2012 4th International Conference on Computer Engineering and Technology (ICCET 2012)
IPCSIT vol.40 (2012) © (2012) IACSIT Press, Singapore

Design of a Complex Binary Associative Dataflow Processor

Tariq Jamil

Department of Electrical and Computer Engineering

Sultan Qaboos University AlKhod 123 – Muscat - OMAN

Abstract. Complex numbers in engineering applications such as digital signal processing and image processing deserve formulation of an efficient method for their representation and processing in a computer system. Complex Binary Number System (CBNS) allows both real and imaginary parts of a complex number to be represented as a single unit (instead of two separate units as in today's microprocessors), thus allowing for reduced number of arithmetic operations dealing with complex numbers and, hence, increase in overall system performance. Incorporating CBNS within a parallel processing environment, such as associative dataflow processor (ADP), provides an ideal amalgamation of novelty in computer arithmetic with the use of associative memory in computer architecture. In this paper, an introduction to CBNS has been presented followed by the presentation of the associative dataflow concept and the design of a complex binary associative dataflow processor (CBADP). This design has been granted an innovation patent no. 2010100706 by the Australian Patent Office in July 2010.

Key Words: Complex binary, Associative dataflow, Microprocessor, Parallel processing.

1. Introduction

Complex numbers play pivotal role in various electrical and computer engineering applications. To represent these numbers, for storage in memory or to process these numbers for various arithmetic operations within the Arithmetic and Logic Unit (ALU) of a Central Processing Unit (CPU), a technique called divide-and-conquer is used wherein the complex number is broken into its real and imaginary parts and then arithmetic operations are carried out on each part, independent of any knowledge about its twin real or imaginary part in the complex number. Thus, addition of two complex numbers, (a+jb) and (c+jd), involves two separate additions, (a+c) and (b+d), one for the real parts and one for the imaginary parts. Similarly the multiplication of the same two complex numbers requires four individual multiplications (ac, ad, bc, bd), one addition (ac+j(ad+bc)) and one subtraction (j2bd = –bd). The number of nested arithmetic operations within the complex numbers' addition and multiplication can be effectively reduced to just one addition and one multiplication for the given cases if we are able to represent each complex number (both real and imaginary parts) as a single entity instead of as two separate numbers.

Efforts in providing one-unit representation to complex numbers include work by Knuth [1], Penny [2], and Stepanenko [3]. Jamil et. al. have revived the idea of a (–1+j)-base complex binary number system (CBNS) and have presented a detailed analysis of arithmetic algorithms involving these numbers [4,5].

This paper is organized as follows: In Section 2, a brief introduction to (–1+j)-base complex binary number system will be presented with some examples. This will be followed by the information about the concept of associative dataflow in Section 3. In Section 4, the block diagram of a complex binary associative dataflow processor (CBADP) will be given with brief description about various components. Finally, conclusion will be presented in Section 5. Acknowledgment and References have been listed at the end of the paper.

2. (-1+J)-Base Complex Binary Number System (Cbns)

The value of an n-bit binary number with base (–1+j) can be written in the form of a power series as follows: an-1(-1+j)n-1 + an-2(-1+j)n-2 + an-3(-1+j)n-3 + an-4(-1+j)n-4 + … + a2(-1+j)2 + a1(-1+j)1 + a0 (-1+j)0 where the coefficients an-1, an-2,an-3,an-4,…,a2,a1,a0 are binary (0 or 1). By the application of conversion algorithms outlined by Jamil [6], we can represent any number in CBNS format as shown by the following examples:

2005base10 = 1110000000001110011010001 base –1+j
–2005base10 = 1100 0000 0000 1101 1100 1101base –1+j
j2005base10 = 1000000001100l101110011 base –1+j
– j2005base10 = 1110100000001110111001101l1 base –1+j
0.4375base 10 = 1.11011101base -1+j
2005.4375base 10 = 1110000000001110011011100.11011101base –1+j
j2005.4375base 10 = 1000000001100l101110011.01000111base –1+j
(2005.4375 + j2005.4375)base10
= 1110100000001110111001101000.10000110base(-1+j)

Algorithms for CBNS arithmetic operations and designs of their logic circuits are described in Jamil.[7].

3. Associative Dataflow Concept

Of the currently prevalent ideas for building computers, the two well-known and well-developed are the control-flow and the data-flow [8]. However, both these models are beset with limitations and weaknesses in exploiting parallelism to the utmost limit. Control-flow model lacks useful mathematical properties for program verification and is inherently sequential. The data-flow model is based on partial ordering of the execution model and offers many attractive properties for parallel processing, including asynchrony and freedom from side-effects. A close examination of the problems linked with dataflow model of computation reveals that they are mainly the by-products of using tokens during computations. These tokens need to be matched up with their partner token(s) prior to any operation to be carried out. This involves a time-consuming search process which results in degradation of overall performance. Since associative or content-addressable memories (AMs or CAMs) allow for a parallel search at a much faster rate, their use within the dataflow environment has been investigated under the concept of associative dataflow.

Eliminating the need for generating and handling tokens during the execution of a program, the associative dataflow model of computation processes a dataflow graph (program) in two phases: the search phase and the execution phase. During the search phase, the dataflow graph is conceptually assumed to be upside down and each node at the top of the hierarchy is considered to be the parent of the nodes which are connected to it through the arcs, referred to as children. Taking advantage of the parallel search capabilities rendered by associative memories, the idea behind the search phase is for each parent node to search for its children. Once this search is completed, each node will know what its operands are and where the destination node(s) for the result is. During the execution phase, the operations will be performed as in conventional dataflow paradigm, except the fact that now the matching of tokens will no longer be required. Thus, by eliminating tokens from the dataflow environment and using the search capabilities of associative memories, better performance can be achieved in parallel computers.

To better understand the concept of parent and children nodes, let's consider a simple dataflow graph to compute X = a+b+c+d (Fig.1). The search phase of the associative dataflow concept requires that the given dataflow graph be turned upside-down in order for each parent to search for its children. The inverted dataflow graph to allow progress of this search phase is shown in Fig.2, wherein the node at the top (N3) is at level 0, and the nodes N1 and N2 are at level 1. Node at level 0, i.e., N3, is the parent of the nodes at level 1, i.e., N1 and N2, or in other words, the nodes N1 and N2 at level 1 are the children of the node N3 at level 0. Similarly, operands' pairs (a,b) and (c,d) are the children of the nodes N1 and N2 respectively. During the search phase, each parent node will search for its children and, during the execution phase, the operations will be performed as in conventional dataflow paradigm, except the fact that now there will be no delay due to the matching of the tokens.

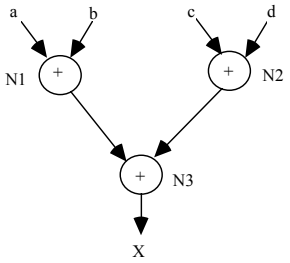

Fig. 1. Dataflow graph to compute X=a+b+c+d

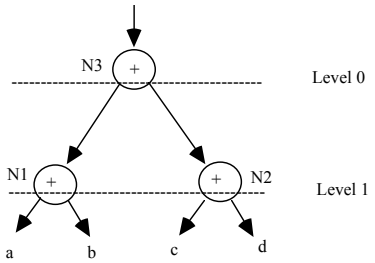

Fig.2. Dataflow graph to compute X=a+b+c+d inverted to allow progress of search phase

4. Complex Binary Associative Dataflow Processor (Cbadp)

CBADP is the hardware realization of the associative dataflow model of computation utilizing CBNS and its algorithms for representation of complex numbers and the execution of their respective arithmetic operations. A schematic block diagram for CBADP is shown in Fig.3.

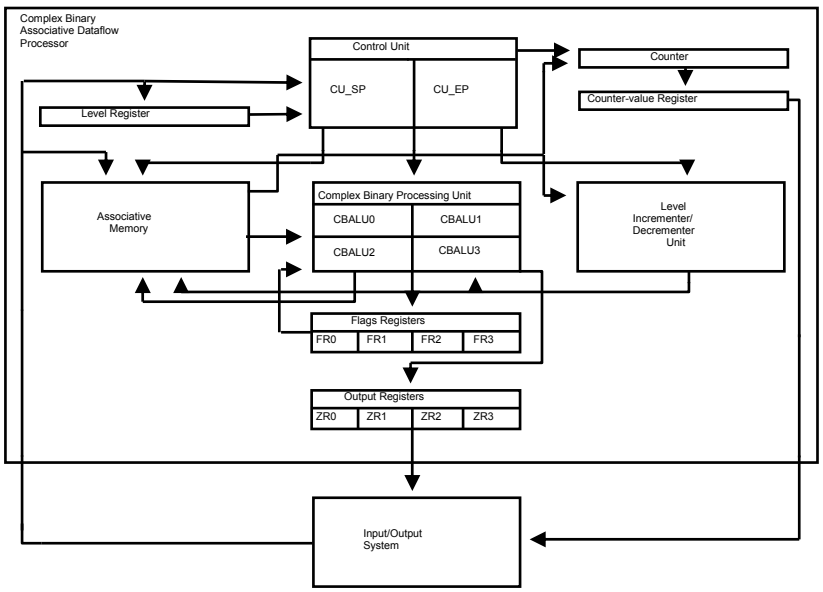

Fig.3. Schematic Block Diagram of a Complex Binary Associative Dataflow Processor

CBADP consists of following components:

(i) Associative Memory (AM) to collect and store the data needed for carrying out the given parallel computation, to store the dataflow graph in a format so as to permit the implementation of search phase of the associative dataflow concept, and to feed the data to the Processing Unit for computation of the result.

(ii) Processing Unit (PU) containing four Complex Binary Arithmetic and Logic Units (CBALUs) to compute the results of the operations carried out on the operands and to set appropriate flags in the Flags Registers (FRs), to forward these results to the appropriate word within the AM (for onward processing at the next dataflow graph level) or to the Output Registers (ZRs) (for final result).

(iii) Level Incrementer/Decrementer Unit (LIDU) to increment the current level number by one and to forward the new level number to the AM during the search phase or to decrement the current level number by one and to forward the new level number to the PU during the execution phase.

(iv) Control Unit (CU) is hardwired with the task of generating appropriate control signals for search phase and execution phase.

(v) Counter-value Register (CR) is used to store the counter value at the completion of each successful search phase.

(vi) Level Register (LR) contains information about the maximum level number in the given dataflow graph.

(vii) Flags Registers (FRs), one corresponding to each CBALU, store the flags as a result of completion of an operation.

(viii) Output Registers (ZRs), one corresponding to each CBALU, store the result of the operation and make it available to the input/output system (IOS) for reading purposes.

5. Conclusion

CBNS is expected to be the binary number system of the future because of extensive applications of complex numbers in various electrical and computer engineering areas and the need to find an efficient representation of these numbers for storage and manipulation. Design of CBADP, based on CBNS, is being implemented on Xilinx FPGA so that its benchmark performance can be compared with those of traditional control-flow and data-flow systems. An Australian Innovation Patent No. 2010100706 entitled Complex Binary Associative Dataflow Processor has been granted to the author in July 2010 and efforts are continuing towards commercial implementation of this number system for various engineering applications.

6. Acknowledgment

The author gratefully acknowledges the financial support provided by Sultan Qaboos University (Oman) for the CBNS-related projects through various research grants during 2001-2012. The technical collaboration of Neville Holmes, David Blest, Bassel Arafeh, Amer Al-Habsi, Usman Ali, Amir Arshad Abdulghani, Ahmad Al-Maashari, and Said Al-Abri with the author on these projects is greatly appreciated. The research grant provided by the Australian Research Council during 1997-1998 for investigation of the associative dataflow concept is also gratefully acknowledged.

7. References

[1]. KNUTH, D. An Imaginary Number System. Communications of the ACM [J], 1960, PP. 245-247.

[2]. PENNEY, W. A Binary System for Complex Number. Journal of the ACM [J], April 1965, PP. 247-248.

[3]. STEPANENKO, V. Computer Arithmetic of Complex Numbers. Cybernetics and System Analysis [J], 1996, 32(4): PP. 585-591.

[4].JAMIL, T; HOLMES, N.; BLEST, D. Towards Implementation of a Binary Number System for Complex Numbers. Proceedings of the IEEE SoutheastCon 2000, Nashville, Tennessee (USA), April 7-9, 2000, PP. 268-274.

[5].BLEST, D; JAMIL, T. Efficient Division in the Binary Representation of Complex Numbers. Proceedings of the IEEE SoutheastCon 2001, Clemson, South Carolina (USA), March 30-April 1, 2001, PP. 188-195.

[6].JAMIl, T. An Introduction to Complex Binary Number System. Proceedings of the Fourth International Conference on Information and Computing (ICIC), Phuket (Thailand), April 25-27, 2011, PP. 229-232.

[7]. JAMIL, T. Design of Arithmetic Circuits for Complex Binary Number System. IAENG Transactions on Engineering Technologies [J], 2011, 1373: PP. 83-97.

[8].JAMIL, T; DESHMUKH, R.G. A Novel Technique for Parallel Computations using Associaitve Dataflow Processor. Proceedings of the IEEE SoutheastCon 1995, Raleigh, North Carolina (USA), March 26-29, 1995, PP. 322-328.