STM JOURNALS

Research & Reviews:
Discrete Mathematical Structures

ISSN: 2394-1979
Volume 10, Issue 1, 2023
DOI Journal: 10.37591/RRDMS

http://computers.stmjournals.com/index.php?journal=RRDMS&page=index

Review RRDMS

A Novel Binary Arithmetic Computational Method

Sumanta Dey[1,*], Avishek Saha[1], Puspen Halder[1], Amit De[1], Arindam Sadhu[2], Gour Gopal Jana[2], Pijush Dutta[2]

Abstract

The proposed algorithm for arithmetic operations presents an innovative and intriguing approach, primarily centered around the use of counters and the manipulation of '1's in binary representations. This algorithm promises to introduce significant advancements in computational efficiency and accuracy, making it a potential game-changer in the field of arithmetic calculations. At the core of this method lies the reliance on the number of '1's in each location within a binary number. The user's task is to count these '1's accurately, as the entire algorithm pivots on this crucial factor. If the count of '1's at a specific location is odd, the resulting sum output is '1'; otherwise, it is '0'. Moreover, to ensure continuity in the calculations, half of the '1's will be carried over to the subsequent level, contributing to a streamlined and consistent computational process. One of the standout features of this approach is its adaptability to handle signed binary computations with ease. The use of the 1's and 2's complement methodology further enhances its capability to deal with negative numbers efficiently, demonstrating its robustness and versatility. With its focus on optimizing the utilization of '1's in binary numbers, this algorithm showcases a fresh perspective in arithmetic operations. By leveraging this fundamental element, it opens up new possibilities for enhancing the speed and accuracy of calculations, potentially revolutionizing diverse applications, such as data processing, encryption, and digital signal processing. Furthermore, the reliance on counters as a foundational concept in this algorithm introduces an element of parallel processing, leading to potential opportunities for harnessing the power of parallel computing architectures, thereby optimizing its implementation on modern computational platforms.

Keywords: Arithmetic operation, logical computation, binary number

INTRODUCTION

The opening section provides a clear outline of the study's purpose, its boundaries, and the key breakthroughs being presented. It also refers to pertinent findings from prior research. A comprehensive account of both theoretical and experimental methodologies is presented to elucidate the researcher's investigative efforts. The results and discussion segment succinctly presents the obtained results and their potential ramifications [1]. Within the discussion, a thorough analysis is offered to highlight the significance of the results when compared to recent developments in the field.

The binary number system employs only two digits, 0 and 1, to represent numbers, and it can be utilized to express or perform calculations with any real number. Understanding the practical applications of binary numbers is essential. Binary numbers are used in the digital systems because we have already come to know that the valid two Boolean values 1 and 0 make the system suitable for use in electronic circuits. Computers use binary numbers that represent all data and instructions. The application of binary numbers helps in comprehending intricate data.

***Author for Correspondence**
Sumanta Dey
E-mail: sumanta.dey2002@gmail.com

[1]Student, Department of Electrical Communication Engineering, Greater Kolkata College of Engineering and Management, Baruipur, West Bengal, India
[2]Assistant Professor, Department of Electrical Communication Engineering, Greater Kolkata College of Engineering and Management, Baruipur, West Bengal, India

Received Date: July 18, 2023
Accepted Date: August 01, 2023
Published Date: August 14, 2023

Citation: Sumanta Dey, Avishek Saha, Puspen Halder, Amit De, Arindam Sadhu, Gour Gopal Jana, Pijush Dutta. A Novel Binary Arithmetic Computational Method. Research & Reviews: Discrete Mathematical Structures. 2023; 10(1): 21–25p.

The value of a number depends on the following two factors:

1. The value and position of the numbers;
2. The base or radix used for representation. The fundamental element of the binary number system is 2.

In binary numbers, there are two compliments [2–5]:

1. 1’s Compliment; and
2. 2’s Compliment.

1’s COMPLIMENT

There are various types of complements possible of the binary number, but 1’s compliment and 2nd compliment are used in most cases for binary numbers [6]. If anyone wants to find the first complement of binary number, then we have to invert the given number. For example, the first or1’s compliment of the number 10000111 is 01111000.

Example: 1st compliment of 1111100 is 0000011

The primary significance of the 1's complement of a binary number lies in its vital role in representing signed binary numbers [7]. Its principal application revolves around the representation of signed binary values. Apart from this, 1’s complement is also used in various types of arithmetic operations like addition, subtraction etc. [8].

The main importance of the 1's complement of a binary number is its crucial role in representing signed binary numbers. Its primary application is centered around signifying and working with signed binary values [9].

1’s Complement Addition Rule

1. Find the total number of required bits to represent the signed number. Required bits for signed number n is [2(n–1)–1].
   Now it will be greater to the maximum of the magnitude or equal to the maximum of the magnitude of X:Y:X+Y.
2. Find first (1’s) complement of the negative number.
3. Find the binary Addition.
4. If any carry is present as a result, then you have to add the carry bit with the last significant bit position.
5. If the most significant bit is set to 1 in the one’s complement form, the number is negative. If the most significant bit is 0 then number will be positive and in true form.

Example

8+9=17 A, B, C A+B+C

Step 1: Find the required bits.

n=6

Step 2: Find out 1’s complement.

+8=001000

+9=001001

Step 3: Binary Addition.

+8=001000

+9=001001

+17=010001

Step 4: As there is no carry then MSB is 0, so sign is present as positive and result is (0 10001)=(+17).

Subtraction Rules for 1's complement

1. Find the total number of required bits to represent the signed number. Required bits for signed number n is [2(n–1)–1].
   Now it will be greater to the maximum of the magnitude or equal to the maximum of the magnitude of X:Y:X-Y.
2. Find the first (1's) complement of the negative number.
3. Find the binary Addition.
4. If any carry is present as a result, then you have to add the carry bit with the last significant bit position.
5. If the most significant bit is set to 1, the number will be represented in a negative form using the ones complement notation. If the most significant bit is 0 then the number will be positive and in true form.

Example

8–9= –1

Step 1: Find the required bits.

n=5

Step 2: Find out 1's complement.

+8=01000

–9=1's complement of (+9)=01000

= 1's compliment is (10111)

Step 3: Binary number Addition

01000

10111

11110=00001

Step 4: As MSB is 1 then number is negative and 1's complement form.

So, number is 1s complement of –(11110)=–(00001) means final result is –1.

2'S COMPLEMENT

First, you have to find the 2's complement of the given negative number 1001. So, to find 2's complement, you have to change all 0 to 1 and all 1 to 0 or you have to find the 1's complement of the binary number 1001 [10]. The 1's complement of the binary number 1001 is 0110, and then you have to add 1 to the last significant bit of the result 0110. So finally, the 2's or second complement of the binary digit 1001 is 0110+1=0111.

110010100=001101011

+1=

001101100

Rules of 2's complement Addition

1. Find the total number of required bits to represent the signed number. Required bits for signed number n is [2(n–1)–1].

Now it will be greater to the maximum of the magnitude or equal to the maximum of the magnitude of X:Y:X+Y.

2. Find the first (1's) complement of the negative number.
3. Find the binary Addition.
4. If any carry is present as a result, then you have to add the carry bit with the last significant bit position.
5. Then find the result; If most significant bit is 1 then number will be negative, and represented in one's complement form. If most significant bit is 0 then number will be positive and in true form [2–5].

Example

(–8)+(–9)= –17

Step 1: Find the required bits.

n=6

Step 2: +8=001000

+9=001001

–8=2's complement of (+8)= ones complement of +8+1

=(001000)+1

=(110111)+1=111000

Same way (–9)=(110110)+1=110111

Step 3: 1

```
  111000
  110111
--------------
  101111
---------------
```

Step 4: The MSB is 1, so the answer is present as negative and 2's complement form.

Step 5: 2's complement of (101111)=(010000)+1=010001

So final answer is –(010001)= –(17)

Subtraction Rules for 2's Complement

1. Find the total number of required bits to represent the signed number. Required bits for signed number n is [2(n–1)–1].
   Now it will be greater to the maximum of the magnitude or equal to the maximum of the magnitude of X:Y:X-Y.
2. Find the first (1's) complement of the negative number.
3. Find the binary Addition.
4. If any carry is present as a result, then you have to add the carry bit with last significant bit position.
5. Then find the result; If most significant bit is 1 then number will be negative negative and in the ones complement form. If most significant bit is 0 then number will be positive and in true form.

Example

8+(–9)= –1

Step 1: Find the required bits.

n=5

Step 2: find out 2’s complement.

+8=01000

–9= 2’s complement of +9 = the ones complement of (+9)+1

= complement of (01001)+1

=(10110)+1=10111

Step 3: Binary Addition

01000

10111

11111

Step 5: As MSB is 1 then number is negative and it is in 2’s complement form. So, binary number 2s complement of (11111)=(00000)+1=00001 means final result is –1.

CONCLUSION

In conclusion, the content emphasizes the importance of 1's and 2's complements in the binary number system for arithmetic operations. The rules and examples provided offer practical insights into their application, enabling efficient manipulation of binary numbers in digital systems and computer circuits. Grasping these concepts improves computational abilities and contributes to progress in technology and data processing.

REFERENCES

1. Yeh WC, Tan SY, Zhu W, Huang CL, Yang GY. Novel binary addition tree algorithm (BAT) for calculating the direct lower-bound of the highly reliable binary-state network reliability. Reliab Eng Syst Saf. 2022 Jul 1; 223: 108509.
2. Jain RP. Modern digital electronics. New Delhi: Tata McGraw-Hill Education; 2003.
3. Fettweis A. Digital circuits and systems. IEEE Trans Circuits Syst. 1984 Jan; 31(1): 31–48.
4. Roth CH. Teaching digital system design using VHDL. In Proceedings of 1994 IEEE Frontiers in Education Conference-FIE'94. 1994 Nov 2; 102–105.
5. Anand R. Digital Electronics. 2nd Edn. India: Khanna Publishing House; 2014.
6. Marpe D, Schwarz H, Wiegand T. Context-based adaptive binary arithmetic coding in the H. 264/AVC video compression standard. IEEE Trans Circuits Syst Video Technol. 2003 Jul; 13(7): 620–636.
7. Yeh WC. Novel binary-addition tree algorithm (BAT) for binary-state network reliability problem. Reliab Eng Syst Saf. 2021 Apr 1; 208: 107448.
8. Belyaev E, Turlikov A, Egiazarian K, Gabbouj M. An efficient adaptive binary arithmetic coder with low memory requirement. IEEE J Sel Topics Signal Process. 2013 Jun 18; 7(6): 1053–61.
9. Santana Jr CJ, Macedo M, Siqueira H, Gokhale A, Bastos-Filho CJ. A novel binary artificial bee colony algorithm. Future Gener Comput Syst. 2019 Sep 1; 98: 180–96.
10. Su YZ, Yeh WC. Binary-addition tree algorithm-based resilience assessment for binary-state network problems. Electronics. 2020 Jul 27; 9(8): 1207.