Comments

Comments on "High-Speed Area-Efficient Multiplier Design Using Multiple-Valued Current-Mode Circuits"

Behrooz Parhami

Abstract—Kawahito et al present multiplier designs using the binary-tree reduction feature of certain highly redundant radix-2 representations, along with multiple-valued current-mode circuit techniques, and show them to compare favorably to those based on less redundant binary signed-digit and carry-save numbers. We point out that these representation schemes, and their potential advantages, have been discussed in earlier publications and that a more general view of the parallel-carries addition process exploited in these multipliers leads to other potentially useful representations.

Index Terms—Binary signed-digit, carry-save, redundant number systems, stored-carry, stored-double-carry, stored-triple-carry, tree multipliers.

1 Introduction

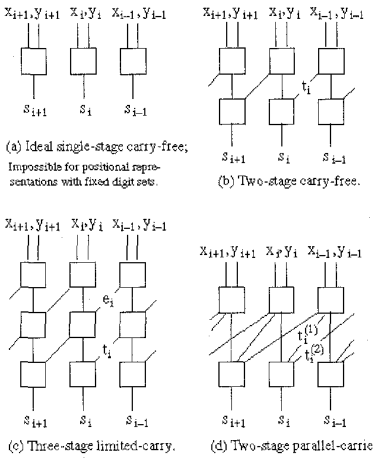
A class of area-efficient parallel tree multipliers based on unsigned-digit redundant radix-r number representations having digit sets of the form $[0, q]$, with $q \geq r$, have recently been presented by Kawahito et al [1]. (Notation: $[a, b] = \{a, a+1, \ldots, b\}$). The digit sets [0, 3] and [0, 4] for $r = 2$ were given special attention and the resulting 24 × 24-bit multiplier designs were shown to compare favorably to known implementations based on binary signed-digit and binary stored-carry (carry-save) representations using the digit sets $[-1, 1]$ and $[0, 2]$, respectively, with regard to delay, transistor count, and number of interconnections. These advantages are gained primarily due to the fact that computation of digit sums and incorporation of transfers are performed by relatively simple current-summing logic associated with multiple-valued current-mode circuits.

The purpose of this comment is to point out that:

1) These number systems are instances of generalized signed-digit (GSD) representations introduced in [3], and further developed in [5], [6]. Discussions in these papers include the special GSD subclass with unsigned digits.
2) Both of the "new" radix-2 redundant representations corresponding to $q = 3$ and $q = 4$, and their advantages in designing regular binary-tree multipliers, have been previously published by this author [4].[1]
3) Taking a more general view of the GSD addition process exploited in these multipliers leads to other potentially useful representation systems that may offer certain advantages (though not necessarily with multiple-valued current-mode circuits).

2 GSD Numbers and Their Addition

A radix-r GSD number system having the digit set $[-\alpha, \beta]$, with the redundancy index $\rho = \alpha + \beta + 1 - r > 0$, is a positional number system in which the digit vector $x_{k-1}x_{k-2} \ldots x_1x_0$ represents the integer $\sum_{i=0}^{k-1} r^i x_i$. Any GSD system with $r > 2$ and $\rho > 2$, or with $r > 2$ and $\rho = 2$ provided that $\alpha \neq 1$ and $\beta \neq 1$, supports "carry-free" addition in which the transfer digit t_i goes from position i to position $i + 1$ and is absorbed there [3], [5]. Thus the sum digit s_i is a function of four operand digits x_i, y_i, x_{i-1}, and y_{i-1} as shown in Fig. 1b.



(a) Ideal single-stage carry-free; Impossible for positional representations with fixed digit sets.

(b) Two-stage carry-free.

(c) Three-stage limited-carry.

(d) Two-stage parallel-carrie

Fig. 1. The ideal carry-free scheme and some practical implementation alternatives for GSD addition.

In all other cases, namely when $r = 2$, $\rho = 1$, or $\rho = 2$ with $\alpha = 1$ or $\beta = 1$, a three-stage process, dubbed "limited-carry" addition, must be used [3], [5]. One implementation alternative is to send a two-valued "estimate" e_i from position $i - 1$ to position i informing this latter position whether the transfer t_i will be in the "high" or in the "low" subrange and enabling it to generate a suitable outgoing transfer t_{i+1} in a manner that would guarantee its own ability to absorb the incoming t_i. Thus, in these cases the sum digit s_i is a function of x_i, y_i, x_{i-1}, y_{i-1}, x_{i-2}, and y_{i-2} as shown in Fig. 1c. A related alternative, which is practical only for small r, is to replace e_i with t'_i, which has the same range as t_i and allows the sharing of hardware between two of the stages or, alternatively, a more regular design with non-shared hardware.

[1] One of the reviewers of this comment pointed out that Kawahito et al. Did not claim novelty for the number representation schemes used and that their primary contribution was the use of current-summing technique. Reviewers also pointed out that Kawahito et al. had presented their multiplication algorithm in an IEICE technical report, dated June 1988. However, in Section II of the paper under discussion, where the number systems and their associated carry-free and limited-carry addition processes are presented, there is no reference to earlier work on unsigned-digit redundant numbers or their use in designing high-speed multipliers; even the authors' own technical report is neither cired not listed among the references. Such a detailed discussion, along with lack of reference citations, is normally understood to cover original work being reported for the first time. This interpretation is reinforced by the claim in the second sentence of the paper's "Concluding Remarks."

• *The author is with the Department of Electrical and Computer Engineering, University of California, Santa Barbara, CA 93106-9560.*
E-mail: parhami@ece.ucsb.edu.

Manuscript received Mar. 15, 1994; revised Sept. 16, 1994.
For information on obtaining reprints of this article, please send e-mail to: transcom@computer.org, and reference IEEECS Log Number C95129.

0018-9340/96$05.00 ©1996 IEEE

Another approach, depicted in Fig. 1d, is to feed some information $t_i^{(2)}$ directly from position $i-2$ to position i. In this scheme, each position i sends two transfers $t_{i+1}^{(1)}$ and $t_{i+2}^{(2)}$ to positions $i+1$ and $i+2$, respectively.

The above summary shows that the new number systems proposed in [1] are instances of GSD numbers discussed in [3], [5], [6], where unsigned-digit redundant representations with digit set $[0, \beta]$, and their special case of stored-double-carry numbers with $\beta = r + 1$, leading to the digit set [0, 3] for $r = 2$, have been explicitly defined.

Two-stage carry-free addition of radix-2 numbers using the digit set [0, 3] (stored-double-carry numbers) or [0, 4] (stored-triple-carry numbers) was first discussed in [4]. It was shown that with transfers $t_{i+2}^{(2)}$ and $t_{i+1}^{(1)}$ from Stage i restricted to [0, 1], the basic addition equation

$$x_i + y_i + t_i^{(2)} + t_i^{(1)} = 4t_{i+2}^{(2)} + 2t_{i+1}^{(1)} + s_i$$

dictates that the parameter β of the digit set $[0, \beta]$ satisfy $\beta \geq 3$ and $\beta \leq 4$. Thus the conclusion that only the digit sets [0, 3] and [0, 4] are feasible for $r = 2$.

Advantages of these two new number representations and their associated two-stage, parallel-carries addition process, with different encodings for the digit set, were discussed in detail, with no assumption about the implementation technology. Fig. 3 in [4], depicting a 24-bit binary-tree multiplier, is quite similar to Fig. 7 in [1].

3 A More General View

The use of parallel carries can be generalized to include more than two receiving positions, non-binary transfers, and nonuniform transfer digit sets. A complete presentation of the analyses pertaining to such a generalization is beyond the scope of this comment. Denoting the transfer going from position $i-j$ to position i by $t_i^{(j)}$,with $1 \leq j \leq h$ and $t_i^{(j)} \in [-\lambda_j, \mu_j]$, one can derive, from the basic addition equation

$$x_i + y_i + \sum_{j=1}^{h} t_i^{(j)} = \sum_{j=1}^{h} r^h t_{i+h}^{(j)} + s_i$$

and the various ranges assumed, the relevant constraints to be met. In the special case of uniform transfers, i.e., $\lambda_j = \lambda$ and $\mu_j = \mu$ $(1 \leq j \leq h)$, the following constraints can be obtained:

$$h \geq \left\lceil \frac{\alpha(r-1)}{h(r^2-r+1)-r} \right\rceil$$

$$\mu \geq \left\lceil \frac{\beta(r-1)}{h(r^2-r+1)-r} \right\rceil$$

$$r - 1 \leq \lambda + \mu \leq \left\lfloor \frac{\rho}{h} \right\rfloor$$

Extensive experimentation with the values of the different parameters involved have shown that such uniform transfer digit sets are unlikely to lead to practically viable number representations for $r > 2$. For $r = 2$, the above conditions become:

$$\lambda \geq \left\lceil \frac{\alpha}{3h-2} \right\rceil$$

$$\mu \geq \left\lceil \frac{\beta}{3h-2} \right\rceil$$

$$1 \leq \lambda + \mu \leq \left\lfloor \frac{\alpha + \beta - 1}{h} \right\rfloor$$

These conditions clearly show the impossibility of two-stage carry-free addition ($h = 1$) as in Fig. 1b for radix 2, lead to the two number systems discussed earlier for $\alpha = 0$ and $h = 2$ allowing two-stage parallel-carries addition as in Fig. 1d, and suggest that the scheme is likely to be impractical for $h > 2$.

However, there is no compelling reason for uniformity in transfer digit sets (see [2] for an illuminating discussion of digit sets). One possibility is to have transfers that alternate in sign. For the sake of simplicity, assume $\lambda_j = \gamma \times (j \bmod 2)$ and $\mu_j = \gamma \times (1 - j \bmod 2)$, leading to $\lambda_j + \mu_j = \gamma$. Then, each position i receives $\lfloor h/2 \rfloor$ positive and $\lceil h/2 \rceil$ negative transfers, each up to γ in magnitude. Hence the interim sum must satisfy $-\alpha + \gamma \lceil h/2 \rceil \leq w_i \leq \beta - \gamma \lfloor h/2 \rfloor$ if all transfers are to be absorbed without exceeding the allowed digit range $[-\alpha, \beta]$. Since w_i should assume at least r different values, a necessary condition is:

$$\beta + \alpha - h\gamma + 1 \geq r \quad \text{or} \quad \rho \geq h\gamma$$

Radices $r > 2$ impose additional restrictions, but for $r = 2$ and $\gamma = 1$, digit sets such [–1, 2] with $h = 2$ (corresponding to the stored-carry-or-borrow representation [3]) and [–3, 1] with $h = 3$, as well as their mirror images [–2, 1], [–1, 3] obtained by letting $\mu_j = \gamma \times (j \bmod 2)$, are quite practical. Such digit sets involve smaller maximum magnitudes than the unsigned versions, thus leading to potential simplifications in various arithmetic operations such as multiplication, division, and square-rooting.

References

[1] S. Kawahito, M. Ishida, T. Nakamura, M. Kameyama, and T. Higuchi, "High-Speed Area-Efficient Multiplier Design Using Multiple-Valued Current-Mode Circuits," *IEEE Trans. Computers*, vol. 43, no. 1, pp. 34-42, Jan. 1994.

[2] P. Kornerup, "Digit-Set Conversions: Generalizations and Applications," *IEEE Trans. Computers*, vol. 43, no. 5, pp. 622-629, May 1994.

[3] B. Parhami, "A General Theory of Carry-Free and Limited-Carry Computer Arithmetic," *Proc. Canadian Conf. VLSI*, Winnipeg, Canada, pp. 167-172, Oct. 1987.

[4] B. Parhami, "A New Method for Designing Highly Parallel Binary Multipliers," *Proc. Third Ann. Parallel Processing Symp.*, pp. 176-185, Fullerton, Calif., Mar. 1989.

[5] B. Parhami, "Generalized Signed-Digit Number Systems: A Unifying Framework for Redundant Number Representations," *IEEE Trans. Computers*, vol. 39, no. 1, pp. 89-98, Jan. 1990.

[6] B. Parhami, "On the Implementation of Arithmetic Support Functions for Generalized Signed-Digit Number Systems," *IEEE Trans. Computers*, vol. 42, no. 3, pp. 379-384, Mar. 1993.