

# Contents

<b>Preface</b>	xiii
<b>0 Getting Started</b>	1
0.1 Real-Time Computing for a Mechatronic System	1
0.2 The Development System	8
0.3 Programming Languages: Their Uses and Differences	11
0.4 Getting Started with C	15
0.5 The myRIO C Library	25
0.6 The T1 C Library	27
0.7 Summary	27
Problems	29
Lab Exercise 0: Programming the Target Computer	30
<b>1 Real-Time Computing Fundamentals and Programming the High-Level User Interface</b>	39
1.1 Computer Architectures	39
1.2 Computer Realization and Packaging	42
1.3 A Programming Model for the ARM Processor	47
1.4 Numeral Systems	49
1.5 Memory and Its Contents	60
1.6 The Paper Computer	66
1.7 Real-Time Computing	70
1.8 Applications: Feedback Control and Real-Time Measurement	73
1.9 The Design Problem, Objectives, Functions, and Requirements	75
1.10 C Characters, Pointers, Strings, Floats, and Functions	78
1.11 Introducing the UI Functions of the T1 Library <span style="border: 1px solid black; padding: 2px;">lab background</span>	87
1.12 Summary	88

	Problems	90
	Lab Exercise 1: Programming the High-Level User Interface	93
<b>2</b>	<b>Efficient Design and Programming the Midlevel User Interface</b>	103
2.1	Efficiency for Real-Time Computing	103
2.2	Algorithmic Efficiency	104
2.3	Programmatic Efficiency and Efficient C Programs	109
2.4	C Operator Precedence and Associativity	122
2.5	The UI Functions of the T1 Library <span style="border: 1px solid black; padding: 2px;">lab background</span>	127
2.6	Summary	128
	Problems	129
	Lab Exercise 2: Programming the Midlevel User Interface	130
<b>3</b>	<b>Digital Communication, Signals, and Programming the Low-Level User Interface</b>	139
3.1	Information Theory and Digital Communication Fundamentals	139
3.2	Digital Signals	145
3.3	Digital Inputs and Outputs and Pull Resistors	146
3.4	Modes of Digital Communication	148
3.5	Digital Communication Standards and Protocols	151
3.6	Universal Asynchronous Receivers-Transmitters	157
3.7	I/O Communication Channels for the myRIO <span style="border: 1px solid black; padding: 2px;">lab background</span>	159
3.8	C Structures	161
3.9	Programming the myRIO Universal Asynchronous Receiver/Transmitters (UARTs) <span style="border: 1px solid black; padding: 2px;">lab background</span>	163
3.10	Programming the myRIO DIO Lines <span style="border: 1px solid black; padding: 2px;">lab background</span>	167
3.11	C Multidimensional Arrays	170
3.12	Summary	171
	Problems	172
	Lab Exercise 3: Programming the Low-Level User Interface	173
<b>4</b>	<b>Motor Control, Finite-State Machines, and Waiting for Real-Time Computing</b>	181
4.1	Modeling the Electromechanical Subsystem	182
4.2	Open-Loop Control and the Step Response	188
4.3	Driving PMDC Motors	190
4.4	Measuring Motor Position and Velocity	209
4.5	Finite State Machines	216
4.6	Waiting for Real-Time Computing	222
4.7	Summary	222

	Problems	224
	Lab Exercise 4: Finite State Machine and Open-Loop Control	228
<b>5</b>	<b>Real-Time Scheduling with Threads and Interrupts, Digital Circuits, and Mechanical Switches</b>	239
	5.1 Real-Time and Conventional Operating Systems and Kernels	239
	5.2 Real-Time Scheduling	242
	5.3 Real-Time Linux and Its Application Programming Interface	245
	5.4 Interrupts and Their Handling in Real-Time Computing	247
	5.5 Digital Circuits and Logic Gates	253
	5.6 Transistors in Digital Circuits	262
	5.7 Mechanical Switches	266
	5.8 Debouncing Mechanical Switches	268
	5.9 Summary	270
	Problems	272
	Lab Exercise 5: Introduction to Interrupts	273
<b>6</b>	<b>Digital Realization of Dynamic Systems</b>	279
	6.1 Analog Signals	280
	6.2 ADC and DAC Hardware	284
	6.3 Sampling	287
	6.4 Discrete-Time Systems	293
	6.5 Analog Input/Output of the myRIO <span style="border: 1px solid black; padding: 2px;">lab background</span>	299
	6.6 Timer Interrupts and Their Programming <span style="border: 1px solid black; padding: 2px;">lab background</span>	301
	6.7 Summary	306
	Problems	308
	Lab Exercise 6: Realizing a Discrete Dynamic System	310
<b>7</b>	<b>Closed-Loop Motor Velocity Control with a Digital Controller</b>	319
	7.1 Continuous Feedback Control Systems and Their Performance	320
	7.2 The Root Locus and Proportional Control	326
	7.3 PI Controller Design via the Root Locus	334
	7.4 Digital Control Systems	341
	7.5 Digital Velocity Control of DC Motors	348
	7.6 The Target Controller Design <span style="border: 1px solid black; padding: 2px;">lab background</span>	354
	7.7 Summary	361
	Problems	363
	Lab Exercise 7: Motor Velocity Control	364

<b>8</b>	<b>PID Motor Position Control and Path Planning</b>	373
8.1	Derivative Compensation	373
8.2	PID Controller Design	375
8.3	Motor Position Control with PID	380
8.4	Realizing a PID Controller: Causality and PIDF Controllers	383
8.5	Automatic Design of Target Motor Control <span style="border: 1px solid black; padding: 2px;">lab background</span>	385
8.6	Path Planning	391
8.7	Summary	392
	Problems	394
	Lab Exercise 8: Motor Position Control	396
<b>A</b>	<b>Target and Development Systems</b>	405
A.1	General T1 Target System	406
A.2	Specific T1 Target Systems	409
A.3	General D1 Development System	409
A.4	Specific D1 Development Systems	409
<b>B</b>	<b>Laboratory Components</b>	411
<b>C</b>	<b>Git Version Control</b>	413
C.1	Setting up Your workspace as a Git Repository	413
C.2	A Remote Git Repository	415
<b>D</b>	<b>Utility C Function Documentation</b>	417
D.1	Saving Data from C to a MATLAB Data File	417
D.2	A Table Editor	419
<b>E</b>	<b>MATLAB Functions</b>	423
E.1	Function <code>sos2header()</code> for Converting Controllers to C	423
<b>F</b>	<b>Code Repository Index</b>	425
<b>G</b>	<b>Additional Target Computer Feature Programming</b>	427
G.1	Programming the myRIO PWM Outputs	427
G.2	Programming the Target Computer AIO	428
<b>H</b>	<b>Discrete Approximations of Some Continuous Controllers</b>	431
<b>I</b>	<b>Lists of Figures and Tables</b>	433
I.1	List of Figures	433
I.2	List of Tables	439

**J Acronyms and Initialisms**

441

References

445

Index

453