A Novel Large-Bit-Size Architecture and Microarchitecture for the Implementation of Superscalar Pipeline VLIW Microprocessors

by

Lee, Weng Fook

0440210010

A thesis submitted

In fulfilment of the requirements for the degree of

Doctor of Philosophy

School of Computer & Communication Engineering

Universiti Malaysia Perlis

Year 2008

Acknowledgement

I would like to take this opportunity to thank my advisor, Prof Dr Ali Yeon, for his valuable guidance throughout my years at Universiti Malaysia Perlis. His advice and help plays a major pivotal role in my accomplishing of the dissertation, not to mention his constant hard work in promoting my research.

I would also like to thank my dear friend Dr Bala Amawasai for his constant encouragement in pursuing my research work. To all the staff at Universiti Malaysia Perlis, of which have been generously helpful throughout my research, thank you for your great support. And to the directors of Emerald Systems Design Center, thank you for your generosity in financing my research work and allowing me to use the valuable compute resources on weekends.

A special word of thanks to Prof Dr Kasmiran Jumari, Prof Dr Nukala S. Murthy and Prof Dr Syed Alwee for their valuable feedback on this research, Prof Dr Mohd Zaki for chairing the viva voce session, Dr R. Badlishah for co-supervising the viva voce session and to Norzaililah Zainoddin for arranging the viva voce session.

Finally, a great thank you to my wife for being able to put up with me, especially during all those weekends that I had missed out with her to work on this dissertation.

Abstract

Microprocessors have grown tremendously in its computing and data crunching capability since the early days of the invention of a microprocessor. Today, most microprocessors in the market are at 32 bits, while the latest microprocessors from IBM, Intel and AMD are at 64 bits. To further grow the computational capability of a microprocessor, there are two possible paths. One method is to increase the bit size of the microprocessor to 128/256/512 bits. The larger the bitsize, the more data can be crunched at any one time. The second method is to implement multiple microprocessor core in a single microprocessor unit. For example, the Intel's Pentium 4 Dual Core and AMD's Athlon Dual Core both have two microprocessor core within a single microprocessor unit. Latest from Intel and AMD are quad core microprocessors with either a configuration of pseudo-quad core or full quad core within a single microprocessor unit. In a pseudo-quad core configuration, two silicon each consists dual core microprocessor is packaged within a single microprocessor unit while a full quad core consists of four microprocessor core on one silicon packaged within a single microprocessor unit.

Both methods have its advantages and disadvantages. Both methods yields different design issues and have different engineering limitations. This work explores the method of increasing the data bus size of the microprocessor from 32/64 bits to 128/256/512 bits to allow for more data crunching capability.

In the course of this work, a superscalar pipeline 64 bits VLIW microprocessor with 4 stages (fetch, decode, execute, writeback) and 3 parallel pipes is implemented on a TSMC 0.35 micron process. The implementation is then expanded to 128/256/512 bits using the same TSMC 0.35 micron process. To prove the concept that such a large bit size VLIW microprocessor can indeed be implemented, the said VLIW microprocessor

of bitsize 64/128/256 is programmed on an Altera Stratix 2 EP2S180F1508I4 FPGA and back annotated for verification.

In the TSMC 0.35 micron process implementation of the work, the critical path of the VLIW microprocessor of data bus size 128/256/512 is analyzed with its worst path within the adder of the ALU in the execute stage. Different adder architectures are investigated for suitability on synthesis implementation of large data bus size adder for efficient usage within the ALU. An adder algorithm using repetitive constructs in a parallel algorithm that allows for efficient and optimal synthesis for large data bus size is proposed as a suitable implementation for the adder within the ALU.

This work has two important findings. One is the proposed adder architecture synthesis of a large bit size adder that provides for improved performance-gatecount-product compared to conventional adder architecture synthesis. Second is the proof of concept that a large bit size VLIW microprocessor is possible by implementing a 64/128/256 bits data size on an Altera Stratix 2 EP2S180F1508I4 FPGA.

Abstrak

Mikropemproses telah berkembang dengan pesat dalam arena pemkomputeran dan pemprosesan data. Hari ini, kebanyakan mikropemproses di pasaran adalah bersaiz 32 bit, sementara mikropemproses yang terbaru dari IBM, Intel dan AMD adalah 64 bit. Terdapat dua kaedah yang boleh dilaksanakan bagi meningkatkan keupayaan sesuatu mikropemproses. Satu kaedahnya adalah dengan meningkatkan saiz bit mikropemproses kepada 128/256/512 bit. Lebih besar saiz bit, lebih banyak data yang boleh diproses dalam satu masa. Kaedah yang kedua adalah dengan melaksanakan pelbagai teras mikropemproses dalam satu unit mikropemproses. Sebagai contoh, Pentium 4 Dual Core dari Intel dan Athlon Dual Core dari AMD kedua-duanya mempunyai dua teras mikropemproses dalam satu unit mikropemproses. Yang terbaru dari Intel dan AMD adalah *quad core microprocessor* yang mempunyai dua konfigurasi, samada dengan konfigurasi *pseudo-quad core* atau *full quad* core dalam satu unit mikropemproses. Dalam konfigurasi *pseudo-quad core*, dua silicon yang mana setiap satu mengandungi dua teras mikropemproses disatukan dalam satu unit mikropemproses sementara *full quad core* mengandungi empat teras mikropemproses dalam satu silicon yang telah di satukan dalam satu unit mikropemproses.

Kedua-dua kaedah ini mempunyai kebaikan dan keburukannya sendiri. Kedua-dua kaedah menghasilkan rekaan yang berbeza dan mempunyai had limit kejuruteraan yang berbeza. Kajian ini menjelajah kaedah meningkatkan saiz *data bus* mikropemproses dari 32/64 bit kepada 128/256/512 bit untuk membenarkan lebih banyak kebolehan pemprosesan data.

Dalam hal ini *superscalar pipeline* 64 bits mikropemproses VLIW dengan 4 kategori (mengambil, menyahkod, melaksanakan, menulis balik) dan 3 sambungan selari dilaksanakan dengan menggunakan fabrikasi proses TSMC 0.35 mikron..

Perlaksanaan ini kemudian dikembangkan kepada 128/256/512 bit dengan menggunakan fabrikasi proses yang sama. Untuk membuktikan bahawa konsep saiz bit mikropemproses VLIW yang besar memang boleh dilaksanakan secara praktikal, mikropemproses VLIW dengan saiz bit 64/128/256 di programkan dalam Altera Stratix 2 EP2S180F150814 FPGA dan disimulasikan FPGA tersebut untuk pengesahan fungsi mikropemproses tersebut.

Dalam pelaksanaan kajian ini menggunakan fabrikasi proses TSMC 0.35 mikron, bahagian yang paling kritikal untuk mikropemproses VLIW dengan *data bus* bersaiz 128/256/512 adalah laluan penambah dalam ALU. Perbezaan senibina untuk jenis penambah yang berbeza di kaji untuk kesesuaian dalam pelaksanaan ALU yang mempunyai *data bus* yang bersaiz besar. Algoritma penambah menggunakan konstruk yang berulang yang menghasilkan sintesis litar digital yang optimal untuk saiz *data bus* yang besar dicadangkan sebagai implementasi yang sesuai untuk penambah dalam ALU.

Kajian ini mempunyai dua penemuan kejuruteraan yang penting. Penemuan pertama adalah cadangan algoritma penambah untuk saiz bit yang besar yang menyediakan peningkatan pencapaian prestasi-gatecount-produk berbanding dengan senibina penambahan yang konvensional. Penemuan kedua adalah pembuktian konsep yang mikropemproses VLIW dengan saiz bit yang besar memang boleh dilaksanakan secara praktikal dengan mengimplementasikan mikropemproses tersebut dengan saiz *data bus* 64/128/256 bit pada Altera Stratix 2 EP2S180F150814 FPGA.

Table Of Contents

Table Of Figures

Table Of Examples

List Of Tables

List of Abbrevations

ADK	ASIC Design Kit
ALU	Arithmetic Logic Unit
ALUT	Altera Look Up Table
AMD	Advanced Micro Devices
APR	Auto Place and Route
ASIC	Application Specific Integrated Circuit
ATM	Auto Teller Machine
ATPG	Automatic Test Pattern Generation
BIST	Built In Self Test
CISC	Complex Instruction Set Computing
CPU	Central Processing Unit
CSA	Carry Save Adder
DRC	Design Rule Check
Dspf	Detailed Standards Parasitic Format
EPROM	Erasable Programmable Read-Only Memory
FA	Full Adder
FPGA	Field Programmable Gate Array
GDSII	Graphic Data System II
IBM	International Business Machine
IC	Integrated Circuit
ILP	Instruction Level Parallelism
IO	Input Output
JTAG	Joint Test Action Group

LVS	Layout Versus Schematic
OS	Operating System
PDA	Personal Digital Assistant
POS	Point of Sale
RAM	Random Access Memory
RC	Resistance Capacitance
RISC	Reduced Instruction Set Computing
ROM	Read Only Memory
RTL	Register Transfer Level
SDL	Schematic Driven Layout
Sdf	Standard Delay Format
Spf	Standard Parasitic Exchange Format
TSMC	Taiwan Semiconductor Manufacturing Company
VLIW	Very Long Instruction Word

1 Introduction

Micro-processors and micro-controllers are widely used in the world today. It is used in everyday electronic systems, be it a system used in the industries or a system used by consumers. Complex electronic systems such as ATM machine, POS systems, financial systems, transaction systems, control systems, database systems all uses some form of micro-controller or micro-processor as the core of their system. Consumer electronic systems such as home security systems, credit cards, microwave ovens, cars, cellphones, PDA, refrigerators and other daily appliances have within the core of the system either a micro-controller or micro-processor.

What is a micro-controllers and micro-processor? If they are such a big part of our daily life, what exactly are their function?

Micro-processors and micro-controllers are very similar in nature. In fact, from a top level perspective, a micro-processor is the core of a micro-controller. A micro-controller basically consists of a micro-processor as its CPU (central processing unit) with peripheral logic surrounding the micro-processor core. As such it can be viewed that a micro-processor is the building block for a micro-controller (Refer Figure 1).

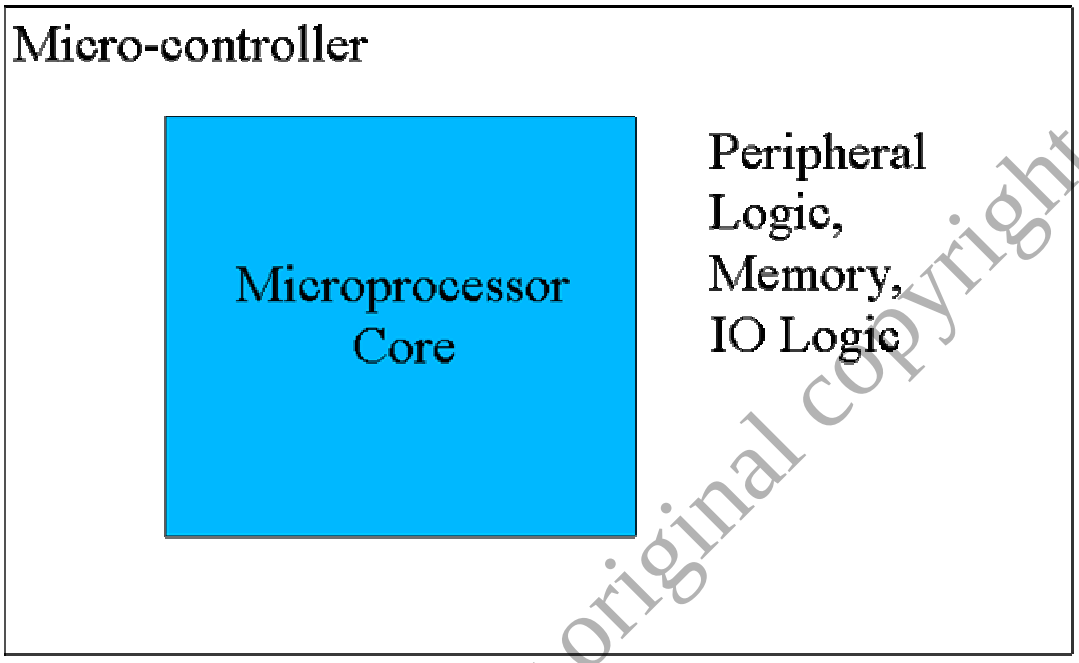

Figure 1. Diagram Showing Microprocessor as Core of Micro-controller

A more general view of a micro-controller is that it is a multi purpose IC chip that has circuitry elements that enables it to perform certain tasks required of an electronic system. A micro-controller is a single IC chip solution that can be used to perform dedicated tasks within a system, such as controlling a pump, controlling a car's engine electronic system, heart of a home security system and many others. A micro-controller consists of:

1. non volatile memory such as EPROM (Electronic Programmable Read Only Memory), ROM (Read Only Memory) that is used to store the systems' central program which allows the system to perform a specific task
2. volatile memory such as RAM (Random Access Memory) that can be used by the micro-controller for storage of information
3. peripheral logic that allows the system to have direct access to IO (input/output)

A microprocessor forms the CPU (central processing unit) of the microcontroller. Within the microprocessor is circuitry that enables the microprocessor to do arithmetic functions, logic functions and execution of instructions provided to the microprocessor.

Our daily lives are filled with usage of a computer, whether we are aware of it or not. For example, when we go to a bank and make a withdrawal using an ATM machine, the ATM machine would identify us and our bank account using an ATM card issued by the bank. That information is relayed from the ATM machine to a central computer system that transmit information back to the ATM machine on the amount of savings in the account and how much can be withdrawn at that moment. When we do decide to withdraw a certain sum of money, that transaction is automatically recorded in the bank's central computer system and the corresponding bank account. This process is automated within a computer system, and at the very heart of the computer systems lies many microprocessors.

Computers that we use daily at home or at work have a microprocessor as its brain. The microprocessor does all the necessary functions of the computer when we are using a word editor, or a spreadsheet or even preparing our electronic presentation. Computers cannot function without a microprocessor.

1.1 Data Crunching Power Of Microprocessors

A microprocessor's capability to crunch data is dependent on its bus width. The larger the bus width the more data that it can crunch at any one time. For example, the crunching capability of a 32 bit microprocessor is at a comparable doubling factor of a 16 bit microprocessor. Therefore having a microprocessor with larger bus size allows for more data crunching capability. However there is a drawback to using larger bus size. The larger the bus size, the greater amount of logic is required, and the larger the die size. Most microprocessors in the market today such as Intel's Xeon and EMT64 microprocessor, AMD's Athlon 64 and Opteron microprocessor, IBM's PowerPC microprocessor are 64 bit microprocessors. They are able to crunch data at 64 bits at a time.

Moving forward, in order to have a microprocessor to have more data crunching capability, there is two methods of progress:

1. increase the bit size from 64 bits to 128/256/512 bits and beyond
2. increase the amount of microprocessor core in a single microprocessor

Method (1) increases the bus width to accommodate for more data crunching capability, while method (2) uses multiple microprocessor core in a single microprocessor to allow for multiple activities. Each method have its advantages and disadvantages.

Figure 2 shows the two methods used for growing the computation capability of the microprocessor.