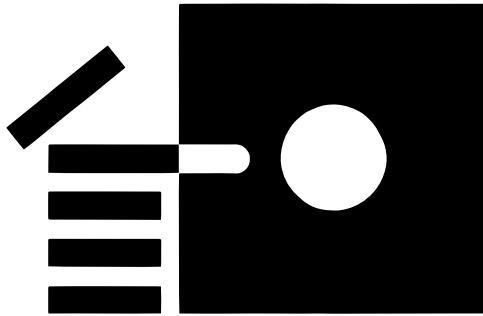


# **ProBASIC**

## **Professional Modular BASIC Programming**



*Alan Bird*

Produced by:

*Brian Wiser & Bill Martens*



Apple PugetSound Program Library Exchange

# **ProBASIC: Professional Modular BASIC Programming**

Copyright © 2018 by Apple Pugetsound Program Library Exchange (A.P.P.L.E.)  
All Rights Reserved.

[www.callapple.org](http://www.callapple.org)

ISBN: 978-0-359-10709-4

## **ACKNOWLEDGEMENTS**

*ProBASIC* was programmed by Alan Bird and published by The Software Touch in 1985. We would like to thank Alan Bird for creating so many tools that were beneficial for Apple II users.

This new manual, produced in coordination with Alan Bird and released with his permission, is copyright by A.P.P.L.E.. No claim to copyright over *ProBASIC* is created outside of those portions created by A.P.P.L.E..

*ProBASIC* disk images are available from the A.P.P.L.E. website: [www.callapple.org](http://www.callapple.org). No warranty of disk images is made or implied and should be used at your own risk.

## **PRODUCTION**

Brian Wisner → Cover, Design, Layout, Editing, Proofreading

Bill Martens → Initial Manual, Software Updates, Production

## **DISCLAIMER**

No part of this book may be reproduced, distributed or transmitted in any form or by any means, including photocopying, recording, or other electronic or mechanical methods, without prior written permission of the publisher, except in the case of brief quotations contained in articles and reviews, and program listings which may be entered, stored and executed in a computer system, but not reproduced for publication.

*ProBASIC* is an independent publication and has not been authorized, sponsored, or otherwise approved by any institution, public or private. All images are under copyright and the property of Apple Pugetsound Program Library Exchange, or as otherwise indicated. Use is prohibited without prior permission.

Apple and all Apple hardware and software brand names are trademarks of Apple Inc., registered in the United States and other countries. All other brand names and trademarks are the property of their respective owners.

While all possible steps have been taken to ensure that the information included within is accurate, the publisher, producers, and author shall have no liability or responsibility for any errors or omissions, or for loss or damages resulting from the use of the information and programs contained herein.

# About the Author

## Alan Bird

Alan Bird has been a software engineer for decades, encompassing the Apple II, Mac, iOS, and other platforms. In 1984 he accepted a position with Beagle Bros in San Diego, which was a very popular Apple II software company known for its signature ads and product documentation that featured old-time artwork. There he developed programming utilities like *Fatcat*, *D Code*, *Extra K*, and *Beagle Compiler*.

In 1986, Alan partnered with fellow Beagle developer Mark Simonsen to form a small software company called The Software Touch. They developed four products for that company along with *TimeOut* add-ons for the popular productivity program *AppleWorks*. Because of their experience with developing *AppleWorks* add-ons, Beagle Bros was hired by Apple to develop version 3 of *AppleWorks*.

In 1990, he joined with another Beagle Bros spinoff called WestCode Software. There he developed a couple of applications for the Apple IIGS and his first Macintosh application, *OneClick*.

In 1996, Alan started work for a rising star in the telecommunications industry, Qualcomm. He spent about five years as part of the Macintosh developer team creating *Eudora* – one of the earliest and most widely-used Internet email applications.

Read more about Alan and his computing history at:  
<https://alanlbird.wordpress.com>

# CONTENTS

## Introduction to ProBASIC

Installing <i>ProBASIC</i> .....	1
Using <i>ProBASIC</i> .....	2

## Programming

Using Modules .....	3
Starting New Programs .....	4
Memory Map .....	5
STOP and CONT .....	5
Handling Errors .....	5

## System Modules

Overview .....	7
HELP .....	8
MAKE .....	9
EDIT .....	10
EXIT .....	11
PARM .....	12
Variable Parameters .....	15
Recursion .....	17
ELSE .....	18
FLAG .....	19

# Changes to ProDOS

New File Types.....	23
Built-in DOS Commands.....	24

# Library Modules

ANYKEY .....	25
BUFFER .....	25
CASE.....	26
CLICK.....	27
DEC.....	27
DISK.....	28
FACTORIAL.....	30
HEX.....	30
HGR .....	31
HGR2 .....	31
INC.....	31
LOCATE .....	31
PI .....	32
REST .....	32
ROUND.....	32
VARs.....	33
VIRTUAL .....	33

# Additional Libraries

LIBRARY 1 – by <i>Rob Renstrom</i> .....	37
--	----

LIBRARY 2 – by <i>Alan Bird</i> .....	39
BUTTON .....	39
CIRCLE.....	39
CONV .....	40
DATE .....	40
DELETE .....	41
ERASE.....	41
ERROR .....	42
FILLCHAR.....	43
FORINT.....	43
INSERT.....	44
LAND.....	44
LINEFIND.....	45
LOR.....	45
MOD.....	46
NEXTINT.....	46
PAGE.....	46
PPEEK.....	47
PPOKE.....	47
READBLOCK.....	47
RESTORE .....	48
RESUME.....	48
STACK.....	49
TIME.....	50
UPPER.....	50
VARPTR.....	51
WINDOW.....	51
WRITEBLOCK .....	52
XOR.....	52

# INTRODUCTION TO ProBASIC

*ProBASIC* enhances the Applesoft BASIC that is built-in to your Apple II series computer. It allows you to add new commands and functions called modules to your programs. This capability is called "extensibility" because the language actually grows.

These modules may be written in either BASIC or machine language. Machine language modules may be written to greatly increase the speed of your programs. Those programmers not familiar with machine language can still easily use machine language modules written by other programmers. BASIC modules add modularity and structure that make it much easier to edit, debug and maintain your programs. Additionally, the modules may be saved on disk and easily integrated into other programs.

Proper use of *ProBASIC* modules will make your programs run faster and will make them much easier to understand for making changes. Variable conflicts are also much easier to avoid since each BASIC module has its own set of variables (local variables). Programs should be broken down into modules of not more than 30 or 40 lines that can be easily understood and quickly edited. Each module can be independently tested and re-used in other programs. Programmers familiar with structured languages such as Pascal and C will quickly see the similarities between a *ProBASIC* module and a procedure or function.

## Installing *ProBASIC*

*ProBASIC* disk images can be downloaded from the publisher's site: [www.callapple.org](http://www.callapple.org). Simply boot the *ProBASIC* disk to install *ProBASIC*. If you have already booted with another ProDOS disk, you may enter "-PRO.SYSTEM" to start *ProBASIC*.