
Table of Contents

Preface.....	xi
1. Introduction.....	1
Embedded Systems Development	2
Compilers and Languages	2
Debugging	3
Resource Constraints	5
Principles to Confront Those Challenges	6
Prototypes and Maker Boards	8
Further Reading	9
2. Creating a System Architecture.....	11
Getting Started	12
Creating System Diagrams	13
The Context Diagram	13
The Block Diagram	14
Organigram	17
Layering Diagram	20
Designing for Change	21
Encapsulate Modules	21
Delegation of Tasks	22
Driver Interface: Open, Close, Read, Write, IOCTL	23
Adapter Pattern	25
Creating Interfaces	27
Example: A Logging Interface	27
A Sandbox to Play In	34
Back to the Drawing Board	38
Further Reading	39

3. Getting Your Hands on the Hardware.....	41
Hardware/Software Integration	41
Ideal Project Flow	42
Hardware Design	43
Board Bring-Up	45
Reading a Datasheet	46
Datasheet Sections You Need When Things Go Wrong	48
Datasheet Sections for Software Developers	50
Evaluating Components Using the Datasheet	53
Your Processor Is a Language	56
Reading a Schematic	58
Practice Reading a Schematic: Arduino!	62
Keep Your Board Safe	64
Creating Your Own Debugging Toolbox	65
Digital Multimeter	66
Oscilloscopes and Logic Analyzers	67
Setting Up a Scope	67
Testing the Hardware (and Software)	70
Building Tests	71
Flash Test Example	72
Command and Response	76
Command Pattern	79
Dealing with Errors	81
Consistent Methodology	81
Error Checking Flow	82
Error-Handling Library	83
Debugging Timing Errors	84
Further Reading	84
4. Inputs, Outputs, and Timers.....	87
Handling Registers	87
Binary and Hexadecimal Math	87
Bitwise Operations	89
Test, Set, Clear, and Toggle	91
Toggling an Output	92
Setting the Pin to Be an Output	92
Turning On the LED	94
Blinking the LED	95
Troubleshooting	95
Separating the Hardware from the Action	96
Board-Specific Header File	97

I/O-Handling Code	98
Main Loop	99
Facade Pattern	99
The Input in I/O	100
Momentary Button Press	103
Interrupt on a Button Press	104
Configuring the Interrupt	104
Debouncing Switches	105
Runtime Uncertainty	108
Increasing Code Flexibility	108
Dependency Injection	109
Using a Timer	111
Timer Pieces	111
Doing the Math	114
More Math: Difficult Goal Frequency	117
A Long Wait Between Timer Ticks	119
Using a Timer	119
Using Pulse-Width Modulation	119
Shipping the Product	122
Further Reading	123
5. Interrupts.....	125
A Chicken Presses a Button	125
An IRQ Happens	127
Nonmaskable Interrupts	128
Interrupt Priority	128
Nested Interrupts	128
Save the Context	130
Retrieve the ISR from the Vector Table	132
Initializing the Vector Table	132
Looking Up the ISR	133
Call the ISR	135
Multiple Sources for One Interrupt	137
Disabling Interrupts	138
Critical Sections	138
Restore the Context	139
Configuring Interrupts	140
When and When Not to Use Interrupts	142
How to Avoid Using Interrupts	142
Polling	143
System Tick	143

Time-Based Events	145
A Very Small Scheduler	146
Further Reading	148
6. Managing the Flow of Activity.....	151
Scheduling and Operating System Basics	151
Tasks	151
Communication Between Tasks	152
Avoiding Race Conditions	154
Priority Inversion	154
State Machines	156
State Machine Example: Stoplight Controller	156
State-Centric State Machine	157
State-Centric State Machine with Hidden Transitions	158
Event-Centric State Machine	159
State Pattern	160
Table-Driven State Machine	161
Choosing a State Machine Implementation	163
Watchdog	164
Main Loops	166
Polling and Waiting	166
Timer Interrupt	168
Interrupts Do Everything	169
Interrupts Cause Events	171
Very Small Scheduler	173
Active Objects	174
Further Reading	177
7. Communicating with Peripherals.....	181
Serial Communication	181
TTL Serial	184
RS-232 Serial	185
SPI	187
I ² C and TWI	189
1-Wire	191
Parallel	191
Dual and Quad SPI	192
USB	193
Considering Other Protocols	193
Communications in Practice	195
External ADC Example: Data Ready with SPI	196

Use a FIFO If Available	197
Direct Memory Access (DMA) Is Faster	199
External ADC Example: SPI and DMA	200
Circular Buffers	203
Further Reading	209
8. Putting Together a System.....	211
Key Matrices	211
Segmented Displays	213
Pixel Displays	215
Display Assets	215
Changeable Data? Flyweight and Factory Patterns	218
External Flash Memory	220
Display Assets	221
Emulated EEPROMs and KV Stores	223
Little File Systems	224
Data Storage	225
Analog Signals	227
Digital Sensors	229
Data Handling	231
Changing Algorithms: Strategy	232
Algorithm Stages: Pipelines and Filters	233
Calculating Needs: Speeds and Feeds	236
Data Bandwidth	236
Memory Throughput and Buffering	238
Further Reading	239
9. Getting into Trouble.....	243
Fighting with the Compiler Optimizations	244
Impossible Bugs	246
Reproduce the Bug	246
Explain the Bug	247
Creating Chaos and Hard Faults	248
Dividing by Zero	248
Talking to Things That Aren't There	249
Running Undefined Instructions	250
Incorrect Memory Access (Unaligned Access)	251
Returning a Pointer to Stack Memory	252
Stack Overflows and Buffer Overflows	253
Debugging Hard Faults	254
Processor Registers: What Went Wrong?	255

Creating a Core Dump	256
Using the Core Dump	259
Merely Very Difficult Bugs	260
Consequences of Being Clever	261
Further Reading	262
10. Building Connected Devices.....	263
Connecting Remotely	263
Directly: Ethernet and WiFi	265
Through a Gateway	266
Via a Mesh	266
Robust Communication	268
Version!	269
Checksums, CRCs, Hashes	269
Encryption and Authentication	270
Risk Analysis	271
Updating Code	272
Firmware Update Security	275
Multiple Pieces of Code	277
Fallback Lifeboat	278
Staged Rollout	279
Managing Large Systems	280
Manufacturing	282
Further Reading	282
11. Doing More with Less.....	285
Need More Code Space	286
Reading a Map File (Part 1)	286
Process of Elimination	290
Libraries	291
Functions Versus Macros: Which Are Smaller?	292
Constants and Strings	294
Need More RAM	295
Remove malloc	295
Reading a Map File (Part 2)	296
Registers and Local Variables	298
Function Chains	300
Pros and Cons of Globals: RAM Versus Stack	302
Clever Memory Overlays	302
Need More Speed	303
Profiling	304

Optimizing for Processor Cycles	308
Summary	319
Further Reading	320
12. Math.....	323
Identifying Fast and Slow Operations	324
Taking an Average	325
Different Averages: Cumulative and Median	327
Using an Existing Algorithm	329
Designing and Modifying Algorithms	332
Factor Polynomials	332
Taylor Series	333
Dividing by a Constant	335
Scaling the Input	336
Lookup Tables	338
Fake Floating-Point Numbers	345
Rational Numbers	346
Precision	347
Addition (and Subtraction)	348
Multiplication (and Division)	349
Machine Learning	351
Look Up the Answer!	352
Further Reading	352
13. Reducing Power Consumption.....	355
Understanding Power Consumption	356
Measuring Power Consumption	357
Designing for Lower Power Consumption	360
Turn Off the Light When You Leave the Room	361
Turn Off Peripherals	362
Turn Off Unused I/O Devices	362
Turn Off Processor Subsystems	363
Slow Down to Conserve Energy	363
Putting the Processor to Sleep	364
Interrupt-Based Code Flow Model	366
A Closer Look at the Main Loop	367
Processor Watchdog	369
Avoid Frequent Wake-Ups	369
Chained Processors	370
Further Reading	370

14. Motors and Movement.....	373
Creating Movement	373
Position Encoding	375
Driving a Simple DC Motor with PWM	376
Motor Control	379
PID Control	379
Motion Profiles	382
Ten Things I Hate About Motors	384
Further Reading	387
Index.....	389