STREAM PROCESSOR ARCHITECTURE – STREAMING MEMORY SYSTEM

NGO WAI LOON

A project report submitted in partial fulfilment of the
requirements for the award of the degree of
Master of Engineering (Electrical - Computer & Microelectronic System)

Faculty of Electrical Engineering
Universiti Teknologi Malaysia

JUNE 2015

Specially to my beloved father and mother

Brothers and all my friends

for their inspiration, support and encouragement

throughout my adventure of educations

ACKNOWLEDGEMENT

To accomplish whatever I have done in my master project, there were many people along the way who have been responsible for guiding me, advising me and encouraging me.

I am deeply grateful and would like to take this opportunity to express my heartfelt appreciation to my supervisor, Prof. Dr. Mohamed Khalil bin Mohd Hani. I am deeply indebted and thankful to my supervisor for supporting me from the starting. I consider myself to be a fortunate person being guided by him on all the knowledge and skills required to accomplish my final year project.

In addition, I would like to take this opportunity to thank Intel Malaysia to support me in financially in pursuing this master degree. At the outset I would like to thank my family especially my parents and my girlfriend, Wong Siew Kuen who have given me support throughout my study. It would be hard without their moral support and understanding for me to finish this project. Nevertheless, my great appreciation is dedicated to Faculty of Electrical Engineering, UTM for the support and encouragement throughout the project. Besides, I would like to express my appreciation to individuals who were involved directly or indirectly with this project.

ABSTRACT

Streaming memory system in this project is defined as a process of an stream processor that need to be able to stream whole chunk of data from/to external memory with real time performance. Real-time implementation of Convolution Neural Network (CNN) application are taking large amount of CPU cycle to fetch and write data from/to external memory are popular issues in the field of media processing. So, a new FPGA-based streaming memory system was designed in order to allow the media processing application to execute multiple streaming operators. It would be a challenging task as computational capability would be increased. The hierarchy of streaming memory system consists of a memory system, stream register file (SRF) and arithmetic units. The memory system and SRF are the key part of the functional architecture of this project. This hierarchy can be used to exploit the parallelism and locality of streaming media applications. The main idea of three storage hierarchy is to allow the ALUs to operate efficiently in parallel. It is impractical to provide data on every cycle to ALU clusters using off-chip DRAM because peak bandwidth of memory system could not effectively support ALU clusters to achieve the computation rate. Moreover, the reason of creating streaming memory system is to fulfill the computational capability in media application that consists of multiple arithmetic operator. Two study case was tested which is RGB to YUV Cluster and 2D convolution cluster, and it was successfully to read or write a particular chunk of data in real time.

ABSTRAK

Sistem memori streaming dalam project ini adalah ditakrifkan sebagai proses di mana satu pemproses aliran yang dapat strim keseluruhan data dari/ke memori luaran dengan prestasi masa nyata. Implementasi masa nyata untuk aplikasi Konvolusi Rangkaian Neural (CNN) memerlukan jumlah kitaran CPU yang besar untuk mendapat dan menulis data dari/ke memori luaran merupakan isu yang popular dalam bidang pemprosesan media. Oleh itu, sistem memori streaming berdasar FPGA yang baru telah direka supaya membolehkan aplikasi pemprosesan imej untuk melaksanakan pelbagai pengendali streaming. Ia merupakan satu cabaran kerana keupayaan komputasi akan dipertingkatan. Hierarki sistem memori streaming terdiri daripada sistem memori, fail daftar aliran (SRF) dan unit aritmetik. Sistem memori dan SRF merupakan bahagian penting dalam arkitek fungsi projek ini. Hierarki tersebut boleh digunakan untuk mengekspoitasi keselarian dan aplikasi media streaming tempatan. Idea utama dalam tiga penyimpanan hierarki adalah untuk membolehkan ALUs beroperasi dengan cekap secara selari. Ia adalah tidak praktikal untuk membekalkan data pada setiap kitaran kepada kelompok ALU dengan menggunakan luar-pemproses DRAM kerana puncak lebar jalur sistem memori tidak dapat menyokong kelompok ALU dengan cekap untuk mencapai kadar komputasi. Bukan itu sahaja, faktor untuk mewujudkan sistem memori streaming adalah untuk memenuhi keupayaan komputasi dalan aplikasi media yang terdiri daripada pelbagai pengendali aritmetik. Dua kes kajian telad diuji di mana RGB ke kelompok YUV dan 2D kelompok konvolusi, dan ia berjaya membaca atau menulis sebahagian data tertentu dalam masa nyata.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

ALU	-	Arithmetic Logical Unit
SRF	-	Stream Register File
VLIW	-	Very Long Instruction Word
LRF	-	Local Register File
SP	-	Scratch Pad
CNN	-	Convolution Neural Network
CPU	-	Central Processing Unit
FPGA	-	Field-Programmable Gate Array
HDL	-	Hardware Description Language
RISC	-	Reduced Instruction Set Computing
ASIC	-	Application-Specific Integrated Circuit
DRAM	-	Dynamic Random-Access Memory
SDRAM	-	Synchronous Dynamic Random-Access Memory
DPU	-	Data Path Unit
DSP	-	Digital Signal Processing
MIPS	-	Microprocessor without Interlocked Pipeline Stages
PE	-	Processing Element
FSM	-	Finite State Machine
VCS	-	Verilog Compiled code Simulator
ISA	-	Instrcution Set Architecture
ASM	-	Algorithm State Machine

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

1.1 Background Study

Stream processors are fully programmable processors that exploit the compute intensity, parallelism, and producer-consumer locality in media applications to provide performance efficiencies comparable to special-purpose processors [1]. Stream processors are computing engines that perform calculations on an input stream with high efficiency, while an output stream is produced to be used by other stream processors and ALU clusters. Stream processors can be grouped in close proximity, and in large numbers, to provide immense parallel processing power.

The stream processor architecture effectively exploits the preferable application characteristic exposed by the stream model. The stream processor meets the bandwidth demand and computation of image processing application by directly processing the chunk of naturally arising data streams within the application [2]. Figure 1.1 shows a block diagram of stream processor architecture. The function of stream controller is to sequence the instruction from host the stream processor. The stream register file (SRF), which is a large on-chip storage for streams effectively isolates the ALU clusters from the memory system. The microcontroller is a very long instruction words (VLIW) control engine, issues instruction to arithmetic cluster. As shown in Figure 1.2, each ALU cluster consists of 3 adders, 2 multipliers, 1 divider, 1 scratch pad, 1 communication units. The ALUs are fed by two local register files (LRFs) each, external ports for accessing the SRF. In addition, there is a scratchpad (SP) unit, used for small indexed addressing operations within a cluster, and an intercluster communication (COMM) unit, used to exchange data between clusters.

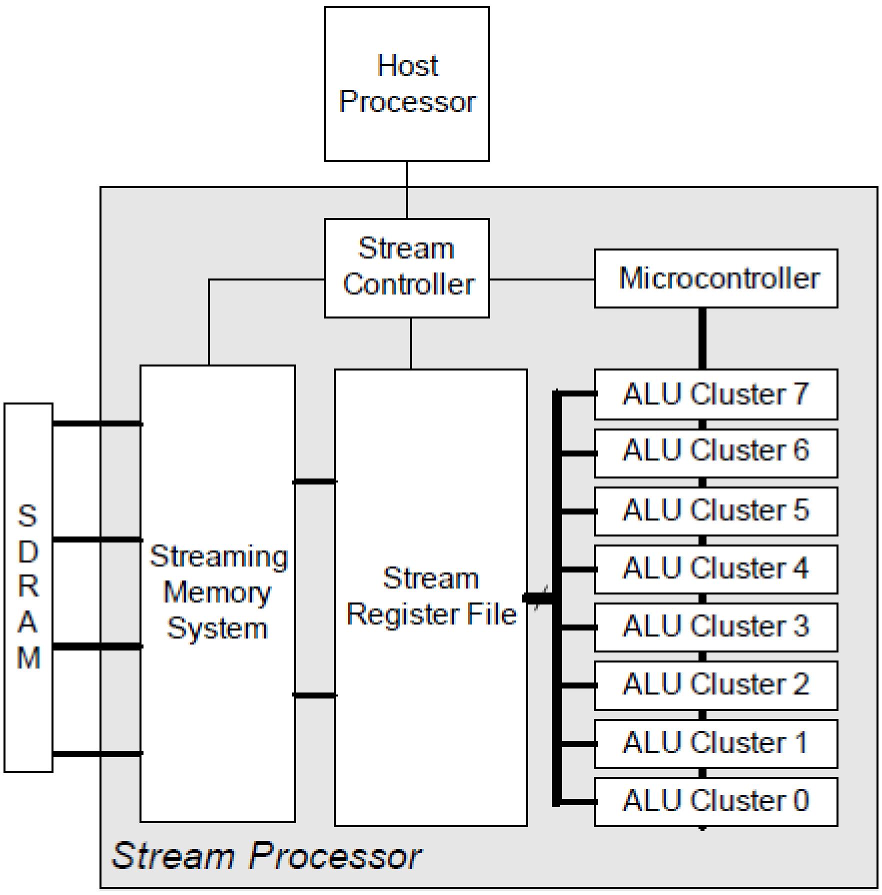

Figure 1.1: Stream Processor Block Diagram

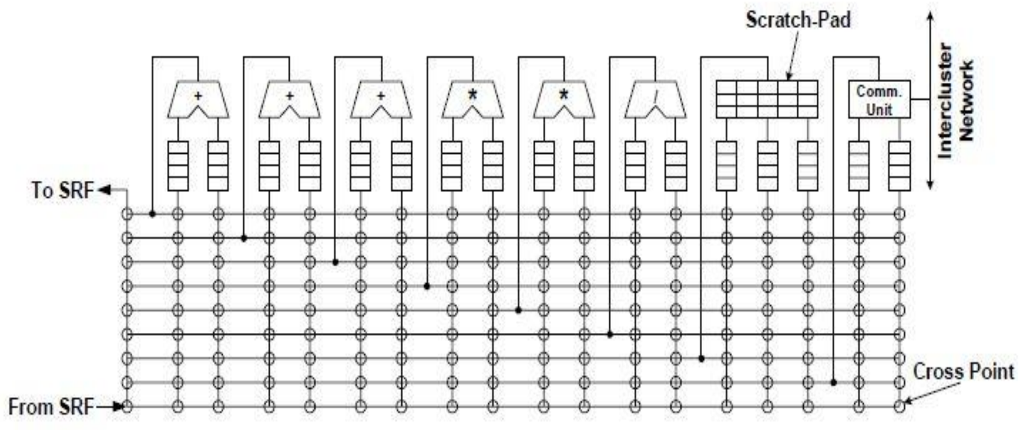

Figure 1.2: ALU Cluster

1.2 Problem Statement

The motivation behind this project is driven by some reason in the previous performance on stream memory system. Most of the real-time implementation of Convolution Neural Network (CNN) application like face recognition needs a very high speed streaming memory system to fetch and write data from/to external memory. The hardware data-path with multiple arithmetic operators and the execution of multiple streaming operators impose the challenge to increase in computational capability with increase in performance.

Moreover, a conventional load/store CPU or more commonly known as general purpose CPU is also capable of processing media applications, a dedicated stream media processor offer greater performance of throughput versus cost. Other than high CPU cycle, the biggest challenge of processing media with conventional CPU is the frequent memory access and continuous cache miss which makes the cache totally useless in this application.

1.3 Objective

There are two main objectives in designing streaming memory system which are:

1. To develop a FPGA-based streaming memory system for real-time performance CNN application with large number of ALU.
2. To develop a streaming memory system with three-tiered storage bandwidth hierarchy.
3. To develop a streaming memory system which has better memory efficiency for media processing compared to a conventional load/store processor.

1.4 Scope

The scope of the project is to focus on FPGA-based streaming memory system which are:

1. The project of streaming memory system include memory system, Stream Register File (SRF) and stream controller using System Verilog HDL in a parameterized and configurable design way, that provide flexibility to alter the parameters like bandwidth of the SRF and stream buffer.
2. The ALU cluster and microcontroller will not be part of the project. However, two dedicated ALU design which are RGB to YUV conversion and convolution will be used as test case in this project.
3. Test-bench will be created to perform functionality check for each configuration.

1.5 Project Achievement

The design of streaming memory system contribute a few achievement in the project which are:

1. The design of streaming memory system is state of the art in stream processor. This can be a new model in designing of future stream processor.
2. The new architecture return in the design is more efficient and configurable compare to a simple RISC processor.
3. The project is successfully target to FPGA design although IMAGINE return the design in ASIC. A new design methodology has been return in the project.

Reference

1. Kapasi, U.J.; Dally, W.J.; Rixner, S.; Owens, J.D.; Khailany, B., "The Imagine Stream Processor," Computer Design: VLSI in Computers and Processors, 2002.
2. Scott Rixner, “Stream Processor Architecture”, 2002.
3. Brucek Khailany, “The VLSI Implementation and Evaluation of Area and Energy-Efficience Streaming Media Processor”, June 2003.
4. Vutukuru, K.; Dessai, S., “Design and Development of Stream Processor Architecture for GPU Application Using Reconfigurable Computing,” International Journal of Reconfigurable and Embedded Systems, 2013.
5. Kapasi, U.J.; Dally, W.J.; Rixner, S.; Owens, J.D.; Khailany, B., "The Imagine Stream Processor," Computer Design: VLSI in Computers and Processors, 2002.
6. Khailany, B.; Williams, T.; Jim Lin; Peters, L.; Mark, R.; Tovey, W.; Dally, J., “A Programmable 512 GOPS Stream Processor for Signal, Image, and Video Processing”, 2009.
7. Ujval J. Kapasi; William J. Dally; Scott Rixner; Peter R. Mattson; John D. Owens; Brucek Khailany, “Efficient Conditional Operations for Data-parallel Architectures”, In Proceedings International Symposium on Computer Architecture, 2000.
8. Kapasi, U. J., Rixner, S., Dally, W. J., Khailany, B., Ahn, J. H., Mattson, P., and Owens, J. D., “Programmable Stream Processors,” IEEE Computer, August 2003.
9. http://cva.stanford.edu/projects/imagine/, Imagine Stanford, retrieved 8 Oct 2009.

10. Khailany, B., Dally, W. J., Rixner, S., Kapasi, U. J., Mattson, P., Namkoong, J., Owens, J. D., Towles, B., and Chang, A., "Imagine: Media Processing with Streams," IEEE Micro, March/April 2001.