A Mesh Architecture for Data Management of Matrix Computations

by

Adam BURDENIUK

B. Eng (Hons) Computer Systems,
The University of Adelaide, 2001

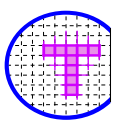
Thesis submitted for the degree of

Doctor of Philosophy

in

Electrical and Electronic Engineering
The University of Adelaide

June 2014



CHiPTec

The Centre for High Performance Integrated Technologies and Systems

ADELAIDE
UNIVERSITY
AUSTRALIA

Contents

List of Figures

List of Tables

Abstract

Using super-resolution techniques to estimate the direction that a signal arrived at a radio receiver; Tracking moving targets using particle filters; Applying advanced coding techniques to radio transmissions. Real-world applications like these rely on high-speed matrix computations. Although the computational ability of general-purpose computer architectures is growing, some numerically intensive calculations (such as those above) may benefit from specialised matrix processing hardware.

Work in this thesis builds on earlier dense matrix accelerators, which were previously implemented as coprocessors attached to general purpose processor cores. It is well known that matrix algorithms are highly parallelisable, so the coprocessor generally contains many cores to take advantage of this. However, encapsulating them within the coprocessor often prohibits their use in non-matrix computations. This limitation is addressed by the Matrix Data Management Layer (MDML), described in this thesis.

The MDML doesnt share the coprocessor-based architecture of earlier systems. Growing transistor budgets have made it feasible to embed general-purpose Commercial Off-The-Shelf (COTS) processors within the MDML. Programming is simplified as the system is homogeneous, and programmers can use familiar tools and languages. Having general-purpose processors embedded throughout the array also opens the possibility to process more than just dense matrix data.

Requirements for the MDML were synthesised by analysing common features and operations in matrix algorithms. This analysis found that data management (including sequencing, distribution and efficient reuse of data) was a significant overhead. Thus, the MDML focuses on accelerating these tasks, which are common to more than just matrix algorithms. Effort was also directed into integrating COTS components to reduce design effort, which resulted in a layered hardware design. The MDML integrates off-the-shelf memory and processing cores into a scalable on-chip mesh array.

The MDML uses a flexible sequencing regime capable of autonomously generating memory access patterns for a variety of complex data structures. This sequencer has been tested by construction of a hardware prototype. A programming style based on sequences of instruction snippets was also developed and shown effective for loop-based algorithms.

A flexible, low-latency data transport allows the MDML to use fine-grain communication. Improved array synchronisation techniques were also proposed and implemented. These techniques reduce the performance impact of barrier synchronisation, and also reduce the frequency with which the array must synchronise.

The MDML concept was verified using both hardware modelling and theoretical analysis:

- Hardware modelling was used to verify the completeness and correctness of the architecture, and demonstrated that the MDML is practical to implement in current hardware.
- Analytic models of performance were developed for algorithms mapped to a theoretical MDML system. These were used to evaluate performance, scalability and programmability of the MDML. Algorithms that were mapped include conformal matrix multiplication, LU decomposition and forward substitution.

Thus, it is concluded that the MDML architecture provides a realistic and efficient way of achieving high-speed matrix computation.

Statement of Originality

I certify that this work contains no material that has been accepted for the award of any other degree or diploma in any university or other tertiary institution and, to the best of my knowledge and belief, contains no material previously published or written by another person, except where due reference has been made in the text. In addition, I certify that no part of thius work will, in the future, be used in a submission for any other degree of diploma in any university or other tertiary institution without the prior approval of the University of Adelaide.

I give consent to this copy of the thesis, when deposited in the University Library, being available for loan, photocopying, and dissemination through the library digital thesis collection, subject to the provisions of the Copyright Act 1968.

I also give permission for the digital version of my thesis to be made available on the internet, via the University's digital research repository, the Library catalogue, the Australasian Digital Thesis Program (ADTP) and also through web search engines, unless permission has been granted by the University to restrict access for a period of time.

Signed Date

Acknowledgments

It is with joy that I sit and write this note, and think of many great colleagues, friends and mentors who helped create this thesis.

First, I am indebted to my supervisors Assoc Professors Cheng Chew Lim and Mike Liebelt for their mentorship and guidance. They nurtured my independence as a new researcher, and patiently shaped and guided the direction of this work. The friendly assistance of school support staff, including Yadi, Rosie and Stephen, was also invaluable.

I am also grateful to John Kitchen and Gareth Parker, who encouraged me to pursue this research and gave me the time and resources in which to do it. I promise it will be done by Christmas!

I must also thank my examiners. It isn't easy to write a thesis, and it's no less easy to review one. I'm grateful for their thoughtful, constructive feedback.

Thanks also to my PhD comrades, for their friendship and encouragement over many years. I'm notably grateful to thank Kiet To for his suggestions and advice. His depth of knowledge and practical capabilities never failed to impress.

I'd like to thank my good friends: Lachie, Darryl, Bobby, Rob and Hong-Gunn... and all members of the Gang. You all supported and distracted me in equal measure.

I'm extremely thankful for my fabulous family for being pillars of emotional support and encouragement. My thesis wouldn't exist without them.

Last, but not least, thanks to my dear Jennie. Without your enthusiasm and reassurance this thesis would still be years from completion.

Definitions and acronyms

ABI Application Binary Interface

AGU Address Generator Unit

AMU Array management unit

ARQ Automatic Repeat Request

ASIC Application Specific Integrated Circuit

BAG Built-in Address Generator

BLAS Basic Linear Algebra Subroutines

BLAST Bell-Labs Layered Space-Time

CBE Cell Broadband Engine

CISC Complex Instruction Set Computer

COTS Commercial Off-The-Shelf

CPU Central Processing Unit

CSP Communicating Sequential Processes

DCT Discrete Cosine Transform

DFT Discrete Fourier Transform

DSP Digital Signal Processor

DMA Direct Memory Access

DP Data Processor

DRAM Dynamic RAM

DTM Data Transport Module

FEC Forward Error Correction

FIFO First-In First-Out

FLOPS Floating Point Operations

FP Floating Point

FPGA Field-Programmable Gate Array

FSM Finite State Machine

GEMM General Matrix/Matrix Multiply

GEMV General Matrix/Vector Multiply

GER General rank 1 update

GPU Graphics Processing Unit

ILP Instruction-Level Parallelism

IPC Instructions Per Clock-cycle

I/O Input-Output

JCU Jump Control Unit

KIMP Kilo-Instruction Multi-Processor

L1 Level-1

L2 Level-2

LCU Loop Control Unit

LRU Least Recently Used

MAC Multiply Accumulate

MAG Matrix Address Generator

MDML Matrix Data Management Layer

MIMD Multiple Instruction Multiple Data

MIMO Multiple Input Multiple Output

MMX MultiMedia eXtensions

MPI Message Passing Interface

MPSoC Multi-Processor System-on-Chip

MuSiC Multiple Signal Classification

NEWS A mesh-type interconnect with links in four directions. These links are named after the four main compass directions.

OFDM Orthogonal Frequency Division Multiplexing

PCI-Express A modern, high-speed interconnect technology.

PE Processing Element

PPE Power Processing Element

RAM Random Access Memory

RAW Read After Write

RISC Reduced Instruction Set Computer

RTEMS Real-Time Executive for Multiprocessor Systems

RTL Register Transfer Level

SCAP Scalable Array Processor

ScaLAPACK Scalable Linear Algebra Package

SDR Software Defined Radio

SIMD Single Instruction Multiple Data

SMP Symmetric Multi-Processor

SoC System-on-Chip

SPE 'synergistic' processing elements

SRAM Static RAM

SVD Singular Value Decomposition

SYMM Symmetric Matrix/Matrix Multiply

TLP Thread-Level Parallelism

TRMM Triangular Matrix/Matrix Multiply

TRSM Triangular Solve with Multiple Right-hand sides

VHDL A hardware modelling language.

VLIW Very Long Instruction Word

VLSI Very Large-scale Silicon Integration