International Journal of Research

Available at https://edupediapublications.org/journa

p-ISSN: 2348-6848
e-ISSN: 2348-795X
Volume 04 Issue 02
February 2017

Encoding Constant Coefficients to Contain the Least Non-Zero Digits

[1] L SANDEEP CHOWDARY [2] HARITHA KAKUMANU

[1]PG SCHOLAR,DEPARTMENT OF ECE, SPHOORTHY ENGINEERING COLLEGE,HYDERABAD

[2]ASSISSTANT PROFESSOR, DEPARTMENT OF ECE, SPHOORTHY ENGINEERING COLLEGE,HYDERABAD

ABSTRACT:

Multimedia and Digital Signal Processing (DSP) programs (e.g., Fast Fourier Transform (FFT), audio/video CoDecs) execute a lot of multiplications with coefficients that don't change throughout the execution from the application. It's observed the pre-encoded NR4SD architectures tend to be more area efficient compared to conventional or pre-encoded MB designs regarding their performance within the cheapest possible clock period. Within this paper, we introduce architecture of pre-encoded multipliers for Digital Signal Processing programs according to off-line encoding of coefficients. A CSD-based prrr-rrrglable multiplier design was suggested for categories of pre-determined coefficients that share certain features. The suggested NR4SD encoding plan uses among the following teams of digit values. To be able to cover the dynamic selection of the 2's complement form, all numbers from the suggested representation are encoded based on NR4SD except the most important one that's MB encoded. For this extend, the Non-Redundant radix-4 Signed-Digit (NR4SD) encoding technique, which utilizes the digit values, is suggested resulting in a multiplier design with less complex partial items implementation. The performance from the suggested designs is recognized as with regards to the width from the input figures. Extensive experimental analysis confirms the suggested pre-encoded NR4SD multipliers, such as the coefficients memory, tend to be more area and power efficient compared to conventional Modified Booth plan.

Keywords: Multiplying circuits, Modified Booth encoding, Pre-Encoded multipliers

I. INTRODUCTION

Constant coefficients could be encoded to retain the least non-zero numbers while using Canonic Signed Digit (CSD) representation. CSD multipliers comprise the least non-zero partial items, which decreases their switching activity. However, the CSD encoding involves serious restrictions [1]. Because the multiplier is really a fundamental component for applying computationally intensive application, its architecture seriously affects their performance. Folding technique, which reduces plastic area by time multiplexing many procedures into single functional models, e.g., adders, multipliers, isn't achievable because the CSD-based multipliers are difficult-wired to particular coefficients. A CSD-based prrr-rrrglable multiplier design was suggested for categories of pre-determined coefficients that share certain features. Also, this process can't be easily extended to large categories of pre-determined coefficients attaining simultaneously high quality. Modified Booth (MB) encoding takes up these restrictions and reduces to half the amount of partial items bringing on reduced area, critical delay and power consumption [2]. How big ROM accustomed to keep categories of coefficients is considerably reduced along with the area and power use of the circuit. Multimedia and Digital Signal Processing (DSP) programs (e.g., Fast Fourier Transform (FFT), audio/video CoDecs) execute a lot of multiplications with coefficients that don't change throughout the execution from the application. However, this multiplier design lacks versatility because the partial items generation unit was created particularly for several coefficients and can't be reused for an additional group.. However, a devoted encoding circuit is needed and also the partial items generation is much more complex. Kim et al. suggested a method similar, for creating efficient MB multipliers for categories of pre-determined coefficients with similar restrictions described in the last paragraph. Because the values of constant coefficients are known ahead of time, we encode the coefficients off-line in line with the MB encoding and keep MB encoded coefficients (i.e., 3 bits per digit) right into a ROM. We make reference to this design as pre-encoded MB multiplier. Then, we explore a Non-Redundant radix-4 Signed-Digit (NR4SD) encoding plan stretching the

serial encoding techniques. The suggested NR4SD encoding plan uses among the following teams of digit values. To be able to cover the dynamic selection of the 2's complement form, all numbers from the suggested representation are encoded based on NR4SD except the most important one that's MB encoded. While using suggested encoding formula, we pre-encode the conventional coefficients and store them right into a ROM inside a condensed form (i.e., 2 bits per digit) [3]. In comparison towards the pre-encoded MB multiplier where the encoded coefficients need 3 bits per digit, the suggested NR4SD plan cuts down on the memory size. Also, in comparison towards the MB form, which utilizes five digit values the suggested NR4SD encoding uses four digit values? Thus, the NR4SD-based pre-encoded multipliers incorporate a less complex partial items generation circuit. We explore the efficiency from the aforementioned pre-encoded multipliers considering how big the coefficients' ROM.

II. EXISTING ALGORITHMS

As with MB form, the amount of partial items is reduced to half. When encoding the 2's complement number B, Modified Booth (MB) is really a redundant radix-4 encoding technique [4]. Thinking about the multiplication from the 2's complement figures A, B. we present the Non-Redundant radix-4, Signed-Digit (NR4SD) encoding technique. As noticed in the NR4SD-encoding technique, the NR4SD form has bigger dynamic range compared to 2's complement form. Thinking about the 8-bit 2's complement number N, two typical values of N, and is definitely the MB, NR4SD- and NR4SD numbers that result when using the related encoding strategies to each worth of N we considered. We added a bar over the adversely signed numbers to be able to distinguish them in the positively signed ones.

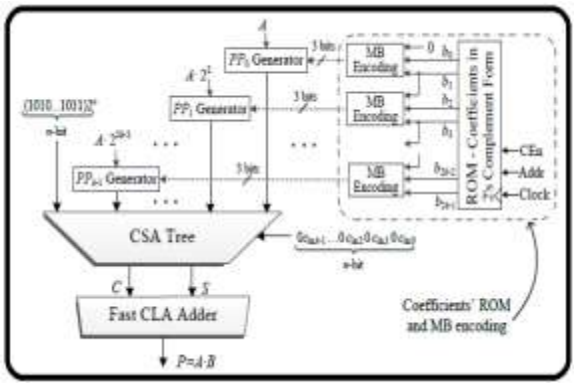

Fig.1. Framework of the conventional multiplier

III. PROPOSED IMPLEMENTION

We take into account that this input develops from a group of fixed coefficients e.g. the coefficients for several filters by which this multiplier is going to be utilized in a devoted system or even the sine table needed within an FFT implementation. We explore the implementation of pre-encoded multipliers. Among the two inputs of those multipliers is pre-encoded in both MB or perhaps in NR4SD+/NR4SD representation. The coefficients are encoded off-line according to MB or NR4SD calculations and also the resulting items of encoding are kept in a ROM. Since our purpose would be to estimate the efficiency from the suggested multipliers, we first present overview of the traditional MB multiplier to be able to compare it using the pre-encoded schemes. The architecture from the system which comprises the traditional MB multiplier and also the ROM with coefficients in 2's complement form. Within the pre-encoded MB multiplier plan, the coefficient B is encoded off-line based on the conventional MB form. The multiplier and also the PPG from the pre encoded MB, NR4SD+ and NR4SD designs present within the conventional MB plan. The comparison one of the designs starts in the cheapest common achievable clock period for those designs and continues at greater clock periods by growing the time period by step .2 ns until it reaches 4 ns. It's observed the pre-encoded NR4SD architectures tend to be more area efficient compared to conventional or pre-encoded MB designs regarding their performance within the cheapest possible clock periods [5]. Regarding power dissipation, the pre-encoded NR4SD_ plan consumes minimal power which, within the installments of 16 and 24 items of input width, is equivalent to the ability consumed through the pre-encoded MB design. The machine architecture for that pre-encoded NR4SD multipliers is presented we lessen the memory requirement to n 1 bits per coefficient as the corresponding memory needed for that pre-encoded MB plan is 3n/2 bits per coefficient. Thus, the quantity of stored bits is equivalent to those of the traditional MB design, except which are more significant digit that requires an additional bit because it is MB encoded. As clock period increases, the datapath from the multiplication circuit changes and also the standard cells employed for its synthesis dwindle complex regarding area occupation, internal capacitance and ports' load.

However, the ROM utilized in each evaluated design is really a standard cell and it is critical delay, area occupation and both internal and ports' load remain unchanged as clock period increases. We used Synopsys Design Compiler and also the Faraday 90 nm standard cell library to synthesize the evaluated designs, thinking about the greatest optimization degree and maintaining your hierarchy from the designs. The performance from the suggested designs is recognized as with regards to the width from the input figures. Finally, we used Synopsys Prime Time to calculate power consumption.

IV. CONCLUSION

The suggested pre-encoded NR4SD multiplier designs tend to be more area and power efficient in comparison towards the conventional and pre-encoded MB designs. We advise encoding these coefficients within the Non-Redundant radix-4 Signed-Digit (NR4SD) form. Within this paper, new types of pre-encoded multipliers are investigated by off-line encoding the conventional coefficients and storing them in system memory. The performance from the suggested designs is recognized as with regards to the width from the input figures. Extensive experimental analysis confirms increases from the suggested pre-encoded NR4SD multipliers when it comes to area complexity and power consumption in comparison towards the conventional MB multiplier.

REFERENCES

[1] N. Weste and D. Harris, CMOS VLSI Design: A Circuits and Systems Perspective, 4th ed. USA: Addison-Wesley Publishing Company, 2010.

[2] B. Paul, S. Fujita, and M. Okajima, "Rom-based logic (rbl) design: A low-power 16 bit multiplier," IEEE J. Solid-State Circuits, vol. 44, no. 11, pp. 2935–2942, Nov. 2009.

[3] G. W. Reitwiesner, "Binary arithmetic," Advances in Computers, vol. 1, pp. 231–308, 1960.

[4] Z. Huang, "High-level optimization techniques for low-power multiplier design," Ph.D. dissertation, Department of Computer Science, University of California, Los Angeles, CA, 2003.

[5] 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.