CHRISTMAS

GEOWORLD

The definative magazine about GEOS

Table of Contents

Ad & Article Copy Deadlines are the 1st of the previous month that you wish the material to appear.

As the world GEOS by...
By Roger Ledbetter

Holiday Greetings GEOWORLD readers, I have been busy as usual here with the printing of GEOWORLD, a full time job, and raising My 3 children (which by the way help alot to get this magazine to You). I would have never thought the 64 computer could change My life as it has this past Year, and I want to thank each and Everyone for the support that I have gotten. Special thanks to the staff at BSW for GEOS and all of the applications that have really in all senses given a new life to the 64. I know GEOS has done the same for many others by the letters I have gotten and I feel priveliged to be of some help.

There has been some activity on other computer services besides Qlink. Bill Coleman (Master Blaster) is now the SYSOP of a GEOS SIG on GEnie. Bill has written a new convert program so that GEOS applications can be uploaded / downloaded. Also Joe Collins has started a group on PlayNet. It is great to have another place for users to go for support so check them out. I will keep You posted on how these new SIGS are doing.

GEOPUBLISH has been out for a while now and I feel that this is the best thing (except for GEOS itself) to have ever happened to the 64. Now that everyone can produce professional looking layouts with an inexpensive system, I wonder what kind of future lies for the "toy computer"? It will be interesting to see in 1988 the impact that GEOPUBLISH will make. Please send in samples of what You have done, I would love to see them and perhaps some of the very best will be published in the pages of GEOWORLD.

Some exciting news comes from the World of Commodore Show, Paul Hughes reports that BSW will be releasing a new desktop for the 64 system. It will support the 1581 3.5" drive which is super fast, and also allow users to configure the RAM disk with 512K! BSW has not set a date for release yet and details are sketchy, but expect to see desktop 1.4 next year in the springtime. I will pass along the info as it becomes available.

Well thats it from Your publisher. All the Best from the Ledbetter Family to You and Yours and I am looking foward to bringing GEOWORLD to You in 1988. Happy New Year !

Roger Ledbetter

User help

GEOPublish-128
FROM: RoyL1
I was told that my Gp will work in 128 mode in 40 columns.I first tried to install GP in the 128 mode but it would not work. It does work when I installed it in 64 mode.What is correct?
SUBJ: geoPub 64/128
FROM: Frank T10
geoPublish WILL work in 128 (40 col) mode. That's how I'm using it. What may have confused you is that it will not INSTALL if your drive #8 is configured as a 1571. You have to -re configure it, temporarily, as a 1541 to do the installation. After that, you shouldn't have any problems. However... I have noticed one apparent bug. If I (forgetfully) click on the geoPublish icon while in 80 columns, I get a dialog box telling me to switch to 40. But after I do, and geo-Publish loads, the system locks up. If I simply load geoPub from 40 columns in the first place, no problem.

THE HOT TIP....
FROM: Denton2
with geocalc is to be sure that your spreadsheet is column-ized. I laid out a credit card sheet horizontally and found that changing a value in a cell (say the minimum payment), would take forever to re-calucate the sheet. Changing the layout to column calcultions rather than row, increased the re-calcultions at least 1000%. In addition, the illegal formula DB loop does not seem to be releated to the REU. I had it happen on a normal drive. I think if you answered the ok enough times you would be 'ok', but I don't have that kind of patience.

"What's it's worth"
FROM: GeoFast
I was having trouble with my "select menus"(i.e. in geoWrite or geoPaint, after selecting OPEN, you get a window so you can select the existing file you want to open. There are other "select menus" of this type thru-out GEOS.Well, I was having trouble with all of them!! When I'd get to one of them, the "highlight" would be sitting in the area ABOVE the window; clicking on a file in the window would bring the highlight down to that area but wouldn't clear up the area above the box where it had been sitting, like on the ruler in geoWrite. Confused yet? Anyway, I called CS at BSW...they said I'd have to send my boot disk back. Almost in tears, I had an idea! I realized that I hadn't noticed this problem until I updated for the 1764 REU. So what did I have to loose?!!...
I dug out my Version 1.2 backup (remember the one you're supposed to make at the outset? and proceeded to restore my boot disk to V1.2. After that was done, I pulled out the upgrade disk and went thru that routine again. And you know what?...I don't have the problem with the "select" menus anymore!! And I'm not bothered with the system lock-ups that would usually occur after using a "select" menu, like in geoLaser; after selecting the document, I'd get the printout but then everything would be locked up! Well, not anymore! I just wanted to let you know that sometimes you can cure the problems without having to send your stuff back to BSW.

2 cents worth   A Letter from the Editor

A while back I was speaking with a LaserDirect customer on the phone regarding some intricate procedure for creating a geoWrite document. The discussion headed into the topic of poor Customer Service at Berkeley Softworks. While I can appreciate the frustration and sympathize with the aggravation of getting through to the Customer Service staff at BSW, I for one am getting a bit tired of hearing the negative talk regarding BSW. What would you propose as a solution to the alleged "lack of Customer Service" at BSW? Do they throw more people on telephones? (How do they pay them?) Are you prepared to pay more for the software you purchase? (Where else do they obtain the payroll?) Does BSW expand its newsletter to address a broader, non-Q-Link, customer base? If Q-Link is the official Customer Service avenue, does BSW give free Q-Link access to all of its customers? (Who pays for this? BSW? With what?)

I belive there is a difference between "system" service and "aesthetic" service. Difficulty in booting your GEOS disk is a system question, and could be answered by reading your manual. Aesthetic questions, as those received by LaserDirect daily, deal with how to layout a page, how to get reverse type within geoPublaser, etc. These aesthetic questions are not properly directed to BSW, but rather, can only be answered by time spent with the software, visiting your library or or calling an expert.

My personal feeling on the subject is that we all should pay our own way. Keep the software price as low as possible and expect those users who require more Customer Service to pay for it. My experience has been that those that complain about not being able to get through to BSW, admit to me that they don't want to read a manual to get the answer-they want to have someone talk them through it. Do we throw more money at the problem? Consider this: Ventura, Desk-Top publishing software from Xerox sets aside a sizeable percentage of its costs to customer service. It also retails for $795!

Here's a challenge- Think long and hard on a solution to the Customer Service "problem" at Berkeley. Write down your proposal and send it to me at Pro-Mark Associates (LaserDirect). I will forward all of your solutions to BSW at my expense. If BSW uses your idea, I'll make sure that you receive 30 pages of GEOS documents LASER PRINTED FREE! Use your head on this one - its bigger than your mouth for a reason.

ONE UP ON COMPUTER SNOBS!
by Susan Albert
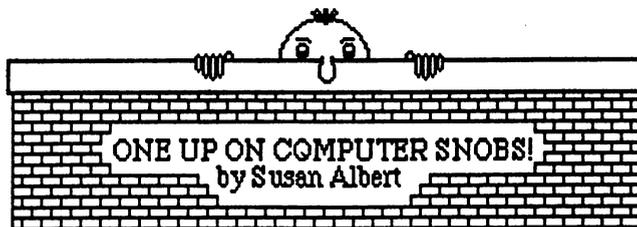
Computer snobs. We all know them. They're the ones that ask what kind of computer you own, then say "Oh" when you tell them a Commodore. Of course, "Oh" is always said a condescending tone that means both "All you can do with that is play games" and "I was talking about real computers" at the same time. I guess everyone will always be loyal to their brand of computer, expounding its virtues to anyone that will listen. I used to argue with those computer snobs over the qualities of my Commodore 128 - good graphics, impressive sound capabilities, and a superior Basic. My arguements did no good, I don't even think they heard them. They always came back with comments on the non-standard RS232 port or the size of a 1571 Disk Drive. I gave up. Until now. Now there's GEOS.

As all GEOS users know, GEOS gives Commodore owners capabilities previously found only in far more expensive systems. Here is a case in point. I'm a Project Director for a vocational education program at a community center. We have eight Apple compatible computers, one Apple IIe, and two IBM compatibles. Although my budget doesn't allow money for alot of software, we do have the necessities. Recently I was asked to design an invitation for a United Way function that we were involved in. Should have been simple since we had software that would do just that. The only problem was that all the copy needed wouldn't fit on one invitation - even using the front, back, and both inside pages. GEOS to the rescue. I had just received my GEOS 128 and decided to put it to the test. Each page of the invitation was designed using Geopaint. This gave me the opportunity to layout the words and graphics exactly were I wanted them. Once the design was completed, all I had to do was cut each page of the invitation and lay them out so that when copied and folded, all pages would be upright.

Now I know that I could have probably done the same thing on one of the computers at work with the right software, but not at the same cost as GEOS. Not to mention the fact that my system only consists of my 128, one 1571, a Star NX-10 printer, and a four dollar joystick. Just the bare essentials. Could any other computer match that? I doubt it. I'm working on a proposal now that will involve duplicating some charts and tables with additional text. Another job for GEOS. It's flexible and efficient without costing a fortune. When the computer snobs start talking, I'll be ready. GEOS puts all of us one up on them!

EDITOR'S NOTE:
In our November issue Nathan Williams, "Installing GEOS", stated that C64 GEOS programs and add-ons would not work within the C128 mode. I have received many calls regarding this subject. If you configure your 1571 as a 1541 before installing the application, you should have no problem in using any C64 application within the 128 mode.

While the term "add-ons" is not clear, remember that C64 geoWrite 2.0 documents will not work within the 128 mode when used with the original 128/1.3 geoWrite. (2.0 is a higher version number). This users 64 geoDex works fine within the 128 environment, as well as my geoWrite, geoPaint and geoFile program and application files. Perhaps Mr. Williams would elaborate on this subject for us in a future issue - perhaps this Editor will be more careful in his proofing of future issues.

In (GEOS) side

INSIDE GEOS 8A   MASTER BLASTER

Well hello again! This month we will continue the discusssion of text and text handling. Last month we discussed fonts in some depth. This time we will learn about how to print characters and character strings, and how to input them from the user. While we are discussing these routines we will focus on BSW font, later we will add other fonts and discuss the differences.

DISPLAY -- WHERE AND WHEN
Before we discuss the actual display routines there are some general rules that apply to all or most of the routines:

DisplayBufferOn ($2F) controls which screen the text will be printed on. If bit 7 is set the text will go to the foreground screen. Bit 6 controls the backround screen. If both bits are set then text will go to both screens (this is the default condition when an application is loaded). If neither bit is set you are wasting your time!

WindowTop, WindowBottom, LeftMargin, and Rightmargin ($33, $34, $35, $37) are used for clipping. Left and Right are word length (0-319). A character will not print outside of these margins. If a character is on the top or bottom edge then only those portions of it that are within the window will be drawn. If a character tries to print off the right or left side of the window then the routine pointed to by StringFaultVector ($84AB) will be called (if it is non-zero). We'll discuss this vector in more depth later. When an application is loaded these variables are set to the entire screen.

Care must be taken if you shrink the text window when menus are active. The menu driver uses these text drawing routines to draw in the menu options so if it trys to draw a menu outside the text window the text will not be displayed (the menu will function, there just won't be any text in the boxes).

DISPLAY -- THE ROUTINES
The following is a list of the routines available for outputing characters/strings to the screen:

PUTCHAR - $C145 Display a character on the screen.
I_PUTCHAR - $????   The BSW Reference Guide says that there is an inline form of PUTCHAR. I've never found one.
SMALLPUTCHAR - $C202   Lower level PUTCHAR. It's faster but does little error checking.
PUTSTRING - $C148   Displays a string on the screen.
I_PUTSTRING - $C1AE Inline form of the above.
PUTDECIMAL - $C184   Converts a number to a string and prints it on the screen.

All of these routines have several variables in common (execpt of course the inline forms):

R1H - Contains the Y position to start printing text. Note that this is the position of the underline (bottom) of the character(s).
R11 - Contains the X position to start printing text.

Both of these variables will be updated as characters are printed so that when a routine returns they will point to the position where the next character should go.

The only other thing PUTCHAR (and SMALLPUTCHAR) requires is the character to print in the accumulater.

For PUTSTRING R0 will hold the address of the null-terminated string to print.

PUTDECIMAL needs a bit more information. First the 16-bit number to print is placed in R0. The accumulater is loaded with a value which will control printing as follows:

Bit 7- Set for left justify, clear for right.
Bit 6 - Set to suppress leading zeros.
Bits 0 to 5 - The width of the field when using right-justification. Since the number is printed from right to left you must pass a starting position. The number contained in these bits will be added to the value passed in R11 and this will become the right margin . After printing R11 will be updated so that the next character will be positioned properly.

The format for I_PUTSTRING is simple:

.WORD X Postition
.BYTE Y Position (from the underline, remember)
.ASC " <text> "

There are quite a few special characters you can use to change styles, move the cursor, etc. I'm not going to print them all here, see the BSW Reference Guide. Note that on page 107 there is a typo: CHR$(26) turns outline on and CHR$(27) returns to Plaintext. Also see page 393 for more accurate information. One last point, there are several multi-character codes, obviously PUTCHAR can not use them because you can only pass it a single character. While StringFaultVector can be used when printing text it doesn't have as much usefulness when printing known text like this. Later on we'll see how to use it when getting input from the user or printing variable text strings.

THE INPUT ROUTINES
Input in Geos is a bit more complicated then output but it's not too bad once you get the hang of it. We might as well jump in with both feet and talk about:

GETSTRING
This is without a doubt the most misunderstood routine in Geos. First let's take a look at the variables you need to pass it:

R11, R1H - X and Y position for text (these are universal for text routines)

R0 - Address of buffer in which to place text entered.

R2L - Maximum number of characters to accept.

R1L - Flag byte. If bit 7 is set use the address in R4 as the StringFaultVector.

R4 - Optional StringFaultVector.

KeyVector ($84A3) - Address to jump to when RETURN key is pressed.

GetString functions differently from most Geos routines. It is driven by the interrupts. Once you have set up all of the above variables then JMP to GetString. It will handle all of the proccessing, the delete key, etc. until the return key is pressed. Once that happens the routine in KeyVector will be called which will return control back to your application. Note that if you are using KeyVector you will have to disable this (by re-loading KeyVector) and restore it after GetString has executed.

The buffer pointed to by R0 must be initialized before calling GetString. If you wish the buffer to be empty, i.e. the user enters text from scratch, then simply store a zero in the first byte. If, however, there is a (null terminated) string in the buffer it will be displayed when the routine is called and the user can edit it (change a file's name from the DeskTop for an example of this).

BSW's Reference Guide says there is a MaxCharFault vector which will be executed if the user tries to enter too many characters. I have never been able to get this to work (with V1.2 anyway). It seems that if the maximum number of characters is exceded then StringFaultChar will be executed (if it is non-zero). It is quite likely that this feature will function with later Kernal versions (V1.3 perhaps); it is hard to tell -- the BSWRG is notorious for being ambiguous when it comes to which version does what. Experimentation is, as usual, the name of the game!

CHARACTER INPUT

The following routines all deal with various aspects of character input in Geos:

INITTEXTPROMPT - $C1C0   Initailizes the text prompt's sprite (#1) to a bar of the height passed to the routine in the accumulator

PROMPTON - $C29B   Turns the text prompt on and positions it to the position stored in StringX ($84BE) and StringY ($84C0). Note the the Y position is relative to the top of the prompt while text is displayed relative to the bottom of the text. Therefore StringY can't be moved to R1H without some massaging first (add the height of the text to StringY); great forethought on BSW's part.

PROMPTOFF - $C29E   This will disable the prompt. To earse the prompt properly use the following routine:

```
100 PHP; Only if you need to save the flags.
110 SEI ; Kill interupts.
120 JSR PromptOff ; Shut off prompt.
130 LDA #0;              This is done to keep the
140 STA AlphaFlag; $84B4. Prompt shut off.
150 CLI ; Interupts back on.
160 PLP ; Only if you used PHP.
```

GETREALSIZE - $C1B1   This routine will return the height, width, and baseline of a character (in X, Y, and A respectivly) passed to it in the accumulater. The X register should be loaded with CurrentMode ($2E) so the routine can take the current style into account.

GETCHARWIDTH - $C1C9   This routine will return only the width (in A) of the character passed to it in the accumulater. The width will be zero if a control character is passed to it.

GETNEXTCHAR - $C2A7   Returns the next character from the keyboard buffer in the accumulator. Zero is returned if the buffer is empty.

VECTORS

There are two vectors associated with text. The first is called KeyVector. Whenever the user presses a key then the routine whos address is stored in this vector will be executed. If this vector is zero then nothing will happen. The key that was pressed can be found in KeyData ($8504). Since it may take a while to process this key you should always call GetNextChar after it's been processed to see if there are any more keys. If a zero is returned then the keyboard que is empty and you can exit. We'll come back to this a little later.

The other vector is called StringFault. Whenever a character tries to print outside of the text window (horizontally) then the routine whos address is in this vector (if any) will be called. If the vector is empty then the character will not print. SmallPutChar does not check for horizontal boundries at all. It is used when the application is doing it's own error checking. SmallPutChar operates faster then PutChar so there is a speed advantage when you are doing your own boundry checking.

Note that if the character tries to print off the left side of the window then StringFaultVector will be executed. Normally this will only occur if the user hits the delete key and tries to delete off the left side. Unfortunatly delete will not call this vector (only printable characters will) therefore the user can delete outside of your window (makes a heck of a mess). As soon as he or she finally hits a printable key then this routine will execute (too late). Special handling is required to prevent this.

GEOPUBLISH GRAPHICS

BY SUSAN PUHN-LAMB

By now, those of us working with geoPublish know how great it is to be able to take copy we've written and format it into neat columns over a series of pages. The program has even more to offer as far as graphic capabilities. This column will explore all these features, hopefully serving as an exchange of ideas that will benefit all of those doing desktop publishing.

This month I will show how to install a full-page border to be used with a page of type. With geoPaint this is difficult and time-consuming; with geoWrite, it is simply impossible. This method is great for flyers, menus, certificates, signs, etc.

My mother had wanted to send out a short Christmas letter, so I thought I would try to produce it for her quickly. I wanted to use various art clips and perhaps two columns. However, she wanted it all on one page, so a border was all I had room for.

I create all my graphics and borders in the 64 version of geoPaint. The 128 version is more difficult to use for precise drawing, especially in pixel edit. I do use the 128 version for composing a page and moving page-wide photo scraps. The bad news is the 128 photo manager can't be used with geoPublish. The good news is that the wide photo scraps can be used. The only problem this creates is that you must move the photo scraps onto the geoPublish disk one at a time.

To start, I went to page layout and created one large text region that I figured would fit all the text. I had typed the letter with all lines centered using a slightly altered zapf 14 point font. I hadn't created a master page as I wanted to see where the border would appear. I had previously transferred a photo scrap containing the top section of my border in a 128 photo scrap.

I then went to page graphics mode and clicked on the bitmap tool. I positioned the cross-hair as far left and as high to the top of the page as possible.

When I clicked, the whole top of the border appeared in perfect position. I then had to move the photo scrap containing the bottom of the border onto the disk, replacing the top section.

Fortunately, once the art is placed in your document, the photo scrap can be erased. I again placed the cross-hair as far left as possible, but about in the middle lengthwise to make sure the bottom was not cut off. I then clicked the bitmap on and moved it to the bottom. Since I created both photo scraps the same way and positioned the cross hair at the same vertical point, the two segments lined up perfectly. In zoom display, it was easy to draw the side pieces with straight lines and the solid box.

Since I could view the text and border together in page graphics mode, it was easy to see where adjustments needed to be made. In the text editor mode, I adjusted the margins until they were aligned within the border. To make sure the whole document receives the margin changes, highlight all the text first. Shorten lines at the top and bottom to fit the border by using a return after the word where you want the line to end. I was surprised at how easy this turned out to be.

If you only have geoPaint 64, the border will have to be imported in more sections. It would be best to design or use borders that have a top and bottom middle section that can be drawn in. Then you would only need to import the four corners. I will figure out how to accurately import all sections of a more complicated border and explain the method in a future column.

If you have any questions about geoPublish graphics or tips and hints you would like to share, send them to:

Lamb Art & Design
1764 Avenue B
Yuma, Arizona 85364
Or contact me on Q-Link with E-mail to *Yumalamb*. Merry Christmas!

Dear Friends:

The Fresno Bee ran an article about the trials and tribulations of sending Christmas cards. Taking a quote from the article, "I was too fatigued, flustered, and mentally incapacitated to send out personalized Christmas greetings, so we're sending one of those delightful, darling, dreary form letters that everyone is so fond of receiving".

We love this time of year—especially hearing from all our old friends. At least we can touch base again and remember all the good times we've had together when our lives touched yours. Each one of you had a positive influence on us and we have happy memories of your friendship.

The year of 1987 passed much too quickly—we did manage to take a few short trips in our car. Walt doesn't want to camp with the van anymore, so we now go the "motel route". We do miss camping—I loved our camp sites; early morning coffee by a stream or under tall trees; the fun of eating home-cooked meals and not waiting for someone to cook it.

Walt takes his weekly trips to the mountains with his friends. On his day out, I visit thrift shops and kick up my heels. I also spend a lot of time gathering and pressing flowers for my cards and stationery. This year I decided to make up some of my long forgotten jelly recipes. I got hungry for the taste of wild gooseberry and elderberry. Since we have a pomegranate tree in the back yard, I put up a lot of my pomegranate jelly.

We have a new love in our life! Her name is "Joy Noel", a four month old miniature poodle. She runs circles around us and keeps life interesting.

We heard from our friends in Thika, Kenya today. John Muthamia honored us by naming his first son, "Puhn Muthamia". John went to college here in Fresno for four years. Through Rotary club we became hosts for his stay in our country.

Our four children continue to be happy and healthy—the grandchildren are growing like weeds. I never thought I would write one of these letters, but time is of the essence. Daughter Susan offered to type this letter with her computer, so we shall see how it comes out.

We will be at daughter Vivian's home at Bass Lake for Christmas again this year. We hope you stay well in 1988 and maybe we'll see you sometime.

GEOS and the STAR NX-10
How to Get the Most Out of Your Printer
By Ken Slaughter

There are a lot of printers out there that are supported by GEOS, and a few of these have more than one driver available. Thanks to two GEOS users, Terry Mills (TerryM13) and Fleximan, the most notable of these printers is the STAR NX-10. GEOS can severly limit the printing capabilities of even the most expensive dot matrix printer. However, if you own a STAR NX-10, you have one of the most versatile printers available for use within the GEOS environment.

When printing a GEOS document, the printer can only follow the instructions that the printer driver gives it. There is no way to directly send a command to the printer to make it do different things, such as condensed or expanded text and graphics, or print in different densities. The interface should be in Transparent mode for GEOS to print properly, so the panel settings won't do any good. The only way to make your printer behave differently is to use a different printer driver.

There are several print drivers written specifically for the NX-10, other than the 'basic' BSW driver. And since the NX-10 is Epson compatible, drivers that are written for the Epson FX-80 printer can be used with the STAR NX-10 with suprising results. It is also important to note; most, if not all, Epson compatibles can also use the STAR NX-10 drivers as well, some with varying results.

To give you an idea of just how astounding the results can be, consider this for a moment:
This entire article was composed with geoWrite 2.0 and printed on the NX-10 (this page, fonts only - no photo scraps, and printed with quad dbl.prt.2 driver). It was then offset printed for use in GEOWORLD. None of it was prepared with the LaserWriter! Obviously, the only way to obtain true professional results with GEOS is by utilizing the LaserWriter, but it should give you a few more ideas about what can be accomplished with your printer.

All of the drivers listed have been made available by the authors for you to download from the QuantumLink computer network. Their work involved in developing these drivers and making them widely available is very much appreciated.

Specifically for the STAR NX-10:
These drivers actually expand or compress the graphic line length.
From Fleximan –
star nx-10 60a – Expands the line length to allow emulation of a 60dpi printer.
star nx-10 72b – Expands the line length to 72dpi, to allow proportional printing of Macpaint /geoPaint converted graphics.
star nx-10 ddc – Compresses the line length to double-density 120dpi.
star nx10 ddhd – High-speed double-density 120dpi.
star nx-10 qde – Compresses the line length for a quad density pass of 240dpi.

For Epson compatibles:
These drivers do not change the 80dpi line length, they only change the density of the print.
From TerryM13 –
epson dbl.prt – Makes a second pass after a 1/216th of an inch (or 1/3 dot) line feed for double density printouts.
**quad sgl.prt.2** – Uses the printer's own quad density mode. Single pass.
quad dbl.prt.2 – Uses the printer's quad density mode. Also makes a second quad density pass for very dense printouts.

In order to get the most out of your printer, you will also need a copy of "change printer" by Cosmac (also available on Qlink). With it, you can change printers from within an application, without having to exit to the deskTop to use Select Printer.

You will find the NX-10, when coupled with a good interface such as the Xetec Super Graphix, is the most cost effective and versatile printer available for use with GEOS.
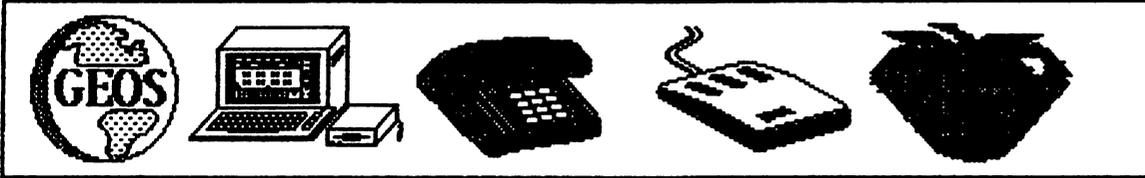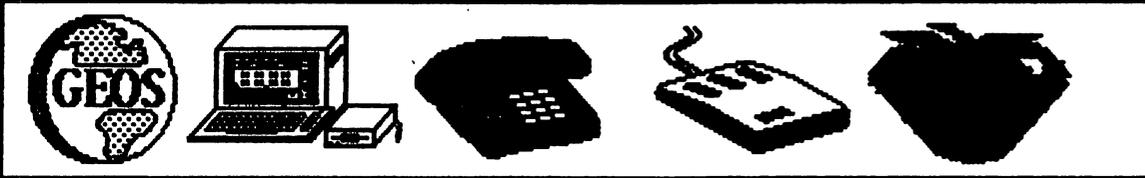
Variety of Printouts on the STAR NX-10
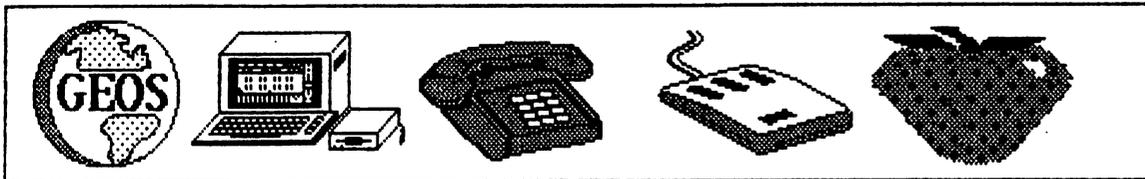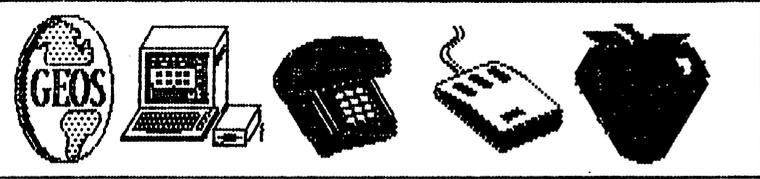

Star NX-10
80 dpi

(Regular Driver)
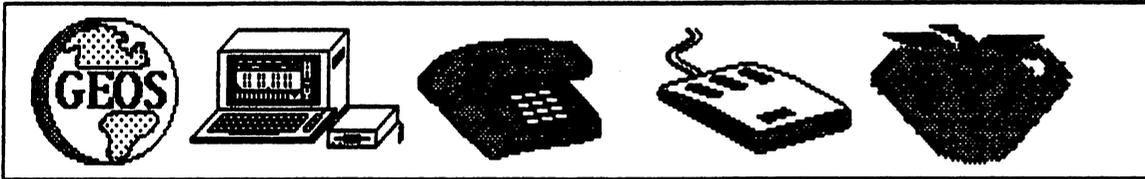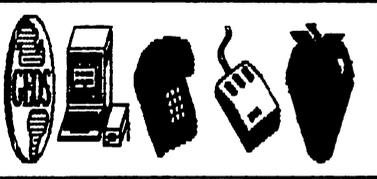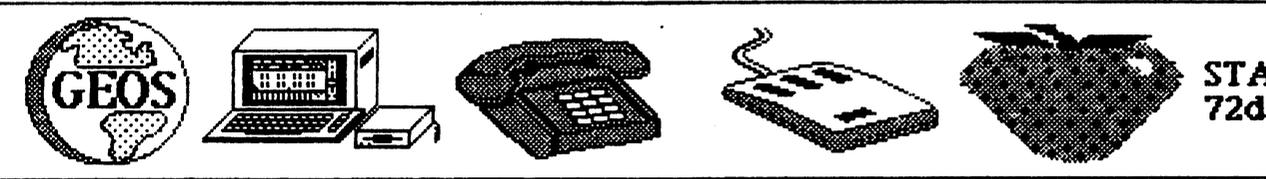
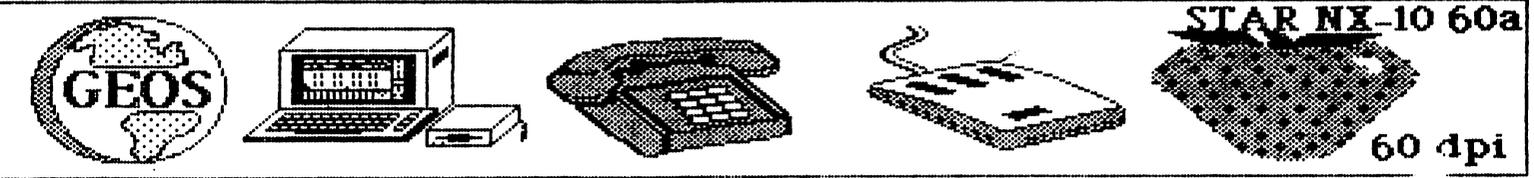EPSON DBL.PRT

QUAD SGL.PRT.2

QUAD DBL.PRT.2

STAR NX-10 ddc

STAR NX-10 ddhd

STAR NX-10 qde

STAR NX-10 72b
72dpi

STAR NX-10 60a

60 dpi

STAR NX-10 vs. QUAD DBL.PRT.2

STAR NX-10

BSW 9 Point – Different fonts and sizes have different qualities.

ROMA 14 Point – Different fonts and sizes have different qualities.

Elmwood 18 Point – Different fonts and sizes have different qualities.

Script 16 Point – Different fonts and sizes have different qualities.

Ormand 12 Point – Different fonts and sizes have different qualities.

STADIUM FontKnox

QUAD DBL.PRT.2

BSW 9 Point – Different fonts and sizes have different qualities.

ROMA 14 Point – Different fonts and sizes have different qualities.

Elmwood 18 Point – Different fonts and sizes have different qualities.

Script 16 Point – Different fonts and sizes have different qualities.

Ormand 12 Point – Different fonts and sizes have different qualities.

STADIUM FontKnox

Because of space limitations, shown above are only two examples of the 80dpi (full line) options. Some fonts and sizes will work better with different drivers. The other two Epson drivers not represented here both work well with most fonts. The driver you choose will depend on the font you choose. And, by using the STAR NX-10 driver, you get NLQ that looks like this.

There are times when you may want to send just basic, readable correspondence, but you still want it to standout. You can use one of the Epson drivers to obtain this type of "enhanced" NLQ. This is particularly useful when the normal STAR NLQ doesn't seem to look just right for the letter, and your printer ribbon can no longer handle the double and quad density graphic fonts.

STAR NX-10

EPSON DBL.PRT

QUAD SGL.PRT.2

QUAD DBL.PRT.2

The type of graphic you want to print, and the final results that you achieve, can be dramatically influenced simply by choosing a different print driver. With the creative use of the overlay technique, proper choices of drivers and types of graphics, you can achieve truly impressive results. Many applications not requiring laser printing, such as neighborhood fliers and informal newsletters, can be produced from start to finish at home on your personal computer, without that "home computer" look. One thing to remember, always keep a fresh ribbon handy for the best results. Do all of your experimenting and layouts with one ribbon, and then pop in the fresh one for the final printout. A lot of quad density printing will wear out a new ribbon in a hurry!

Happy Printing!
Ken Slaughter (on qlink – FAT BEAR)

Is everybody having a great holiday season? We sure hope so. This has turned out to be a fantastic time for all of us here, and a busy one. Kind reviews and mentions in several national publications, along with our own advertizing mailings, have generated an un-anticipated response for us and for GEOWORLD. We've all had to work full-time to make sure everyone's orders arrived by Christmas.

Because of the response, we've decided to extend our special membership package through February 1st. You can still get Tim's Timer, Mystic Jim's membership, subscription to GEOWORLD, access to our BBS, all 13 disk sides, printed MJ documentation; Alex Boyce's GEOS Tech Manual. This a hundred-dollar-plus value for just $74.95.

Several people have had problems assembling their Tim's Timer kits, and we've had to replace damaged chips. In addition, some Timers and other mailings have been lost in the mail.

To solve these problems we're making some changes:

1) All RTCs will be sent UPS, so we can at least find out where the packages were lost.

2) Our mail labeling procedure will be changed. In the past, we sometimes affixed the actual address labels onto our standard yellow label. Unfortunately, the yellow coloring is a wax-type ink. You guessed it: the address label could come off during postal handling, which can be less than gentle. Of course, our return address is printed on the yellow lable, which can't fall off, but the stuff was never returned to us. Where did it go?? From now on, we'll PRINT addresses right on the yellow label.

3) From now on, we will not offer any external kits for RTCs, only finished devices. Internal installation RTCs will be fully assembled; for the 128, only one wire will be attached by the user, and that wire will have a spring-lock to attach it to the computer. The internal 64 RTC will be fully assembled, but six wires will be connected by the user.

Some BBS users are still having problems with the "contribution factor". Our BBS will no longer use the "contribution factor" to limit access to the libraries. We still hope that users will continue to contribute up-loads or post messages in the bulletin boards, but that will no longer affect access to down-loads.

Speaking of contributions, they've really dropped off. We're sending more shareware to non-members than ever before, including many who request additional disks, but most don't make contributions for the use of the programs. Perhaps, since we now offer memberships, folks think we don't expect contributions by non-members. Of course, many of you are generous far above the call of duty, and we just couldn't exist without you. But it just doesn't seem fair for you to support those who don't contribute at all. I hope this doesn't seem critical or judgemental, because it's just a misunderstanding, and I know that just these words are enough to clear that up. This work is very important to me, and I want to be able to afford to keep doing it. See, I had this dream where the Head Mystic in the sky said I'd be taken away unless you all contribute 8 million dollars by March....

By the time you read this, the first Mystic Jim Satellite BBS will be running in Ontario, Canada. It's run by Bill McKibbon; we don't have the phone# yet, and that'll be posted on our BBS when we do. All members will have access to satellite boards. There is still some bug-squashing to be done before we set-up other boards, and we'll be contacting those who've expressed interest in running a satellite board.

There's some neat new stuff in the Update page, so look it over.

Best wishes for the new year!
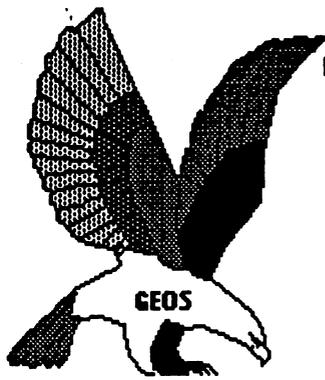
Mystic Jim's Shareware Update

December, 1987 Volume 1 Number 8

DISK#1 New Program Updates.
If you've received disks 4,6 and "R" after Sept 1, you have all of our programs. Programs added after Sept. 1 will be on disk#1 and on the BBS for downloading.

DISK#2: INTRODUCTORY DISK
This is the first disk we send to new friends on their first request. If they like this disk, they can request others, or join our shareware group.
 Bootstraps 1.2, 1.3U, 1.3D
 (bootdisk makers) (7)
Input drivers:
 KoalaPad: eight kinds (3,7)
 Lightpens: ALL kinds (7)
 SketchPad. (1)
 Mouse 1351
 Mouse1351, port#2. (7)
 Mouse1350.
ACCESSORIES
 GEODump: screen printer (3)
 Change Input (3)
 Change Printer (3)
 GEOFormat1.3: fast format(3)
 Note Printer (3)
 Auto Preference1.2, 1.3 (3)
 Watchset. (2)
 Quick Dateset (4)
 Icon Maker (7)
 Sprite Magic (x)
 Icon Edit2.1 (11)
 O'Clock: display time (7)
 Convert1.4: ALL types (9)
 GEOTab: for GEOWrite. (1)
 TextGrab1.3 (6)
 GrafGrab,Dex,Merge Patches.
GRAPHICS.
 Imagecon. (7)
 Piconvert: Doodle-GEOS (x)
 Computer Eyes: for GEOS
 GEOPaint Help Files (7)
DISK#3 FONTS
Balloon 24pt
Banner 36pt.
Dvorak10pt
Esperanto6,10,12,14,18,24pt
Greek 12pt.
IBM 10pt.
Fill-48 48pt.
MegaBanner 280pt.
Mystic 18pt.
Oxford 12,24pt.

Paintbrush 36pt.
Zapf 14pt.
Andrea's Calligraphy
Star Trek 48pt.
Additional fonts added regularly. Fonts designed by Cosmac, P. Hughes MysticJim Andrea Needham and others.
DISK#4 Programming Tools
(members only)
 Convert,mL: ML to GEOS (5)
 GEOStart. (1)
 DISPMEM:display memory. (15)
 GEOThreader. Single Step Sim.
 GEOLinker: header linker. (2)
 GEOSplitter: header splitter (2)
 Disk Editors 64 and 128
 Micromon 64
 Spelunker:display memory.(10)
 Techman Updates: fixes OGPRG
 Alex Boyce's Tech Manual!!
 GEOHACKER: Strips copy- protection from ALL GEOS versions and add-on programs, and edits serial#: EVEN GEOS128!, GEOCalc, GEOPublish, new GEOFile. (version 1.6) (7)
DISK# 5: Terms and utilities for Mystic Jim's BBS (members only)
DISK#6: Graphics Converters
 PS-GEOS: all Print Shops(7)
 PS/NewsRoom/PrintMaster
 Imagecon: hi-res/multi-GEOS(7)
 Piconvert: Doodle-GEOS
 GEOVERT: Converts GEOS to hires/multicolor and back (7)
DISK #7: Printer Drivers
(of interest to user groups)
 All GEOS printer drivers, including laserwriters, GEOCable, and Mill's double/quad drivers.
RANDY'S ROUNDUP: A disk full of Randy Winchester's best stuff
TIM'S TIMER: Operating system and utilities for Tim Corcoran's Real Time Clock for Commodore64, Commodore128, GEOS, time and date.
New internal plug-in version for the 128.
Now GEOS128 compatible!!
 Auto-Read : Reads RTC at boot.
 RTC-Write: Set time and date.
 Show Time 1.2 Display time
 Show Time All: for 1.0, 1.2, 1.3
 Reminder: reminds of special times or dates. Never forget another anniversary. BASIC Time./display
 * COMING SOON:
 * Hack'n GEOS. All of the programs that we use to get into GEOS
 Disk Digger. Seek, find, alter code by Seq, VLIR, T&S. Decimal, Hex, or Chr$tring

GEOS-Mon. Full-featured machine language monitor. DA: Drop into mon. from anyplace in GEOS.
Discompare: See what those "installations" really do, by VLIR, Seq, T&S
* **GEOFontastic.**
Fontmaker. Fastest, easiest font designer available at any price, from any source. Makes up to 128 point fonts.
Fontgrabber: Easily convert any font to GEOS.
Fonts: new fonts and megafonts just to get you started.
GEOGraphics.
 Lots of neat new clip-art.
GEOPUB/GRAF (members only)
 Extended editing for GEOPub files, including full color, pixel editing for text and graphics, move cut copy area BY SINGLE PIXEL., and more. by MASTER BLASTER/Mystic Jim.

Programmers: semi-alphabetic
1. Alex Boyce 2. Bill Coleman
3. Art Dahm. 4. Jeff Fox
5 Don Mosedale
6. Randy Winchester
7. Mystic Jim 8. Bruce Moore
9. Ben Taylor 10.Tony Reynolds
11. Terry Mullett 12. Mitch Romm
13. Tim Corcoran 14.Terry Mills
15. Lynn Kerby. 17.Andrea Needham
16. James Hastings-Trew

To order disks, membership, hardware, or for more information, send mail, or use our BBS.
Mystic Jim, 2388 Grape
Denver, CO 80207
(303) 321-8954

MYSTIC JIM ENCHANTS THE EAGLE

PUTTING IT ALL TOGETHER

Ok, we now how enough information to put together a text input routine. Here is an outline of the steps required:

1. Initialize the text prompt.
2. Initialize the buffer for text entered.
3. Initialize StringX and StringY.
7. Call PromptOn.
5. Load StringFaultVector.
6. Load KeyVector.

When the text has been entered (by hitting Return, etc.)
1. Call PromptOff
2. Zero KeyVector.
3. Zero StringFaultVector (optional, it won't get called).

That's about all that there is to it! It seems like there is alot to do but it really isn't that hard if you only need a line or two. It does get a little hairy when you start getting into the more complicated aspects such as using the cursor keys, etc. Did you ever notice that the INFO text box from the DeskTop won't print if the DeskTop isn't on the disk? DeskTop is a VLIR file. Record #0 is the main program and record #1 is the routine to handle this box. The text handling features are rather simple (no cursor keys though you can postion with the joystick) yet the record is almost 2K long!

Let's try an example. Assume we are dealing with BSW font only and that we need a routine to get two lines of text from the user. Our buffer is located at $4000 and, curiously enough, is called BUFFER. The routine to get each keypress is called INPUT and the one to handle string faults is called OOPS. For this example we will use the following equates:

```
100 LEFT .WORD 0 ; Left margin (set by calling routine)
110 WALL .WORD 0 ; Right margin (ditto)
120 TOP .BYTE 0 ; Starting Y position
130 SIZE = 9 ; Height of a character
140 BUFFER = $4000 ; Text buffer
150 LINE .BYTE 0 ; Holds the line cursor is on (0 or 1)
160 EDGE .WORD 0 ; Holds the last position on line 0
170 SEAM .BYTE 0 ; First character position in line 1
180 POSITION .BYTE 0 ; Holds current index into buffer
```

Now for the initialization routine DOINPUT. I am assumming that the margins have already been set by the routine which calls it. This allows the same routine to be used regardless of where the text window is.

```
500 DOINPUT LDA #0 : STA buffer ; empty text buffer
510 STA line : STA position ; start at line zero, position 0 of buffer
520 LDA #size : JSR inittextprompt
530 LDA top : STA stringy ; position the
540 LDA left : STA stringx ;   text
550 LDA left+1 : STA stringx+1 ; prompt
560 JSR prompton ; Now turn it on
570 LDA #>opps : STA stringfaultvector ; now we set the vectors
580 LDA #<opps : STA stringfaultvector+1
590 LDA #>input : STA keyvector
```

```
600 LDA #<input : STA keyvector+1
610 RTS ; now we exit. The MainLoop will take care of the rest.
```

So far so good. Initialization is straightforward. First we zero the variable that need to be zeroed, then we set up the text prompt, and lastly we load the appropriate vectors. Note the RTS at the end, this is very important. We must return to the Main Loop or INPUT will never get called. Speaking of which...

```
700 INPUT LDA keydata ; get the first key code
710 INPLOOP CMP #13 : BEQ we-r-done ; return was pressed
720 CMP #8 : BEQ delete ; delete was pressed
730   ; Check for any other special keys here
740 LDX position ; get index into buffer
750 STA buffer,x ; put character in buffer
760 INX : STX position ; update position for next char
770 PHA ; save character
780 CLC : LDA stringy : ADC #height : STA R1H ; set up for print
790 LDA stringx : STA R11L
800 LDA stringx+1 : STA R11H
810 PLA ; retrieve character
820 JSR putchar ; now print it
830 INP1 LDA R11L : STA stringx ; update prompt's position
840 LDA R11H : STA stringx+1
850 JSR prompton
860 JSR getnextchar ; any more characters?
870 CMP #0 : BNE inploop ; yes, process
880 RTS ; We're finished until next key is pressed.
890 ;
900 WE-R-DONE LDX position ; get index
910 LDA #0 : STA buffer,x ; null terminate buffer
920 STA keyvector : STA keyvector+1 ; disable INPUT
930 JSR killprompt ; use the routine I gave earlier to shut off prompt
935 ; We're done! Jump to the routine to process the string. You can
936 ; also use a local vector for this so that you can use the routine
937 ; in many places in your application.
940 JMP ???? ; or JMP (????)
950 ;
960 DELETE LDX position : BEQ DFINI ; No chars to delete
970 LDA buffer-1,x ; get last character printed
980 JSR getcharwidth : STA lastwidth ; set width to erase
990 LDA #8 : JSR putchar ; delete
1000 LDX position
1010 CPX seam : BNE DEL1 ; Check if deleted 1st char on line 1
1020 LDA #0 : STA seam : STA line ; back to line 0 and clear seam
1030 LDA edge : STA R11L
1040 LDA edge+1 : STA R11H
1050 DEL1 DEX : STX position ; update index
1060 JMP INP1 ; done ... process next character
```

Whew! That was a mouthful! This routine should be self-explanatory but here's a few comments. Note that once GetNextChar returns zero we simply RTS. When the next key is pressed INPUT will again be executed. The variable POSITION keeps track of where we are in the buffer. Under most conditions only one character will be retrieved for each call to INPUT. When the user finally hits return then we simply store a null in the buffer

after the string, disable KeyVector so it won't be called again, and jump either directly or indirectly to the routine that will process the string.

This is the main reason why the routine given in BSW's RG won't work properly. Unless the user can enter the whole string in a couple of milliseconds all that will happen is the buffer will put each character in the first position of the buffer (the index is set to zero every time the routine is called)!. Also the return key won't shut off KeyVector the way it should and there isn't any way to call the routine which will process the string, i.e. an endless loop.

The delete routine needs a bit of explaination. Since we are dealing with multiple lines of text we must have a way of deleting back to the previousline (remember chr$(8) will not callStringFaultVector). As we will see OOPS will store the current text position on the screen and the current index into the buffer whenever it is called, i.e. when we move from line 0 to line 1.

After the delete routine deletes a character it will check to see if the current index (before it gets decremented) matches the one saved by OOPS. If they are the same it then we have just deleted the first character in line 1 so load R11 with the value OOPS stored in EDGE. Note that you can't simply load R11 from RightMargin; the last character will probably not end exactly on the margin. Delete will then store a zero into SEAM and LINE. Clearing SEAM is necessary so that delete won't try to use it again (let's say you delete a capitol W then type ii - SEAM will match again even though you are still on line 0). Notice that the entry point back into INPUT (INP1) will update the text prompt so it will now be at the end of line 0.

Ok, the last routine we have to look at is OOPS. This routine will get called by PutChar whenever there isn't enough room to print the character without crossing RightMargin. Upon entry to this routine the character that needs to be printed will be in the accumulator (athough you could retrieve it with LDA buffer-1,position).

```
1100 OOPS LDX line : BNE OPPSFINI ; already on last line
1110 INX : STX line ; now line = 1
1120 PHA ; save the character
1130 CLC : LDA stringy ; move down a line
1140 ADC # (size+2) : STA stringy
1150 LDA stringx : STA edge ; save current position
1160 LDA stringx+1 : STA edge+1
1170 LDA position : STA seam ; save index into buffer
1180 LDA left : STA stringx ; back to left margin
1190 LDA left+1 : STA stringx+1
1200 PLA ; retrieve character
1210 JMP putchar ; print character and exit
1220 OOPSFINI DEC position : RTS
```

This routine shouldn't be hard to figure out by now. Note that if LINE = 1 when this routine is called, i.e. we are trying to move to a third line, then POSITION will be decremented. This has the effect of deleting the character just typed from the text buffer. Unless the user deletes a character or enters a character that's small enough to fit on the current line no futher characters will be accepted (except return of course).

This example can easily be expanded. For instance if you want to input more then two lines simply make SEAM and EDGE tables rather then individual variables. Then simply use LINE as an index into the tables (remember that EDGE is word length, you'll have to do an ASL before indexing). Adding cursor control becomes more complex because you can't simply move the cursor up or down (the characters are proportional, remember)? Also keep in mind that if the cursor is positioned in the middle of a line of text then the entire line will have to be reprinted every time a character is entered.

Allowing the user to position the cursor using the mouse presents it's own problems. Since the width of each character is different there is know easy way to know what character the user is pointing to by mouse position alone. Probably the easiest way to accomplish this is to compute the line of text that the user is pointing to (which is easy enough unless you allow variable point sizes) then execute a loop that starts at the first character of the line and repeatedly calls GetCharWidth. When the sum of the widths (plus LeftMargin) becomes greater then the mouse's X position just force the mouse to this position and update the current buffer index accordingly. Of course you will have to make sure that the mouse is within the text window before you even start. Of course if you have memory to spare you could always set up a table that contains the starting value of R11 and R1H for each character entered. Now when you need to located a character you just index into the table using POSITION.

I must admit that I cheated a bit when I wrote GeoHeader. The font that I wrote for this program is non-proportional, i.e. all the characters have the same width (in this case 7). Now text handling becomes much easier because it is easy to tell where any giving character is within the window. This allowed me to add many more text handling functions without the code getting overly long. And when it comes right down to it you don't need proportional fonts for general utility programs anyway.

By the way if all you need is single character input without any output (such as parsing commodore key commands) You can do away with most of this. Simply have INPUT check which key was pressed and execute the proper routine to handle it. Before calling the routine it's a good idea to call GetNextChar in a loop until it returns zero. This will prevent extra keypresses from appearing.

USING OTHER FONTS
The previous example applies just as much to other fonts as it does to BSW. The one thing you will have to do is use GetRealSize instead of hardcoding the character height as I did in the example (although if you are only using one point size at a time you could store the height in a variable and let the text routines use that). Style changes can be a problem while inputting text because while each style change would have to be imbedded in the buffer it won't print on the screen. For most purposes it won't be necessary or even desirable. In fact during input you should parse out control characters (the example doesn't do this but it would be easy to add just after the check for the delete key, CMP #32 : BCC).

Well that's about all I have room for this month. Next time I'll have some more goodies for you. MB

Improving the geoPublish Tutorial

by Marte Brengle

Like a lot of people, when I got my new geoPublish program I was so eager to use it that I started out with the Tutorial before I'd read all the instructions carefully. And while the tutorial will work that way, it doesn't work to the program's best advantage.

Here's some suggestions on how to go through the geoPublish tutorial and get the best out of it.

Follow the directions as given on page 3-2 and 3-3. BSW already sketched out the idea for the newsletter, so you don't have to do a sketch of your own. If you have a REU, create your "sample work disk" using the REU as a RAMdisk (I used it as a RAM 1571) and you'll see an incredible speedup in the program. Plus, when you're finished, the sample document is in the REU, and if you don't want to keep it around, you won't have to erase it from a "real" disk.

The "text files" that are provided on the disk are already formatted for the columnar layout that the tutorial uses. The files are in LW Roma font, though, and the manual doesn't note that you have to have that font on the disk with your document. If you don't copy this font over to your work disk, you'll find that the program re-fonts your file in BSW 9 point. I'm not crazy about the Roma fonts in general, and the LW Roma looks terrible on a dot matrix printer, so I put the LeConte font on my work disk. I happen to like the look of that font, but you can choose any font that appeals to you. Just make sure it's on the same disk as the sample newsletter you're creating.

I found that setting the guidelines precisely wasn't always easy. You can watch the measurement change in the "document ID box" to set the markers at a precise distance. Sometimes the distance would measure out exactly, but sometimes my pointer would jump from a measurement that was too small to one that was too big, with never a stop in between. Just keep fiddling with this till it works right.

The manual's directions for creating a "running header" with the date in it omit one important piece of information. The date will be displayed in a rather large font, and you either have to make sure that the boundary box for the header is the proper size to contain it all, or you have to use the "attributes" function to re-size the font. (See page 4-36) The same applies to the footer.

Draw the Master Page as directed, and enter Layout mode as directed. Follow the directions for creating a text region and placing the text -- with this exception. If you follow the directions as given, you'll have two articles starting on the first page and continuing onto the second page, which makes them rather difficult to read. So after you've created your first text region on the left side of the page, and placed text file 1 into it, create the second text region as directed. But don't install text file 2 into it. Click on the T, open text file 1 again, and put the remainder of this file into the second column. It won't fill the second column completely. So after your text is rippled in, you can create a graphics area in the area not filled by the text. Or you could put the "photo scrap" graphic (which doesn't have a whole heck of a lot to do with the articles, by the way) somewhere on the page, as directed, and add your own graphics wherever you desire.

Follow the same procedure for putting text file 2 on page 2. You can create short text regions as indicated in the manual, but the entire article may not fit. Try creating whole-column text regions, rippling in the text, and then adding graphics.

If you follow the directions for putting in a headline, you'll end up with a headline that doesn't look very good, all the more so because you weren't given directions to put the LW Roma font on your work disk to begin with. If you do the headline in LW Roma, create it as the manual directs, but before you leave the "attributes" dialog box, be sure to click on the "smoothed" box. This will improve the appearance of the letters considerably.

If you create the document with the margins specified in the manual, you'll find that your left text region looks misplaced when it's printed out. You may want to change the margins to be more even, or else have the program print vertical lines along the left and right margin guidelines to show that the text is supposed to be formatted in that manner.
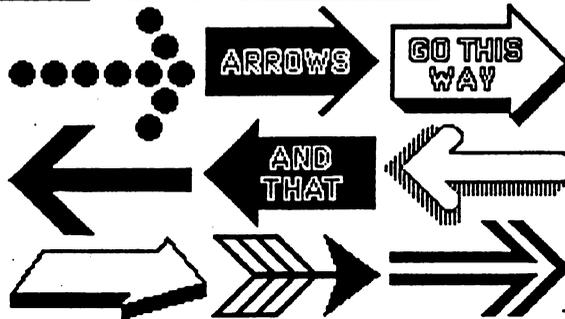
Once you've read through the manual, you'll find that there are a lot more possibilities than the tutorial can cover. But if you just want to get a good idea of what this program can do, and how BSW has once again transformed the C64, try printing out "The Jelly Roll" according to the manual and these suggestions and I believe you'll be very favorably impressed.
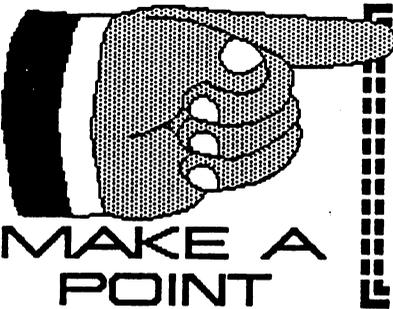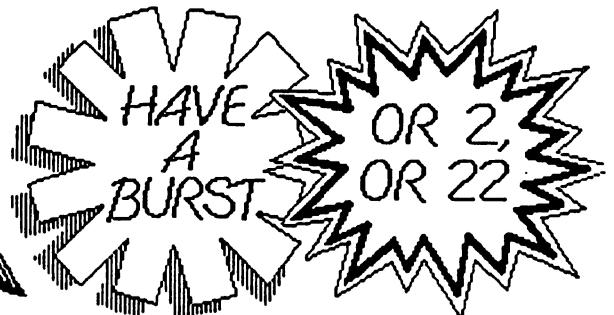
A Word from BSW

Rob Siegel

The Latest Official Word from Berkeley Softworks

Updates

The following upgrades are available from Berkeley Softworks if you own the old version:

GEOS 128: If you own **GEOS 64,** the cost is $22.00 + $2.50 shipping and handling. (Send back **GEOS** package sleeve).

geoWrite 2.1: If you own **geoWrite 2.0,** send $15.50 + $4.50 shipping and handling. Include **Writer's Workshop** disk. The 2.1 version has the ability to expand the margins to a full 8.0 across your page.

geoWrite Workshop 128: If you own **Writer's Workshop,** send $20.50 + $4.50 shipping and handling. Include **Writer's Workshop** disk. This is the 80 column, 2.1 version of the above program.

geoFile: No cost for version 1.0 owners. Send original 1.0 disk.

geoFile 128: If you owm **geoFile,** send $20.50 + $4.50 shipping and handling. Include geoFile disk. This is the 80 column version of geofile. (Available in December)

geoCalc 128: If you own **geoCalc,** send $20.50 + $4.50 shipping and handling. Include geoCalc disk. This is the 80 column version of geoCalc. (Available in December)
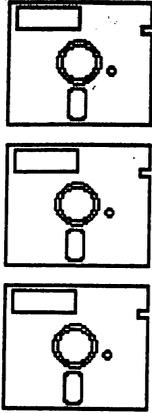
Send money and disk to:
Berkeley Softworks Update Offer
2150 Shattuck Avenue
Berkeley, CA. 94704

Happy Holidays from all the folks at Berkeley Softworks

Next Month here from BSW:
An Official inside review of
geoCalc and geoCalc 128!