



*Computing That Works*

# ASSEMBLY LANGUAGE PROGRAMMING FOR THE 80386



COMPUTER PROFESSIONALS

FERNANDEZ / RUTH ASHLEY

2005-374-1

2-005-374-1

# Assembly Language Programming for the 80386



**Judi N. Fernandez**

**Ruth Ashley**

**McGraw-Hill Publishing Company**

New York St. Louis San Francisco Auckland Bogotá  
Caracas Hamburg Lisbon London Madrid Mexico  
Milan Montreal New Delhi Oklahoma City  
Paris San Juan São Paulo Singapore  
Sydney Tokyo Toronto

# Contents

<b>Preface</b>	<b>xiii</b>
<b>Chapter 1. The 80xxx Family Architecture</b>	<b>1</b>
80286 Features and Facilities	3
Memory Capacity and Virtual Memory	4
Protected Mode	4
Real Mode	8
Exceptions and Interrupts	8
80386 Features and Facilities	9
Assembly Programming Resources	9
Registers	10
Segments	16
The Instruction Set	19
Data Types	24
The Role of the Operating System and the Assembler	26
<b>Chapter 2. Programming Overview</b>	<b>27</b>
Modular Design	28
Directives vs. Instructions	28
Format	29
The HELP Program for MASM-OS/2	29
Overview of OS/2	31
Establishing the Processing Environment	32
Data Segment	32
Code Segment	33
The Exit_Proc Procedure	34

The Write_Msg_Proc Procedure	34
The End Directive	35
Preparing the Program	35
8086 Version	38
The 386   DOS Version	41
Differences in Directives	42
Differences in Operating System Services	44
Preparing the Program	44
Adding an Input Step	45
MASM-OS/2 Version	45
8086 Version	48
'386 Version	51
Looking Ahead	54
<b>Chapter 3. Basic Data Movement</b>	<b>57</b>
Moving Data	57
The Mov Instruction	58
The Xchg Instruction	59
Loading Registers	60
Loading the Effective Address	60
Loading Pointer Registers	61
Loading Flags	62
Storing Registers	62
Pushing and Popping Data	63
The Push and Pop Instructions	64
Pushing All the General Registers	67
The Pushf, Pushfd, Popf, and Popfd Instructions	68
Conversions	69
The Movsx and Movzx Instructions	70
The Cbw, Cwd, Cwde, and Cdq Instructions	70
The Xlat Instruction	71
String Moves	72
The Load and Store String Instructions	77
String Instruction Exceptions	78
System Instructions	78
<b>Chapter 4. Program Flow</b>	<b>81</b>
Using (E)IP	81
Near and Far Transfers	82
Jumps	83
Unconditional Jumps	83
Conditional Jumps	85

Jump Tables	89
Calls	92
Mechanics	93
The Call Instruction	94
The Ret Instructions (2)	96
Passing Data in the Stack	97
Comparisons	101
The Cmp Instruction	102
The Cmps Instructions	103
The Scas Instructions	103
Bit Comparisons	104
Preserving a Condition	104
Loops	104
The Loop Instruction	104
The Loopcond Instructions	106
The Rep Instructions	108
Interrupts	111
The Bound Instruction	111
Summary Program	112
<b>Chapter 5. Arithmetic without the Coprocessor</b>	<b>117</b>
Increments and Decrements	118
Addition and Subtraction without Carrying or Borrowing	119
Carrying and Borrowing	120
Creating an Addition Loop	121
Generalized Long Addition Procedure	123
Creating a Long Subtraction Procedure	124
Multiplication	125
Unsigned Multiplication	126
Signed Multiplication	129
Division	133
Unsigned Division	133
Signed Division	134
Decimal Arithmetic	137
Packed BCD Adjustments	137
ASCII Adjustments	139
ASCII Adjustments with Multiplication and Division	140
More Complex Decimal Arithmetic Procedures	142
<b>Chapter 6. Math with the Coprocessor</b>	<b>145</b>
Data Types	146
Binary Integers	146

Real Numbers	146
Packed BCD	147
Special Numeric Formats	148
Registers	148
The Register Stack	148
The Control Word Register	149
The Status Word	150
The Tag Word	151
The Instruction and Data Pointers	151
Defining Fields for Coprocessor Data	152
The Instruction Set	152
Data Transfer Instructions	152
The Nontranscendental Operations	155
Arithmetic Operands	156
The Basic Arithmetic Operations	158
Other Arithmetic Operations	159
Arithmetic Example	161
Comparisons	163
The Comparison Instructions	164
The Transcendental Instructions	166
The Constant Instructions	167
The Control Instructions	167
Synchronizing the Two Processors	167
Examining the Status Word	170
Initializing the Coprocessor	172
Other Control Instructions	172
Differences among the Coprocessors	172
<b>Chapter 7. Bit Manipulation</b>	<b>175</b>
Logical Operations	175
Turning Bits Off with And	176
Testing Bits	177
Turning Bits On with Or	177
Reversing Bits with Xor	178
Shifting Bits	179
Shifting to the Left	180
Shifting to the Right	182
Double Precision Shifts	183
Rotates	184
Bit Tests	185
Bit Scans	186
Setting and Clearing Flags	187

<b>Chapter 8. Defining and Using Data</b>	<b>189</b>
Forming Symbols	190
Values	191
Constants	191
Expressions	196
Equates	201
The Difference between Equ and =	202
Variables	203
Defining Variables	203
The Dup Operator	204
Data Structures	204
Structure Definitions	205
Structure Declarations	206
Declaring Multiple Structure Occurrences	206
Referencing Fields in Structures	207
Records	207
Defining Records	207
Record Declarations	208
Addressing Records and Fields	209
The Mask Operator	210
The Width Operator	210
Labels	211
Global Symbols	211
Making Symbols Public	211
Using Global Symbols	211
Local Symbols	212
<b>Chapter 9. Macros</b>	<b>215</b>
Defining a Macro	216
Using Macros	216
Comments in Macros	217
Using Parameters in Macros	218
The Macro Operators	219
Local Symbols	222
Repeat Blocks	223
Macro Libraries	225
The Include Directive	225
The Purge Directive	226
Nested Macros	226
Advantages and Disadvantages of Macros	228

<b>Chapter 10. Conditional Assembly and Errors</b>	<b>231</b>
Conditional Assembly	231
Conditions Based on Expressions	234
Conditions Based on Definitions	236
Conditions Based on Strings	236
Conditions Based on String Comparisons	238
Exiting the Macro Early	239
Forcing Assembly Errors	240
Forcing an Unconditional Error	241
Conditional Errors	241
Nested Conditions	241
MASM's Pass Conditionals	243
<b>Chapter 11. Assembler Control Directives</b>	<b>245</b>
Instruction Sets	245
Listing Control	247
Titles	249
Paging	250
Contents	252
Module Names	253
Location Counter	255
Origin	256
Alignment	256
Assembler Messages	258
Program and Segment Structure	259
Defining Program Structure (MASM Only)	259
Shorthand Segment Definitions	261
Full Segment Definitions (Both MASM and 386   ASM)	263
Grouping Segments	269
Shorthand Segment Assumption (MASM Only)	271
<b>Chapter 12. Assembling</b>	<b>273</b>
Assembling with 386   ASM	273
Assembling Multiple Files	274
Switches	275
Assembling with MASM	279
Displaying Command Help	281
Overriding the Object File	281
Controlling the Listing File	282
Controlling the Cross Reference Listing	283
Controlling Segment Order	283
Defining Symbols on the Command Line	283

Controlling the Message Display	285
Case Sensitivity	286
Setting up the Assembly Environment	286
Setting up Debugging Information	288
Using the Prompted Version of MASM	288
The Assembler Listing	288
Using CREF (MASM Only)	291
<b>Chapter 13. Link Editing</b>	<b>295</b>
Using 386   LINK	295
Displaying a Command Summary	297
Working with the Map File	297
Controlling the Linker Environment	300
Controlling the Program File	301
Streamlining 386   LINK Commands	303
Using Microsoft LINK	304
LINK Command Format	304
Using the Prompted Command	306
Using a Response File	307
Specifying the Object Files	307
Specifying the Run File	308
Manipulating the Map File	309
Using Libraries	312
Using Definition Files	314
Controlling the Job	314
Fixup Error Messages	314
<b>Chapter 14. Introduction to Online Debuggers</b>	<b>317</b>
Introduction to MINIBUG	318
Sample Program	318
Getting Ready to Use MINIBUG	318
Some Basic MINIBUG Commands	325
Other MINIBUG Commands	332
Introduction to CodeView	333
Preparing for CodeView	333
Starting CodeView	334
Getting Online Help	335
Controlling CodeView	336
Other CodeView Commands	344
<b>Appendix A. Instruction Reference</b>	<b>347</b>
<b>Appendix B. System Instructions</b>	<b>365</b>

xii Contents

Appendix C. Coprocessor Instruction Reference 369

Index 379