Multi_radix Complex Division with Operand Rounding

Youssef Abdul-Aziz Taher
Information Technology Department,
Faculty of Computer and Information
Technology, Al-Razi University, Yemen - Sana'a

ABSTRACT

Complex Division is one of the most popular operations used in the field of complex numbers operations. Improving systems and applications functionality using non-decimal radix required extra processing time. Therefore, complex division attracts the attention of researchers and manufacturers aiming at improving their systems and applications. This paper proposed a new approach to reduce large pre-scaling tables required for radices greater than 4. The proposed approach dedicated to cope with this problem to enhance power consumption, area, and speed needed to perform this type of operations.

General Terms

Complex numbers, radices

Keywords

Pre-scaling, complex, post-scaling, radices, non-decimal, dividend, remainder, quotient

1. INTRODUCTION

Complex numbers are one of the most topics attracting researcher's attention these days. Complex numbers consist of real and imaginary parts that must be handled carefully and accurately, especially when using with arithmetic operations of different radices. Arithmetic may varied between addition, subtraction, multiplication, and division. Complex operations are widely used in the modern science, technology, and industry, especially during the rapid development in wireless communication systems. The importance of complex operations comes because of the modern digital renaissances, which need an accurate, tiny, and fast hardware architecture. Fig 1 shows the implementation of a complex number on a plane. If z is a complex number, then z must be

z=X+Yi, where X is the real part and Y is the imaginary part, and i denotes for the imaginary unit.

Division for complex numbers could be obtained easily using the following formula [1].

The preceding equation leads to overflows during the intermediate calculations, exactly when calculating c2+d2. In addition, using floating point to represent results leads to overflow. This problem has been discussed by many researchers such as [2, 3, and 4]. The proposed approaches in [5],[6] considered as serial and fixed which affect the flexibility and the system execution time.

Pre-scaling simplify the selection of the real and imaginary parts of the complex digits. It could be obtained using a scalar factor to compute the scaled divisor and the scaled dividend. This operation needs more requirements, which has a direct effect on the system size. More requirements such as additional Rom or look up tables to cope with the requirements of the required radix may affect the system execution time.

2. RELATED WORK

As mentioned before, the complex division with pre-scaling is a hot research topic these days. Many researchers has worked on achieving a faster operation on different radices, especially with high radices. The proposed model in [5] considered a sequential design, which can led to extra delay especially with high radix operations.

Kevin Brady Hil in [6]. Proposed a fixed radix model and prove its efficiency with radix 4 division.

The proposed models n [7][8][9] designed to cope with multi task operations such as square root, multiplication, and division which led to tradeoff between the functionality of the system and system size. The proposed algorithm in [10] introduced algorithm for complex division; his algorithm is derived from the real digit recurrence algorithm and uses the pre-scaling of the operands to allow selection by rounding. The recurrences in the algorithm are suitable for the high radix. On the other hand, the algorithm is more complicated and uses a table size considered as the limit factor of the choice of the radix, in addition, implementing this algorithm has a higher cost than other algorithms.

3. PROPOSED APPROACH

The proposed approach is equipped to handle complex numbers with 32_bits by five modules, as shown in Fig 2, operate in parallel to get the maximum possible speed. This model consists of six multipliers, two adders, subtractor, and two divisors. All of these units work simultaneously to reach another unit equipped to round or post-scaling and convert results into the required radix number system. The proposed model also dedicated to handle complex division for different radix with high speed compared to other models.

The system size also taken into account during the implementation. The proposed model display the result sequentially through four outputs ports represents the real and fraction of each part of the complex number. As shown in Fig 2, pipelining technique carried out to separate calculations to achieve the lowest possible latency value. Therefore, the proposed model perform the operation on five levels. It use six 32_bits multiplication units working in parallel in the first level.

The second level consists of two 32 bits adder and one subtactor to prepare results to be passed through the division units.

Unlike addition, subtraction and multiplication, division does not produce an exact answer because the dividend is not necessarily a multiply of the divisor. To make design more flexible, the proposed model divide the results into two parts, quotient and remainder. The proposed model can adjust the accuracy of the results according to the requirements of the application what make the design more flexible. Therefore in the third level "sequential restoring division" the proposed model was carried out to cube with the results of the previous level.

In the proposed model, the 2n dividend is loaded into both halves of shift register and add a sign bit to the left. Then it adds a sign bit to the left of the divisor and generate the 2's complement of the divisor then shift to the left. After that, it adds the 2's complement of the divisor to the upper half of the shift register including sign bit (subtract). Then it checks the state if sign of the result is cleared (positive), it sets LSB of the lower half of the shift register to one. Else, it clears the LSB of the lower half and add the divisor to upper half of shift register (restore). After repeating this loop n times (shift left an add two's complement), it terminate the operation and load the quotient to the lower half of shift register and load the remainder to the upper half of shift register. Therefore, the proposed model ensures less waste of hardware. On the other hand, it takes the advantage of the time to perform operations in parallel with other operations. Finally, in the last level, the proposed model dedicates a unit to convert the results to the desired root and bring the results to the nearest fractional values. The simulation results showed the efficiency of the proposed model in terms of dealing with any radix required depending on the application requirements at high speed and acceptable size.

4. SIMULATION RESULT

Fig 3 shows the result of dividing two complex numbers (a+ib)/(c+id) in the two systems decimal within the digit set [0 to 9] and radix4 [0 to 3] and display the results represented as value and the fraction. This model is equipped to carry out the division of two complex numbers with size of 32-bits. As also shown in the Fig, the proposed model is dedicated to round results to the nearest 3 fraction digits to provide the maximum amount of space required to save the results. Simulation results describe the results of dividing two complex numbers in the radix 4 and radix 10 number system. The proposed model can change the value of n to perform the division in the required number system. As mentioned earlier, the proposed model designed to provide as little as possible space and the largest possible execution speed. This model works at a frequency of 310 MHz devices virtex 7 and needs only 575 to slice out of and 1,424,000 and 1.2 watts.

5. FUTURE WORK

Design on air high-speed processing unit in the beginning of the calculation process. Specifically, in the first levels of the design instead of the LUT in the pre-scaling models to make the selection of quotient digits simple. This leads to a simple hardware implementation, and allows correct rounding of the complex quotient.

6. CONCLUSION

This paper introduced a new model for complex division. The proposed model used the Sequential Restoring Division algorithm together with unit equipped to round the results and turn them into the required conventional number system depending on the requirements of the application. The cost of implementing of complex numbers is twice the cost of dividing the real numbers in terms of area and speed. Therefore, the proposed model used many techniques to reduce the cost and increase the speed as much as possible. The proposed model used parallel processing and pipelining techniques to prove higher speed and less area in comparison with other complex division approaches.

7. REFERENCES

[1] A Cariow, "An algorithm for dividing two complex numbers", arXiv preprint, 2016.

[2] Anila Ann Varghese, C Pradeep, Madhuri Elsa Eapen, "FPGA Implementation of Area-Efficient IEEE 754 Complex Divider", 2016.

[3] Aleksandr Cariow, G Cariowa, " Hardware-Efficient Structure of Complex Numbers Divider", Measurement Automation Monitoring , 2017.

[4] Shaobing Huang, Li Yu, Fang-jian Han,"A pipelined architecture for user-defined floating-point complex division on FPGA", 2017 IEEE 30th Canadian Conference, 2017.

[5] Milena Svobodova, Edita Pelantova, Marta Pavelka,"On-line algorithms for multiplication and division in real and complex numeration systems",Discrete Mathematics & Theoretical Computer Science , 2019.

[6] Kevin Brady Hil, "Efficient Implementation of Radix-4 Division by Recurrence", 2018.

[7] Suresh Mopuri, Swati Bhardwaj, Amit Acharyya,"Coordinate Rotation based Design Methodology for Square root and Division computation", IEEE Transactions, 2018.

[8] Jatinder Thaku, "Design and Implementation of 32-Bit Arithmetic Divider and Multiplier using Single Stage Design _ampersandsignnbsp", International Journal of Advances in Scientific Research and Engineering, 2017.

[9] Javier D Bruguera, "Low Latency Floating-Point Division and Square Root Unit", IEEE Transactions on Computers, 2019.

[10] M. D. Ercegovac, J. M. Muller; "Complex Division with Pre-scaling of Operands" Proceedings of the Application Specific Systems, Architectures, and Processors, 2003.