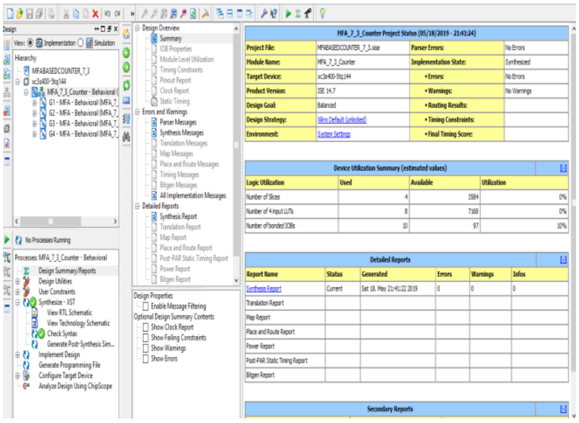


Fig 17.RTL schematic **Fig 18.**Technology view





MFA_7_3_Counter Project Status (05/18/2019 - 21:41:24)			
Project File:	MFABASEDCOUNTER_7_3.xise	**Parser Errors:**	No Errors
Module Name:	MFA_7_3_Counter	**Implementation State:**	Synthesized
Target Device:	xc3s400-5tq144	• **Errors:**	No Errors
Product Version:	ISE 14.7	• **Warnings:**	No Warnings
Design Goal:	Balanced	• **Routing Results:**	
Design Strategy:	Xilinx Default (unlocked)	• **Timing Constraints:**	
Environment:	System Settings	• **Final Timing Score:**	

Device Utilization Summary (estimated values)

Logic Utilization	Used	Available	Utilization
Number of Slices	4	3584	0%
Number of 4 input LUTs	8	7168	0%
Number of bonded IOBs	10	97	10%

Detailed Reports

Report Name	Status	Generated	Errors	Warnings	Infos
Synthesis Report	Current	Sat 18. May 21:41:22 2019	0	0	0
Translation Report					
Map Report					
Place and Route Report					
Power Report					
Post-PAR Static Timing Report					
Bitgen Report					

Secondary Reports

Fig 19. Xilinx results **Fig 20.** Summary report

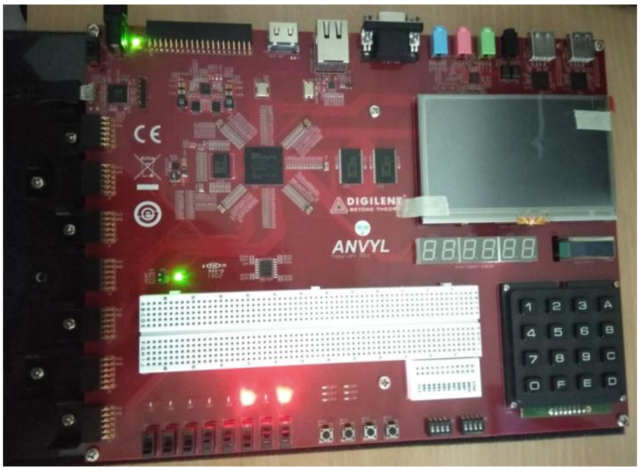


Fig 21. FPGA result

V. Conclusion and Future scope

In this brief, a new binary MFA counter approach is proposed. This counting method can be used to implement 6:3 and 7:3 counters, which can be used in any binary multiplier circuit to add the partial products. 6:3 counters implemented with this mux based technique achieve higher speed than other higher order counter designs while reducing power consumption and area. This proposed model can also be used to implement Wallace tree i.e., the 64-bit and 128-bit counter based Wallace tree multiplier can be built using the proposed 6:3 counter and 7:3 counter which helps to reduce latency and increase speed. Further, these counter based techniques used by multipliers, help to increase the speed of multipliers and in Wallace tree there is presence of half adders and full adders these proposed counters helps to reduce the delay.

References

[1]. C. S. Wallace, "A suggestion for a fast multiplier," IEEE Trans. Electron. Comput., vol. EC-13, no. 1, pp. 14–17, Feb. 1964.

[2]. L. Dadda, "Some schemes for parallel multipliers," Alta Freq., vol. 34, pp. 349–356, May 1965.

[3]. Z. Wang, G. A. Jullien, and W. C. Miller, "A new design technique for column compression multipliers," IEEE Trans. Comput., vol. 44, no. 8, pp. 962–970, Aug. 1995.

[4]. Swartzlander, E E., Jr., (1973) "Parallel Counters" IEEE Trans. Computers, Vol. 22, pp 1021-1024.

[5]. S F Hsiao, M R Jiang and J S Yeh (1998) "Design of high speed low power 3-2 counter and 4-2 compressor for fast multipliers"Electron Lett., Vol.34, no. 4, pp.341-343.

[6]. Christopher Fritz, Adly T Fam (2017) "Fast binary counters based on symmetric stacking" IEEE Trans VLSI Systems., pp. 1-5.

[7]. M. Mehta, V. Parmar, and E. Swartzlander, "High-speed multiplier design using multi-input counter and compressor circuits," in Proc. 10th IEEE Symp. Comput. Arithmetic, Jun. 1991, pp. 43–50.

[8]. S. Veeramachaneni, L. Avinash, M. Krishna, and M. B. Srinivas, "Novel architectures for efficient (m, n) parallel counters," in Proc. 17th ACM Great Lakes Symp. VLSI, 2007, pp. 188–191.

K.V.Jyothi" Design and Implementation of High speed counters using "MUX based Full Adder (MFA)". IOSR Journal of Electronics and Communication Engineering (IOSR-JECE) 14.4 (2019): 57-64.