International Journal of Engineering Research-Online
A Peer Reviewed International Journal
Articles available online http://www.ijoer.in

Vol.3., Issue.6., 2015
(Nov.-Dec.,)

RESEARCH ARTICLE

ISSN: 2321-7758

ARCHITECTURE OF PREFIX ADDER FOR IMPLEMENTATION OF FIR FILTER AND VERIFICATION USING UVM

G.DEVI SUDHARANI[1], P. PUSHPALATHA[2], M.RAMKUMAR[3]

[1] M.Tech, VLSI Design, [2] Assistant Professor,

[2]Dept. of ECE, University College of Engineering, University College of Engineering, JNTUK, Kakinada, India

[3]Seer Akademi, Hyderabad

ABSTRACT

This paper presents the design of low power and area efficient Finite Impulse Response (FIR) filter implemented using Wallace Tree Multiplier and Sparse Tree Adder (SPA) as its main components. The parallel-prefix Sparse Tree Adder uses less power with significantly reduced complexity due to the reduced interconnections when compared to the Kogge-Stone Adder (KGA). The Wallace Tree Multiplier uses compressor technology because it reduces the number of adder circuits and the number of levels required for the computation of the multiplier result and the Sparse Tree Adder is used to perform the addition in the final stage. The design is verified using Universal Verification Methodology (UVM) which is the most advanced verification technology The modules are coded with Verilog-HDL, verified using UVM and the simulation and the synthesis are performed using VCS Synopsys Tool and Synopsys Design Compiler.

Keywords: FIR, SPA, KGA, Wallace Tree Multiplier, Compressor, UVM

INTRODUCTION

Filters are widely used in multimedia and Digital Signal Processing. Most of the computations involved in the realization of Digital Signal Processors (DSP) uses multiply and accumulate operations so there is a need for efficient multiplier and adder units which utilizes less power and area. The serial adders compute the present carry based on the result given by the previous unit. Therefore serial adders take a lot of time to perform the addition. Parallel adder uses a tree like structure to compute the individual carry thereby increasing the speed of the computation. The compressor takes more number of input bits at once and produces the sum and carry thereby reducing the need for large number of full adder and half adder circuits. The digital filters are generally linear time invariant filters. The best way of implementing linear time invariant Finite Impulse Response filter is convolution of input series with impulse response filter coefficients and is given by

$y[n] = x[n] f[n]$ ……(1)

$y[n] = b0\ x[n] + b1\ x[n-1] + \ldots + bN\ x[n-N]$ ……(2)

In the above equation x[n] represents the input signal, y[n] represents the output signal, N is the order of the filter and bn represents the filter coefficient.

The above equation expressed in Z domain is given by

$y(z) = x(z)\ B(z)$ ……(3)

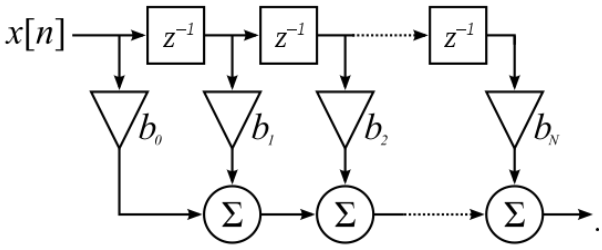

Figure 1: Direct form N-Tap FIR Filter

PARALLEL PREFIX ADDER

Parallel prefix adders offer a good basis to provide a wide range of design trade-offs between area, power and delay. Carry look ahead adder is considered as a base to design Parallel prefix adders. Parallel prefix adder architecture uses the 3-stage structure shown in Figure.2 to compute the sum.

Initialization stage receives two 16 bit operands and consists of logic blocks to generate bitwise Generate and Propagate signals. Generate(Gi) signal signify whether a carry is generated from that bit and Propagate(Pi) signal signify whether a carry is propagated through that bit.

Gi = Ai & Bi ……(4)

Pi = Ai xor Bi ……(5)

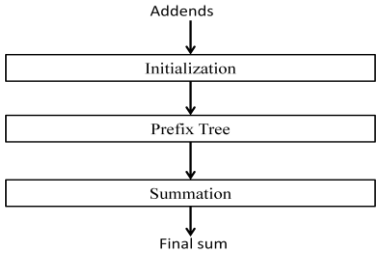

Figure 2: Stages in prefix adder

The Prefix Tree stage consists of gray cells and black cells. The black cell considers two pairs of generate and propagate signals as inputs and produces a pair of generate and propagate signals. The gray cell also considers two pairs of generate and propagate signals as inputs and produces a generate signal.

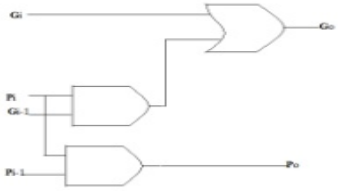

Figure 3: Logic diagram of black cell

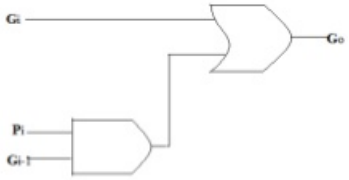

Figure 4: Logic diagram of gray cell

The boolean equations of black cell are

Pi:j = Pi:k & Pk-1:j ……(6)

Gi:j = Gi:k or (Pi:k & Gk-1:j) ……(7)

The boolean equation of gray cell is

Gi:j = Gi:k or (Pi:k & Gk-1:j) ……(8)

The summation stage considers these generate and propogate signals to produce the final sum.

KOGGE-STONE ADDER

It is a parallel prefix form carry look ahead adder. It is developed by peter M.Kogge and Harlord S.stone which they published in 1973. It is the fastest parallel adder. It reduces the critical path to great extent. The main problem with this adder is that it uses so many interconnections to provide the speed. The inter connections lead to increase in the area and power requirements. In the summation stage, the generate bits which are produced in the last prefix stage are XORed with initial propagate bits.

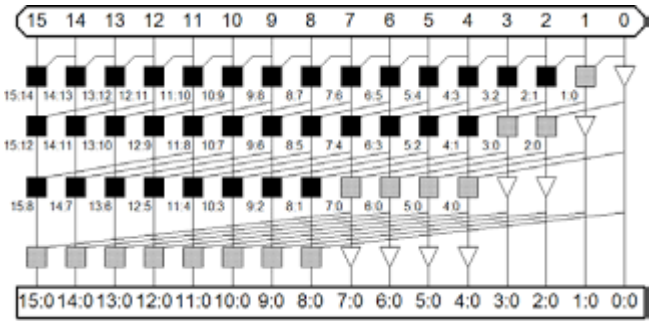

Figure 5: 16-bit Kogge-Stone Adder

SPARSE TREE ADDER

The Sparse Tree adder overcomes the drawback of Kogge Stone adder by reducing the number of wiring junctions needed for its implementation without any prominant effect on its speed. It reduces the total bias current, power consumption and area leading to a more energy efficient system.

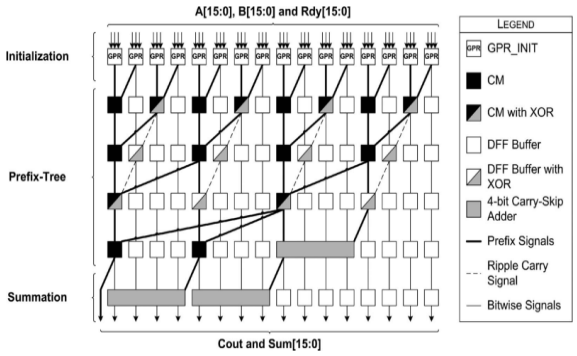

Figure 6: 16-bit Sparse Tree Adder

The GPR_INIT logic blocks creates the bitwise generate and propagate signals which are used by the prefix tree stage. The prefix tree stage has carry merge (CM) blocks which are used to merge the generate and propagate signals and to provide a group carry to the 4-bit adder as shown in the fig.6. The summation stage has a 4-bit adder block for each 4-bit group. The lower half of the adder bits (7:0) can start the
summation stage early.

WALLACE TREE MULTIPLIER

The Wallace Tree Multiplier is efficient in terms of power and regularity without increase in delay and area. It efficiently multiplies two numbers by partitioning the rows of partial products. The power is provided only to the level which is currently taking part in the computation. The multiplier architecture consists of a partial product generation stage in which partial products are generated, partial product reduction stage in which the partial products are reduced to one row of sum and one row of carries, the final addition stage in which an adder circuit performs the addition of sum and carry bits to generate the final result.

CONVENTIONAL WALLACE TREE MULTIPLIER

The partial products are generated in parallel using AND gates. The summation of partial products is done using hierarchal Wallace tree which is divided into levels. The partial products are reduced by using full adders and half adders. To further improve the performance, a parallel prefix sparse tree adder is used at the final stage.

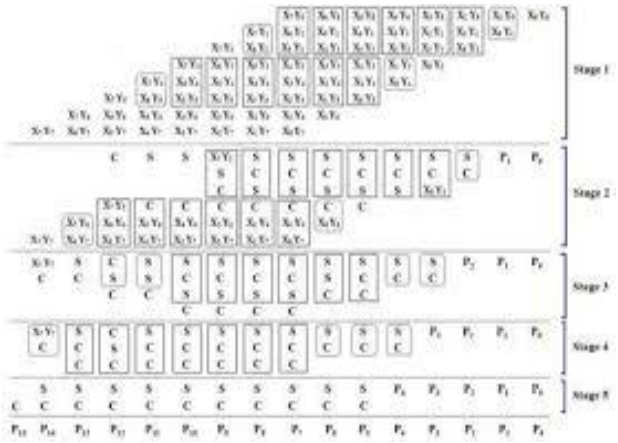

Figure 7: Conventional Wallace Tree Multiplier

MODIFIED WALLACE TREE MULTIPLIER

In the modified Wallace tree multiplier architecture, multi bit compressors are used for performing the reduction in the number of partial product addition stages. The factors including minimum delay , low power and low transistor count makes the compressors the best choice for implementation.

4:2 COMPRESSOR

The 4:2 compressor has 4 inputs and 2 outputs (sum, carry) along with a carryin and carryout. The input cin is the output from the previous lower significant compressor. The cout is the output to the compressor in the next significant stage. To further reduce the delay, the XOR gates are replaced with multiplexers because they have less logical effort when compared to XOR gates.

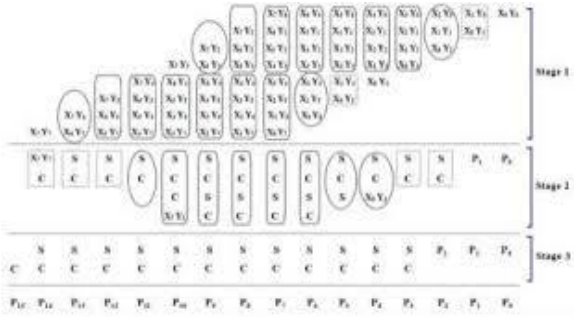

Figure 8: Modified Wallace Tree Multiplier

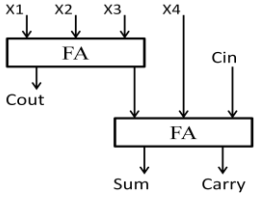

Figure 9: 4:2 Compressor

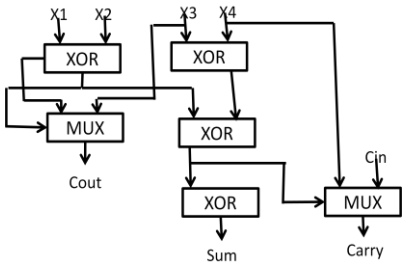

Figure 9: 4:2 Compressor using XOR

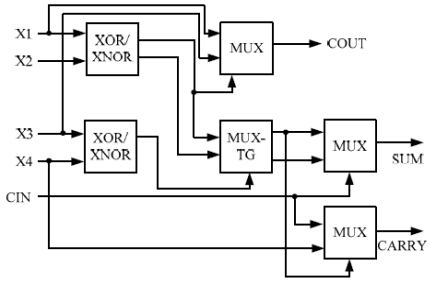

Figure 10: 4:2 Compressor using MUX

The boolean equations of 4:2 Compressor are

Sum = [((x1^x2) & (x3^x4)') + (((x1^x2)') & (x ^x4) &cin')+{(((x1^x2) & (x3 ^x4)') +((x1^x2)' & (x3^x4)))' & cin)}……(9)

Cout = [{(x1 ^ x2) & x3} + { (x1 ^ x2)'& x1}] ……(10)

Carry = [{(x1 ^ x2 ^ x3 ^ x4) & cin} + {(x1 ^ x2 ^ x3 ^ x4)' & x4}] ……(11)

UNIVERSAL VERIFICATION METHODOLOGY

UVM is a methodology for functional verification using SystemVerilog. UVM was created by Accellera based on the Open Verification Methodology(OVM) version 2.1.1. Some of the advantages of UVM include verification reuse, reducing engineer efforts, randomization, defining checkers, covergroups and constraints.

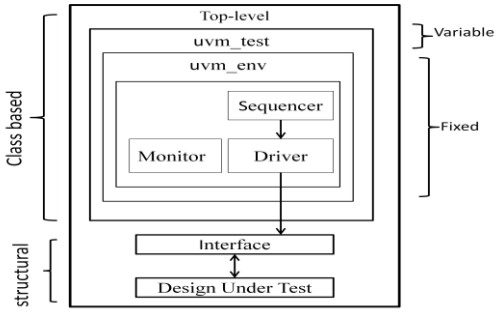

Figure 11: UVM Architecture

The environment is the top-level component of the verification . It contains configuration properties to customize the topology and the behavior and make it reusable. It can contain one or more agents. The agent encapsulate a driver, a sequencer, and a monitor. Agents can be of active or passive type. Passive agents only monitor Design Under Test (DUT) activity. The driver is an active entity that drives the DUT. It receives a data item and drives it to the DUT by sampling and driving the DUT signals. A sequencer is an advanced stimulus generator. It controls the items that are provided to the driver for execution. A monitor is a passive entity that samples DUT signals but does not drive them. It collects the coverage information and performs checking.

RESULTS

An 24 tap, 48 tap, 72 tap FIR filter using the modified Wallace tree multiplier and the parallel prefix sparse tree adder is implemented. The synthesis results shows considerable reduction in the area and power when compared with the fir filter implemented using conventional Wallace tree multiplier and Kogge-stone adder. The coefficients for the filter are generated using MATLAB software. The coverage reports are generated using Universal Verification Methodology.

The designs are simulated and synthesized using synopsys VCS tool and Design Compiler, 90 nm technology library at 25^{o}C temperature and 1.2V operating voltage.

SIMULATION RESULTS

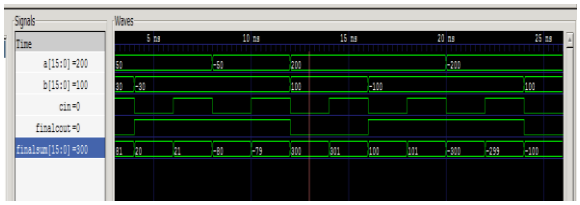

Figure 12: Sparse Tree Adder DVE waveform

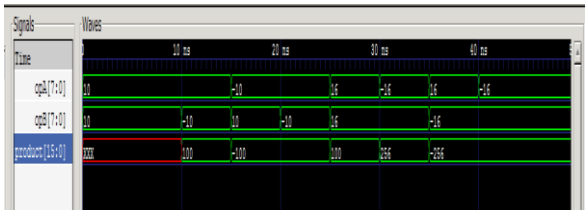

Figure 13: Modified Wallace Multiplier DVE waveform

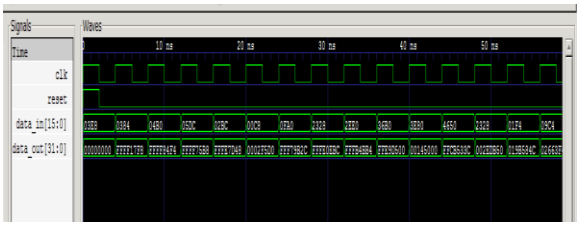

Figure 14: 72 Tap FIR Filter DVE waveform

SYNTHESIS REPORTS

Table.2 Performance comparison table for Wallace Multipliers

Parameter	Area (μm^2)	Power (μW)
Conventional Wallace	3554.14	257.55
Modified Wallace	3493.47	225.52

Table.3 Performance comparison table for 24 Tap FIR Filter

Parameter	Area (mm^2)	Power (μW)
KGA, Conventional Wallace	93.27	872.44
SPA, Modified Wallace	57.57	665.79

Table.4 Performance comparison table for 48 Tap FIR Filter

Parameter	Area (mm^2)	Power (μW)
KGA, Conventional Wallace	188.70	993.90
SPA, Modified Wallace	116.45	720.06

Table.5 Performance comparison table for 72 Tap FIR Filter

Parameter	Area (mm^2)	Power (μW)
KGA, Conventional Wallace	199.93	1046.4
SPA, Modified Wallace	126.33	805.72

COVERAGE REPORTS

Group : \$unit ::spa_coverage::cg

SCORE	WEIGHT	GOAL	AT LEAST	PER INSTANCE	AUTO BIN MAX	PRINT MISSING	COMMENT
91.67	1	100	1	0	64	64	

Summary for Group \$unit ::spa_coverage::cg

CATEGORY	EXPECTED	UNCOVERED	COVERED	PERCENT
Variables	7	1	6	91.67

Variables for Group \$unit ::spa_coverage::cg

VARIABLE	EXPECTED	UNCOVERED	COVERED	PERCENT	GOAL	WEIGHT	AT LEAST	AUTO BIN MAX	COMMENT
rst	1	0	1	100.00	100	1	1	0	
a	1	0	1	100.00	100	1	1	0	
b	1	0	1	100.00	100	1	1	0	
cin	1	0	1	100.00	100	1	1	0	
finalsum	2	1	1	50.00	100	1	1	0	
finalcout	1	0	1	100.00	100	1	1	0	

Figure 14: UVM coverage report for Sparse Tree Adder

Group : \$unit ::wallace_coverage::cg

SCORE	WEIGHT	GOAL	AT LEAST	PER INSTANCE	AUTO BIN MAX	PRINT MISSING	COMMENT
87.50	1	100	1	0	64	64	

Summary for Group \$unit ::wallace_coverage::cg

CATEGORY	EXPECTED	UNCOVERED	COVERED	PERCENT
Variables	8	1	7	87.50

Variables for Group \$unit ::wallace_coverage::cg

VARIABLE	EXPECTED	UNCOVERED	COVERED	PERCENT	GOAL	WEIGHT	AT LEAST	AUTO BIN MAX	COMMENT
rst	2	1	1	50.00	100	1	1	0	
opA	2	0	2	100.00	100	1	1	0	
opB	2	0	2	100.00	100	1	1	0	
product	2	0	2	100.00	100	1	1	0	

Figure 15: UVM coverage report for Wallace Multiplier

Group : \$unit ::fir_coverage::cg

SCORE	WEIGHT	GOAL	AT LEAST	PER INSTANCE	AUTO BIN MAX	PRINT MISSING	COMMENT
83.33	1	100	1	0	64	64	

Summary for Group \$unit ::fir_coverage::cg

CATEGORY	EXPECTED	UNCOVERED	COVERED	PERCENT
Variables	4	1	3	83.33

Variables for Group \$unit ::fir_coverage::cg

VARIABLE	EXPECTED	UNCOVERED	COVERED	PERCENT	GOAL	WEIGHT	AT LEAST	AUTO BIN MAX	COMMENT
reset	2	1	1	50.00	100	1	1	0	
data_in	1	0	1	100.00	100	1	1	0	
data_out	1	0	1	100.00	100	1	1	0	

Figure 16: UVM coverage report for 72 Tap FIR Filter

CONCLUSION

The Direct Form Finite Impulse Response Filter is designed by using Parallel Prefix Adders and Wallace Tree Multiplier. Experimental results indicate that the implemented designs are very effective in terms of area and power. The verification is performed using Universal Verification Methodology (UVM) and more than 80% of coverage is obtained for every design.

Acknowledgment

I express my profound sense of gratitude to my guide Smt.P.PushpaLatha, Assistant professor, JNTU Kakinada for her systematic guidance and valuable advices. I would like to express my sincere gratitude

to Dr.A.M.Prasad, professor, Head of the department and Dr. K.Babulu, professor, course coordinator, for providing me with all the necessary facilities for the work.

References

[1]. R. Nakamoto, S. Sakuraba, T. Onomi, S. Sato, and K. Nakajima, "4-bit SFQ Multiplier Based on Booth Encoder," IEEE Trans. Appl. Supercond., vol. 21, no. 3, pp. 852–855, Jun. 2011.

[2]. Mikhail Dorojevets, Christopher L.Ayala, Nobuyuki Yoshikawa and Akira Fujimaki, "16-Bit Wave-Pipelined Sparse-Tree RSFQ Adder," IEEE Transactions on Applied Superconductivity, vol. 23, no. 3, June 2013.

[3]. M.Dorojevets, C.L.Ayala, and A.K.Kasperek, "Data-flow microarchitecture for wide datapath RSFQ processors: Design study," IEEE Trans. Appl. Supercond., vol. 21, no. 3, pp. 787–791, Jun. 2011.

[4]. T.Filippov, M.Dorojevets, A.Sahu, A.Kirichenko, C. Ayala, and O. Mukhanov, "8-bit asynchronous wave- pipelined RSFQ arithmeticlogic unit," IEEE Trans. Appl. Supercond., vol. 21, no. 3, pp. 847–851, Jun. 2011.

[5]. S. Saddam Hussain , "Comparative Design of 16-Bit Sparse-Tree RSFQ Adder ,"Int. Journal of Engineering Research and Applications , Vol. 4, no. 6, pp.184-188 . June 2014 .

[6]. V.Gowrishankar, D.Manoranjitham, P. Jagadeesh "Efficient FIR Filter Design Using Modified Carry Select Adder & Wallace Tree Multiplier," International Journal of Science, Engineering and Technology Research, Vol. 2, no. 3, March 2013.

[7]. Deepshikha Bharti , K. Anusudha , "High Speed FIR Filter Based on Truncated Multiplier and Parallel Adder," International Journal of Engineering Trends and Technology, Vol. 5, no. 5, Nov 2013.

[8]. Sarita Chouhan, Yogesh Kumar, "LOW POWER DESIGNING OF FIR FILTERS," International Journal of Advanced Technology & Engineering Research, Vol.2, no.2, May 2012.

[9]. K.GopiKrishna, B.Santhosh2 , V.Sridhar, "Design of Wallace Tree Multiplier using Compressors," INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY, Vol.2, no.9, Sept 2013.

[10]. Jagadeshwar Rao , Sanjay Dubey, "A High Speed Wallace Tree Multiplier Using Modified Booth Algorithm for Fast Arithmetic Circuits," IOSR Journal of Electronics and Communication Engineering, Vol.3, no.1, Sept 2012.