

Contents

1	Introduction	1
1.1	Contributions	3
1.2	Organisation	4
2	Architecture	7
2.1	Architectural Models	8
2.2	Architectural Flexibility	11
2.2.1	Classifying Customisations	11
2.2.2	Cost of Flexibility	12
2.3	Architectural Design Space Exploration	15
2.3.1	Classifying Architectural Explorations	15
2.3.2	Ranking Architectural Merits	17
2.4	Multi-core Architectures	18
2.4.1	Commercial Multi-core Processors	19
2.4.2	Limitations of Existing Multi-core architectures	22
2.5	Initiatives for Customisable Multi-core Processors	24
2.6	The Concept: Run-time Reconfigurable Multiprocessors	27
2.6.1	Reconfiguration Mechanism	28
2.6.2	Advantages of the New Reconfiguration Mechanism	29
2.7	Summary	30
3	Application	33
3.1	Programmability	34
3.2	Methods of Application Description	35
3.2.1	Application Description for Parallel Processors	37
3.2.2	Managing Communication and Synchronisation	38
3.2.3	Drawbacks of Existing Methods	40
3.3	Architecture-Independent Application Characteristics	40
3.3.1	Model for Computation	41
3.3.2	Model for Synchronisation	43
3.3.3	Model for Communication	44
3.4	Comparing Application-specific Attributes	45
3.4.1	DSP Applications	46

3.4.2	Multiplier used in Elliptic Curve Cryptography	48
3.4.3	Self-organising Maps	50
3.4.4	Priorities: Computation, Communication, or Synchronisation . .	51
3.5	Restating Amdahl's Law	54
3.5.1	Speedup: Comparison to Amdahl's Law	55
3.5.2	Power: Comparison to Amdahl's Law	58
3.5.3	Impact on Energy	59
3.6	Summary	61
4	Application to Architectural Mapping	63
4.1	Applications and Architectures: Fixed vs. Alterable	64
4.1.1	Fixed Applications, Fixed Architecture	65
4.1.2	Alterable Applications, Fixed Architecture	66
4.1.3	Fixed Application, Alterable Architectures	67
4.1.4	Alterable Applications, Alterable Architecture	68
4.2	Application Mapping: Objectives and Methods	69
4.2.1	Compilation Flow	69
4.2.2	FPGA Flow	72
4.2.3	Comparing the two Design Flows	74
4.2.4	Merging Compilation and Synthesis Design Flows	76
4.2.5	Considerations for Merging Spatial and Temporal Design Flows	76
4.2.6	Optimisation Objectives	78
4.2.7	Cost Function	79
4.3	Adaptive Mapping in Reconfigurable Multiprocessors	79
4.3.1	Reconfiguration for Application Mapping	80
4.3.2	Advantages of the Multi-dimensional Mapping Approach	85
4.4	Summary	85
5	QuadroCore: Architecture	87
5.1	Reconfiguration Design Space	88
5.1.1	Instruction to Control Reconfiguration	89
5.1.2	Synchronisation	91
5.1.3	Communication	93
5.1.4	MIMD and SIMD operation	96
5.1.5	Word-length Configurability	97
5.1.6	Additional Instructions for Co-operative Multiprocessing	99
5.1.7	Compilation Flow	99
5.2	Time and Power Characteristics	100
5.2.1	Timing Characteristics	101
5.2.2	QuadroCore Power Distribution	101

5.2.3	Time and Power variations in the Reconfiguration Design Space	103
5.3	Instruction-level Power Model	104
5.3.1	Instruction Life Cycle	106
5.3.2	Memory Accesses	107
5.3.3	Register Accesses	108
5.3.4	ALU Accesses	108
5.3.5	Multiprocessor Synchronisation	108
5.3.6	Instruction Set Characterisation	109
5.4	Impact of Compilation Techniques	112
5.5	Implementation and Performance Measurements	115
5.5.1	Standard Cell Implementation	116
5.5.2	Post-layout Implementation Reports	120
5.5.3	FPGA Reports	121
5.6	Summary	122
6	QuadroCore: Applications	125
6.1	Design Flow for Resource Efficiency	126
6.2	Applications Mapped to QuadroCore	127
6.2.1	Timing Advantage of Reconfiguration	128
6.2.2	DSP Algorithms	129
6.2.3	Multiplier used in Elliptic Curve Cryptography	132
6.2.4	Self-organising Maps	136
6.2.5	Comparison: Parallelism, Speedup, Energy	140
6.2.6	Comparable Architectures	141
6.3	Extending the QuadroCore Multiprocessor	143
6.3.1	Platform for Validating Parallel Programs	143
6.3.2	Environment for Run-time Processor Customisation	144
6.4	Summary	145
7	Conclusions and Future Work	147
7.1	Summary	148
7.2	Future Work	150
	Glossary	153
	List of Figures	158
	List of Tables	160
	References	161
	Author's Publications	171