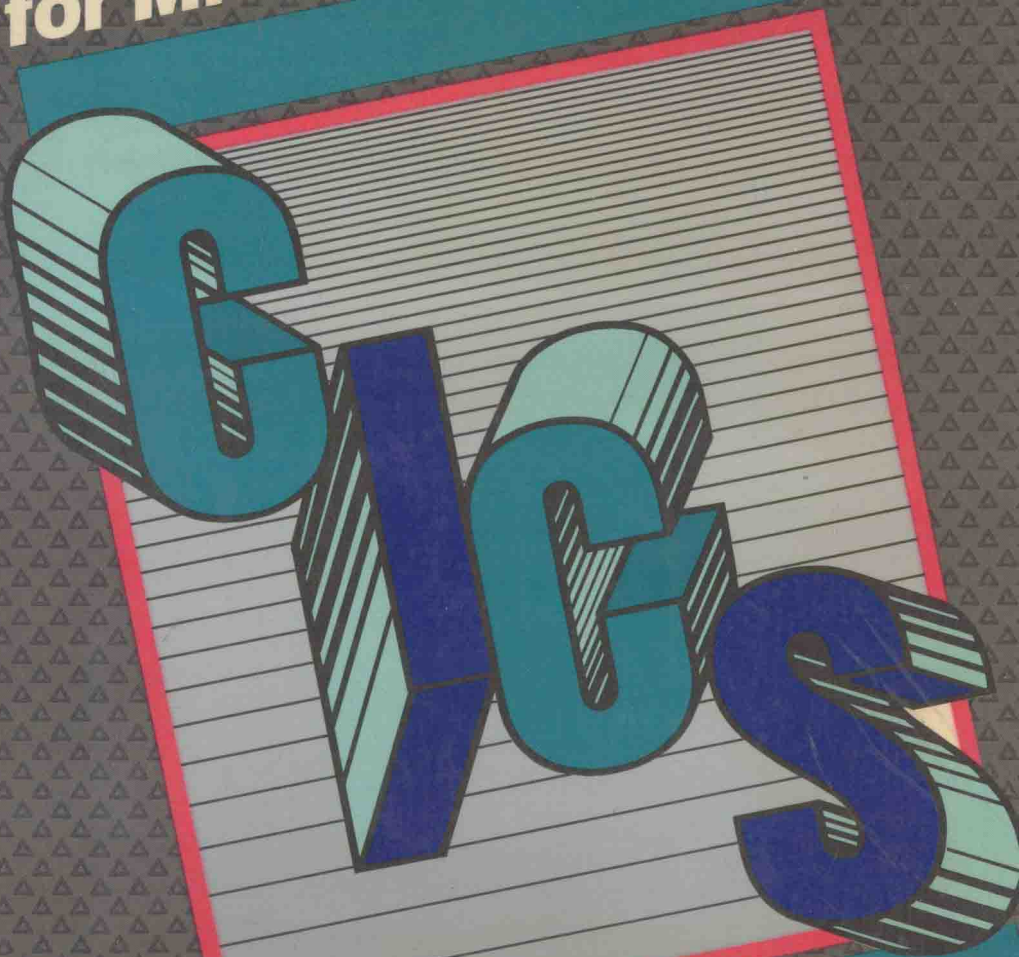




Computing That Works

CICS for Microcomputers



Joseph LeBer

Customer

Information

Control

Systems

for microcomputers



Joseph J. Le Bert

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

Le Bert, Joseph J.

CICS for microcomputers.

Rev. ed. of: Customer information control systems
made easy. c1986.

Bibliography: p.

Includes index.

1. CICS (Computer system) I. Le Bert, Joseph J.
Customer information control systems made easy.

II. Title.

QA76.76.T45 1989 005.4'3 88-27137

ISBN 0-07-036968-2

Copyright © 1989, 1986 by McGraw-Hill, Inc. All rights reserved. Printed
in the United States of America. Except as permitted under the United
States Copyright Act of 1976, no part of this publication may be
reproduced or distributed in any form or by any means, or stored in a
database or retrieval system, without the prior written permission
of the publisher.

This book was previously published under the title *CICS Made Easy*.

1234567890 DOC/DOC 89432109

ISBN 0-07-036968-2

*The editors for this book were Theron Shreve and Lucy Mullins, the
designer was Naomi Auerbach, and the production supervisor was Richard
A. Ausburn. It was set in Century Schoolbook. It was composed by the
McGraw-Hill Publishing Company Professional Reference Division
composition unit.*

Printed and bound by R. R. Donnelley & Sons Company.

An excerpt from Form 3270 Information Display System Layout Sheet
(GX27-2951) is reprinted by permission from International Business
Machines Corporation and is used in the following figures: 2-1 through
2-10, 6-3, 13-1, 14-1, and 15-1.

Screen images displayed in appendixes E, F, and G are reprinted by
permission respectively from Realia, Inc.; Micro Focus, Innovative
Solutions, and Phaser Systems, Inc.; and Triangle Software Company.

Information contained in this work has been obtained by
McGraw-Hill, Inc., from sources believed to be reliable. However,
neither McGraw-Hill nor its authors guarantees the accuracy or
completeness of any information published herein and neither
McGraw-Hill nor its authors shall be responsible for any errors,
omissions, or damages arising out of use of this information. This
work is published with the understanding that McGraw-Hill and its
authors are supplying information but are not attempting to render
engineering or other professional services. If such services are
required, the assistance of an appropriate professional should be
sought.

*For more information about other McGraw-Hill materials,
call 1-800-2-MCGRAW in the United States. In other
countries, call your nearest McGraw-Hill office.*

*To my wife, Maryann, who encouraged me,
and to my children, Joseph, Dorothy, and Jean,
who inspired me.*

PREFACE

CICS for Microcomputers is for you if you wish to gain a strong foundation in COBOL command-level CICS (Customer Information Control Systems) design, coding, and concepts. CICS which was originally available only on IBM mainframe computers can now be executed on microcomputers using software that makes it *highly* compatible with mainframe CICS. All program examples in this text were translated, compiled, and debugged on an IBM PC/XT using CICS software available from Realia, Inc.; Micro Focus; and Triangle Software Company. The programs ran exactly as intended with only minor *design* modifications. They should be equally compatible with any additional microcomputer CICS software that becomes available. The vendors supply screen generators that make the creation of basic mapping support (BMS) code easy, and they have debugging tools that are easy to use. These tools consist of interactive source code debuggers and a microcomputer version of the mainframe's execution diagnostic facility (EDF). Application programmers, system analysts, and project leaders should find *CICS for Microcomputers* easy to read, as well as an ideal reference book. It can be used as a textbook in a one-semester course for information processing, business, and computer science majors. Familiarity with COBOL, 3270 terminals or microcomputer monitors, data processing concepts, and the DOS operating system is the only prerequisite to understanding this book. Knowledge of the mainframe's virtual storage access methods (VSAM) or the microcomputer's indexed sequential access methods (ISAM) is helpful but not required. The use of ISAM in CICS application programs is similar in a microcomputer environment to the use of VSAM on mainframes.

The original intent in preparing this book was to substantially rewrite parts of my text *CICS Made Easy* in order to reflect differences between the use of CICS on a mainframe and on a micro. The main differences that exist are due to restrictions of the DOS operating system that prevent multitasking and multiterminal support. The OS/2 operating system will eliminate some of these restrictions and should

make microcomputer CICS even more compatible with its mainframe counterpart. The use of program function keys on a micro simulates the use of the corresponding keys on a mainframe. For instance, F1 is used in place of PF1, and Ctrl-End might be used to erase to the end of a field. The stated objective of software vendors currently marketing CICS for microcomputer use is to achieve as high a level of compatibility with mainframe CICS as possible. With this objective in mind it was decided that this book would contain most of the material presented in *CICS Made Easy*. That book was written for mainframe computer users. Consult the CICS user's guide for your microcomputer software to note any minor differences that are due mainly to the lack of multitasking and multiterminal support.

A new chapter has been added that describes the color attributes which are likely to be used in a microcomputer environment. A second new chapter was added that gives a brief introduction to additional commands and concepts that are supported by microcomputer CICS software. This latter chapter also discusses numeric editing considerations and contains code for a flexible edit module that should satisfy most of your editing needs. That module was developed and tested using Micro Focus's PC-CICS but should work equally well using Realia's Real-CICS or Triangle's CICS/pc; it will also work on a mainframe computer. Several appendixes have been added that should be of interest to both current and potential microcomputer users of CICS software. The ease of program development, simulation, and testing on a micro should improve productivity and the quality of CICS systems. System prototyping prior to starting detailed system design and program development should become commonplace. Personnel trained in the mainframe use of CICS will likely use CICS on micros, while those who learn CICS on a micro may want to use it on a mainframe. Most mainframe command-level COBOL CICS applications that access only VSAM files can be run using existing microcomputer software with *no* changes required.

Training costs and employees' learning curves should be reduced as those new to CICS are able to use the theory they learn immediately on a micro without the fear of *crashing* a mainframe system. Ambitious employees may be encouraged and motivated to improve their CICS skills on their own microcomputers. The quality of design and programming is likely to improve, and the number of persons knowledgeable in CICS is sure to expand rapidly. This will increase the supply of individuals who are able to implement the transition of a mainframe system to a more flexible interactive microcomputer system.

Extensive use of CICS on microcomputers is expected for both program development and testing. The ability to transport CICS code between mainframes and micros with little or no change required will increase the appeal of microcomputer development and testing. Any mainframe programmer who has *crashed* a CICS test region due to a bug in his or her program will appreciate the use of micros for testing. When a mainframe CICS region crashes, *all* users of that region are disabled and are unable to proceed until technical support personnel restart CICS. Data processing managers and technical support personnel, who are under pressure to bring up a crashed region, will feel a sense of relief as testing and debugging shifts to micros. The ability to translate and compile programs without being placed at the bottom of the list of a mainframe queue AWAITING EXECUTION will please programmers and make them more productive. The ability to easily

print screen images should be welcomed by designers and programmers who do not have ready access to a nearby printer. Technical support personnel will be able to respond more easily to requests for program control table (PCT), processing program table (PPT), and file control table (FCT) entries. These entries will have been pretested and can be presented to the systems programming group as a unit instead of having to be submitted as individual entries when they are needed.

Online systems which send, receive, and process data via a terminal are here to stay and will soon exceed the number of batch systems in many companies. CICS makes it easy for a programmer to code instructions that facilitate online functions. The more you learn about CICS, the better prepared you will be to play a significant role in the transition from batch to online systems.

This book takes a very simple nuts-and-bolts approach toward CICS commands and concepts. It does not attempt to convey every detail of CICS; many of these details are confusing to the beginner. I believe that once you master the material in this book, you will be prepared to develop and maintain most CICS programs. The initial foundation gained from this book will provide you with the knowledge and confidence to pursue advanced studies in CICS. You will not be overwhelmed with terminology at the start—only those terms and concepts which will actually be used are introduced as needed. A person unfamiliar with CICS will benefit most from this book by reading it from start to finish. Experienced CICS programmers can read those chapters which cover the material that they wish to review.

CICS for Microcomputers illustrates techniques and concepts through numerous simple examples and programs. Designers and programmers using this text should become productive very quickly. The programs in the examples employ a simple structured programming style which eliminates most GO TO's without confusing you with numerous nested IF statements. The examples can serve as models for future programs and should result in increased programmer productivity.

Maintenance should be reduced because most of the program code is reusable. If you have a good model to begin with, it should comprise as much as 80 percent of the code required for similar programs. It follows that if you start each program with a substantial percent of its code pretested, program development should be quicker and easier.

Many CICS concepts and examples in the text are associated with COBOL commands and techniques with which you are familiar. The best way to learn anything is through examples and use; you will master these principles quite easily by using them when writing and testing your own programs.

Chapter 1 defines a typical CICS system as viewed by a user and introduces basic terms and concepts.

Chapter 2 is devoted to designing maps which are easy to work with. Map design and conversational flow from screen to screen play an important role in the success or failure of an online system. Several map design guidelines are presented to make your screen design more effective.

Pseudoconversational programming, a technique which minimizes computer main storage requirements, is explained in Chap. 3 and is used for all programs throughout the text. System and programming documentation standards, discussed in Chap. 4, are essential to the success of a CICS system. Emphasis has been

placed on those standards and documentation methods which are most effective and easiest to create and maintain.

Chapter 5 presents short easy-to-understand programs which build a foundation for the more comprehensive programs introduced in later chapters.

Chapter 6 on basic mapping support (BMS) explains how assembler language macros are coded in order to create maps: BMS maps are included with program examples. Chapter 7 is devoted to the use of assembled BMS maps. It is easy to create BMS maps when you have models to use for new maps, but their use can cause much difficulty if you do not understand certain fundamentals. Map attribute usage and its relation to physical and symbolic maps are covered extensively. You will probably use a map generator for creating your own BMS maps, but it is important to understand the basic BMS entries.

Program control commands, covered in Chap. 8, aid in the design of programs which communicate with one another. They are necessary in order to design effective systems and to develop efficient programming techniques. Chapter 11 on file control presents and explains all the commands that you are likely to use. File control is easier in CICS programs than is the case in a batch environment.

Chapters 9, 10, 12, 13, 14, 15, and 17 are devoted, with complete examples, to the following programs, respectively: menu, linked, inquiry, addition, change, deletion, and browse. Commands and concepts which have been presented are discussed on a section-by-section basis. All programs function as a part of a simplified system, enabling you to see the relation of the parts to the whole. This should prepare designers for new system development. High-quality work is more important than ever in online systems because it is difficult to conceal poor design and programming in an online environment.

Temporary storage, for which creative programmers will find many applications, is covered in Chap. 16. Several paging techniques, which can be coded with or without using temporary storage, are discussed. The browse program uses temporary storage queues for its paging routines. Routines used to accomplish forward and backward paging are presented. Methods for resetting a browse starting point and for returning a selection to an invoking program are also explained.

Chapter 18 on debugging and testing concentrates on the execution diagnostic facility which is available with all command-level CICS mainframe systems. Appendix G explains microcomputer software that simulates the use of mainframe EDF. Techniques are explained that can decrease the amount of time required to debug CICS programs.

Chapter 19 explains the use of color and other attribute extensions that are likely to be used for microcomputer CICS systems because of the availability of color monitors. The use of these attributes with CICS is the same on microcomputers as on mainframe computers.

Chapter 20 gives a brief introduction to additional CICS commands and concepts that are available to microcomputer users. The `FORMATTIME` command is covered in more detail, and possible applications of its use are explained. This chapter discusses numeric editing and contains code for a flexible module that should satisfy most of your editing needs.

Appendix C briefly reviews some basic DOS commands and concepts, such as the use of batch files, which may be helpful in a microcomputer environment.

Appendix D presents more detail on the following CICS tables: PCT, PPT, and FCT. This material should be helpful to applications programmers who may need to become more familiar with the basic entries in these tables in order to use CICS in a microcomputer environment.

Appendix E highlights the features of Realia's RealCICS software, App. F that of Micro Focus's PC-CICS, and App. G that of Triangle Software Company's CICS/pc. These companies are working to improve the early releases of their software which already provide an impressive simulation of mainframe CICS. Each company's stated goal is to make the software as compatible with mainframe CICS as is possible within the constraints of the microcomputer's operating system. The companies also provide some easy-to-use development and testing tools which should increase overall CICS development and maintenance productivity.

Appendix H contains the source code for the DEMO system's initialization screen. This program will also initialize the common work area (CWA) for the software which supports this feature. Appendix I contains complete BMS maps for program examples in this text that do not include the full code for their maps. Appendix J is a quick reference to CICS command formats and to the chapter in which they are explained.

Online systems have been around for many years. The extent of their application over the next few years should far exceed all that has been done in the past. Without knowledgeable and competent data processing personnel and users, this transition will not occur smoothly. I hope that this book will be successful in preparing you for the role you will play in putting your company on line.

Joseph J. Le Bert

ACKNOWLEDGMENTS

CICS for Microcomputers contains much information gathered through experience and contact with knowledgeable associates. I am grateful for the help obtained from the numerous proofreaders of my original CICS text, *CICS Made Easy*. I would like to thank the highly professional personnel at Micro Focus; Realia, Inc.; and Triangle Software Company for assisting me in evaluating their CICS software for this book. Most of all I want to thank my wife Maryann and children Joseph R., Dorothy, and Jean for their patience and understanding for the countless times that I was home but not with them.

About the Author

Joseph J. Le Bert has played a major role in the development, programming, and maintenance of numerous on-line and batch systems while functioning as a consultant, project leader, and senior programmer analyst at several companies. He predicts that the use of online interactive systems will grow rapidly, replacing many obsolete batch systems. He is the author of *CICS MADE EASY*, published by McGraw-Hill in 1986, and coauthor with James B. Massoni of *Advanced Interactive Cobol for Micros: A Practical Approach* published by Prentice-Hall in 1988.

CONTENTS

Preface	xv
Acknowledgments	xxi

Chapter 1	The User's View of a CICS System	1
	Terminal Keyboard Usage	1
	Attention Identifier (AID) Keys	1
	Miscellaneous Keys	2
	Screen Format	2
	Screen Classifications	2
	Types of CICS Systems	4
	Test System	4
	Production System	4
	Promotion from Test to Production	4
	User Involvement	5
	Getting on the CICS System	5
	Summary of Sign-on Procedure	5
	Initiating a CICS Transaction	6
	Logging off CICS	6
	Typical CICS Session	6
	Purchasing System Menu	6
	Erroneous Menu Entries	7
	Valid Operator Menu Selection	8
	Ending a Session	9
	Inquiry Function	10
	Maintenance Functions	10
	Additions	11
	Changes	13
	Deletions	14
	Browse Function	15
	Summary	16

Chapter 2	Designing User-Friendly Maps	17
	Designing Maps	17
	Map Design Guidelines	19
	Consistency in Map Design	20
	Map Attributes	21
	Map Body and Instructions	22
	Designing the Control Map	23
	Body of the Control Map	23
	Entering Data into a Key Field	24
	Instructions on the Control Map	25
	Messages Displayed on the Control Map	25
	Detail Maps	26
	Inquiry Map	26
	Addition Map	27
	Change Map	28
	Deletion Map	29
	Browse Map	30
	Summary	30
 Chapter 3	 Background Terminology and Considerations	 33
	Pseudoconversational Programming	33
	Conversational Programming	34
	Master Terminal Commands	34
	CICS Tables	35
	Program-Control Table	35
	Processing Program Table	35
	Multiple PCT Entries	37
	CICS Command Format	38
	Task	38
	CICS Multitasking and Multithreading	39
	Program Considerations and Restrictions	41
	Considerations	41
	Restrictions	41
	System and Program Security	41
	System Security	41
	Program Security	42
	Summary	43
 Chapter 4	 System and Programming Documentation Standards	 45
	System Design Standards	45
	Structure Charts	45
	Programming Standards	47
	Program Structure Charts	47
	Program Comments	47
	Keep Programs Short and Simple	48
	Different Levels of Menus	49
	Passwords	49
	Message File	50
	Program Tables	51
	Tables Defined in a Program	51

	Tables Loaded into Main Storage	51
	VSAM Table Files	51
	Summary	52
Chapter 5	Basic Program Examples	53
	Send a Message to a Terminal	53
	SEND Command	53
	RETURN Command	54
	Compiling a CICS Program	54
	Running a Compiled Program	56
	Receive a Message	57
	HANDLE CONDITION Commands	59
	General Information about Handle Conditions	62
	Cursor Position (EIBCPOSN)	62
	Summary	64
Chapter 6	Basic Mapping Support	65
	Map Types	65
	Physical Maps	65
	Symbolic Maps	66
	Format and Structure	66
	BMS Format Rules	67
	Map Structure	68
	BMS Map for a Menu	73
	DFHMD Macro Usage	73
	Refresh the PPT When a Map Changes	76
	Command-Level Interpreter	77
	The Generated Symbolic Map	77
	Creating User-Friendly Symbolic Maps	79
	Summary	80
Chapter 7	Using Assembled BMS Maps	81
	Send Map Command	81
	Types of SEND Commands	82
	SEND MAP AND DATA	83
	SEND MAP MAPONLY	83
	SEND MAP DATAONLY	84
	Symbolic Cursor Positioning	84
	SEND MAP Transaction ABEND	85
	Disabling a Program	85
	Low-Values Are Not Transmitted	85
	RECEIVE MAP Command	86
	Map Attributes	86
	Standard Attribute Characters	87
	Changing Map Attributes	88
	Physical Map Strings	88
	Physical Map Interaction with Symbolic Map	90
	Manipulating MDTs	91
	FRSET	91
	Attribute Usage Summary	92

	RECEIVE MAP Considerations	93
	Length Field on Map Receive	93
	MAPFAIL Conditions	94
	Testing Data-Entry AID Keys	95
	Summary	99
Chapter 8	Program Control	101
	Program Logical Levels	101
	Task Number Considerations	106
	Communications Area	106
	RETURN Command	108
	Return with a TRANSID	108
	XCTL Command	109
	LINK Command	110
	LOAD Command	111
	RELEASE Command	112
	Summary	112
Chapter 9	The Menu Program	115
	Vendor File Maintenance Submenu	116
	Working-Storage Section	116
	Linkage Section	120
	Procedure Division	120
	Summary	125
Chapter 10	The Linked Program	127
	Linking Routine	127
	E121-POST-JOURNAL RECORD	128
	Journal-Posting Program	128
	Linkage Section	128
	Procedure Division	130
	Summary	133
Chapter 11	File Control	135
	Opening and Closing Files	136
	File Control Commands	136
	Options Common to Many File Control Commands	136
	READ Command	137
	WRITE Command	137
	READ FOR UPDATE Command	138
	REWRITE Command	138
	Exclusive Control	139
	UNLOCK Command	139
	DELETE Command	139
	FILE BROWSE Commands and Considerations	140
	STARTBR Command	141
	Browse Inefficiencies	141
	READNEXT Command	141

	Generic Search	142
	READPREV Command	143
	End Browse (ENDBR) Command	143
	Reset Browse (RESETBR) Command	144
	Summary	144
Chapter 12	The Inquiry Program	145
	BMS Maps	145
	BMS Control Map	145
	BMS Detail Map	146
	User-Friendly Symbolic Maps	147
	Inquiry Program Source Code	150
	Working-Storage Section	150
	Linkage Section	150
	Procedure Division	150
	Summary	161
Chapter 13	The Addition Program	163
	BMS Maps	163
	BMS Control Map	163
	BMS Detail Map	164
	User-Friendly Symbolic Maps	165
	Addition Program Source Code	165
	Procedure Division	165
	Summary	179
Chapter 14	The Change Program	181
	BMS Maps	181
	BMS Control Map	181
	BMS Detail Map	182
	User-Friendly Symbolic Maps	182
	Change Program Source Code	182
	Procedure Division	182
	Summary	194
Chapter 15	The Delete Program	195
	BMS Maps	195
	BMS Control Map	195
	BMS Detail Map	196
	User-Friendly Symbolic Maps	196
	Delete Program Source Code	196
	Procedure Division	196
	Summary	198
Chapter 16	Temporary-Storage Control	207
	Relating TS Queues to VSAM Files	207
	Required Working-Storage Fields	208

	Write a TS Queue	209
	Rewrite a TS Queue	209
	Read a TS Queue	210
	Delete a TS Queue	211
	Using Single Queue Records	211
	TS Paging Logic Techniques	211
	Write the Entire Symbolic Map to Temporary Storage	212
	Write Only Significant Map Data to Temporary Storage	212
	Write Only Record Keys to Temporary Storage	212
	Read All Records before Displaying Any Maps	213
	Paging Logic Techniques Without Using Temporary Storage	213
	TS Random Access Methods	213
	Summary	214
Chapter 17	The Browse Program	217
	BMS Browse Map	217
	User-Friendly Symbolic Maps	218
	Browse Program Source Code	219
	Procedure Division	219
	Summary	233
Chapter 18	CICS Debugging and Testing	235
	Execution Diagnostic Facility	235
	Program Initiation	236
	Program Initiation Screen	236
	ABOUT-TO-EXECUTE Command Screen	238
	Program Map Display Screen	239
	Command Execution Complete Screen	239
	Working-Storage Display Screen	240
	RETURN EDF Screen	241
	Program Termination Screen	241
	Task Termination Screen	242
	Program Initiation Redisplayed	243
	EDF Stop Conditions Screen	243
	Program ABEND Display	244
	ABEND a CICS Task	245
	Interactive Testing Tools	246
	CICS Dumps	246
	CICS ABEND Determination Aids	246
	CICS Testing Techniques	246
	Summary	247
Chapter 19	Color and Reverse-video Attribute Extensions	249
	Extended Attributes	249
	EXTATT Operand and Parameters	250
	The COLOR Operand	252
	The HIGHLIGHT Operand	253
	Extended Attribute Symbolic Maps	254
	Summary	256