

HARDWARE ACCELERATION OF 2D CONVOLUTION USING SYSTOLIC  
ARRAY

WONG XUE YUAN

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

*Dedicated to my beloved father, mother, brother, and all my friends for their  
inspirations and supports*

## **ACKNOWLEDGEMENT**

First and foremost, I would like to express my earnest gratitude to my project supervisor, Professor Dr. Mohamed Khalil bin Mohd Hani for his benevolent guidance as well as continuous motivation given throughout this venture. Without his supervision and constant support this project would not be accomplished successfully.

I would like to thank my parents, siblings and friends who have been truly supportive throughout my master study. Their encouragements do mean a lot to me in overcoming all forms of hardships throughout the period of my study.

In addition, I would like to take this opportunity to thank Intel Malaysia to support me financially in pursuing this master degree. My sincere appreciation also goes to the faculty members for providing all the necessary resources to enable the work in this project. Lastly, I would like to thank Lee Yee Hui for giving me guidance at the beginning of the project to help me in preparing the necessary research materials related to this project.

## ABSTRACT

Two-dimensional convolution is a prevalent mathematical operation used in different areas of digital signal processing such as image processing, video processing and analog signal transmission. The computation intensive nature of 2D convolution operation along with the stringent demand of real-time image processing in term of response time and throughput rate dismiss the viability of general-purpose processor to be used as part of the image processing solutions. Thus, the design work of a fully-dedicated 2D convolution hardware based on systolic array architecture with integrated pipeline design is proposed in this project in order to achieve optimum hardware performance in term of processing time and throughput rate. To achieve the objective, the entire hardware design is fully described in SystemVerilog and cut-set systolization procedure is applied to map 2D convolution algorithm to a 3 x 3 based systolic array hardware design. Upon the end of design and integration, the accelerated 2D convolution hardware design goes through performance benchmark. Based on the performance benchmark report, the implemented 2D convolution hardware is capable to achieve a throughput rate of 168M outputs per second. In addition, it takes 1.54 ms to complete the execution of 2D convolution based on 512 x 512 grayscale image. In comparison with general-purpose processor, the implemented design outperforms general-purpose processor in term of execution speed by 43%. The performance breakthrough marks an important milestone to the pipelined 2D convolution hardware design based on systolic array architecture as the design is proven to be essential for the future use of real-time image processing.

## ABSTRAK

Convolusi dua dimensi adalah operasi matematik yang biasa dan luas digunakan dalam bidang pemprosesan isyarat digital seperti pemprosesan imej, pemprosesan video digital, dan penghantaran isyarat analog dan digital. Oleh sebab operasi 2D convolusi memerlukan pengiraan yang amat intensif dan permintaan masa tindak balas yang ketat oleh pemprosesan imej secara langsung, unit pemproses umum yang biasa digunakan oleh komputer tidak dapat memenuhi keperluan dan spesifikasi yang dinyatakan oleh pemprosesan imej secara langsung. Oleh itu, unit pemproses umum tidak lagi dipentingkan untuk kegunaan di dalam bidang pemprosesan imej. Untuk menyelesaikan masalah ini, projek ini memainkan peranan penting untuk menghasilkan satu reka bentuk yang berdasarkan seni bina “*systolic array*” serta seni bina “*pipeline*” untuk mencapai prestasi yang paling optimum dari aspek masa pelaksanaan dan kadar hasilan. Reka bentuk projek ini dicipta sepenuhnya berdasarkan SystemVerilog. Prosedur “*cut-set systolization*” juga digunakan untuk pertukaraan algoritma 2D convolusi ke reka bentuk berdasarkan “*systolic array*” yang berbentuk 3 x 3 dimensi. Setelah reka bentuk telah siap dicipta, reka bentuk tersebut telah menjalani satu ujian prestasi. Berdasarkan laporan prestasi, reka bentuk projek ini berjaya mencapai kadar hasilan sebanyak 168M hasil/saat. Selain itu, reka bentuk ini memakan masa sebanyak 1.54ms untuk satu operasi 2D convolusi berdasarkan imej yang berdimensi 512 x 512. Berbanding dengan pemproses umum, reka bentuk tersebut boleh berfungsi lebih laju daripada unit pemproses umum sebanyak 43 peratus. Prestasi yang amat kagum ini telah membuktikan bahawa ciptaan ini memainkan peranan yang amat penting untuk kegunaan di dalam bidang pemprosesan imej secara langsung pada masa yang akan datang.

## TABLE OF CONTENTS

CHAPTER	TITLE	PAGE
	DECLARATION	ii
	DEDICATION	iii
	ACKNOWLEDGEMENTS	iv
	ABSTRACT	v
	ABSTRAK	vi
	TABLE OF CONTENTS	vii
	LIST OF TABLES	x
	LIST OF FIGURES	xi
	LIST OF ABBREVIATIONS	xiv
	LIST OF APPENDICES	xv
<b>1</b>	<b>INTRODUCTION</b>	
	1.1 Background of Study	1
	1.2 Problem Statements	2
	1.3 Research Objective	3
	1.4 Scope	3
	1.5 Project Achievement	4
<b>2</b>	<b>LITERATURE REVIEW</b>	
	2.1 Two-dimensional (2D) Convolution	5
	2.2 Systolic Array Architecture	6
	2.3 Study on Existing Designs	
	2.3.1 Window-based Image Processor with Systolic Architecture	10

	2.3.2 Run-time Self-reconfigurable 2D Convolution Core	13
2.4	Design Comparison	15
<b>3</b>	<b>RESEARCH METHODOLOGY</b>	
3.1	Work Flow Model	17
3.2	Design Methodology	18
3.3	Design Tools	20
3.4	Algorithm Specification	20
3.5	Software Simulation	21
3.6	Mapping Algorithm to Systolic Array	
	3.6.1 Dataflow Graph (DFG)	22
	3.6.2 Signal Flow Graph (SFG)	23
	3.6.2 Applying Systolization	24
	3.6.3 Generalization of Systolic Structure	26
3.7	Pipeline Diagram	26
3.8	ASM Chart	28
3.9	Block Diagrams	
	3.9.1 Top-Level Overview	30
	3.9.2 Input and Output	31
	3.9.3 Processing Element (PE)	32
	3.9.4 Linear Systolic Row	34
	3.9.5 Adder Tree	34
	3.9.6 Division Module	35
	3.9.7 Saturation Correction Unit	38
	3.9.8 Control Unit (CU)	39
	3.9.9 Datapath Unit (DU)	40
	3.9.10 Integration of CU-DU	41
	3.9.11 Summary of Design Blocks	42
<b>4</b>	<b>RESULTS AND DISCUSSION</b>	
4.1	Hardware Design Simulation	
	4.1.1 Simulation of Processing Element	45
	4.1.2 Simulation of Linear Systolic Row	46

	4.1.3 Top-level Design Simulation	47
4.2	Case Study 1 : Edge Detection with Laplacian Filter	51
4.3	Case Study 2 : Smoothing with Gaussian Filter	53
4.4	Performance Benchmark	56
<b>5</b>	<b>CONCLUSION</b>	
5.1	Conclusion	62
5.2	Future Work	63
	<b>REFERENCES</b>	64
	Appendices A – C9	65-76



## LIST OF TABLES

<b>TABLE NO.</b>	<b>TITLE</b>	<b>PAGE</b>
2.1	FPGA resource utilization of runtime self-reconfigurable 2D convolver for each algorithm	15
2.2	Summary of various 2D convolution implementation approaches	16
3.1	Purposes of every stage in project-specific waterfall design model	19
3.2	List of design tools used in this research project	20
3.3	RTL-CS table for the FSM implemented in control unit (CU)	28
3.4	Port descriptions for the top-level module of the design	32
3.5	Summary of modules in the proposed 2D convolution design	42
4.1	Generated output pixel corresponding to the input vector set	47
4.2	Benchmark results of the proposed design operating on a 512 x 512 input image	58
4.3	Performance comparison table on the execution of 2D convolution by different designs	60
4.4	Hardware resource utilization for different 2D convolution designs	61

## LIST OF FIGURES

<b>FIGURE NO.</b>	<b>TITLE</b>	<b>PAGE</b>
2.1	Illustration of 2D convolution operation in image processing context	6
2.2	General configuration of systolic array architecture	7
2.3	Design B1 (semi-systolic design)	8
2.4	Design W2 (full-systolic design)	8
2.5	Cut-set technique applied on SFG for matrix multiplication	9
2.6	Functional block diagram of configurable window processor (CWP)	11
2.7	Overview of the window-based image processor architecture	12
2.8	Architecture diagram of runtime self-reconfigurable 2D convolution design	13
3.1	Work flow model of the research project	18
3.2	Project-specific waterfall design model	19
3.3	(i)Convolution filter (ii)Section of input image for convolution	20
3.4	Execution flow of 2D convolution in software	22
3.5	Dataflow graph (DFG) of 2D convolution algorithm	23
3.6	Localized signal flow graph (SFG) for 2D convolution algorithm	24
3.7	Cut-sets applied on localized SFG	25

3.8	Systolic array processor design	25
3.9	General representation of the systolic configuration for the current 2D convolution design	26
3.10	Pipeline diagram of 2D convolution design	27
3.11	ASM chart of the 2D convolution hardware design	29
3.12	General overview of the top-level block diagram	31
3.13	Input-output block diagram (IOBD) of the design	31
3.14	Functional block diagram of processing element (PE)	33
3.15	Functional block diagram of linear systolic chain	34
3.16	Functional block diagram of adder tree	35
3.17	Basic long division in base 10	35
3.18	Functional block diagram of elementary division unit	37
3.19	Functional block diagram of basic pipeline division unit	37
3.20	Functional block diagram of complete 16-bit division module	38
3.21	Functional block diagram of saturation adjuster module	38
3.22	Functional block diagram of control-unit (CU)	40
3.23	Functional block diagram of datapath unit (DU)	41
3.24	Functional block diagram of CU-DU	42
4.1	Simulation waveform of processing element (PE)	46
4.2	Simulation waveform of linear systolic chain	47
4.3	Content of filter coefficient register in each processing element	47
4.4	Simulation waveform of input vectors to the 2D convolution processor at the initial stage of execution	49
4.5	Simulation waveform of input pixel vector buses for three linear systolic array rows	50

4.6	Simulation waveform shows the content of the filter coefficient register in each processing element	50
4.7	Simulation waveform shows the completion of 2D convolution processor execution	50
4.8	Comparison between simulation result and expected result	51
4.9	Pre-process grayscale image with a resolution of 186 x 182	52
4.10	Beginning of the simulation waveform for edge detection case study	52
4.11	Output pixels (represented in decimal base) stored in the output file	52
4.12	MATLAB code to generate the expected output image	53
4.13	Comparison between expected result and actual result for the first case study	53
4.14	Pre-process grayscale input image with dimension of 512 x 512	54
4.15	Beginning of the simulation waveform the execution of smoothing effect	54
4.16	Output pixels ported to variable window in MATLAB to render the full image	55
4.17	MATLAB code to generate the expected output image	55
4.18	Illustration of various output images with smoothing effect	56
4.19	Maximum operating frequency, $F_{\max}$ , of the proposed systolic-based 2D convolution design	57
4.20	Benchmark test coded in MATLAB	58
4.21	Variation of benchmark result by using general-purpose processor	59

## LIST OF ABBREVIATIONS

2D	-	Two dimensional
CWP	-	Configurable window processor
PE	-	Processing element
FPGA	-	Field-Programmable Gate Array
HDL	-	Hardware Description Language
ASM	-	Algorithmic state machine
FSM	-	Finite-state machine
DFG	-	Dataflow graph
SFG	-	Signal flow graph
IOBD	-	Input-output block diagram
FBD	-	Functional block diagram
CU	-	Control unit
DU	-	Datapath unit

**LIST OF APPENDICES**

<b>APPENDIX</b>	<b>TITLE</b>	<b>PAGE</b>
A	FYP 1 Gantt Chart	65
B	FYP 2 Gantt Chart	66
C1	Testbench	67
C2	SystemVerilog for CU	69
C3	SystemVerilog for DU	70
C4	SystemVerilog for top-level module	71
C5	SystemVerilog for processing element (PE)	72
C6	SystemVerilog for linear systolic row	73
C7	SystemVerilog for 16-bit division module	74
C8	SystemVerilog for ValidCounter module	76
C9	SystemVerilog for ExpireCounter module	77

## **CHAPTER 1**

### **INTRODUCTION**

#### **1.1 Background of Study**

In recent years, the design of efficient and performance-oriented 2D convolution unit has received a lot of attentions from many researchers particularly from image processing and video processing field. The key factor is the involvement of spatial two-dimensional convolution operation in most of the pervasively-used image and video processing techniques such as image filtering, image restoration, feature recognition, etc. [1]. Previously, general-purpose processor was commonly chosen as the generic execution unit to perform 2D convolution operation. However, the sequential execution of 2D convolution, which is a parallel computational conception by nature, results in poor execution performance and low execution efficiency.

Moreover, typical modern image processing techniques have high computational cost and not appropriate to real time implementation using digital signal processor or general-purpose processor [2]. This inhibits the use of general-purpose processor as a viable computation core to compute 2D convolution. Hence, this leads to more researches on the possibilities of executing 2D convolution flow in hardware approach rather than in software approach to meet the performance and timing constraints of real-time image/video processing.

In this research project, a fully-dedicated 2D convolution hardware design based on systolic array architecture is presented. The proposed design leverages the advantage of parallel processing design in hardware to speed-up the execution flow of 2D convolution. This is achieved by integrating systolic architecture in the computational block to introduce concurrent processing and pipeline concepts into the hardware design. Apart from parallel processing advantage presented in this design, the custom design specifically for 2D convolution in image processing context allows optimization work to be done carefully at each level of the design. As the result, the proposed design exhibits great computational capability for 2D convolution and has better design area and lower design cost.

## **1.2 Problem Statements**

In this project, two problem statements related to the execution of 2D convolution have been identified. The problem statements question mainly on the performance of two-dimensional convolution in software execution approach and the availability of comprehensive design documentation for dedicated 2D convolution hardware solution.

The execution of two-dimensional convolution computation flow is seen to be slow and ineffective when the computation is operated via general-purpose processor. A performance comparison on the execution of 2D convolution was previously made in a research and the performance benchmark showed that the execution of 2D convolution using general-purpose processor is slower than the execution of 2D convolution via FPGA by at least 40% [3]. The slow performance is mainly due to the sequential execution of 2D convolution in software by general-purpose processor. The slowdown is further amplified by the iterative nature of 2D convolution which involves repeating multiplication and addition operations. As a result, using general-purpose processor to execute 2D convolution will cause bottleneck to real-time image processing solutions and hence they turn away from using general-purpose processor as the execution platform.



The lacking of detailed design documentation to build dedicated 2D convolution hardware poses concerns to people who look for embedded 2D convolution solution. Many research papers had presented various hardware architecture for 2D convolution but implementation details and procedures are mostly hidden or incomplete. Even though some presented detailed work on the implementation of 2D convolution hardware design using FPGA, the designs are not tailored to meet the area and resource criteria of an embedded hardware design. Thus, this problem brings difficulties to those who want to duplicate the relevant hardware design for their needs.

### **1.3 Research Objective**

For this research project, there are three main objectives. The objectives are:

- i. To create a fully-dedicated 2D convolution hardware for image processing purpose
- ii. To optimize the existing two-dimensional convolution algorithm to achieve better hardware performance
- iii. To integrate systolic array architecture along with pipeline design into the two-dimensional convolution processor design

### **1.4 Scope**

- i. Execution of 2D convolution algorithm is based on image processing context.
- ii. The research project focuses on RTL hardware design and the design is coded based on SystemVerilog as the hardware descriptive language (HDL)
- iii. The dedicated hardware is designed to handle image which comprises of 8-bit image pixels and convolution filter with the size of 3 x 3.

- iv. The design is created based on systolic array architecture with pipelining feature.
- v. Hardware design simulation is limited to simulation in Mentor Graphics Modelsim only while software simulation is done using MATLAB.

### **1.5 Project Achievement**

Upon the completion of this research project, we have unlocked a few achievements such that:

- i. The accelerated 2D convolution hardware design based on systolic array architecture is the first among the other relevant 2D convolution designs to be fully created in SystemVerilog.
- ii. The research project has succeeded to show the ability to map the conventional 2D convolution algorithm into a hardware design based on systolic array architecture.
- iii. The created design has achieved better enhancement in terms of performance such that the processing time of the created design is at least 40% shorter compared to the processing time needed by general-purpose processor in computing 2D convolution.

## REFERENCES

- [1] Perri, S., Lanuzza, M., Corsonello, P., & Cocorullo, G. (2005). A high-performance fully reconfigurable FPGA-based 2D convolution processor. *Microprocessors and Microsystems*, 29(8), 381-391.
- [2] Castillo-Atoche, A., Torres-Roman, D., & Shkvarko, Y. (2010). Towards real time implementation of reconstructive signal processing algorithms using systolic arrays coprocessors. *Journal of Systems Architecture*, 56(8), 327-339.
- [3] Torres-Huitzil, C., & Arias-Estrada, M. (2004). Real-time image processing with a compact FPGA-based systolic architecture. *Real-Time Imaging*, 10(3), 177-187.
- [4] Bhattacharyya, S. S., Deprettere, E. F., Leupers, R., & Takala, J. (2013). Handbook of signal processing systems (Vol. 2). *Springer Science & Business*.
- [5] Gonzales, R. C., & Woods, R. E. (2002). *Digital Image Processing*, 2<sup>nd</sup> Edition.
- [6] Kung, H. T. (1982). Why Systolic Architectures? *Computer*, 15(1), 37-46
- [7] Kung, S. (1985). VLSI Array Processors. *IEEE ASSP Magazine*, 2(3), 4-22
- [8] Fons, F., Fons, M., & Cantó, E. (2011). Run-time self-reconfigurable 2D convolver for adaptive image processing. *Microelectronics Journal*, 42(1), 204-217.
- [9] Mohd Hani, M. K. (2014). *RTL Design of Digital Systems with Verilog*, 2<sup>nd</sup> Edition. Universiti Teknologi Malaysia, 2014, pp.
- [10] Guild, H. H. (1970). Some cellular logic arrays for non-restoring binary division. *Radio and Electronic Engineer*, 39(6), 345.
- [11] A. B. Fortes, K. S. Fu, B. W. Wah. Systematic approaches to the design of algorithmic specified systolic arrays. *Proc. IEEE ICASSP'85, IEEE Computer Society Press*. March 1985, pp. 300-303