

# Contents

<b>Preface</b>	<b>vii</b>
<b>PART I Overview</b>	<b>1</b>
<b>1 Assembly Language</b>	<b>3</b>
Introduction . . . . .	3
What Is Assembly Language? . . . . .	5
Advantages of High-Level Languages . . . . .	6
Why Program in Assembly Language? . . . . .	7
Typical Applications . . . . .	8
Summary . . . . .	8
<b>PART II Computer Organization</b>	<b>9</b>
<b>2 Digital Logic Circuits</b>	<b>11</b>
Introduction . . . . .	11
Simple Logic Gates . . . . .	13
Logic Functions . . . . .	15
Deriving Logical Expressions . . . . .	17
Simplifying Logical Expressions . . . . .	18
Combinational Circuits . . . . .	23
Adders . . . . .	26
Programmable Logic Devices . . . . .	29
Arithmetic and Logic Units . . . . .	32
Sequential Circuits . . . . .	35
Latches . . . . .	37
Flip-Flops . . . . .	39
Summary . . . . .	43
<b>3 Memory Organization</b>	<b>45</b>
Introduction . . . . .	45
Basic Memory Operations . . . . .	46
Types of Memory . . . . .	48
Building a Memory Block . . . . .	50

Building Larger Memories . . . . .	52
Mapping Memory . . . . .	56
Storing Multibyte Data . . . . .	58
Alignment of Data . . . . .	59
Summary . . . . .	60
<b>4 The IA-32 Architecture</b>	<b>61</b>
Introduction . . . . .	61
Processor Execution Cycle . . . . .	63
Processor Registers . . . . .	63
Protected Mode Memory Architecture . . . . .	67
Real Mode Memory Architecture . . . . .	72
Mixed-Mode Operation . . . . .	74
Which Segment Register to Use . . . . .	75
Input/Output . . . . .	76
Summary . . . . .	78
<b>PART III Linux</b>	<b>79</b>
<b>5 Installing Linux</b>	<b>81</b>
Introduction . . . . .	81
Partitioning Your Hard Disk . . . . .	82
Installing Fedora Core Linux . . . . .	92
Installing and Removing Software Packages . . . . .	107
Mounting Windows File System . . . . .	110
Summary . . . . .	112
Getting Help . . . . .	114
<b>6 Using Linux</b>	<b>115</b>
Introduction . . . . .	115
Setting User Preferences . . . . .	117
System Settings . . . . .	123
Working with the GNOME Desktop . . . . .	126
Command Terminal . . . . .	132
Getting Help . . . . .	134
Some General-Purpose Commands . . . . .	135
File System . . . . .	139
Access Permissions . . . . .	141
Redirection . . . . .	145
Pipes . . . . .	146
Editing Files with Vim . . . . .	147
Summary . . . . .	149
<b>PART IV NASM</b>	<b>151</b>
<b>7 Installing and Using NASM</b>	<b>153</b>
Introduction . . . . .	153
Installing NASM . . . . .	154

Generating the Executable File . . . . .	154
Assembly Language Template . . . . .	155
Input/Output Routines . . . . .	156
An Example Program . . . . .	159
Assembling and Linking . . . . .	160
Summary . . . . .	166
Web Resources . . . . .	166
<b>8 Debugging Assembly Language Programs</b>	<b>167</b>
Strategies to Debug Assembly Language Programs . . . . .	167
Preparing Your Program . . . . .	169
GNU Debugger . . . . .	170
Data Display Debugger . . . . .	179
Summary . . . . .	184
<b>PART V Assembly Language</b>	<b>185</b>
<b>9 A First Look at Assembly Language</b>	<b>187</b>
Introduction . . . . .	187
Data Allocation . . . . .	188
Where Are the Operands . . . . .	193
Overview of Assembly Language Instructions . . . . .	196
Our First Program . . . . .	205
Illustrative Examples . . . . .	206
Summary . . . . .	209
<b>10 More on Assembly Language</b>	<b>211</b>
Introduction . . . . .	211
Data Exchange and Translate Instructions . . . . .	212
Shift and Rotate Instructions . . . . .	213
Defining Constants . . . . .	217
Macros . . . . .	218
Our First Program . . . . .	221
Illustrative Examples . . . . .	223
When to Use the XLAT Instruction . . . . .	227
Summary . . . . .	229
<b>11 Writing Procedures</b>	<b>231</b>
Introduction . . . . .	231
What Is a Stack? . . . . .	233
Implementation of the Stack . . . . .	234
Stack Operations . . . . .	236
Uses of the Stack . . . . .	238
Procedure Instructions . . . . .	239
Our First Program . . . . .	241
Parameter Passing . . . . .	242
Illustrative Examples . . . . .	248
Summary . . . . .	252

---

<b>12 More on Procedures</b>	<b>255</b>
Introduction . . . . .	255
Local Variables . . . . .	256
Our First Program . . . . .	257
Multiple Source Program Modules . . . . .	260
Illustrative Examples . . . . .	261
Procedures with Variable Number of Parameters . . . . .	268
Summary . . . . .	272
<b>13 Addressing Modes</b>	<b>273</b>
Introduction . . . . .	273
Memory Addressing Modes . . . . .	274
Arrays . . . . .	278
Our First Program . . . . .	281
Illustrative Examples . . . . .	282
Summary . . . . .	289
<b>14 Arithmetic Instructions</b>	<b>291</b>
Introduction . . . . .	291
Status Flags . . . . .	292
Arithmetic Instructions . . . . .	302
Our First Program . . . . .	309
Illustrative Examples . . . . .	310
Summary . . . . .	316
<b>15 Conditional Execution</b>	<b>317</b>
Introduction . . . . .	317
Unconditional Jump . . . . .	318
Compare Instruction . . . . .	321
Conditional Jumps . . . . .	322
Looping Instructions . . . . .	327
Our First Program . . . . .	328
Illustrative Examples . . . . .	330
Indirect Jumps . . . . .	335
Summary . . . . .	339
<b>16 Logical and Bit Operations</b>	<b>341</b>
Introduction . . . . .	341
Logical Instructions . . . . .	342
Shift Instructions . . . . .	347
Rotate Instructions . . . . .	353
Bit Instructions . . . . .	354
Our First Program . . . . .	355
Illustrative Examples . . . . .	357
Summary . . . . .	360

---

<b>PART VI</b>	<b>Advanced Assembly Language</b>	<b>361</b>
<b>17</b>	<b>String Processing</b>	<b>363</b>
	String Representation . . . . .	363
	String Instructions . . . . .	364
	Our First Program . . . . .	372
	Illustrative Examples . . . . .	373
	Testing String Procedures . . . . .	376
	Summary . . . . .	378
<b>18</b>	<b>ASCII and BCD Arithmetic</b>	<b>379</b>
	Introduction . . . . .	379
	Processing in ASCII Representation . . . . .	381
	Our First Program . . . . .	384
	Processing Packed BCD Numbers . . . . .	385
	Illustrative Example . . . . .	387
	Decimal Versus Binary Arithmetic . . . . .	389
	Summary . . . . .	390
<b>19</b>	<b>Recursion</b>	<b>391</b>
	Introduction . . . . .	391
	Our First Program . . . . .	392
	Illustrative Examples . . . . .	394
	Recursion Versus Iteration . . . . .	400
	Summary . . . . .	401
<b>20</b>	<b>Protected-Mode Interrupt Processing</b>	<b>403</b>
	Introduction . . . . .	403
	A Taxonomy of Interrupts . . . . .	404
	Interrupt Processing in the Protected Mode . . . . .	405
	Exceptions . . . . .	408
	Software Interrupts . . . . .	410
	File I/O . . . . .	411
	Our First Program . . . . .	415
	Illustrative Examples . . . . .	415
	Hardware Interrupts . . . . .	418
	Direct Control of I/O Devices . . . . .	419
	Summary . . . . .	420
<b>21</b>	<b>High-Level Language Interface</b>	<b>423</b>
	Introduction . . . . .	423
	Calling Assembly Procedures from C . . . . .	424
	Our First Program . . . . .	427
	Illustrative Examples . . . . .	428
	Calling C Functions from Assembly . . . . .	432
	Inline Assembly . . . . .	434
	Summary . . . . .	441

<b>22 Floating-Point Operations</b>	<b>443</b>
Introduction . . . . .	443
Floating-Point Unit Organization . . . . .	444
Floating-Point Instructions . . . . .	447
Our First Program . . . . .	453
Illustrative Examples . . . . .	455
Summary . . . . .	458
<b>APPENDICES</b>	<b>459</b>
<b>A Number Systems</b>	<b>461</b>
Positional Number Systems . . . . .	461
Conversion to Decimal . . . . .	463
Conversion from Decimal . . . . .	463
Binary/Octal/Hexadecimal Conversion . . . . .	464
Unsigned Integers . . . . .	466
Signed Integers . . . . .	466
Floating-Point Representation . . . . .	469
Summary . . . . .	471
<b>B Character Representation</b>	<b>473</b>
Character Representation . . . . .	473
ASCII Character Set . . . . .	474
<b>C Programming Exercises</b>	<b>477</b>
<b>D IA-32 Instruction Set</b>	<b>485</b>
Instruction Format . . . . .	485
Selected Instructions . . . . .	487
<b>E Glossary</b>	<b>517</b>
<b>Index</b>	<b>527</b>