Area Efficient Architecture for NR4SD+ Encoding using Pre-Encoded Dadda Multiplication Algorithm

Govind Singh Solanki[1] , Prof. Devesh Kishore[2] , Prof. Aastha Hajari[3]
[1]Mtech. Research Scholar, [2]Research Guide, [3]HOD Department of Electronics and Communication
SKSITS Indore

Abstract - Multiplication is a notable basic arithmetic operation used in everyday life estimations. Since multiplication commands the execution time of most DSP algorithms, so always there is a need of high speed multiplier. Right now, multiplication time is as yet that predominant factor in deciding the instruction process duration of a DSP chip. Due to its complexity in logic multipliers are always a large block of any processor unit. Previously, numerous original thoughts for multipliers have been proposed to accomplish superior performance. The interest for rapid processing has been expanding because of extending PC and signal processing applications. Higher throughput arithmetic operations are basic need to accomplish the desired performance in some continuous signal, image processing and graphic processing applications. Multiplication is the one of the fundamental arithmetic operation for such application. Over the decade's improvement of multiplier logic circuit complexity is the subject of research. The fundamental requirement for most of applications to reduce the delay in time and power consumption. To overcome the above requirement an Area Efficient Architecture for NR4SD+ Encoding using Pre-Encoded Dadda Multiplication Algorithm has been implemented and simulated on Xilinx ISE using HDL language.

Keywords- Arithmetic operation , Multiplication, Adder, Fast multiplier, decoder, Radix, Dadda Multiplication Algorithm.

I. INTRODUCTION

Multiplication is a major operation in most signal processing algorithms. Multipliers have huge area, long idleness and devour impressive power. Accordingly low-control multiplier configuration has been an imperative part in low-control VLSI framework outline. There has been broad work on low-control multipliers at innovation, physical, circuit and logic levels. A framework's performance is for the most part controlled by the performance of the multiplier in light of the fact that the multiplier is by and large the slowest component in the framework. Besides, it is for the most part the most area expending. Consequently, enhancing the speed and area of the multiplier is a noteworthy outline issue. Notwithstanding, area and speed are typically clashing requirements with the goal that enhancing speed comes about for the most part in bigger areas. Therefore, an entire range of multipliers with various area-speed imperatives has been planned with completely parallel.

The requirement for low-control VLSI system emerges from two fundamental powers. To begin with, with the enduring growth of working frequency and processing capacity per chip, expansive streams must be conveyed and the heat because of huge power utilization must be dissipated by appropriate cooling methods. Second, battery life in compact electronic gadgets is constrained. Low power configuration specifically prompts delayed operation time in these versatile gadgets.

Digital Signal Processors (DSPs) furthermore, application specific integrated circuits depend on the effective usage of arithmetic circuits to execute devoted algorithms, for example, convolution, connection and separating. In this part, two new systems are proposed to actualize multiplier circuits. The principal procedure utilizes decay logic and enhances the general power defer result of the multiplier. Disintegration calculation is tried on various multiplier circuits, for example, Carry-Save multiplier, Wallace multiplier and Dadda multiplier.

Dadda and Carry-Save multipliers are commonly accessible models for performing multiplication. Carry Save multiplier is gotten from an array multiplier. Tree organized multipliers like Wallace and Dadda passage well, in spite of their anomaly and overabundance wiring. This is because of the way that tree multipliers offer littler profundity of incomplete item decrease hardware, which thusly appears to counterbalance the power misfortune in wiring. This diminishes the general power dispersal and delay.

Multiplication can be considered to comprise of three essential advances: age of fractional item (PPG), incomplete items decrease (PPR), lastly toward the end expansion of carry propagate(CPA). In general combinational and successive multiplier usage. Here taking into consideration the combinational case only, in light of the fact that the size of joining now has turned out to be sufficiently immense to begin obliging parallel multiplier applications in digital VLSI circuits. Distinctive multiplication algorithms shift in the methodologies of age and decrease of Partial Products and the expansion procedure. With a specific end goal to decrease the quantity of PPs included and in this manner reduce the area/delay of the circuit, one operand is normally recoded into high-radix digit sets. A standout amongst the most utilized and across the board radix-2n calculation is the radix-4 which has an arrangement of digits.

The end procedure of expansion requires a quick adder design since it is on the basic way. In a couple of cases, finishing up summation is conceded in the event that it is profitable to keep excess outcomes from PPG to complete further arithmetic operations.

II. DADDA MULTIPLIER

Dadda (1965) generalized and extended Wallace's results by noting that a full adder can be thought of as a circuit, which counts the number of ones in the input and outputs that number in 2-bit binary form. Using such a counter, Dadda postulated that, at each stage, only minimum amount of reduction should be done in order to reduce the partial product matrix by a factor of 1.5. Dadda's method requires the same number of levels as that of Wallace method. However Dadda's method does the minimum reduction necessary at each level. This results in a design with fewer full adders and half adders. The number of (3,2) and (2,2) counters required is minimized in Dadda's technique compared to Wallace tree. The disadvantage of Dadda's method is that it requires a slightly wider fast Carry Propagate Adder (CPA) and has a less regular structure than Wallace. Figure 2.1 shows a 8x8 Dadda multiplier.

For a N × N bit Dadda multiplier , there are N^2, bits in the original partial product matrix and 4×N-3 bits in the final row matrix.

The total number of (3,2) counters required is given by $N^2 - 4 \times N + 3$. The length of the

carry propagate adder (CPA) is given by $2 \times N - 2$. The total number of (2,2) counters required is given by $N - 1$.

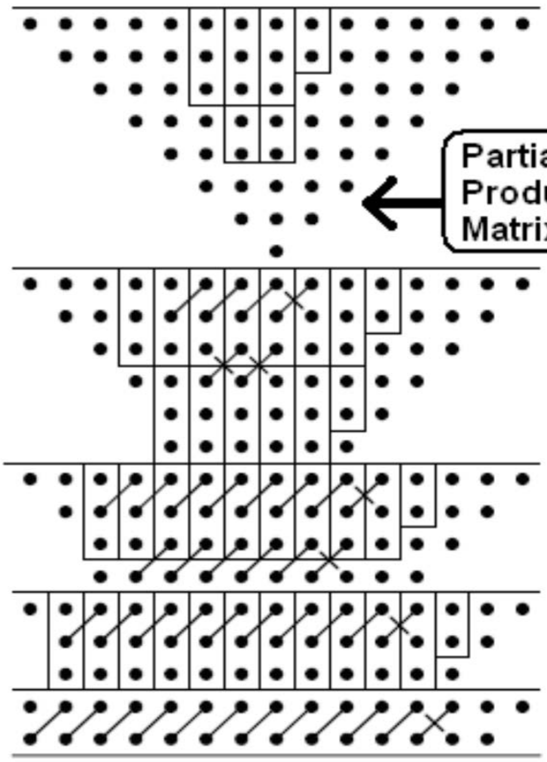


Figure 2.1Illustration of Dadda Multiplier for 8*8 Multiplication.

III. PROPOSED ARCHITECTURE

The proposed architecture is based on NR4SD+ Encoding using Pre-Encoded Dadda Multiplication Algorithm. In figure 3.1 illustrated RTL schematic of proposed design with 32 bit width. there are three sub modules in the proposed design are A ROM of 32 Bit, two encoder M1 and M2.

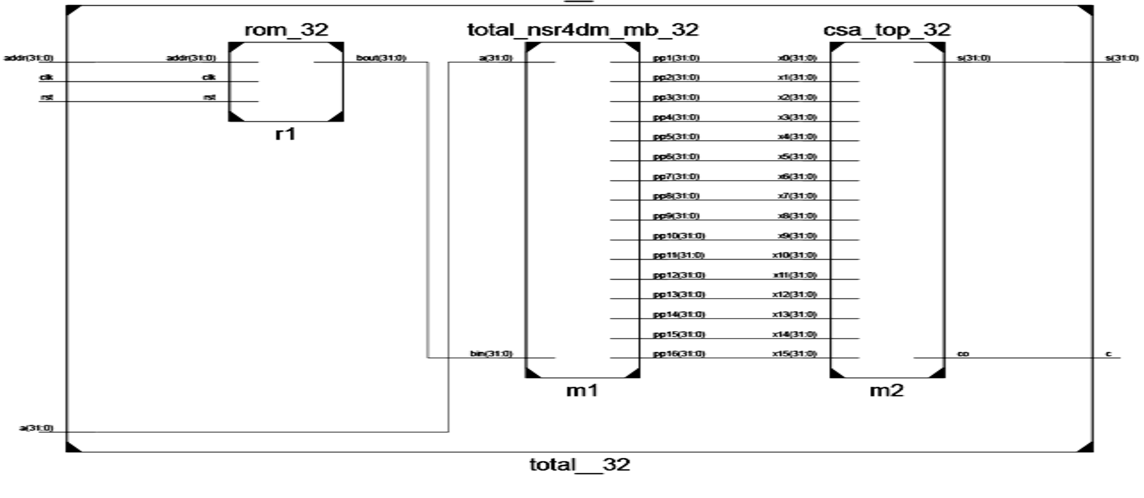


Figure 3.1 RTL Schematic of Main Modules of Proposed Design.

Dadda NR4SD based 32-bit multiplier and 64-bit multipliers have been coded in VHDL HDL and simulated and synthesized using Xilinx ISE design suite 13.1. Radix-4 Booth's recoding algorithm is utilized to generate the partial products. Number of partial product rows gets reduced to half with the utilization of radix-4 Booth's algorithm. Accumulation of these partial product rows is done using different compressors and also using tree compressions techniques. Last step in multiplication is the accumulation of remaining two rows which is done with the help of ripple carry adder.

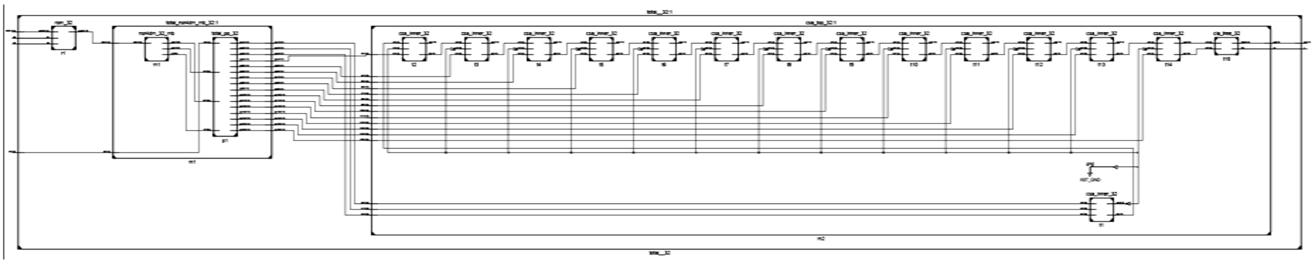
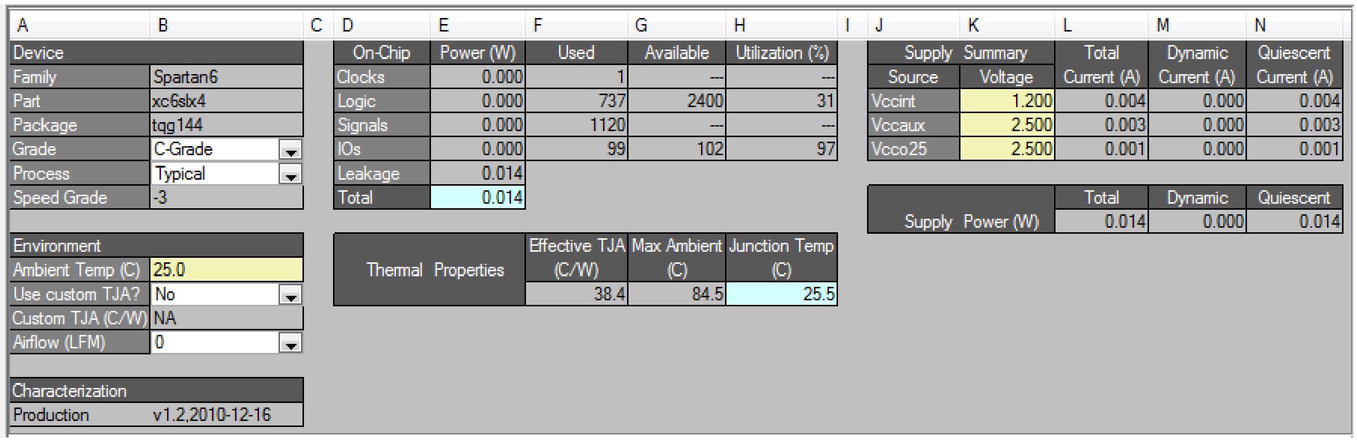


Figure 3.2 Extended RTL Schematic of Proposed Design.

IV. SIMULATION RESULTS

Simulation of proposed work has done on Xilinx 13.1 ISE design suite using VHDL hardware descriptive language the outcomes of proposed work has give in fig. below figure 4.1 has give power analyst of proposed design. the proposed system outperforms as compared to existing system. The comparative analysis of proposed system with respect to existing system has given in table 4.1 in terms of power in mV and percent improvement in power consumption.



Device	
Family	Spartan6
Part	xc6slx4
Package	tqg144
Grade	C-Grade
Process	Typical
Speed Grade	-3

Environment	
Ambient Temp (C)	25.0
Use custom TJA?	No
Custom TJA (C/W)	NA
Airflow (LFM)	0

Characterization	
Production	v1.2,2010-12-16

On-Chip	Power (W)	Used	Available	Utilization (%)
Clocks	0.000	1	---	---
Logic	0.000	737	2400	31
Signals	0.000	1120	---	---
IOs	0.000	99	102	97
Leakage	0.014			
Total	0.014			

Thermal Properties	Effective TJA (C/W)	Max Ambient (C)	Junction Temp (C)
	38.4	84.5	25.5

Supply Summary Source	Voltage	Total Current (A)	Dynamic Current (A)	Quiescent Current (A)
Vccint	1.200	0.004	0.000	0.004
Vccaux	2.500	0.003	0.000	0.003
Vcco25	2.500	0.001	0.000	0.001

	Total	Dynamic	Quiescent
Supply Power (W)	0.014	0.000	0.014

Figure 4.1 power utilization summary of proposed work.

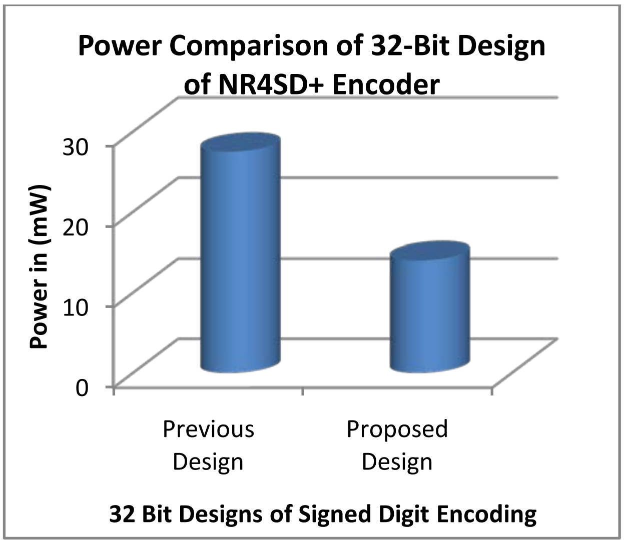


Figure 4.2 Power Comparison of 32- bit design of NR4SD+ Encoder.

Table 1: Comparison of Power Consumption.

Design	Previous Design	Proposed Design	% Improvements
32-Bit	27.5 mW	14mW	49% Reduction

Table 2: Comparison of Power Consumption.

Parameters	32-Bit Design	64-Bit Design
Delay	72.827 ns	41.708 ns

Slice Registers	32	62
Slice LUTs	1044	112
FF Pairs	1050	113
Bonded IOBs	99	130

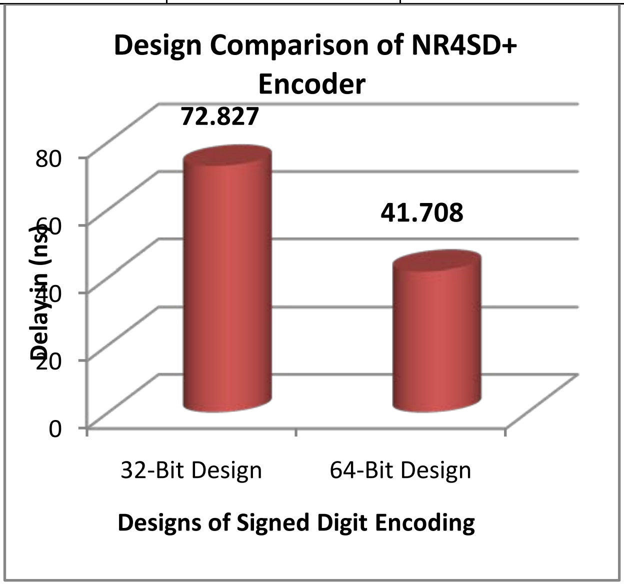


Figure 4.3 Design Comparison of NR4SD+ Encoder-32 bit and 64-bit.

Figure 4.3 shows the design comparison of proposed NR4SD encoder 32 bit design with the 64 bit. hare delay is represented in nano second. delay of earlier 32-Bit Design was 72.827 ns which is reduced to 41.708 ns. for 64 bit design.

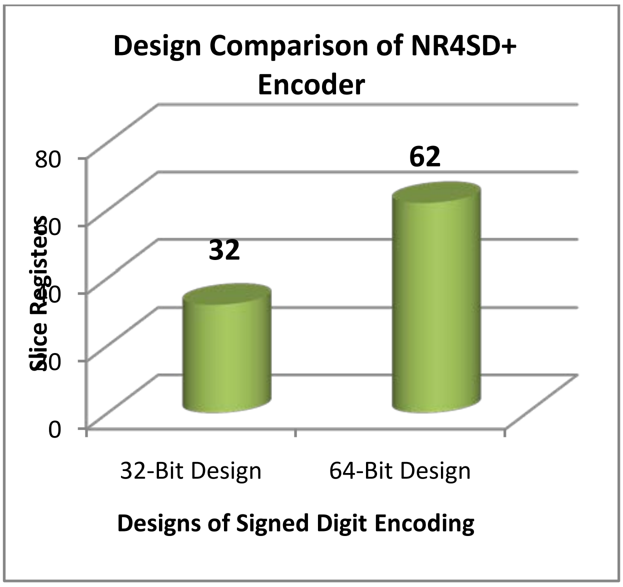


Figure 4.4 Deign Comparison of proposed work in terms of Slice registers.

Figure 4.4 Shows the graphical representation of proposed design for a signed digit encoding where comparison of area it terms of slice registers are shown in figure. for 32 bit existing work and 64 bit proposed work. Comparatively number of slice registers are reduced for 64 bit design which is also great achievement of proposed work.

The graphical representation of Number of Lookup table are comparison is illustrated in figure 4.5. for 32 bit existing design and 647 bit proposed design. which is comparatively very less.

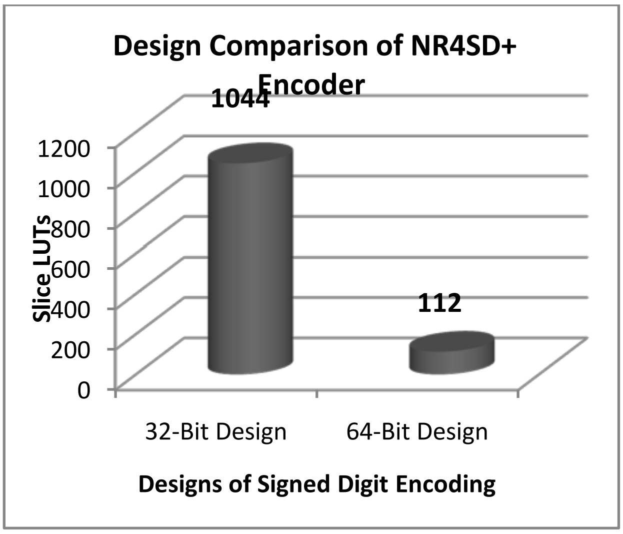


Figure 4.5 Design comparison in Slice LUTs.

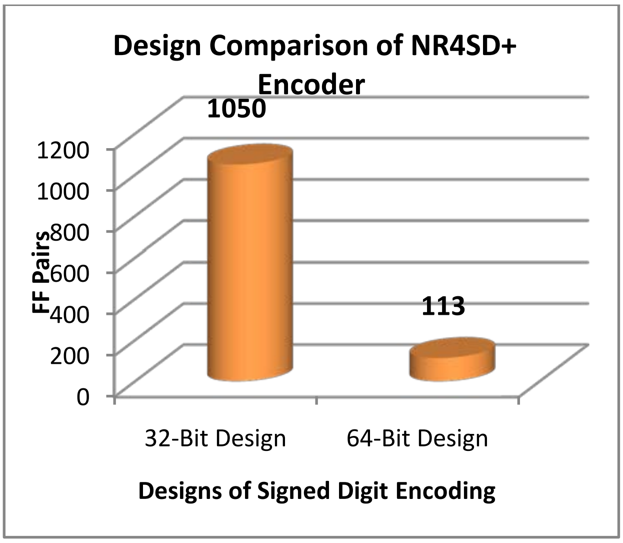


Figure 4.6 Design Comparison in terms of Flip Flop pairs.

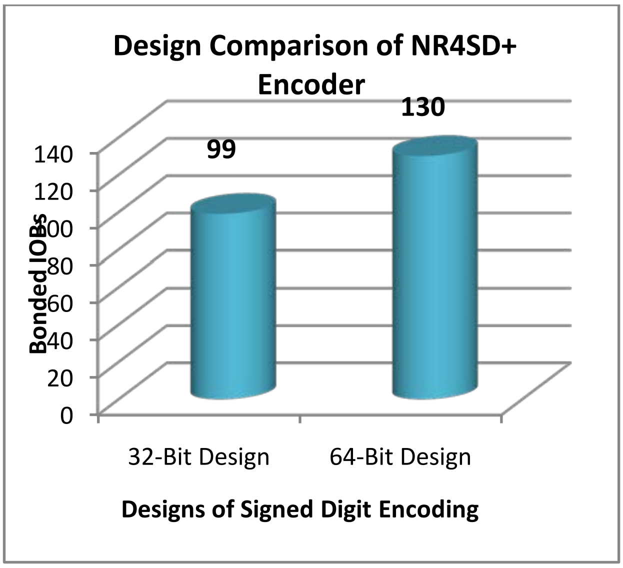


Figure 4.7 Design Comparison of Bonded IOBs.

The design comparison of No of Flip Flop pairs and Design Comparison of Bonded IOBs are shown in figure 4.6 and fig. 4.7 respectively for 32 bit existing work and 64 bit proposed. The performance of proposed work is batter in every aspect such as area power and dely.

V. CONCLUSION

Multiplier in light of Modified Booth calculation and Wallace expansion is one of the quick and low power multiplier. An Area Efficient Architecture for NR4SD+ Encoding using Pre-Encoded Dadda Multiplication Algorithm has been implemented in this work and simulated on Xilinx ISE suite using HDL language. The outcome or proposed work has been compared with existing work for the comparative study. the performance of proposed work has batter than the existing work. The percent improvement in proposed work with respect to existing work has been evaluated is about to 49%. As an attempt to develop, arithmetic algorithm and architecture level optimization techniques for low power high-speed multiplier design, techniques presented in this research work has obtained good results on expectation. However, there are limitations in this work and several future research directions are possible.

REFERENCES

[1] K. Tsoumanis, N. Axelos, N. Moschopoulos, G. Zervakis and K. Pekmestzi, "Pre-Encoded Multipliers Based on Non-Redundant Radix-4 Signed-Digit Encoding," in IEEE Transactions on Computers, vol. 65, no. 2, pp. 670-676, Feb. 1 2016.

[2] S. Asif and Yinan Kong, "Performance analysis of Wallace and radix-4 Booth-Wallace multipliers," 2015 Electronic System Level Synthesis Conference (ESLsyn), San Francisco, CA, 2015, pp. 17-22.

[3] M. Kumm, S. Abbas and P. Zipf, "An Efficient Softcore Multiplier Architecture for Xilinx FPGAs," 2015 IEEE 22nd Symposium on Computer Arithmetic, Lyon, 2015, pp. 18-25.

[4] C. Xiaoping, H. Wei, C. Xin and W. Shumin, "A New Redundant Binary Partial Product Generator for Fast 2n-Bit Multiplier Design," 2014 IEEE 17th International Conference on Computational Science and Engineering, Chengdu, 2014, pp. 840-844.

[5] K. Javeed and Xiaojun Wang, "Radix-4 and radix-8 booth encoded interleaved modular multipliers over general Fp," 2014 24th International Conference on Field Programmable Logic and Applications (FPL), Munich, 2014, pp. 1-6.

[6] R. Muralidharan and C. H. Chang, "Radix-4 and Radix-8 Booth Encoded Multi-Modulus Multipliers," in IEEE Transactions on Circuits and Systems I: Regular Papers, vol. 60, no. 11, pp. 2940-2952, Nov. 2013.

[7] H. Ding, P. Shu, X. Wang and J. Yang, "A Design and Implementation of Decimal Floating-point Multiplication Unit Based on SOPC," 2012 Third International Conference on Digital Manufacturing & Automation, GuiLin, 2012, pp. 36-41.

[8] C. Wang, W.-S. Gan, C. C. Jong, and J. Luo, “A low-cost 256-point fft processor for portable speech and audio appli- cations,” in Int. Symp. on Integrated Circuits (ISIC 2007), Sep. 2007, pp. 81–84.

[9] A. Jacobson, D. Truong, and B. Baas, “The design of a recon- figurable continuous-flow mixed-radix fft processor,” in IEEE Int. Symp. on Circuits and Syst. (ISCAS 2009), May 2009, pp. 1133–1136.

[10] Y. T. Han, J. S. Koh, and S. H. Kwon, “Synthesis filter for mpeg-2 audio decoder,” Patent US 5 812 979, Sep., 1998.

[11] M. Kolluru, “Audio decoder core constants rom optimization,” Patent US 6 108 633, Aug., 2000.

[12] H.-Y. Lin, Y.-C. Chao, C.-H. Chen, B.-D. Liu, and J.-F. Yang, “Combined 2-d transform and quantization architectures for h.264 video coders,” in IEEE Int. Symp. on Circuits and Syst. (ISCAS 2005), vol. 2, May 2005, pp. 1802–1805.

[13] G. Pastuszak, “A high-performance architecture of the double- mode binary coder for h.264.avc,” IEEE Trans. Circuits Syst. Video Technol., vol. 18, no. 7, pp. 949–960, Jul. 2008.

[14] J. Park, K. Muhammad, and K. Roy, “High-performance fir filter design based on sharing multiplication,” IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 11, no. 2, pp. 244–253, Apr. 2003.