

The FKey Manager

The fact that you're reading this document means (I hope!) that you've also obtained the file FKM3.PIT, containing three distinct files: **FKey Manager**, an application; **Pop-Keys**, an INIT resource; and **KeyPad**, an INIT resource. These three utilities, together with this documentation, can, I believe, increase your enjoyment of the Macintosh and increase the productive work you do with it. Read on to find out how.

An Aside...

Before proceeding, remember that *the FKM utilities work only on machines with 128K or larger ROMS, that is, Mac 512E's, Mac Plusses, Mac SE's, and Mac II's!!!!* If you have an older machine, especially if you've spent good money and time downloading these files, I'm sorry. Where I have posted the program, I have included a notice -- 128K+ ROMs ONLY! -- to try to forestall this; unfortunately not everyone who re-posts the program to new BBSes is mindful of this problem. If you do have this unfortunate experience, let the Sysop of your BBS know to post such a notice; or, tell me where you got the program from, and I will take some action.

What Are FKEYs, and What Good Are They?

Every Mac owner knows what FKEYs are even if she doesn't know the name -- when you press Command-Shift-4 to print what's on the screen, you're using an FKEY. Every factory Mac comes with this and 3 more FKEY functions: saving the screen as a MacPaint file, ejecting the disk in the external drive, and ejecting the internal disk. These are accessed by holding down the Command (the key with the "cloverleaf", and the Apple on newer keyboards) and Shift keys and simultaneously pressing the top-row number keys 3, 2, or 1. Furthermore, the Mac ROM provides for you to add more FKEY functions -- specifically six more, at which point you run out of number keys on the top row.

The **advantages** of this FKEY scheme are

- the tasks are available at any time, no matter what program is running on the Mac.
- they add functionality to existing programs without taking up much space on your disks.

Unfortunately, there are **limitations** as well:

- FKEYs don't coexist with your program; unlike desk accessories, they seize control of the computer until they finish their task.

- You're limited to ten functions (one for each number key in the top row).
- The three-keystroke method of invoking the functions is somewhat awkward and distinctly "un-Mac-like". There's no way to "preview"; you have to remember which number goes with which FKEY. It's opaque rather than transparent to the first-time user.

What does the FKey Manager package do about all this? It can't get around the first limitation: FKEYs still interrupt whatever you're doing to do what **they** want to do. However, you do escape the numeric limitation: you can have as many FKEYs as you want. And, you don't need to use the keyboard to access them, or remember which function goes with which number: you can choose the FKEY functions from a menu, by name. Of course you can still use the keyboard if you wish, in a number of ways.

The FKey Manager System

The FKey Manager System consists of three parts:

- FKey Manager, a standard Mac application, much like Apple's Font/DA Mover;
- Pop-Keys, an INIT resource, the heart of the FKey Manager system, which must be **dragged into the System Folder** in order to become active;
- KeyPad, an INIT resource, which must be **dragged into the System Folder** to become active.

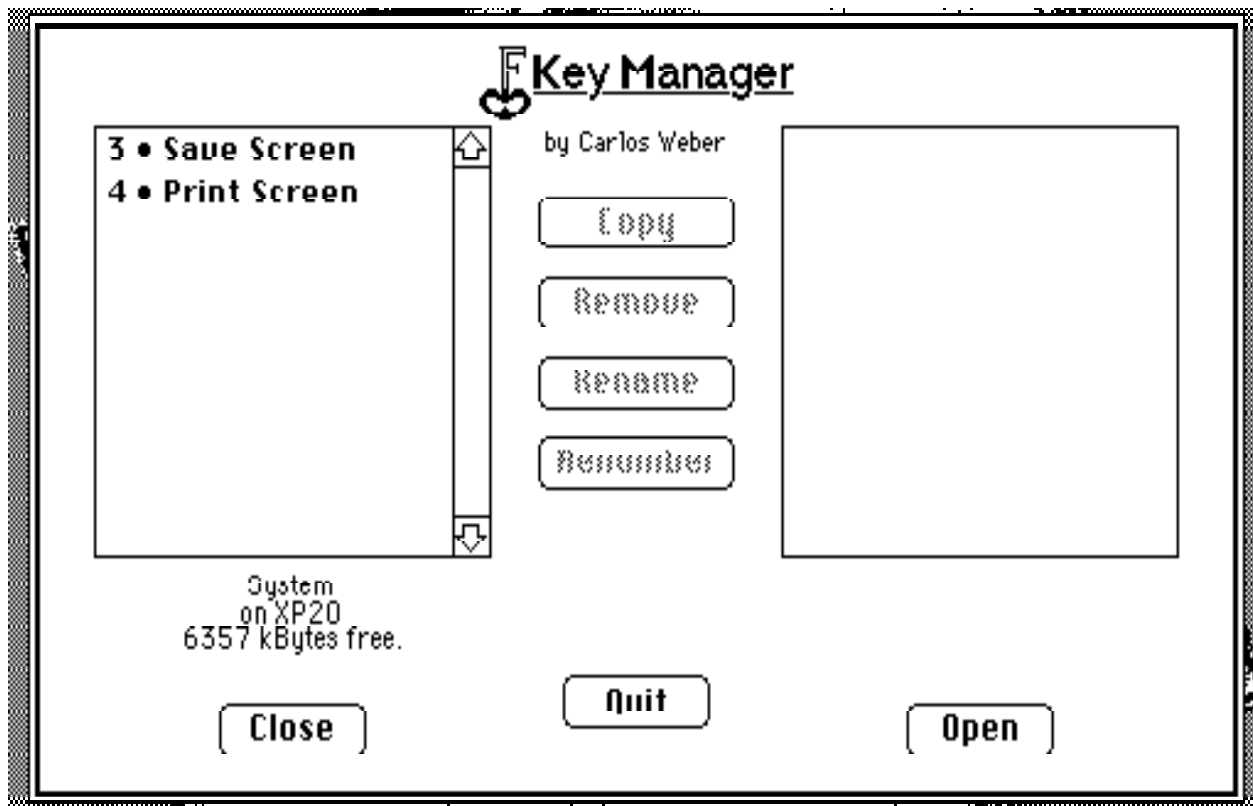
I recommend that you fool around with the FKM stuff, on a **back-up disk**, before you install it in the System(s) you use and depend on every day. Starting with a back-up disk containing a System Folder (including System and Finder) and any other applications you want to play around with, do the following:

- copy the FKey Manager application to that disk.
- copy the INIT resources Pop-Keys and KeyPad into the System Folder on that disk.
- Reboot your Mac, using the backup disk as your startup disk.

Now you're ready to experiment in safety and comfort; when you're satisfied you know what's going on you can use the same steps to install the FKM package into your everyday Systems.

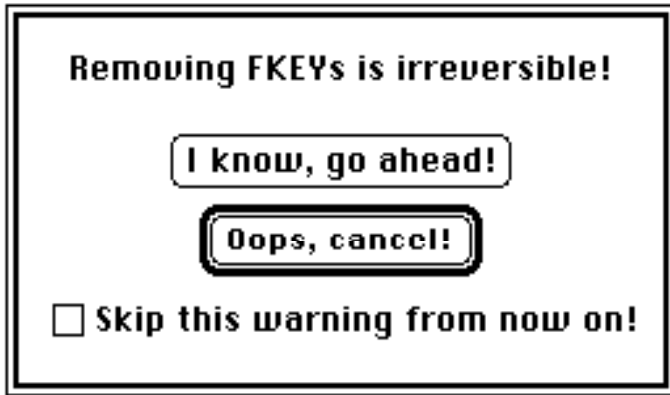
Using FKey Manager to Manipulate FKEYs

Let's assume you've got the FKey Manager (hereinafter termed FKM) on a disk -- a **back-up** disk, remember! -- and you've dragged the Pop-Keys and KeyPad INITs into the System Folder. Let's also assume that you have some FKEYs you want to install in your System and experiment with. Sources of FKEYs? Most users' groups have several disks full of 'em in their software libraries; they are available on commercial and local BBSes; and some are available as commercial programs (check out Dreams of the Phoenix' Q&D Utilities Disk #2!). If you open the FKM application (by double-clicking it) you'll see a screen that should remind you of Apple's Font/DA Mover:



Indeed, you'll find that FKM works pretty much the way the F/DA Mover does: open a file (use the Open/Close buttons at the bottom of the screen) and you see the FKEYs in that file, if any, displayed in the list window above the button. You can then select one or more FKEYs and apply the actions represented by the four middle buttons. The names of the buttons explain what they do, but there are a few things to know about each one.

You can **Remove** one or more FKEYs from a file. FKM won't let you do this, however, without issuing you a reminder:



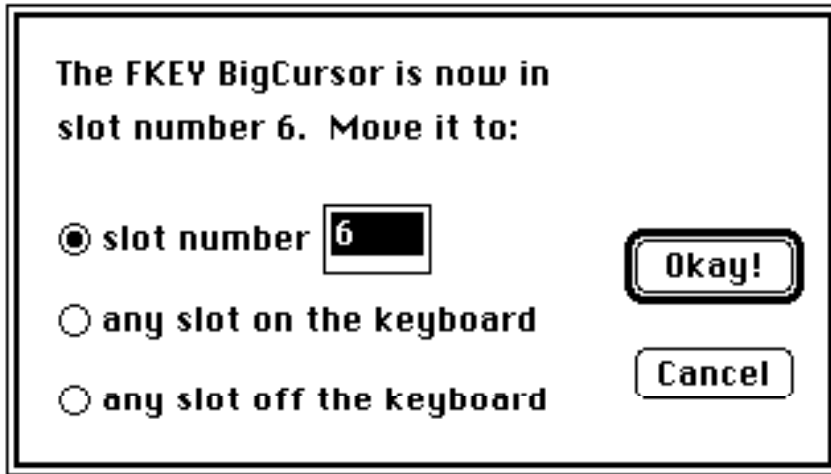
You can choose not to see the reminder, after the first time.

Renaming is also straightforward. You get this dialog box,



you fill in the blank, you press okay. If you're like me, you'll want all your FKEYs to have names, preferably short names, suggestive of their function. This is essential if you want to use the Pop-Keys menu to get at them. Don't worry if that last statement doesn't mean anything to you -- all will become clear.

You can **Copy** one or more than one FKEY from one file to another; **both** files must be opened. If you need to create a file to copy to, choose Open, then choose New and name your new file. When you copy an FKEY from an old file to a new, you have to make a choice about what number to assign it in its new home:



(This is the same choice confronting you when you **Renumber** an FKEY.) The first radio button lets **you** assign the new number -- the same one it had in the old file, or any other specific number you type in the box -- assuming you don't choose a number already belonging to an FKEY in the destination file (if you do, you'll hear about it!). Or, you can let **FKM** assign the FKEY a new number, in one of two ways. If you tell it to put the FKEY "on the keyboard", FKM will try to assign it a number in the range 0 to 9; FKEYs with these numbers can be invoked in the standard way, with the Command-Shift-# key combination -- in other words, from the keyboard. If you ask for a slot "off the keyboard", FKM assigns the FKEY a number larger than nine. This means the FKEY won't be accessible with a key combination, but only via the Pop-Keys menu. Again, don't worry if this doesn't seem to make sense just yet; the Pop-Keys menu has all the answers, and it's coming up very shortly.

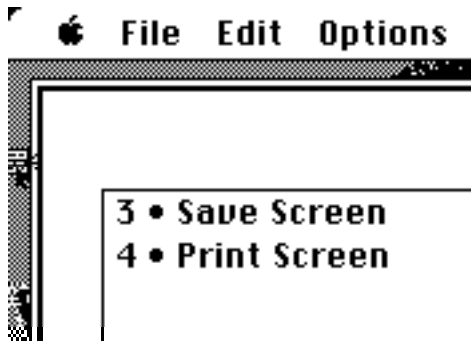
A few more things about copying: if you're copying multiple FKEYs, you have the above numbering choice to make for each FKEY; to ease the tedium somewhat, FKM "remembers" your last numbering mode selection. Thus, for example, if you wanted to copy five FKEYs to a new file, using the lowest available numbers, you would only need to select "any slot on the keyboard" for the first FKEY; after that, just hit the return key (or click Okay) with each appearance of the dialog box.

If you **really** don't like dialog boxes, there is a "mass copy" mode, which simply assigns all the selected FKEYs in the source file to free numbers in the destination file, beginning with the number 10 -- no questions asked. You invoke this operation (after selecting the FKEYs you want to copy) by pressing the Option key while clicking the Copy button.

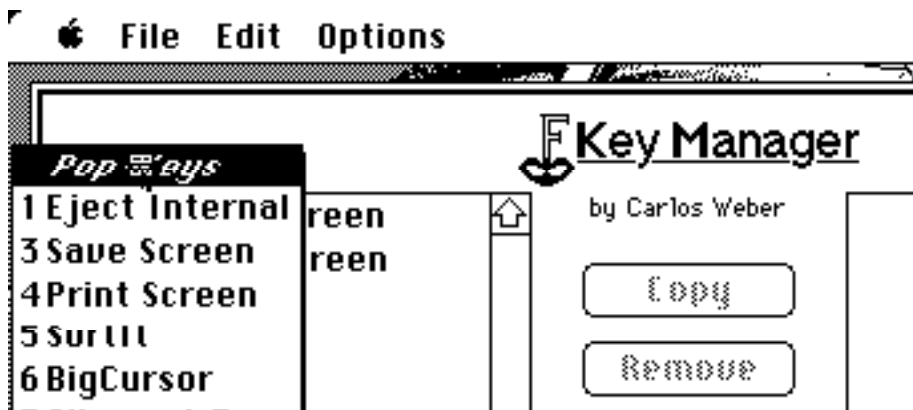
Finally, there's a specialized copy operation, called **Archiving**, which will be dealt with later, along with its complementary function, **Restoring**. For the moment, however, you have all the tools you need to completely customize your line-up of FKEYs: add as many as you want to your System file, change their names and numbers, get rid of ones you don't like, and generally create quite a bit of chaos.

Using the Pop-Keys Menu

At this point I want to assume that you have managed to install some FKEYs into your System file (on that **back-up** disk), and that all the FKEYs you've installed have names (don't forget FKEYs 3 and 4, the old Apple standards -- they need names, too, such as Save Screen and Print Screen). Mouse over to the left edge of your screen and observe what happens to the arrow cursor. You should see something like this:



The cursor should change to look like a miniature menu! (If it didn't, it means you either don't have the Pop-Keys INIT in your System Folder, or you didn't boot from the disk that *does* have Pop-Keys in its System Folder.) The little menu cursor is a signal: you've moused into the **magic rectangle**; if you click the mouse now, you'll get a new menu, like this:



The menu you see will no doubt have different items (they should be the names of the FKEYs you've installed in your System!), and its appearance may differ in other ways, too. You'll discover how to alter the appearance of the Pop-Keys menu to suit your taste in a minute. For now, just note that this menu works just like any other Macintosh menu! You select an item by dragging the mouse down over its name, and releasing. These "items", however, are your installed FKEYs. So, for example, to save a copy of the current Mac screen display as a MacPaint file, you could either

- press Command-Shift-3; or
- mouse to the **magic rectangle**, pull down the Pop-Keys menu, and choose Save Screen! Pretty slick.

Now that you see how the Pop-Keys menu works, you can understand the importance of naming all your FKEYs; otherwise they don't appear in the menu! You can also see that, with a menu to access them, you can have **as many FKEYs as you like** in your System. The ones you use most frequently, or need to invoke from the keyboard, can be put in slots 0 through 9 (using FKM, of course), and the rest can have numbers above 9 -- they'll still appear on the menu, where you can access them with a mouse-click! While testing Pop-Keys I used a System file with more than forty FKEYs; I only know a few people who would want that many, but you might use 12 or 15!

Here's another notable fact about the Pop-Keys menu: it's immortal! Go from the Finder to an application, back to the Finder, to another application... every time you mouse into the **magic rectangle**, Pop-Keys is there waiting. This makes sense; the functions FKEYs provide aren't limited to a single application, so they should always be available to you.

Now it's time to learn another one of Pop-Keys' tricks. Let's say you're fortunate enough to have a big screen monitor for your Mac, and it's just too far to drag that cursor all the way to the upper left-hand corner; or maybe you have a regular screen and you're an impatient sort. Well, wherever your mouse happens to be, you can get the Pop-Keys menu to appear, just by using the **Command Key** (on some keyboards this "Cloverleaf" or "Pretzel" key has an Apple on it, but in this manual I intend to disregard both those symbols and call it the Command Key, and you can silently call it -- and me -- whatever you wish). Try it. Leave the cursor somewhere in the middle of the screen, press the Command key, and click. You should see the Pop-Keys menu "pop" up in the middle of your screen under your cursor:



This "pop-up" menu works just the same way, except it's not in the menu bar; it's wherever your mouse happens to be when you "Command-click" ("Cloverleaf-click", "Apple-click", whatever). Now you see how Pop-Keys got its name!

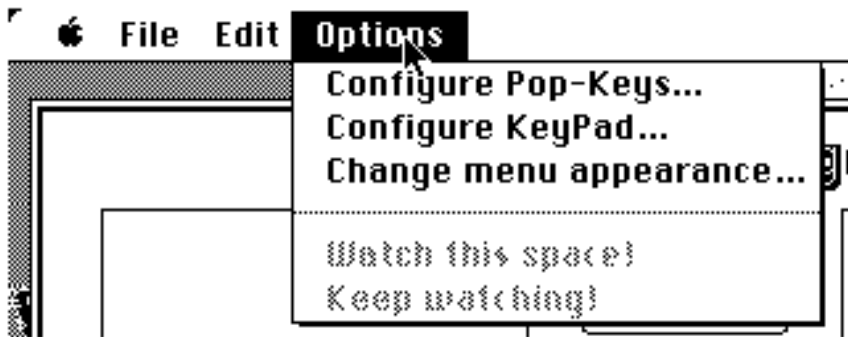
Notice, by the way, that in the picture above the cursor is selecting the first item, called "Pop-Keys". Try it yourself; you'll see that when you "Command-click" Pop-Keys into existence, the cursor is almost always highlighting the first item when the menu first appears. Should you inadvertently release the mouse button at this point (thus selecting that item!) not much happens -- you'll see a little message from me. This is that first item's sole purpose in life: to protect you from unwittingly selecting a **real** FKEY, and causing some action you don't really want, such as a disk ejection.

What about the last item, called *Other...*? This item has a function analogous to Lofty Becker's great desk accessory of the same name: it's a way for you to run FKEY resources that **aren't** installed in your System. Let's imagine you have an FKEY which you use occasionally, but not often enough to warrant installing it in your System and cluttering up the Pop-Keys menu. Instead, put the FKEY in its own file (use FKM to create a **separate** file for each FKEY), put it in a folder where you can find it again when you need it, and then use *Other...*'s "mini-finder" style dialog to Open it. It will run just as if it was installed in your System.

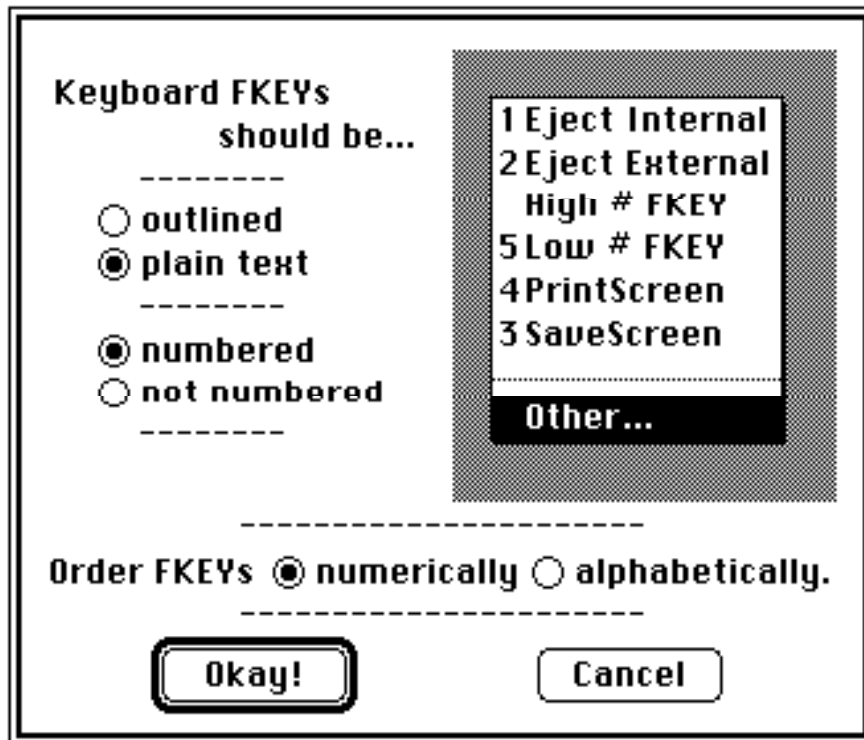
At this point you have four ways to get an FKEY to run: use the Command-Shift-# method from the keyboard, mouse to the **magic rectangle** and select it from the Pop-Keys menu, use the "Command-Click" method to "pop up" the Pop-Keys menu and select, or use the *Other...* item to run an FKEY from a file on disk. There are still two more methods to be discussed, but not just yet.

Customizing the Pop-Keys Menu

You've seen that you have a choice about how to invoke the Pop-Keys menu; you actually have quite a bit of control over the way Pop-Keys behaves and looks. You exercise this control from within the FKey Manager application. Return to FKM and pull down the Options menu:



Ignore for the moment the fact that the last 2 items on your menu may look different from the picture. We are concerned here with the first and third items only. We'll attack them in reverse order; if you choose "Change menu appearance..." you see the dialog box below:

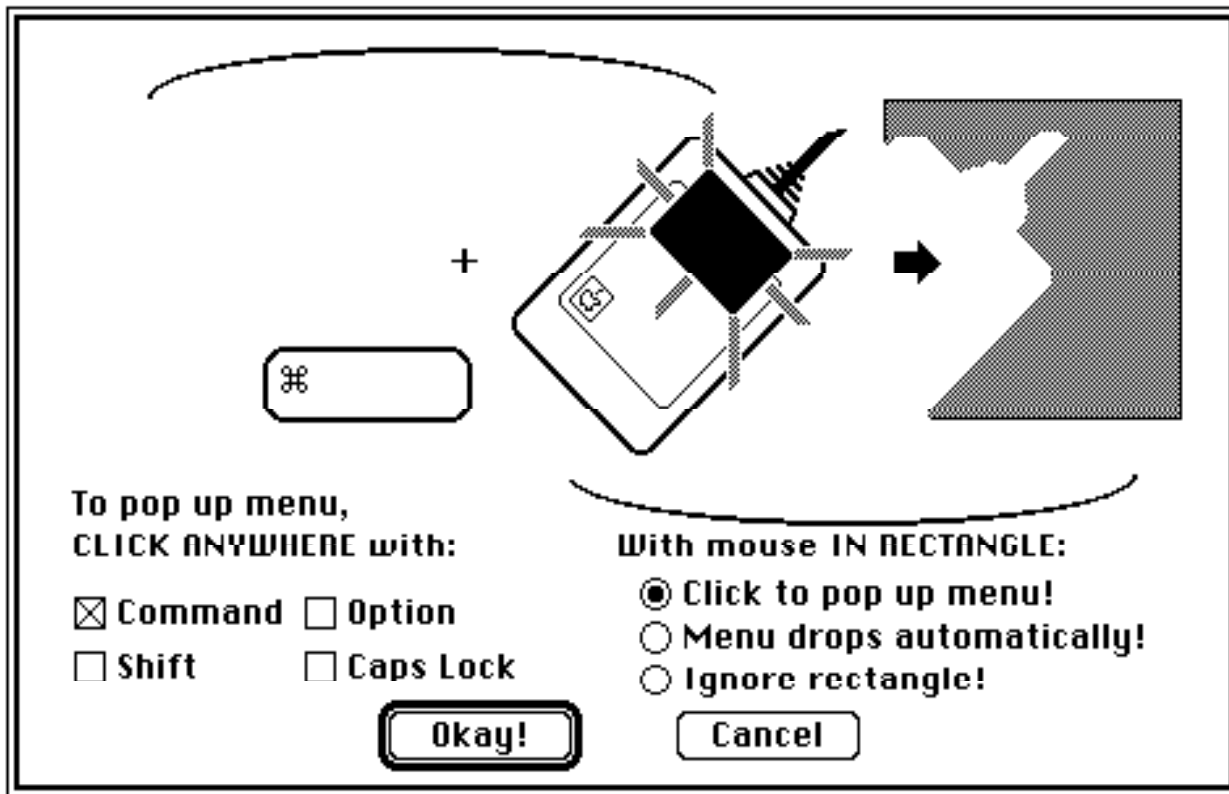


Here are three choices about the appearance of the Pop-Keys menu, which you can make according to your personal taste. The first two deal with ways of distinguishing visually between FKEYs with low numbers which you can invoke also from the keyboard, and higher-numbered FKEYs which are only accessible from the Pop-Keys menu. Remember? You might like to remind yourself which FKEYs are "keyboard" FKEYs by seeing their names *outlined* on the Pop-Keys menu. Or you might feel that the outlining creates visual "clutter". Up to you. Similarly, you might want the "keyboard" FKEYs displayed with their

corresponding numbers to remind you that they can be "Command-Shift-#"ed; I do this because I simply can't remember which FKEY goes with which number otherwise. If, however, you never use the keyboard to invoke FKEYs (and you don't have to if you are using Pop-Keys!), you may feel the numbers just get in your way. So, eliminate them. I hereby give you permission to play around until you find the Pop-Keys menu style which suits your particular pattern of FKEY use.

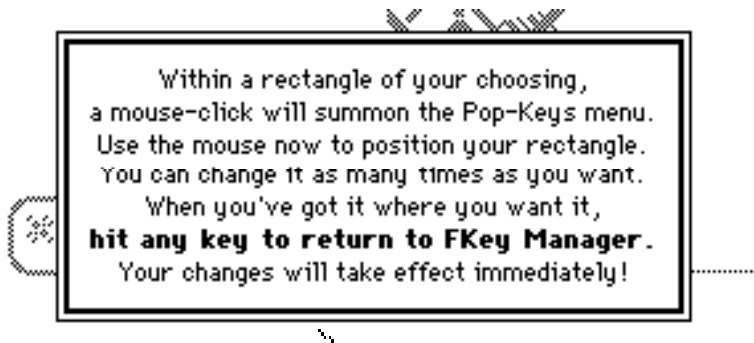
The last choice in this dialog box applies to all your FKEYs, not just the low-numbered ones. You can choose to have them listed alphabetically on the menu -- this is the way desk accessories are listed in the Apple menu -- or numerically, with the low-number "keyboard" FKEYs at the top of the menu. If you number your keyboard FKEYs, as explained above, this method looks somewhat neater than the alphabetical method. You can try it both ways and see which you like.

We're still not done with customizing Pop-Keys; in fact we've saved the most important things for last. So far the choices have all been cosmetic: changes in the way the menu appears. You can also individualize the way Pop-Keys **functions**, by choosing "Configure Pop-Keys..." from the Options menu:

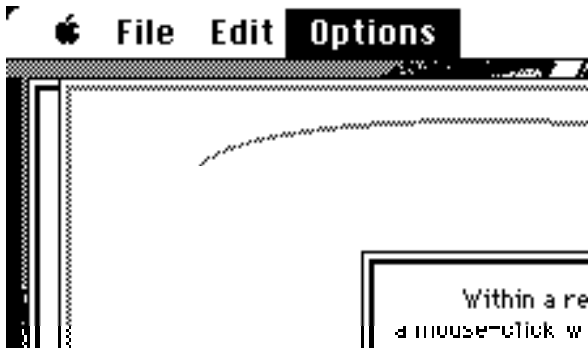


There's a lot going on here, but it's really quite straightforward. First, notice that the selected check box and radio button at the bottom of the dialog box are simply describing the *two ways of accessing the Pop-Keys menu* which we've already used. The check boxes on the left are telling us that, with the mouse ANYWHERE, we can get at Pop-Keys by holding down the Command key; we knew that. Notice also that, as foretold, you are not limited to the Command key; instead you could choose to "Option-click", or "Shift-click", or (save us!) "Command-Option-Shift-Caps Lock-click" to invoke Pop-Keys. All you need do is check or uncheck the corresponding boxes. (If you don't want this "pop-up" feature at all, you can simply uncheck **all** the boxes to disable it.) The right-hand side of the dialog, with the three radio buttons, describes the behavior of Pop-Keys when your mouse goes into the **magic rectangle**. Currently Pop-Keys is set to behave like a normal menu: click the mouse, the menu drops, you drag to the item you want, and release the mouse button to select. However, if you're a fan of Don Brown's EZ-Menus, you can make Pop-Keys behave the same way, with the second radio button: when the mouse finds the **magic rectangle**, the menu just drops into view with no mouse-click; you drag to your item and click to select it. Try it; you may like it. If you're a sourpuss and don't believe in magic or **magic rectangles**, you can ignore the **rectangle** method of accessing Pop-Keys, and just use the "Command-click" method. I recommend an initial period of experimentation; you'll probably settle on one method of "popping" which seems natural to you.

If you've tried using this dialog box, you've probably gotten a surprise: when you click "Okay" to quit, you still have to make one more choice -- where to position the **magic rectangle**. You see a message in the middle of your screen:



You should also notice that part of your screen (probably a long thin rectangle along the extreme left edge) is blackened:



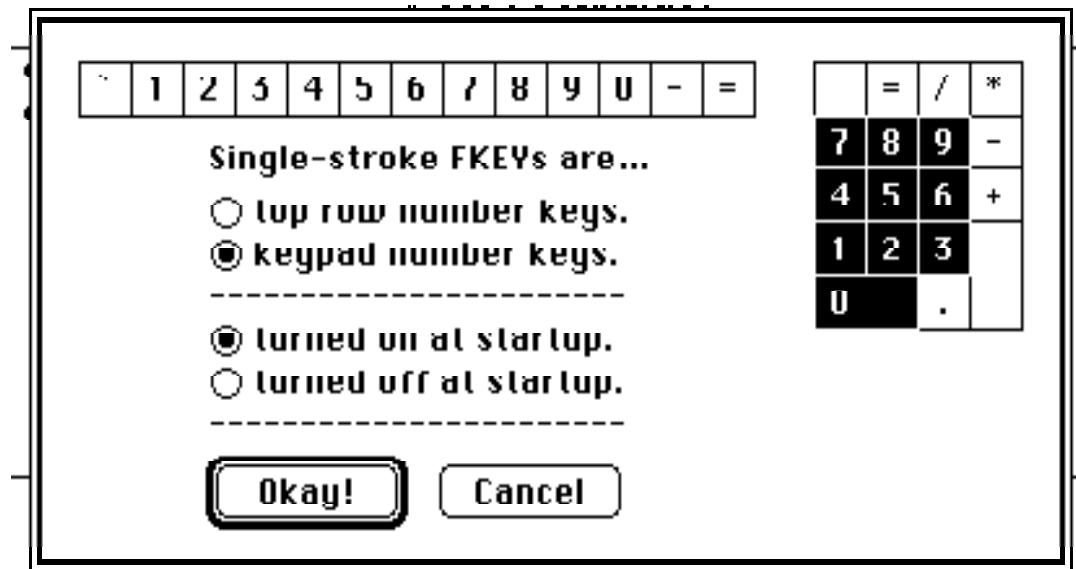
The **magic rectangle** can be anywhere on the screen; just click the mouse where you want one corner of the rectangle to be, then drag to the diagonally opposite corner. When you release the mouse button your new **magic rectangle** will appear in black (or any other pattern you choose, if you want to use ResEdit to change it as explained later on), and the old one will disappear. You can reposition the **magic rectangle** as many times as you want -- it won't "take" until you hit a key on the keyboard.

A word of warning: the **magic rectangle** should be a portion of your screen where your mouse doesn't go very often, otherwise the unwanted appearance of the little menu cursor (or the Pop-Keys menu!) can be annoying, to say the least. The upper leftmost corner of the screen (the portion of the menu bar to the left of the Apple menu) works pretty well, **unless** you are also using the EZMenu INIT, in which case positioning the **magic rectangle** in the menu bar can cause problems. A thin rectangle along either edge of the screen also works well, and tends to keep the Pop-Keys menu out of your way but accessible when you want it. Again, your best bet is to try out several different positions until you find the one that works best for you.

Keyboard Alternatives for FKEY Access

On other computers, "function keys" work differently than they do on the Mac: there are separate keys, usually 10 or 12 of them, which have no other function on the keyboard, and which let you do something with a single keystroke. The new Macintosh Extended Keyboard (for the Mac SE and Mac II) has these keys also, but most programs (including FKey Manager) don't take advantage of them yet. On those "other" computers, the "something" you do with a function key is usually different in every program. People tend either to love this capability, wishing the Mac had it or even using its absence as a reason to think of the Mac as a "toy" computer, or they tend to hate it and damn it as confusing and "un-Mac-like" in spirit. My own feeling is: there should always, first and foremost, be an "obvious", transparent, immediate, and "Macintosh-like" way of doing something, but if a user wants to do that thing another way, which *to that user* seems more natural or faster or more familiar, then within reasonable limits she should be able to. So... if you *really want* to use your keyboard to invoke FKEYs... how about a method that lets you do it with **a single** keystroke instead of three?

Here's how it works (assuming you have one the the newer ADB keyboards, a Mac+ keyboard, or a separate numeric keypad -- if not, this method will not be very convenient for you): with the numeric keypad either built in or added on to the Mac keyboard, you have two sets of number keys. The Keypad INIT (you put it in your System Folder quite a few pages back) lets you dedicate one set to your "keyboard" FKEYs, the ones you have assigned the low numbers to. Go back into FKM and choose "Configure KeyPad..." from the Options menu:



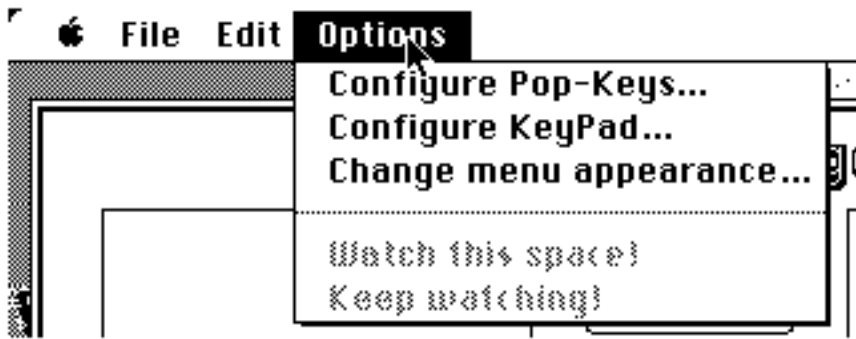
The idea is that one or the other set of number keys becomes a set of dedicated "single-stroke" function keys. For example, hitting 3 on the numeric keypad would cause the "Save Screen" FKEY to run, and you'd get a MacPaint copy of your current screen. With the top two radio buttons you can toggle back and forth between using the top-row keys or the keypad keys as "single-stroke FKEYs"; the other set of number keys behave normally when you need to type numbers.

Now if you're not expecting it, it can be quite a shock to open an Excel spreadsheet, start entering some data with the keypad, and suddenly be bombarded with disk ejections, printers coming alive, and other FKEY actions! So, even though the KeyPad INIT is in your System folder, you can choose not to have it enabled at startup, until you choose to turn it on. Or you can just let 'er rip from the beginning. That's the function of the second group of radio buttons.

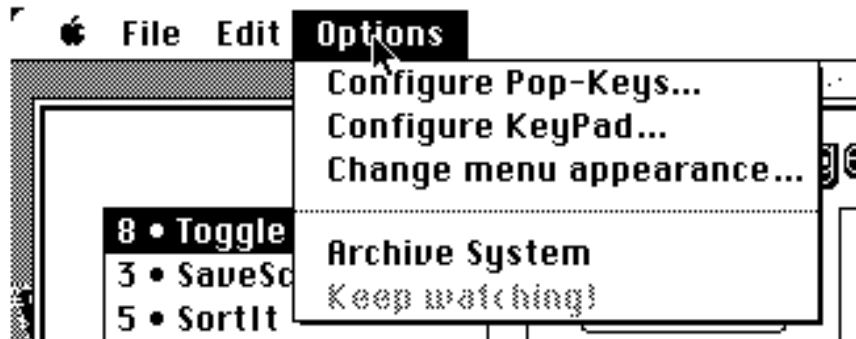
Whichever way you start out, you will want to be able to switch the KeyPad on and off without rebooting each time: if you start up with KeyPad off, you need to turn it on when you want it; if it's on when you boot, you may need to turn it off temporarily to fill out a spreadsheet. You toggle the KeyPad on and off with the following key combination: Command-Shift-Option-k. (The "k" is KeyPad's first initial.) Pressing these four keys at the same time will turn KeyPad **on** if it's currently **off**, or **off** if it's **on**.

Archive and Restore

These functions are designed to assist you with the task of System File Upkeep and Maintenance. Archiving first: when you **archive** the FKEYs from a file, you *extract all the FKEYs* from some file which contains FKEYs *and other kinds of resources* -- this will usually be the System File, though applications can contain FKEYs, too. The FKEYs go to a special file created to receive them, and they preserve the numbers and names they had in the original file. So how does one archive a file? The first requisite is that the file must be opened in FKM. If you've played around with FKM, by now you will have noticed that sometimes the Options menu looks like this:



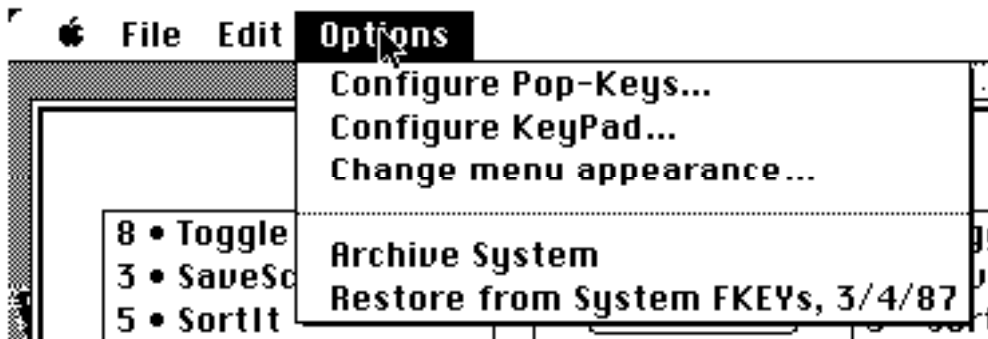
while at other times it may look like this:



Whenever a suitable file is open, FKM offers you the choice of Archiving it. FKM defines "suitable" as follows: the file must contain FKEYs *and* other kinds of resources. These are the only situations in which it makes sense to "extract" the FKEYs. The System File fits this description, and it's probably the only one you will ever want to archive from. The System under consideration **need not** be the "active" System; in fact it's not a good idea to perform *any* operation on the System you are currently running under. I've tested FKM and this Archive function a lot, but I don't recommend you Archive the active System. *Always have a back-up; don't monkey with the System on your startup disk while you're running!*

Enough said. You've got *some* System file open, and you want to Archive it. You select "Archive System" from the Options menu, and you are asked to provide a name for the file which will receive the archived FKEYs. FKM suggests a name for you; for example, if you are archiving from the System file, and it's March 4, 1987, the suggested name will be "System FKEYs, 3/4/87". Pretty imaginative, huh? You can change the name if you want.

Restoring goes the other way. Imagine it's now March 12, 1987, and your hard disk crashes, blowing away your System file in the process. No problem; everything was backed up, right? In particular, when you reconstruct your System you like to start with a virgin, plain, unadorned System and add in your favorite fonts, DAs, etc. It's a snap to add back all your FKEYs, because you can **Restore** them from your archived file. You just have to find the file you named "System FKEYs, 3/4/87", or whatever, and open it with FKM. You can double-click the file's icon (it looks like a trunk) or open FKM first and then open the file. The options menu will now appear like this:



Choose the Restore from... menu item and you'll be asked what file you want to restore; you probably want the System, but remember you should be working on a System that's not the active one! If there are any FKEYs in the System file you are restoring to, you'll get a warning that these FKEYs may be overwritten by the FKEYs in the archived FKEY file. The premise of restoring is that you want the FKEYs in the System file to be *exactly as they were on 3/4/87*, so FKM considers it okay to wipe out any stray FKEYs that get in its way while accomplishing that task. As long as you understand the limitations of archiving and restoring, you shouldn't get into trouble.

Menu by Menu...

We've already covered all the items in the **Options** menu; along with the buttons in the center of the main dialog screen which select the various FKEY-manipulating options, they comprise the real meat of the FKey Manager application.

The **Apple** menu permits access to desk accessories, as you've come to expect well-behaved Mac applications to do. This is a distinct difference from the behavior of the Font/DA Mover, by the way. The About... item informs you which version of FKM you have (these documents go with version 3.0), and give credit to *some* of the people who helped me with ideas, code, suggestions, bug reports, etc. Check it out at least once.

The **File** menu has only one item: Quit. It's the equivalent of the button of the same name in the main dialog box. Want to hazard a guess as to what it does?

The **Edit** menu is for the convenience of desk accessories which expect to find the standard commands (Cut, Copy, Paste...) in this location. These commands are not used from within FKM, and are dimmed unless a desk accessory window is open.

Command-Key Equivalents

Some FKM actions, normally initiated by selecting a menu item or clicking a button in a dialog, can also be accomplished with the keyboard. You use the Command key for this:

<u>To do this:</u>	<u>Press Command-</u>
Quit	Q
Copy	C
"Mass Copy"	Option-C
Remove	R
Rename	N
ReNUMBER	#

Hidden Features

In this section I've attempted to gather all the tricks and short cuts in FKM, Pop-Keys, and KeyPad that you probably wouldn't find eventually on your own through exploration. Some of them have been mentioned already; some are only listed here.

- *Command-key equivalents* are listed above. These affect the central row of buttons in FKM's main dialog box.
- You can *change the pattern* FKM uses to show the magic rectangle in the "Configure Pop-Keys..." dialog sequence. You have to know how to use ResEdit, something this manual doesn't have space to explain. Using ResEdit, open a **backup copy** of FKey Manager, and look for the 'PAT ' resource -- there's only one, called "Magic Rect", with ID #6666. It's solid black, but you can use ResEdit to change it to anything you want. FKey Manager will use this pattern to paint the magic rectangle when you are changing its location with the "Configure..." dialog.

- *Getting rid of Pop-Keys or KeyPad temporarily:* if at any time while you're using your Mac you don't want Pop-Keys or KeyPad to function, you can shut them down temporarily. I already mentioned how to do this for the **KeyPad** INIT: press the key combination Command-Shift-Option-**k**. Want to guess how to toggle **Pop-Keys** off and on again? Very good -- the key combination Command-Shift-Option-**p** will turn Pop-Keys off if it's on, or on if it's off.
- *Booting without Pop-Keys or KeyPad:* If you don't want either Pop-Keys or KeyPad around during a *whole session* of using your Mac, there are a number of things you can do. Most straightforwardly, just remove either or both from your System Folder and reboot. If this is too much trouble, or if you forgot to do it before you turned off your machine, all is not lost. Holding down the "**k**" key while the machine is starting up ("Welcome to Macintosh.", etc...) will result in **KeyPad** not being loaded into memory and thus not being operative for the entire session until you restart the Mac. If, for example, you're having problems with a new utility you just downloaded from somewhere, and you want to see if KeyPad could be the source of the trouble, you could boot without KeyPad by holding down the "**k**" key and see if the trouble goes away. (Then, of course, you would inform me and the author of the utility so that we could fix the problem!) Pop-Keys works the same way: holding down "**p**" during the boot sequence prevents **Pop-Keys** from loading and becoming active for that session. Still one more trick: if you hold down the "Shift" key while booting the Mac, *neither* Pop-Keys nor KeyPad will load; in fact, most INIT resources will decline to load under these circumstances. This is a useful "bail-out" if you have a) a problem, b) a lot of INITs in your System Folder, and c) no idea where to begin looking for the source of the trouble.
- *Finding FKEYs with FKM's "Open" button and Pop-Keys' "Other..." item:* when you're looking for an FKEY file, either to change it in some way with the FKey Manager program or to execute it from the Pop-Keys menu with "Other...", there's a built-in short cut which you can take advantage of, if you store all your individual FKEY files in the same place. FKM and Pop-Keys will remember where you last looked for an FKEY, and open to that folder first when you look again. So, if all your FKEYs are in the same folder on your disk, you won't have to do any hunting in the Standard File dialog after the first time.

Using FKM and Pop-Keys with Suitcase™

At the time of this writing (early August 1987), only a few days remain before the official release of one of the most useful and slickest Macintosh utilities ever. No, I'm not blowing my own horn -- I'm referring to Suitcase™, an INIT utility developed by Steve Brecher and available by direct mail from

Software Supply
599 N. Mathilda Ave. #210
Sunnyvale, CA 94086-3504

or by calling 800-443-0100, ext. 451.

Using Suitcase allows you to have instant access to fonts, desk accessories, and... yes, **FKEYs**! without having to install them into your System file. The original idea was to get around the limitation of 15 DA's imposed by Apple's Font/DA Mover; thanks to Suitcase's smooth, invisible trickery you can have as many DA's as you want under your Apple menu. Steve has extended Suitcase's capabilities to include FKEYs, and has graciously allowed me to test FKM and Pop-Keys with pre-release versions of Suitcase. They work very well together, and it's a great combo! Here's how I have things set up: my System *File* is almost exactly as it came off the original Apple disk; the only change I've made to the FKEYs is to name FKEYs 3 and 4 so they would function under the Pop-Keys menu. All the other dozen or so FKEYs I commonly use are in a separate FKEY file called "FKEYs", which I created with the FKey Manager, and which lives in my System *Folder*. The Suitcase INIT (also in the System Folder) makes Pop-Keys and the rest of the Mac believe that all those extra FKEYs are really installed in my System, so that they appear in the Pop-Keys menu, are runnable from the keyboard, etc. It's so slick that you forget it's happening. If you are reading this manual, let me assure you that you are the type of Mac user who *needs* Suitcase. So buy it.

DAs as FKEYs -- the FKEY Maker

Lofty Becker is a Macintosh programmer who has probably written more FKEYs than any other single individual, and thus has to bear some of the responsibility for the existence of this program. He's now contributed yet another nifty utility to the Mac world in the form of the small application FKey Maker. Using this program you create FKEYs which invoke any of your installed desk accessories. Why would anyone want to do a crazy thing like that? Suppose you have one DA you use *a lot*, such as Disk Info or DiskTop or Disk Tools or McSink; it would be nice to bring it up with one keystroke, and you can do this by making the appropriate FKEY (use Lofty's program!), numbering it "on the keyboard" with FKM, and using KeyPad. It sounds elaborate until you try it; once you try it you won't be able to get along without it. Lofty's program is free, and available from users' groups and for downloading from CompuServe (probably from the other major online services as well).

Incompatibilities

Although users of previous versions of FKM won't notice much difference, the innards of the program have been extensively revised to function well with Apple's "Universal" system software, System 4.1/Finder 5.5; users of Mac Plusses, Mac SE's, and Mac II's should be using this version of the System or one of its descendants. FKey Manager will also run fine on machines with System 3.2 (the officially preferred System version for Mac 512Es). As mentioned above, FKey Manager will *not* run correctly on earlier Macintosh models, such as the 128K Mac and 512K Mac, that contain the 64K ROMs.

Users of **Macintosh II's** will find that Pop-Keys and KeyPad function without a hitch. If you go to run the FKey Manager application with your Mac II in anything other than monochrome mode, however, you will be greeted by a dialog importuning you to switch into monochrome mode before going further. Some of FKey Manager's visual effects, specifically Mike Morton's great "dissolve" routines (thanks, Mike!), don't work unless the "screen depth" is 1. These routines are used in the "About FKey Manager..." dialog, and in the "Configure Pop-Keys..." menu item sequence, where the magic rectangle position is chosen. So, the easiest way to avoid trouble on your Mac II is just to heed the warning dialog and use the Control Panel to change your monitor to black&white mode when using FKey Manager. In reality, if you need to use FKM for something other than changing Pop-Keys' configuration, you can just ignore the warning and proceed, for example, to renumber an FKEY. Everything except the "About..." sequence and the "Configure Pop-Keys..." sequence should work.

Let me reiterate that these problems with "deep" screens affect you *only* when using the FKey Manager application -- the normal use of the Pop-Keys and KeyPad INITs in your other applications is *not* affected.

Within the limitations imposed by the fact that I am one person with one Mac (Plus), with one real-world job having nothing to do with computers, and one income to buy software with, I've tried to make FKM, Pop-Keys, and KeyPad work with as many Mac products as I could. There are one or two conflicts I have not been able to resolve, so the best I can do is tell you about them.

- **Dark Castle**, the highly acclaimed game from Silicon Beach Software, is incompatible with Pop-Keys. To get its great graphic effects, Dark Castle uses an area of memory called the "Alternate Video Page", a feature of the Mac architecture which probably will not exist on future Macs, according to Apple. Pop-Keys uses part of this area of memory for its own purposes, following Apple's guidelines for resources of its type (INIT). Since I'm following the rules and they're not, and since side-stepping the incompatibility would require a *major* rewrite of Pop-Keys, I'm not going to fix this one. Too bad, because Dark Castle is neat. Other programs which use the "alternate video page" can be expected not to work with Pop-Keys, either.
- **MacLightning 2.0**, a pretty good spelling checker program from Target Software, doesn't get along with Pop-Keys. I've tried, and failed, to isolate and fix the incompatibility.
- **LaunchKey**, an FKEY utility for going from one application to another without returning to the Finder, has an irreconcilable conflict with Pop-Keys.
- Users of some versions of **Tempo**, the macro utility, may have trouble.

That's All, Folks!

This information is all you need (probably more than you need) to use and enjoy the FKey Manager, Pop-Keys, and KeyPad. I hope they increase your enjoyment of your Macintosh, and of FKEYs.

The FKey Manager package is free, but it is **not** in the public domain. It is Copyright ©1987 by Carlos A. Weber, M.D. (that's me!). Commercial distribution is expressly prohibited. You may, however, distribute it for free, provided that

- you don't charge for it
- you don't alter the programs in any way, without my express permission
- you distribute this documentation along with the program.

From these conditions, my intentions should be clear to all right-minded folks: I didn't write this utility to make a profit from it, and I don't intend anyone else to profit from it either; I want to be assured that mutant versions of FKM don't proliferate wildly; I want anyone who obtains the program and has a problem with it to be able to reach me, so that I can try to solve the problem.

Contact me with any suggestions, problems, bugs, praise, gripes, etc... I am:

Carlos Weber, M.D.

250 Douglass Street #12

San Francisco, CA 94114

(415) 861-8956 (home telephone # - San Francisco time!)

CompuServe PPN# 70277,726

MCI CWeber

AppleLink D0639

Please notice that I am **not** a programmer by profession, and don't routinely stay up until 3 a.m. The best time to reach me is in the evening (7 pm to 10 pm, Pacific time). I have an answering machine and will return your call if you call at another time.

If you send me a blank disk and a SASE, I will send you the latest version of the FKM package and its documentation -- eventually! (I may wait until I fix just one more bug...)