On Improvement of Stefanelli's Division Algorithm by Converting the Dividend to the Redundant Form

A. I. NOZIK and A. A. SHOSTAK

Belorussian Polytechnic Institute

This paper was originally published in *Izvestia Vuzov, Priborostroenie (Proceedings of Universities, Instrument Making Series), vol. 33, no. 12, pp. 21-25, 1990 (in Russian)*

***Abstract*— A number of modifications of Stefanelli's division and reciprocal algorithms is suggested. These modifications are based on the redundant representation of the dividend and allow for both an increase to the speed and a decrease to the hardware cost of division and reciprocal devices based on Stefanelli's algorithm.**

Stefanelli's division algorithm [1] is based on the idea of using the redundant set of the allowed quotient digits. The calculation of the binary quotient $Q = C / A$ using Stefanelli's division algorithm consists of two steps. First, the quotient is formulated as a binary number:

$$Q = q_0 2^0 + q_1 2^{-1} + q_2 2^{-2} + \ldots + q_{m-1} 2^{-(m-1)}$$

using the redundant binary digits q_k $(0 \leq k \leq m-1)$. The redundant digits q_k assume either positive or negative integer values to satisfy the following system of algebraic equations:

$$\begin{aligned}
q_0 &= 1 \\
q_1 &= c_2 - a_2 q_0 \\
q_2 &= c_3 - a_2 q_1 - a_3 q_0 \\
&\ldots \\
q_{m-1} &= c_m - a_2 q_{m-2} - a_3 q_{m-3} - \ldots - a_m q_0. \qquad (1)
\end{aligned}$$

Equations (1) are obtained by representing the multiplication $A \cdot Q$ as a partial product array and assigning the sums of the partial product array elements with equal binary weights to the corresponding binary digits of the dividend C. Divisor $A = 0.a_1 a_2 \ldots a_n$ and dividend $C = 0.c_1 c_2 \ldots c_n$ are assumed to be positive and normalized binary fractions. To calculate the quotient with the precision equal to the precision of A and C, the number m of the redundant quotient digits has to be greater than the number n of binary digits of the dividend and the divisor. The binary digits $c_{n+1}, c_{n+2}, \ldots c_{m-1}$ of the dividend and the binary digits $a_{n+1}, a_{n+2}, \ldots a_{m-1}$ of the divisor in (1) are assumed to be equal to zero.

In the second step, the quotient formulated in the redundant form is converted to the non-redundant binary notation $Q = q_0^{*}.\, q_1^{*}\, q_2^{*} \ldots q_{n-1}^{*}$, where $q_l^{*} \in \{0,1\}$ and $0 \le l \le n-1$.

This paper presents the modifications of Stefanelli's division and reciprocal algorithms that allow both an improvement to the performance and a decrease in the hardware cost of the respective devices. The improvement is achieved by decreasing the allowed ranges of the redundant quotient digits and decreasing of the quotient's error. (The ranges of the redundant quotient digits formulated using Stefanelli's algorithm have been determined theoretically in [2]). Also, an additional improvement has been achieved by modifying the basic structure of the respective devices for the division and the reciprocal algorithm.

The suggested modifications of the division and reciprocal algorithms are based on the idea of converting the dividend into the redundant form preserving its algebraic value. For instance, the conversion can be performed using the following equivalency for even values of *n:*

$$C = C^{*} = 0.\, c_1\, c_2 \ldots c_{n-1}\, c_n = c_1\, 2^{-1} + c_2\, 2^{-2} + \ldots + c_{n-1}\, 2^{-(n-1)} + c_n\, 2^{-n} =$$

$$= (2c_1 + c_2)\, 2^{-2} + (2c_3 + c_4)\, 2^{-4} + \ldots + (2c_{n-1} + c_n)\, 2^{-n}\, .$$

Based on the above, dividend C can be represented in the following form: $C = C^{*} = 0.\ 0\ c_2^{*}\ 0\ c_4^{*} \ldots 0\ c_n^{*}$, where $c_2^{*} = 2c_1 + c_2$, $c_4^{*} = 2c_3 + c_4$, …, $c_n^{*} = 2c_{n-1} + c_n$. A similar conversion can be performed for odd values of n as well. The values of the redundant digits q_k of quotient Q are determined from the equations obtained by assigning the sums of the partial product array elements with equal binary weights to the corresponding binary digits of the dividend C^{*} in the redundant binary form.

Obviously, there are multiple ways of converting the dividend into the redundant form. For example, the dividend can be converted to the redundant form using the following equivalency: $C = C^{*} = 0.\ 0\ 0\ c_3^{*}\ 0\ 0\ c_6^{*} \ldots 0\ 0\ c_n^{*}$, where $c_3^{*} = 4c_1 + 2c_2 + c_3$, $c_6^{*} = 4c_4 + 2c_5 + c_6$, …, $c_n^{*} = 4c_{n-2} + 2c_{n-1} + c_n$.

Computer simulation was used to find one of the possible conversion algorithms of the dividend into the redundant form that allows significant reduction in both the range of the redundant quotient digits and the quotient's error. According to the found conversion algorithm, the dividend is converted to the redundant form using the following equivalency: $C = C^{*} = 0.\ c_1\, c_2\ 0\ c_4^{*}\ 0\ c_6^{*} \ldots 0\ c_n^{*}$, where $c_4^{*} = 2c_3 + c_4$, $c_6 = 2c_5 + c_6$, …, $c_n^{*} = 2c_{n-1} + c_n$. Table 1 presents the maximum absolute values of the redundant quotient digits calculated using the dividend converted to the redundant form according to the above mentioned conversion rule.

Table 1

	q_k	q_0	q_1	q_2	q_3	q_4	q_5	q_6	q_7	Q_8	q_9
$max\ \vert q_k\vert$	C	1	1	2	3	5	8	13	21	34	55
	C^*	1	1	1	4	5	7	9	13	21	34

Note: $C = 0.\ c_1\, c_2\, c_3\, c_4 \ldots c_n$ *and* $C^* = 0.\ c_1\, c_2\, 0\, c_4^*\, 0\, c_6^* \ldots 0\, c_n^*$.

Table 2 presents the scaled maximum values of the quotient's error Δ_{max} in terms of the units of least precision for the case of $m=n$.

Table 2

	N	5	6	7	8	9
Δ_{max} x 2^n	$C = 0.\ c_1\, c_2\, c_3\, c_4 \ldots c_n$	7	11	19	31	51
	$C^* = 0.\ c_1\, c_2\, 0\, c_4^*\, 0\, c_6^* \ldots 0\, c_n^*$	5	10	12	19	31

For the cases where $n < 5$, the suggested approach does not have the advantage over Stefanelli's unmodified division algorithm. For this reason, the quotient's error, for those values of n, are not presented in Table 2.

A similar approach can be used to modify the calculation of the reciprocal algorithm $Q= 1/A$ of divisor A. The modifications of the reciprocal algorithm are based on the redundant representation of the dividend C *when* equal to 1. For example, $1 = 2{\cdot}2^{-1} = 0.\ 200 \ldots 0$ or $1 = 1 \cdot 2^{-1} + 2{\cdot}2^{-2} = 0.\ 120\ldots0$ or $1 = 4{\cdot}2^{-2} = 0.\ 040\ldots0$ or $1 = 1{\cdot}2^{-1} + 4{\cdot}2^{-3} = 0.\ 1040\ldots0$ etc.

These redundant binary representations of *1* shall be called *1-codes.* In this case, the values of $c_2, c_3, \ldots c_m$ in (1) are substituted with the respective redundant binary digits of the particular 1-code. The computer simulation proved that both the ranges of the redundant digits of the reciprocal algorithm and its error depend on the particular 1-code used with (1).

One of 1-codes which provides for the decrease both in the ranges of the redundant digits of the reciprocal algorithm and its error is 1-code $0.1111030303\ldots0304 = 1$. Table 3 presents the maximum absolute values of the redundant digits of the reciprocal algorithm for that 1-code. Table 4 presents the scaled values of the maximum absolute error of the reciprocal algorithm in terms of the units of the least precision for the case of $m=n$. For comparison, these tables also present the respective data for Stefanelli's unmodified reciprocal algorithm which uses the dividend $C = 0.1111\ldots1$ as an approximation of 1 for the case of the reciprocal algorithm calculation.

Table 3

	q_k	q_0	q_1	q_2	q_3	q_4	q_5	q_6	q_7	q_8
$max\ \lvert q_k \rvert$	*1 = 0. 111…1*	1	1	1	1	2	3	5	8	13
	1 = 0. 11110303…0304	1	1	1	1	2	4	6	9	11

Table 3 (cont.)

	q_k	q_9	q_{10}	q_{11}	q_{12}	q_{13}	q_{14}	q_{15}	q_{16}	q_{17}
$max\ \lvert q_k \rvert$	*1 = 0. 111…1*	21	34	55	89	144	233	377	610	987
	1 = 0. 11110303…0304	16	22	28	39	59	78	106	141	201

When $n<9$, 1-code *0.1111030303…0304* does not have advantages over Stefanelli's unmodified algorithm with $C =$ *0.1111…1*. For this reason, the respective data for $n<9$ is not presented in Table 4.

Table 4

	N	9	10	11	12	13	14	15	16	17
$\Delta_{max} \times 2^n$	*1 = 0. 111…1*	19	31	51	81	133	214	348	561	958
	1 = 0. 11110303…0304	16	24	24	39	48	78	98	142	160

Consider the following example of using the described modification of Stefanelli's division algorithm for the case of $m=n=6$. Fig. 1 presents the divider in a form of a triangle cellular array consisting of multipliers *M*, subtractors *S*, and converter *C* used to convert the redundant binary code into the non-redundant binary code.

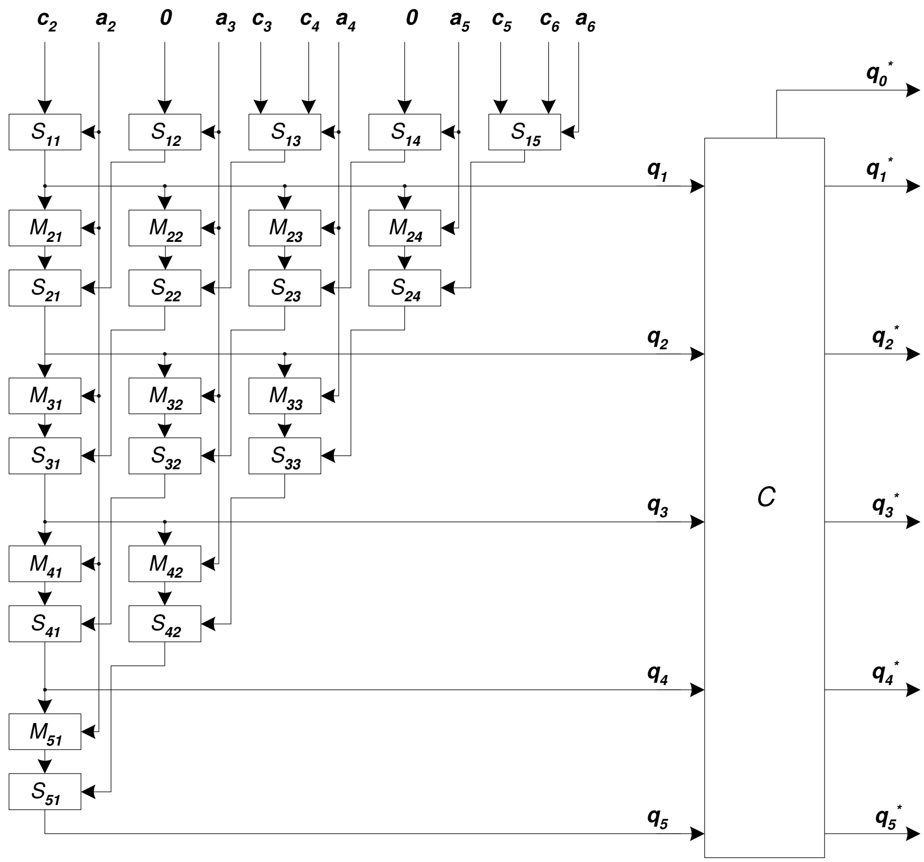

Fig. 1

The redundant quotient digits in this device are formulated based on the following system of algebraic equations:

$$q_0 = 1$$
$$q_1 = c_2 - a_2$$
$$q_2 = 0 - a_3 - a_2q_1$$
$$q_3 = 2c_3 + c_4 - a_4 - a_3q_1 - a_2q_2$$
$$q_4 = 0 - a_5 - a_4q_1 - a_3q_2 - a_2q_3$$
$$q_5 = 2c_5 + c_6 - a_6 - a_5q_1 - a_4q_2 - a_3q_3 - a_2q_4$$

The redundant digit q_1 is formulated on the output of subtractor S_{11} which subtracts the value of a_2 from the value of c_2. The redundant digit q_2 is formulated on the output of subtractor S_{21} which subtracts the value of a_2q_1, formulated on the output of M_{21} , from the value of $0\text{-}a_3$ formulated on the output of subtractor S_{12} . The remaining quotient digits q_3, q_4, and q_5 are formulated in a similar manner. The converter C converts the quotient from the redundant binary code to the non-redundant binary code $Q = q_0^* . q_1^* q_2^* q_3^* q_4^* q_5^*$. Note that the redundant quotient digit q_0 is always equal to one and therefore is not formulated explicitly. However, its value is taken into account by the converter C of the redundant binary code to the non-redundant binary code.

The time latency required to calculate the quotient by both the divider considered above and the known device [1] is determined by the following expression:

$$\sum_{k=1}^{m-1} t_{q_k} + t_c ,$$

where t_{q_k} is the time latency required to formulate the redundant digit q_k after the redundant digit q_{k-1} has been formulated, and t_c is the time latency of the converter C. For the known divider [1], the time latency t_{q_k} is determined by the time latency of three sub-modules: multiplier, adder, and subtractor. At the same time, for the divider considered above, the time latency t_{q_k} is determined only by the time latency of the multiplier and the subtractor. The fact that the considered device has smaller ranges of the redundant quotient digits, allows for the decrease also in the time latency of the converter C compared to the respective time latency of the converter for the known divider [1]. Also, an additional decrease of the time latency for the considered device is achieved because of the smaller number of the redundant quotient digits q_k required to obtain the non-redundant binary quotient with the required precision. For large values of n, the suggested divider allows the ability to produce the quotient twice as fast as with the known divider [1].

In [3] and [4], the dividers based on the redundant representation of the dividend, are considered in detail. The modified reciprocal device is considered in detail in [5].

The suggested devices for reciprocal and division algorithms require significant amounts of hardware and therefore represent a practical interest for the implementations based on LSI and VLSI technologies.

REFERENCES

[1] R. Stefanelli, "A suggestion for a high-speed parallel binary divider," *IEEE Trans. Comput.,* vol. C-21, no. 1, pp. 42-55, Jan. 1972.

[2] A. Nozik and A. Shostak, "On determining the ranges of the redundant quotient digits of Stefanelli's division algorithm," *Voprosy Radioelektroniki, seria EVT (*Problems of Radio-Electronics, Scientific-Technical Collection. Computer Equipment Series), no. 14, 1985, pp. 32-37 (in Russian).

[3] A. Nozik and A. Shostak, "Dividing device," Avtroskoe svidetelstvo SSSR no. 1231498 (Patent USSR SU1231498), Bulletin Izobretenii, vol. 18, 1986 (in Russian).

[4] A. Nozik and A. Shostak, "Dividing device," Avtroskoe svidetelstvo SSSR no. 1332312 (Patent USSR SU1332312), Bulletin Izobretenii, vol. 31, 1987 (in Russian).

[5] A. Nozik and A.Shostak, "Device for determining inverse value," Avtorskoe svidetelstvo SSSR no. 1196853 (Patent USSR SU1196853), Bulletin Izobretenii, vol. 45, 1985 (in Russian).