

DESIGN OF A LOW AREA TRIANGULAR DIVISION CIRCUIT FOR SD2 POSITIVE INTEGERS

S.Radha^{#1}, S.Kannappan^{#2}, M.Mahesh babu^{#3}

#1+919703143168, #2+919160215302, #3 +919490197645

ABSTRACT

Division is the highest latency arithmetic operation in present digital architectures and high-performance computing systems; as such drives the demand for efficient hardware division units. Accordingly, this paper proposes a novel architecture for a non restoring divisor based on the radix-2 signed-digit (SD2) representation. This notation has been chosen to achieve fast computation, but the architecture presented in this paper, due to its structure and the definition of the cell implementing its architecture, saves area as well. The proposed divisor architecture is able to achieve a delay of order n instead of other solutions that give $O(n^2)$ growth. Our cells are larger than those proposed in literature, considering them as single circuit, but considering the overall structure there is a saving of some 40% in the number of gates and a gain of 55% in terms of power saving when compared with the state of the art.

Key words: Integer division, nonrestoring division, radix-2 signed-digit (SD2).

INTRODUCTION

DIVISION is the most complex of the four basic arithmetic operations and, consequently, the most time-consuming. Its very large scale integration (VLSI) implementation is generally slower and more area consuming than those of the other three basic arithmetic operations. Even considering this complexity in current complex digital systems, achieving an efficient hardware unit becomes essential.

Division is regarded as a slow operation, characterized by a delay of $O(n^2)$. Radix-2 digit-recurrence algorithms, both restoring and nonrestoring, are widely used to perform integer division. Our paper will focus on the definition of a novel nonrestoring division unit that exhibits all the advantages of the radix-2 signed-digit (SD2) representation of the partial remainders, but involves a triangular architecture that leads to a saving in the required area. The main novel contributions for this work are:

- definition of novel compact architecture for a nonrestoring divisor based on the SD2 representation for the partial remainder.
- triangular hardware implementation of a divider that meets time performance while reducing area requirements.

PROBLEM DESCRIPTION

The fundamental division operation is defined by (1), where the dividend N and the divisor D (with $D \neq 0$), the quotient Q and the remainder R are all n -bit positive values. $N=QD+R$.

The quotient is obtained via a sequence of subtractions and shifts. In the generic step j of this sequence, the intermediate remainder R_j is compared with the divisor D . If $R_j > D$, the quotient bit q_j is set to 1, otherwise to 0. A scheme that can be adopted to this situation is the non restoring division Algorithm. In the nonrestoring method, we avoid the restoring operation, temporarily retaining a negative remainder.

SD2 notation will lead to interesting results in a nonrestoring context. SD2 notation is used to implement a nonrestoring floating point divider. The approach is interesting and proves the benefits in terms of computational speed. The architecture proposed is characterized by the fact that all the carries have the same direction, thus obtaining a processing time of $O(n)$ instead of $O(n^2)$.

THE BASE ARCHITECTURE

In the present architecture all the carries have the same direction, resulting in a processing time of $O(n)$ instead of $O(n^2)$. The final remainder is represented using the SD2 representation. This solution represents the classical architecture of the division unit defined as a rectangular array of n rows and $2n$ columns. The n^2 cells, each composed of the absc, hc and sum circuits, characterizing the architecture can be grouped into four triangles.

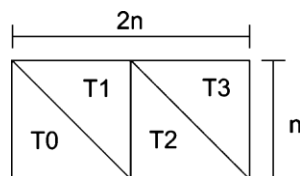


Fig.1. Original architecture of a division unit with N , Q , D and R defined each with n -bit.

A. SD2 Sum Circuits

Every elementary cell in Fig. 1 is composed of the following circuits: a circuit hc , a sum circuit, and an $absc$ circuit. The former one accepts as inputs two signals $abss$, which can have one of the following values $(-1, 0, +1)$ from the output of the upper level, and the $-d$ signal which can be 0 or 1. It also produces an h vertical output and a horizontal carry c .

TABLE 1

SUM ($abss-d$), THE HORIZONTAL OUTPUT (CARRY c) AND THE VERTICAL OUTPUT(h) OF THE CIRCUIT.

$abss-d$	-2	-1	0	+1
h	0	+1	0	+1
c	-1	-1	0	0

B. The absc Circuit

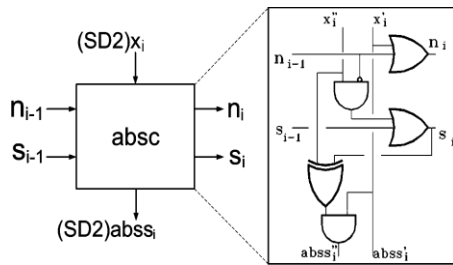


Fig.2. Absolute value and sign detecting circuit.

The absc circuits accept as input three signals: x_i , $null_{i-1}$ and $sign_{i-1}$. Both, $null_{i-1}$ and $sign_{i-1}$ are 1-b values, while x_i is represented using the SD2 representation. The same notation is used for the three outputs, $null_i$ and $sign_i$ are 1-b values while $abss_i$ is represented using SD2 notation.

PROPOSED ARCHITECTURE

It presents an architecture of a divisor divided into two main circuits: a triangle, right circuit (triangle T2 in Fig. 1), and a rectangle, left circuit (triangles T0 and T1 in Fig.1). In order to delete the left circuit we must do the following.

- Remove the signals going from the right circuit to the left one.
- Compute the value of all the input signals of the absc circuits in the right part of the divisor.

FINAL ARCHITECTURE

This paper presents a triangular architecture for a non restoring division without the need of introducing comparison operations. The idea was to compute the subtraction and to drive the corresponding MUX enable input with its sign. Working with SD2 notation leads to interesting results in a non restoring context. In this architecture where all the carries have the same direction, obtaining a processing time of $O(n)$ instead of $O(n^2)$ and it is based on the radix-2 non restoring division algorithm. The SD2 representation was used for representing partial remainders. Each quotient digit is directly obtained from the sign of the corresponding partial remainder.

This paper proposes a novel architecture for a nonrestoring divisor based on the SD2 representation, used to perform integer division. A radix-2 SD2 notation has been chosen to achieve fast computation, but the architecture presented in this paper, due to its triangular structure and to the definition of the cell implementing its architecture, allows a saving in area as well. The cells used to implement the proposed architecture are bigger than the others proposed in literature, when taking them as a single circuit, but the overall structure can save 40% in number of gates.

CONCLUSION

The divisor architecture presented is a novel architecture for a nonrestoring divisor based on the SD2 notation. The radix-2 notation was chosen to achieve fast computation presenting a triangular architecture that will lead to a saving in area. In such architectures, the computation of the absolute value of each partial remainder is performed concurrently with the sign detection of the partial remainder and may start before the completion of the sign detection of the preceding partial remainder. In this paper, we present a divisor architecture able to achieve a delay of an n -order, instead of the $O(n^2)$, but characterized by a more compact architecture.

Our cells are bigger than the others, taking them as single circuit, but the overall structure can save 40% in number of gates as shown in Fig.3, where the x-axis represents the number of bits while the y-axis represents the ratio between our solution and the existing one.

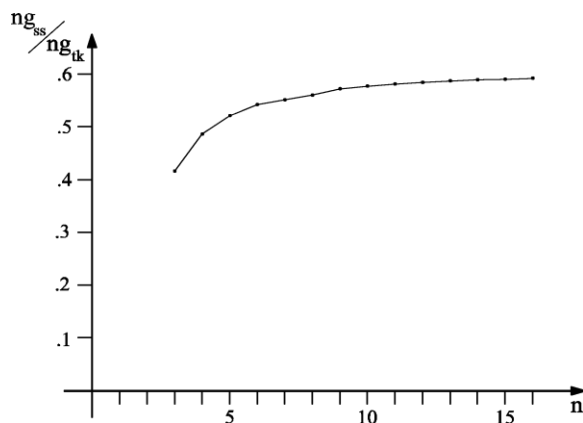


Fig.3. Number of gates comparison between the proposed architecture and the existing one.

The x-axis(n) represents the number of bits while in the y-axis, ng_{ss} is the number of gates in the present solution and ng_{ik} represents the number of gates in the previous one.

REFERENCES

- [1] A. Avizienis, "Signed-digit number representations for fast parallel arithmetic," *Trans. Electron. Comput.*, vol. EC-10, no. 3, pp. 389–400, Sep. 1961.
- [2] N. Takagi, S. Kadowaki, and K. Takagi, "A hardware algorithm for integer division using the sd2 representation," *IEICE Trans. Fundam. Electron. Commun. Comput. Sci.*, vol. E89-A, no. 10, pp. 2874–2881, 2006.

[3] M. D. Ercegovac and T. Lang, *Division an Square Root: Digit-Recurrence Algorithms and Implementations*. Norwell, MA: Kluwer, 1994 .

[4] S. Kadowaki, "A hardware algorithm for integer division," in *Proc.17th IEEE Symp. Comput. Arithmetic*, Washington, DC, 2005, pp.140–146.