Design and Analysis of an Efficient Repository System for Protein Coefficients in Systolic Array-based Architecture by Using Xilinx Virtex-5 FPGA

by

Ku Noor Dhaniah Binti Ku Muhsen
(1532521563)

A dissertation submitted in partial fulfilment of the requirements for the degree of Master of Science (Microelectronic System Design Engineering)

School of Microelectronic Engineering
UNIVERSITI MALAYSIA PERLIS

2015

ACKNOWLEDGEMENT

All the praises and thanks to Allah. This dissertation would not have been possible without the guidance of my supervisor and lecturers, help from friends, and support from my family and husband.

Foremost, I would like to express my sincere gratitude to my supervisor Dr. Mohd Nazrin Md. Isa for his excellent guidance, caring, patience, providing me with an excellent atmosphere for doing research and let me experience the research of development an efficient repository system for protein sequence alignment issues beyond the microelectronic engineering since it bridging with bioinformatics field. My research would not have been possible without his motivational support. I could not have imagined having a better advisor and mentor for my MSc. study.

Secondly, I would like to thank the technical staff and fellow lab mates who always willing to help and give the best companion while using the lab. It would have been a lonely lab without them.

Special thanks to Assoc. Prof. Dr. Rizalafande Che Ismail, Dean of School of Microelectronic Engineering, UniMAP and also Dr. Sanna Taking and Dr. Norhawati Ahmad, the Postgraduate Coordinator who give this opportunity to continue my study in master's degree.

I would like to thank my parents, my in-laws and my two elder brothers. They were always supporting me and encouraging me in many ways.

Last but not least, I would like to thank my husband, Zuhayr Md. Ghazaly and my daughter, Puteri Zahra Zuhayr who always there cheering me up and stood by me through the bad and good times. These hard works was dedicated to both of them.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS

k	Number of fold
n	Number of processing elements
t_{config}	Configuration time
T_{config}	Total configuration time
$T_{config(new)}$	Total configuration time for proposed method
x	The query sequence
y	The subject sequence
n	Query sequence length
m	Subject sequence length
$F(i,j)$	The similarity score of node at the (i,j) position
$s(x_i,y_j)$	The similarity score by residue comparison of x and y sequence
x_i	The i-th character of a query sequence
y_j	The j-th character of a database sequence
d and e	Gap penalty
X and Y	The subsequences
g	Gap length
eb	Element bit-size
n_{row}	Number of row
n_{col}	Number of column

LIST OF ABBREVIATIONS

BCB	Bioinformatics and Computational Biology
BLAST	Basic Local Alignment Search Tool
BLOSUM	Block Substitution Matrix
CE	Configuration element
CLB	Configurable Logic Block
DDBJ	DNA Data Bank of Japan
DP	Dynamic Programming
EMBL-EBI	EMBL European Bioinformatics Institute
FASTA	Fast Alignment
FPGA	Field Programmable Gate Array
GPU	Graphic Processing Unit
ICAP	Internal Configuration Access Port
LUT	Look-up table
NRE	Non-recurring Engineering
PAM	Point Accepted Mutation
PC	Personal computer
PE	Processing element
PLB	Programmable Logic Block
RTR	Run time reconfiguration
RVE	Recursive variable expansion
SA	Systolic array

Rekabentuk dan Analisa Sistem Repositori Cekap Bagi Pemalar-Pemalar Protein Dalam Berasaskan Seni Bina Tatasusunan Sistolik

ABSTRAK

Penjajaran urutan telah menjadi alat yang penting dalam bidang bioinformatik dan perkomputeran biologi. Tujuan utama penjajaran urutan digunakan adalah mencari persamaan bagi urutan biologikal termasuk urutan DNA, RNA atau protein. Pencarian urutan yang baru ditemui (tidak diketahui) dan urutan yang telah diketahui (subjek) dari pangkalan data urutan ini boleh dijalankan dalam pencarian berpasangan atau antara beberapa urutan. Oleh kerana pangkalan data urutan mengalami pertumbuhan eksponen dan penggunaan masa yang lama oleh algoritma penjajaran urutan berasaskan pengaturcaraan dinamik, penyelidikan dalam percepatan berasaskan FPGA telah dilaporkan dalam kajian literasi secara meluas. Kebiasaannya, penjajaran urutan dalam perkakasan ini melibatkan dua peringkat; peringkat konfigurasi dan pengiraan. Kedua-dua peringkat ini memainkan peranan penting bagi menghasilkan keputusan penjajaran dalam masa yang realistik. Dari segi kompleksiti masa, peringkat pengiraan adalah paling lama dan diikuti dengan peringkat konfigurasi. Penggunaan rekabentuk tatasusunan berasaskan sistolik dalam penjajaran urutan protein telah membuktikan bahawa ianya salah satu cara yang terbaik dan pantas. Walau bagaimanapun, peringkat konfigurasi juga adalah cabaran yang besar terutamanya dalam rekabentuk tatasusunan berasaskan sistolik justeru perlu dipertingkatkan. Oleh itu, penyelidikan ini telah merekabentuk satu sistem repositori yang efisien terutamanya bagi penjajaran urutan protein dalam rekabentuk tatasusunan berasaskan sistolik bagi mempertingkatkan prestasi dari segi peringkat konfigurasi. Peringkat konfigurasi melibatkan kemaskini pelbagai pemalar protein yang kerap bagi semua unsur pemprosesan (PE). Ini kerana pemalar protein atau dikenali sebagai skor-skor matrik penggantian adalah penting dalam penjajaran urutan protein. Kebiasaannya, konfigurasi PE menggunakan teknik rantai konfigurasi bersiri di mana setiap PE dalam rekabentuk tatasusunan sistolik akan dikemaskini secara bersiri bermula daripada PE pertama hingga terakhir. Ini menyebabkan masa konfigurasi PE berkadar terus dengan bilangan PE dan meningkatkan masa keseluruhan sistem konfigurasi. Projek ini mencadangkan teknik alternatif bagi mengurangkan masa konfigurasi dan kebergantungan terhadap bilangan PE. Selain mengkonfigurasikan PE secara bersiri, pemalar protein boleh dipindahkan kepada PE secara serentak. Projek ini merekabentuk pemuat serentak bagi konfigurasi PE menggunakan Verilog HDL. Selain mengurangkan masa konfigurasi, penggunaan kawasan logik juga telah dikurangkan dengan mengurangkan baris dan lajur matrik penggantian yang tidak perlu, daripada 32 darab 32 kepada 20 darab 20 atau pengurangan sebanyak 61 peratus. Rekabentuk teras ini disimulasi menggunakan perisian Xilinx ISIM bagi menguji kefungsiannya. Ia juga disintesis ke atas peranti Xilinx FPGA bernombor XC5VLX50T. Ia mencapai operasi frekuensi 389.03 MHz. Dari segi prestasi kelajuan, cadangan pemuat serentak ini mengurangkan masa konfigurasi jauh lebih laju berbanding rekabentuk yang telah dilaporkan. Dari segi penggunaan kawasan logik, cadangan pemuat serentak ini menggunakan hirisan jadual carian (LUT) untuk menyimpan pemalar-pemalar protein bagi menggantikan penggunaan RAM blok. Oleh itu ia mengurangkan kebergantungan terhadap unsur ingatan terhad dalam FPGA. Hirisan dalam Xilinx Virtex-5 FPGA adalah 7200 manakala pemuat serentak menggunakan 57 hirisan iaitu hanya 0.79 peratus hirisan telah digunakan, lalu PE dalam tatasusunan sistolik lebih besar dalam FPGA dapat direka.

Design and Analysis of Efficient Repository System for Protein Coefficients in Systolic Array-based Architecture

ABSTRACT

Sequence alignment is a fundamental tool in bioinformatics and computational biology. It aims to search for regions of similarity between biological sequences, which includes DNA, RNA or protein sequences. The search for a newly discovered/unknown (query) sequence and known (subject) sequences from biological databases can be done in either pairwise or multiple sequence alignment. Due to exponential growth of biological database and the time-consuming dynamic programming-based sequence alignment algorithm, researches on FPGA-based accelerators have been extensively reported in literature. Typically, performing sequence alignment in hardware requires two stages; configuration and computation stages. These stages have an important role towards producing alignment results in realistic time. In terms of time complexity, the computation stage is the most time consuming part, followed by the configuration stage. The use of systolic array-based architecture in protein sequence alignment has been proven to be one of best and efficient ways to get alignment results in realistic time. However, the configuration stage is still a big challenge especially in the systolic array-based architecture, thus needs for improvements. Therefore, in this research, an efficient repository system specifically for systolic array-based protein sequence alignment core architecture will be designed to improve performance on the configuration side. Configuration stage involves regular and rapid updates of various protein coefficients in the processing elements (PEs). This is due to the fact that, considerations of biological factors i.e. the probability scores between pairs of amino acids characters (the protein coefficients) or known as substitution matrix is crucial in protein sequence alignment. Typical PE configuration elements used serial configuration chain, whereby each PE in the systolic array architecture will be updated sequentially from the first PE until the last one. Consequently, the PE configuration time will be proportional to the number of PEs, hence increases the overall system configuration time. This research proposes alternative to the existing approach to improve the dependency on the number of PEs and reduce the configuration time. Instead of configuring PE serially, the protein coefficients can be transferred to the PE in parallel from the proposed loader. In this work, a parallel loader for PE configuration has been designed using Verilog HDL. Besides reducing the configuration time, an area optimization of the design has been done by reducing unnecessary substitution matrix columns and rows i.e. from 32 by 32 to only 20 by 20 or 61 percent area reduction. The design core was simulated using Xilinx ISIM simulator to verify its functionality. The core was also synthesized on Xilinx FPGA with device number XC5VLX50T. The resultant operating frequency of the proposed parallel loader was 389.03 MHz. In terms of performance speed up, the proposed loader reduces the configuration time to be higher performance than reported architectures in literature. In terms of area utilization, the proposed parallel loader used slices' LUT to store the substitution matrix scores instead of using the block RAM. This reduces the design dependency on the restricted block RAM elements in FPGA. In terms of slice utilization, the proposed parallel loader utilized 57 slices when implemented in Xilinx Virtex-5 FPGA. With the total slices of 7200, the loader only utilized 0.79% of the FPGA area, thus allows for more generation of bigger PEs systolic array in FPGA.

CHAPTER 1

INTRODUCTION

1.1 Overview

In bioinformatics and computational biology (BCB) field, there is a fundamental tool which is very crucial in searching sequence homology called sequence alignment. It is important for understanding human diseases and discover new molecular for drug engineering. This alignment tool is able to search for homology of either protein (amino acid) or DNA (deoxyribonucleic acid) residues (Lavenier & Giraud, 2005). The main target of sequence alignment is to search for similarity regions between newly discovered (query) sequences and known (subject) sequences from the databases. The outcomes of this alignment may be the consequences of functional, evolutionary or structural relationships between the searched sequences.

The sequence alignment can be for pairwise or multiple sequences. In term of accuracy, sequence alignment can be done either by optimal or suboptimal technique. The suboptimal search technique will undergo heuristic approach algorithm such as BLAST (Basic Local Alignment Search Tool) (Altschul, Gish, Miller, Myers & Lipman, 1990), Gapped BLAST and FASTA (Fast Alignment) (Pearson & Lipman, 1988). These algorithms find the similarity between sequences by obtaining the number of edit distance between biological sequences. The number defines how many edit operations are required for one sequence to transform to the other. Edit operations consists of inserts, delete or substitution of one residue for another. Even though this approach has proven to be faster than optimal search technique, it fails in finding some of the similarities due to the

sensitivity concern. This is due to heuristic approach only scans related portions instead of the whole databases sequence. The optimal search technique usually used dynamic programming (DP) for accurate alignment e.g. the Smith-Waterman (SW) (Smith & Waterman, 1981) and Needleman-Wunsch (NW) (Needleman & Wunsch, 1970) algorithm which find the local and global alignment respectively. These algorithms compute the best match score between biological sequences. It is preferable than the suboptimal methods since it guarantees more accurate result due to its exhaustive search technique. SW algorithm is practical and more desirable, thus it has become the basic algorithm for many current modified algorithm used nowadays in terms of improvise the original algorithm. However, both SW and NW algorithm suffer with time consuming computation and having inflexibility dedicated hardware (Jiang, Liu, Xu, Zhang & Sun, 2007).

Upon these days, database of sequences having an exponentially growth every year, hence fast computing architecture is needed for aligning biological sequences. It can be enhanced by implementing the DP algorithm into the programmable logic and reconfigurable hardware such as Field Programmable Gate Array (FPGA) (Jiang et al., 2007; Hoang & Lopresti, 1992; Yamaguchi, Muruyama & Konagaya 2002; Yamaguchi, Yosuke, Muruyama & Konagaya, 2002; Oliver, Schmidt & Maskell, 2005; Dydel & Bala, 2004; Gök & Yilmaz, 2006; K. Benkrid, Liu & A. S. Benkrid, 2009; Isa, Benkrid, Clayton, Ling & Erdogan, 2011; Zhang, Tan & Gao, 2007; Urgese, Graziano, Vacca, Awais, Franche & Zambioni, 2012; Yamaguchi, Tsoi & Luk, 2011). Other than that, various accelerator techniques have been developed to speed up the original algorithm such as systolic arrays (Urgese et al., 2012; Yamaguchi et al., 2011; Kung & Lohman, 1980; Lipton & Lopresti, 1985; Marmolejo-Tejada et al., 2014). However, the bottleneck of implementing DP algorithm especially by using FPGA-based platform is to suffer with

not having enough resources mainly memory bandwidth. Thus, the processing element (PE) architecture for the homology search needs to be improved in terms to optimise the PE area complexity.

Throughout the search, there are two stages involved which are configuration stage and computation stage. In protein sequence alignment, it required substitution matrix for computation process when each time the two residues aligned. The substitution matrix is typically been loaded in each PE in a form of look-up table (LUT) during the configuration stage. The substitution matrix access or load time is also attributing to overall performance. Thus, configuration stage is more crucial for protein sequence alignment compared to DNA sequence alignment.

This research focused on protein pairwise sequence alignment and mainly concern on the PE configuration architecture in order to reduce the configuration time. Basically, there are two approaches that used for PE configuration such as serial configuration chain and parallel configuration chain approach. Through serial configuration chain approach, the PEs are loaded with the configuration data in sequentially thus the configuration time will be dependent to the number of PE (t_{config} x nPE). This research used parallel configuration chain technique for the PE loader and it updates all the PEs simultaneously, therefore the configuration time can be reduced to configuration time per PE (t_{config} / nPE). This PE loader will supply substitution matrix for the PEs in systolic array-based architecture as shown in Figure 1.1. Then the result of using this approach will be compared with the previous works which used the serial configuration chain approach for PE configuration.

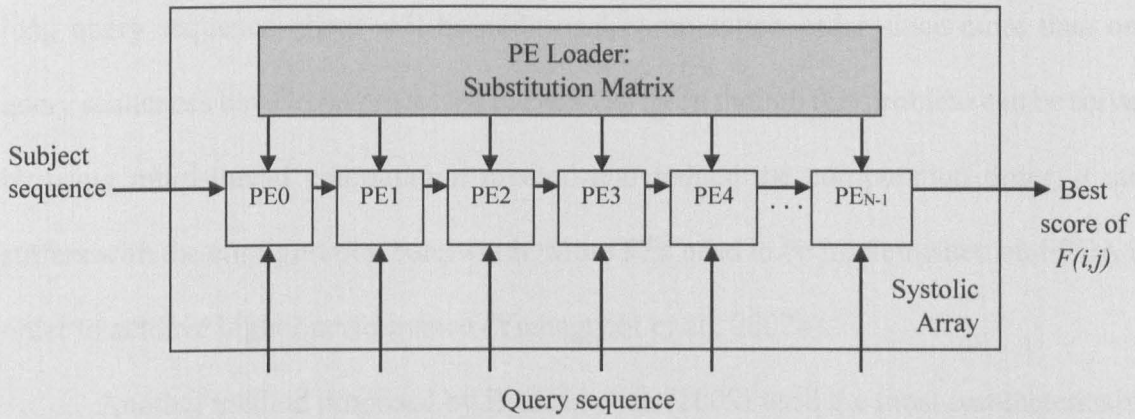

Figure 1.1: Systolic Array-based protein sequence.

1.2 Problem Statement and Motivation

Every year, the database of biological sequences is having rapid growth. Thus, various hardware implementation architectures for sequence alignment such as FPGA-based system has been proposed in previous works. Based on work by Yamaguchi et al. (2002b), the processing element architecture can optimise the configuration both in time and space complexities. It has achieved optimization in computing performance by divided it into two phases. First phase decides almost the total performance of sequence alignment since the best match scores obtained in the first phase and only routing information for trace back the best match score need to be decided in the second phase. The processing element (PE) configuration is done in the first stage by using serial configuration chain technique. This approach updates the PEs in sequentially by first, load each PEs with query sequence residue and then configuring the PE by loading substitution matrix score corresponded to the query residue loaded earlier. Thus, this approach showed that the search time for first phase increased proportionally with the query sequence length. If the length of query sequence is not larger than the number of PEs (*nPE*) on the FPGA, then the query sequence can be processed at once. In case of

long query sequence, there will be additional computation order since more than one query sequences have to be processed in each PE. Even though this problem can be solved by using multi-thread computation method and reduce the computation order, it still suffers with the configuration bandwidth. More PEs need to be implemented on FPGA in order to achieve higher performance (Yamaguchi et al., 2002b).

Another method proposed by Benkrid et al. (2009) used the most parameterizable field-programmable gate array (FPGA)-based skeleton. Through this design, it can clarifies which criteria are desirable for the alignment process; for instance in terms of the sequence symbol type (DNA, RNA or protein sequences), sequence length, the match score (score attribute to a symbol match, mismatch or gap) and the matching task (the algorithm used to match sequences including global alignment, local alignment and overlapped matching). This approach used folded systolic array where alignment matrix computation processed in sequentially by multiple passes over the same size of systolic array and consequently it can solve the issue rises due to long query sequence length. However, this technique suffers with the rise of configuration time alongside increasing number of look-up tables (LUTs) since the alignment process is performed in k passes over the linear array (Benkrid et al., 2009).

Another approach from work by Isa et al. (2011) also used folded systolic array in order to accelerate the computation time of long query sequence length but it enhanced the PE design with multiple number of configuration elements (CEs) holding the desired column of substitution matrix and this number (nCEs) depends on the total number of passes (n-pass) or folding factor. Thus, the alignment computation process can be continued without the need of reconfiguring the PE with new LUT for the next fold of computation process. Although it provides the PE with substitution matrix coefficient needed for each passes, yet is still updates the PEs by serial configuration chain. Hence,

the configuration time will increase by the number of processing element, *n*PE (Isa et al., 2011). In order to optimize the configuration time which is independent to *n*PE, new approach for PE configuration using parallel configuration chain is needed.

During implementation of dynamic programming for sequence alignment into FPGA-based hardware, it needs rapid access to obtain the probability scores from substitution matrix for PE configuration; especially for folded systolic array to compute long length sequences. The *k* folded (where *k* denotes as number of folded) will happens in case of long query sequence since it has longer length than the maximum number of PEs that can be fit on the FPGA chip in hand. Typical sequence alignment with hardware implementations configures the pipeline of processing elements by using a serial configuration chain with *k* different lookup tables. Hence, it updates the PEs with probability scores from the substitution matrix sequentially during configuration process. Therefore, more space of PE have been used since it have *k* different lookup tables or this also means each PE will be updated with more than one column of substitution matrix. Thus longer configuration time needed to complete the PE configuration. The configuration time will depend on the number of PE ($T_{config.} = t_{config} \times nPE$) and multiply with k fold for the long sequence length case ($T_{config.} = t_{config} \times knPE$). Consequently, both configuration time and space complexities increase proportional with $k.n$PE. Therefore the time configuration will increase by a factor of *k* compared to a non-folded fully pipelined architecture. This research tries to avoid the bottleneck problem of the memory management issue by using the same hardware resources and perform new approach of parallel loader for PE configuration whereby. The parallel loader will load and supply the substitution matrix coefficient to the PE in systolic array-based architecture. As a consequence of using parallel configuration chain approach for PE configuration, the configuration time can be improved.

The problem statements for this research are:

1. The conventional of sequence alignment tools using a standard PC is very time consuming. Besides, the databases of sequences is having an exponentially growth in every year. Therefore, high performance tools are needed for sequence alignment.
2. Several FPGA-based architectures have been proposed to improve the computation time of dynamic programming algorithm. However, not all architectures have improved PE configuration architecture and reduce the configuration time. Thus, an efficient PE loader to load the configuration data into PEs is needed especially for protein sequence alignment.
3. Various storage methods of storing the substitution matrix during configuration stage have been proposed in order to avoid the bottleneck of FPGA-based implementation especially the limited memory. Some previous works stored the substitution matrix in the external memory due to insufficient hardware resources. Hence, the PE configuration architecture is necessary to be optimized to improve the performance of protein sequence alignment.

1.3 Research Scope

This research focused on protein pairwise sequence alignment using FPGA-based platform. It implements the dynamic programming of Smith-Waterman algorithm into FPGA. The main focus of this research is to propose a new approach of PE loader which efficiently load and supply substitution matrix coefficient into the PE in systolic array-based architecture. During configuration stage, the PE loader used parallel configuration chain approach to update the PEs in simultaneously, thus reduce the configuration time. This new approach of parallel loader also managed to avoid the bottleneck of hardware

implementation in FPGA especially the limited memory bandwidth. The parallel loader architecture consists of fixed number of configuration elements. Therefore, the main concern of this research is to improve the configuration time which affect the overall performance and reduce the usage of hardware resources especially the dedicated memory in FPGA.

1.4 Aim & Objectives

The aim is to propose a new approach of parallel loader for protein sequence alignment in folded systolic array by optimizing in terms of time and area complexities and make comparison with other applications of protein sequencing. To achieve the aim, the objectives of this research are:

1. To investigate other PE configuration methods of FPGA-based protein sequence alignment architectures.
2. To design and optimize a parallel loader for PE configuration in protein sequence alignment systolic array-based architecture.
3. To analyse the configuration time of the parallel loader in terms of time and area complexities.

1.5 Research Module

The general view of the research area will be shown in the research module but the main focused of this research is shown by the bold line with green colour box in the research module. The flow with dashed line will not be considered in this research.

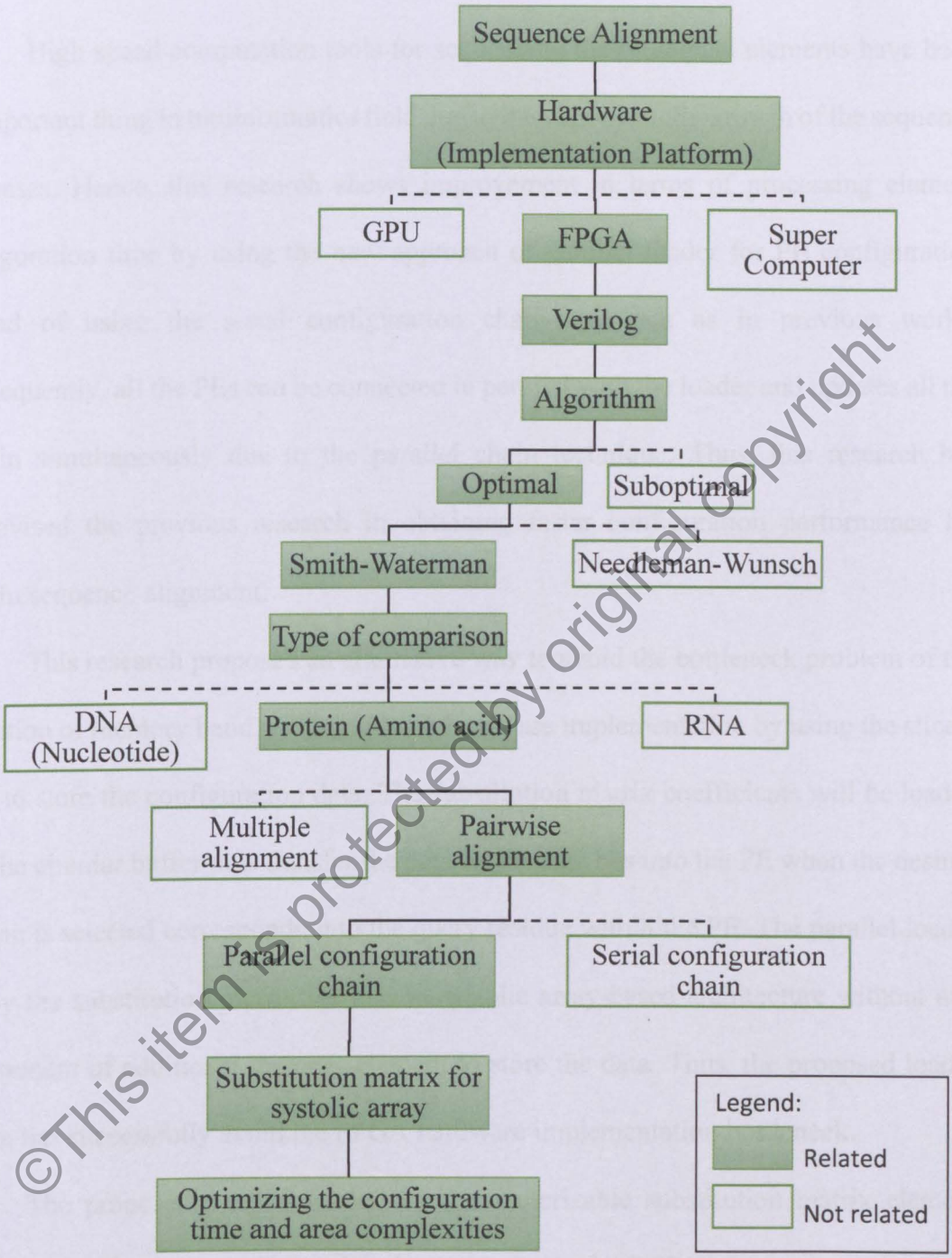

Figure 1.2: The research module.

1.6 Research Contributions

High speed computation tools for sequencing the biological elements have been an important thing in bioinformatics field due to the exponentially growth of the sequence databases. Hence, this research shows improvement in terms of processing element configuration time by using the new approach of parallel loader for PE configuration instead of using the serial configuration chain approach as in previous works. Consequently, all the PEs can be connected in parallel with the loader and updates all the PEs in simultaneously due to the parallel chain technique. Thus, this research has improvised the previous research in obtaining faster configuration performance for protein sequence alignment.

This research proposed an alternative way to avoid the bottleneck problem of the limitation of memory band width in FPGA hardware implementation by using the slice's LUT to store the configuration data. The substitution matrix coefficients will be loaded into the circular buffer thus transfer the data by PE data bus into the PE when the desired column is selected corresponding to the query residue within the PE. The parallel loader supply the substitution matrix for PEs in systolic array-based architecture without any requirement of additional memory element to store the data. Thus, the proposed loader design has successfully avoid the FPGA hardware implementation bottleneck.

The proposed parallel loader has parameterizable substitution matrix element word length. Thus, it can be used to load any type of substitution matrix such as the BLOSUM series and PAM series. This research has used to load BLOSUM 50, BLOSUM 62, PAM 160 and PAM 250.

In furtherance of this research, typical PE configuration for protein sequence alignment in systolic array-based architecture needs only one element query sequence to

process at a time. Thus, only one desired column of the substitution matrix corresponding to the query residue resides in the PE will be required by the configuration element. Hence, the proposed parallel loader will transfer only one corresponding column of the substitution matrix to each PEs and achieved smaller size of PE architecture. Thus, it also contributes in reducing the overall operation time.

1.7 Research Organization

This research will investigate on how to optimize the computation speed and area for pairwise protein sequence alignment with FPGA-based implementation using the Smith-Waterman algorithm. This implementation will use systolic array to perform parallel computation process and become accelerator to the algorithm. The main focus of this research is to propose another method to enhance the initialise phase performance by using parallel configuration chain to supply substitution matrix to the systolic array. Therefore, a new approach of parallel loader using shift register was designed to provide substitution matrix in parallel for systolic array to perform the computation process. The configuration time will be compared with the previous works which used serial configuration approach.

The research organisation will be as followed. **Chapter 1** explains briefly about the introduction of this research project. The background of the project also will be enlightened in this chapter. This section also includes the research aim, objectives and the problem statements and motivations.

Chapter 2 consists of the literature review which is related to the research project since the beginning of the research. Several topics will be discussed in this chapter such as sequence alignment, protein element, optimal alignment, Smith-Waterman

algorithm, hardware implementation and processing element architecture. The purpose of this literature review is to give more understanding and gain more knowledge and information which related to this research. Besides, it also helps in learning how the sequence alignment process will be done.

Chapter 3 explains about the tools, method and device used to design the proposed parallel loader. A brief explanation on basic FPGA design also included in this section in order to have understanding on how to implement the design on the device by used the Xilinx ISE.

Chapter 4 discusses on the simulation results when the proposed parallel loader is synthesized by using Xilinx ISIM embedded simulator. The results was analysed in terms of performance speed and area utilization compared to the prior works related. There are also discussions on the optimization done on the initial design of the proposed loader.

Chapter 5 summarises this research based on the concept used which is the parallel configuration chain technique, the operation results achieved and contribution of this research in terms of reducing the configuration time and area utilization. Some future works recommendation also been stated.

CHAPTER 2

LITERATURE REVIEW

2.1 Overview

Several applications in bioinformatics such as early disease detection, forensics, and drug engineering, need fast result from sequence alignment tool. Therefore, various approaches of hardware implementation have been proposed to achieve fast computation of sequence alignment. Some previous works implemented the dynamic programming (DP) algorithm for sequence alignment into the Supercomputer, microprocessor-based hardware such as Graphic Processing Unit (GPU) (Dohi, Benkrid, Ling, Hamada & Shibata, 2010; Munekawa, Ino & Hagihara, 2008; Feng, Jing, Zheng, Zhu & Dai, 2013; Liu, Schmidt, Voss, Schroder & Muller-Wittig, 2006; Hasan, Kentie & Al-Ars, 2011), and also in programmable logic and reconfigurable hardware such as Field Programmable Gate Array (FPGA) (Jiang et al., 2007; Hoang & Lopresti, 1992; Yamaguchi et al., 2002a; Yamaguchi et al., 2002b; Oliver et al., 2005; Dydel & Bala, 2004; Gök & Yilmaz, 2006; Benkrid et al., 2009; Isa et al., 2011; Zhang et al., 2007; Urgese et al., 2012; Yamaguchi et al., 2011).

Other than that, various accelerator techniques have been developed in the implementation hardware to speed up the original algorithm such as systolic arrays (Urgese et al., 2012; Yamaguchi et al., 2011; Kung & Lohman, 1980; Lipton & Lopresti, 1985; Marmolejo, Trujillo, Renteria & Velasco, 2014), linear recursive variable expansion (Hasan & Al-Ars, 2009; Nawaz, Nadeem, Someren & Bertels, 2010),