VLSI Implementation of a Floating-point Divider

Jasbir N. Patel, Z. Abid and Wei Wang
Department of Electrical and Computer Engineering
University of Western Ontario, London, ON, Canada.
jpatel44@uwo.ca; zeabid@eng.uwo.ca; wwang@eng.uwo.ca

Abstract

In this paper, we present the VLSI implementation of a low power floating-point divider in CMOS 0.18µm technology using radix-2 over redundant number system. This divider implementation is well suited for IEEE 754 floating point standard and can be widely used in DSP applications. In the proposed divider designs, different PPM adders, based on 24, 22 and new 16-transistor circuits are used to implement the carry-free addition/subtraction unit. Different designs of quotient selection logic and reducing unit are also presented. Then, these designs are compared for power dissipation, time delay, and area. Our new design of a 4-bit divider has 9% less EDP and 17% less area than previous work of [2].

1. Introduction

Digital Signal Processing (DSP) has applications in many areas including efficient computational blocks. Low power dissipation and reduced area are some of the major concerns in implantable and portable electronics used in DSP systems. Divider is one of the important arithmetic blocks in DSP application. Floating point division algorithms for high speed VLSI employ signed digit number systems [1] because of its carry-free property. There are several algorithms for floating-point divider using different radix redundant number representation [6][7][8]. These algorithms use quotient selection scheme using different number of digits.

A popular radix-2 division algorithm with over redundant quotient selection logic was presented [2]. In this algorithm [2], the quotient selection logic was designed using the 2 most-significant digits of partial reminder. The quotient generated is in over-redundant radix-2 representation. The over redundant quotient is then converted in normally redundant quotient digits for on-the-fly conversion to binary number system. A Major contributor to the cycle time for divider is the quotient digit selection function logic [2]. Therefore, we can modify quotient selection logic to further reduce the hardware without compromising the speed. Similar circuit level modifications are also done in the reducing unit. PPM adder/subtractor is used to design addition/subtraction unit that calculates partial reminder in every cycle. Different PPM designs are implemented using 24, 22 and new 16-transistor circuits [4] [2] [5].

2. Divider Algorithm and Structure

In this section, we first review the divider algorithm [2]. Then, we discuss the divider design and its structure.

2.1. Divider algorithm

The divider algorithm described in [2] consists of three main blocks which are quotient selection logic, reducing unit and carry free adder/subtractor unit. The divider algorithm starts with the dividend being loaded in the parallel $n+1$ digit ($2n+2$ bits) register beginning from least-significant bit in least-significant bit position. Therefore, the most-significant digit in the register is 0. In all the iterations, partial reminder is stored in the same register. In the algorithm of [2], the authors presented a two digit quotient selection function using partial reminder therefore quotient selection logic is independent of the divisor. The quotient selection function selects quotient in over-redundant radix-2 form from normally redundant radix-2 partial reminder. This extracted quotient should be converted to binary form. There is a popular on-the-fly converter design to convert normal radix-2 redundant to binary form. Therefore, the quotient extracted in over-redundant radix-2 form should be converted to normally redundant radix-2 form. This conversion is done using the reducing unit. More detailed description of the quotient selection unit and the reducing unit is given in [2]. After generation of the quotient in radix-2 form, the carry-free adder/subtractor unit adds or subtracts the divisor (or left shifted divisor) from partial reminder depending on the magnitude and

sign of quotient. This partial reminder is then left shifted and loaded to the register again for the next iteration.

After n-iterations for n-bit divider, the quotient and final reminder are available. A restoration step is then performed to restore quotient and reminder value according to IEEE 754 standard. The structure of the divider [2] is shown in section 2.2.

2.2. Structure of the divider

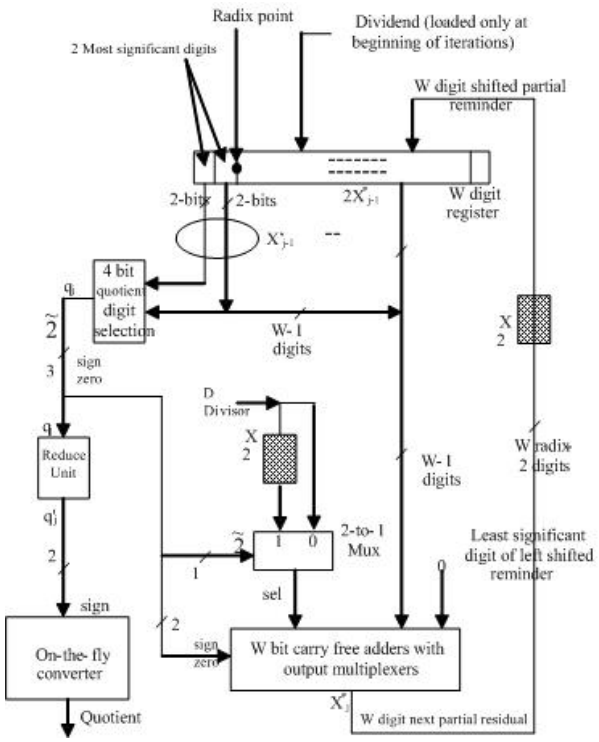

Figure 1: Divider architecture [2]

The overall architecture of the divider [2] is shown in Fig. 1. There are three main blocks, as described in previous sub-section. The other blocks, including left shift blocks, registers and multiplexers, are standard digital blocks. The quotient selection logic, reducing unit and carry-free adder/subtractor with output multiplexer according to [2] are shown in Fig. 2, Fig. 3 and Fig. 5 respectively. From Fig. 2, we can see the quotient selection unit generate output in over-redundant radix-2 (-2, -1, 0, 1, 2) form. This over-redundant radix-2 is converted to redundant radix-2 (-1, 0, 1) form by the reducing unit of Fig. 3. Fig. 5 shows the adder/subtractor block of n+1 digit carry-free adder/subtractor unit. From Fig. 4 it is clear that the 22-transistor PPM circuit is the main block of adder or subtractor unit of Fig. 5.

Design 1 - Previous design of the Divider

This design is implemented as in [2]. The quotient selection logic unit, reducing unit and carry-free adder unit are shown in Figures 2, 3, 4, and 5:

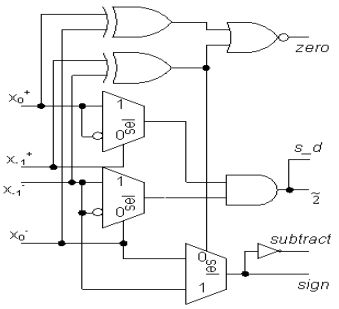

Figure 2: Quotient selection logic Unit [2]

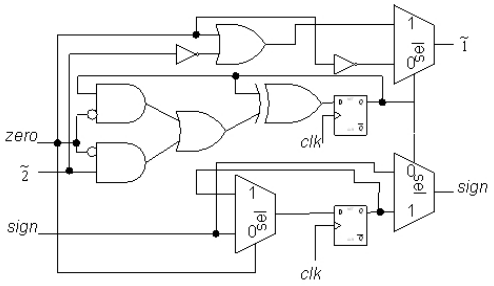

Figure 3: Reducing unit [2]

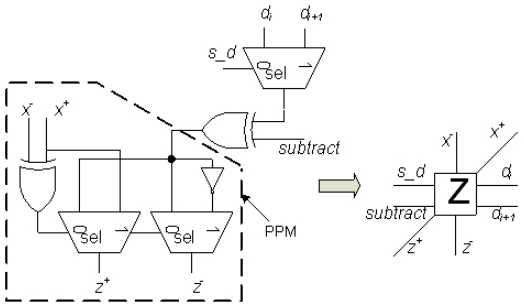

Figure 4: Carry-free addition/subtraction Unit [2]

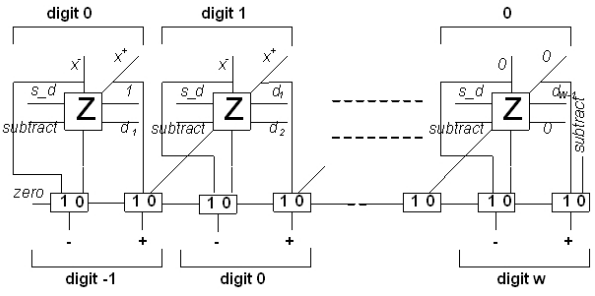

Figure 5: Carry-free addition/subtraction unit with output multiplexers [2]

3. Proposed three designs of the Divider

In this section, we present three new designs of the divider with modifications in quotient selection logic and reducing unit blocks. We also present different options of PPM adders. The three new designs are presented as:

- Design-2 :- The quotient selection logic and reducing unit are improved. The carry-free adder block is the same as in [2]
- Design-3 :- This Design has improved quotient selection logic and reducing unit with standard 24 transistor PPM based carry-free adder block
- Design-4 :- A new 16 transistor PPM circuit is used in the carry-free adder block. The quotient selection logic and reducing unit are improved.

The overall divider architecture is same as in Fig. 1.

3.1. Design 2 of the Divider

In this design, we modified the quotient selection logic unit and reducing unit of the previous design of [2] to reduce power and area. The new designs are shown in Figures 6 and 7 respectively.

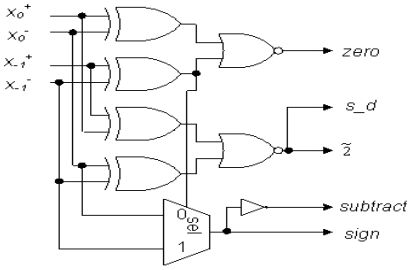

Figure 6: A new design of the quotient selection unit

The new quotient selection logic is designed with 4-XOR, 2-NOR, 1-MUX and 1-INVERTER gates. These results in less number of transistors than in [2] and consequently reduce the area. The XOR gate shown in the Fig. 6 can be implemented using standard 8-transistor CMOS logic, or 8-transistor pass-logic, or 6-transistors pass logic [3]. Similarly, MUX gate can be implemented using standard CMOS logic or pass-logic. NOR and INVERTER gates are designed using standard CMOS logic.

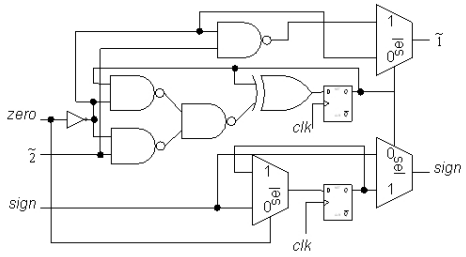

Figure 7: A new design of the reducing unit

A new reducing unit is similarly designed with 4-NAND, 1-XOR, 3-MUX, 2-D-FF and 1-INVERTER gates. Similar pass-logic approach is used. In this design, the carry-free adder/subtractor block consists of 22-transistor PPM adder as in [2].

3.2. Design 3 of the Divider

In this design the quotient selection logic unit and reducing unit are as shown in Fig. 6 and Fig. 7 respectively. The 24-transistor standard CMOS PPM adder [4] is used.

3.3. Design 4 of the Divider

The quotient selection logic and reducing unit are shown in Fig. 6 and Fig. 7. A new 16-transistor PPM adder [5] is used to design carry-free adder/subtractor block with multiplexers. The new design of 16-transistor PPM adder is shown in Fig. 8.

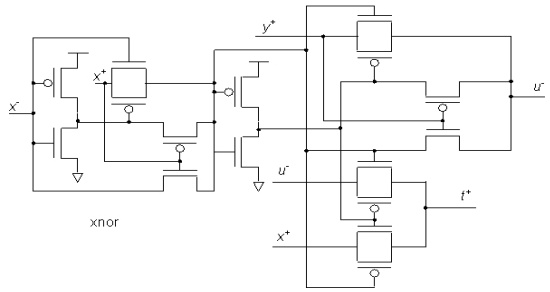

Figure 8: Transistor level diagram of the 16 transistor PPM adder [5]

4. VLSI implementation of the Divider

All the designs of a 4-bit divider were implemented in CMOS 0.18µm technology in Virtuoso Schematic Editor. The sizes of transistors are selected for equal rise and fall time with minimum size possible. Input dividend, divisor and output reminder are in parallel form. The output quotient is in serial form. The quotient is available without any clock period latency. The quotient is in sign and magnitude form of radix-2 redundant representation. For 4-bit divider, a total of five clock cycles are required. In four clock cycles, the 4-digit quotient is generated and an extra clock cycle is required for restoration of quotient and reminder as mentioned in [2]. Simulations are done to measure power-dissipation and time-delay using SpectreS Analog Environment simulation tool available in Cadence Tools. The total area for the quotient selection logic unit, the reducing unit and the carry-free adder/subtractor unit are used for

comparison purpose as the other blocks are similar for all the four designs.

Power-dissipation, time-delay, energy-delay product and area are compared for all four designs. The results are shown in Table-1, Table-2, Table-3 and Table-4 respectively. T_{QS} is the number of transistors in quotient selection logic, T_{RU} is the number of transistors in reducing unit and T_{ASU} is the number of transistors in the adder/subtractor unit. The total width of transistors in all the designs are provided in multiples of W, where W is 220nm.

Table 1: Simulation results of power dissipation

Design	Power	
	in µW	%
Design 1 [2]	238.30	100.00
Design 2	215.50	90.43
Design 3	225.60	94.67
Design 4	213.10	89.42

Table 2: Simulation results of time delay

Design	Delay	
	in pSec.	%
Design 1 [2]	727.6	100
Design 2	735.6	101.1
Design 3	774.2	106.4
Design 4	733.7	100.84

Table 3: Energy-delay product (EDP)

Design	EDP	
	x 10^{-21}	%
Design 1 [2]	0.12616	100
Design 2	0.11661	92.39
Design 3	0.13522	107.13
Design 4	0.11472	90.89

Table 4: Total number and width of transistors in quotient selection logic, reducing unit and add/subtract block

Design	$T_{QS}+T_{RU}+T_{ASU}$	Total Width	
	Total	Width	%
Design 1 [2]	282	340W	100.00
Design 2	260	311W	91.47
Design 3	270	521W	153.24
Design 4	230	281W	82.65

It is clear from Tables 1 to 4 that design-4 (with new quotient selection logic, reducing unit and a new 16-transistor PPM adder) has superior performance than the other three designs including the previous design [2]. Design-4 has 9% saving in EDP and 17% saving in area compared to the previous work [2].

Moreover, by replacing quotient selection unit and reducing unit with new designs of relevant blocks (namely, Design-2), we can save around 8% in EDP and 8% in area than the previous work [2]. For design-3, power dissipation is reduced by 5% but time delay increased by 6% resulting in 7% higher EDP than the previous work [2]. Design-3 has less number of transistors but ends up with higher area than the previous work [2]. This is due to higher area of the 24-transistor PPM adder implementation.

5. Conclusion

In this paper, we presented the VLSI implementation of a low power floating-point divider in CMOS 0.18µm technology using radix-2 over redundant number system. This divider implementation is well suited for IEEE 754 floating point standard and can be widely used in DSP applications. In the proposed divider designs, different PPM adders, based on 24, 22 and new 16-transistor circuits are used to implement the carry-free addition/subtraction unit. Different designs of quotient selection logic and reducing unit are also presented. Based on the implementation results, we conclude that one of the proposed designs, namely, Design 4, uses a new method of quotient selection logic and a new 16-transistor PPM adder, resulting in 9% saving in EDP product and 17% saving in area compared to the previous design.

6. References

[1] A. Avizienis, "Signed Digit Number Respresentation for Fast Parellel Arithmetic," *IRE Trans. Electronic computers*, vol. 10, pp. 389-400, Sept. 1961.

[2] H. R. Srinivas, K. K. Parhi and L. A. Montalvo, "Radix 2 Division with Over-Redundant Quotient Selection", *IEEE Transactions on Computers*, Vol. 46, No. 1, pp. 85-92, Jan. 1997.

[3] K. Cheng and C. Huang, "The novel efficient design of XOR/XNOR function for adder applications", *Electronics, Circuits and Systems, 1999. Proceedings of ICECS '99. The 6th IEEE International Conference*, Volume: 1, Pages:29 – 32, 5-8 Sept. 1999.

[4] A. Guyot, Y. Herreros and J. Muller, "JANUS, an On-line Multiplier/Divider for manipulating large numbers", *in Proceedings of 9th Symposium on Computer Arithmetic*, pp 106-111, 1989.

[5] Y. Yang, W. Wang and Z. Abid, "Novel PPM Adder Designs and VLSI Implementation", *Submitted to TCAS – II*. journal, 2004

[6] S. Kuninobu, T. Nishiyama, H. Edamatsu, T. Tanaguchi, and N. Takagi, "Design of High Speed MOS Multiplier and Divider Using Redundant Binary Representation," *Proc. Eight Symp. Computer Arithmetic*, pp. 80-86, Como, Italy, 1987.

[7] A. Vandemeulebroecke, E. Vanzieleghem, T. Denayer, and P.G.A. Jespers, "A New Carry-Free Division Algorithm and Its Application to a Single-Chip 1024-b RSA Processor," *IEEE J. Solid State Circuits*, vol. 25, pp. 748-756, June 1990.

[8] M.D. Ercegovac and T. Lang, *Division and Square Root.* Norwell, Mass.: Kluwer Academic, 1994.