

Contents

List of Figures	xvii
List of Tables	xix
List of Algorithms	xxi
List of Symbols	xxiii
1. Introduction	1
1.1. Motivation	2
1.2. Thesis Goals	3
1.3. Thesis Contributions	4
1.4. Thesis Outline	4
2. Background	7
2.1. Embedded System Design	7
2.2. Reconfigurable Computing Overview	9
2.2.1. Coupling CPU and Reconfigurable Hardware	11
2.2.2. Reconfigurable System Design	11
2.3. Reconfigurable Hardware Technology	13
2.3.1. Hybrid Architecture	14
2.3.2. Configuration Techniques	15
2.3.3. Partial Reconfiguration Feature	16
2.4. Chapter Conclusions	17
3. Related Work Survey	19

3.1.	(Re)Configurable Operating Systems	19
3.1.1.	Statically Reconfigurable OS	20
3.1.2.	Dynamically Reconfigurable OS: Application Triggered	21
3.1.3.	Dynamically Reconfigurable OS: System Triggered	23
3.1.4.	Towards Online Reconfigurable DREAMS	24
3.1.5.	Further Comments	24
3.2.	Operating System for Reconfigurable Computing	25
3.2.1.	Low-level OS Support for Reconfigurable Hardware	26
3.2.2.	Application Model	29
3.2.3.	OS Services for Reconfigurable Hardware	29
3.2.4.	RTOS issues in High Level Design	30
3.2.5.	Offline Approaches	31
3.2.6.	Run-time Support	32
3.2.7.	Multitasking Issues	35
3.2.8.	Dynamically Hybrid Architectures	36
3.3.	Further Approaches	37
3.3.1.	Hardware Accelerator for RTOS	37
3.3.2.	Multithreading on Hybrid Architectures	38
3.4.	Chapter Conclusions	40
3.4.1.	Correlation With This Thesis	40
3.4.2.	Additional Comments	41
4.	Run-time Reconfigurable RTOS	43
4.1.	System Overview	43
4.1.1.	Target Applications	44
4.1.2.	Target RTOS	45
4.1.3.	Instrumented OS API	46
4.1.4.	Run-time Reconfiguration Manager - RRM	49
4.2.	Hardware Architecture	49
4.3.	Design Support	50
4.4.	Chapter Conclusions	51
5.	Modeling & Problem Formulation	53
5.1.	Component Assignment	53
5.1.1.	Constraints Definition	54
5.1.2.	Objective Function Definition	55
5.1.3.	Allocation Example	56
5.2.	Reconfiguration Costs	56
5.2.1.	Temporal Specification	58
5.3.	Communication Costs	59
5.4.	Chapter Conclusions	61
6.	Run-Time Methods	63
6.1.	OS Service Allocation	63

6.1.1.	OS Service Assignment Phase	64
6.1.2.	OS Service Assignment Example	64
6.1.3.	Balance <i>B</i> Improvement Phase	65
6.1.4.	Balance <i>B</i> Improvement Example	67
6.1.5.	Reconfiguration Cost Reduction	68
6.2.	Communication-aware Allocation Algorithm	69
6.3.	Handling Reconfiguration Activities	71
6.4.	OS Component Reconfiguration	72
6.4.1.	Applying Total Bandwidth Server	74
6.4.2.	Deriving Migration Conditions	75
6.4.3.	Software to Hardware Migration	75
6.4.4.	Hardware to Software Migration	77
6.4.5.	Software Service Reconfiguration	77
6.4.6.	Hardware Service Reconfiguration	78
6.4.7.	Migrating by Preempting	78
6.5.	Schedulability Analysis	81
6.6.	OS Components Scheduling	82
6.6.1.	Partial Schedule	83
6.6.2.	Complete Schedule	84
6.7.	Chapter Conclusions	85
7.	Methods Evaluation	87
7.1.	OS Components Allocation	87
7.1.1.	OS Components Assignment	87
7.2.	Balancing Heuristic	88
7.2.1.	Reconfiguration Cost Reduction	89
7.3.	Components Reconfiguration Scheduling	91
7.4.	Communication costs reduction	92
7.5.	Chapter Conclusions	95
8.	Design Support	97
8.1.	Hardware-Software Interface Synthesis	98
8.1.1.	OS Driver Extension	99
8.1.2.	Software Interface for Reconfigurable IPs	99
8.1.3.	Integration into IFS Tool	100
8.1.4.	Further Extension for DREAMS	101
8.2.	Relocatable Tasks Design	101
8.2.1.	Unified Task Representation	102
8.2.2.	A Framework for Relocatable Task Design	104
8.3.	Chapter Conclusions	107
9.	Case Study	111
9.1.	Target OS Service	111
9.2.	Relocatable Triple-DES	113

9.3. Testbed Set Up	115
9.4. Quantitative Results	116
9.5. Chapter Conclusions	118
10. Conclusion & Outlook	121
10.1. Summary	121
10.2. Outlook	123
A. Further Evaluation Results	125
B. HW/SW Interface Generation	129
C. Hardware/Software Task Design	131
C.1. Hardware Task Controller Template	131
C.2. Sequence Graphs for Two Migration Cases	131
D. TBS Server Bandwidth Estimation	135