

# Contents

Preface	xi
<b>1 Introduction</b>	<b>1</b>
1.1 What Is a Cyber-Physical System? . . . . .	1
1.2 Key Features of Cyber-Physical Systems . . . . .	2
1.3 Overview of Topics . . . . .	5
1.4 Guide to Course Organization . . . . .	7
<b>2 Synchronous Model</b>	<b>13</b>
2.1 Reactive Components . . . . .	13
2.1.1 Variables, Valuations, and Expressions . . . . .	13
2.1.2 Inputs, Outputs, and States . . . . .	14
2.1.3 Initialization . . . . .	15
2.1.4 Update . . . . .	16
2.1.5 Executions . . . . .	18
2.1.6 Extended-State Machines . . . . .	19
2.2 Properties of Components . . . . .	21
2.2.1 Finite-State Components . . . . .	21
2.2.2 Combinational Components . . . . .	22
2.2.3 Event-Triggered Components* . . . . .	24
2.2.4 Nondeterministic Components . . . . .	26
2.2.5 Input-Enabled Components . . . . .	29
2.2.6 Task Graphs and Await Dependencies . . . . .	30
2.3 Composing Components . . . . .	36
2.3.1 Block Diagrams . . . . .	36
2.3.2 Input/Output Variable Renaming . . . . .	38
2.3.3 Parallel Composition . . . . .	38
2.3.4 Output Hiding . . . . .	47
2.4 Synchronous Designs . . . . .	49
2.4.1 Synchronous Circuits . . . . .	50
2.4.2 Cruise Control System . . . . .	54
2.4.3 Synchronous Networks* . . . . .	58
Bibliographic Notes . . . . .	63

<b>3</b>	<b>Safety Requirements</b>	<b>65</b>
3.1	Safety Specifications . . . . .	65
3.1.1	Invariants of Transition Systems . . . . .	65
3.1.2	Role of Requirements in System Design . . . . .	70
3.1.3	Safety Monitors . . . . .	75
3.2	Verifying Invariants . . . . .	78
3.2.1	Proving Invariants . . . . .	78
3.2.2	Automated Invariant Verification * . . . . .	85
3.2.3	Simulation-Based Analysis . . . . .	87
3.3	Enumerative Search * . . . . .	90
3.4	Symbolic Search . . . . .	97
3.4.1	Symbolic Transition Systems . . . . .	98
3.4.2	Symbolic Breadth-First Search . . . . .	103
3.4.3	Reduced Ordered Binary Decision Diagrams * . . . . .	109
	Bibliographic Notes . . . . .	123
<b>4</b>	<b>Asynchronous Model</b>	<b>125</b>
4.1	Asynchronous Processes . . . . .	125
4.1.1	States, Inputs, and Outputs . . . . .	126
4.1.2	Input, Output, and Internal Actions . . . . .	126
4.1.3	Executions . . . . .	131
4.1.4	Extended-State Machines . . . . .	132
4.1.5	Operations on Processes . . . . .	136
4.1.6	Safety Requirements . . . . .	141
4.2	Asynchronous Design Primitives . . . . .	142
4.2.1	Blocking vs. Non-blocking Synchronization . . . . .	142
4.2.2	Deadlocks . . . . .	143
4.2.3	Shared Memory . . . . .	145
4.2.4	Fairness Assumptions * . . . . .	154
4.3	Asynchronous Coordination Protocols . . . . .	162
4.3.1	Leader Election . . . . .	163
4.3.2	Reliable Transmission . . . . .	167
4.3.3	Wait-Free Consensus * . . . . .	170
	Bibliographic Notes . . . . .	179
<b>5</b>	<b>Liveness Requirements</b>	<b>181</b>
5.1	Temporal Logic . . . . .	181
5.1.1	Linear Temporal Logic . . . . .	182
5.1.2	LTL Specifications . . . . .	189
5.1.3	LTL Specifications for Asynchronous Processes * . . . . .	193
5.1.4	Beyond LTL * . . . . .	197
5.2	Model Checking . . . . .	199
5.2.1	Büchi Automata . . . . .	200
5.2.2	From LTL to Büchi Automata * . . . . .	206
5.2.3	Nested Depth-First Search * . . . . .	212
5.2.4	Symbolic Repeatability Checking . . . . .	216

5.3	Proving Liveness*	222
5.3.1	Eventuality Properties	222
5.3.2	Conditional Response Properties	224
	Bibliographic Notes	229
<b>6</b>	<b>Dynamical Systems</b>	<b>231</b>
6.1	Continuous-Time Models	231
6.1.1	Continuously Evolving Inputs and Outputs	231
6.1.2	Models with Disturbance	241
6.1.3	Composing Components	242
6.1.4	Stability	243
6.2	Linear Systems	247
6.2.1	Linearity	248
6.2.2	Solutions of Linear Differential Equations	251
6.2.3	Stability	259
6.3	Designing Controllers	263
6.3.1	Open-Loop vs. Feedback Controller	263
6.3.2	Stabilizing Controller	264
6.3.3	PID Controllers*	269
6.4	Analysis Techniques*	276
6.4.1	Numerical Simulation	277
6.4.2	Barrier Certificates	280
	Bibliographic Notes	287
<b>7</b>	<b>Timed Model</b>	<b>289</b>
7.1	Timed Processes	289
7.1.1	Timing-Based Light Switch	289
7.1.2	Buffer with a Bounded Delay	291
7.1.3	Multiple Clocks	292
7.1.4	Formal Model	294
7.1.5	Timed Process Composition	297
7.1.6	Modeling Imperfect Clocks*	300
7.2	Timing-Based Protocols	301
7.2.1	Timing-Based Distributed Coordination	302
7.2.2	Audio Control Protocol*	305
7.2.3	Dual Chamber Implantable Pacemaker	310
7.3	Timed Automata	317
7.3.1	Model of Timed Automata	318
7.3.2	Region Equivalence*	319
7.3.3	Matrix-Based Representation for Symbolic Analysis	328
	Bibliographic Notes	338

<b>8</b>	<b>Real-Time Scheduling</b>	<b>339</b>
8.1	Scheduling Concepts	339
8.1.1	Scheduler Architecture	340
8.1.2	Periodic Job Model	341
8.1.3	Schedulability	345
8.1.4	Alternative Job Models	350
8.2	EDF Scheduling	352
8.2.1	EDF for Periodic Job Model	352
8.2.2	Optimality of EDF	356
8.2.3	Utilization-Based Schedulability Test	358
8.3	Fixed-Priority Scheduling	361
8.3.1	Deadline-Monotonic and Rate-Monotonic Policies	361
8.3.2	Optimality of Deadline-Monotonic Policy *	365
8.3.3	Schedulability Test for Rate-Monotonic Policy *	371
	Bibliographic Notes	378
<b>9</b>	<b>Hybrid Systems</b>	<b>379</b>
9.1	Hybrid Dynamical Models	379
9.1.1	Hybrid Processes	379
9.1.2	Process Composition	386
9.1.3	Zeno Behaviors	389
9.1.4	Stability	393
9.2	Designing Hybrid Systems	395
9.2.1	Automated Guided Vehicle	395
9.2.2	Obstacle Avoidance with Multi-robot Coordination	398
9.2.3	Multi-hop Control Networks *	406
9.3	Linear Hybrid Automata *	413
9.3.1	Example Pursuit Game	414
9.3.2	Formal Model	417
9.3.3	Symbolic Reachability Analysis	420
	Bibliographic Notes	430
	<b>Bibliography</b>	<b>431</b>
	<b>Index</b>	<b>439</b>