



# Advanced Techniques in Framework<sup>TM</sup> Programming in FRED



Alan Simpson

# ***Advanced Techniques in Framework:<sup>TM</sup>***

---

## ***Programming in FRED***

***Alan Simpson***



***Berkeley • Paris • Düsseldorf • London***

Cover design: Sylvia Giblin  
Design and layout: Ingrid Owen

dBASE II, dBASE III, Framework, FRED, and MITE are trademarks of Ashton-Tate.  
IBM, PC, PC-DOS, and XT are registered trademarks of International Business Machines Corp.  
Okidata is a registered trademark of Okidata Corporation.  
WordStar is a registered trademark of MicroPro International.  
SYBEX is not affiliated with any manufacturer.

Every effort has been made to supply complete and accurate information. However, SYBEX assumes no responsibility for its use, nor for any infringements of patents or other rights of third parties which would result.

Copyright ©1985 SYBEX Inc., 2344 Sixth Street, Berkeley, CA 94710. World rights reserved. No part of this publication may be stored in a retrieval system, transmitted, or reproduced in any way, including but not limited to photocopy, photograph, magnetic or other record, without the prior agreement and written permission of the publisher.

Library of Congress Card Number: 85-61339  
ISBN 0-89588-246-9  
Printed by Haddon Craftsmen  
Manufactured in the United States of America  
10 9 8 7 6 5 4 3 2 1

***Advanced Techniques  
in Framework:***  

---

***Programming in FRED***

***To Carole***

# ***Acknowledgments***

---

Every book is a team effort, and once again the people at SYBEX were a great team to work with. Many thanks to Carole Alden, Acquisitions Editor; Jim Compton, Manuscript Editor; Jeremy Elliot, Technical Review; Brenda Walker, Proofreading; Ingrid Owen, Artwork and Design; Donna Scanlon, typesetting.

Thanks to Doug Hergert, most of the text in Appendices A and B is his.

Many thanks to Bill and Cynthia Gladstone, my literary agents.  
And thanks to Julie, for patience.

# *Introduction*

---

Whenever I tell someone about the hottest new microcomputer programming language since dBASE II, they always interrupt and exclaim, “A programming language named Fred ?!”

The authors of the FRED language weren’t trying to be cute when they decided on the name. It’s an acronym for FRamework EDitor, and FRED is a programming language that allows full programming capability within the environment of the Framework integrated software system.

And what a language FRED is! If you have any programming experience at all, you’re sure to find FRED to be a real boost to your programming creativity. And if you’ve never written a program in your life, you’ll find FRED to be an excellent first language, because it provides an enormous amount of programming power without a lot of “nitty-gritty” technical detail. FRED is a language that allows even the novice programmer to perform feats of magic that were previously reserved for the professional programmer.

## *Why Program in FRED?*

Framework is a very powerful and easy-to-use system, and you can certainly do a lot with it without ever writing a single FRED program. But there are several reasons why you might want to program in FRED.

First, FRED extends the capabilities of Framework and allows you to perform tasks that cannot be performed directly from the menus—printing mailing labels, drawing customized graphs, and processing noncontiguous groups of numbers in a spreadsheet, for example. Second, FRED programming allows you to create automated “menu-driven” systems that a complete novice can use without any knowledge of Framework. Third, FRED programming allows you to build automatic interfaces between Framework and other software systems.

## *Who Is This Book For?*

This book is written for the more advanced Framework user who is ready to move on to developing custom applications within the Framework environment. Although you need not be thoroughly fluent with all of Framework's many capabilities, you should at least be familiar with the basic concepts involved in managing frames. Some general experience with word processing, spreadsheets, database management, or graphics will also be helpful, since the FRED language works within these application areas.

Prior programming experience in a language other than FRED is not necessary. However, if you do have programming experience in any language, whether it is dBASE, BASIC, Pascal, or C, that experience will certainly help. Like all programming languages, FRED uses basic elements such as loops, if/then/else constructs, and functions. More advanced programmers will be happy to discover that FRED also supports recursion, user-defined functions, parameter passing, macros, and unlimited nesting of functions and expressions.

But if you *are* a beginner, don't worry. You'll not only learn what these terms mean, you'll learn how to use the programming features they describe.

## *Structure of the Book*

This book is designed to teach FRED programming from the ground up. The first four chapters provide general discussions of the FRED programming language and Framework macros. Sample programs and exercises are included to give you practice using these new tools.

Chapters 5 through 8 discuss more specific programming techniques in Framework's main applications: word processing, spreadsheets, graphics, and database management. Again, many sample programs are provided that will not only help you to learn, but will probably come in handy in your work with FRED and Framework.

Chapters 9 and 10 discuss two of the more advanced programming techniques, interfacing with external commands and programs and developing menu-driven systems.

Appendices A and B are technical references for the FRED programming language. Every FRED function is discussed in detail, and examples of the functions are provided.



Finally, Appendix C is for the programmer who is already familiar with either of Ashton-Tate's other programming languages, dBASE II or dBASE III. This appendix provides a general discussion of the similarities and differences between dBASE and FRED, as well as a listing of dBASE commands and functions and their counterparts in the FRED language.

The programs in this book were all written using Framework, version 1.1, on an IBM XT with 512K RAM. The programs will run, however, on any computer that supports Framework, such as the IBM PC, AT, and 3270. Some larger programs, such as the animations in Chapter 7, may require more than 256K RAM.

### *Can I Buy the FRED Programs in This Book?*

The answer is a resounding "Yes," with a few not-so-resounding "Maybes." Since I had to write all the FRED programs contained herein prior to writing this book, I certainly have them handy on disk. And, obviously, your own learning is likely to progress more rapidly if you can avoid keying in all the programs yourself. But the "maybes" are these:

First of all, please keep in mind that I am a one-man author, not an international software marketing firm. I cannot possibly provide technical support over the telephone, or provide modifications to match a particular business need. Therefore, if you buy these programs, remember that you do so only to expedite your own learning process; not to get inexpensive custom software to fulfill a particular business need.

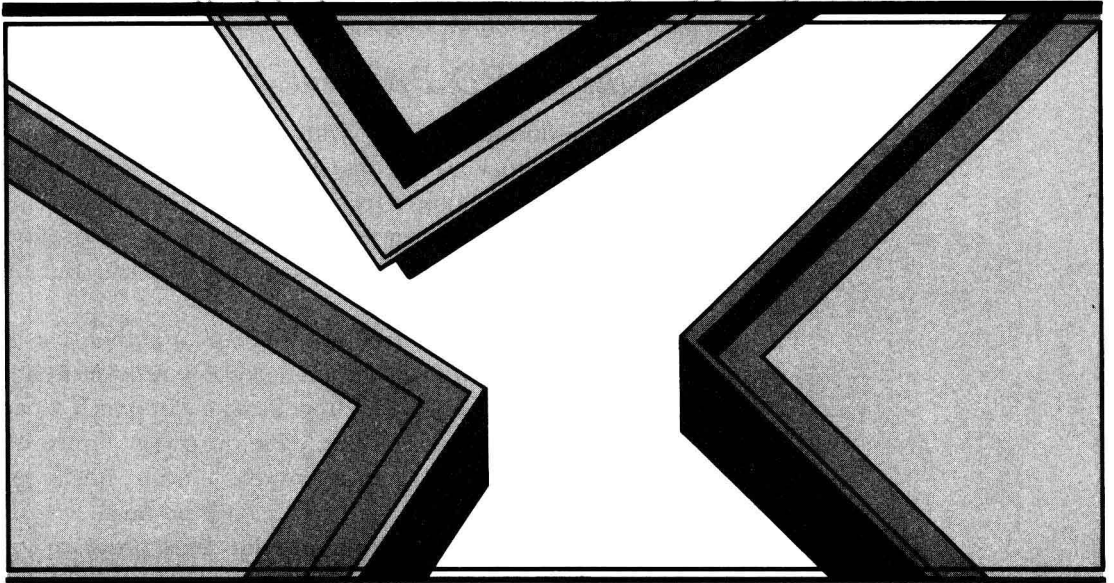
The second "maybe" is that I can only provide these programs on 5<sup>1</sup>/<sub>4</sub>-inch double-sided, double-density diskettes with IBM PC-DOS version 2.1 format. If your computer uses a different disk format, you can probably convert the disk to your format through one of the many disk-conversion services available throughout the country.

If you are still interested in purchasing a copy of the programs, send \$20.00 to:

IBM Data Files  
P.O. Box 2802  
La Jolla, CA 92038-2802

Be sure to specify that you want the programs from Alan Simpson's FRED Book. California residents please add 6% sales tax.

# **1** *The FRED Programming Language*



# Selections from The SYBEX Library

---

## *IBM PC and Compatibles*

### **THE ABC'S OF THE IBM® PC**

**by Joan Lasselle and Carol Ramsay**

143 pp., illustr., Ref. 0-102

This book will take you through the first crucial steps in learning to use the IBM PC.

### **THE MS™-DOS HANDBOOK**

**by Richard Allen King**

320 pp., illustr., Ref. 0-185

The differences between the various versions and manufacturer's implementations of MS-DOS are covered in a clear, straightforward manner. Tables, maps, and numerous examples make this the most complete book on MS-DOS available.

### **ESSENTIAL PC-DOS**

**by Myril and Susan Shaw**

300 pp., illustr., Ref. 0-176

Whether you work with the IBM PC, XT, PCjr, or the portable PC, this book will be invaluable both for learning PC-DOS and for later reference.

## **Integrated Software**

---

### **SYMPHONY™ ENCORE: PROGRAM NOTES**

**by Dick Andersen**

325 pp., illustr., Ref. 0-247

Organized as a reference tool, this book gives shortcuts for using Symphony commands and functions, with troubleshooting advice.

### **MASTERING SYMPHONY™**

**by Douglas Cobb**

763 pp., illustr., Ref. 0-244

This bestselling book provides all the information you will need to put Symphony to work for you right away. Packed

with practical models for the business user.

### **JAZZ ON THE MACINTOSH™**

**by Joseph Caggiano**

400 pp., illustr., Ref. 0-265

The complete tutorial on the ins and outs of the season's hottest software, with tips on integrating its functions into efficient business projects.

### **MASTERING FRAMEWORK™**

**by Doug Hergert**

450 pp., illustr., Ref. 0-248

This tutorial guides the beginning user through all the functions and features of this integrated software package, geared to the business environment.

### **ADVANCED BUSINESS MODELS WITH 1-2-3™**

**by Stanley R. Trost**

250 pp., illustr., Ref. 0-159

If you are a business professional using the 1-2-3 software package, you will find the spreadsheet and graphics models provided in this book easy to use "as is" in everyday business situations.

### **THE ABC'S OF 1-2-3™**

**by Chris Gilbert and Laurie Williams**

225 pp., illustr., Ref. 0-168

For those new to the LOTUS 1-2-3 program, this book offers step-by-step instructions in mastering its spreadsheet, data base, and graphing capabilities.

### **MASTERING APPLEWORKS™**

**by Elna Tymes**

250 pp., illustr., Ref. 0-240

Here is a business-oriented introduction to AppleWorks, the new integrated software package from Apple. No experience with computers is assumed.



***are different.***

---

**Here is why . . .**

At SYBEX, each book is designed with you in mind. Every manuscript is carefully selected and supervised by our editors, who are themselves computer experts. We publish the best authors, whose technical expertise is matched by an ability to write clearly and to communicate effectively. Programs are thoroughly tested for accuracy by our technical staff. Our computerized production department goes to great lengths to make sure that each book is well-designed.

In the pursuit of timeliness, SYBEX has achieved many publishing firsts. SYBEX was among the first to integrate personal computers used by authors and staff into the publishing process. SYBEX was the first to publish books on the CP/M operating system, microprocessor interfacing techniques, word processing, and many more topics.

Expertise in computers and dedication to the highest quality product have made SYBEX a world leader in computer book publishing. Translated into fourteen languages, SYBEX books have helped millions of people around the world to get the most from their computers. We hope we have helped you, too.

***For a complete catalog of our publications please contact:***

---

SYBEX, Inc. 2344 Sixth Street, Berkeley, California 94710  
Tel: (415) 848-8233 Telex: 336311

# ***Table of Contents***

---

Introduction   xi

## ***1 The FRED Programming Language   1***

Features of FRED   1

Data Types   2

Operators   4

Rules of Syntax   5

Programmer Comments   7

So Now What?   7

## ***2 Creating FRED Programs   9***

Entering FRED Programs   9

Running FRED Programs   12

Stopping FRED Programs   12

Editing FRED Programs   12

Saving FRED Programs   14

Retrieving Saved FRED Programs   15

Printing FRED Programs   17

Debugging FRED Programs   18

## ***3 Programming in FRED   21***

Looping with @WHILE   21

Displaying Data with @PROMPT   23

Getting User Input with @INPUTLINE   26

Making Decisions with @IF   31

Cleaning Up   39

## **4 *Macros and the FRED Programming Language* 41**

- Creating a Macro 41
- Putting the New Macro in the MacLib Frame 43
- Framework's Predefined Macros 44
- Moving Macros to FRED Functions 47
- Moving Recorded Macros from MacLib  
to Custom Frames 57
- Macros and FRED Programs 59
- A Handy Macro-Building Macro 61
- Moving Ahead 63

## **5 *Word Processing Programs* 65**

- Getting Started 65
- Setting Up a Containing Frame 65
- Program to Type Company Name 65
- Program to Type Date and Time 69
- A Boilerplate Macro 72
- Automatic Document Formatting 76
- Automatic Printer Formatting 82
- Cut and Paste Programs 88
- Modifying Word Processing Keystrokes 92
- A Help Menu 96
- Using the Word Processing Programs 103
- Printing Several Programs at Once 104

## **6 *Spreadsheet Programming* 107**

- Custom FRED Programs 107
- Creating Your Own Custom Functions 111

Region-Walking Programs 122

Printing Spreadsheets 135

## **7 Graphics Programming 141**

Business Graphs 141

Free-Form Drawing 145

Handy Graphics Utilities 152

Animations 156

## **8 Database Management Programming 169**

Record Numbers 170

Program to Add New Records 171

Sorting Programs 174

Program to Locate Data 177

Database Filtering Programs 178

Printing Mailing Labels 184

A Help Menu 191

Other Capabilities 193

## **9 Interfacing Programs 195**

Accessing External Programs 195

Running System Commands 197

Running DOS Batch Files 199

Copying Spreadsheet Data to a Word Frame 201

Copying Database Data to a Word Frame 205

dBASE II and III Interface 210

Interfacing with WordStar 211

Telecommunications Programming 212

## ***10 Advanced Techniques 215***

- Custom Menu Systems 215
- Selecting Menu Items 219
- Full-Screen and Bottom Menus 220
- Menu Help Messages 221
- Accessing Programs Indirectly 222
- Advanced Programming Techniques 223

## ***Appendices***

### ***A Programming and Macro Functions 231***

### ***B General FRED Functions 269***

- Numeric Functions 269
- Statistical Functions 274
- Financial Functions 276
- Date and Time Functions 278
- Printing Functions 284
- Region-Walking Functions 294
- Tailoring Functions 297

### ***C dBASE to FRED Conversion Aid 301***

- Major Language Differences 301
- Syntax Differences 302
- Skipping Records in a FRED Database 305
- Equivalent Commands and Functions 306



Before we start writing FRED programs, let's take a moment to discuss some of the features of this powerful language, as well as some of its ground rules.

## *Features of FRED*

In a nutshell, here are some of the many features of the powerful FRED programming language:

- 160 Built-in functions (commands)
- Custom programs and spreadsheet formulas up to 32,000 characters long
- Complete graphics capability, including basic business graphics, free-form drawing, and animation
- Complete sound-generating capabilities
- Total integration between word processing, spreadsheet, graphics, and database management capabilities
- Liberal rules of syntax to make programming easier
- User-defined functions, which allow you to add your own commands and functions to the language easily!
- An outstanding text editor for creating programs (Framework's word processor!)
- Variable names of any length; even multiple words
- Full manipulation of character strings up to 32,000 characters long
- The ability to perform any prerecorded series of keystrokes (macros) from within any FRED program
- Complete control of program logic with looping, if/then/else, subroutines, parameter passing, and recursion
- The ability to run external programs via DOS, and capture the results of the programs
- Easy development of user-friendly, menu-driven custom systems
- Powerful debugging aids such as tracing and single-stepping

The list goes on and on. If you are new to programming, some of these features might not mean much to you yet. But don't worry. By