A Parallel Decimal Multiplier Based on a novel 4:2 Fully Redundant Decimal Adder

Zewei Yao, Xiaoping Cui, Tao Wang

College of Electronic and Information Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing, Jiangsu 211106, China
yaozweii@nuaa.edu.cn

Abstract. Due to requirements for computational accuracy in the fields of business computing, the decimal arithmetic computing system has gradually become a research hotspot. A parallel decimal multiplication consists of three stages: partial product generation (PPG), partial product reduction (PPR) and the final product generation (FPG). In the PPR stage, a novel 4:2 fully redundant decimal adder based overloaded decimal digit set (ODDS) is proposed in this paper. The 4:2 ODDS adder is formed by the binary 4:2 compressors, a decimal compressor and a converter. Moreover, the signed-digit radix-10 code, the redundant excess-3 code and the ODDS code are used in the PPG stage. In the FPG stage, an ODDS–BCD conversion is adopted to quickly generate BCD-8421 product. The analysis and comparison by using the 45nm technology show that the proposed decimal multiplier based on the 4:2 ODDS adder has less area and better synthesis performance.

1. Introduction

Fast processing of decimal data in currency, web-based, and human interaction application requires decimal arithmetic hardware. The multiplier is one of the important components of the CPU to process data, and is also one of the core components of the Digital Signal Processor (DSP). A draft IEEE standard for floating-point arithmetic (P754/D2.50) is proposed in 2019[1], the IBM Power 6 microprocessor [2], z/System family of microprocessors [3] and the Fujitsu Sparc X microprocessor [4] already include decimal floating-point arithmetic units.

The decimal multiplier is mainly composed of three modules: partial product generation (PPG), partial product reduction (PPR) and final product generation. The PPR module compresses all partial products into two partial product rows. The two kinds of partial product compression modules based on BCD-4221/5211 coding and ODDS coding are widely used. The decimal carry save adder (CSA) based on BCD-4221/5211 coding is used to compress the partial product [5], [6], [7] and the 4-bit CSA structure is regular, which effectively improves the performance of the multiplier. The advantage of ODDS coding is that there is no need to correct six invalid 4-bit numbers in binary PPR tree, and its PPR tree can be used to realize fast and efficient decimal multiplier. Reference [9] proposed a 2:1 fully redundant ODDS adder (also defined as compressor). The two input operands and the resultant sum of the decimal ODDS adder are all redundant decimal numbers based on ODDS codes. A 2:1 reduction ratio is obtained with the ODDS adder and the disadvantage is that it costs more circuit resources.

The proposed fully redundant 4:2 ODDS adder is formed by the binary 4:2 compressors, a special decimal compressor and a converter. It can be effectively used in decimal PPR tree and is conducive to large-scale implementation.
2. The 4:2 Fully Redundant ODDS Adder

A fully redundant 4:2 ODDS adder consists of the 4-bit binary 4:2 compressors, a special decimal compressor and a converter. As shown in Fig.1, the 4-bit binary 4:2 compressor is composed of four binary compressors. The $a_i (a_i = a_i^3 a_i^2 a_i^1 a_i^0), b_i (b_i = b_i^3 b_i^2 b_i^1 b_i^0), c_i (c_i = c_i^3 c_i^2 c_i^1 c_i^0)$ and $d_i (d_i = d_i^3 d_i^2 d_i^1 d_i^0)$ are the four inputs of the 4:2 ODDS adder at $i$-digit column. The four binary 4:2 compressors are used in the adder to compress the 4-bit ODDS input to $S_i^3, S_i^2, S_i^1, S_i^0, C_2_i^3, C_2_i^2, C_2_i^1, C_2_i^0$ and carry $C_1_i^1$. The $C_1_i^1$ and $C_2_i^1$ with weight of 16 are divided into $Co_1_i^3,L_i^1$ and $Co_2_i^3,P_i^1$ respectively, where $Co_1_i^1$ and $Co_2_i^1$ with weight of 10 are carried to the $(i+1)$-digit. The $L_i^1$ and $P_i^1$ with weight of 6 are transferred to the special decimal compressor (compressor-A and compressor-B). In compressor-A, $L_i^1$ and $P_i^1$ are added by a half adder to get sum $B_i^1$ (bit weigh is 6) and a carry $D_i^1$ (with weigh of 12). The weight of carry $D_i^1$ is split into $Co_3_i$ with weigh 10 and $Co_4_i$ with weigh 2, in which 10 is transferred to the $(i+1)$-digit. The bit weigh of $S_i^0, C_1_i^0$ and $D_i^0$ is 2. The bit weigh of $S_i^0, Co_3_i-1$ and $Co_2_i-1$ is 1. As shown in Fig.2, the six numbers are added by the two full adders to obtain $B_1^1$ and $B_0^1$. The $B_i^1, B_i^3, B_i^2, B_i^1$ and $B_i^0$ are compressed to ODDS number $(F_i^3, F_i^2, F_i^1$ and $F_i^0)$ by compressor-B. The expression in compressor-B is shown in Formula (1) and its circuit diagram is shown in Figure 3.

$$F_i^1 = B_i^3 \cdot (B_i^3 B_i^2 B_i^1) + B_i^1 \cdot (B_i^1 + B_i^2 + B_i^1)$$
$$F_i^2 = B_i^3 B_i^2 + B_i^1 B_i^1 + B_i^2 B_i^1 \cdot (B_i^1 + B_i^1) + (B_i^1 B_i^2 + B_i^4 B_i^1 + B_i^2 B_i^1) \cdot B_i^1 + B_i^1$$
$$F_i^1 = B_i^4 + B_i^3 + B_i^1$$
$$F_0 = B_0^0$$

The feature of the converter is that the input operands ($\in [0,2,4]$) in i-digit is converted to an ODDS number with bit weight of 10,8,4,2. The 10 is transferred to the lowest bit of $(i+1)$-digit decimal number. The special converter converts $S_i^3, S_i^2, C_1_i^2, C_1_i^1$ (with weigths of 8,8,4,4) to $H_i^0, H_i^3, H_i^2, H_i^1$ (with weights of 10,8,4,2). $H_0^0$ of i-digit is transferred to the lowest bit of $(i+1)$-digit decimal number, the $H_0$ of $(i-1)$-digit is transferred to i-digit. The expression of the converter is shown in Formula (2) and its circuit diagram is shown in Figure 4.
3. Evaluation and comparison

In this paper, the 16×16-digit decimal multiplier adopts signed-digit radix-10 code, redundant XS-3 coding and ODDS coding to quickly generate 17 rows of ODDS partial product [6], [8]. The 4:2 full redundancy ODDS adder is applied to PPR stage to reduce partial product to two PP rows. The 2:1 ODDS adder [10] is used to reduce two PP rows to one PP. The ODDS-BCD converter in [10] is used in FPG stage.

The synthesis results based on Design Compiler are given under NanGate Open Cell 45nm Library. Three standard inverters of 4X strength are used for the output load and a standard AND of 4X strength is used for the input drivers. Under the same compression ratio, the 4:2 ODDS adder consists of two 2:1 ODDS adder in reference [9]. The proposed 4:2 ODDS adder is compared with two 2:1 ODDS adders. The delay and area for two 4:2 ODDS adders are given in Table 1. Under the condition of only increasing the delay of 10.7%, the area of 4:2 ODDS adder is reduced by 46.3%, and the comprehensive performance is greatly improved.

| 4:2 ODDS adder     | Delay (ns) | Ratio | Area (nm²) | Ratio |
|-------------------|------------|-------|------------|-------|
| The 4:2 ODDS adder in Ref. [9] | 0.56       | 0.903 | 221.84     | 1.862 |
| Proposed 4:2     | 0.62       | 1.000 | 119.17     | 1.000 |

As shown in Table 2, the proposed 16×16 multiplier increases the delay by 7.8% and reduces area by 30.7% compared with [9].
TABLE 2. Delay and Area Comparison of 16×16-digit (Using NanGate 45nm Open Cell Library)

| Scheme     | Delay (ns) | Ratio | Area ($\mu m^2$) | Ratio |
|------------|------------|-------|-----------------|-------|
| 16×16-digit|            |       |                 |       |
| Ref.[9]    | 3.18       | 0.927 | 0.0646          | 1.442 |
| Proposed   | 3.43       | 1.000 | 0.0448          | 1.000 |

4. Conclusion
A 16×16-digit redundant decimal multiplier based on a novel fully redundant 4:2 ODDS adder has been proposed in this paper. The proposed 4:2 ODDS adder is composed by four regular binary 4:2 compressors, a special decimal compressor and a converter. It is used to compress decimal partial products. The synthesis results from the Synopsys Design Compiler with NanGate 45nm Open Cell Library show that the proposed decimal multiplier significantly reduces the area of the circuit and has better synthesis performance.

References
[1] IEEE, Std 754(TM)-2019, IEEE Draft Standard for Floating-Point Arithmetic[S]: IEEE, 2019.
[2] L. Eisen, J. Ward, H. Tast, et al. IBM POWER6 Accelerators: VMX and DFU[J]. IBM Journal of Research & Development, 2007, 51(6):1-21.
[3] F. Busaba, C. Krygowski, W. Li, et al. The IBM Z900 Decimal Arithmetic Unit[C]. Conference Record of the Asilomar Conference on Signals, Systems and Computers, Pacific Grove, CA, USA, 2001, 1335-1339.
[4] T. Yoshida, T. Maruyama, Y. Akizuki, et al. Sparc64 X: Fujitsu's New-Generation 16-Core Processor for Unix Servers[J]. IEEE Micro, 2013, 33(6):16-24.
[5] A. Vazquez, E. Antelo, and P. Montuschi. Improved Design of High-Performance Parallel Decimal Multipliers. IEEE Transactions on Computers, 2010, 59(5): 679-693.
[6] X. Cui, W. Dong, W. Liu, et al. “High Performance Parallel Decimal Multipliers using Hybrid BCD Codes,” IEEE Transactions on Computers, 2017, 66(12):1994-2004.
[7] A. Vazquez, E. Antelo, and P. Montuschi. A new family of high-performance parallel decimal multipliers. Montepellier, France, IEEE, 2007, 195–204.
[8] A. Vazquez, E. Antelo, and J. Bruguera. Fast Radix-10 Multiplication Using Redundant BCD codes. IEEE Transactions on Computers, 2014, 63(8):1902-1914.
[9] S. Gorgin and G. Jaberipur. A Fully Redundant Decimal Adder and Its Application in Parallel Decimal Multipliers. Microelectronics Journal, 2009, 40(10): 1471-1481.
[10] L. Zhang, X. Cui and W. Dong. High-Performance Parallel Fully Redundant Decimal Multiplier [J]. Acta Electronica Sinica, 2018, 46(6) : 1519-1523.