## CONTENTS

**Preface** ix

1 **A GENERAL VIEW** 1
1.1 Introduction 1
1.2 What does real time mean? 4
1.3 Achieving predictability 13

2 **BASIC CONCEPTS** 23
2.1 Introduction 23
2.2 Types of task constraints 25
2.3 Definition of scheduling problems 34
2.4 Scheduling anomalies 42

3 **APERIODIC TASK SCHEDULING** 53
3.1 Introduction 53
3.2 Jackson’s algorithm 54
3.3 Horn’s algorithm 58
3.4 Non-preemptive scheduling 63
3.5 Scheduling with precedence constraints 70
3.6 Summary 76

4 **PERIODIC TASK SCHEDULING** 79
4.1 Introduction 79
4.2 Timeline scheduling 84
4.3 Rate Monotonic scheduling 86
4.4 Earliest Deadline First 100
4.5 Deadline Monotonic 103
4.6 EDF with constrained deadlines 110
4.7 Comparison between RM and EDF 116
## 5 FIXED-PRIORITY SERVERS

5.1 Introduction 119
5.2 Background scheduling 120
5.3 Polling Server 121
5.4 Deferrable Server 130
5.5 Priority Exchange 139
5.6 Sporadic Server 143
5.7 Slack stealing 149
5.8 Non-existence of optimal servers 153
5.9 Performance evaluation 155
5.10 Summary 157

## 6 DYNAMIC PRIORITY SERVERS

6.1 Introduction 161
6.2 Dynamic Priority Exchange Server 162
6.3 Dynamic Sporadic Server 167
6.4 Total Bandwidth Server 171
6.5 Earliest Deadline Late Server 174
6.6 Improved Priority Exchange Server 178
6.7 Improving TBS 181
6.8 Performance evaluation 185
6.9 The Constant Bandwidth Server 189
6.10 Summary 201

## 7 RESOURCE ACCESS PROTOCOLS

7.1 Introduction 205
7.2 The priority inversion phenomenon 206
7.3 Terminology and assumptions 209
7.4 Non-Preemptive Protocol 210
7.5 Highest Locker Priority Protocol 212
7.6 Priority Inheritance Protocol 214
7.7 Priority Ceiling Protocol 226
7.8 Stack Resource Policy 234
7.9 Schedulability analysis 246
7.10 Summary 247
## Contents

8 **LIMITED PREEMPTIVE SCHEDULING** 251  
8.1 Introduction 251  
8.2 Non-preemptive scheduling 257  
8.3 Preemption thresholds 261  
8.4 Deferred Preemptions 266  
8.5 Task splitting 270  
8.6 Selecting preemption points 274  
8.7 Assessment of the approaches 279  

9 **HANDLING OVERLOAD CONDITIONS** 287  
9.1 Introduction 287  
9.2 Handling aperiodic overloads 293  
9.3 Handling overruns 316  
9.4 Handling permanent overloads 326  

10 **KERNEL DESIGN ISSUES** 349  
10.1 Structure of a real-time kernel 349  
10.2 Process states 351  
10.3 Data structures 356  
10.4 Miscellaneous 361  
10.5 Kernel primitives 366  
10.6 Intertask communication mechanisms 385  
10.7 System overhead 392  

11 **APPLICATION DESIGN ISSUES** 397  
11.1 Introduction 398  
11.2 Time constraints definition 401  
11.3 Hierarchical design 408  
11.4 A robot control example 413  

12 **REAL-TIME OPERATING SYSTEMS AND STANDARDS** 419  
12.1 Standards for real-time operating systems 419  
12.2 Commercial real-time systems 428  
12.3 Linux related real-time kernels 432  
12.4 Open-source real-time research kernels 437  
12.5 Development Tools 452
<table>
<thead>
<tr>
<th>Chapter/Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>13 SOLUTIONS TO THE EXERCISES</td>
<td>457</td>
</tr>
<tr>
<td>GLOSSARY</td>
<td>487</td>
</tr>
<tr>
<td>REFERENCES</td>
<td>497</td>
</tr>
<tr>
<td>INDEX</td>
<td>515</td>
</tr>
</tbody>
</table>
Hard Real-Time Computing Systems
Predictable Scheduling Algorithms and Applications
Buttazzo, G.
2011, XVI, 524 p., Hardcover
ISBN: 978-1-4614-0675-4