Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X
IMPACT FACTOR: 6.017

IJCSMC, Vol. 6, Issue. 8, August 2017, pg.36 – 39

Design and Implementation of 4-bit Decimal Multiplier using Sign Magnitude Encoding

Kiran.R.Pawar, Asst. Prof. Mahesh B Neelagar
Department of Studies in VLSI Design & Embedded System Engineering, VTU Belagavi, India
kiranp660@gmail.com
Asst. Professor, Department of Studies in VLSI Design & Embedded System Engineering, VTU Belagavi, India
neelagarmahesh@gmail.com

Abstract- Decimal X×Y multiplication is a complex operation, where intermediate partial products (IPPs) are commonly selected from a set of pre-computed radix-10Xmultiples.Some works require only[0,5]×Xvia recoding digits of to one-hot representation of signed digits in [−5,5]. This reduces the selection logic at the cost of one extra IPP. Two's complement signed-digit (TCSD) encoding is often used to represent IPPs, where dynamic negation (via one xor per bit of X multiples) is required for the recoded digits of Y in[−5,−1].In this paper, despite generation of 5IPPs, for 4-digit operands, we manage to start the partial product reduction (PPR) with 5 IPPs that enhance the VLSI regularity. Moreover, we save 75% of negating xors via representing pre-computed multiples by sign-magnitude signed-digit (SMSD) encoding. For the first-level PPR, we devise an efficient adder, with two SMSD input numbers, whose sum is represented with TCSD encoding. Thereafter, multilevel TCSD 2:1 reduction leads to two TCSD accumulated partial products, which collectively undergo a special early initiated conversion scheme to get at the final binary-coded decimal product. As such, a VLSI implementation of 4×4-digit parallel decimal multiplier is synthesized, where evaluations show some performance improvement over previous relevant designs.

Keywords- Radix-10 multiplier, redundant representation, sign-magnitude signed digits (SMSDs), VLSI design.

I. INTRODUCTION

DECIMAL arithmetic hardware is highly demanded for fast processing of decimal data in monetary, Web-based and human interactive applications [1]. Fast radix-10 multiplication, in particular, can be achieved via parallel partial product generation (PPG) and partial product reduction (PPR),which is, however, highly area consuming in VLSI implementations. Therefore, it is desired to lower the silicon cost, while keeping the high speed of parallel realization. Let P=X×Y represent an n×n decimal multiplication, where multiplicand X, multiplier Y and product P are normal radix-10 numbers with digits in [0,9]. Such digits are commonly represented via binary-coded decimal (BCD) encoding. However, intermediate partial products (IPPs) are represented via a diversity of often redundant decimal digit sets and encodings. The choice of alternative IPP representations is influential on the PPG, which is of particular importance in decimal multiplication from two points of view: one is fast and low cost generation of IPPs and the other is its impact on representation of IPPs, which is influential on PPR efficiency. In this paper, we aim to take advantage of[−5,5]SMSD recoding of multiplier and dynamic negation of X multiples, while reducing the number of XOR gates via generating[−6,6]SMSD pre-computed X multiples.

II. Methodology

Binary coded decimal Digit multiplier; Decimal computer arithmetic is preferred in decimal data processing environments such as scientific, commercial, financial and Internet-based applications. considering these requirements we have designed BCD multipliers which performs the multiplication operation of decimal numbers.it mainly performs three operations i.e partial product generation, partial product reduction and partial product computation.in this design we used an Full adder, half adder and basic gates.in our design we considered two four bit decimal numbers i.e,X0X1X2X3 and Y0Y1Y2Y3,output is P0P1P2P3P4P5P6.by using Full adder and Half adder circuits we uses more number of transistors due to this total delay for BCD multipliers are more. Due to this BCD multipliers are slower and this design leads to more regular VLSI implementation and does not require special registers for storing easy multiples.

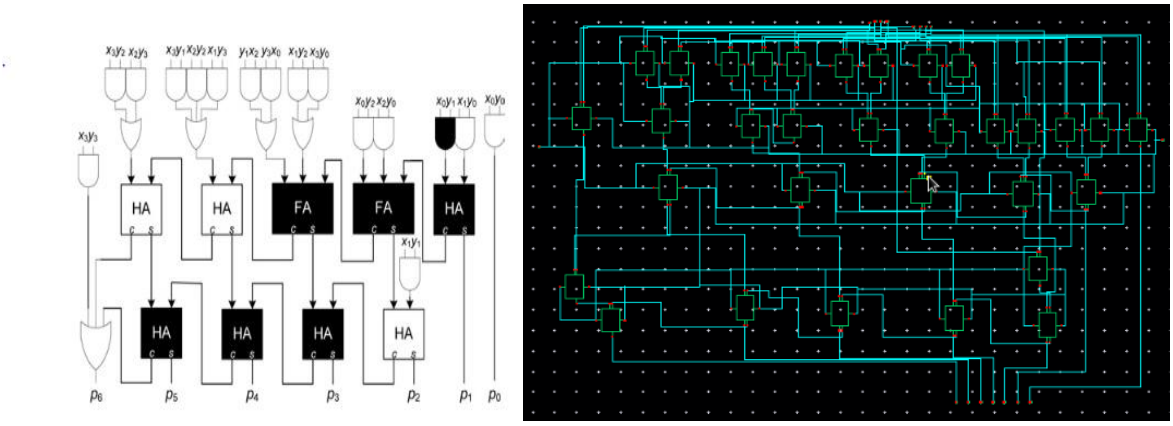

Fig.1 BCD multiplier Diagram

III. Implementation

Decimal multiplier with sign magnitude encoding; In existing method, BCD multipliers have more delay so to overcome this we preferred sign magnitude encoding in decimal multipliers.Fig.2depicts the general architecture of the proposed 4×4 multiplication P=X×Y; the details of each building block will be explained later. In particular, in the top three blocks, the multiplier's digits are recoded to n one-hot [−5,5]SMSDs (i.e., one sign and five magnitude bits), augmented with a 10 n-weighted carry bit. The multiples[0,5]×X are precomputed as n[−6,6] SMSDs and a 10 n-weighted[−5,4] SMSD. Each SMSD contains a sign bits and 3-b magnitude. The negative multiples[1,5]×(−X) are achieved via dynamic sign inversion of multiples[1,5]×X at the cost of only one XOR gate per digit. This design mainly consists an five operations as follows;

Recoding of multiplier digits; In this sign magnitude encoding the multiplier digit recoding in the form [-5,5] with X multiples in the form of [0,5]*X it includes only one hard multiple i.e,3X.

Precomputed multiples; Here we generate pre-computed multiple set in which we get hard multiple 3X in carry-free manner.

Partial product generation; Partial product can be generated by performing an multiplication of multiplicands.

Partial product reduction; Partial product reduction was performed by introducing an reduction tree.

Final product computation; It is the final stage of decimal multiplication in which we perform and final product computation.

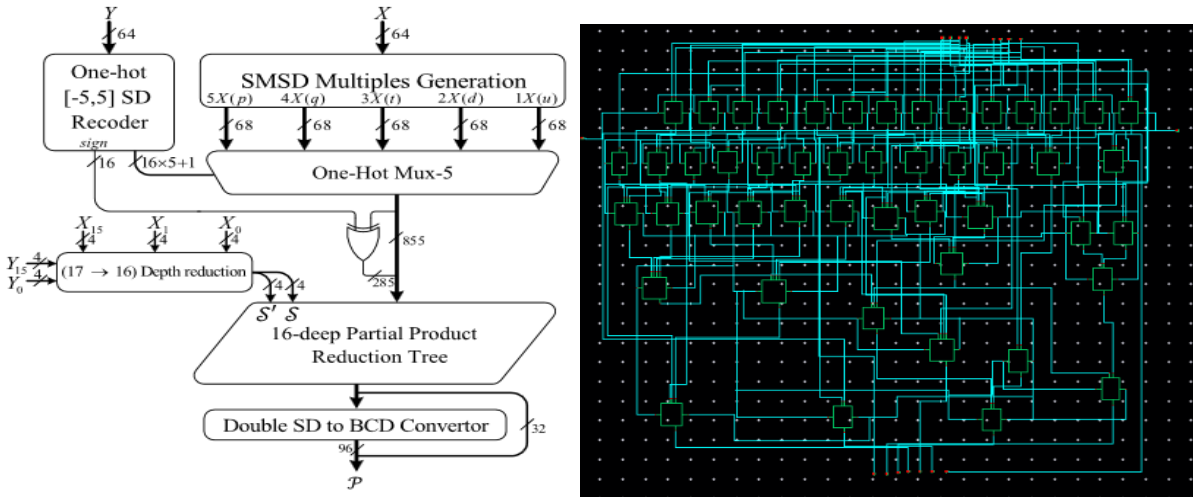

Fig.2 Decimal multiplier with sign magnitude encoding.

IV. Experimental Result

In this section we compare BCD decimal multiplier and Decimal multiplier with sign magnitude encoding, the below figure shows the wave form result of BCD decimal multiplier and Decimal multiplier with sign magnitude encoding. In fig.3 and fig.4 we consider an inputs x0x1x2x3x4,y0y1y2y3 and outputs p0p1p2p2p3p4p5p6.we can take one example i.e multiplying an two decimal 4bit numbers, 2(0010)*3(0011) the output will be 6(0110).while concentrating on design architecture the number transistors will be less as compare to conventional design due to this delay of the proposed design will be less so the speed will be more for this proposed design.

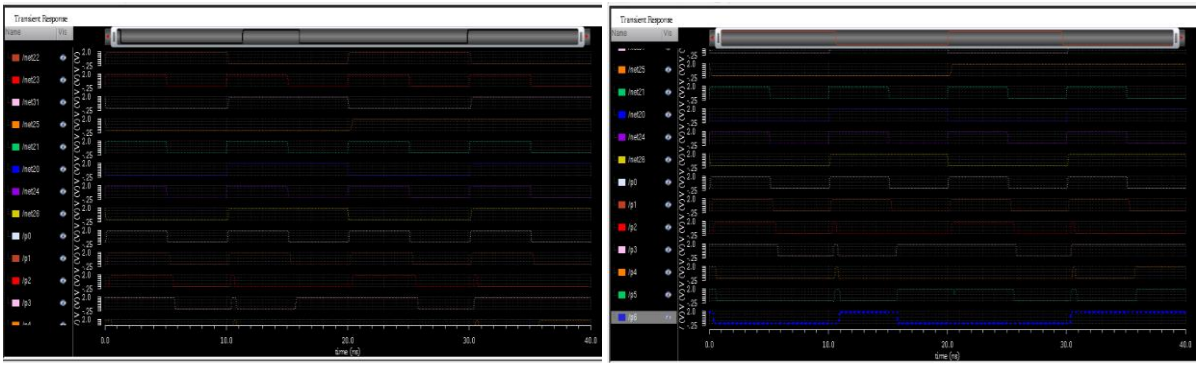

Fig. 3 output waveforms of BCD multiplier.

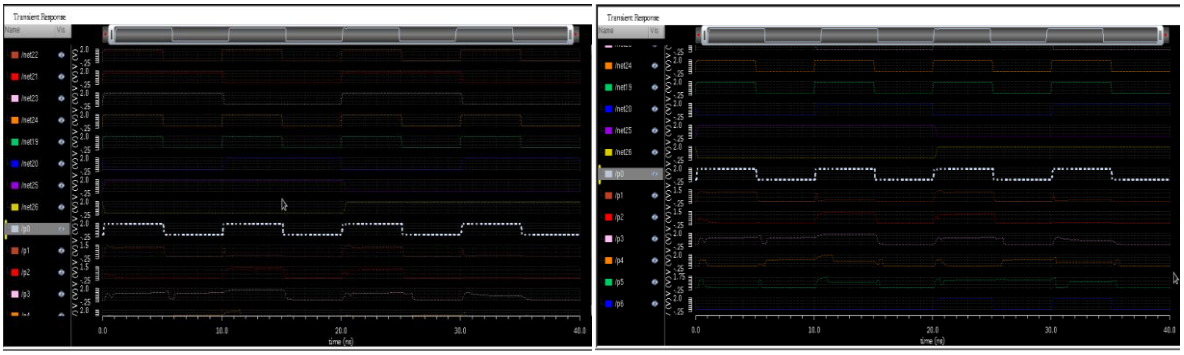

Fig.4 output wave forms of sign magnitude encoded decimal multiplier.

Comparison Performance Parameters for BCD multipliers and Decimal multiplier with sign magnitude encoding

Parameters'	BCD multiplier	Decimal multiplier with sign magnitude encoding
1)Power(uW)	744.1	353.4
2)Delay(ps)	428.38	176.76

V.CONCLUSION

We propose a parallel 4×4 radix-10 Decimal multiplier, where partial products are generated with SMSD representation. Some innovations of this paper and use of previous techniques, have led to less power dissipation. The least possible delay for the latter is 4.8 ns, while the proposed design leads the synthesis tool to meet the 4.4-ns time constraint (i.e., 9% faster). In other words, the advantage is that the proposed design can operate in 9% higher frequency and dissipates less power.

1)SMSD Representation: The exclusively employed SMSD representation of partial products saves more than 850 XOR gates (≈75%) in comparison with other 4×4 decimal multipliers with dynamic negation of partial products.

2)On the Fly Depth Reduction: Two SMSD digits of the sole deepest column of partial product matrix are reduced to one, in parallel with PPG that leads the VLSI-regular 2:1 PPR to start with 16 partial products.

3)4-in-1 SMSD-to-TCSD Adder: This is the most novel contribution of this paper. The reason is that sign magnitude addition conceptually entails separate consideration of four sign combinations. To avoid the corresponding in efficiency, the first-level reduction is undertaken via eight special SMSD adders. However, enforcing the SMSD signs via polarity of magnitudes has led to a unified 4-in-1 adder logic, which is no more complex than a simple TCSD adder.

4)Early Initiation of Redundant-to-BCD Conversion: To take advantage of early signal arrivals, conversion of the four least significant digits to BCD starts in the middle of PPR. A parallel prefix compound KS adder produces BCD sum and sum1 for the nine most significant digits.

5) Parallel Prefix Carry Select Addition: A special parallel prefix decimal carry select adder adds up the middle TCSD digits and produces BCD sum digits and a borrow that selects one of the two BCD sums of the most significant part.

References

1) R. D. Kenney, M. J. Schulte, and M. A. Erle, "A high-frequency decimal multiplier," in Proc. IEEE Int. Conf. Comput. Design (ICCD), Oct. 2004.
2) G. Jaberipur and A. Kaivani, "Binary-coded decimal digit multipliers,"IET Comput. Digit.Techn, vol. 1, 2007.
3) G. Jaberipur and A. Kaivani, "Improving the speed of parallel decimal multiplication,"IEEE Trans. Comput, vol. 58, no. 11, Nov.2009.
4) L. Dadda and A. Nannarelli, "A variant of a radix-10 combinational multiplier," in Proc. IEEE In Symp. Circuits Syst. (ISCAS), May 2008,
5) T. Lang and A. Nannarelli, "A radix-10 combinational multiplier," in Proc. 40th Asilomar Conf. Signals, Syst., Comput., Oct./Nov. 2006.

