Power Optimization in 8-Bit Binary Divider

Yatendra Kumar
Department of Electronics and Communication Engineering
Galgotias College of Engineering & Technology Greater Noida, India
yatendrakumar0804@gmail.com

Rachit Yadav
Department of Electronics and Communication Engineering
Galgotias College of Engineering & Technology
Greater Noida, India
rachit.r.yadav@gmail.com

Shivaji Shukla
Department of Electronics and Communication Engineering
Galgotias College of Engineering & Technology Greater Noida, India
shiva.0711.ksj@gmail.com

Amit Gupta
Department of Electronics and Communication Engineering
Galgotias College of Engineering & Technology Greater Noida, India
amit.gupta@galgotiacollege.edu

ABSTRACT

Nowadays, power consumption is the primary concern for the designing of digital systems (VLSI circuits). Power consumption is a considerable issue in the emergence of green computing also. The need for low power design has become a major issue in high performance digital system such binary dividers, etc. To meet the power requirement in today’s world, scientists are focusing on designs for low power applications. The failure of reducing power consumption below a specific point can lead to the increase in the cost of systems, reduction in the market life of the systems along with other problems. An optimization of power dissipation can be achieved through various methods. Binary division is an important part of binary arithmetic. An 8-bit binary divider is capable of dividing two 8-bit numbers of base 2 i.e., binary numbers. Therefore, this paper proposes to solve this problem of power consumption in 8-bit binary divider. Thereby, optimize the power used by an 8-bit binary divider by using and further modifying the Goldschmidt Algorithm.

Keywords
Goldschmidt division algorithm, power optimization, dividers, digital systems.

1. INTRODUCTION

Digital systems which include computers, have dividers as their essential parts among others. Division operation is one of most important operations that constitute digital systems. The division operation has proved to be a more complicated operation than other operations to implement because the result of each iteration needs to be checked before the next one is started. In the field of VLSI and digital system algorithms, fast implementations of the basic arithmetical operators such as dividers are of great importance. Many digital applications require division operation for operands of significant size to get executed with minimum power. There are numerous types of division algorithms proposed by various scientists to realize fast division techniques, such as- restoring, non-restoring algorithm, Goldschmidt division algorithm, Newton Raphson algorithm. Out of these, for the optimization of power consumption in an 8-bit binary divider, Goldschmidt algorithm has been chosen. This division algorithm is named after Robert Elliott Goldschmidt who was an Electrical Engineer from Massachusetts Institute of Technology. It is an iterative division algorithm used in many processors. It uses an iterative method of multiplying the dividend and the divisor by a common factor, till the divisor converges to 1. This makes the dividend to converge to a value which the required quotient.

In the base paper, a Booth’s multiplier is used for carrying out the multiplication operations of the Goldschmidt division algorithm, which results in more power consumption. So, in order to optimize the power, this paper proposes to replace the Booth’s multiplier with 2s multiplier because power
consumption is a major issue in the designing of digital systems.

2. METHOD

2.1 Explanation

Goldschmidt division algorithm is an algorithm that includes multiplication of the dividend and divisor by a common factor. This common factor is chosen to make the divisor converge to 1. This causes the dividend to converge to the sought quotient Q.

This algorithm is an iterative procedure. This paper would investigate a division algorithm called Goldschmidt’s Algorithm, change the multiplier used in it and reduce the power consumption. Goldschmidt’s Division Algorithm was very keenly improved and investigated thoroughly. Along with division, this algorithm was utilized for the calculation of Square Root and Square Root reciprocal values. The fundamental process of this algorithm from is described as follows. Assume the respective values of the Numerator (N) and the Denominator (D) to satisfy the limit 1 ≤ N and D < 2 (Both of these are being treated as the significands of floating-point numbers that have been normalized already).

The basic process for division is executed as follows:

Quotient \( Q = \frac{N}{D} \). The goal of the Goldschmidt’s algorithm is to find a sequence \( K_1, K_2, K_3, \ldots, K_i \) such that the product \( n_i = D \cdot K_1 \cdot K_2 \cdot K_3 \cdot K_4 \cdot K_5 \ldots \ldots \) Ki approaches 1 as it tends to infinity.

Therefore,

\[ q_i = N \cdot K_1 \cdot K_2 \cdot K_3 \cdot K_4 \cdot K_5 \ldots \ldots, \quad K_i \rightarrow Q. \]

K1 is taken from the lookup table. It is an optimal reciprocal table which takes p-bits as input and gives p+2 bits as output where the value of D is utilized as input and it helps obtain a p+2 bit approximation K1 to 1/D.

The input bits and their contemplation of accuracy were already met earlier so the process would proceed with the basic idea to optimize the power.

2.2 Flow Chart

![Flow chart of the complete process](image)

3. RESULTS

The power being consumed by the 8-bit binary divider has successfully been optimized by replacing the multiplier of the Goldschmidt division algorithm. The Booth’s multiplier which was being used in the base paper has been replaced with 2s multiplier.

The total power consumption has drastically decreased from when the Booth’s multiplier was being used to now, when Booth’s multiplier has been replaced with 2s multiplier.

The total power being consumed by the 8-bit binary divider with Booth’s multiplier was 36mW and now the total power being consumed with 2s multiplier is 14mW.

| Table 1. Comparison of values for Goldschmidt divider with booth’s multiplier and 2S multiplier |
|---------------------------------|---------------------------------|---------------------------------|
|                                 | Values for Goldschmidt divider  | Values for Goldschmidt divider  |
|                                 | with Booth’s Multiplier          | with 2S Multiplier              |
| Resources                      | Clk | Logic | Signal | i/o | Clk | Logic | Signal | i/o |
| Used                           | 3   | 7021  | 7876   | 40  | 1   | 56    | 145    | 40  |
| Available                      | -   | 2728  | -      | 31  | -   | 2400  | -      | 102 |
| Utilization(%)                 | -   | 26    | -      | 13  | -   | 2     | -      | 39  |
| Total power mW                 | 36  | 14    |        |     |     |       |        |     |
Total power consumed by the 8-bit binary divider using Goldschmidt algorithm with Booth’s multiplier is 36 milliWatts (mW).

Total power consumed by the 8-bit binary divider using Goldschmidt algorithm with 2S multiplier is 14 milliWatts (mW).

**Fig 2:** Snapshot of the power consumed shown by Xilinx

**Fig 3:** RTL view of Goldschmidt algorithm using Booth’s multiplier

**Fig 4:** Snapshot of the optimized power using 2S multiplier shown by Xilinx

**Fig 5:** RTL view of Goldschmidt algorithm using 2S multiplier

4. DISCUSSION

4.1 Booths Multiplier

The modified Booth’s multiplication algorithm is designed using a high-speed adder. As the name suggests, the high-speed adder is used to speed up the operation of Multiplication. Designing of this algorithm is done by using VHDL and is also simulated.

Signed multiplication is a complicated process which requires attention and concentration because the sign of the operands is also to be taken into consideration. For unsigned multiplication operation, there is no need to take the sign of the numbers into consideration. However, in signed multiplication the same process as that of unsigned multiplication cannot be applied because the signed number is in a 2’s compliment form which would yield an incorrect result if multiplied in a similar fashion as that of unsigned multiplication. This is the place where the role of
Booth’s algorithm comes into play. Booth’s algorithm conserves the sign of the result.

4.2 2S Multiplier

There exist two methods to implement the multiplication part of two’s complement by hand. First is to implement these algorithms directly into the circuitry. But this would end up in very slow multiplication. In actual, there exist far more complex implementations. The algorithms used, produce the product of more than one bit for each clock cycle.

It must be kept in mind that the result can require twice as many bits as there were in the originally introduced operands. It can be assumed that both operands contain the same number of bits. In reference [3], five major types of division algorithms were presented. So, we learnt about Goldschmidt’s division algorithm and stepped ahead to use it in this research paper for the optimization of power. The research paper mentioned under the references column on “Low Power Divider”[4] shows different techniques for the designing of a low-power divider by remitting the recurrence, changing the redundant representation to reduce the number of flip-flops, using gates with lower drive capability, equalizing the paths to reduce glitches, disabling the clock and many more. Here, in this paper, a completely different and more efficient technique of power optimization in 8-bit binary divider has been discussed and proposed. It is done by the replacement of the multiplier used for the Goldschmidt’s division algorithm i.e., Booth’s multiplier with 2S multiplier. In [10], a low power and cost effective architecture of a divider using the ancient Indian Vedic division algorithm is proposed. Since dividers are the inevitable parts of digital systems and Goldschmidt’s algorithm is more accurate, this paper proposes to reduce the power consumption by an 8-bit binary divider using the Goldschmidt’s algorithm.

5. CONCLUSION

The Goldschmidt algorithm of fast division has been successfully implemented using Xilinx Vivado and the work summary has been successfully shown in the diagrams, above. As stated earlier, in the base paper, for carrying out the multiplication operations of the Goldschmidt division algorithm, a Booths multiplier is used. Due to the use of Booths multiplier, the power consumption is high. The need for low power consumption has become a major issue in the designing of digital systems. Therefore, in order to optimize the power, this paper proposes a solution i.e., to replace the Booths multiplier with 2s multiplier.

6. ACKNOWLEDGEMENTS

It gives us a great sense of pride to present this research paper, undertaken by us as our B.Tech. final year project. A special debt of gratitude is owed to our project mentor, Mr. Amit Gupta, Assistant Professor, Department of Electronics and Communication Engineering, Galgotias College of Engineering & Technology for his constant support and guidance throughout the course of our work. His sincerity and perseverance have been a constant source of inspiration for us. It is only his efforts that our ventures have seen the light of the day. Our deep sense of gratitude is accorded to Dr. Lakshmanan M, Head, Department of Electronics and Communication Engineering, for his constant official support, encouragement and motivation for our project work. Our sincere thanks to our Project Coordinator, Dr. S.P. Singh, Associate Professor, Department of Electronics and Communication Engineering for the transmission of his enthusiasm, competence, as well as for the richness of his guidelines and invaluable suggestions throughout the project.

The opportunity to acknowledge the contribution of all faculty members and lab-Istructors of the department for their kind assistance and cooperation during the development of our project should also not be missed. Last but not the least, the acknowledgement our friends for their contribution in the completion of the project.

REFERENCES


