

## Editing and Compiling Wizardry Code with Apple Pascal 1.1

By: Eric Labelle (email: [snafaru@zimlab.com](mailto:snafaru@zimlab.com))

As of 26 Dec 2022

This guide has all you need to know to edit and compile Wizardry's Pascal game code.

The baseline is the re-engineered Pascal source code (and assembler code) from the Wizardry\_I program by Thomas William Ewers. This project would not exist without it.

### Wizardry Proving Grounds 3.0 Web Site and GitHub:

Web Site: <https://www.zimlab.com/wizardry/proving-grounds-v3>

GitHub: <https://github.com/snafaru>

### Final compiled file names:

- WIZARDRY.CODE (Boot and scenario sides)
- SYSTEM.STARTUP (Boot side only)

### Requirements:

- An emulator in Apple IIe mode to get 80-Columns display.
- The emulator must support 4 diskette drives.
- Thomas W. Ewers .dsk images of source code.
- Apple Pascal 1.1 disk images. These are reference disks to make the custom work disks below. Do not overwrite or make these disks writable.
- Apple Pascal custom disk images for editing and compiling.

### Important to always do when you start Pascal:

- **\*\*\*CRITICAL\*\*\*** In the main menu, type "S" to Toggle Swapping to ON.
- In the F)iler, type "D" to set the date.

### Bonus:

- AutoHotkey keystroke player script "AutoHotkey-Pascal-Wizardry.ahk". To facilitate and automate the Pascal commands.

- Companion Excel spreadsheet "Conversion for AutoHotkey.xlsx". To convert a line into AutoHotkey script, which you can then paste into AutoHotkey and run it with CTRL-0 (zero).
- Wizardry Proving Grounds v3.0 Boot and Scenario disks with sample characters provided so you can do your own testing.

NOTICE: Those boot and scenario images are provided for testing purposes only. They are definitely not "Release" versions, nor are they final versions as many more fixes have already been done.

## Recommended Books:

### Essential book:

- Apple Pascal - Operating System Reference Manual and Addendum\*

\*The addendum has important information.

<https://www.zimlab.com/wizardry/proving-grounds-v3/files/Apple-Pascal-1980-Operating-System-Reference-Manual-OCR.pdf>

- Apple Pascal - Apple Pascal Update 1.0 to 1.1

<https://www.zimlab.com/wizardry/proving-grounds-v3/files/Apple-Pascal-Update-1.0-to-1.1-OCR.pdf>

### Other books:

- Apple Pascal - Language Reference Manual

<https://www.zimlab.com/wizardry/proving-grounds-v3/files/Apple-Pascal-1980-Language-Reference-Manual-OCR.pdf>

- Apple Pascal - Language Reference Manual Addendum

<https://www.zimlab.com/wizardry/proving-grounds-v3/files/Apple-Pascal-1980-Language-Reference-Manual-Addendum-OCR.pdf>

- Apple Pascal - A Hands-on Approach

<https://www.zimlab.com/wizardry/proving-grounds-v3/files/Apple-Pascal-A-Hands-on-Approach-OCR.pdf>

### A library of other Pascal books:

- On the Internet Archives

[https://archive.org/details/apple\\_documentation?query=apple+pascal](https://archive.org/details/apple_documentation?query=apple+pascal)

## **Essential Commands:**

All commands are menu driven.

### General:

? To see more commands in any menu.

Wildcards "\$" and "=" and "?".

### Essential E)ditor Commands:

I Enter insert mode.

CTRL-C Exit "insert mode" while keeping changes.

CTRL-O Enter "move cursor mode".

CTRL-O Move up.

CTRL-L Move down.

<- Move left.

-> Move right.

CTRL-C Exit "move cursor mode".

### Essential F)iler Commands:

D Set the date.

V View all devices.

E Extended directory listing.

K Krunch file space on diskette to remove gaps.

G Get a file for then editing in the E)ditor.

#4: Slot 6, Drive 1. This is the boot drive.

#5: Slot 6, Drive 2.

#11: Slot 5, Drive 1.

#12: Slot 5, Drive 2.

Note: If disk space is low, you must erase the file you want to replace first, otherwise it will fail.

## Apple Pascal Operating System Diskettes Contents:

These are your originals.

Do not overwrite or make these diskettes writable.

Use custom diskettes made from these instead to work.

Note: the **SYSTEM.LIBRARY** on the custom diskette **EDIT**: is the one from the Wizardry boot disk, not the one from the Pascal disks.

Disk image file name: UCSD Pascal 1.1\_0.dsk

```
APPLE0:  
SYSTEM.PASCAL      41  22-Sep-80  
SYSTEM.MISCINFO    1   4-May-79  
SYSTEM.COMPIILER   75  19-Sep-80  
SYSTEM.EDITOR      47  24-Sep-80  
SYSTEM.FILER       28  18-Sep-80  
SYSTEM.LIBRARY     34  19-Sep-80  
SYSTEM.CHARSET     2   14-Jun-79  
SYSTEM.SYNTAX      14   1-Aug-80  
8/8 files, 32 unused, 32 in largest
```

Disk image file name: UCSD Pascal 1.1\_1.dsk

```
APPLE1:  
SYSTEM.APPLE      32  9-Nov-80  
SYSTEM.PASCAL     41 22-Sep-80  
SYSTEM.MISCINFO   1  4-May-79  
SYSTEM.EDITOR     47 24-Sep-80  
SYSTEM.FILER      28 18-Sep-80  
SYSTEM.LIBRARY    34 19-Sep-80  
SYSTEM.CHARSET    2 14-Jun-79  
SYSTEM.SYNTAX     14  1-Aug-80  
8/8 files, 75 unused, 75 in largest
```

Disk image file name: UCSD Pascal 1.1\_2.dsk

```
APPLE2:  
SYSTEM.COMPILER   75 19-Sep-80  
SYSTEM.LINKER     24 16-Sep-80  
SYSTEM.ASSMBLER  54 19-Sep-80  
6500.OPCODES      2 20-Dec-78  
6500.ERRORS       7 28-Mar-79  
5/5 files, 112 unused, 112 in largest
```

Disk image file name: UCSD Pascal 1.1\_3.dsk

```
APPLE3:
SYSTEM.APPLE          32   9-Nov-80
FORMATTER.CODE       4   14-Aug-80
FORMATTER.DATA       6   14-Aug-80
LIBRARY.CODE         8   15-Sep-80
LIBMAP.CODE          9   1-Aug-80
SETUP.CODE           33  7-Feb-79
BINDER.CODE          5   1-Aug-80
CALC.CODE            8   28-Dec-78
LINEFEED.TEXT        4   1-Aug-80
LINEFEED.CODE        2   1-Aug-80
SOROCGOTO.TEXT       4   29-Mar-79
SOROCGOTO.CODE       2   14-Aug-80
SOROC.MISCINFO       1   13-Mar-79
HAZELGOTO.TEXT       4   29-Mar-79
HAZELGOTO.CODE       2   14-Aug-80
HAZEL.MISCINFO       1   19-Mar-79
CROSSREF.TEXT        8   4-Mar-80
CROSSREF.CODE        3   1-Aug-80
SPIRODEMO.TEXT       6   4-May-79
SPIRODEMO.CODE       2   14-Aug-80
HILBERT.TEXT         6   4-May-79
HILBERT.CODE         2   14-Aug-80
```

```
GRAFDEMO.TEXT      28  4-May-79
GRAFDEMO.CODE      12 14-Aug-80
GRAFCHARS.CODE     3 14-Aug-80
GRAFCHARS.TEXT     6 22-Jun-79
TREE.TEXT          8 22-Jun-79
TREE.CODE          3 14-Aug-80
BALANCED.TEXT     12 22-Jun-79
BALANCED.CODE     4 14-Aug-80
DISKIO.TEXT       22 14-Aug-80
DISKIO.CODE        7 14-Aug-80
32/32 files, 17 unused, 17 in largest
```



## Wizardry Reversed Engineered Custom Diskettes Contents:

These diskettes have the reverse engineered code by Thomas W. Ewers.

For the scenario side: WIZARDRY.CODE on WIZ1A, all the .CODE on WIZ1D must be erased before recompiling in order to not run out to disk space.

For the boot side, all the .CODE and SYSTEM.STARTUP on WIZ1E must be erased before compiling.

Note: the **SYSTEM.LIBRARY** on the custom diskette **EDIT**: is the one from the Wizardry boot disk, not the one from the Pascal disks.

Wiz1A through Wiz1D contain the code to fixing the WIZARDRY.CODE file.

Disk image file name:      Wiz1A.DSK

```
WIZ1A:
WIZ.TEXT          28 13-Jun-14
WIZ2.TEXT         20 13-Jun-14
SHOPS2.TEXT       18  9-Jun-14
WIZARDRY.CODE     143 15-Jun-14
4/4 files, 65 unused, 65 in largest
```

Disk image file name: Wiz1B.DSK

```
WIZ1B:
UTILITIE2.TEXT      26   9-Jun-14
UTILITIE3.TEXT       4   9-Jun-14
SHOPS.TEXT          28   9-Jun-14
SPECIALS.TEXT       22  10-Jun-14
SPECIALS2.TEXT      28  10-Jun-14
COMBAT.TEXT         18  11-Jun-14
COMBAT2.TEXT        38  10-Jun-14
COMBAT3.TEXT        18  10-Jun-14
COMBAT4.TEXT        36  13-Jun-14
COMBAT5.TEXT        28  10-Jun-14
10/10 files, 28 unused, 28 in largest
```

Disk image file name: Wiz1C.DSK

```
WIZ1C:  
UTILITIE.TEXT      26  9-Jun-14  
CASTLE.TEXT       16 10-Jun-14  
CASTLE2.TEXT      26 10-Jun-14  
ROLLER.TEXT       38 10-Jun-14  
CAMP.TEXT         26 10-Jun-14  
CAMP2.TEXT        22 10-Jun-14  
REWARDS.TEXT      34 11-Jun-14  
REWARDS2.TEXT     12 10-Jun-14  
RUNNER.TEXT       16 12-Jun-14  
RUNNER2.TEXT      28 11-Jun-14  
10/10 files, 30 unused, 30 in largest
```

Disk image file name: Wiz1D.DSK

```
WIZ1D:
CLRRECT.TEXT      6 13-Jun-14
MUCURSOR.TEXT    10 13-Jun-14
PRGRCHR.TEXT      6 13-Jun-14
DRAWLINE.TEXT    12 12-Jun-14
CLRPICT.TEXT     10 12-Jun-14
KEYAVAIL.TEXT     4 12-Jun-14
RANDOM.TEXT        6 13-Jun-14
CLRRECT.CODE      3 15-Jun-14
MUCURSOR.CODE     3 15-Jun-14
PRGRCHR.CODE      3 15-Jun-14
DRAWLINE.CODE     3 15-Jun-14
CLRPICT.CODE      3 15-Jun-14
KEYAVAIL.CODE     3 15-Jun-14
RANDOM.CODE        3 15-Jun-14
14/14 files, 199 unused, 199 in largest
```

Wiz1E contain the code to fixing the SYSTEM.STARTUP file.

Disk image file name: Wiz1E-17-Dec-2022.DSK

```
WIZ1E:
WIZUTILB.TEXT      20 15-Jun-14      6 Text
WIZUTILC.TEXT      30 15-Jun-14     26 Text
TITLELOA.TEXT      10 15-Jun-14     56 Text
LZDECOMP.TEXT      10 14-Jun-14     66 Text
CHKSYNCH.TEXT      16 14-Jun-14     76 Text
SCREENPT.TEXT       8 14-Jun-14     92 Text
STARTUP.TEXT       10 17-Dec-22    100 Text
OPTIONS.TEXT        4 17-Dec-22    110 Text
LZDECOMP.CODE      3 17-Dec-22    114 Code
CHKSYNCH.CODE      3 17-Dec-22    117 Code
SCREENPT.CODE       3 17-Dec-22    120 Code
SYSTEM.STARTUP     23 17-Dec-22    123 Code
< UNUSED >        134
12/12 files, 134 unused, 134 in largest
```

## Pascal Custom Work Diskettes Contents:

Note: the **SYSTEM.LIBRARY** on the custom diskette **EDIT:** is the one from the Wizardry boot disk, not the one from the Pascal disks.

The **EDIT:** disk is the one you will boot with almost always.

Disk image file name:      **EDIT.DSK**

```
EDIT:
SYSTEM.APPLE      32  9-Nov-80      6  Data
SYSTEM.PASCAL    41 22-Sep-80     38 Code
SYSTEM.MISCINFO   1  4-May-79     79 Data
SYSTEM.FILER     28 18-Sep-80     80 Code
SYSTEM.EDITOR    47 24-Sep-80    108 Code
SYSTEM.COMPILER  75 19-Sep-80    155 Code
SYSTEM.LIBRARY   36  3-Jan-81    230 Data
< UNUSED >      14                               266
7/7 files, 14 unused, 14 in largest
```

The COMPILE: disk is mainly used to compile the SYSTEM.STARTUP

Disk image file name: COMPILE.DSK

```
COMPILE:
SYSTEM.APPLE      32  9-Nov-80      6  Data
SYSTEM.PASCAL    41 22-Sep-80     38  Code
SYSTEM.MISCINFO   1  4-May-79     79  Data
SYSTEM.COMPILES  75 19-Sep-80    80  Code
SYSTEM.LINKER    24 16-Sep-80   155  Code
SYSTEM.ASSMBLER  54 19-Sep-80   179  Code
6500.OPCODES     2 20-Dec-78   233  Data
6500.ERRORS      7 28-Mar-79   235  Data
< UNUSED >      38                               242
8/8 files, 38 unused, 38 in largest
```

The FILER: disk is mainly used to Format disks. It also has the Editor and base Compiler. Not really used for this project, but it is nice to have the Formatter at hand when you need it.

From the main menu: X)ecute, then type: FILER:FORMATTER

Disk image file name: FILER.DSK

```
FILER:
SYSTEM.APPLE      32  9-Nov-80    6  Data
SYSTEM.PASCAL    41 22-Sep-80   38 Code
SYSTEM.MISCINFO   1  4-May-79   79 Data
SYSTEM.FILER     28 18-Sep-80   80 Code
SYSTEM.EDITOR    47 24-Sep-80  108 Code
SYSTEM.COMPILER  75 19-Sep-80  155 Code
FORMATTER.CODE   4 14-Aug-80  230 Code
FORMATTER.DATA   6 14-Aug-80  234 Data
< UNUSED >      40                               240
8/8 files, 40 unused, 40 in largest
```



## Wizardry Compiling Instructions

First, a bit of a description of what is going on from Thomas W. Ewers on Google Groups:

Ref: <https://groups.google.com/g/comp.sys.apple2/c/a15ob1mLUwY/m/CIqEeVUvDAAJ>

TommyGoog

Jun 16, 2014, 12:15:38 PM

to

Today I uploaded the source code and listings to ASIMOV.

/pub/apple\_II/images/games/rpg/wizardry/wizardry\_I/

In this message I will duplicate the content of WiziSourceCode.ASM.txt, and following that I have more detailed instructions about assembling and compiling the code.

=====

Begin

Title: Wizardry\_i\_SourceCode

Author: Thomas William Ewers

Version:

Entered-date: June 15, 2014

Description:

Pascal (and assembler) source code files and generated listings for Wizardry\_I, Proving Grounds of the Mad Overlord.

I have re-engineered Pascal source code (and assembler code) from the Wizardry\_I program. The original executable files can be found on:

Wizardry\_i\_boot.dsk

In this package:

/WizardryCode

Wiz1A.DSK

Wiz1B.DSK

Wiz1C.DSK

Wiz1D.DSK

Wiz1WizardryAssembler.txt

Wiz1WizardryPascal.txt

/SystemStartup

Wiz1E.DSK

Wiz1StartupAssembler.txt

Wiz1StartupPascal.txt

The Pascal code found on the enclosed disks is NOT the code that the original authors wrote. However, when you compile (and assemble and link) these files, the EXACT executable code in Wizardry.code and System.Startup are created.

The source code can be compiled using Apple Pascal 1.1 on an Apple[[ computer (or simulator) that has 4 diskette drives.

See the discussion on Usenet at:

comp.sys.apple2

"Wizardry re-engineering"

My Google Groups name is: TommyGoog

Keywords: Wizardry, Proving Grounds of the Mad Overlord, SIR-TECH

Uploader: ewe...@msn.com

Primary-site: ftp://ftp.apple.asimov.net

/pub/apple\_II/images/games/rpg/wizardry/wizardry\_1/

Wizardry\_1\_SourceCode.zip

WiziSourceCode.ASM.txt

/WizardryCode

Wiz1A.DSK

Wiz1B.DSK

Wiz1C.DSK

Wiz1D.DSK

Wiz1WizardryAssembler.txt

Wiz1WizardryPascal.txt

/SystemStartup

Wiz1E.DSK

Wiz1StartupAssembler.txt

Wiz1StartupPascal.txt

Platform: Apple][

End

=====

**Here are more detailed instructions on building and verifying the code.**

Note: If using AutoHotkey, then all AutoHotkey commands start from the main menu.

## **Begin - Build System.Startup**

Build Instructions:

Insert disks into drives:

4: EDIT.DSK

5: Wiz1E-17-Dec-2022.DSK or newer (To have the copy protection check bypass fix included).

Note: If you work with the WIZ1E.DSK, Wizardry will hang on boot.

11: COMPILE.DSK

12: WizardryDotCode.dsk

Restart AppleWin (to make 100% sure the Editor is empty and that nothing else stupid happens, clicking the "Apple" logo to reset is not enough under some circumstances).

If you want to have a printout:

- Delete PRINTER.TXT on your computer (not the virtual computer).
- Set AppleWin printer to APPEND.

Boot with EDIT.DSK, if you did not so yet.

**\*\*\*CRITICAL\*\*\* S)wap ON ("Y")**

F)iler

**D)ate (Set the date)**

Q)uit to main menu

If using AutoHotkey -begin- press ctrl-6

F)iler

R)emove

#5:SYSTEM.STARTUP if present, if that throws an error it probably means your disk is write-protected.

"Y"

R)emove

#5:=.CODE to remove all .CODE files.

"Y"

K)runch #5: to end of disk.

"Y"

E)xtended #5 (to list the contents of Slot 6, Drive 2)

STARTUP.TEXT

WIZUTILB.TEXT

WIZUTILC.TEXT

TITLELOA.TEXT

OPTIONS.TEXT

LZDECOMP.TEXT

CHKSYNCH.TEXT

SCREENPT.TEXT

< UNUSED > 166 114

Q)uit to main menu

A)ssemb

Assemble what text?

#5:LZDECOMP

To what codefile?

\$

Output file

<cr> or PRINTER: (Note: for the rest of this document choose PRINTER: to have a printout, other just keypress <ENTER> to continue)

A)ssem

#5:CHKSYNCH

\$

<cr> or PRINTER:

A)ssem

#5:SCREENPT

\$

<cr> or PRINTER:

C)ompile

Compile what text?

#5:STARTUP

To what codefile?

\$

E)xtended #5: (to list the contents of Slot 6, Drive 2)

Q)uit to main menu

[If using AutoHotkey -end- press ctrl-6](#)

[If using AutoHotkey -begin- press ctrl-7](#)

F)iler

R)emore

#12:=STARTUP.CODE

"Y"

K)runch #12: to end of disk.

"Y"

E)xtended #12: (to list the contents of Slot 5, Drive 2)

Should show < UNUSED > 274 6

Q)uit to main menu

L)ink

Host file?

#5:STARTUP

Lib file?

#5:LZDECOMP

Lib file?

#5:CHKSYNCH

Lib file?

#5:SCREENPT

Lib file?

<cr>

Map file?

<cr>

see:

Reading WIZBOOT

Reading LZDECOMP

Reading CHKSYNCH

Reading SCREENPT

Output file?

#12:STARTUP

see:

Linking UTILS #7

Linking TITLELOA #8

Linking OPTIONS #9

Linking WIZBOOT #1

Copying proc LZDECOMP

Copying proc CHKSYNCH

Copying proc SCREENPT

F)iler

E)xtended #12: (to list the contents of Slot 5, Drive 2)

Should see STARTUP.CODE

Q)uit to main menu

If using AutoHotkey -end- press ctrl-7

If using AutoHotkey -begin- press ctrl-8

F)iler

T)ransfer

#12:STARTUP.CODE

to:

#5:STARTUP.CODE

Remove Old #5:STARTUP.CODE ?

“Y”

C)hange

#5:STARTUP.CODE

to:

#5:SYSTEM.STARTUP

E)xtended #5: (to list the contents of Slot 6, Drive 2)

If using AutoHotkey -end- press ctrl-8

You can now copy that new SYSTEM.STARTUP to any Wizardry Proving Grounds boot disk. If it does not boot, you might have to make the .dsk file read-only (if the ST003 Write Protect Check fix has not been implemented yet in your build, after making the file read-only, completely close the AppleWin program and restart it).

Here is an extra AutoHotkey to do that:



If using AutoHotkey -begin- press ctrl-9

Following up from ctrl-8, put your Wizardry boot disk in drive #5:

F)iler

R)emove

#5:SYSTEM.STARTUP

“Y”

T)ransfer

#12:STARTUP.CODE

to:

#5:STARTUP.CODE

C)hange

#5:STARTUP.CODE

to:

#5:SYSTEM.STARTUP

E)xtended #5: (to list the contents of Slot 6, Drive 2)

If using AutoHotkey -end- press ctrl-9

**End - Build System.Startup**

## Begin - Verify System.Startup

Note 1 from Eric: I have not verified this process as of writing this document. It requires the custom DeCompile.DSK from Thomas.

Note 2 from Eric: This section is only if you want to prove that the code generated by Thomas' original disks matches exactly the Wizardry code.

Note 3 from Eric: I am leaving this here for historical purposes. Since this project modifies the code to fix bugs and issues, the new generated code will not match.

Eject disk "WizardryDotCode" from #12:

Insert "WizardryDotCode.DSK" into #5:

Insert "DeCompile.DSK" into #4:

Reboot

"Y" for all

Output from all the steps above should be in PRINTER.TXT on my computer.

To find start of PASCAL code, search for "PRINTER:".

To find start of pCode, search for "jtab".

Copy pCode to Wizardry1StartupPcode.txt

Use WINDIFF to compare pCodeProvGrnd STARTUP.txt

to Wizardry1StartupPcode.txt

Compare 6502 code:

RUN DeCompiler

PROCESS ALL?

N

1 STARTUP.CODE

TYPE A FILE NUMBER (or -1)

1

0 WIZBOOT

1 UTILS

2 TITLELOA

3 OPTIONS

TYPE A SEGMENT NUMBER (or -1)

0

SEGMENT'S START ADDR(\$5000):

<cr>

ENTER PROCEDURE # (OR -1)

-1

TYPE A SEGMENT NUMBER (or -1)

-1

TYPE A FILE NUMBER (or -1)

-1

] (BASIC prompt)

]CALL-151

\* (monitor prompt)

\*511ALL

40 "L"s and should display to \$5643

Output is in PRINTER.TXT

Save it to TEMP2.

Repeat this displaying of 6502 code using Proving

Grounds Boot Disk.

Save it to TEMP1.

Compare Temp1 to Temp2. They should be the same.

**End - Verify System.Startup**

## Begin - Build Wizardry.Code

If you want to have a printout:

- Delete PRINTER.TXT on your computer (not the virtual computer).
- Set AppleWin printer to APPEND.

Insert disks:

4: EDIT.DSK

5: Wiz1D.DSK

11: COMPILE.DSK

12: WizardryDotCode.dsk

Restart AppleWin (to make 100% sure the Editor is empty and that nothing else stupid happens, clicking the "Apple" logo to reset is not enough under some circumstances).

Boot with EDIT.DSK, if you did not so yet.

**\*\*\*CRITICAL\*\*\* S)wap ON ("Y")**

F)iler

**D)ate (Set the date)**

Q)uit to main menu

**If using AutoHotkey -begin- press ctrl-2**

Remove ".CODE" files from #5:

F)iler

R)emove

#5:=.CODE to remove all .CODE files.

"Y"

K)runch #5: to end of disk.

"Y"

E)xtended #5:

CLRRECT.TEXT

MVCURSOR.TEXT

PRGRCHR.TEXT

DRAWLINE.TEXT

CLRPICT.TEXT

KEYAVAIL.TEXT

RANDOM.TEXT

<UNUSED> 220 60

Remove files from #12:

R)emove

#12:=.CODE

“Y”

<UNUSED> 274 6

Q)uit to main menu

A)ssem

Assemble what text?

#5:CLRRECT

To what codefile?

\$

Output file

<cr> or PRINTER: (Note: for the rest of this document choose PRINTER: to have a printout, other just keypress <ENTER> to continue)

Assemble the others:

Assemble what text?

#5:MVCURSOR

\$

<cr> or PRINTER:

#5:PRGRCHR

\$

<cr> or PRINTER:

#5:DRAWLINE

\$

<cr> or PRINTER:

#5:CLRPICT

\$

<cr> or PRINTER:

#5:KEYAVAIL

\$

<cr> or PRINTER:

#5:RANDOM

\$

<cr> or PRINTER:

F)iler

E)xtended #5:

CLRRECT.TEXT  
MVCURSOR.TEXT  
PRGRCHR.TEXT  
DRAWLINE.TEXT  
CLRPICT.TEXT  
KEYAVAIL.TEXT  
RANDOM.TEXT  
CLRRECT.CODE  
MVCURSOR.CODE  
PRGRCHR.CODE  
DRAWLINE.CODE  
CLRPICT.CODE  
KEYAVAIL.CODE  
RANDOM.CODE  
<UNUSED> 199 81

If using AutoHotkey -end- press ctrl-2

Insert Disks:

04: Edit

05: Wiz1A.DSK

11: Wiz1B.DSK

12: Wiz1C.DSK

If using AutoHotkey -begin- press ctrl-3

F)iler

R)emove

#5:=.CODE to remove all .CODE files.

"Y"

K)runch #5: to end of disk.

"Y"

E)xtended #5:

WIZ.TEXT

WIZ2.TEXT

SHOPS2.TEXT

<UNUSED> 208 72

Q)uit to main menu

C)ompile

Compile what text?

#5:WIZ

To what codefile?

\$

If using AutoHotkey -end- press ctrl-3

Insert Disks:

04: COMPILE.DSK

05: Wiz1A.DSK

11: Wiz1D.DSK

12: WizardryDotCode.DSK (this disk WIZCODE: should be empty).

Note: If you get this error:

L)ink

No file :SYSTEM.LINKER

Insert COMPILE.DSK into #4:

If using AutoHotkey -begin- press ctrl-4

L)ink

Host file?

#5:WIZ.CODE

Lib file?



#11:CLRRECT

Lib file?

#11:MVCURSOR

Lib file?

#11:PRGRCHR

Lib file?

#11:DRAWLINE

Lib file?

#11:CLRPICT

Lib file?

#11:KEYAVAIL

Lib file?

#11:RANDOM

Lib file?

<cr>

Map file?

<cr>

see:

Reading WIZARDRY

Reading CLRRECT

Reading MVCURSOR

Reading PRGRCHR

Reading DRAWLINE

Reading CLRPICT

Reading KEYAVAIL

Reading RANDOM

Output file?

#12:WIZARDRY

see:

Linking UTILITIE #7

Linking SHOPS #8

Linking SPECIALS #9

Linking COMBAT #10

Linking CINIT #11

Linking CUTIL #12

Linking MELEE #13

Linking CASTASPE #14

Linking SWINGASW #15

Linking CASTLE #16

Linking ROLLER #17

Linking CAMP #18

Linking REWARDS #19

Linking RUNNER #20

Linking WIZARDRY #1

Copying proc CLRRECT

Copying proc MVCURSOR

Copying proc PRGRCHR

Copying proc DRAWLINE

Copying proc CLRPICIT

Copying proc KEYAVAIL

Copying proc RANDOM

[If using AutoHotkey -end- press ctrl-4](#)

The system will ask you to put EDIT.DSK back into drive 04:

Last step: copy the new mint WIZARDRY.CODE to the necessary disks.

Insert EDIT.DSK into 04: (if not already there).

Disk #5:Wiz1A.DSK should already be there.

If using AutoHotkey -begin- press ctrl-5

F)iler

R)emove

#5:=.CODE

T)ransfer

#12:WIZARDRY.CODE

to:

#5:WIZARDRY.CODE

If using AutoHotkey -end- press ctrl-5

Repeat the steps above to update the boot and scenario sides by putting them in drive 05: in succession.

**End - Build Wizardry.Code**

## **Begin - Verify Wizardry.Code**

Note 1 from Eric: I have not verified this process as of writing this document. It requires the custom DeCompile.DSK from Thomas.

Note 2 from Eric: This section is only if you want to prove that the code generated by Thomas' original disks matches exactly the Wizardry code.

Note 3 from Eric: I am leaving this here for historical purposes. Since this project modifies the code to fix bugs and issues, the new generated code will not match.

Insert 01 EDIT.DSK into #4

Eject #12: 16 WizardryDotCode.DSK

Insert #5: 16 WizardryDotCode.DSK

Insert #4: 05 DeCompile.DSK

ReBoot

PROCESS ALL?

Y

Printer.TXT has all the previous output.

Create the following:

Wizardry1WizAssem So Far.txt

Wizardry1WizPascal So Far.txt

Wizardry1WizPcode.txt

Compare Wizardry1WizPcode.txt with pCodeProvGrndBoot.txt

pCodeProvGrndBoot has ALL files on boot diskette pcode,  
including SYSTEM.PASCAL.

Compare 6502 code.

RUN DeCompiler

Do same trick as for STARTUP.

Only WIZARDRY.CODE, then SEGMENT 0, then \$5000, then -1 -1 -1.

\*CALL -151

\*5836||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||

about 65 "L"s should do it.

Compare from \$5836 to \$5D00 as shown in following table.

Link

Order Enter Exit PROC/FUNC

-----

5 26 5B96 5C80 CLRPICT

4 27 59D8 5B90 DRAWLINE

7 28 5CC0 5D00 RANDOM

6 29 5C86 5CBA KEYAVAIL

1 30 5836 58B6 CLRRECT

2 31 58BC 594E MVCURSOR

3 32 5954 59D2 PRGRCHR

Tommy (Thomas W. Ewers)

**End - Verify Wizardry.Code**