Windows

An Apple II computer is not a Macintosh, yet we're seeing more and more Mac-like software for the Apple II, II+, IIe, and IIc. Duplicating Macintosh-style windows—just one of the useful features of that machine's operating system—is simple with this program. For all Apple II-series computers, using either DOS 3.3 or ProDOS.

One of the features that makes the Macintosh so easy to use is its ability to open and close multiple *windows* on the screen. These windows—basically smaller text screens superimposed on the main screen—can provide additional information, offer menu selections, or provide a notepad-style environment where you can enter and save text. Once the information has appeared or the menu item has been chosen, the window can be erased, letting you get on with the task at hand.

The Apple II-series computers can create windows, too, even automatically save and restore text screens. With "Windows" at your disposal, you can open a window in an existing text screen and make it disappear, all without having to reprint the underlying screen. Windows easily simulates a Macintosh appearance in your own BASIC programs, letting you operate with as many as nine windows (ten if you count the main screen).

Machine Language The Easy Way

Though Windows is a machine language program, you don't need to know anything about machine language programming to enter or use it. Program 1, "Windows Creator" is a BASIC program that you can type in, save, and run. Once it's run, it creates a machine language file on the disk. (Because Program 1 uses the name WINDOWS for the machine language file it writes to disk, you cannot use that name for Program 1 itself. If you save Program 1 with the name WINDOWS, you'll get a FILE TYPE MISMATCH error when you run Program 1.) To load Windows (the machine language program Program 1 created), enter:

```
BLOAD WINDOWS
```

Windows is now in memory, waiting. Simple.

But Windows does nothing all by itself. It must be used in conjunction with a BASIC program. Let's take a look at a demonstration of what Windows can do.

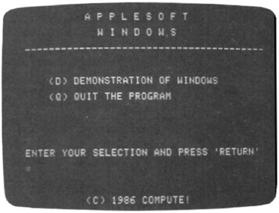
Showing Off

Type in and save Program 2, "Windows In BASIC." (Remember, you must use some name other than WINDOWS for this program.) This is a complete illustration of Windows' power, and works in either DOS 3.3 or ProDOS. If you're using the latter, however, you must make one change. Modify line 110 so that it reads:

```
110 HIMEM: 33792
```

Make sure a copy of the WINDOWS file created by Program 1 is on the same disk as Program 2, then type RUN. You'll see this:

Figure 1: The Main Screen

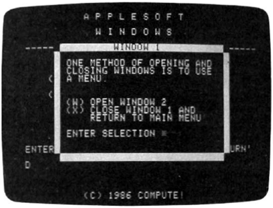


```
            A P P L E S O F T
              W I N D O W S
 ----------------------------------------

     (D) DEMONSTRATION OF WINDOWS
     (Q) QUIT THE PROGRAM


 ENTER YOUR SELECTION AND PRESS 'RETURN'


           (C) 1986 COMPUTE!
```

Press the *D* key, then hit Return to run the demonstration. The computer will display window 1, as you can see in Figure 2. Window 1 is superimposed over the main screen, so parts of the latter still show around the solid white border of the window. The computer has saved the main screen to be restored later.

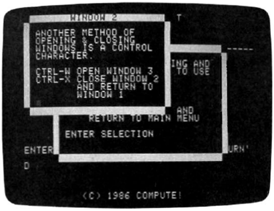
Press *W* to open window 2. This second window is also superimposed on the previous screen, so parts of both the main screen and window 1 show around its edges.

Figure 2: First Window Added



```
A P P L E S O F T
  W I N D O W S

WINDOW 1
ONE METHOD OF OPENING AND
CLOSING WINDOWS IS TO USE
A MENU.

(W) OPEN WINDOW 2
(X) CLOSE WINDOW 1 AND
    RETURN TO MAIN MENU

ENTER SELECTION

(C) 1986 COMPUTE!
```

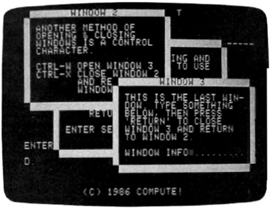
Figure 3: Another Window



```
WINDOW 2
ANOTHER METHOD OF
OPENING & CLOSING
WINDOWS IS A CONTROL
CHARACTER.

CTRL-W OPEN WINDOW 3
CTRL-X CLOSE WINDOW 2
       AND RETURN TO
       WINDOW 1

RETURN TO MAIN MENU
ENTER SELECTION

(C) 1986 COMPUTE!
```

Type *Ctrl-W* (Hold the Ctrl key and press *W*). The computer places the third window over the ones already on the screen.

Figure 4: Window Three



```
WINDOW 3
THIS IS THE LAST WIN-
DOW. TYPE SOMETHING
BELOW, THEN PRESS
'RETURN' TO CLOSE
WINDOW 3 AND RETURN
TO WINDOW 2.

WINDOW INFO.........

(C) 1986 COMPUTE!
```

Type your name (or anything else) in the blank on window 3 and press Return. The computer remembers what you typed, but closes window 3 and returns to window 2. Press Ctrl-X to close window 2, then hit the X key to close window 1. You're back where you started, with the main screen displayed.

In this demonstration, windows 1 and 2 were menus, but if a program allows you to type something on a window, it will be restored when you close the window.

Each time you opened a window, the computer saved the current screen to memory. Each time you closed a window, the computer restored the screen it had saved.

Now press Q and hit Return. The computer exits the program, printing the text you typed in window 3 on the screen as it says goodbye.

Inside Windows

The Apple's 40-column text screen is located at memory addresses 1024–2047. When you open a window, the machine language program Windows copies the data on the current text screen to a safe place above HIMEM, and transfers it back when you close a window. Windows also stores information about the screen size and cursor location so that the computer remembers the exact screen arrangement when you close the window.

In Apple II-series computers, memory addresses 32–37 maintain information about the text screen:

Address	Contents
32	Left margin (default = 0)
33	Width (default = 40)
34	Top margin (default = 0)
35	Bottom margin (default = 24)
36	Horizontal cursor location
37	Vertical cursor location

Program 2 POKEs values to these addresses to change the text screen characteristics. Take a close look at the listing. Though it's only a simple example, it shows how you can use Windows in your own programs.

Windows Of Your Own

Lines 100–130 in Program 2 are mandatory to initialize the program parameters. *You must include these same lines (slightly modified) in your own program in order to use Windows.*

Line 110. The value of HIMEM in line 110 depends on the maximum number of windows you intend to use, and whether you're using DOS 3.3 or ProDOS. See Table 1 for the appropriate values.

Table 1: HIMEM Values

Maximum # of Windows	DOS 3.3	ProDOS
1	36352	35840
2	35328	34816
3	34304	33792
4	33280	32768
5	32256	31744
6	31232	30720
7	30208	29696
8	29184	28672
9	28160	27648
10	27136	26624

Line 120. These POKEs should be specified early in the program. (Table 2 shows the values which must be POKEd into memory to open and close windows—you'll find the locations in line 120 listed in this table.) Of these three POKEs, the only one which you'll need to change in your own program is POKE 769,WMAX. Simply set WMAX to the maximum number of windows your program will allow.

Line 130. These POKEs establish the default characteristics of the Apple II text screen. Take a look at the listing above (locations 32–37), and you'll see that the four POKEs in this line set up the default values of:

Left margin	0
Width	40
Top margin	0
Bottom margin	24

Enter these POKEs in your own program just as you see them in line 130 of Program 2.

Opening Windows

Lines 300 and 310 in Program 2 are an example of the information you *must* provide to open a window. The POKEs in line 300 define the size and location of the window, while the POKE and CALL in line 310 activates Windows. Each window is defined by POKEing the window characteristics before CALLing Windows with CALL 37376. For example, line 300 defines window 1 as having a left margin in column 5 (POKE 32,5), a width of 30 characters (POKE 33,30), a top margin at text line 4 (POKE 34,4) and a bottom margin at text line 19 (POKE 35,19).

Closing Windows

Line 430 is an example of closing a window. You need only to POKE 768,0 and CALL 37376—you don't need to redefine the window parameters. When Windows opens a window, it stores the window parameters, then automatically restores them when it closes the window.

Windows stores the parameters for each window in the normally unused space beginning at memory location 768 ($0300 in hexadecimal). Table 2 lists the values stored at each address.

Each text screen is saved in a separate area above HIMEM, beginning with Window 0 (the main screen), stored from memory addresses 36352 to 37376, and working downward.

When you close a window, the computer

Table 2: Windows Variable Storage

Memory Address	Description	Monitor Address	Range
768	Direction of window movement	n/a	0=Open, 1=Close
769	Maximum number of windows	n/a	1–*n*
770	Current window number	n/a	0–10
771	Window 1, left margin	32	0–39
772	Window 1, width	33	1–40
773	Window 1, top margin	34	0–22
774	Window 1, bottom margin	35	1–24
775	Window 0, horizontal cursor position	36	0–39
776	Window 0, vertical cursor position	37	0–23
777	Window 2, left margin	32	0–39
778	Window 2, width	33	1–40
779	Window 2, top margin	34	0–22
780	Window 2, bottom margin	35	1–24
781	Window 1, horizontal cursor position	36	0–39
782	Window 1, vertical cursor position	37	0–23
783	Window 3, left margin	32	0–39
784	*And so on*		

restores the original screen by POKEing the screen characteristics in locations 32–37 and moving the text screen from storage back to the text screen buffer at memory addresses 1024–2047. Note, too, that with each window's margin and width values are stored the *previous* window's cursor positions. Thus, when you close a window, the cursor appears at the position it occupied *before* that window was opened.

Using Windows on your Apple II won't turn it into a Macintosh, but it can add some of the sophistication of the Macintosh to your BASIC programs. Open a window and see for yourself.

Program 1: Windows Creator

Be sure to use "Apple Automatic Proofreader," found elsewhere in this issue, to enter the following programs.

```
Ø4 1Ø REM BASIC PROGRAM FOR
7C 2Ø REM GENERATING THE
4A 3Ø REM BINARY FILE
C4 4Ø REM  'WINDOWS'
4E 5Ø HOME
36 6Ø VTAB 12: PRINT "WORKING ..."
F8 7Ø FOR I = Ø TO 841
B8 8Ø READ A
19 9Ø POKE 37376 + I,A
BF 1ØØ VTAB 12: HTAB 13: PRINT I + 1
DF 11Ø NEXT I
72 12Ø PRINT CHR$ (4)"BSAVE WINDOWS, A373
       76, L1Ø12"
2B 13Ø PRINT : PRINT "DONE!"
22 14Ø DATA 173,89,17Ø,72,165,217,72
2B 15Ø DATA 165,118,72,169,2,133,118
65 16Ø DATA 169,255,133,217,169,191,133
6A 17Ø DATA 51,169,Ø,133,243,76,35
CB 18Ø DATA 146,Ø,Ø,Ø,146,Ø,8
FA 19Ø DATA 169,29,133,133,169,146,16Ø
B7 2ØØ DATA Ø,162,5,32,47,149,173
F6 21Ø DATA Ø,3,141,29,146,169,Ø
7E 22Ø DATA 141,3Ø,146,173,29,146,2Ø1
77 23Ø DATA 1,2Ø8,1Ø,173,3Ø,146,2Ø1
A6 24Ø DATA Ø,2Ø8,3,76,67,147,173
FF 25Ø DATA 2,3,141,29,146,169,Ø
95 26Ø DATA 141,3Ø,146,173,1,3,141
77 27Ø DATA 31,146,169,Ø,141,32,146
6B 28Ø DATA 238,29,146,2Ø8,3,238,3Ø
E1 29Ø DATA 146,173,3Ø,146,2Ø5,32,146
C5 3ØØ DATA 48,15,2Ø8,1Ø,173,29,146
Ø5 31Ø DATA 2Ø5,31,146,144,5,24Ø,3
B8 32Ø DATA 76,237,148,32,125,148,169
EB 33Ø DATA 32,141,33,146,169,Ø,141
19 34Ø DATA 34,146,173,34,146,2Ø1,Ø
ED 35Ø DATA 48,14,2Ø8,9,173,33,146
F6 36Ø DATA 2Ø1,37,144,5,24Ø,3,76
4Ø 37Ø DATA 223,146,173,33,146,141,176
C7 38Ø DATA 146,173,34,146,141,177,146
C5 39Ø DATA 173,37,Ø,141,29,146,169
DC 4ØØ DATA Ø,141,3Ø,146,173,31,146
12 41Ø DATA 141,2Ø2,146,173,32,146,141
55 42Ø DATA 2Ø3,146,173,29,146,141,14
D2 43Ø DATA 3,238,31,146,2Ø8,3,238
EB 44Ø DATA 32,146,238,33,146,2Ø8,3
59 45Ø DATA 238,34,146,76,142,146,32
EA 46Ø DATA 187,148,169,Ø,141,33,146
BF 47Ø DATA 169,4,141,34,146,173,34
F8 48Ø DATA 146,2Ø1,7,48,14,2Ø8,9
7A 49Ø DATA 173,33,146,2Ø1,255,144,5
F7 5ØØ DATA 24Ø,3,76,61,147,173,33
C8 51Ø DATA 146,141,14,147,173,34,146
73 52Ø DATA 141,15,147,173,255,7,141
F5 53Ø DATA 29,146,169,Ø,141,3Ø,146
ØB 54Ø DATA 173,31,146,141,4Ø,147,173
DC 55Ø DATA 32,146,141,41,147,173,29
7F 56Ø DATA 146,141,255,141,238,31,146
79 57Ø DATA 2Ø8,3,238,32,146,238,33
BA 58Ø DATA 146,2Ø8,3,238,34,146,76
A3 59Ø DATA 236,146,32,88,252,76,237
23 6ØØ DATA 148,173,2,3,141,29,146
16 61Ø DATA 169,Ø,141,3Ø,146,173,29
2F 62Ø DATA 146,2Ø8,3,2Ø6,3Ø,146,2Ø6
7E 63Ø DATA 29,146,173,3Ø,146,2Ø1,Ø
63 64Ø DATA 48,9,2Ø8,1Ø,173,29,146
37 65Ø DATA 2Ø1,Ø,176,3,76,237,148
D2 66Ø DATA 32,125,148,169,32,141,33
27 67Ø DATA 146,169,Ø,141,34,146,173
16 68Ø DATA 34,146,2Ø1,Ø,48,14,2Ø8
CC 69Ø DATA 9,173,33,146,2Ø1,37,144
6F 7ØØ DATA 5,24Ø,3,76,2Ø2,147,173
26 71Ø DATA 31,146,141,155,147,173,32
4Ø 72Ø DATA 146,141,156,147,173,2,3
39 73Ø DATA 141,29,146,169,Ø,141,3Ø
6B 74Ø DATA 146,173,33,146,141,181,147
9E 75Ø DATA 173,34,146,141,182,147,173
63 76Ø DATA 29,146,141,37,Ø,238,31
B7 77Ø DATA 146,2Ø8,3,238,32,146,238
7E 78Ø DATA 33,146,2Ø8,3,238,34,146
53 79Ø DATA 76,121,147,32,187,148,56
6E 8ØØ DATA 173,31,146,233,Ø,141,31
C1 81Ø DATA 146,173,32,146,233,4,141
76 82Ø DATA 32,146,169,Ø,141,33,146
BB 83Ø DATA 169,4,141,34,146,173,34
F4 84Ø DATA 146,2Ø1,7,48,14,2Ø8,9
76 85Ø DATA 173,33,146,2Ø1,255,144,5
12 86Ø DATA 24Ø,3,76,57,148,173,31
D3 87Ø DATA 146,141,1Ø,148,173,32,146
B6 88Ø DATA 141,11,148,173,255,145,141
Ø5 89Ø DATA 29,146,169,Ø,141,3Ø,146
F1 9ØØ DATA 173,33,146,141,36,148,173
87 91Ø DATA 34,146,141,37,148,173,29
ØC 92Ø DATA 146,141,255,7,238,31,146
75 93Ø DATA 2Ø8,3,238,32,146,238,33
B6 94Ø DATA 146,2Ø8,3,238,34,146,76
D6 95Ø DATA 232,147,173,2,3,141,29
27 96Ø DATA 146,169,Ø,141,3Ø,146,173
63 97Ø DATA 29,146,2Ø1,Ø,2Ø8,26,173
92 98Ø DATA 3Ø,146,2Ø1,Ø,2Ø8,19,169
CD 99Ø DATA Ø,133,32,169,4Ø,133,33
1E 1ØØØ DATA 169,Ø,133,34,169,24,133
31 1Ø1Ø DATA 35,76,237,148,169,Ø,133
B1 1Ø2Ø DATA 138,169,6,174,3Ø,146,172
DA 1Ø3Ø DATA 29,146,32,1,149,142,3Ø
2E 1Ø4Ø DATA 146,14Ø,29,146,76,237,148
ØE 1Ø5Ø DATA 173,29,146,141,2,3,56
Ø8 1Ø6Ø DATA 173,29,146,233,1,141,31
19 1Ø7Ø DATA 146,173,3Ø,146,233,Ø,141
EF 1Ø8Ø DATA 32,146,169,Ø,133,138,169
56 1Ø9Ø DATA 6,174,32,146,172,31,146
52 11ØØ DATA 32,1,149,142,32,146,14Ø
C9 111Ø DATA 31,146,24,169,3,1Ø9,31
D1 112Ø DATA 146,141,31,146,169,3,1Ø9
E8 113Ø DATA 32,146,141,32,146,96,173
9C 114Ø DATA 2,3,141,31,146,169,Ø
3A 115Ø DATA 141,32,146,169,4,133,138
92 116Ø DATA 169,Ø,174,32,146,172,31
EØ 117Ø DATA 146,32,1,149,142,32,146
E6 118Ø DATA 14Ø,31,146,56,169,Ø,237
B4 119Ø DATA 31,146,141,31,146,169,146
```

```
1A 1200 DATA 237,32,146,141,32,146,96
09 1210 DATA 104,133,118,104,133,217,104
ED 1220 DATA 141,89,170,169,141,141,1
39 1230 DATA 2,169,1,133,52,96,133
27 1240 DATA 137,132,135,134,136,169,0
18 1250 DATA 133,133,133,134,70,136,102
A1 1260 DATA 135,144,13,24,165,137,101
7C 1270 DATA 133,133,133,165,138,101,134
79 1280 DATA 133,134,6,137,38,138,165
9D 1290 DATA 136,5,135,208,227,164,133
9F 1300 DATA 166,134,96,133,134,132,135
1B 1310 DATA 160,0,169,0,145,133,200
ED 1320 DATA 208,2,230,134,138,208,4
1E 1330 DATA 198,135,48,4,202,76,53
E5 1340 DATA 149,96
```

Program 2: Windows In BASIC

```
72 100 WMAX = 3
DF 110 HIMEM: 34304: REM SEE TABLE 1
38 120 POKE 768,0: POKE 769,WMAX: POKE 77
       0,0
6F 130 POKE 32,0: POKE 33,40: POKE 34,0:
       POKE 35,24
5A 140 D$ = CHR$ (4)
3E 150 PRINT D$"BLOAD WINDOWS"
50 160 HOME
F9 170 PRINT TAB( 11)"A P P L E S O F T"
F0 180 PRINT
F3 190 PRINT TAB( 13)"W I N D O W S"
CD 200 PRINT : PRINT "--------------------
       --------------------": REM 40 DAS
       HES
69 210 VTAB 24: PRINT TAB( 11)"(C) 1986 C
       OMPUTE!";
15 220 VTAB 9: HTAB 5: PRINT "(D) DEMONST
       RATION OF WINDOWS"
6E 230 PRINT : PRINT TAB( 5)"(Q) QUIT THE
        PROGRAM"
D4 240 VTAB 18: PRINT "ENTER YOUR SELECTI
       ON AND PRESS 'RETURN'"
06 250 VTAB 20: HTAB 1: CALL - 868: INPUT
        "";A$
20 260 IF A$ = "D" THEN 300
46 270 IF A$ = "Q" THEN 780
21 280 GOTO 250
E1 290 REM WINDOW 1
3A 300 POKE 32,5: POKE 33,30: POKE 34,4:
       POKE 35,19
82 310 POKE 768,0: CALL 37376
51 320 GOSUB 860
AE 330 INVERSE : VTAB 5: HTAB 11: PRINT "
       WINDOW 1": NORMAL
1E 340 VTAB 7: HTAB 3: PRINT "ONE METHOD
       OF OPENING AND"
5A 350 HTAB 3: PRINT "CLOSING WINDOWS IS
       TO USE"
C3 360 HTAB 3: PRINT "A MENU."
28 370 VTAB 12: HTAB 3: PRINT "(W) OPEN W
       INDOW 2"
8D 380 HTAB 3: PRINT "(X) CLOSE WINDOW 1
       AND"
8D 390 HTAB 7: PRINT "RETURN TO MAIN MENU
       "
BD 400 VTAB 16: HTAB 3: PRINT "ENTER SELE
       CTION"
0D 410 VTAB 16: HTAB 19: GET A$
C6 420 IF A$ = "W" THEN 460
88 430 IF A$ = "X" THEN POKE 768,1: CALL
       37376: GOTO 250
18 440 GOTO 410
EB 450 REM WINDOW 2
F7 460 POKE 32,0: POKE 33,25: POKE 34,0:
       POKE 35,13
8F 470 POKE 768,0: CALL 37376
5E 480 GOSUB 860
26 490 VTAB 1: HTAB 9: INVERSE : PRINT "W
       INDOW 2": NORMAL
8A 500 VTAB 3: HTAB 3: PRINT "ANOTHER MET
       HOD OF"
30 510 HTAB 3: PRINT "OPENING & CLOSING"
37 520 HTAB 3: PRINT "WINDOWS IS A CONTRO
       L"
89 530 HTAB 3: PRINT "CHARACTER."
1D 540 VTAB 8: HTAB 3: PRINT "CTRL-W OPEN
        WINDOW 3"
E8 550 HTAB 3: PRINT "CTRL-X CLOSE WINDOW
        2"
E3 560 HTAB 10: PRINT "AND RETURN TO"
8C 570 HTAB 10: PRINT "WINDOW 1"
5F 580 VTAB 12: HTAB 3: GET A$
51 590 IF A$ = CHR$ (23) THEN 630
18 600 IF A$ = CHR$ (24) THEN POKE 768,1:
        CALL 37376: GOTO 410
9B 610 GOTO 580
F7 620 REM WINDOW 3
62 630 POKE 32,15: POKE 33,25: POKE 34,9:
        POKE 35,21
8B 640 POKE 768,0: CALL 37376
5A 650 GOSUB 860
58 660 VTAB 10: HTAB 9: INVERSE : PRINT "
       WINDOW 3": NORMAL
D2 670 VTAB 12: HTAB 3: PRINT "THIS IS TH
       E LAST WIN-"
E6 680 HTAB 3: PRINT "DOW. TYPE SOMETHING
       "
EC 690 HTAB 3: PRINT "BELOW, THEN PRESS"
8F 700 HTAB 3: PRINT "'RETURN' TO CLOSE"
EF 710 HTAB 3: PRINT "WINDOW 3 AND RETURN
       "
6D 720 HTAB 3: PRINT "TO WINDOW 2."
F3 730 VTAB 19: HTAB 3: PRINT "..........
       ..........."
EE 740 VTAB 19: HTAB 3: INPUT "";B$
96 750 POKE 768,1: CALL 37376
A6 760 GOTO 580
69 770 REM QUIT
5A 780 HOME
CD 790 IF B$ = "" THEN B$ = "NOTHING"
DA 800 VTAB 10: PRINT "YOU ENTERED"
4E 810 VTAB 12: PRINT CHR$ (34)B$ CHR$ (3
       4)
21 820 VTAB 14: PRINT "ON WINDOW 3"
A7 830 VTAB 20: PRINT "GOODBYE" CHR$ (7)
       CHR$ (7)
9B 840 END
36 850 REM BORDER
0B 860 BL$ = "
                   ": REM 40 SPACES
7D 870 I = PEEK (770)
13 880 IF I = 0 THEN RETURN
73 890 I = 771 + 6 * (I - 1)
7F 900 WL = PEEK (I):WW = PEEK (I + 1):WT
        = PEEK (I + 2):WB = PEEK (I + 3)
30 910 INVERSE
BE 920 HTAB 2: PRINT LEFT$ (BL$,WW - 2);
63 930 VTAB WB: HTAB 2: PRINT LEFT$ (BL$,
       WW - 2);
3A 940 FOR I = WT + 2 TO WB - 1
A7 950 VTAB I: HTAB 1: PRINT " "
20 960 VTAB I: HTAB WW: PRINT " ";
F3 970 NEXT I
DC 980 NORMAL
2B 990 RETURN
```