

Contents

Preface	vii
PART I: Overview	1
1 Introduction	3
Processor Architecture	3
RISC Versus CISC	5
What Is Assembly Language?	7
Advantages of High-Level Languages	9
Why Program in Assembly Language?	10
Summary	11
2 Processor Design Issues	13
Introduction	13
Number of Addresses	14
The Load/Store Architecture	20
Processor Registers	22
Flow of Control	22
Procedure Calls	26
Handling Branches	28
Instruction Set Design Issues	32
Summary	36
3 RISC Principles	39
Introduction	39
Evolution of CISC Processors	40
Why RISC?	41
RISC Design Principles	43
Summary	44

PART II: Architectures	45
4 MIPS Architecture	47
Introduction	47
Registers	48
Register Usage Convention	48
Addressing Modes	50
Instruction Format	51
Memory Usage	52
Summary	53
5 SPARC Architecture	55
Introduction	55
Registers	56
Addressing Modes	58
Instruction Format	59
Instruction Set	59
Procedures and Parameter Passing	69
Summary	76
6 PowerPC Architecture	79
Introduction	79
Register Set	81
Addressing Modes	83
Instruction Format	84
Instruction Set	86
Summary	96
7 Itanium Architecture	97
Introduction	97
Registers	98
Addressing Modes	100
Procedure Calls	101
Instruction Format	102
Instruction-Level Parallelism	105
Instruction Set	106
Handling Branches	112
Speculative Execution	114
Branch Prediction Hints	119
Summary	119

8	ARM Architecture	121
	Introduction	121
	Registers	123
	Addressing Modes	125
	Instruction Format	128
	Instruction Set	131
	Summary	145
	PART III: MIPS Assembly Language	147
9	SPIM Simulator and Debugger	149
	Introduction	149
	Simulator Settings	152
	Running a Program	153
	Debugging	154
	Summary	157
10	Assembly Language Overview	159
	Introduction	159
	Assembly Language Statements	160
	SPIM System Calls	161
	SPIM Assembler Directives	162
	MIPS Program Template	165
	Data Movement Instructions	165
	Load Instructions	166
	Store Instructions	167
	Addressing Modes	167
	Sample Instructions	168
	Our First Program	172
	Illustrative Examples	174
	Summary	182
11	Procedures and the Stack	183
	Introduction	183
	Procedure Invocation	186
	Returning from a Procedure	188
	Parameter Passing	189
	Our First Program	189
	Stack Implementation in MIPS	192
	Parameter Passing via the Stack	196
	Illustrative Examples	200
	Passing Variable Number of Parameters	207
	Summary	210

12 Addressing Modes	211
Introduction	211
Addressing Modes	212
Processing Arrays	214
Our First Program	217
Illustrative Examples	219
Summary	224
13 Arithmetic Instructions	225
Introduction	225
Addition	226
Subtraction	226
Multiplication	228
Division	229
Our First Program	230
Illustrative Examples	232
Summary	242
14 Conditional Execution	243
Introduction	243
Comparison Instructions	244
Unconditional Branch Instructions	246
Conditional Branch Instructions	248
Our First Program	249
Illustrative Examples	252
Indirect Jumps	259
Indirect Procedures	262
Summary	267
15 Logical and Shift Operations	269
Introduction	269
Logical Instructions	270
Shift Instructions	276
Rotate Instructions	280
Our First Program	281
Illustrative Examples	284
Summary	290
16 Recursion	291
Introduction	291
Our First Program	292
Illustrative Examples	295

Recursion Versus Iteration	303
Summary	304
17 Floating-Point Operations	305
Introduction	305
FPU Registers	306
Floating-Point Instructions	307
Our First Program	312
Illustrative Examples	314
Summary	322
Appendices	323
A Number Systems	325
Positional Number Systems	325
Conversion to Decimal	327
Conversion from Decimal	328
Binary/Octal/Hexadecimal Conversion	329
Unsigned Integers	330
Signed Integers	331
Floating-Point Representation	334
Summary	336
B Character Representation	339
Character Representation	339
ASCII Character Set	340
C MIPS Instruction Set Summary	343
D Programming Exercises	365
Bibliography	375
Index	379