

Table of Contents

Preface

xiii

Your Book Comes with Exclusive Perks - Here's How to Unlock Them xix

Part 1: Introduction to RTOSes, the Dev-Board, and Tools 1

Chapter 1: Introducing Real-Time Systems 3

What is “real-time” anyway? 4

Hardware and software roles in real-time systems 8

Defining RTOSes 11

Practical real-time systems 14

Choosing an RTOS 18

Summary 19

Chapter 2: Introducing the Development Board 21

Dev-board overview 21

Understanding the dev-board's components 24

Understanding the hardware documentation 32

Summary 36

Chapter 3: Introducing the Development-Tools 37

Using the book's instructions and example-programs 38

The IDE and the example-programs 39

The debugging tools: J-Link, Ozone and SystemView	49
Summary	62
Chapter 4: Understanding Super-Loops	63
Introducing super-loops	64
Improving the super-loop with interrupts	68
Adding a timer to the super-loop	72
Summary	82
Chapter 5: Implementing the Super-Loop	83
The super-loop example-program	84
Learning about system APIs and hardware features	97
How to develop the super-loop program	101
Summary	116
Part 2: The FreeRTOS Scheduler, Tasks, and Primary APIs	117
Chapter 6: Understanding RTOS Tasks	119
The FreeRTOS scheduler and tasks: a conceptual view	120
Choosing FreeRTOS as our RTOS	131
Hands-on with FreeRTOS's scheduler and tasks	132
Summary	148
Chapter 7: Running the FreeRTOS Scheduler	149
Creating tasks and starting the scheduler	150
Deleting tasks	158
Trying out the code	161
Understanding the scheduler	167
Task implementation and troubleshooting	176
Summary	184

Running the chapter's example-programs	186
Semaphores	188
Priority inversion (how not to use semaphores)	206
Mutexes	213
Race conditions	220
Software timers	224
Summary	231
Further reading	231

Chapter 9: Intertask Communication **233**

Understanding RTOS queues	234
Passing data through queues by value	240
Passing data through queues by reference	250
Task notifications	255
Summary	260
Further reading	260

Part 3: Using FreeRTOS to Build Systems **261**

Chapter 10: Drivers and ISRs **263**

Introducing the UART	264
Creating a polled UART driver	271
FreeRTOS-tasks vs. ISRs	278
Creating ISR-based drivers	285
Summary	300
Further reading	301

Chapter 11: More Efficient Drivers and ISRs **303**

Creating a basic DMA-based driver	304
Creating an enhanced DMA-based driver	319
Choosing a driver model	334
Using third-party libraries and drivers	340
Summary	341
Further reading	342

Chapter 12: Sharing Hardware Peripherals among Tasks **343**

Understanding how messages are sent over USB	344
Sending messages using a simple USB API	349
Creating a VCP driver, using a stream-buffer	354
Enhancing the VCP driver, for use by multiple tasks	362
Summary	382
Further reading	383

Chapter 13: Creating Loose Coupling with Queues **385**

Creating a command queue	386
Reusing a queue definition for a new target	402
Understanding queues as interfaces	405
Summary	406
Further reading	406

Part 4: System Design and Troubleshooting **409**

Chapter 14: FreeRTOS Memory Management **411**

Understanding memory allocation	412
Static and dynamic allocation of FreeRTOS primitives	418
FreeRTOS heap implementations	423
Replacing malloc and free	428

Monitoring for stack and heap problems	429
Using a memory protection unit (MPU)	433
Summary	435
Further reading	435

Chapter 15: Multi-Processor and Multi-Core Systems 437

Introducing multi-core and multi-processor systems	438
Exploring multi-core systems	440
Exploring multi-processor systems	450
Exploring inter-processor communication	453
Choosing between multi-core and multi-processor systems	458
Summary	459
Further reading	459

Chapter 16: Troubleshooting Tips and Next Steps 461

General FreeRTOS debugging-techniques	462
Troubleshooting for the debugging tools	472
Next steps	484

Part 5: Reference Information and FreeRTOS Installation 485

Appendix A: Tools Quick-Reference 487

STM32CubeIDE	487
Ozone	488
SystemView	489

Appendix B: Reference Information 491

Chapters 2 and 3: Introducing the dev-board and tools	491
Chapter 4: Understanding super-loops	496
Chapter 5: Implementing the super-loop	496
Chapters 6 and 7: RTOS tasks and the FreeRTOS scheduler	497

Chapter 10: Drivers and ISRs	498
Chapter 11: More efficient drivers and ISRs	498
Chapter 12: Sharing hardware peripherals among tasks	499
Chapter 14: FreeRTOS memory management	500
Appendix C: FreeRTOS installation	501
Appendix C: Creating FreeRTOS Projects, and Installing FreeRTOS	505
Creating IDE-projects for use in learning FreeRTOS	505
Copying and renaming projects (cloning)	506
Installing FreeRTOS and SystemView	509
Other Books You May Enjoy	533
Index	537