IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 5, Issue 5, May 2016

Design and Implementation of a novel FIR Filter based on MCM for improved Power and Delay Efficiency

Puja Sutar[1], Ramya S.K[2]

M. Tech Student, Department of Electronics and Communication Engineering, Alpha College of Engineering, Bangalore, Karnataka, India[2]

Assistant Professor, Department of Electronics and Communication Engineering, Alpha College of Engineering, Bangalore, Karnataka, India[1]

Abstract: This paper proposes about new FIR filters that are implemented using multiple constant multiplications (MCM). This technique helps in reducing delay and power efficiency. It also helps to reduce area to some extent. All symmetric computations are performed by multipliers as the multipliers functions in symmetric manner. Multiple constant multiplications make use of two techniques common sub-expression (CSE) algorithm and GB algorithm technique to implement the multipliers. In this proposed system we have reduced down the use of number of adders, subtracters, shifters etc. to minimum and these are replaced by use of multipliers for increasing the efficiency of the filters. This paper also describes that the delay and power is reduced by replacing the conventional multipliers by multiple constant multiplications multipliers. Simulation is done by using Xilinx ISE tool suite 14.6 tool. The results obtained in this project are considered with respect to 8-bit inputs.

Keywords: Multiple constant multiplications (MCM), Common sub-expression algorithm (CSE), GB algorithm**.**

I. INTRODUCTION

Finite impulse response (FIR) filter are widely used in digital signal processing (DSP) systems. Their specifications in terms of in linear-phase and feed-forward implementations are used for stable high performance filters. The direct-form and transposed-form of FIR filter designs are present. Both the architectures have similar hardware complexity; the transposed form is preferred mostly because of its power efficiency and higher performance. The transposed form is available for the multiplier block of FIR filter, where the multiplication of filter coefficients with the filter input is realized, and has significant impact on the complexity and performance of the design because a large number of constant multiplications required.

This is called as the multiple constant multiplications (MCM). It is a performance bottle neck a central operation in many other DSP systems.

II. RELATED WORK

The direct and transposed-form of FIR filter implementations are illustrated in the following figure (a) and (b). Both the architectures have similar hardware complexity, but the transposed form is mostly preferred because of its higher performance and power efficiency.

The transposed form of FIR filter is available where the multiplication of filter terms with the filter input is carried out, and has great impact on the and performance complexity because of a large number of constant multiplications are required for the design.

This is known as the multiple constant multiplications (MCM).

It is a performance bottleneck and central operation in many of the DSP systems.

However, the digit-based recoding technique doesn't exploit the exchanging of the common partial products that allows greater decrease in the number of functions and obviously, in area and power dissipation of the MCM design.

Hence, MCM problem is described to find the less number of arithmetic functions that implements the constant multiplications.

The algorithms that are designed for the MCM problem are divided in two classes: common sub expression elimination (CSE) algorithms and graph-based (GB) techniques. First of all the CSE algorithm draws all the possibilities of sub-expression that are defined in binary.

Then a suitable sub-expression is used among constant multiplication values.

GB methods are unlimited to any one particular number presentation that provides good results compared to CSE algorithms.

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 5, Issue 5, May 2016

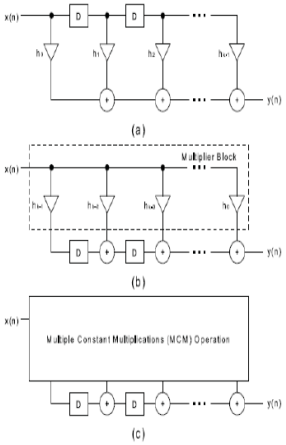

Fig.1 FIR filters (a) Implementation in Direct form. (b) Implementation in Transposed form with generic multipliers. (c) Implementation in Transposed form with an MCM block.

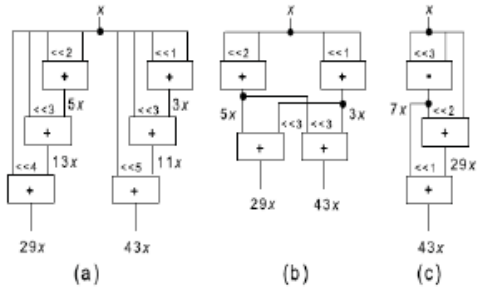

Fig.2 29x and 43x Shift-adds implementations (a) without partial product sharing and with partial product sharing. (b) Exact CSE algorithm. (c) Exact GB algorithm

III. PROPOSED SYSTEM

In the proposed system we have implemented carry select adder and carry lookahead adder using GB algorithm which is one of the technique of multiple constant multiplication (MCM). With this technique we have successfully shown that there is much reduction in delay, power and area as compared to use of conventional adders. Also compared to carry select adder, carry lookahead adder provides much reduction in delay, power and area.

1. Modified Carry Lookahead Adder

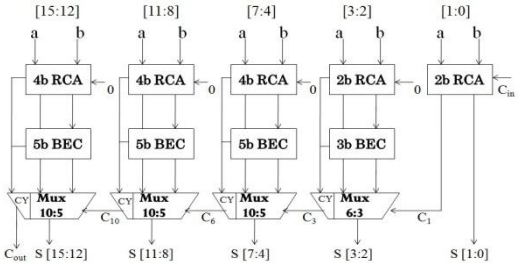

Fig.3 16-Bit Modified Carry Select Adder Schematic

The structure of carry select adder using binary to excess 1 converter for RCA with Cin=1 to minimize the area and power is shown in figure. In our proposed method the carry 1 RCA is replaced by the BEC. The n-bit RCA is replaced by the n+1bit BEC. The number of gate used in BEC is less compare with RCA.

A. Area calculation for MCSLA

The area calculation of modified CSLA is derived from the following steps. From the structure of MCSLA, 8-bit, 16-bit, 32-bit and 64-bit area is calculated.

The Group 1 architecture calculation is,

Gate Count	=	19	(FA+HA)
FA	=	13	(1×13)
HA	=	6	(1×6)

The Group 2 architecture calculation is,

Gate Count	=	43	(FA + HA + Mux + BEC)
FA	=	13	(1×13)
HA	=	6	(1×6)
Mux	=	12	(3×4)
BEC:			
AND	=	1	
NOT	=	1	
XOR	=	10	(2×5)

The Group 3 architecture calculation is,

Gate Count	=	61	(FA + HA + Mux + BEC)
FA	=	26	(2×13)
HA	=	6	(1×6)
Mux	=	16	(4×4)
BEC:			
AND	=	2	
NOT	=	1	
XOR	=	15	(3×5)

The Group 4 architecture calculation is,

Gate Count	=	84	(FA + HA + Mux + BEC)
FA	=	13	(3×13)
HA	=	6	(1×6)
Mux	=	20	(5×4)
BEC	=	24	

The Group 5 architecture calculation is,

Gate Count	=	107	(FA + HA + Mux + BEC)
FA	=	52	(4×13)
HA	=	6	(1×6)
Mux	=	24	(6×4)
BEC	=	30	

2. Modified Carry Lookahead Adder

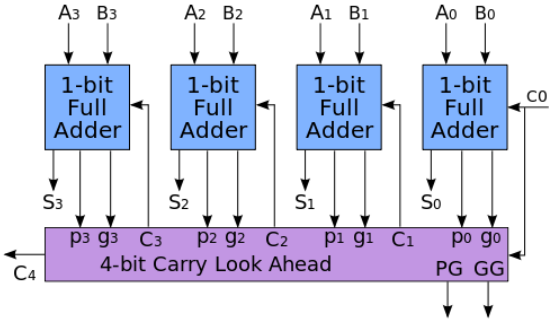

Fig.4 Modified Carry Lookahead Adder

DOI 10.17148/IJARCCE.2016.55201

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 5, Issue 5, May 2016

The method used in modified carry lookahead adder is propagating and generating carry. Addition of inputs A and B is said to generate if there is carry on addition. For binary addition, if both A and B values are 1 then carry is generated.

$G(A, B) = A * B$

Addition of inputs A and B is said to propagate, if the addition results in carry when there is input carry. For binary addition, if any one of the input value A or B is 1, carry is propagated.

$P(A, B) = A + B$

The logic values for generating G and propagating P are as follows for a carry lookahead adder combining ripple carry adder.

$C_1 = G_0 + P_0 * C_0$
$C_2 = G_1 + P_1 * C_1$
$C_3 = G_2 + P_2 * C_2$
$C_4 = G_3 + P_3 * C_3$

Putting C_1 value into C_2, C_2 value into C_3, C_3 value into C_4 we get,

$C_1 = G_0 + P_0 * C_0$
$C_2 = G_1 + G_0 * P_1 + * C_0 * P_0 * P_1$
$C_3 = G_2 + G_1 * P_2 + G_0 * P_1 * P_2 + C_0 * P_0 * P_1 * P_2$
$C_4 = G_3 + G_2 * P_3 + G_1 * P_2 * P_3 + G_0 * P_1 * P_2 * P^3 + C_0 * P_0 * P_1 * P_2$

Logic for generating a bit pair carry:

$G_i = A_i * B_i$

Logic statements for propagating a bit pair carry:

$P_i = A_i \wedge B_i$
$P_i = A_i + B_i$

IV. SIMULATION RESULTS

The simulation results for carry select adder and carry lookahead adder using GB technique are as shown below:

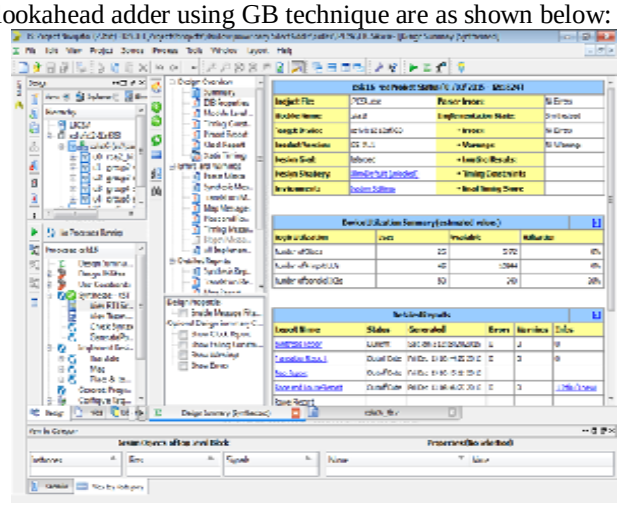

Fig.5 Design Summary

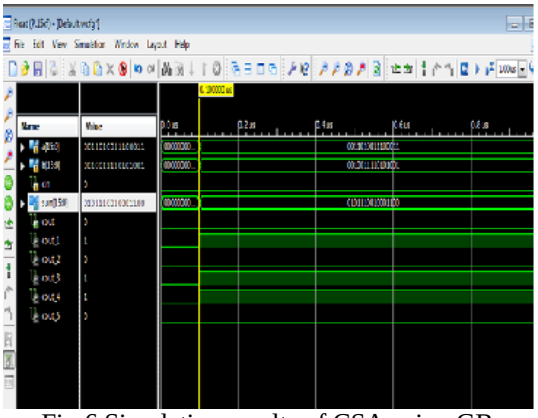

Fig.6 Simulation results of CSA using GB

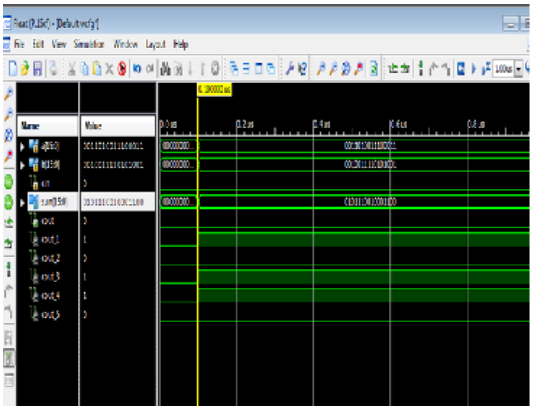

Fig.7 Simulation results of CLA using GB

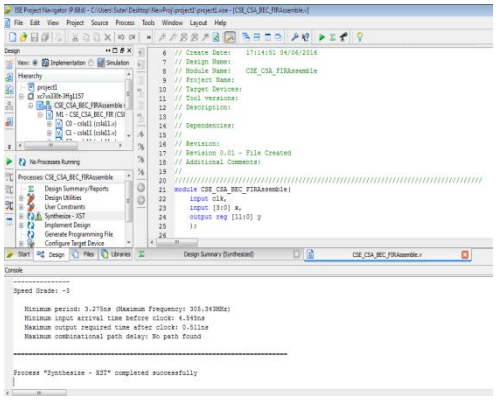

Fig.8 Delay analysis of CSA using GB

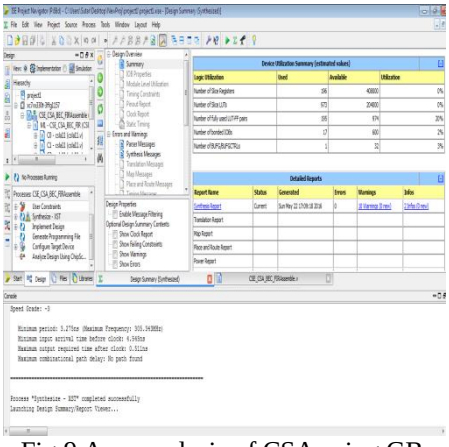

Fig.9 Area analysis of CSA using GB

DOI 10.17148/IJARCCE.2016.55201

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 5, Issue 5, May 2016

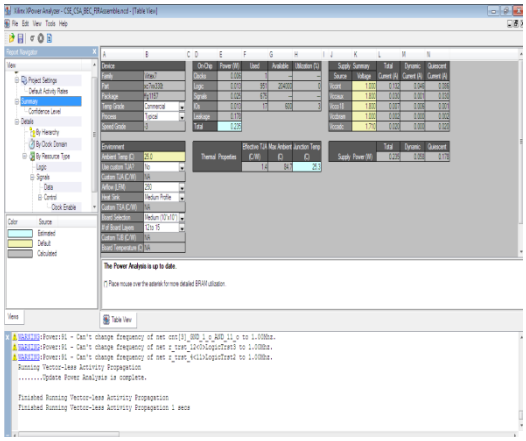

Fig.10 Power analysis of CSA using GB

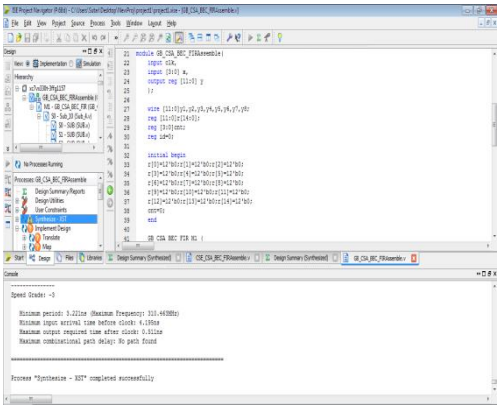

Fig.11 Delay analysis of CLA using GB

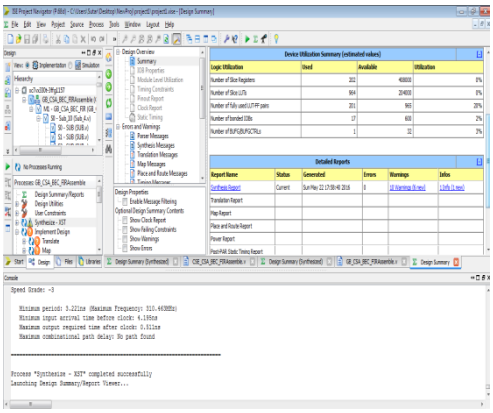

Fig.12 Area analysis of CLA using GB

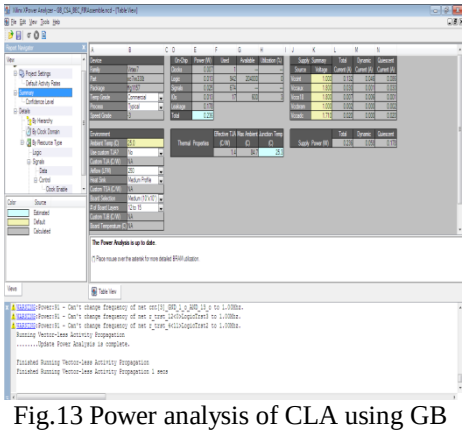

Fig.13 Power analysis of CLA using GB

V. CONCLUSION AND FUTURE WORK

FIR filters are of very much important in DSP systems. They are mostly used in implementation of any VLSI circuits. In this project we have made use of the GB algorithm method along with CSA and CLA which uses both RCA and BEC for giving better results. The results obtained so far shows that there is much more decrease in the power utilization, decrease in the delay and also utilization of area is also reduced to some extent. In future, the complexity of various applications can be reduced to its minimum level by making use of MCM technique in FIR filters. Though multipliers are the most expensive operators in the filters but MCM technique helps to reduce this complexity to a large extent. This work is presently restricted to the simulation only. Further enhancement of implementation on hardware would add more impact.

REFERENCES

[1]. Ram Kumar .B and Kittur H.M, "Low-Power and Area-Efficient Carry Select Adder", IEEE transactions on very large scale integration (VLSI) systems, vol. 20, no. 2, February 2012.

[2]. AnithaKumari R D, Nayana N D, "Low power and Area Efficient Carry Select Adder", National Conference on Electronics, Communication and Signal Processing, NCECS-2011.

[3]. Padma Devi, AshimaGirdher and Balwinder Singh, "Improved Carry Select Adder with Reduced Area and Low Power Consumption", International Journal of Computer Applications (0975 – 8887), Volume 3 -No.4, June 2010.

[4]. Ceiang T Y and Hsiao M J, "Carry-select adder using single ripple carry adder," Electron. Lett., vol. 34, no. 22, pp. 2101–2103, Oct. 1998.

[5]. Jeong .W and Roy .K, "Robust high-performance low power adder", Proc. of the Asia and South Pacific Design Automation Conference, pp. 503-506, 2003.

[6]. He Y, Chang C H, and Gu J, "An area efficient 64-bit square root carry select adder for low power applications," in Proc. IEEE Int. Symp. Circuits Syst., 2005, vol. 4, pp. 4082–4085.

[7]. Hosseinghadiry M, Mohammadi H and Nadisenejani M, " Two New Low Power High Performance Full Adders with Minimum Gates", World Academy of Science, Engineering and Technology 52 2009.

[8]. Kim Y and Kim L S, "64-bit carry-select adder with reduced area", Electron.Lett., vol. 37, no. 10, pp. 614–615, May 2001.

[9]. Manoj Kumar, Sandeep K. Arya and SujataPandey, "Single bit full adder design using 8 transistors with novel 3 transistors XNOR gate", International Journal of VLSI design & Communication Systems (VLSICS) Vol.2, No.4, December 2011.

[10]. KeivanNavi and NedaKhandel, "The Design of a High-Performance Full Adder Cell by Combining Common Digital Gates and Majority Function", European Journal of Scientific Research, ISSN 1450-216X Vol.23 No.4 (2008), pp.626-638.