

Contents

<i>List of Tables</i>	<i>xiii</i>
<i>List of Figures</i>	<i>xv</i>
<i>List of Listings</i>	<i>xvii</i>
<i>Preface</i>	<i>xxi</i>
<i>Companion Website</i>	<i>xxv</i>
<i>Acknowledgments</i>	<i>xxvii</i>

PART I ASSEMBLY AS A LANGUAGE

1

Chapter 1: Introduction	3
1.1 Reasons to Learn Assembly	4
1.2 The ARM Processor	8
1.3 Computer Data	9
1.3.1 Representing Natural Numbers.....	9
1.3.2 Base Conversion	11
1.3.3 Representing Integers	15
1.3.4 Representing Characters	20
1.4 Memory Layout of an Executing Program	28
1.5 Chapter Summary	31
Chapter 2: GNU Assembly Syntax	35
2.1 Structure of an Assembly Program	36
2.1.1 Labels.....	37
2.1.2 Comments	37
2.1.3 Directives	37
2.1.4 Assembly Instructions	38
2.2 What the Assembler Does.....	38
2.3 GNU Assembly Directives	40
2.3.1 Selecting the Current Section	40
2.3.2 Allocating Space for Variables and Constants	41
2.3.3 Filling and Aligning.....	43
2.3.4 Setting and Manipulating Symbols.....	45

2.3.5 Conditional Assembly	46
2.3.6 Including Other Source Files	47
2.3.7 Macros.....	48
2.4 Chapter Summary	50
Chapter 3: Load/Store and Branch Instructions	53
3.1 CPU Components and Data Paths	54
3.2 ARM User Registers	55
3.3 Instruction Components	58
3.3.1 Setting and Using Condition Flags	58
3.3.2 Immediate Values.....	59
3.4 Load/Store Instructions	60
3.4.1 Addressing Modes.....	61
3.4.2 Load/Store Single Register.....	64
3.4.3 Load/Store Multiple Registers.....	65
3.4.4 Swap	68
3.4.5 Exclusive Load/Store	69
3.5 Branch Instructions	70
3.5.1 Branch.....	70
3.5.2 Branch and Link	71
3.6 Pseudo-Instructions	73
3.6.1 Load Immediate	73
3.6.2 Load Address	75
3.7 Chapter Summary	76
Chapter 4: Data Processing and Other Instructions	79
4.1 Data Processing Instructions	79
4.1.1 Operand2	80
4.1.2 Comparison Operations	81
4.1.3 Arithmetic Operations	83
4.1.4 Logical Operations	85
4.1.5 Data Movement Operations	86
4.1.6 Multiply Operations with 32-bit Results	87
4.1.7 Multiply Operations with 64-bit Results	88
4.1.8 Division Operations	89
4.2 Special Instructions	90
4.2.1 Count Leading Zeros.....	90
4.2.2 Accessing the CPSR and SPSR	91
4.2.3 Software Interrupt	91
4.2.4 Thumb Mode	92
4.3 Pseudo-Instructions	93
4.3.1 No Operation	93
4.3.2 Shifts.....	94
4.4 Alphabetized List of ARM Instructions	95
4.5 Chapter Summary	96

Chapter 5: Structured Programming	99
5.1 Sequencing.....	100
5.2 Selection.....	101
5.2.1 Using Conditional Execution	101
5.2.2 Using Branch Instructions	102
5.2.3 Complex Selection	103
5.3 Iteration.....	104
5.3.1 Pre-Test Loop.....	105
5.3.2 Post-Test Loop	106
5.3.3 For Loop	106
5.4 Subroutines	108
5.4.1 Advantages of Subroutines	109
5.4.2 Disadvantages of Subroutines	110
5.4.3 Standard C Library Functions	110
5.4.4 Passing Arguments	110
5.4.5 Calling Subroutines	113
5.4.6 Writing Subroutines.....	117
5.4.7 Automatic Variables.....	118
5.4.8 Recursive Functions.....	119
5.5 Aggregate Data Types.....	123
5.5.1 Arrays	124
5.5.2 Structured Data.....	124
5.5.3 Arrays of Structured Data	126
5.6 Chapter Summary	131
Chapter 6: Abstract Data Types	137
6.1 ADTs in Assembly Language.....	138
6.2 Word Frequency Counts	139
6.2.1 Sorting by Word Frequency	147
6.2.2 Better Performance	150
6.3 Ethics Case Study: Therac-25	161
6.3.1 History of the Therac-25	162
6.3.2 Overview of Design Flaws	163
6.4 Chapter Summary	165

PART II PERFORMANCE MATHEMATICS **169**

Chapter 7: Integer Mathematics	171
7.1 Subtraction by Addition	172
7.2 Binary Multiplication	172
7.2.1 Multiplication by a Power of Two	173
7.2.2 Multiplication of Two Variables.....	173
7.2.3 Multiplication of a Variable by a Constant	177

7.2.4 Signed Multiplication	178
7.2.5 Multiplying Large Numbers	179
7.3 Binary Division	181
7.3.1 Division by a Power of Two	181
7.3.2 Division by a Variable	182
7.3.3 Division by a Constant	190
7.3.4 Dividing Large Numbers	194
7.4 Big Integer ADT	195
7.5 Chapter Summary	216
Chapter 8: Non-Integral Mathematics	219
8.1 Base Conversion of Fractional Numbers	220
8.1.1 Arbitrary Base to Decimal	220
8.1.2 Decimal to Arbitrary Base	220
8.2 Fractions and Bases	223
8.3 Fixed-Point Numbers	226
8.3.1 Interpreting Fixed-Point Numbers	226
8.3.2 Q Notation	230
8.3.3 Properties of Fixed-Point Numbers	230
8.4 Fixed-Point Operations	231
8.4.1 Fixed-Point Addition and Subtraction	231
8.4.2 Fixed Point Multiplication	232
8.4.3 Fixed Point Division	234
8.4.4 Division by a Constant	236
8.5 Floating Point Numbers	242
8.5.1 IEEE 754 Half-Precision	243
8.5.2 IEEE 754 Single-Precision	245
8.5.3 IEEE 754 Double-Precision	245
8.5.4 IEEE 754 Quad-Precision	246
8.6 Floating Point Operations	246
8.6.1 Floating Point Addition and Subtraction	246
8.6.2 Floating Point Multiplication and Division	247
8.7 Computing Sine and Cosine	247
8.7.1 Formats for the Powers of x	248
8.7.2 Formats and Constants for the Factorial Terms	249
8.7.3 Putting it All Together	251
8.7.4 Performance Comparison	259
8.8 Ethics Case Study: Patriot Missile Failure	261
8.9 Chapter Summary	263
Chapter 9: The ARM Vector Floating Point Coprocessor	265
9.1 Vector Floating Point Overview	266
9.2 Floating Point Status and Control Register	268
9.2.1 Performance Versus Compliance	271
9.2.2 Vector Mode	272
9.3 Register Usage Rules	273

9.4	Load/Store Instructions	274
9.4.1	Load/Store Single Register	274
9.4.2	Load/Store Multiple Registers	275
9.5	Data Processing Instructions	277
9.5.1	Copy, Absolute Value, Negate, and Square Root.....	277
9.5.2	Add, Subtract, Multiply, and Divide	278
9.5.3	Compare	279
9.6	Data Movement Instructions	279
9.6.1	Moving Between Two VFP Registers	279
9.6.2	Moving Between VFP Register and One Integer Register	280
9.6.3	Moving Between VFP Register and Two Integer Registers	281
9.6.4	Move Between ARM Register and VFP System Register	282
9.7	Data Conversion Instructions	282
9.7.1	Convert Between Floating Point and Integer.....	282
9.7.2	Convert Between Fixed Point and Single Precision	284
9.8	Floating Point Sine Function	285
9.8.1	Sine Function Using Scalar Mode	285
9.8.2	Sine Function Using Vector Mode	287
9.8.3	Performance Comparison.....	291
9.9	Alphabetized List of VFP Instructions	292
9.10	Chapter Summary	293
Chapter 10	The ARM NEON Extensions	297
10.1	NEON Intrinsics	299
10.2	Instruction Syntax	299
10.3	Load and Store Instructions	302
10.3.1	Load or Store Single Structure Using One Lane	303
10.3.2	Load Copies of a Structure to All Lanes	305
10.3.3	Load or Store Multiple Structures	307
10.4	Data Movement Instructions	309
10.4.1	Moving Between NEON Scalar and Integer Register.....	309
10.4.2	Move Immediate Data	310
10.4.3	Change Size of Elements in a Vector	311
10.4.4	Duplicate Scalar.....	312
10.4.5	Extract Elements	313
10.4.6	Reverse Elements	314
10.4.7	Swap Vectors	315
10.4.8	Transpose Matrix	316
10.4.9	Table Lookup	317
10.4.10	Zip or Unzip Vectors	319
10.5	Data Conversion	321
10.5.1	Convert Between Fixed Point and Single-Precision.....	321
10.5.2	Convert Between Half-Precision and Single-Precision.....	322
10.6	Comparison Operations	322
10.6.1	Vector Compare	323

10.6.2	Vector Absolute Compare	324
10.6.3	Vector Test Bits	325
10.7	Bitwise Logical Operations	326
10.7.1	Bitwise Logical Operations	326
10.7.2	Bitwise Logical Operations with Immediate Data	327
10.7.3	Bitwise Insertion and Selection	328
10.8	Shift Instructions	329
10.8.1	Shift Left by Immediate	329
10.8.2	Shift Left or Right by Variable	330
10.8.3	Shift Right by Immediate	331
10.8.4	Saturating Shift Right by Immediate	332
10.8.5	Shift and Insert	333
10.9	Arithmetic Instructions	335
10.9.1	Vector Add and Subtract	335
10.9.2	Vector Add and Subtract with Narrowing	336
10.9.3	Add or Subtract and Divide by Two	337
10.9.4	Add Elements Pairwise	338
10.9.5	Absolute Difference	339
10.9.6	Absolute Value and Negate	340
10.9.7	Get Maximum or Minimum Elements	341
10.9.8	Count Bits	342
10.10	Multiplication and Division	343
10.10.1	Multiply	343
10.10.2	Multiply by Scalar	345
10.10.3	Fused Multiply Accumulate	346
10.10.4	Saturating Multiply and Double (Low)	347
10.10.5	Saturating Multiply and Double (High)	348
10.10.6	Estimate Reciprocals	348
10.10.7	Reciprocal Step	349
10.11	Pseudo-Instructions	351
10.11.1	Load Constant	351
10.11.2	Bitwise Logical Operations with Immediate Data	352
10.11.3	Vector Absolute Compare	353
10.12	Performance Mathematics: A Final Look at Sine	354
10.12.1	Single Precision	354
10.12.2	Double Precision	355
10.12.3	Performance Comparison	357
10.13	Alphabetized List of NEON Instructions	358
10.14	Chapter Summary	361

PART III ACCESSING DEVICES**363**

Chapter 11: Devices	365
11.1 Accessing Devices Directly Under Linux	365

11.2	General Purpose Digital Input/Output	376
11.2.1	Raspberry Pi GPIO	378
11.2.2	pcDuino GPIO	382
11.3	Chapter Summary	392
Chapter 12: Pulse Modulation		395
12.1	Pulse Density Modulation	396
12.2	Pulse Width Modulation	397
12.3	Raspberry Pi PWM Device	398
12.4	pcDuino PWM Device	400
12.5	Chapter Summary	403
Chapter 13: Common System Devices		405
13.1	Clock Management Device	405
13.1.1	Raspberry Pi Clock Manager	406
13.1.2	pcDuino Clock Control Unit	409
13.2	Serial Communications	409
13.2.1	UART	410
13.2.2	Raspberry Pi UART0	413
13.2.3	Basic Programming for the Raspberry Pi UART	418
13.2.4	pcDuino UART	422
13.3	Chapter Summary	429
Chapter 14: Running Without an Operating System		431
14.1	ARM CPU Modes	432
14.2	Exception Processing	434
14.2.1	Handling Exceptions	438
14.3	The Boot Process	442
14.4	Writing a Bare-Metal Program	442
14.4.1	Startup Code	443
14.4.2	Main Program	445
14.4.3	The Linker Script	447
14.4.4	Putting it All Together	449
14.5	Using an Interrupt	449
14.5.1	Startup Code	449
14.5.2	Interrupt Controllers	449
14.5.3	Timers	458
14.5.4	Exception Handling	461
14.5.5	Building the Interrupt-Driven Program	461
14.6	ARM Processor Profiles	461
14.7	Chapter Summary	464