

Steve Schneider



Concurrent and Real-time Systems

The CSP Approach

 WILEY

Contents

Preface	ix
Acknowledgements	xv
Part I The language of CSP	
1 Sequential processes	3
1.1 <i>Events and processes</i>	3
1.2 <i>Performing events</i>	8
1.3 <i>Recursion</i>	13
1.4 <i>Choice</i>	20
2 Concurrency	31
2.1 <i>Alphabetized parallel</i>	31
2.2 <i>Interleaving</i>	44
2.3 <i>Interface parallel</i>	50
3 Abstraction and control flow	55
3.1 <i>Hiding</i>	55
3.2 <i>Event renaming</i>	62
3.3 <i>Sequential composition</i>	69

3.4	<i>Interrupt</i>	72
3.5	<i>Notes</i>	74

Part II Analysing processes

4	Traces	87
4.1	<i>Sequences</i>	87
4.2	<i>Trace semantics</i>	91
4.3	<i>Recursion</i>	117
4.4	<i>Testing</i>	133
4.5	<i>Congruence</i>	136
5	Specification and verification with traces	141
5.1	<i>Property-oriented Specification</i>	141
5.2	<i>Verification</i>	143
5.3	<i>Recursion induction</i>	155
5.4	<i>Case study: Distributed sum</i>	160
5.5	<i>Process-oriented specification</i>	168
6	Stable failures	173
6.1	<i>Observing processes</i>	174
6.2	<i>Process semantics</i>	178
6.3	<i>Recursion</i>	188
7	Specification and verification with failures	195
7.1	<i>Property-oriented specification</i>	195
7.2	<i>Verification</i>	197
7.3	<i>Recursion induction</i>	206
7.4	<i>Process-oriented specification</i>	209
7.5	<i>Case study: Distributed sum</i>	212
8	Failures, divergences, and infinite traces	221
8.1	<i>Observing processes</i>	221
8.2	<i>Process semantics</i>	229
8.3	<i>Recursion</i>	242
8.4	<i>Specification and verification</i>	248
8.5	<i>Recursion induction</i>	253
8.6	<i>Case study: Distributed sum</i>	256

8.7	<i>Must testing and FDI equivalence</i>	257
8.8	<i>Notes</i>	258

Part III Introducing time

9	The timed language	267
9.1	<i>Timed computational model</i>	268
9.2	<i>Transitions</i>	269
9.3	<i>Performing events</i>	270
9.4	<i>Choice</i>	275
9.5	<i>Recursion</i>	286
9.6	<i>Concurrency</i>	290
9.7	<i>Abstraction</i>	297
9.8	<i>Flow of control</i>	304
10	Timed transition systems	313
10.1	<i>Evolution</i>	313
10.2	<i>Executions</i>	315
10.3	<i>Well-timed processes</i>	327
10.4	<i>Notes</i>	329

Part IV Timed analysis

11	Semantics of timed CSP	335
11.1	<i>Timed observations</i>	336
11.2	<i>Timed failures semantics</i>	347
11.3	<i>Recursion</i>	358
11.4	<i>Testing and timed failures equivalence</i>	363
12	Timed specification and verification	369
12.1	<i>Specification</i>	369
12.2	<i>Verification</i>	375
12.3	<i>Recursion induction</i>	385
12.4	<i>Ill-timed processes</i>	388
12.5	<i>Case study: Fischer's protocol</i>	390
13	Timewise refinement	399

<i>13.1 Trace timewise refinement</i>	<i>400</i>
<i>13.2 Failures timewise refinement</i>	<i>410</i>
<i>13.3 Refinement and parallel composition</i>	<i>413</i>
<i>13.4 Case study: a railway crossing</i>	<i>431</i>
<i>13.5 FDI timewise refinement</i>	<i>436</i>
<i>13.6 Testing and timewise refinement</i>	<i>439</i>
<i>13.7 Notes</i>	<i>442</i>

Appendix A: Event-based time **447**

<i>A.1 Standard CSP and tock</i>	<i>448</i>
<i>A.2 Translating from timed CSP</i>	<i>454</i>
<i>A.3 Notes</i>	<i>465</i>

Appendix B: Model-checking with FDR **469**

<i>B.1 Interacting with FDR</i>	<i>470</i>
<i>B.2 How FDR checks refinement</i>	<i>473</i>
<i>B.3 Machine readable CSP</i>	<i>478</i>

References **485**

Notation **493**

Index **498**

Index of Processes **507**