

AIN SHAMS UNIVERSITY

FACULTY OF ENGINEERING

Electronics Engineering and Electrical Communications

Hardware Accelerator for Robotics and Autonomous Systems (RAS)

A Thesis submitted in partial fulfilment of the requirements of the degree
of

Doctor of Philosophy in Electrical Engineering

(Electronics Engineering and Electrical Communications)

by

Hossam Omar Ahmed Omar

Master of Science in Electrical Engineering

(Electronics Engineering and Electrical Communications)

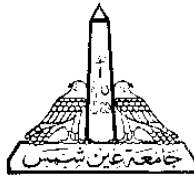
Faculty of Engineering, Ain Shams University, 2015

Supervised By

Prof. Mohamed Amin Dessouky

Dr. Maged Ghoneima

Cairo - (2019)



AIN SHAMS UNIVERSITY

FACULTY OF ENGINEERING

Electronics and Communications

Hardware Accelerator for Robotics and Autonomous Systems (RAS)

by

Hossam Omar Ahmed Omar

Master of Science in Electrical Engineering

(Electronics Engineering and Electrical Communications)

Faculty of Engineering, Ain Shams University, 2015

Examiners' Committee

Name and Affiliation

Prof. Khaled Ali Shehata
Electronics and Communications , Arab Academy of
Science and Technology
Prof. Mohamed Watheq El-Kharashi
Computer and Systems , Ain Shams University
Prof. Mohamed Amin Dessouky
Electronics and Communications , Ain Shams
University

Signature

.....
.....
.....

Date:23 August 2019

Statement

This thesis is submitted as a partial fulfilment of Doctor of Philosophy in Electrical Engineering Engineering, Faculty of Engineering, Ain shams University.

The author carried out the work included in this thesis, and no part of it has been submitted for a degree or a qualification at any other scientific entity.

Hossam Omar Ahmed Omar

Signature

.....

Date:23 August 2019

Researcher Data

Name : Hossam Omar Ahmed Omar
Date of birth : 01/09/1985
Place of birth : North Sinai - Egypt
Last academic degree : Master of Science in Electrical
Engineering
Field of specialization : Electronics and Communications
University issued the degree : Ain Shams University
Date of issued degree : 2/2/2015
Current job : Instructor, Computer Engineering and
Technology, American College of the Middle East, Kuwait

Thesis Summary

The swift growth of data size and accessibility in recent years has initiated a shift of philosophy in algorithm designs for artificial intelligence and machine learning, since the ability to learn modern systems and applications automatically from massive amounts of data depending on the conventional algorithms has led to ground-breaking performance in important domains such as natural language processing, Robotics and Autonomous Systems (RAS), speech recognition, and computer vision. Nowadays, the most popular class of techniques used in these domains is called deep learning and is seeing important attention from industry. However, these models require extraordinary massive amounts of data and compute power to train and are limited by the need for better hardware acceleration to be appropriate for scaling beyond current data and model sizes.

While the present hardware acceleration solution has been to use clusters of graphics processing units (GPU) as general purpose processors (GPGPU), the use of field programmable gate arrays (FPGA) or Application Specific Integrated Circuit (ASIC) provide interesting alternatives, since FPGA and ASIC architectures are flexible which give them the ability to explore model-level optimizations beyond what is possible on fixed architectures such as GPUs and CPU hardware based solutions. As well, FPGAs and ASICs tend to provide high performance per watt of power consumption, which is very remarkable for developing large scale server-based deployment or resource-limited embedded applications. Without a doubt, many artificial intelligence and machine learning algorithms are biologically inspired algorithms which mainly depend on dense concurrent computational processing which lead to rely on using FPGA and ASIC as the ideal Platforms for their capabilities to perform Data parallelism, Model parallelism, and Pipeline Parallelism.

In this thesis, we proved that by changing the conventional embedded multiplier blocks and the DSP blocks of the silicon fabric architecture of the FPGA chips we will boost the acceleration capabilities of the FPGA to process any generic Deep Neural Network (DNN) systems while having the ability to have more number of such accelerators due to the optimization level that has been adopted. The four proposed units proved their abilities to exceed the state-of-the-art conventional accelerators by Xilinx and Intel Altera vendors in the computational performance as will be explained in the following chapters. These results have been achieved under two main constrains. The first constrain was the shallow information about the detailed performance features of state-of -the-art DSP block by these manufacturers; and the second constrains is the unavailability of

having a high-technology files to perform a deep testing to the proposed units in order to examine their real performance capabilities. The low power consumption of the proposed units is a clear indication of their promising future to be suitable for critical applications such as the Robotics and Autonomous Systems (RAS).

Key words: Deep learning, convolutional neural networks, computational intelligence, Multiply Array Grid, Multiply Parallel Adder, Pyramidal Neuron Accelerator Architectures (PNAA).

Acknowledgment

All praise and glory go to Almighty Allah who gave me the strength and patience to carry out this work.

First and foremost, gratitude is to the esteemed university, Ain Shams University, and its faculty of engineering members for their high-quality education they supplied me with through my postgraduate studies.

My deep appreciation and gratitude go to my thesis advisors Prof. Mohamed Dessouky, and Dr. Maged Ghoneima, for their constant guidance, support and valuable time they supplied me with through my work in this thesis.

Many thanks and appreciations go to my dear parents, my uncle Sheikh Saleh Abo Khalil and all his family members, my brother and my two sisters for their continuously encouragement and support, and my friends, especially, Mr. Ahmed Hesham, Mr. Ahmed Ashraf, and Mr. Shady Ahmed, for their prayers and Support, they really supplied me with through my postgraduate studies.

August 2019

Table of Contents

LIST OF FIGURES	XIV
LIST OF TABLES.....	XVII
LIST OF ABBREVIATIONS	XIX
CHAPTER ONE INTRODUCTION.....	1
1.1 THESIS MOTIVATION	1
1.2 THESIS OBJECTIVES	3
1.3 THESIS CONTRIBUTION	5
1.4 THESIS ORGANIZATION	5
CHAPTER TWO THEORETICAL BACKGROUND AND STATE OF THE ART ..	7
2.1 INTRODUCTION.....	7
2.2 CONVOLUTIONAL NEURAL NETWORK BASICS.....	7
2.3 CHALLENGES OF IMPLEMENTING CNN ON HARDWARE PLATFORMS.....	9
2.4 MULTIPLY-ACCUMULATE UNIT	10
2.5 STATE OF THE ART OF MAC UNITS	12
2.6 TRANSITION FROM MAC TO NEURON EMBEDDED UNITS.....	14
2.7 RECONFIGURABLE SYSTEM VS RECONFIGURABLE COMPUTING.....	18
2.8 DSP APPLICATIONS USING VARIABLE-PRECISION DSP BLOCKS ON FPGAs.....	20
2.9 THE ADVANTAGES OF THE VARIABLE-PRECISION DSP BLOCKS	21
2.10 THE GENERIC COMPONENTS OF DSP BLOCKS	25
2.10.1 <i>The Pre-adders subblock</i>	26
2.10.2 <i>The Coefficient Banks</i>	28
2.10.3 <i>The Feedback Registers</i>	28
2.10.4 <i>The Multipliers</i>	30
2.11 THE DSP ARITHMETIC.....	31
2.11.1 <i>The Basic multiplication circuit design</i>	31
2.11.2 <i>The Distributed Multiplication Arithmetic</i>	32
2.11.3 <i>The Distributed Multiplication Arithmetic using LUTs</i>	32
2.12 THE SUPPORTED MULTIPLIER IMPLEMENTATIONS IN FPGA.....	33
2.12.1 <i>The Multipliers using DSP blocks, Embedded Multipliers, or Logic Resources</i>	34
2.12.2 <i>The Firm Multiplier</i>	34
2.12.3 <i>The Soft Multiplier</i>	37
2.12.3.1 <i>The Parallel Multiplication</i>	37
2.12.3.2 <i>The Semi-Parallel Multiplication</i>	39
2.12.3.3 <i>The Sum of Multiplication</i>	40
2.12.3.4 <i>The Hybrid Multiplication</i>	42
2.12.3.5 <i>The Fully Variable Multipliers</i>	43
2.13 CONCLUSION.....	47
CHAPTER THREE THE PROPOSED MULTIPLY- ACCUMULATE MODELS ..	48
3.1 INTRODUCTION.....	48
3.2 CONCURRENT MAC UNIT DESIGN.....	48
3.3 HIGH-SPEED 2D PARALLEL MAC UNIT HARDWARE ACCELERATOR.....	50
3.4 VERIFICATION ANALYSIS FOR THE PROPOSED MAC MODELS	51
3.4.1 <i>The analysis of the Concurrent MAC Unit</i>	51

3.4.2 <i>The analysis of the High-Speed 2D Parallel MAC Hardware Accelerator Unit Design</i>	57
3.5 CONCLUSION.....	61
CHAPTER FOUR THE PROPOSED PYRAMIDAL NEURON ACCELERATOR MODELS.....	62
4.1 INTRODUCTION	62
4.2 SYSTOLIC-BASED PYRAMIDAL NEURON ACCELERATOR BLOCKS	62
4.3 VERIFICATION ANALYSIS FOR THE PROPOSED PNAA MODELS ON FPGA.....	66
4.4 VERIFICATION ANALYSIS FOR THE PROPOSED PNAA MODELS ON ASIC.....	72
4.4.1 <i>The analysis related to the 3X3 PNAA unit</i>	74
4.4.2 <i>The analysis related to the 5X5 PNAA unit</i>	74
4.4.3 <i>The analysis related to the 7X7 PNAA unit</i>	74
4.4.4 <i>The Comparison of the proposed PNAA units</i>	75
4.4.5 <i>The Study limitations of the Systolic-based pyramidal neuron accelerator blocks on ASIC</i>	80
4.5 RECONFIGURABLE SYSTOLIC-BASED PYRAMIDAL NEURON BLOCK.....	80
4.6 VERIFICATION ANALYSIS FOR THE PROPOSED CSPN MODELS ON FPGA	83
4.7 CONCLUSION.....	87
CHAPTER FIVE CASE STUDY OF THE PROPOSED PYRAMIDAL NEURON ACCELERATOR SYSTEM	88
5.1 INTRODUCTION	88
5.2 SIMULATION VALIDATION PROCESS OF A SIMPLIFIED NEURON VERSION.....	88
5.3 FPGA SYSTEM FOR THE VALIDATION PROCESS OF A SINGLE 3X3 NEURON	93
5.4 FPGA SYSTEM FOR THE VALIDATION PROCESS OF A COMPLETE CONVOLUTION LAYER ON AN IMAGE.....	94
5.5 CONCLUSION.....	99
CHAPTER SIX CONCLUSION AND FUTURE WORK	100
6.1 CONCLUSION.....	100
6.2 FUTURE WORK	101
APPENDIX	102
APPENDIX A: CNN VERIFICATION TOP-LEVEL CODE.....	102
APPENDIX B: THE SERIAL INTERFACE CODE.....	107
APPENDIX C: ON-CHIP ROM MEMORY TOP-LEVEL CODE	109
APPENDIX D: PARALLEL REGISTER CODE.....	114
APPENDIX E: 3x3 PNAA TOP-LEVEL CODE	115
APPENDIX F: PROCESSING CORE 1 TOP-LEVEL CODE	121
PUBLICATION LIST	122
REFERENCES	123

List of Figures

Figure 1.1 (a) Special Purpose Dexterous Manipulator on the end of the Space Station Remote Manipulator System (SSRMS) (b) Artist's conception of Mars Exploration Rover.	1
Figure 1.2 Example of a strawberry farm that depends on RAS operations.	2
Figure 1.3 An example of an infrastructure robot in action: CISBOT in a pipe. Source: ULC robotics.	2
Figure 1.4 (a) Image of the armed predator drone. (b) Image of the ATLAS bipedal humanoid robot developed for DARPA.	3
Figure 1.5 The reflection of the main idea of this thesis	4
Figure 2.1 Generic convolutional neural network	8
Figure 2.2 Convolutional neural network hardware optimization levels	10
Figure 2.3 Generic MAC unit block diagram	12
Figure 2.4 Basic DSP48E2 functionality	13
Figure 2.5 18-bit precision mode of Stratix V FPGA	13
Figure 2.6 High-Precision Mode of Stratix V FPGA	14
Figure 2.7 Graphical representation of a biological neuron	15
Figure 2.8 Graphical representation of a convoluted neuron operation	15
Figure 2.9 Graphical representation of a RELU activation function	16
Figure 2.10 Block diagram of CNN accelerator unit by Intel	16
Figure 2.11 Block diagram of CNN accelerator unit by Xilinx	17
Figure 2.12 Graphical representation of milestone of the state-of-the-art CNN accelerators development	17
Figure 2.13 A generic island-style FPGA routing architecture	18
Figure 2.14 The types of configuration in hardware	19
Figure 2.15 Applications vs DSP precision requirements	20
Figure 2.16 The Arria V and Cyclone V 18-bit precision DSP mode	22
Figure 2.17 The Arria V and Cyclone V high precision DSP mode	22
Figure 2.18 Intel Altera vs Xilinx multiplier precision comparison	23
Figure 2.19 Intel Altera V 5AGXB3 device vs Xilinx Kintex-7 XC7K355T multiplier precision comparison	24
Figure 2.20 The high-level view of the Pre-adders subblocks	25
Figure 2.21 The multiplication reduction process using the Pre-adders subblocks	26
Figure 2.22 Generic block diagram of implementing a symmetric FIR filter	26
Figure 2.23 An application of the multiplication reduction process in symmetric FIR filter	27
Figure 2.24 The high-level view of the coefficient banks	28
Figure 2.25 The high-level view of the feedback registers	28
Figure 2.26 The high-level view of the multipliers	29
Figure 2.27 Example of a 2-bits multiplication logic schematic	30
Figure 2.28 Distributed arithmetic with four constant multiplicands	31
Figure 2.29 Distributed arithmetic with four constant multiplicands using LUTs	32
Figure 2.30 An example of the decomposition of the 12×9 multiplier	34

Figure 2.31 An Example of the implementation of a 12×9 firm multiplier circuit	34
Figure 2.32 An example of the decomposition of the 12×12 multiplier	35
Figure 2.33 An example of the implementation of a 12×12 firm multiplier circuit	35
Figure 2.34 An example of the decomposition of a 16-bit input, 10-Bit coefficient parallel multiplier	37
Figure 2.35 An example of 16-Bit Input, 10-bit coefficient parallel multiplication implementation using M4K RAM blocks as LUTs	37
Figure 2.36 An example of decomposition of a 16-bit input, 14-bit coefficient semi-parallel multiplier	38
Figure 2.37 An example of 16-bit input, 14-bit coefficient semi-parallel multiplication implementation using M512 RAM blocks as LUTs	39
Figure 2.38 An equivalent circuit of a four-multiplier sum of multiplication function	39
Figure 2.39 An example of 4-input sum of multiplication implementation using M512 RAM blocks as LUTs	40
Figure 2.40 An example of using multiple M512 RAM blocks for an 8-coefficient multiplier	41
Figure 2.41 An example of using a M4K RAM block for a 7-coefficient multiplier	41
Figure 2.42 An example of 2 input hybrid multiplication implementation using M512 RAM blocks as LUTs	42
Figure 2.43 An example of 8-bit fully variable multiplier implementation using M4K RAM blocks as LUTs	43
Figure 2.44 Deep neural network applications vs DSP precision requirements	45
Figure 2.45 Considerations for designing DNN hardware accelerators	45
Figure 3.1 The Proposed 8-bits fixed-point MAC unit	48
Figure 3.2 The General sliding window processing in CNN networks	50
Figure 3.3 The detailed 2D MAC unit hardware architecture operation	50
Figure 3.4 Slack histogram of the clock signal for the Proposed 8-bits fixed-point MAC unit using cyclone IV E EP4CE115F29C7 device	53
Figure 3.5 Slack histogram of the clock signal for the Proposed 8-bits fixed-point MAC unit using Arria 10 10AX115R4F40E3SG device	53
Figure 3.6 Slack histogram of the clock signal for the Proposed 8-bits fixed-point MAC unit using Stratix V E 5SGXEABN3F45I3YY device	53
Figure 3.7 Logic elements consumption, maximum operating frequency, and core dynamic thermal power dissipation for the proposed MAC unit	54
Figure 3.8 Slack histogram of the clock signal of the proposed 3X3 2D MAC unit	56
Figure 3.9 Slack histogram of the clock signal of the proposed 5X5 2D MAC unit	57
Figure 3.10 Slack histogram of the clock signal of the proposed 7X7 2D MAC unit	57
Figure 4.1 The Generic PNAA block diagram array grid unit	63
Figure 4.2 The Graphical hierarchy representation of the proposed PNAA using the 3x3 multiplier array grid unit	63
Figure 4.3 The Graphical hierarchy representation of the proposed PNAA using the 5x5 multiplier array grid unit	64