

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Reconfigurable Computing . . . . .	2
1.2	Contribution of the Thesis . . . . .	4
1.3	Abstracting Layered Approach . . . . .	5
1.4	Outline of the work . . . . .	5
<b>2</b>	<b>Reconfigurable Computing</b>	<b>7</b>
2.1	Introduction . . . . .	7
2.1.1	Evolution . . . . .	7
2.1.2	Makimoto's Wave . . . . .	11
2.1.3	Remainder of the Chapter . . . . .	12
2.2	Technical Aspects . . . . .	12
2.2.1	Reconfigurable versus Programmable . . . . .	13
2.2.2	Granularity . . . . .	14
2.2.3	Field Programmable Gate Arrays . . . . .	15
2.2.4	Programming FPGAs . . . . .	20
2.2.5	Run-Time Reconfiguration . . . . .	22
2.2.6	Coupling . . . . .	28
2.3	Fields of Application . . . . .	29
2.3.1	ASIC Design . . . . .	29
2.3.2	Replacement of Dedicated Circuits . . . . .	30
2.3.3	Adaptive Processing . . . . .	31
2.4	Design Approaches for Reconfigurable Systems . . . . .	34
2.4.1	Execution Environments/Architectures . . . . .	34
2.4.2	Placement/Scheduling Methods . . . . .	37
2.4.3	Comprehensive Design Systems/Design Methods . . . . .	39
2.4.4	Miscellaneous Concepts . . . . .	40
2.5	Lesson Learned . . . . .	40
2.6	Summary . . . . .	42
<b>3</b>	<b>Two-Slot Framework</b>	<b>43</b>
3.1	Introduction . . . . .	43
3.1.1	Layered Approach . . . . .	45
3.1.2	Organization of the Chapter . . . . .	45
3.2	Concept . . . . .	45
3.2.1	Problem Abstraction . . . . .	46

3.2.2	Problem Solution Strategy . . . . .	48
3.3	Run-Time Architecture . . . . .	49
3.3.1	Fundamental Design Constraints . . . . .	50
3.3.2	Intermodule Communication . . . . .	51
3.4	Partitioning . . . . .	53
3.4.1	Simple Temporal Partitioning . . . . .	53
3.4.2	Spectral Based Partitioning . . . . .	54
3.5	Scheduler . . . . .	59
3.6	Experiment . . . . .	61
3.6.1	Proof of Concept Implementation . . . . .	61
3.6.2	Cryptography Example . . . . .	63
3.7	Extensions . . . . .	65
3.7.1	Low Power Considerations . . . . .	65
3.7.2	The Two Slot Framework as an IP Core . . . . .	69
3.8	Lesson Learned . . . . .	70
3.9	Related Work . . . . .	71
3.10	Summary . . . . .	73
<b>4</b>	<b>Specification Graph Approach for Reconfigurable Fabrics</b>	<b>75</b>
4.1	Introduction . . . . .	75
4.1.1	Layered Approach . . . . .	76
4.1.2	Organization of the Chapter . . . . .	76
4.2	Concept . . . . .	77
4.2.1	Problem Abstraction . . . . .	77
4.2.2	Background . . . . .	78
4.2.3	Problem Solution . . . . .	79
4.3	Problem Graph . . . . .	80
4.4	Architecture Graph . . . . .	82
4.5	Mapping . . . . .	83
4.5.1	Basic Mapping Edges . . . . .	83
4.5.2	Extensions . . . . .	84
4.6	Design Space Exploration . . . . .	85
4.7	Experiment . . . . .	91
4.8	Lesson Learned . . . . .	94
4.9	Related Work . . . . .	95
4.10	Summary . . . . .	96
<b>5</b>	<b>Reconfiguration Port Scheduling</b>	<b>97</b>
5.1	Introduction . . . . .	97
5.1.1	Layered Approach . . . . .	98
5.1.2	Remainder of the Chapter . . . . .	99
5.2	Concept . . . . .	99
5.2.1	Execution Environment . . . . .	99
5.2.2	Problem Abstraction . . . . .	100

5.2.3	Reconfiguration Port Scheduling . . . . .	101
5.2.4	Parallel Machine Problems with a Single Server . . . . .	104
5.3	Aperiodic Task Scheduling . . . . .	104
5.3.1	Synchronous Arrival . . . . .	104
5.3.2	Asynchronous Arrival . . . . .	108
5.3.3	Experimental Results . . . . .	111
5.4	Fixed Priority Periodic Task Scheduling . . . . .	111
5.4.1	Schedulability Analysis . . . . .	111
5.4.2	A Server for <i>Full Load of Slots</i> Sections . . . . .	112
5.4.3	Resource Access Protocol for <i>Full Reconfiguration Capacity</i> Sections	114
5.4.4	DM + SS + PIP Schedulability Test . . . . .	116
5.4.5	Experiment . . . . .	116
5.5	Caching . . . . .	117
5.5.1	Offline caching methods . . . . .	119
5.5.2	Dynamic/On-line Caching Methods . . . . .	122
5.5.3	Combination of the Methods . . . . .	124
5.5.4	Implementation . . . . .	125
5.6	Experiment . . . . .	125
5.7	Lesson Learned . . . . .	129
5.8	Related Work . . . . .	130
5.9	Summary . . . . .	132
<b>6</b>	<b>Algorithmic Skeletons for Dynamic Reconfiguration</b>	<b>133</b>
6.1	Introduction . . . . .	133
6.1.1	Layered Approach . . . . .	136
6.1.2	Remainder of the Chapter . . . . .	136
6.2	Concept . . . . .	136
6.2.1	Algorithmic Skeletons . . . . .	137
6.2.2	Application for Reconfigurable Computing . . . . .	139
6.3	Run-time Execution Environment . . . . .	140
6.4	Stream Parallelism . . . . .	144
6.4.1	Farm Paradigm . . . . .	144
6.4.2	Pipeline Paradigm . . . . .	145
6.5	Dynamic Reconfiguration . . . . .	147
6.5.1	Dynamic Reconfiguration on a Tile-Based Execution Environment	147
6.5.2	Dynamic Reconfiguration on a Skeleton-centric Execution Environment . . . . .	148
6.6	Experiment . . . . .	151
6.6.1	Cryptography Experiment . . . . .	151
6.6.2	Application-centric . . . . .	153
6.7	Lesson Learned . . . . .	157
6.8	Related Work . . . . .	158
6.9	Summary . . . . .	159

**7 Conclusion and Outlook** **161**  
7.1 Conclusion . . . . . 161  
7.2 Outlook . . . . . 164

**A The Design Tool Part-E** **165**

**Author's Own Publications** **169**

**Bibliography** **173**