High Performance 32-Bit Logarithmic Number System for Non-Linear Arithmetic Operations

by

Muhammad Sufyan Safwan Bin Mohamad Basir
(1840312663)

A thesis submitted in fulfillment of the requirements for the degree of Doctor of Philosophy

Faculty of Electronic Engineering Technology
UNIVERSITI MALAYSIA PERLIS

2021

ACKNOWLEDGMENT

First and foremost, praises and thanks to the God, the Almighty, for His showers of blessings throughout my research work to complete the research successfully. I feel grateful for the guidance and mercy until the completion of my study. Please guide me so there will be no arrogant feeling inside me for the achievement today.

In preparing this thesis, I was in contact with many peoples, researchers, academicians and practitioners. I would like to express my deep and sincere gratitude to my research supervisor, Dr. Rizalafande Che Ismail M.Eng., Ph.D., P.Eng., SM IEEE, Professor and Deputy Vice-Chancellor (Research & Innovation), Albukhary International University, for giving me the opportunity to do research and providing invaluable guidance throughout this research. His dynamism, vision, sincerity and motivation have deeply inspired me. He has taught me the methodology to carry out the research and to present the research works as clearly as possible. It was a great privilege and honor to work and study under his guidance. I am extremely grateful for what he has offered me.

I am extremely grateful to my parents, Mohamad Basir Wakil Ahmad and Naseem Syed Abdur Rahman for their love, prayers, caring, encouragement and sacrifices for educating and preparing me for my academic dream. Thanks for also being a fans and sponsorship for my study. Also, I express my thanks to my sister and brothers, for their support and valuable prayers.

I would like to say thanks to previous supervisors and research colleagues, Assoc. Prof. Ir. Dr. Abdul Rahim Abdullah, Nur Hazahsha Shamsudin, Universiti Teknikal Malaysia Melaka and Faridah Jamil, Politeknik Merlimau Melaka, for their constant encouragement. I express my special thanks Dr. Siti Zarina Md Naziri, Faculty of Electronic Engineering Technology, Universiti Malaysia Perlis, for her genuine support throughout this research work.

I am extending my thanks to the postgraduate students of Faculty of Electronic Engineering Technology, Universiti Malaysia Perlis for their support during my research work. Their views and tips are useful indeed. I also thank all the staff of Research section of Universiti Malaysia Perlis for their kindness.

Finally, my thanks go to all the people who have supported me to complete the research work directly or indirectly.

Thank you.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

ALU	Arithmetic Logic Unit
BK	Brent Kung
BTFP	Better Than Floating-Point
CLA	Carry Look Ahead Adder
CPU	Central Processing Unit
CSA	Carry Skip Adder
CSLA	Carry Select Adder
DIVA	Data-Intensive Architecture
DSP	Digital Signal Processing
ELM	European Logarithmic Microprocessor
FLP	Floating-Point
FPGA	Field-Programmable Gate Array
FXP	Fixed-Point
IEEE	Institute of Electrical and Electronics Engineers
KS	Kogge Stone
LF	Ladner Fisher
LNS	Logarithmic Number System
LUT	Look-Up Table
MONARCH	Morphable Networked Micro-Architecture
RCA	Ripple Carry Adder
ROM	Read Only Memory
RTL	Register Transfer Level
VLSI	Very Large-Scale Integration

LIST OF SYMBOLS

$\|e\|_{\text{av rel arith}}$	Average relative arithmetic error
Δ	Region
A	Actual values
$\hat{A}$	Approximating value
C_{in}	Carry input
C_{out}	Carry output
D	Stored the values of derivative, $db'(r_h)$
e	Exponent
E	Stored the values of maximum error, E
e	Absolute error
$e_{\text{max rel}}$	Maximum relative error
$e_{\text{max rel arith}}$	Maximum relative arithmetic error
$e_{\text{min rel}}$	Minimum relative error
$e_{\text{min rel arith}}$	Minimum relative arithmetic error
f	Fraction
F	Stored the values of function, $db(r_h)$
$f(r)$	Complex function for addition and subtraction
F1	Stored the co-transformation 1 values within region, $-\Delta[1] < r < 0$
F2	Stored the co-transformation 1 values within region, $-0.5 < r < -\Delta[1]$
F3	Stored the co-transformation 2 values within region, $1 < r < -0.5$
$F_A(r)/sb(r)$	Addition over number of bit
$F_S(r)/db(r)$	Subtraction over number of bit
m	Mantissa
P	Stored the proportion values of the largest absolute maximum error
$p(r)$	Interpolation function
$r/n/p$	Number of bit
s	Sign
ε	Error

Sistem Nombor Logaritma Prestasi Tinggi untuk Operasi-operasi Aritmetik Bukan Linear

ABSTRAK

Di dalam aritmetik komputer, sifat algebra yang mudah pada sistem nombor logaritma (LNS) terhadap pendaraban dan pembahagian adalah penting. Penambahbaikan versi LNS diperlukan memandangkan pengiraan bukan linear pada penambahan dan penolakan merupakan cabaran ketara. Di samping itu, fungsi kompleks penolakan berpunca dari ketunggalan memerlukan saiz memori yang besar. Fokus tesis ini adalah mengimprovisasi dua algoritma, iaitu penjelmaan-bersama dan penentudalaman untuk 32-bit unit aritmetik berprestasi tinggi. Untuk prosedur penjelmaan-bersama, rantau ketunggalan yang merangkumi reka bentuk sebelumnya dioptimumkan untuk mengurangkan kependaman. Sementara itu, aneka skema penentudalaman dalam operasi penambahan dan penolakan juga dinilai untuk jadual carian (LUT) padat untuk memberi kecepatan dan kejituan pengiraan. Penilaian dalam kerja penyelidikan mendedahkan seni bina LNS yang optimum dengan menggunakan teknik penjelmaan-bersama serupa secara puratanya mempunyai kelajuan 47% lebih cepat apabila ia ditanda aras terhadap reka bentuk LNS baru. Daripada skema penentudalaman hibrid dicadangkan, kecepatan dan luas bagi reka bentuk LNS baru didapati jauh lebih unggul berbanding reka bentuk LNS sedia ada dan juga setanding dengan reka bentuk titik apung (FLP) baru. Jumlah memori yang dikuasai oleh seni bina penjelmaan-bersama juga dapat dikurangkan. Sistem yang dicadangkan disintesis dengan memilih penambah Ladner Fisher (LF) dan juga pengganda Booth with Wallace Tree untuk menambah kelajuan pengiraan. Telah didapati bahawa sistem yang dicadangkan dapat memberikan penjimatan luas dengan pengiraan pantas di samping mengekalkan kejituan lebih baik dari titik apung (BTFP), yang bersesuaian dengan keputusan tanda aras dibuat terhadap unit nombor logaritma dan titik apung sebelumnya.

High Performance 32-Bit Logarithmic Number System for Non-Linear Arithmetic Operations

ABSTRACT

In a computer arithmetic, a straightforward algebraic property of a logarithmic number system (LNS) towards the multiplication and division are of importance. An improvised version of a LNS is needed since the non-linear computations at the addition and subtraction represent significant challenges. In addition, the complex function of the subtraction caused by singularity requires an enormous size of memory. This thesis focuses on the improvisation of two algorithms, namely the co-transformation and interpolation for a high performance 32-bit arithmetic unit. For the co-transformation procedure, the singularity region covered by the previous architecture is optimized to reduce the latency. Meanwhile, assorted interpolation schemes in addition and subtraction operations are also evaluated for a compact lookup table (LUT) to give a rapid and an accurate computation. The evaluation in this research work reveals an optimized LNS architecture with a double co-transformation technique which has an averagely 47% faster speed when it is benchmarked against the recent LNS design. From the proposed hybrid interpolation scheme, the speed and area of the new LNS design are found to be far more superior than the existing LNS design and are also on par with the recent floating-point (FLP) design. The amount of memory which is dominated by the co-transformation architecture can also be reduced. The proposed system is synthesized by selecting a Ladner Fisher (LF) adder as well as a Booth with Wallace Tree multiplier to boost the computation speed. It is found that the proposed system is able to provide an economical area with rapid computation while sustaining the accuracy better than the floating-point (BTFP), which is in agreement with the benchmarking results made against the previous logarithm number and floating-point units.

CHAPTER 1 : INTRODUCTION

1.1 Introductions

Designing a high-performance digital signal processing (DSP) architecture is crucial in advanced computational area of time-varying signal processing, image processing and machine vision (Yao et al, 2015; Khirade and Patil, 2015 and Wang et al, 2016). Operations of DSP microprocessors involve executions of the input data through arithmetic functions (i.e. addition, subtraction, multiplication, division, square, and square root) to perform the calculations. For real-time applications that usually involve wide dynamic range of numbers, the DSP algorithms need to be computed at high speed with low delay. Modification in the computations of the arithmetic functions becomes an interesting topic (Rodríguez-Andina et al, 2015) in improvising the DSP architectures for wordlength (which governs the range and precision), accuracy (i.e., error, both in quantisation and processing), speed and area (Coleman and Ismail, 2016).

In the early stage of DSP architecture, a fixed-point (FXP) technique is employed where the operands are involve with the integers using a fixed-length fraction. The FXP can perform well in complex DSP tasks with minimal bit-width, resulting in a minimization of the overall area, power, and delay for finite word-length (Fang et al, 2003). However, the drawback of the FXP is the accuracy problems which are the results from the use of the fixed-length fraction. As an alternative, a floating-point technique (FLP) is proposed with has better precision and higher dynamic range of numbers as compared to FXP. In addition,

the advantage for the multiplication process is that it can be computed by a real-time embedded system in high speed for a dynamic range of numbers. However, FLP cannot tolerate complex operations i.e., division and square root, making the execution time much slower. Thus, Coleman and his colleagues (Coleman and Ismail, 2016; Coleman et al, 2000; Ismail and Coleman, 2011 and Coleman et al, 2001) proposed a microprocessor which is based on logarithmic operations, known as a logarithmic number system (LNS), which simplifies the multiplication, division, square, and square root operations. The microprocessor is designed with a single precision 32-bit numbering format that is well known for its outstanding performance, especially for today ARM technology that had been employed in mobile computing (Abbasinezhad-Mood et al, 2019).

Using similar architectures for addition and subtraction perform by FXP, LNS provides simple calculations for multiplication and division which greatly improve the accuracy and speed. Unlike multiplication and division operations, involvement of a non-linear function that occurs at LNS addition and subtraction operations is time consuming. In spite of simply ignoring this drawback, research conducted by Coleman et al, 2000; Naziri et al, 2015 and Naziri et al. 2014 proposed an interpolation and co-transformation methods towards LNS addition and subtraction operations. These methods provide significant reduction in the look-up table (LUT) being used in terms of its size and complexity, which makes LNS comparable with FLP (Coleman and Ismail, 2016). From the proposed first-order co-transformation (Coleman et al, 2000), the delay for subtraction computes is 28 ns which is like the results for FLP under 0.7 μm^2 under CADENCE framework. In addition, the error results from 32-bit computation for both

addition and subtraction are kept within FLP-equivalent error of 0.5 LSB, which is a benchmark for high accuracy arithmetic logic system right now. For interpolation case, by simplifying the LNS functions can boost up the computation speed in accordance to small LUT. However, it may introduce to an error.

Since then, various studies have been carried out in improvising the LNS architecture which also include focusing on the non-linear functions. For this new LNS system proposed, the co-transformation architecture and recent interpolations scheme studied will be rearranged to optimise the system performance especially for speed and accuracy. The targeted system may suffer enormous of LUT at co-transformation results from less complex algorithm, by means modification for first-order co-transformation (Coleman et al, 2000) will be employed rather than much complex second-order co-transformation algorithm (Ismail and Coleman, 2011) considering the use of adder and multiplier in speed path than can cause delay. Besides, interpolation scheme proposed will gives a benefit in LUT size reduction far better than recent interpolations.

1.2 Problem Statements

In the previous section, it is found that the use of a logarithmic function in executing the arithmetic operations makes LNS as a potential replacement for the existing FLP in future. With no rounding error towards multiplication, division, square and square root (Equations (1.1) - (1.4)), the FXP addition and subtraction show benefits in terms of

high accuracy with low time consumption. For $i = \log_2 x$, $j = \log_2 y$, $r = j - i$, and assuming $j \leq i$:

Multiplication : $F_{MUL} \rightarrow \log_2\left(2^i \times 2^j\right) = i + j$ (1.1)

Division : $F_{DIV} \rightarrow \log_2\left(2^i \div 2^j\right) = i - j$ (1.2)

Square : $F_{SQ} \rightarrow \log_2\left(x^2\right) = 2i$ (1.3)

Square Root : $F_{SQRT} \rightarrow \log_2\left(\sqrt{2}\right) = i/2$ (1.4)

} Linear function

However, addition and subtraction operations (Equations (1.5) and (1.6)) lead to drawbacks for LNS in terms of their complex procedures and circuitries. The non-linear function being used during the operations becomes an obstacle for LNS to be implemented in a wider range of applications.

Addition : $F_A \rightarrow \log_2\left(2^i + 2^j\right) = i + \log_2\left(1 + 2^r\right) = i + sb(r)$ (1.5)

Subtraction : $F_S \rightarrow \log_2\left(2^i - 2^j\right) = i + \log_2\left(1 - 2^r\right) = i + db(r)$ (1.6)

} Non-Linear function

Since LNS was invented, the singularity function that occurs in the subtraction operation makes the LNS still above FLP-equivalent error of 0.5 LSB for practical table size at 32-bit precision (Coleman et al, 2000). In addition, the need of LUT to cover for every single values stored are completely illogical, although those which are quantised to zero has already been discarded. Implementation of the interpolation method seems to be a practical option to resolve the issue at the addition function. However, it is still impractical to implement the interpolation method in subtraction due to the requirement of smaller

interpolation interval when there is a rapid change derivative as r approaches zero. Thus, the co-transformation method is proposed for subtraction operations for $r > -1$ (Coleman and Ismail, 2016 and Ismail and Coleman, 2011). Instead of directly using an interpolation method along with singularity function at $r = 0$, the co-transformation method is applied to convert the equivalent of r away from zero.

By exercising the co-transformation technique, the singularity values that is infinity will be transformed to the values far from that, hence it is possible to store the transformed values inside LUT. Furthermore, interpolation scheme using mathematical approximate will reduce the complexity of LNS functions hence allow the values to be stored inside much smaller LUT. From the new co-transformation proposed as well as the simplification in LUT with interpolation scheme for addition and subtraction at the negative region, the complete LNS system will be better than FLP in terms of speed and accuracy (operate within FLP-equivalent error of 0.5 LSB) for DSP architecture (Coleman et al, 2000). It is proven from the study conducted by Ismail et al, 2013 that adapting the first-order co-transformation is faster (13.15 ns) compared to second-order co-transformation (14.79 ns) and it is results from complex algorithm using second-order co-transformation. But, employing first-order co-transformation may cause disadvantage, by means additional 107776 bits need to be stored inside LUT. Considering this issue, an effort to reduce the LUT size using interpolation will take place on non-linear region in LNS functions. The complexity of the functions is reduced while the accuracy of the system will be maintained within FLP-equivalent error of 0.5 LSB.

1.3 Objectives of Research

The main objective of this research is to analyse the performance of the new LNS system based on the cost of storage requirements, speed of data processing and accuracy of data analysis in comparison with FLP arithmetic units. More specifically, the listed sub-objectives below facilitate achieving the main objective:

1. To propose a new double co-transformation technique in the negative region for substantial improvements in terms of speed and accuracy.

2. To enhance the interpolation architectures using various existing techniques for the reduction of the total system storage and delay.

1.4 Research Questions

The research is intended to resolve the following questions.

1. How does the co-transformation procedure being proposed in this work can resolve the issue of singularity to zero that occurs in LNS addition and subtraction functions, and how will this procedure improves the LNS performances in terms of speed and accuracy?

2. Can the new developed LNS system be able to compete with the FLP system in terms of the cost of storage requirements, speed of data processing and accuracy of data analysis?

1.5 Research Scope

This research aims to improvise the LNS arithmetic architectures of addition ($sb(r)$) and subtraction ($db(r)$) for a 32-bit single precision format, that follows the IEEE floating-point standard 754-1985 (IEEE Std. 754-1985) in which the numbers are organized into sign (1-bit), exponent (8-bit), and mantissa (23-bits). Figure 1.1 illustrated the non-linear functions of $sb(r)$ and $db(r)$ that occur in the negative region, which are impractical since an irrational LUT size is needed to cover all the possible wordlengths. Hence, several potential interpolation schemes which include linear i.e. Taylor, Lagrange, and trisection, non-linear i.e. high-order degree and, spline are investigated, which is aimed to reduce the LUT size while maintaining the accuracy within half FLP-equivalent LSB.

The singularity issue that occurs during LNS subtraction in both the positive and negative regions when r nears to zero causes the number to rise without boundary. A novel double co-transformation method is then proposed to eliminate the singularity within $-1 < r < 0$ region. This method is expected to be comparable or more superior than the existing European logarithmic microprocessor (ELM) system in terms of speed and accuracy. Alternative such as Arnold co-transformation that contributes toward the positive region will not be covered in this work.

The output of this research would be in terms of a new LNS system design using novel co-transformation that works in sync with interpolation scheme being proposed, by which the performance in terms of the ROM storage, speed, and accuracy should be competitive with the established FLP system. A VHDL code is scripted using Quartus II software and the performance is extracted using Synopsys. The research will not look into the background applications for LNS architecture problem and will also not be fabricated as in ELM.

1.6 Thesis Organization

In finding the research gap for LNS, a basic concept of the LNS architecture, including their arithmetic operations, characteristics, applications, advantages and disadvantages will be explored in Chapter 2. The main item in this chapter is the overview of a co-transformation architecture that is used to eliminate the singularity issue at LNS $db(r)$ which occurs in the negative region and the various interpolation schemes at both $sb(r)$ and $db(r)$ functions. Other alternatives for a fast and efficient LNS system are also critically reviewed.

Metrics of measurements that are related to the design methodology and procedure of the new LNS will be described in Chapter 3. This research focuses on the singularity issues that occurs in the non-linear functions at the addition and subtraction operations when r nears to zero. In this research, a novel co-transformation towards negative $-1 > r > 0$ region is designed with a new function characteristic using various interpolation