

geoSHELL
user's manual

geoSHELL V2.2

user's manual

program and manual are
Copyright (c) 1992,2000 by Maurice Randall
All Rights Reserved

Second Edition, First Printing, November 2000

Created using GeoWrite running in Wheels 128 and/or Wheels 64. Formatted and printed on a PostScript laser printer using PostPrint 3. All operations were performed with either a Commodore 64 or 128.

No part of this manual, nor the program, nor any files on the program disk may be copied or duplicated in any manner, except for the registered owners own backup or archival purposes. Anyone not abiding by this is not only hurting the author by lost revenues, but also hurting all GEOS users by not encouraging the author to continue development of this and other programs.

Warranty: The original program disk is warranted for a period of 12 months from the date of purchase. It is not copy protected, so you are encouraged to make backup copies for your own use and to prevent the loss of the program should the original become damaged or corrupted.

Bug Warranty: This program is also warranted for life to be bug-free. By this, if you should find any definite bugs or programming flaws in the original, unmodified version of this program, you are encouraged to report the bug. The author will promptly fix the bug and release an updated version of the program with credit for the bug discovery going to you. You, of course, will receive a copy of the new version free of charge. A bug is considered to be a flaw that causes a crash of your system or data corruption of a disk or the screen appearance. Incompatibilities with certain hardware features can not be considered a bug, although every effort has been taken to insure that this program will work with any hardware setup that can be run in the GEOS environment.

Wheels 64 and Wheels 128 are products and trademarks of Click Here Software Co.

GEOS 64 and GEOS 128 are products and trademarks of Berkeley Softworks
deskTop is a product of Berkeley Softworks

Commodore 64 and 128 are products of Commodore Business Machines
RamLink, HD Series and FD Series are products of Creative Micro Designs
RamDrive is a product of Creative Micro Designs and Performance
Peripherals

gateWay is a product of Creative Micro Designs

Table Of Contents

	An Introduction To geoSHELL	2
1.	Getting Started With geoSHELL	4
	Setting Up Your Boot Disk	
	Setting Up Your Workdisks	
	Entering Commands	
	Using Multiple Commands	
2.	Working With The Directory	9
3.	Load And Run Files	13
4.	Using The Keyboard	15
	CLR-HOME and SHIFTED/CLR-HOME Keys	
	RETURN, CMDR/RETURN, ENTER, and the UP-DOWN Cursor Keys	
	LEFT and RIGHT Cursor Keys	
	CONTROL, STOP, and NO SCROLL Keys	
	DELETE and INSERT Keys	
	The ALT Key	
	HELP Key	
	SHIFT/RUN Key Combination	
	The Numerical Keys	
	LEFT-ARROW Key	
	UP-ARROW and CMDR/UP-ARROW Keys	
5.	The Startup File	19
	What Is A Startup File?	
	Creating A Startup File	
	The 'exec' Command	
	The 'getkey' Command	
6.	Resident Commands	23
7.	Transient Commands	40
	What Is A Transient Command	
	Internal Transient Commands	
	External Transient Commands	
8.	The Programmable Function Keys	58
9.	The Built-In Hot Keys	60
	What Are The Hot Keys	
10.	Special Features For CMD Devices	63
	Special DOS Wedge Commands	
	The 'path' Command	
11.	The Commands At A Glance	66

An Introduction To geoSHELL

What is geoSHELL? geoSHELL is a user interface that is used exclusively with the GEOS operating system on a Commodore 64 or 128. Much like the DeskTop, geoSHELL can perform most of the functions that a GEOS user needs for managing disk files. Unlike the DeskTop, however, geoSHELL is almost entirely text-based, rather than using icons and pull-down menus. This does not mean that icons and menus cannot be used, however. Therefore, the initial appearance of geoSHELL leads one to believe that it is not user-friendly. However, once you become familiar with it, you will find yourself getting around in GEOS to be much faster and easier.

geoSHELL recognizes a number of commands that the user may type on the keyboard. As an example, simply type the command 'dir' and hit 'RETURN' if you wish to view a directory of the disk. The commands that geoSHELL recognizes are easy to remember and easy to type as there are no complicated series of commas, colons, parentheses, or slashes to remember. If a command requires a parameter to follow it, just remember to include one space between the command and its parameter.

geoSHELL loads faster than the DeskTop for various reasons. It doesn't have to load in the icons and display them whenever opening a disk or flipping to another directory page. In fact, geoSHELL doesn't even read a disk directory until it is told to do so. This allows you to insert a different disk into the drive at any time. With the DeskTop, you always have to open the disk whenever inserting a new one.

geoSHELL allows you to find a file on the disk faster than the DeskTop can because you simply type in the filename and hit RETURN and geoSHELL will load and run the file for you. If it is not found on the active drive, then any other drive that is on your system will be searched as well. With the DeskTop, you have to search through each page of files until you find the one you want and then double-click on its icon. This is OK if the desired icon is already displayed on the screen, but if you have to search your ot

her drive, it takes a few moments. Unfortunately, some of us computer users no longer have the patience we once had.

Now, don't get me wrong, the DeskTop is a very good piece of software. It has an abundance of features and is very easy to comprehend and use. It was well-written and has very few bad points. But the GEOS world has been without any form of interface that gets the user more on an even level with his machine. Now, there is geoSHELL. You now have added capabilities all built into one piece of software, that, coupled with your favorite icon-type interface (such as the DeskTop), makes working with GEOS much more enjoyable.

geoSHELL may also be referred to as a CLI, or 'command line interface'. But once you start using geoSHELL, you will find that it is much friendlier than most any

other CLI for any type of computer.

Read through the rest of this manual and become familiar with the many features and commands that geoSHELL possesses. Don't be intimidated by any part of this manual or by geoSHELL itself. You don't have to know everything there is to know about it in order to begin using it. The structure of the manual is such that you are taken a step at a time in learning how to use this program. You will find yourself becoming accustomed to it's use very quickly. Just think in terms of the DeskTop when you are issuing a command in geoSHELL, and you will understand what you wish to accomplish and what is taking place.

So, continue on and explore the world of GEOS through the use of geoSHELL.

Chapter 1

Getting Started With geoSHELL

Before you use geoSHELL for the first time, copy it to another disk. Refrain from using your original disk that geoSHELL was supplied on. From the DeskTop, copy the file called 'geoSHELL' to one of your workdisks. For now, you won't need any of the other files from the geoSHELL disk. Become familiar with using the program and then check out the rest of the disk. Once you've made a copy of geoSHELL, you may continue on.

Like any other application in GEOS, the first thing you need to do to run geoSHELL is to simply click on its icon from the DeskTop. Within a few moments, geoSHELL will be loaded and displayed on your screen. It's as simple as that. geoSHELL is now in command of your machine instead of the DeskTop. If you wish to return to the DeskTop, simply click on the 'close' button at the upper right corner of the geoSHELL window.

With geoSHELL running, you are now in a different type of user environment. geoSHELL is waiting, with its cursor blinking, for you to begin typing in commands. Just in front of the cursor is a letter representing the drive that is the currently active drive. Try entering the command 'dir', which is short for 'directory'. Be sure to hit RETURN after typing the three letters. What you will see now is a directory listing of all of the files that are contained on the disk. While the DeskTop displays the files a page at a time with eight files to a page, geoSHELL will scroll through the whole directory all at once. This is much like you would do it from Basic, using a dos wedge command.

Now, if you should see a particular application or desk-accessory that you would like to try loading, simply enter the name of the file and hit RETURN. When you are ready, exit your program by whatever means you normally would. For instance, from GeoWrite, click on 'quit' from the drop-down menu. When you exit the application, you are brought right back to geoSHELL instead of the DeskTop.

If you know the name of the file you would like to load, you don't even have to look for it in the directory. Just enter the name and geoSHELL will find it for you. If it is

not on the active drive, then geoSHELL will search the other drives for it and load and run it from the drive it is found on. This feature saves a lot of time that you would normally spend searching through the DeskTop pages for the program that you need. The only thing you must keep in mind if you are using more than two drives is that some applications do not work real well from drive C. Where they usually have a problem is when they are trying to access the drives themselves. GeoWrite has some quirks about being used from drive C. It expects to search drives A and B for it's data files. It will work, however, within certain limitations. Try various programs and get to know what works from drive C and what doesn't. At least, now you have that option. The DeskTop won't let you use drive C, not even for copying files. It makes you first swap drive C with either A or B. In fact, geoSHELL also has provisions for installing and using a fourth drive as drive D. You will find details of this in other parts of this manual.

Setting Up Your Bootdisk

Once you become familiar with geoSHELL, you may wish to copy it to your boot disk. This is the best way to use it since it will automatically run when you first boot up GEOS. This is not absolutely necessary since you can double-click on geoSHELL from the DeskTop at any time. And likewise, you can exit geoSHELL to return to the DeskTop, or whatever interface you loaded geoSHELL from, at any time.

There are some things to understand when copying geoSHELL to your bootdisk. First of all, geoSHELL is of a filetype known as AUTO-EXEC. An AUTO-EXEC file is nothing more than an application with one difference, it will execute automatically from your boot disk when you first boot up GEOS. There is another AUTO-EXEC file on your boot disk that is important, the CONFIGURE file. There might be other AUTO-EXEC files on your bootdisk also. The thing to understand about these files is that the order in which they are placed is important. They are executed in the order in which they are found on your bootdisk. But remember that this only occurs when you first boot up. Once geoSHELL is loaded, no more AUTO-EXEC files will be executed. So, for this reason, make sure that geoSHELL is the last AUTO-EXEC file in order on your boot disk. Generally, you will want CONFIGURE to be your first AUTO-EXEC file. This is because CONFIGURE is in charge of setting up your drives and ramdisks before the DeskTop first appears, or in this case, before geoSHELL first appears.

The way to identify the filetype of a file would be to either look at it's directory entry with geoSHELL, or you could use the 'info' command, or you could use the DeskTop's function for viewing the info for a file. This can be done by clicking on

'info' from the DeskTop's 'file' drop-down menu. Also, when viewing files by filetype on the DeskTop, you will see the order in which the files are placed for each different filetype.

Setting Up Your Workdisks

Setting up a workdisk to use with geoSHELL is no different than what you have been doing so far. All you have to do is copy geoSHELL to your disks and have it available for GEOS just like you would the DeskTop. But, if you wish to save some space on the bulk of your disks, you will find a file on your geoSHELL program disk called 'getshell'. It has an icon that looks just like the transient commands but is really a stand-alone AUTO-EXEC file. Yes this file will automatically run when GEOS first boots up. But, you would want it placed just in front of geoSHELL.

When 'getshell' is loaded and run, nothing happens on the screen but 'getshell' has the job of finding geoSHELL and getting it loaded for you. Now, this does not mean that getshell is needed in order to use geoSHELL, because you can do without it if you want. But once getshell has been loaded and run, it becomes the default file that GEOS looks for whenever you exit an application. If you should exit from drive C, GEOS will only look at drives C and D for the DeskTop, or in this case it will look for getshell. As long as getshell is on drive C or D it will be loaded. Then getshell will search through all of the drives for geoSHELL and get it for you.

Now you are wondering, why bother with putting getshell on a disk to get geoSHELL when you could just as easily put geoSHELL on each disk? Well the reason is because getshell only occupies 2k on the disk and geoSHELL is over 30k. You could copy getshell to every single GEOS disk and not use much space on any of them. Just have geoSHELL on one disk in any of your drives. The other advantage to this is that getshell has some very powerful routines contained within it that will search a 'path' and find geoSHELL even if it is on a partition of a CMD device that is not presently visible to the system. Chapter 10 has more information on this particular feature.

You only have to activate getshell once during your session. Either double-click on it's icon from the DeskTop or you can run it from geoSHELL also. It will then be activated in your system. The only way to turn it off would be to exit geoSHELL back to the DeskTop, and then double-click on the geoSHELL icon instead of the getshell icon.

When getshell is loaded by GEOS after exiting an application it will always look at the ramdisk first when it searches for geoSHELL. Then it looks at your 'path partition' if

Chapter 1 - Getting Started

one has been set. If it is not found there, then it will search the rest of the drives. If geoSHELL is still not found, you will be informed as such and told to insert a disk with geoSHELL.

Entering Commands

Once you have geoSHELL on the screen, you will need to know a few rules that you must follow in order to use its many features. Fortunately, geoSHELL is a very friendly user-interface and there is very little to remember. These rules apply mainly to how you enter a command or a series of commands.

Of course, to enter a command, you just type it in and hit the RETURN key. Once the RETURN key is struck, geoSHELL will analyze your input and perform the function that you desire. Some commands are simple ones that are entered all by themselves and some require you to add a parameter to them. Then there are some that work with or without a parameter. For instance, if you enter the command 'date', geoSHELL will respond by displaying the current date that is set in the GEOS kernel. If you enter 'date' along with a parameter that 'date' recognizes, then geoSHELL will set the date to the one that you specified in the parameter.

Whenever you enter a command followed by a parameter, you must put exactly just one space between the command and parameter. All the commands expect to find their parameters exactly one space after the command.

Here's an example:

```
run photo manager
```

It is obvious as to what will happen when you hit RETURN after entering the above line. After analyzing the line, geoSHELL will search the drives for the photo manager and if it finds it, it will load and run it for you. Now, if you had placed two spaces between 'run' and 'photo manager', geoSHELL would have looked for a file that begins with a space and would not have found the one you were looking for. One thing is good, you don't have to enclose the filename inside quotation marks like you do from Basic.

Another little trick that can save some typing when entering a 'resident' command is you are allowed to use a shorter version of the command. In fact, geoSHELL will recognize anything from the first three characters up to the entire spelling of the command. This means that you can use the command 'dir' or you can use the

Chapter 1 - Getting Started

command 'directory'. Also, 'direc' will perform the same function. This only applies to the commands that are resident in the computer. Any command that resides on the disk, the transient commands, must be typed in full.

Using Multiple Commands

geoSHELL allows you to enter more than one command on a line. Just make sure to put at least one space between each command. You could put more than one space between commands if you like, the extra spaces are ignored. It is between a command and it's parameter that you must remember to put just one space. You are allowed to enter as many commands on one line as will fit. geoSHELL will execute each command in the order that they are placed.

Now, when you type in a command that requires a parameter that is not necessarily of a fixed length such as in the case of a filename, geoSHELL needs a way to know where the end of the parameter is. A terminator is used to end any command that uses a variable length parameter. This terminator is the 'up-arrow' (^). You will find the key for it just to the left of the RESTORE key. Here is our previous example with a second command following it:

```
run photo manager^ date
```

In this case, the desk accessory called 'photo manager' will run and after returning to geoSHELL, you will be given the current date. If you failed to place the terminator in this example, geoSHELL would have looked for a file called 'photo manager da'. Since a filename can never exceed 16 characters, the 'te' would have been ignored and your whole line entered would be wasted. You wouldn't hurt anything by making this mistake, other than waiting while geoSHELL makes a quick search for a non-existent file. geoSHELL will respond with 'photo manager da Not Available'. geoSHELL is smart, but it can't read your mind entirely.

So, don't forget to use the up-arrow terminator at the end of any variable length parameter if you are following it with another command on the same line. If you are only entering one command on a line or if this command is the last one on a line, then the terminator is not needed. The end of your input is also a terminator.

These terminators are also used in your startup files and exec files. You will learn more about these files in other parts of this manual.

Chapter 2 Working With The Directory

The most important function of any type of user interface is working with the directory of the disks. geoSHELL is no exception. We are not always organized in the way that we place our files on our disks and sometimes our directories become quite long and files can easily get lost. geoSHELL incorporates several ways of viewing a directory, so that even a disorganized directory can be displayed in an orderly manner.

The 'dir' Command

The most basic directory command is simply, 'dir'. Just enter the command 'dir' and the entire directory from the disk in the drive will be displayed in order from the first file to the last. You will notice that the drive letter and disk name is displayed first, followed by the filenames, any border files that exist on the disk, and finally the amount of free space left on the disk. You know from using the DeskTop that the border files are files that you may have placed on the lower border of the screen and forgot to put back onto the pad before closing the disk. geoSHELL displays these for you so that you may have access to them. It is always a good idea to put them back into the main directory, though.

The 'ddate' and 'dtype' Commands

In addition to the filenames that are displayed in a directory listing, you will see some additional information on each file, when you display a directory. The type of file, in an abbreviated form is displayed along with the approximate size of the file, plus the date and time it was created or last modified. On the 80 column screen of the 128, there is room for all of this info to be displayed for each file. But on the 40 column screen of the 128, and on the 64, only the filename, the filetype and the filesize will be displayed. If you wish to see the date and time for the file you must enter the command 'ddate'. Now you will be able to view the date and time and not the filetype

Chapter 2 - The Directory

or size. If you wish to go back to viewing the filetype and size again, use the command 'dtype'. Both of these commands will serve no purpose on the 80 column screen of the 128. However, if you issue one of these, then it will be in effect when you switch to the 40 column screen.

The 'fastdir' and 'slowdir' Commands

Now, let's say that you have an extremely large amount of files on your disk, and you don't have to be bashful, because geoSHELL has no limit on the number of files that are allowed on a disk. The only limit is the disks' own directory. For instance, the 1581 disk drive is capable of having 296 directory entries, but the DeskTop will only access the first 144 of them. geoSHELL will access all of them. If a hard drive is capable of holding several thousand files in it's directory, geoSHELL can access every single one of them. The reason is because geoSHELL doesn't store the directory in memory. It reads it as it needs to. And it does it quickly. You won't see any pause when changing disks with geoSHELL. Now, with a large amount of files in your directory, you may want to scroll through the directory a little faster. You can do this by entering the command 'fastdir'. Now any time you view a directory, you will only see the filenames without the additional information. In many cases, this is all you will need. To switch back to viewing the information along with the filenames, use the command 'slowdir'.

Using 'dir' with Wildcards

The command 'dir' can be used in a couple of other ways also. 'dir' can call up a selective directory. Enter 'dir' followed by a partial filename containing one or more wildcards. Here's an example:

```
dir g*
```

This would display all the files that begin with the letter 'g'.

```
dir *g
```

This would display all the files that end with the letter 'g'. You can use one or two asterisks to represent wildcard characters. The asterisk takes the place of one or more characters.

If you enter 'dir m*t*e', you will display any file that starts with 'm', has a 't'

Chapter 2 - The Directory

somewhere in the middle and ends with an 'c'. Remember that the asterisk represents one or more characters, so that the 't' in this example would not be in the second position or the second from the last position of the filename.

The question mark is also used as a wildcard. It takes the place of exactly one character.

```
dir ha??
```

This will display any four-letter file that starts with 'ha' and ends with any other two characters. You can mix question marks and asterisks. Wildcards are very handy if you name your files with similar names, or if you want to find a file and can't remember exactly how it is spelled. If you know the first letter, just enter it followed by an asterisk. Any number of question marks are allowed, up to a maximum of 16, since a filename cannot exceed sixteen anyway.

Displaying by Filetype

There is one more use for the command 'dir'. If you enter 'dir' followed by a one or two digit number that represents a GEOS filetype, you will display a directory of all the files of that type. GEOS assigns a number to its various filetypes. For instance, desk accessories are assigned a filetype of '5'. So to list all of the DA's on the disk, just enter 'dir 5'. To help you to learn and remember all of the numbers that are assigned to the various filetypes, they are listed whenever the directory is displayed, next to the actual name that is given to that filetype. For instance, next to GEOWRITE 128 in a directory listing, you will see '6=Applic'. This means that GEOWRITE 128 is an application file and the filetype of 6 represents the 'application' filetype.

The 'cdir' Command

Let's assume you've been working on your computer for a little while and you want to see the files that you have created today. For this purpose, geoSHELL has the command 'cdir'. Just think of this command as representing the 'current directory'. Enter cdir and all of the files that are dated for the present day will be displayed.

Using 'cdir' With A Time Parameter

Now, 'cdir' is a good command for viewing files that you have created or modified

Chapter 2 - The Directory

during the day, if you are a normal person. But, most of us computer buffs are not necessarily normal. Many of us are up late at night, working past midnite on stuff. So, now you use the command 'cdir' to view the directory of files that you have been working on and it only displays files that you have done since midnite. Just add a parameter to 'cdir' that corresponds to a particular time of day and all the files from that point on will be displayed. Let's say, for instance, that you began your computing session at about 8:00 pm on a Thursday night. It is now past midnite.

```
cdir 0800p
```

This will display all of the files created since 8pm. geoSHELL will display from whatever time you enter within the last 24 hours if you use the command 'cdir' followed by a time parameter. You'll notice that you enter the time the same way as you would if you were entering the time to set the clock.

Using 'cdir' with a Date Parameter

If you would like to display all of the files from a specific date, 'cdir' can also use a date parameter.

```
cdir 041292
```

This will display all of the files created or modified from April 12, 1992 and later. This parameter is entered in the same format as if you were entering the date to set the clock.

From these examples, you can see how easy it is to locate a file no matter where it is in your directory. If you can remember just one thing about the file, whether it be part of it's filename or it's filetype, you can find it fast.

Chapter 3

Load And Run Files

Once you become familiar with the various ways of viewing directories, you will find that there is more than one way to load and run a file from geoSHELL. Of course, you can simply type the name of the file and geoSHELL will locate it and run it. Or you can save typing the filename if you find it in a directory listing. Now you can cursor back up to the filename in the directory listing and hit 'RETURN'. This will have the same effect as typing the filename directly. You could also click on the line containing the desired filename, which will place the cursor at the beginning of the line, and then hit 'RETURN'. Or, you could simply 'double-click' on the filename. Use whichever method you feel most comfortable with.

Once in awhile, you will need to load a file with the same name as one of the geoSHELL commands. The problem with this is that when you type the name of the file, the command will be executed instead. In this case, use the command 'run' followed by the filename. geoSHELL will then know to go and look for a file of that name and will not look for a corresponding command of the same name.

You can try to load any file that you may find on a disk. If it is not of a type that can be loaded and run, geoSHELL will inform you so, just as the DeskTop would. If you enter the name of a data file, geoSHELL will search for it's parent application and load it.

geoSHELL also accepts wildcards when you enter a filename. For instance, if you enter:

```
GEO*
```

This will cause geoSHELL to look for a file that starts with the three letters 'GEO'. If it finds one, and chances are it will in this case, it will load and run it for you.

geoSHELL will search all of the drives for a file when you enter a filename. The currently active drive, however, takes priority, and is searched first.

There are also other ways to load and run a file. In other parts of this manual, you will

Chapter 3 - Load and Run Files

learn how to define a hotkey command for loading a file. Chapter 9 covers the information you will need to know about the hotkeys. You can also define a function key to load and run a file. You will find more information about the function keys in Chapter 8.

Chapter 4 Using The Keyboard

It is easy to adapt to geoSHELL for various reasons. One reason is the use of the keyboard. Before GEOS came along, we were all accustomed to using the keyboard for most of our activities. geoSHELL is more keyboard oriented than most other programs in GEOS.

When typing, you are always in insert mode. Any characters to the right of the current cursor position are moved to the right until they meet the right side of the window.

The CLR-HOME and SHIFTED/CLR-HOME Keys

From Basic, we clear the screen with the SHIFTED/CLR-HOME combination. This does the same thing with geoSHELL. Try it and you'll see the geoSHELL window clear itself and leave the cursor blinking in the upper left corner with the active drive listed. The CLR-HOME key by itself moves the cursor to the upper left of the window without clearing it.

The RETURN, CMDR/RETURN, ENTER, and UP-DOWN Cursor Keys

The cursor keys are a little different. The UP-DOWN cursor keys move the cursor up or down one line at a time. The cursor gets placed at the end of each line that it comes to. This allows you to hit RETURN and repeat previous commands that you typed in or to load and run programs from a directory listing. The RETURN key is important. Hitting RETURN causes geoSHELL to look at the line that it was on and to interpret what it sees. Remember this! Don't use RETURN to move down to the next line unless you want the commands on the line to execute. You can use the UP-DOWN cursor keys to move up or down or you can also use the CMDR/RETURN combination to ignore the line and move to the next line down. This will cause the next line to erase clean and be ready for new commands. The UP-DOWN keys will not erase the line. The ENTER key on the 128 serves the same function as the RETURN key does.

The LEFT and RIGHT Cursor Keys

The LEFT and RIGHT cursor keys work the command line memory that geoSHELL has. geoSHELL remembers the last ten command lines that were entered. Maybe you misspelled a command or a filename. Just hit CURSOR-LEFT to redisplay the line and then correct your typing. CURSOR-LEFT moves backward through memory and CURSOR-RIGHT moves forward. This can save a lot of typing when you use similar command sequences over and over, or when you just wish to repeat a previously entered command.

The CONTROL, STOP, and NO SCROLL Keys

Whenever geoSHELL is scrolling through something, such as a directory listing or a text file, you may use the CONTROL key to pause the listing as long as you keep holding the key down. On the 128, you can press the NO SCROLL key to pause the listing until another keypress is made with any key. The STOP key will stop the listing and exit the entire command line execution. The STOP key will also exit a command line execution between multiple commands if pressed.

The DELETE Key

DELETE works as you would expect. DELETE will delete the character to the left of the cursor and bring the characters to the right along with it, unless the cursor is already to the left side of the window. The cursor will be left blinking on the spot where the character was deleted from.

The ALT Key

The ALT key is found only on the 128 and performs the function of switching back and forth between the 80 and 40 column modes.

The HELP Key

The HELP key on the 128 will activate the transient command 'help' if it exists on any

Chapter 4 - The Keyboard

one of the drives, just as if you typed in the command 'help'.

The SHIFT/RUN Key Combination

If you've ever used a fast-load type cartridge or even JiffyDos, then you have probably used the SHIFT/RUN combination to load and run the first file on the disk. This also works with geoSHELL. If the first file is not of a type that can be loaded and run, then geoSHELL will keep looking until it comes to one that can.

The Numerical Keys

You can use the number keys to make geoSHELL use different pages of your startup file whenever geoSHELL is loading. For instance, hold down the 5 key to execute the startup sequence on page 5 of your startup file. Using the zero key causes the startup file to be ignored. The startup sequence begins just after the copyright notice appears. So have your finger on the desired key before this appears and hold it until you see the screen freeze, then release it. If you do not touch any of the keys, the first page of your startup file will execute.

The Left-Arrow Key

Using the left-arrow key will move the cursor to the start of the line that it is on. This allows you to begin typing commands in front of whatever text is already on the line. This accomplishes the same function as if you clicked once on a line. Don't confuse the left-arrow key with the left-cursor key. The left-arrow key is at the upper left of your keyboard.

The UP-ARROW and CMDR/UP-ARROW Keys

The UP-ARROW and the CMDR/UP-ARROW have special meanings in geoSHELL. Multiple commands may be entered on a line. When geoSHELL is finished executing a command, it will look to see if there is another command on the same line. If so, it will be handled also. Some commands need a terminator after their parameter, so that geoSHELL knows where the next command is.

If you were to type in a command that requires a filename for a parameter and you

Chapter 4 - The Keyboard

want to type in another command after this one, then you need to use the UP-ARROW to terminate the filename. The reason for this is that geoSHELL has no idea as to how long your filename is. It can be from one character to sixteen characters long. If the command with a filename is the last command or the only command on the line, don't worry about it, because geoSHELL will see the end of the line. But if you follow with another command, geoSHELL will come to the UP-ARROW and know that what follows is another command and not the rest of your filename. The UP-ARROW is also used to terminate the 'echo' command's parameter.

The CMDR/UP-ARROW is only used when you are defining a function key. You would use it to terminate the function key definition. You do not use the UP-ARROW as a terminator here because it can be used as a terminator within the function key definition at the end of a filename. If the function key definition is the last command on the line, then the terminator is not needed. By the same token, if a command that uses a filename is the last command in a function key definition, the UP-ARROW terminator is not required.

Chapter 5

The Startup File

First of all, the use of a startup file is optional. geoSHELL can operate totally without it. However, it may be to your benefit to use one on each of your workdisks. If you like, until you become thoroughly familiar with geoSHELL, you can keep your startup files simple.

What is a Startup File?

A startup file is a GeoWrite file that you create with GeoWrite, of course. It is appropriately named 'Startup' on your disk (notice the capital 'S'). You can create different startup files for different disks, depending on your particular needs. A startup file contains the startup sequence that is a maximum of one GeoWrite page in length. Any one of pages two through nine may also be accessed instead of page one. This will be described later.

The startup file allows you to specify most any allowable geoSHELL command and to have those commands execute whenever geoSHELL is activated such as upon exiting an application and returning to geoSHELL.

One of the many uses of the startup file would be to define your function keys. You would do this if the default function key definitions that geoSHELL uses are not suited for your purpose. If you put geoSHELL on your boot disk, then you would want a startup file on your boot disk, also. In it you would include the commands to install your input driver and your printer driver. You might include various copy commands for setting up your ramdisk. You can use the drive swapping commands for setting up your drive letter (device number) designations before you begin your work.

Creating a Startup File

When you create your startup file, be sure to name it 'Startup', or geoSHELL will never find it. When you write your startup file, you are free to do it any way to like. You can use any font you choose. If you wish to use tabs for positioning commands or for indenting certain commands for clarity, feel free to do so. You can use as many

spaces in between each command, but you must use at least one space. For commands that require parameters, remember to only use one space between the command and it's parameter. The same rules apply here as for the immediate mode.

There is a limit to how large your startup file can be. First of all, it must be contained on one page. You are limited to about 1500 characters depending on how you format your file. Tabs and font changes will add to the size and take away from the number of commands that you can put into the file. But, in most cases, you will find that you won't exceed the limit. If you do, geoSHELL will ignore the startup file and prompt you with a message that the startup file is too large.

geoSHELL allows additional startup sequences to be on pages two through nine of your startup file. Page one of your startup file will always be executed unless you are holding down one of the number keys, 2 through 9 when geoSHELL is activated. Whatever key you hold down is the page that geoSHELL will look at. If you would like the startup file to be ignored altogether, hold down the zero key. Do this if there are commands on the startup file that you do not wish to execute. The exact point when geoSHELL is ready to load in the startup file is after the screen has been drawn and immediately after the copyright message appears. Have the desired key held down before this or geoSHELL won't catch it if you are too late. If you are too late, it will ignore the key you are holding and go right ahead and execute the first page of the startup file.

Once in awhile, you might find yourself creating a fairly complex startup file. There is a way to place comments within a startup file. Anything enclosed inside a left and a right curly brace will be ignored by geoSHELL's command processor. This allows you to include little notes to remind you as to what your startup file is supposed to do. To produce the curly braces, hold down the CMDR key and press the colon key for the left curly brace and the semi-colon key for the right one.

The 'exec' Command

Exec performs essentially the same thing as the 'startup' command would do with a few minor differences. It will load in a GeoWrite file and execute any commands that it finds on page one. All other pages are ignored.

In reality, 'exec' mainly gets the file loaded into memory and then hands the job over to the 'startup' command to finish the operation.

Another difference with 'exec' is that an exec file can have any name. A startup file

Chapter 5 - Startup File

must be named 'Startup'. You can set up an exec file to perform routine tasks that you might do on a regular basis. This can save on a lot of repetitious work. Also, where 'startup' only looks at the current drive for a startup file, 'exec' will look at all of the drives for the requested exec file, even a 'path' partition if one has been set.

Just like with a startup file, if you issue an exec command from within a startup file or another exec file, the new exec file will take over control and any command that followed the exec command in the file will be ignored. But, with this power, you could chain a series of exec files together. Just end each file with an exec command that calls another file in. There is no limit to how many can be chained together. This might be useful for a demo or text presentation of some sort. From within an exec file, you could call the 'type' command to display a text file for someone to read. As time goes on, you will find many uses for the exec command. Here's an example:

```
exec filename
```

The parameter required is the filename of the file you wish to load. In this example, the requested file called 'filename' will be loaded and executed, provided it is in a GeoWrite format.

The 'getkey' Command

Supplied on your geoSHELL program disk is a transient command called 'getkey'. This command was written to give the user a means for creating his own menus within the geoSHELL window. We're not talking about the drop-down menu that you normally see at the top of a geoSHELL program. This would be a menu that appears within the geoSHELL window. The command is intended to be used within a startup or exec file. When geoSHELL sees this command, it will stop right there and wait for the user to press a key. At this point, it is similar to the 'wait' command. This allows a startup file or exec file to jump to a specific spot should the user press a specific key. Here is a sample startup file:

```
clr
echo Load GeoPublish? Y/N^
getkey
{n end }
{y run GEOPUBLISH^ }
```

Study the commands in this startup file and you will get an idea as to what it does. But what about those commands that are inside the curly braces? Normally geoSHELL

Chapter 5 - Startup File

will ignore anything inside a curly brace, and allow you to have comments placed there. But 'getkey' uses these for its own purpose. It will look at the left curly-brace and see if the character that follows it is the same as the key that the user pressed. If it finds a match, then any commands that follow it will be passed on to geoSHELL for processing.

So with this in mind, the first command in our sample startup file is 'clr'. This will clear the geoSHELL window to make ready for any new text that we might want placed there. The next command is 'echo'. It will print to the screen the message that follows it. After that, the command 'getkey' is invoked. Now, the computer seemingly freezes. It is waiting for you to press a key. Once a key is pressed, a search will begin. The remainder of the startup file is searched for a left curly brace. If one is found, the character after it will be checked to see if it matches the keypress. In our case here, if the 'n' key was pressed, it will be found, and the next command after it is 'end'. This will cause the startup or exec file to immediately cease and you will be returned to the immediate mode. Now, if the 'y' key was pressed, it would have been found and the following command tells geoSHELL to 'run GEOPUBLISH'. Since GEOPUBLISH is an application, it will be loaded and the startup file will end at that point. If it was a command that did not cause an end to the startup file, then execution of the startup file would keep on going from that point. The right curly brace will be ignored.

Once a keypress match is found, 'getkey' is finished with its job and returns control to the command processor. If a match had not been found, then 'getkey' would still be waiting for a keypress.

Some rules must be followed when putting 'getkey' in a startup file. The letter or number that will be put next to the left curly brace must not have a space between it and the curly brace. However, a space must follow this character. Plus, there must be at least one space to the left of the right curly brace and the last command inside the curly braces.

You can see where this one single command can provide us with a very flexible means of receiving input from the user.

Chapter 6 Resident Commands

Contained within geoSHELL when it is loaded into the computer are a number of commands. They are known as 'resident' commands since they reside in memory. Commands that must be loaded from disk are known as 'transient' commands. Refer to Chapter 7 for more information on the transient commands.

Resident commands are available any time geoSHELL is loaded. When you type the name of a resident command, it will be immediately executed. The resident commands also have the advantage over transient commands of allowing you to type in an abbreviated form of the command. Anything from the first three characters up to and including the full name of the command is accepted. For instance, 'del' will perform the same as 'delete'.

Some commands require a parameter to be specified. Whenever that parameter might be of a variable length, you must end it with an 'up-arrow' terminator unless it is the last command on a line. In that case, the terminator is not necessary.

Here is a listing of each command along with a description, arranged in alphabetical order. Examples are given for commands that require a parameter. Some of the commands, such as 'directory', might already be described elsewhere in this manual. In this case, you will be referred to that portion of the manual.

a:, b:, c:, or d:

These four commands have the simple job of selecting a drive to be the currently active drive. As soon as you enter a letter a, b, c, or d, followed by a colon, you are telling geoSHELL to open that drive. If the desired drive is online and properly installed, you will be presented with the corresponding letter at the left side of the geoSHELL window followed by a right angle bracket and the familiar blinking cursor. If, for some reason, geoSHELL is unable to open that drive, you will be told that it is unavailable.

Chapter 6 - Resident Commands

You can use this command, followed by another command on the same line to perform the job of opening a different drive and perform a command on that drive. You can even end this line with a command to reopen your original drive to return to where you were.

always

This command is only used in a startup file or an exec file. It has a simple purpose. When geoSHELL finds this command in a startup or exec file, any commands that follow it will be executed. This essentially cancels the effect of the 'onboot' command. Refer to the 'onboot' command for more information.

No parameters are required for this command.

basic

This command does exactly as expected. It will take you out of GEOS and return you to the Basic mode of your computer.

On the 64, you can perform various functions such as running a Basic program and then return to GEOS by pressing the RESTORE key. On the 128, GEOS does not always do a clean exit to Basic. Sometimes, by going into the MONITOR and then typing: G 1C000 will get you back into GEOS. Sometimes it doesn't work. 128 GEOS leaves some vectors in a state that does not allow certain functions to operate properly, unfortunately. In some cases, it is necessary to reboot.

No parameter is required for this command.

boot

This command works only on a 128. If the current drive contains a valid boot disk, you will leave the GEOS environment and the disk will be booted. If the disk is not bootable, you will be informed as such and nothing will happen.

catalog

Chapter 6 - Resident Commands

For an explanation of this command, refer to the 'directory' command in chapter 2. This command performs exactly the same as 'directory'. The reason for it's existence is that some people have become accustomed to using catalog instead of directory. For the same reason, you may also use the dollar sign, '\$', in place of 'directory' or 'catalog'. Don't confuse the dollar sign with the DOS wedge command '@\$'. When used following an '@', it performs a directory listing that is slightly different. Refer to the DOS wedge commands for more information.

cdir

You will want to familiarize yourself with this command, as you will find many uses for it.

The job of 'cdir' is to display a 'current directory'. There are also three types of current directories. You can display the files that were created during the current day by using the command without any parameters. Or, if you supply a time for a parameter, you will be given a directory of files that were created since that time during the previous 24 hours. With a date parameter, you will get a directory of files that are dated from that point and later.

Refer to chapter 2 for more information about this command.

clr

This command is intended for use in a startup or exec file. When geoSHELL sees this command, it will clear the geoSHELL window and put the cursor at the upper left of the window.

You might want to use this when creating custom menus or before using the 'echo' command.

No parameter is required with this command.

date

This command by itself will display the current date that is set within the GEOS kernel. It will also allow you to change the date if you follow it with a valid parameter. The parameter must be in the format of 'mmddy'. That is two digits to

Chapter 6 - Resident Commands

represent the month, followed by two digits to represent the day of the month and two digits for the year. Here's an example:

```
date 021292
```

This will set the date to February 12, 1992. Notice that the month of February was represented by '02'. You must supply a leading zero even when a month or a day would equal less than 10. geoSHELL expects to see a total of six digits for the date parameter. Since this is a fixed length parameter, you are not required to use a terminator after the parameter when following it with another command on the same line.

ddate

Don't confuse this command with the 'date' command. This command affects the way directories are displayed on a 40 column screen. It stands for 'display date'. So therefore, whenever you are viewing a directory, you will see the filenames followed by the date and time that the files were created or last modified. The command 'dtype' performs the opposite of this command.

No parameter is required for this command.

delete

This command will delete a file from the currently active drive. It will not search through the other drives for the file. This prevents the possibility of accidentally deleting a file that you did not intend to delete. You must supply a filename for a parameter. If the file is found on the disk, it will be promptly eliminated from the disk.

```
delete filename
```

This example will delete a file named 'filename'. If you wish you could supply a parameter containing wildcards.

```
delete file*
```

This would look for the first file that begins with 'file' and delete it, if found. Only the first occurrence of the filename will be deleted. This does not allow you delete a series

Chapter 6 - Resident Commands

of files that all match the wildcard selection. It was felt that there might be times when this would be undesirable, so this feature was left out.

directory

This command is used to display a directory of the disk that is in the current drive. For a more thorough explanation of it's use, refer to chapter 2.

dtype

Don't confuse this command with the 'type' command. This command affects the way directories are displayed on a 40 column screen. It stands for 'display type'. So therefore, whenever you are viewing a directory, you will see the filenames followed by the filetype and size of each file. The command 'ddate' performs the opposite of this command.

No parameter is required for this command.

echo

This command is used primarily in startup files and exec files. It's job is to display a message or string of characters to the screen. All you have to do is supply the parameter and end it with an up-arrow terminator. There is no limit to the length of the parameter either. 'echo' will just keep echoing to the screen until it reaches the terminator. Word-wrap is performed at the end of each screen line also. Your only limit on the length is the size of a startup or exec file that is allowed. Here's an example:

```
echo Continue? y/n^
```

This would display the string 'Continue? y/n' to the screen. You might want to refer to the 'getkey' command or the 'wait' command for ways to put the 'echo' command to effective use in creating a menu.

```
echo ^
```

Using echo with just a terminator will cause a carriage return to be performed. Use this to separate two different lines when displaying a message to the screen.

end

This command is used in a startup file or exec file. When geoSHELL sees this command, it will immediately cease the executing of the startup or exec file and return you to the immediate mode. Normally this command is not needed, since geoSHELL will end the execution when it reaches the end of a startup or exec file.

So why have this command? Well if you refer to the 'getkey' command, you will find that it can be used to terminate a portion of a startup or exec file when 'getkey' is jumping around looking for a string of commands to execute.

No parameter is required for this command.

exit

This command will exit geoSHELL and return you to the DeskTop. If you loaded geoSHELL from an alternative DeskTop then you will exit to that one. This command performs the same as if you were to click on the close button at the upper right of the geoSHELL window.

Normally a parameter is not required with this command. But you might find a time when you wish to exit to a different interface than the one which you entered geoSHELL from. Yes, you can boot up from the DeskTop and then exit to the gateWay. All you have to do is supply a filename after the 'exit' command, like this:

```
exit gateWay
```

This will exit geoSHELL and make 'gateWay' the new default that GEOS will look for when you exit an application. In order to return to geoSHELL, you will double-click on the geoSHELL icon from the DeskTop or whatever other interface you might be using. If you are using the 'getshell' application with geoSHELL, then you would want to double-click on that one instead in order to keep it active.

Be careful when supplying a parameter with this command. geoSHELL has no idea as to what kind of program you are specifying. You could enter 'exit GEOWRITE', and sure enough, GEOWRITE would then appear on the screen. But when you try to quit GEOWRITE from the drop-down menu, guess what happens? You will return right back to GEOWRITE. You need to exit to a program that is capable of loading and

Chapter 6 - Resident Commands

running other programs. In the case of exiting to GEOWRITE, you would be stuck and have to shut down and reboot.

fastdir

This command causes a faster directory display to occur since it will tell geoSHELL to only display the filenames and no other information about the files. This command does not display a directory by itself, it only sets a flag telling geoSHELL how to display the directories from this point on. You can cancel this effect by using the 'slowdir' command. If you wish to make this setting your default, use the 'custom' command.

No parameter is required for this command.

geos

This command will affect the disk that is in the current drive by making it into a GEOS disk. If the disk is already a GEOS disk, then nothing will happen. But if it is not a GEOS disk, it will have a few changes made to it that occur to GEOS disks. One of the changes involves delegating a sector on the disk for use in storing border files entries. This sector is a little directory by itself. It can hold up to eight directory entries.

geoSHELL does not require any disk to be a GEOS disk, but this command was implemented for compatibility with the DeskTop.

No parameter is required with this command.

getkey

Here is a handy command that is used in startup or exec files. It will wait for the user to press a key on the keyboard. The main intention of this command is to allow the user to set up a menu system with geoSHELL. Refer to Chapter 5 for more info on this command.

input

Chapter 6 - Resident Commands

This command will display the filename of your current input driver. The input driver is a little machine language routine that is stored in the GEOS kernel that controls the function of your mouse or your joystick. This command is used to remind you as to what input driver you are currently using. In most cases you would not need to use this command unless you frequently change input drivers.

You might want to refer to the 'install' command for information on how to change input drivers.

No parameter is required with this command.

insert

Here is a handy command for use in a function key definition. This command will save you a lot of repetitious typing of the same string of commands in front of other commands or filenames. First, here's an example on how to set it up:

```
key7 insert fcopy b ^
```

What we are doing here is defining function key 7. When the insert command is encountered, geoSHELL will insert the text that follows it at the current cursor position. Any text following the current cursor position will be moved to the right. In the case above, the command 'fcopy b ' including a trailing space will be inserted at the cursor position. All text up to the up-arrow terminator is used. Check out it's use in this example:

```
filename
```

Here is a filename in a directory listing. Go to this line by either clicking once on it to place the cursor at the start of the line, or by using the cursor keys to move to the line and then hit the 'left-arrow' key at the upper left of your keyboard. This will also place the cursor at the start of the line. Now hit F7.

```
fcopy b filename
```

This is what you will get after hitting F7. 'fcopy b ' is inserted in front of the desired filename. Now you can merely hit the RETURN key to execute this command line.

There are many other uses for 'insert' in addition to this example. As time goes on,

Chapter 6 - Resident Commands

you will discover your own ideas for it's use.

key

This is a simple command that will display your current function key definitions for you. If you change your definitions from time to time, you might not always remember how they are set. This command is used to remind you.

No parameter is required with this command.

key1, key2, key3,, key8

These eight commands allow you to redefine your function keys. They of course require a parameter in order to do so. If you do not specify a parameter, the corresponding key will do nothing when pressed.

You can define a function key to perform your most used commands, and will find that using them becomes second nature. You will use them all of the time, and it is to your advantage to do so.

For more information on redefining your function keys, refer to Chapter 8.

load

There will be times when you may want to use a transient command that is not on any of the disks that are presently sitting in your drives. You could, however, just insert the disk with the command on it and then type in the command that you want. But, maybe you don't want to execute that command while that particular disk is in the drive. For this reason, we have the 'load' command. Enter 'load' followed by the name of the command you wish to load and geoSHELL will load it into memory for you. Now the next time you type the command that you just loaded, it will be executed. Here's an example:

load type

In this case, maybe you didn't have geoSHELL anywhere on a disk that is in your drives. The 'type' command is an internal transient command that is contained within the geoSHELL file. 'load' will fetch it into memory for you. Now, you would remove

Chapter 6 - Resident Commands

the disk and reinsert your previous one and then proceed to view any text files that you wanted to view with the 'type' command.

The load command only works with geoSHELL commands and not with any application or desk accessory. As long as the desired command is somewhere on one of the drives that are on your system, you won't have to use this command.

mono

This command is only for the 128 while in 80 column mode. It switches the 128 into a monochrome mode which is the standard GEOS mode. You would use this command after using the 'rgb' command. The colors that are used in monochrome mode are established with the preference manager.

This command requires no parameter.

onboot

This command would be used in a startup file. It's purpose is to separate certain commands in the startup file that resides on your boot disk. Once geoSHELL sees the 'onboot' command in a startup file, it will execute any commands that follow it only during the initial booting of GEOS. Once geoSHELL or the DeskTop has loaded for the first time, the onboot command will inform geoSHELL to ignore any commands that follow it, should that same startup file be executed at any other time during your session. The importance of this command is that there may be a sequence of commands that you only want executed when you first bootup. There is no reason to reinstall an input driver once it has been done for the first time, for instance. Or you might have a series of copy commands executed when you first bootup, in order to fill your ramdisk with the files you are going to work on. Why copy them again?

In order to cancel the effect of the 'onboot' command, you would use the command 'always'. Any commands in between these two commands will only be executed during the initial bootup. Any command after 'always' will be executed at anytime.

You may have as many 'onboot' and 'always' commands as you wish in a startup file.

There is no parameter required with this command.

output

This command will display the filename of your current printer driver. This command is used to remind you as to what printer driver you are currently using. In most cases you would not need to use this command unless you frequently change printer drivers.

You might want to refer to the 'install' command for information on how to change printer drivers.

No parameter is required with this command.

parent

This command works when using native partitions on CMD devices, such as the HD or RamLink. From BASIC, you might already know that you can use '@cd' followed by a left-arrow to move to the subdirectory that is the parent of the one you are currently in. And with geoSHELL, we can send commands to the drive, but we cannot produce the left-arrow character. So, this command was created to replace this sequence of keys. By using 'parent', you can return to whatever directory you passed through to get to the current subdirectory you are in.

This will move up one directory:

Example: parent

This will move up two directories:

Example: parent parent

print

This command will perform the same function as if you clicked on a data file's icon and dragged it down to the printer icon and dropped it while using the DeskTop. The desired data file along with its parent application would be loaded and then the application would proceed to print the file. When finished, it will exit back to geoSHELL. Here's an example:

```
print mydatafile
```

This will print a file called 'mydatafile' and then return to geoSHELL when finished.

Chapter 6 - Resident Commands

Your current printer driver will be used for this purpose.

rename

This command allows you to rename a file. All you have to do is supply the new filename that you desire and the current name of the file that you wish to rename. This will only work on the currently active drive. Here's an example:

```
rename newfile=oldfile
```

This will look for 'oldfile' on the drive and if it is found, it will be renamed to 'newfile'. Notice that you must use an '=' sign in between each filename. Use only the '=' sign with no extra spaces, unless the spaces are contained within either of the filenames.

rgb

This command only works with the 128 while in 80 column mode. You need a color monitor for this command to look appropriate on the screen. It will switch the 128 into a nice looking multi-color mode. If you wish, you can use the command 'color80' to alter the colors to suit your own taste. The command 'custom' will also save your new settings if you'd like.

No parameters are required for this command.

root

This command works when using native partitions on CMD devices, such as the HD or RamLink. The 'root' command will take you to the root directory of the current partition you are in. If you are in any subdirectory, you can use this command to go straight to the root directory.

run

This command will load and run a file, if the desired one can be found on one of the drives on your system. In most cases, this command is not needed. You can avoid

Chapter 6 - Resident Commands

using this command just by typing the name of the file by itself. Where this command might be needed is when you wish to run a file that also has the same name as a geoSHELL command. If you were to type the name of a command, the command would take first priority and be executed instead of the file you really wanted to run. For this reason you would use 'run' followed by the filename. Here's an example:

```
run filename
```

This will cause geoSHELL to ignore any commands that are resident in memory and instead to look for a file called 'filename' on one of the drives.

s40 and s80

These two commands will make geoSHELL switch a 128 to the desired screenmode, 40 column or 80 column. It's as simple as that. Of course, you can also use the 'switch' command, but if this is being done from a startup or exec file, you may not always know which mode the computer is going to be in. 'switch' is a toggle function, while s80 and s40 will definitely set a specific mode. From the immediate mode, the easiest way to switch screens is by using the 'ALT' key.

There are no parameters required for these commands.

slowdir

This command will tell geoSHELL to display other information along with the filenames, whenever you are issuing a directory command. You would use this if you need to view information besides just the filenames. The info that can be displayed is the filetype, the filesize, and the date and time that each file was created or last modified.

There is no parameter required for this command.

startup

This command by itself will execute the startup file that is residing on the current drive, if there is one. If you supply a number from 2-9, it will execute the corresponding page from the startup file. The startup file must be named 'Startup' with a capital 'S'.

Refer to Chapter 5 for more info on using startup files and exec files.

status

This command will display the current status of your drives. It will show you the drives that are currently online and the disks that are in them. The type of drive is also displayed. This command is handy since it is easy to forget which drive is which if you do any swapping around.

There is no parameter required for this command.

switch

This command is a toggle command. It will switch the 128 back and forth between 40 and 80 column modes. You might only need to use this in a startup file or an exec file since it is easier to just hit the 'ALT' key from the immediate mode. You could also include this command in a function key definition along with the filename of a program that needs to run in a different screenmode. This command is provided for that convenience.

This command requires no parameter.

time

Here is a command that you want to make sure that you use, along with the 'date' command whenever you bootup. This command will set the time in the GEOS clock. By itself, it will display the current setting of the clock. Here is an example of how to set the clock:

```
time 0623p
```

This would set the time to 6:23 PM. geoSHELL expects to see exactly 4 digits plus either an 'a' or a 'p' in the parameter. Notice that the '06' represents 6 o'clock and the '23' represents 23 minutes after. Then the 'p' stands for PM. Of course, an 'a' would stand for AM. You must supply a valid time parameter or geoSHELL will just display the current time and then try to search for a file with a name of the invalid parameter that you typed. For this reason, you can enter the command 'time' followed by any

other command and there will be no errors.

wait

This command is used in a startup file or an exec file to cause the computer to pause until the user types any key on the keyboard. It can be used, therefore, to cause a break in the action during the startup file. You might have a place during the startup or exec file where you need to change disks in one of the drives. Just use the 'wait' command at that point in the file. Then once you type any key, the startup file will continue on and process the remainder of the commands in the file.

There is no parameter required for this command.

The DOS Wedge

Built right into geoSHELL are some routines that provide the user with a DOS wedge. To some degree, this is geoSHELL's own interpretation of a DOS wedge. There are some similarities between this one and others and there are also some nice enhancements that you won't find in any other DOS wedge for the Commodore, much less while being in the GEOS environment.

For those of you who are not familiar with a wedge, this is a way for you to send a command directly to the disk drive's own command channel. It allows a certain degree of communication with the little built-in computer contained within the disk drive. The way to invoke a DOS wedge command is with the '@' key. All DOS wedge commands used in geoSHELL begin with this character.

For most purposes you won't need to use a DOS wedge. However, there are some advantages in doing so, especially with the CMD devices. If you were to type the '@' key by itself and hit RETURN, you will be presented with a message from the drive that looks something like this:

```
GO,OK,0,0
```

This tells you that the drive is not currently detecting any errors. If you follow the '@' key with any other characters, these characters will be sent to the drive and the drive will then interpret them and execute them, if they form a valid command that

Chapter 6 - Resident Commands

the drive recognizes. Here's another example:

```
@S
```

This will tell the drive to start sending a directory listing. If it does, geosHELL will then display it to the screen.

If you have ever used a DOS wedge from Basic, you have always sent commands without using the SHIFT key. The drive expects the commands to be lower case letters, unless you are including a filename of some sort within the command. But the commands themselves are always lower case. But we have a problem with this while working in GEOS. GEOS is taking our input from the keyboard and converting it to ASCII characters instead of the PETASCII characters that the disk drive expects. The difference is that a lower case GEOS character is a foreign language to a Commodore drive. However an uppercase GEOS character is the same as a lower case Commodore character. What this means is that when you use the DOS wedge, you have to remember to press the SHIFT key and enter an uppercase GEOS character as in the following example:

```
@S0:filename
```

The 'S' in this example stands for 'scratch'. You are telling the drive to scratch a file called 'filename'. If you typed a lower case 's' instead, this would actually have been sent as a graphic character that the drive does not recognize and you will have gotten a 'SYNTAX ERROR'.

Now the beauty of the DOS wedge that is built into geosHELL is due to a few very special commands. One is '@P'. This command is intercepted and not really sent to the drive. It will cause your screen output to now also go to the printer. This is a toggle command. The next time you enter it, output will go only to the screen. You can use the 'pconf' command to set various settings so that it will work properly with your printer. Refer to Chapter 7 for more info on the 'pconf' command.

Another very handy DOS wedge command is '@CP'. This command by itself will tell you the current partition that is being used on a CMD device. If you wish, you can include a partition number such as in this example:

```
@CP3
```

This will now open partition number three, if it is of the same type as the current

Chapter 6 - Resident Commands

partition. If not, you will be told so.

Another DOS wedge command is '@\$=P'. This will display a directory of partitions. The beauty of this command is that you may then cursor up to the partition that you wish to open and hit RETURN or just double-click on the desired one and it will be opened. If you refer to your CMD manual, you will also see how you can use wildcards when viewing a partition directory.

Most all of the commands that are in your CMD manual will work with geoSHELL's DOS wedge. Experiment with the ones you might use most.

You will find many uses for this DOS wedge once you try it. But don't be intimidated by it if you don't understand it. You can get along just fine without it. It has been provided for those who have been hungering for it and those people will greatly appreciate it.

Chapter 7

Transient Commands

What is a Transient Command?

There are a number of commands supplied with the geoSHELL disk that reside either within the geoSHELL file or are contained within their own file. These commands are known as 'transient' commands. The transients that are contained within the geoSHELL file are considered 'internal' transients and the ones that are contained within their own file are considered 'external' transients. As far as you, the user, is concerned, both types may be treated the same as far as any rules governing their use is concerned. They may be executed just like any resident command in geoSHELL with the exception that they are not stored in memory for immediate access. They must be loaded from the disk. However, once a transient command is loaded, it will stay resident in memory until you perform an action that requires use of that same memory area where a transient command is loaded. Certain resident commands need to also use this memory area for a work area, and so they will erase any transient command that might be loaded. This is not really a problem, other than the next time you need the transient command, it will have to load from disk again. Fortunately, load times are very quick.

There are actually three transient command areas in memory that are designated for this use. However, most commands will generally use the main area, known as Trans1Comm. You need not be concerned with this aspect except for the few commands that will use Trans2Comm and Trans3Comm. The latter two areas are actually in the same buffer as that which is used by the startup and exec commands. This is not a problem except that when a startup file or exec file is loaded, it will erase a command from this area and will have to reload from disk the next time you use the command. Likewise, a transient command that uses this area will only work as the last command in a startup file or exec file. Once again, don't worry about this, since there are very few commands that will need to load into this area. Any that do must follow certain rules in their function. Once again, don't be concerned with this aspect unless you are a programmer wishing to write a command for geoSHELL. If so, all the needed information is included in the 'geoSHELL Programmer's Package'.

Internal Transient Commands

When geoSHELL loads into memory, it contains a number of built-in resident commands. But there are still a number of commands contained in the geoSHELL file on disk. These are the internal transient commands. To access any of these, you simply type the command at the keyboard just as you would for any other command. geoSHELL will take care of the rest for you.

atype

This command will display a text file to the screen that is stored as true ASCII characters. You, however will find that you very rarely need to use this command since the 'type' command can also display a 'true' ASCII file. You would only need to use this command on those rare occasions when 'type' fails to properly identify the file it is displaying. Refer to the 'type' command for more info on this one.

custom

While using geoSHELL, you might find yourself using some of the commands that change certain settings. When you have these settings established that you would like to use all of the time, you can use the 'custom' command to save them to the geoSHELL file on the current drive. This will only act on the current drive. This gives you control in case you wish to establish different defaults for different geoSHELL files.

The settings that are saved are as follows:

*** videomode ***

This is established on the 128 with either the 'mono' command or the 'rgb' command.

*** display type ***

This is set with either 'ddate' or 'dtype'.

*** fastdirectory ***

Set this one with either 'fastdir' or 'slowdir'.

*** 80 column colors ***

Use 'color80' for these settings.

*** background patterns ***

The command 'backpatt' will set this one. Each mode, 64, 128-40 and 128-80 can have separate settings with this command.

*** printer settings ***

The command 'pcont' sets these for you. These settings are then used when using the '@P' command.

*** function keys ***

Your function key settings are also saved with 'custom'.

Your hotkey settings can only be saved with the 'savekeys' command. The command 'custom' has no effect on your hotkeys.

This command requires no parameter.

dcopy

This is a 'whole disk' copier meaning that it does not copy individual files by themselves, but it copies individual sectors on the disk that contain valid data. Since every file on the disk is stored as valid data, then every file will be copied with this method.

Using 'dcopy' to copy every file from one disk to another is much faster than copying each file one at a time. This is a two-drive copier. Both drives must be of similar types. One of the differences allowed is when copying to/from a ramdisk. Still, the ramdisk must be of the same type as the real drive. You are also allowed to copy from a single-sided 1541 disk to a double-sided 1571 disk. The 1571 disk will still be double-sided with half of it still available for copying additional files to it.

Only one parameter needs to be supplied when using this command. You must specify the destination drive, or the destination drive and partition, if it is a CMD device. If you do not specify a partition, then the currently open partition will be used.

Here's one example:

```
dcopy b
```

This will copy from the active drive (source) to drive B.

`dcopy a4`

This will copy to drive A, partition number 4, if drive A is a CMD device and if partition 4 is compatible with not only the source drive, but also the currently open partition on the destination drive. In the case of copying to a partition, the source and destination drives are allowed to be the same device. The only stipulation of course would be the source and destination partitions must be different, but of the same type.

defkey

This command will reload your default function key definitions from the current drive if geoSHELL is residing on it. If not, then it will search the other drives until it finds geoSHELL and load them from there. You will use this command whenever you wish to revert back to your default function keys after you have redefined them. You might want to refer to Chapter 8 for more info on the function keys.

No parameter is required for this command.

exec

This command functions much like the 'startup' command and uses files that are created just as you would for a startup file. One difference between an exec file and a startup file is an exec file can be given any filename desired. Also, the exec command will only execute commands from page one of an exec file. Another difference is that a startup file will only execute from the current drive, whereas the cxc command will search all of your drives and even your path partition for the desired file. This makes for a very powerful command. With just a few keystrokes, you can execute a whole series of commands at will.

`exec filename`

Naturally, a filename is used for the parameter. Also, just as with a startup file, you create an exec file using GeoWrite. Refer to Chapter 5 for more info on exec files and startup files.

fcopy

This is a filecopier that can copy any type of GEOS file as well as any standard CBM

file except for relative files. When you use this command, the currently active drive is the source drive and you must specify the destination drive. If a file of the same name is found on the destination drive, it will be deleted and replaced with the file that you are copying.

'fcopy' also has the ability to duplicate a file on the same disk, and it can also copy files from one partition to another or from one drive to another drive's partition on a CMD device (HD, FD, RD or RL).

Here is one example:

```
fcopy b filename
```

This is the format that you would use the most. The 'b' in this example is your destination drive, and 'filename' would be the name of the file you want to copy. Be sure to put a space before and after the 'b'. The destination drive can be any other drive that is currently logged on to your system.

To use fcopy in it's 'duplicate' mode, you must specify the desired file much like you would when you are using the 'rename' command:

```
Example:  fcopy b newfile=oldfile
```

This will make another copy of the desired file and give it the name that you specify. In this manner, fcopy will work with the currently active drive as the destination. If the new filename already exists, then the copy will not proceed. This mode of fcopy is handy for making successive backups of a data file you might be working on. You can duplicate a file to any drive or partition on your system. Since fcopy normally overwrites a file with the same name, you can use the duplicate mode to prevent this from happening. By making the newname the same as the existing name, you will be assured that an existing-copy is not overwritten whenever you wish to copy a file and keep the name the same.

```
fcopy a2 filename
```

This example would copy a file from the current drive to drive A, partition number 2. If drive A is the current drive, then it would copy from the current partition to partition 2, as long as both partitions are of the same type (i.e. 1581 and 1581). If partition 2 is the current partition, then you will be reported with an error. For this purpose, you would want to use the duplicate mode as suggested above.

Example: `fcopy h3 filename`

In this manner, you are specifying the HD as the destination drive with the 'h' in the destination parameter. You are also telling `fcopy` to use partition 3 as the destination partition. It makes no difference as to which drive is the currently active drive in this case, the destination will be whatever you specify. In this case, it is partition 3 on the hard drive. The only time this won't work is if partition 3 on the HD is the current partition on the currently active drive. If you have an FD Series floppy drive, you would use an 'f' to specify the destination drive. For a RamLink or RamDrive, use an 'r'. Always follow this with the desired partition number. Of course, the partition must be of the same type as you are currently using on that particular device. You can't open a native partition if you are using 1581 partitions and vice-versa.

'`fcopy`' requires two drive devices except in the duplicate mode, or when copying from partition to partition. It is unable to copy a file from one disk to another on a single drive system. It was felt that this would require too much disk swapping and was not implemented. For copying with single drive systems, you should use the DeskTop, since it uses a larger buffer in the computer, thereby requiring fewer disk swaps. It is still an uncomfortable process, however.

info

You can use '`info`' to display useful information about a file. Every GEOS file has what is known as a header block attached to it. Within this header block on the disk is various bits of info such as the type of file, the author of the file, etc. If the file requested is a data file, '`info`' will display the 'parent' file that this data file belongs to, such as GeoWrite being a parent to one of its data files. Applications and desk accessories have the author's name stored within the header block. '`Info`' displays this also. '`Info`' will check the date in the directory that goes with this file and display it for you. Also contained within the header block is a memo about the file. This memo may have been created by the original author of the file or it might be a memo that you added yourself.

`info filename`

Just as in this example, you merely supply the filename and `geOSHELL` will then display the available information for that file.

install

The command 'install' has two purposes. It is used to install either a printer driver or an input driver. Whenever you wish to change one of these drivers, you would use this command. You would also want to include this command in your startup file on your boot disk. Otherwise, GEOS defaults to the joystick. Also, your printer driver would not be installed unless you do this. The DeskTop accomplishes the same thing by installing the first printer driver and input driver it finds on the disk. With geoSHELL, you specify the same operation in your startup file. As an example:

```
install NX1000
```

This would install the file named 'NX1000' and make it your default printer driver. Of course, you would specify whichever printer driver you yourself uses. On a 64, you would want to make sure that your printer driver is on any disk that you do printing from. On a 128, the printer driver stays loaded in the computer's memory and is always available.

```
install COMM 1351
```

This will install a mouse driver named 'COMM 1351' and will be used until you specify a different one.

```
install COMM 1351^ date
```

Notice that, in the above example, an 'up-arrow' was placed after the mouse driver's filename. If another command is to follow the install command, a terminator must be placed after the filename, so that geoSHELL knows that you did not request a file named 'COMM 1351 date'. In the above example, you wanted geoSHELL to display the date after it was finished loading in the mouse driver.

You can change your input or printer drivers any time you wish. Just issue this command and geoSHELL will look for the corresponding file and install it for you.

ptype

This command will display a text file to the screen that is stored as PETASCII characters. You, however will find that you very rarely need to use this command since the 'type' command can also display a PETASCII file. You would only need to use this command on those rare occasions when 'type' fails to properly identify the file it is displaying. Refer to the 'type' command for more info on this one.

stype

This command will display a text file to the screen that is stored in Commodore screencode characters. You, however will find that you very rarely need to use this command since the 'type' command can also display a 'screencode' file. You would only need to use this command on those rare occasions when 'type' fails to properly identify the file it is displaying. Refer to the 'type' command for more info on this one.

swap

If you've used at least three drives with the DeskTop, then you are already familiar with this command. This command allows you to swap drive C with either A or B so that you may run your applications from A or B if they are presently residing on drive C. geoSHELL, however, has a more powerful and useful swapping routine. It allows you to also swap drive A with drive B, thereby eliminating a couple steps that you would normally have to do if you did this with the DeskTop. With the DeskTop, you would first have to swap A with C, then B with C, and finally A and C again.

In addition to this, you can swap a drive with a non-existent one. This has the same effect as renumbering a drive.

You must supply a parameter that has the two desired drive letters in it. Here's an example:

```
swap ab
```

This will swap drive A with drive B. Now, assuming you have three drives on your system, look at this next example:

```
swap bd
```

This would swap drive B with a non-existent drive D. You would now have drives A, C, and D on your system. This feature was implemented for the purpose of installing a fourth drive to the system. Refer to the 'learn' command for more info on having a four drive setup.

The one requirement that swap has is that you must have an REU for it to work. If you

did not have an REU, you would be limited to two drives anyway and this command would not be needed. An alternative to having an REU is either a RamLink or a RamDrive from Creative Micro Designs.

type

This is one command that you will find yourself using all of the time. This one command by itself would make a good stand-alone program. The command 'type' is capable of reading and displaying text files to the screen. It can display any type of file actually. But it's beauty is in recognizing the difference between a GeoWrite file and a standard ASCII text file. It will display your GeoWrite files to the screen, so that you may read them without loading in GeoWrite. But it can also display files that were created on any other type of wordprocessor. Some wordprocessors store their files with true ASCII characters and some use Commodore's PETASCII character set. Then there are others that save them to disk using screencode. Screencode is the value that would be poked to a text screen to display a particular character. The 'type' command first loads in the file and then analyzes it. As soon as it has determined the method of character storage that was used, it will begin displaying the file to the screen.

There is no limit to the size of file that 'type' can handle. If the file is too large to fit into the buffer area in the computer, it will load in the next segment after it has displayed what is already in the buffer.

Just like most any other command that causes text to scroll on the screen, you may use the CTRL key to pause the display and STOP to abort the display. On the 128, the NO SCROLL key also functions to pause the display.

There are three other commands that are similar to 'type' that you will very rarely need to use. These are 'atypc', 'ptype', and 'stypc'. In a rare case where 'type' is unable to properly analyze the format of the text file, you would use one of these commands if you know the format yourself. 'atypc' is used if the file is an ASCII file, 'ptype' if it is a PETASCII file, and 'stypc' if it is in screencode. Perhaps only one time out of a hundred will you need to use one of these instead of 'type'. But they have been provided for just that rare instance.

type filename

This will display the file requested to the screen.

`type Text Scrap`

Yes, you can also display your text scrap with this command. In fact, you can display an entire text album by specifying the name of the album.

Type can also be used to read in any file. It doesn't necessarily have to be a text file. Many programs have text contained within them for their own purpose such as a screen message. Using 'type' can be useful at times to view any text that might be contained within a program.

External Transient Commands

These are the commands that have been supplied with the original geoSHELL disk and they can be found in their own separate files on disk. In order to access any of these commands, just make sure that they are on a disk that is in any one of your drives and then enter the name of the command. geoSHELL will take care of fetching it for you. From time to time, there will be additional disks released that will contain new commands that you can add to your system. They may also contain updated versions of existing commands.

backpatt

This is a command that is used to change the appearance of your background screen. Contained within the GEOS Kernal are 32 different patterns that are accessible for use by any GEOS application. geoSHELL uses these patterns for the background. You can choose the pattern you would like with this command and if desired, use the command 'custom' to save your defaults. Here's an example:

```
backpatt 14
```

This will use pattern number 14 for your background. Any number from 0-31 is valid.

color80

Now, here is a command that might surprise you a little. This one is for the 80 column 128 and allows you to set multiple colors while working in geoSHELL. For those 128

users with a color monitor, you can have a nicer look to your screen with this command.

The next thing that might surprise you is that this is a command that has a graphical look to it. The geoSHELL window is erased from the screen while this command is being used. You are now presented with a picture of a monitor along with various color settings that you may select from. Your settings are viewed as you choose them.

Once you selected the ones you'd like, just click on the exit box to redraw the geoSHELL window. If you like these settings, use the command 'custom' to save them as your default settings.

external

There are two ways to use this command. By issuing the command without any parameters, all of the external commands that are currently available will be listed. All of the drives are checked including the path partition if one is designated. This allows you to quickly check to see if a particular command is available. If you use 'external' followed by the name of an existing 'internal' transient command, then that command will be removed from the geoSHELL file on the currently active drive and placed into it's own file on the disk, thereby making the command into an external transient.

Example: `external type`

This will remove the type command from geoSHELL and make it a separate external transient command. The 'external' command is useful for trimming your copy of geoSHELL of commands that are rarely used. There is no way to remove 'resident' commands.

format

This command allows you to format a disk drive. In it's present version, it will only work with the three CBM drives, the 1541, 1571, and 1581. All you need to do to use it is to specify a name for the disk. Here's an example:

`format diskname`

This will then begin to format the disk that is in the currently active drive. As soon as the format process is finished, the disk will be made into a GEOS disk and then it will

be done.

Now, when using this command with a 1571 drive, the default is to format a disk as a double-sided disk. You can cause this command to only format one side of the disk, however. You would do it like this:

```
format 1 diskname
```

This will create a disk that can be properly read by a 1541 disk drive.

Another way to format a disk is with the DOS wedge command that geoSHELL provides. But, if you do it with the DOS wedge, you would want to use the command 'geos' in order to make it a GEOS disk. Refer to Chapter 6 for more info on the DOS wedge.

help

This command will provide the user with a means to view a 'helpfile' on the screen. Contained within this helpfile are numerous subjects pertaining to geoSHELL and its use. The bulk of the helpfile describes the commands.

To access this helpfile, simply type 'help'. You will now be presented with the initial help screen. From here you can request additional help as is displayed on the screen just by typing help followed by a subject that is contained within the helpfile. This helpfile always guides you along the way with subjects that pertain to the subject you are currently viewing. The best way to understand how it works is to just try it. Enter 'help' and follow the prompts from there.

hotkey

You would use this command by itself whenever you wish to view your current hotkey settings. This command also allows you to define a new hotkey or to erase an existing one. Here's an example:

```
hotkey mf=myfile
```

This will define 'mf' to load and run a file called 'myfile'. It is not put into effect until it is saved with 'savekeys'. Refer to Chapter 9 for a more detailed explanation of this command.

internal

There are two ways to use this command. By issuing the command without any parameters, all of the internal commands that are currently available will be listed. The command does not necessarily check every drive. It will only search until it finds the geoSHELL file. Then that particular copy of geoSHELL is checked and the commands contained within are displayed. The normal way that geoSHELL is searched is also used with this command. If you issue the command followed by a valid external transient command name, then that command will be eliminated as an external command and placed within the geoSHELL file on the currently active drive. It will then become an internal transient command. For this to work, the desired command must be first copied to the drive that geoSHELL is on. Then open the drive containing geoSHELL and issue the command 'internal' followed by the name of the desired command that is to be altered.

Example: `fcopy b color80^ b: internal color80`

This example assumes drive B contains geoSHELL. You are copying 'color80' to drive B and then putting 'color80' into the geoSHELL file that resides on drive B.

There may be commands released from time to time that must remain in the form in which they are supplied. In other words, a command may not work unless it resides in it's own file as an external transient. The documentation supplied with a command should detail this. The commands supplied with the geoSHELL program disk can all be moved to an internal state.

learn

This command actually creates another command when you use it. When you use CONFIGURE to set up your drives, you are actually getting the disk drivers into the system that are stored within the CONFIGURE file. CONFIGURE handles all of the drudgery for you. So with a different driver for each type of drive and the possibility that new drivers may be written for each drive type from time to time, it would be impossible for geoSHELL to supply drivers for your drives. Besides you already have the drivers in your possession. The 'learn' command provides a means for fetching those drivers and creating a command that will allow you to install a drive whenever necessary, much like CONFIGURE does.

Chapter 7 - Transient Commands

Using 'learn' is very simple. Just activate the drive that you would like the new command saved to and issue the 'learn' command followed by the drive letter of the drive you wish to create a driver for. Here's an example:

```
learn b
```

Assuming that drive A is our currently active drive, this example will create a disk driver for drive B and save it to drive A.

Now, the command that is created will look something like this:

```
install1541.64
```

If drive B is a 1541, and you are using a 64, this is what the new command will be named. If you are using a 128 and a 1581 as drive B, the new command would look like:

```
install1581.128
```

Now, whenever you want to get this particular drive back online, such as after using the 'remove' command, you would issue this new command to get the drive up and running. Whatever the device number is must not be currently used by another device on the system. For instance, if you wish to turn on a 1541 that is set as device 9, then you must not have a device currently set as drive B. If you do, just use the 'swap' command to move drive B to another empty spot. 'swap' will let you swap positions with a non-existent drive. Now, turn on the 1541 and issue the following command:

```
install1541.64 b
```

You could have these 'installxx.xx' commands saved to your boot disk and let your initial 'onboot' startup file install the fourth drive for you.

```
Example: install1581.128 c
```

This would make a 1581 that is drive 10 be recognized by GEOS on a 128 system.

```
Example: installRM71.64 d
```

This would make your REU emulate a RAM_1571 as drive D on a 64. The ramdisk can be installed in any drive position A through D as long as there are no other drives being used in the desired position.

If you are installing a CMD device, it is always a good idea to immediately follow with a 'change partition' command so that the DOS in the drive and GEOS are working in the same partition as follows:

Example: `installHD8i.128 b @cp10`

This eliminates any problems you might encounter with the built-in DOS wedge. In the case of a ramdisk driver on a device that is capable of having several ramdisks or partitions such as an expanded REU or a BBG ram with several partitions, you can rename the `installxxxx.128` or `installxxxx.64` command to any name you wish to reflect the particular portion of the REU that is being used. With a little thought, you can install more than one ramdisk on the same device. Keep in mind, however, that there can only be one `RAM_1581` in use at any one time.

Let's say you are using a BBG ram with 5 1571 partitions. Use the `learn` command on each partition, but rename the created commands before using 'learn' on the next partition. Let's assume now that you have named the commands `part1`, `part2`, etc. Once you have done this with each of the five partitions, you can now change partitions right from `geoSHELL` without the use of a separate utility. All you have to do is use 'remove' in conjunction with your new commands.

Example: `remove b part2`

This example removes drive B from the system, but the rest of the command line reinstalls the BBG ram with partition 2 being opened. It's as simple as that. The best way to use the BBG ram would be to have `geoSHELL` and all of its commands in one of the five partitions and that partition could be installed as drive D. Then, any of your other partitions could be used as drive A, B or C, or a combination of two or three of them. `geoSHELL` allows you to have as many as four partitions open at once, but you might want to keep at least one floppy running as one of the devices.

lock

This command will either set or clear the write-protect bit of a file. It is a toggle function. If the file is already write-protected, this command will unlock the file. A locked file cannot be overwritten or deleted under normal circumstances. Only the currently active drive is searched for the desired file.

Example: `lock filename`

This will toggle the write-protect status of the desired file.

path

This command will come in very handy to those who use a device made by Creative Micro Designs. It works with either the hard drive, the FD drives, RamLink or RamDrive. It allows you to specify a 'path partition' for geoSHELL to look at when it searches for it's commands. You will find more information about this command in Chapter 10.

pcode

This will send control codes to your serial printer or interface. The parameters used must be ascii representations of numbers from 0 through 255 with a space between each number.

Example: `pcode 27 64`

This is the code to reset most printers.

You can send as many codes as will fit on a command line, or until an up-arrow terminator is found. You will find this command very useful for configuring your printer or interface for special functions in GEOS.

Example: `pcode 27 83 48 27 77 27 15 27 65 6^ @p dir @p`

This example works on a Star NX-1000 and may also work on other printers as well. It will set the printer for condensed elite superscripts, and set the line spacing at 1/12 inch and then print the directory of your current drive in a very small print.

pconf

This command stands for 'printer configuration'. It allows you to either view your current printer settings or to set them to work with your printer.

Now, this has nothing at all to do with your printer driver. It sets up geoSHELL for use with the '@P' command. Whenever you use '@P', all output to the screen will also be directed to your printer.

Chapter 7 - Transient Commands

Some printers need different signals sent to them than others. The settings that you can change with this command are as follows:

- * device number (4 or 5)
- * secondary address (0-255)
- * linefeeds (y or n)
- * character mode (a or p)

Here's an example for using these settings:

```
pconf 47ya
```

This will use device number 4 with a secondary address of 7. The 'y' tells geoSHELL to send linefeeds after a carriage return. And the 'a' informs it to use ASCII characters. If your printer did not need linefeeds, you would use an 'n' in place of the 'y'. And if your printer required PETASCII characters, you would use a 'p' in place of the 'a'. Here's another example:

```
pconf 512ya
```

This will use device number 5 with a secondary address of 12, send linefeeds and ASCII characters. You might want to refer to Chapter 6 for more info on using the DOS wedge command '@P'.

remove

This command will remove a drive from the system. All you have to do is specify the desired drive, A, B, C, or D and geoSHELL will then make that drive transparent to GEOS. After issuing this command, it is a good idea to turn the drive off, if it is a real drive. If it is a ramdisk, then geoSHELL handles the whole job for you. Here's an example:

```
remove d
```

This would take drive D out of the system. If you use four drives and wish to exit back to either the DeskTop or gateWay, you should remove drive D from the system. The reason is that the DeskTop and gateWay do not support the fourth drive like geoSHELL does. It is likely that you could crash the system and have to reboot. It has been found that the DeskTop won't bother the fourth drive unless you attempt to swap drives while in the DeskTop. Do all of your drive swapping with geoSHELL. All

other operations will work however. With `gateWay`, your fourth drive will either cause a crash or it will be gone anyway when to go back to `geoSHELL`.

You might also find yourself with a need to use `'remove'` after a disk crash of some sort. Just turn the drive off and issue the `remove` command. This way the drive won't confuse GEOS until you can get it reinstalled. Refer to the `learn` command on how to accomplish this, or use `CONFIGURE`.

rtc

You can use this command to set your GEOS clock if you have a CMD device with a real-time clock. You may use it at any time, or preferably, you should place it in your boot startup file to automatically set the clock when you first boot up. The command only checks the current drive for the existence of a clock. So, you must first select the drive containing the clock and then enter the command. It was done this way since it is possible to have more than one device with a clock in it.

Example: `b: rtc`

This example assumes that drive B contains a clock.

savekeys

This command will only work once you have redefined one or more hotkey with the `'hotkey'` command. It will save your new definitions to the `geoSHELL` file that is on the current drive. Refer to Chapter 9 for more info on this command.

This command requires no parameter.

Chapter 8

The Programmable Function Keys

geoSHELL gives you the ability to use the function keys. You may use the default settings or you may also assign your own settings to the function keys. Programming a function key is very simple. You assign whatever command or multiple command sequence you would like by first typing in one of eight commands representing the desired function key, followed by the desired sequence of commands. Here's an example:

```
key3 date time dir
```

The first command tells geoSHELL which key to assign the string of commands that follows. 'key3' of course would indicate the 'F3' key. The three commands following 'key3' in our example are then assigned to the 'F3' key. Now anytime you press 'F3' you will be given the date and the time, followed by a directory listing from the currently active drive.

If you type in the command 'key', geoSHELL will display all eight function key definitions. From this point, you can easily change one of the function key definitions by cursoring up to one of them. Just make whatever change you need and hit RETURN. This is handy for making small changes to an existing sequence of commands.

Function keys may be programmed from within a startup file. Whenever you exit an application to return to geoSHELL, the startup file is read and any commands found will be executed. If you need your function keys programmed a certain way for certain disks, simply define them in your startup files. You can do it differently for each different disk if you wish. Of course, if you forget how they are set, just type in the command 'key' to view the current definitions. If geoSHELL either does not find any commands in the startup file to define the function keys, then they will be set to the default setting as contained in geoSHELL itself. You could even program one function key to program another, or even itself, when it is pressed.

If you would like to revert back to the default function key settings after they have

been altered, use the command 'defkey'. Whatever settings are contained within the geoSHELL file on disk will be used. Or, after redefining your function keys, you can use the command 'custom' to save your new settings to the geoSHELL file on the disk. Refer to the 'custom' command in chapter 6 for more information.

There are many uses for the function keys as you will soon find out. Program one of them to load and run your favorite Desk Accessory or Application, or to execute your most used command sequence. One thing to keep in mind is the length of the sequence that is allowed by geoSHELL. You are limited to 80 characters for each function key. Of course, you cannot exceed this limit from the immediate mode, anyway. However, from a startup file, you could. But, geoSHELL will catch it, quit executing the startup file and report the error. This error might also be caused if you should happen to forget to include the terminator at the end of the function key definition. Unless a function key definition is the last thing in your startup file, you have to terminate it with the CMDR/UP-ARROW. Notice, this terminator is not the usual 'UP-ARROW', but the CMDR/UP-ARROW. With this terminator, geoSHELL knows where the end of your definition is and will once again begin executing commands following the definition. In 80 column mode on the 128 your limit is 73 characters. On the 64 or the 128 in 40 column mode, this limits you to 33 characters. These limits are due to the amount of characters that will fit on a line in the geoSHELL window. For most purposes, this is adequate. Remember, if you need the full 80 allowed characters, use either a startup file or an exec file to define the functions key.

If you should happen to have a function key definition that contains more characters than can be entered from the immediate mode, do not try to edit it from the immediate mode or you will actually shorten it to the maximum of 33 characters (40 column mode) or 73 characters (80 column mode) without realizing it. You can view the definition (at least a portion of it) but you won't want to edit it, unless you wish to shorten it anyway.

There may be times when you might want to define a function key to do absolutely nothing. To do this you would enter the command such as 'key6' without a parameter. This would erase the existing definition for the corresponding key. Then any time you press that key, nothing will happen.

Make use of the power you have with the function keys, they are provided as a means of preventing repetitious typing.

Chapter 9

The Built-In Hotkeys

What are the Hot Keys?

A 'hotkey' is a combination of any two keys typed in from the keyboard to perform the function of loading and running an application or desk accessory. They are provided to save a little typing when you wish to load a file. This is just one more way that geoSHELL is user-friendly.

A number of hotkey commands have already been defined for your use. They are implemented for executing all of the popular programs that have been supplied for GEOS. For instance, simply enter the two letter combination 'wr' and geoSHELL will search your drives for GEOWRITE. If it is found, then geoSHELL will load and run it. 'pa' will load and run GEOPAINT. The built-in hot keys are designed to be easy to remember.

You can have as many as 128 different hotkey definitions for the 128 and just as many for the 64. There is a storage area in the geoSHELL file for each computer. This allows the user of both systems to have different hotkey definitions for each one.

When you enter a hotkey such as 'wr', geoSHELL first must find the list of hotkeys in the geoSHELL file. If it finds the hotkey that you typed, it will then search for the corresponding file. So, for this reason, geoSHELL must be on one of the disks in any of your drives.

The 'hotkey' and 'savekeys' Commands

There is a transient command supplied on your geoSHELL program disk that allows you to view your current hotkey settings. This command is called 'hotkcy'. This command also has the ability to define new hotkeys. Or you can use it to redefine existing ones or to even erase existing ones. It is quite versatile.

Viewing Your Current Hotkey Settings

In order to view your current hotkey settings, simply type the command 'hotkey' by itself without any parameter. You will be presented with a list of hotkey definitions that will scroll on your screen. As usual, you can use the CTRL-C key to pause the listing, and STOP to abort it. Using 'hotkey' by itself has no effect on your current settings. It merely lets you view them.

Changing Your Hotkey Settings

The command 'hotkey' will let you redefine your hotkey settings if you supply a parameter with it. The best way to explain how to do this is with an example:

```
hotkey wr=GEOWRITE
```

Of course, this one is not necessary since we already had GEOWRITE defined, but it is being shown for this example since we are all familiar with this program. Now, 'hotkey' can also erase a hotkey definition. And it works very similar to the way you would erase a function key definition. You merely enter the hotkey command followed by the two letters to be erased plus an equal sign, and follow it with nothing. Just like this:

```
hotkey wr=
```

Now GEOWRITE will no longer be loaded when we type the two letters 'wr'. Instead, geoSHELL would look for a file called 'wr' and would fail to find it.

We can redefine an existing hotkey command exactly the same way that we define a new one. When you define a hotkey that already exists, it will be replaced by the new definition.

Saving Your New Definitions

The command 'hotkey' can be used over and over to define any number of hotkeys up to the maximum of 128 for the 64, and 128 for the 128. Just keep using the command until you have defined all of the hotkeys you need. Every time you define a hotkey, it gets added to a temporary table in memory. The command that allows you to save this table to the geoSHELL file on disk is 'savekeys'. When you finished defining a new hotkey, or a series of hotkeys, just enter the command 'savekeys'. If 'savekeys' is available on one of the disks in your drives, geoSHELL will fetch it and execute it. At

Chapter 9 - Hotkey Commands

that point 'savekeys' will then save the new table of hotkeys to the geoSHELL file that is on the currently active drive. It will not save to any other drive. If you wish to save to one of the other drives that geoSHELL is residing on, you must open that drive and then enter 'savekeys' again. This does, however, allow you to use the command over and over again to save the hotkey table to any number of geoSHELL files. You may also change disks in the current drive and issue the command again to a different disk.

There is one thing to keep in mind when using 'hotkey' and 'savekeys'. These two commands are designed to work together. If you were to use another transient command or one of the resident commands that uses the transient command memory area, you would erase the hotkey table that you created and would be forced to start over. 'savekeys' will catch this and report that it can't find the hotkeys in memory.

Take advantage of this powerful feature of geoSHELL and define new hotkeys for all of your most used programs. Make them easy to remember. This is one way that geoSHELL makes life easier.

Chapter 10 Special Features For CMD Devices

geoSHELL has been designed to work with any type of disk device that is capable of being used with GEOS. One of the main companies that is supporting the Commodore computer and the GEOS community with some excellent hardware is Creative Micro Designs. They have an excellent line of HD Series hard drives, FD Series 3.5 inch drives, and the RamLink and RamDrive units. geoSHELL does not leave you out in the cold if you own one of these machines. In fact, geoSHELL is the ideal interface for use with any of these devices. Many of the features contained within geoSHELL are centered around these devices and help you to use them more effectively.

Special DOS Wedge Commands

If you use a CMD device, especially the hard drive, you will have a need for changing partitions. geoSHELL incorporates a built-in DOS wedge and let's you change partitions with the following command:

```
@CP3
```

This will change to partition number 3 if it is the same type of partition as the one you are currently in. If you are currently working in a 1581 partition, the one you are changing to must also be of a 1581 type. If it isn't, geoSHELL will not let you change to it.

Another way to change partitions is to enter the DOS command '@S=P' to first display a list of partitions. While the list is scrolling, use the CTRL key to pause it and then the STOP key to halt it. Now if you see the partition you want, just cursor up to it and hit RETURN or double-click on it and the partition will be opened for you. geoSHELL makes it painless for changing partitions. You could also use wildcards to display a selective directory of partitions to narrow down your selection. You would use these wildcards just the same way as is stated in your manual that came with your CMD device.

You will find that most any of the DOS wedge commands or JiffyDos commands that are listed in your CMD manual will work while in geoSHELL.

Refer to chapter 7 under 'Internal Transient Commands' to find out how the copy commands contained within geoSHELL allows you to copy files between partitions and also to copy whole partitions from one to another or to and from a floppy drive.

The 'path' Command

The 'path' command should become a favorite among owners of CMD devices. Especially those that have enough space to allow multiple partitions. Contained within geoSHELL are some special routines that it uses whenever the user types in a transient command. geoSHELL is capable of looking for it's commands on a partition that is predefined with the path command. This allows you to keep geoSHELL and all of it's commands on just one partition instead of having to have a copy of the same files on each and every partition. It's almost like having an extension of the computer's memory. Yes, you can have just one copy of geoSHELL on your path partition and it will be used as though it is on every drive and partition on your system. It is always available for your use. The secret is to have a copy of the file called 'getshell' on every partition that you use as well as your ramdisk and any other drive that is online. As long as GEOS can get to the file 'getshell', then getshell will take care of getting geoSHELL loaded when needed. Once geoSHELL is in memory, then it has it's own routines for fetching it's own commands from the path partition as needed. 'getshell' has the advantage of only being less than 2K in size and so it takes up very little room on each partition and disk.

To establish your path, you simply use the command 'path' followed by a parameter that specifies the path. Here's an example:

```
path h3
```

This will make the hard drive the default path device with partition number 3 as the path partition. If you use an FD series drive, you would substitute an 'f' in place of the 'h' in the parameter. For a RamLink or a RamDrive, use an 'r'. It's as simple as that.

You only have to specify a path one time during your session. It is in effect until you turn the computer off. For this reason, it would be handy to specify it in your boot startup file. Of course, you can change it at any time. Also, the path is not established

until it is issued. For this reason, geoSHELL must be on your boot disk, since it must be loaded before the path command can be used.

If you wish to view your current path, just issue the command by itself without a parameter. Your current setting will then be displayed.

Some of the geoSHELL commands themselves will also fetch information or data from the path partition. The 'help' command will read the 'helpfile' from the path partition if it is there. The 'exec' command will also check the path partition for the desired exec file. Other commands only work with the currently active drive or partition, such as the 'type' command. Naturally, the 'rename', 'delete', and 'custom' commands only affect the current drive or partition. The reason is because these commands write to the disk and you have more control as to what drive is affected by these commands.

The beauty of geoSHELL's multiple commands on a single line can let you change partitions, perform a command there and then change back. All of this from a single line of commands. Use your imagination when you wish to perform a series of commands.

There might be an occasion when you will want to turn your path off. This is very simple, as in the following example:

```
path off
```

After executing this, geoSHELL will no longer look at a partition that is not currently visible to the system. In most cases, however, you will want to take advantage of this powerful feature of geoSHELL and leave it turned on all the time.

CMD has provided us with these fine products for our Commodore computers and geoSHELL has been provided to make use of these products.

Chapter 11

The Commands At A Glance

This Chapter is designed to give you a quick look at all of the available commands that are used in geoSHELL. For a more indepth discussion, you may want to refer to other parts of this manual. This is a good place to look for a quick memory refresher. The commands are listed in alphabetical order with a very brief description and one or two examples if needed.

a:, b:, c:, or d: (resident)
Activate a drive.

always (resident)
Used only in a startup file. Commands following it are executed always.

atyp (internal transient)
View an ASCII text file.
Example: atyp filename

backpatt (external transient)
Change the background pattern. Valid parameter is 0-31.
Example: backpatt 14

basic (resident)
Exit GEOS and return to Basic.

boot (resident)
On a 128, this will exit to BASIC and boot the disk in the current drive if the disk contains a valid boot sector.

catalog (resident)
This is the same as 'directory'.

cdir (resident)

Chapter 11 - Command Summary

View a directory according to a time parameter.

Example: `cdir`

Example: `cdir 0623p`

Example: `cdir 102292`

clr (resident)

Clear the geoSHELL window.

color80 (external transient)

Set the colors on the 128's 80 column color mode.

custom (internal transient)

Save certain settings to make them your defaults.

date (resident)

Display the current date or set the date.

Example: `date`

Example: `date 102392`

dcopy (internal transient)

A whole disk copier

Example: `dcopy b`

ddate (resident)

Causes directories to display filenames with the date and time.

defkey (internal transient)

Load the default function key definitions.

delete (resident)

Delete a file on the active drive.

directory (resident)

Display a directory with or without wildcards.

Example: `dir`

Example: `dir G*`

dtype (resident)

Causes directories to display filenames with the filetype and size.

echo (resident)

Chapter 11 - Command Summary

Echo a message to the screen.

Example: `echo This message to the screen^`

end (resident)

Ends a startup or exec file

exec (internal transient)

Execute a file of commands

Example: `exec filename`

external (external transient)

If used by itself, this will display the external transient commands currently available to geoSHELL. Or if followed by an internal transient command name, it will turn that command into an external one. The following example will change the "info" command into an external command on disk.

Example: `external info`

exit (resident)

Return to the DeskTop.

fastdir (resident)

Causes directories to display just the filenames.

fcopy (internal transient)

A filecopier.

Example: `fcopy b filename`

format (external transient)

Format a disk in the current drive.

Example: `format diskname`

geos (resident)

Make a disk a GEOS disk.

getkey (resident)

Wait for a keypress in a startup or exec file, and then process the input.

help (external transient)

Display a help screen from a file called 'helpfile'.

Example: `help`

Example: `help subject`

Chapter 11 - Command Summary

hotkey (external transient)

Define your hotkey settings or display the current ones. Works in conjunction with 'savekeys'.

Example: hotkey

Example: hotkey pa=GROPA IN'

info (internal transient)

Display various bits of info about a file..

Example: info filename

input (resident)

Display the name of the current input driver.

internal (external transient)

If used by itself, this will display the internal transient commands currently available to geoSHELL. Or if followed by an external transient command name, it will turn that command into an internal one. The following example will change the "hotkey" command into an internal command contained within the geoSHELL file.

Example: external hotkey

insert (resident)

Insert a string of characters at the current cursor position.

install (internal transient)

Install a printer or input driver.

Example: install 128 COMM 1351(a)

Example: install NX1000

key (resident)

Display the function key definitions.

key1 (resident)

Redefine a function key. Also use key2, key3, etc. CMDR/UP-ARROW required as a terminator when other commands follow.

Example: key3 insert fcopy b |

learn (external transient)

Create a disk driver command such as 'install41.64'.

Example: learn b

Chapter 11 - Command Summary

load (resident)

Load a transient command into memory without executing it.

Example: load backpatt

lock (external transient)

This command will lock or unlock a file which changes its write-protect status.

Example: lock filename

mono (resident)

Switch the 128 to monochrome mode.

onboot (resident)

Used only in a startup file. Commands that follow will execute only during initial bootup.

output (resident)

Display the filename of the current printer driver.

parent (resident)

Move up one directory in a native partition.

path (external transient)

Define your path partition or display the current one.

Example: path

Example: path h2

pcode (external transient)

This will send control codes to your serial printer or interface. The parameters used must be ascii representations of numbers from 0 through 255 with a space between each number.

Example: pcode 27 64

pconf (external transient)

Define your printer settings for the '@P' command, or display the current ones.

Example: pconf

Example: pconf 47ya

print (resident)

Print a data file.

Example: print filename

Chapter 11 - Command Summary

ptype (resident)

Display a PETASCII text file.

Example: ptype filename

remove (external transient)

Remove a drive from the system.

Example: remove c

rename (resident)

Rename a file. This works only on the active drive.

Example: ren newfilename=oldfilename

rgb (resident)

Switch the 128's 80 column screen into multi-color mode.

root (resident)

Move to the root directory if in a native partition.

rtc (external transient)

Set your GEOS clock from the rtc chip in the current drive.

run (resident)

Load and run a file.

Example: run filename

Example: run file*

s40 (resident)

Switches the 128 to 40 column mode.

s80 (resident)

Switches the 128 to 80 column mode.

savekeys (external transient)

Save your new hotkey settings to the geoSHELL file on the current drive.

slowdir (resident)

Causes directories to display filenames plus additional info.

startup (resident)

Execute a startup file.

Example: startup

Chapter 11 - Command Summary

Example: startup 5

status (resident)

Display the drives that are currently online.

stype (resident)

Display a text file that is stored in Commodore screencode.

Example: stype filename

swap (internal transient)

Swap drives.

Example: swap ac

switch (resident)

Switch the 128 back and forth between 40 and 80 columns.

time (resident)

Display the time or set the time.

Example: time

Example: time 0834a

type (internal transient)

Display any type of text file.

Example: type filename

wait (resident)

This works only in a startup file. It waits for the user to hit any key.

A Special Thanks...

I want to thank you for purchasing this software for your Commodore computer. An extreme amount of work went into developing it. I hope you enjoy using it as much as I enjoyed ... creating it.

I want to thank John Lindemann and Ron Gearhart for their excellent testing of geoSHELL as it developed from one stage to another. They also contributed many ideas that I incorporated into the program.

I also want to thank Berkeley Softworks (GeoWorks) for creating the GEOS operating system for the Commodore 64 and 128. If it weren't for GEOS, I would have had no reason to write this program.

My biggest gratitude goes to my wife, Brenda. She put up with the many hundreds of hours that I sat (plopped) in front of the computer, working on this program.

Maurice Randall

Maurice Randall
Click Here Software Co.
P.O. Box 606
Charlotte MI 48813

PH: 517-549-6202