NEWSNET99ER

Newsletter of the NET99ER TI-99/4a and Geneve 9640
Computer Users Group

Vol 10 Num 10 & 11   November 1992

NEXT MEETING:

SATURDAY

DECEMBER 5th

9:30 AM at the North Richland Hills Community Center
Loop 820 at Rufe Snow Drive

CLUB OFFICERS

Barbara Massey ————————————————————— President
Jim Lesher ————————————————————————— Vice Pres
Cal Koether ———————————————————————— Treasurer
JoNell Thompson ———————————————————— Secretary
Barbara Massey ————————————————————— NL Editor
Tom Collins —————————————————————— BBS SysOp
Jeff Drinan ——————————————————————— Librarian
Bill Duncan ———————————————————————— M/S Chrmn

Call the

NET99ER BBS
300/1200/2400 BPS
7E1 - 24 HRS
237-3409

FORMATTER "CRIBSHEET"

Text Dimension commands, as the name lies implies, move or shape the words in the document (margins, linespacing, right justify, etc.).

```
.FI       : FILL      :PUTS AS MANY WORDS ON A LINE AS WILL FIT.
.NF       : NO FILL   :CANCELS FILL.
.AD       : ADJUST    :ALIGNS THE TEXT TO THE LEFT AND RIGHT MARGINS.
.NA       : NO ADJUST:CANCELS ADJUST.
.LM n     : LF MARIGN:SETS LEFT MARGIN TO "n".
.RM n     : RT MARGIN:SETS RIGHT MARGIN TO "n".
.IN       : INDENT    :CREATES AN AUTO-INDENT FROM LEFT MARGIN.
.LS n     : LINE SP   :SETS LINE SPACING TO "n" LINES.
.PL n     : PG LENGTH:DEFINES NUMBER OF LINES TO A PAGE.
.BP       : BEGIN PG :DEFINES FIRST LINE OF NEW PAGE.
```

Internal Format commands control the spacing of characters on a line.

```
.SP n     : SPACE     :SIMILAR TO TAB FUNCTION.
.CE n     : CENTER.   :CENTERS NEXT "n" LINES BETWEEN MARGINS.
```

Highlighting commands control functions such as underline or bold and allow you to redefine characters to use them to send CRTL codes to the printer.

```
^         : REQUIRED :JOINS WORDS TOGETHER WHEN REQUIRED TO PREVENT
          : SPACE     :SPLITTING IN REFORMATTING, UNDERLINE, ETC.
&         : UNDERLINE:UNDERLINES ALL TEXT FOLLOWING UNTIL NEXT PAGE.
@         : BOLD      :RETYPES FOLLOWING TEXT FOUR TIMES.
.TL xx    : TRANS-    :ALLOWS REASSIGNMENT OF ONE CHARACTER TO REPRESENT
            LITERATE :A NUMBER OF CHARACTER VALUES TO SEND CODES TO THE
                      :PRINTER.
.CO t     : COMMENT   :SIMILAR TO REM IN BASIC--ALLOWS NOTES THAT DON'T
                      :PRINT.
```

Page identification commands print notes in the upper or lower corner of each page, either headers or footers.

```
.HE t     : HEADER    :PRINTS TEXT (t) AND PAGE NUMBER AT TOP OF PAGE.
.FO t     : FOOTER    :PRINTS TEXT (t) AND PAGE NUMBER AT BOTTOM OF PAGE.
.PA       : PAGE #    :RESET PAGE NUMBER IN .HE AND .FO
```

File management commands.

```
.IF f     : INCLUDE   :MERGES A FILE TO PRINT A DOCUMENT TOO LARGE FOR ONE
                      :FILE.
```

Mail Merge commands are used to supply values to the variables in a letter that has been set up for the mail merge option.

```
.ML f     : MAIL LIST:IDENTIFIES VALUE FILE ((f) FOR MAIL LIST.
*n*       : VARIABLE :INSERTED IN TEXT AS VARIABLE FOR REASSIGNMENT FROM
                      :VALUE FILE.
.DP n:t   : DISPLAY   :PROMPTS YOU USING TEXT "t" TO ASSIGN TO VARIABLE
                      :(*n*).
.         :  PROMPT   :
```

Last month we did not have a Newsletter. Due to hardware problems, I have been unable to get Jack (my computer) up and running. I even spent two unsuccessful evenings with JoNell, hoping to be able to get one out. However, I guess my "touch" beat me over there. Even with the two of us we were unable to get a word processor program to work. I called Tom Collins but poor Tom Collins has been under the weather, and I mean the weather has stomped all over him! I got in touch with James Crosson and HOPEFULLY he will be able to come over sometimes during the holiday and get Jack up and going, and get JoNell up and running.

Due to the fact that Jack is not working, I have had to modify this Newsletter. Hopefully Jack will be back soon!

By the time you read this, Thanksgiving is over and done, just a memory and a few pounds. Christmas is just around the corner and in a blink of they eye, it is 1993! The new year brings tax season, and for me the month of February is one long day. Working 7 days a week 10-14 hours. I am going to need to have someone help get the Newsletter out. January, February and March will be completely tied up with work. So, don't everyone volunteer at once, but someone, please do.

I spoke with Jim Lesher today, and he told me to be sure to let everyone know about the DOM. It is the latest, newer, better than ever, DM1000 v6! A must for us all. Also,we have a program that will make borders on mailing labels. As a bonus we also have a calendar program.

I was hoping to be able to have this out in time to let everyone know well in advance about the upcoming meeting on December 5th so that we might be able to have a mini party...an informal get together with a few snacks, but since we are not allowed to have food and/or drinks at the center, I guess we had better forgo the munchies. I do hope to see some faces that haven't been seen for awhile.

Remember that the January meeting will be held the SECOND Saturday, January 9th. We will be meeting Saturday, December 5th, in the North Richland Hills Community Center, Rufe Snow Road and 820. We start at 9:30 am - hope to see you there! ***BAM***

-----MINUTES OF NET99er MEETING
of OCTOBER 3, 1992-----

Barbara Massey called the meeting to order at 9:30 am. The minutes and treasury report were read and approved with a slight update on the treasury.

Jeff Drinan reported that very few callers are using the BBS and that each morning he needed to reset it.

Jim Lesher demoed the DOM, then a short Buy-Sell and Swap session was held.

Barbara and Jim Lesher set up the MBX using a cartridge that Jim had brought so we could see the MBX in action and it was an amazing demonstration. Hopefully Ron Miller will be able to locate his cartridges to accompany the MBX that he had so graciously donated to the club.**JoNell Thompson-Secretary**

-----MINUTES OF NET99er MEETING
of OCTOBER 3, 1992-----

The meeting was called to order at 9:30 am. A discussion of nominations for officers for the election in January 1993 was held. The tentative officers are: President-Barbara Massey; Treasurer-Chuck Tolsma; Treasurer-Cal Koether; Secretary-JoNell Thompson; NL Editor-Barbara Massey; BBS SysOp-Tom Collins; Librarian-Jeff Drinan; M/S Chairperson-open; Member at large-open. If anyone is interested in ANY opening, please let us know. You are more than welcome to help.

We discussed and approved the purchase of flowers or appropriate remembrance for JoNell Thompson regarding the loss of her j\husband, Hugh.

A reminder that the December meeting will be held at the normal time and place.

Starting January, Jim Lesher will no longer be bring his personal system to the club meetings. We will need someone to start bringing the club's equipment.

December meeting to be a Christmas oriented i.e. mostly a social meeting. Bring party favors? Food? Candy? Any ideas anyone? Act quickly...Time is short!!

Jim Lesher presented the DOM.

The meeting adjourned about 12:00 noon.**your substitute secretary-Chuck Tolsma**

PAGE PRO PAGE COMPOSER
BY DEANNA SHERIDAN
NORTHCOAST 99'ERS

I HAVE HAD SOME DIFFICULTY IN GETTING INTO THIS PROGRAM; SO PLEASE BEAR WITH ME IF I RAMBLE. FIRST OF ALL, THE WAY TO REALLY LEARN A PROGRAM IS TO EITHER DO A DEMO AT THE MONTHLY MEETING OR A REVIEW FOR THE NEWSLETTER. THEY ARE EDUCATIONAL EXPERIENCES, TO SAY THE LEAST!

ASGARD'S FLYER ON THIS SOFTWARE STATES THAT COMBINED WITH PAGEPRO, YOU CAN NOW CREATE AND PRINT MULTI-PAGE DOCUMENTS IN EITHER LANDSCAPE (SIDEWAYS) OR PORTRAITE MODE, WITH COLUMNS OF 60, 88 OR 128 WIDE. READING THE DOCUMENTATION GAVE ME VERY LITTLE IDEA OF HOW THIS WAS TO BE ACCOMPLISHED. SO, I DECIDED TO LOAD AND PRINT THE DEMO FILES TO SEE WHAT WAS REALLY HAPPENING. AN EXTRA PARAGRAPH EXPLAINING THE TYPES OF FILES PAGE COMPOSER WILL HANDLE AND WHY WOULD HAVE CLEARED UP THE MATTER IN SHORT ORDER

FIRST TO USE PAGE COMPOSER EFFECTIVELY, YOU DEFINITELY NEED PAGE PRO; LOTS OF GRAPHICS AND LOTS OF FONTS AND WHATEVER ELSE YOU WANT TO ENHANCE YOUR DOCUMENT; SUCH AS BORDERS, ETC.

A TWO-COLUMN PORTRAITE DOCUMENT FROM PAGE COMPOSER IS ACTUALLY TWO FULL-PAGE DOCUMENTS FROM PAGE PRO PRINTED SIDE BY SIDE IN HIGH RESOLUTION. IN OTHER WORDS; THE DOCUMENTS ARE "SHRUNK" TO FIT THE PAGE. AS YOU CAN SEE FROM THIS PAGE, THE ONE-CHARACTER HIGH FONT IS REDUCED TO WHAT WOULD NORMALLY BE CALLED "CONDENSED" PRINT. THE PRINTING IS DONE IN A MANNER THAT DOES NOT DISTORT GRAPHICS BY ELONGATING THEM AS SOME HIGH-DENSITY PRINTING DOES.

YOU WILL DEFINITELY WANT A VERSION OF PAGE PRO THAT WILL SAVE A PAGE PRO PAGE AS A PICTURE. THIS "SEEMS" TO BE THE ONLY TYPE OF FILE THAT PAGE COMPOSER WILL LOAD. SO IT TAKES AWHILE (OR AT LEAST IT DID FOR ME) TO CATCH ON TO THE FACT THAT I WOULD HAVE TO PREPARE EACH PART OF MY PAGE IN PAGEPRO, SAVE IT AS A PICTURE, AND THEN PLACE THE PICTURES WHERE I WANT THEM IN COMPOSER TO ASSEMBLE THE PAGE OR PAGES. YOU CANNOT TYPE TEXT ON THE SCREEN IN THE EDITOR...ONLY PLACE PRE-FORMATTED FILES IN THE DESIGNATED AREAS.

THE PROGRAM USES A "MAC"-TYPE POINT AND SHOOT INTERFACE THAT SUPPORTS EITHER KEYBOARD, JOYSTICKS OR MOUSE. I HAVE FOUND THAT THE JOYSTICK OPTION HAS VERY SLOW RESPONSE AND I HAVE TO CLICK ON SOME OF THE OPTIONS SEVERAL TIMES BEFORE THEY RESPOND. ALSO, THE JOYSTICK MUST BE IN "EXACTLY" THE RIGHT PLACE TO GET A RESPONSE. KEYBOARD IS A LITTLE SLOWER TO MOVE AROUND, AND THE RESPONSE "ENTER" KEY IS ALMOST AS BAD. I WOULD HOPE THAT A MOUSE USER HAS BETTER LUCK

A FILE MUST BE NAMED AND OPENED BEFORE ANYTHING ELSE CAN BE ACCOMPLISHED. AFTER THAT, YOU LOAD YOUR PICTURES TO PRINT, EITHER FULL PAGES OR INDIVIDUAL AS PREPARED IN PAGE PRO. DEPDENING ON THE COMPLEXITY OF THE PAGE, YOU SHOULD "DRAFT" A MOCKUP OF WHERE THE PICTURES ARE TO BE PLACED.

AFTER THE PICTURES HAVE BEEN PLACED, THE FILE CAN BE PRINTED OR SAVED. THE OTHER MENU OPTIONS ARE "NEXT PAGE", "PREVIOUS PAGE" AND "GO TO". THE DOCS SAY YOU CAN HAVE UP TO 999 PAGES IN A DOCUMENT. HOWEVER, DUE TO THE LARGE SIZE OF A FULL-PAGE PICTURE FILE FROM PAGE PRO, AND TWO OF THESE FILES TO A PAGE, YOU HAD BETTER HAVE A HARD DISK FOR THOSE FILES.

WE COMPLAIN AND COMPLAIN THAT OUR TI'S WON'T ACCOMPLISH THIS OR THAT OTHER COMPUTERS CAN DO. I THINK WE EXPECT A LOT FROM OUR LITTLE 2 MEGAHERTZ SYSTEM WITH ABOUT 48K OF WORKING MEMORY. WHEN WE DO GET PROGRAMS THAT BRING US INTO THAT IBM WORLD, WE COMPLAIN AGAIN THAT THEY ARE TOO HARD OR TOO SLOW. I THINK THE FINISHED PRODUCT FROM THIS PROGRAM IS EXTREMELY PROFESSIONAL. HOWEVER, I NEED TO STRESS THAT IT WILL TAKE A LOT OF PATIENCE AND TIME TO CREATE THESE PAGES.

IT HELPS A LOT TO HAVE EITHER A RAM DISK OR A HARD DRIVE BECAUSE YOU HAVE TO SWITCH BETWEEN PAGE PRO, FUNNELWEB, AND COMPOSER FREQUENTLY. IF YOU DON'T HAVE YOUR FAVORITE GRAPHIC OR FONT IN PAGE PRO FORMAT, YOU WILL HAVE TO TAKE TIME TO DO SOME CONVERTING. IF YOU WANT SPECIAL HEADLINES WITH ADDITIONAL FONTS (PAGE PRO ONLY SUPPORTS TWO AT A TIME), YOU WILL NEED TO DO SOME CREATIVE WORK IN TI-ARTIST AND THEN CONVERT TO PAGE PRO FORMAT.

LASTLY, BECAUSE THE DIFFERENT EFFECTS IN COMPOSER ARE CREATED THROUGH RESOLUTION RATHER THAN SIZE OF FONTS, PRINTING IS EXTREMELY SLOW. THE DOCUMENTATION STATES THAT THE SPEED OF COMPOSER IS TIED TO THE DISK DRIVES AND PRINTER, AND THE ONLY WAY TO SPEED IT UP IS TO USE A FASTER DRIVE AND PRINTER. IT ALSO STATED THAT RUNNING PAGE COMPOSER FROM A RAM-DISK CAN REDUCE THE AMOUNT OF TIME IT TAKES TO PRINT A SINGLE-PAGE 960 RESOLUTION PORTRAIT DOCUMENT WITH 18 PICTURES FROM 20 MINUTES TO 5 MINUTES. WELL, I PUT MY FILES ON MY RAM DISK AND USED MY LASER JET FOR PRINTING, AND FOUND THAT IT TOOK A LOT MORE THAN 5 MINUTES TO PRINT THE LEFT SIDE OF THIS ARTICLE SO THAT I COULD GET A GOOD IDEA OF MY PLACEMENT. THEREFORE, IF THE FINISHED PRODUCT ISN'T WHAT YOU EXPECT THE FIRST TIME AROUND, IT WILL TAKE TIME TO REFORMAT AND REPRINT. PATIENCE IS THE KEY.

ALSO, IF AFTER YOU PRINT THE FIRST TIME, AS I DID, AND FIND THAT YOU PREFER TO USE A DIFFERENT RESOLUTION, YOU WILL NEED TO CREATE AN ENTIRELY NEW FILE, AS YOU CAN ONLY CHOOSE THE RESOLUTION AT THE TIME YOU CREATE THE FILE AND CANNOT BE CHANGED AFTER IT IS SAVED AND CALLED BACK TO EDIT. A SEPARATE UTILITY FILE IS INCLUDED ENTITLED "CLIPIX" WHICH WILL CLIP GRAPHICS FROM LARGE FILES FOR USE WITH PAGE PRO. MOST OF THE "CLIPPING" PROGRAMS WILL ONLY SAVE A GRAPHIC THE SIZE OF THE SCREEN AREA, OR THE LARGEST SIZE THAT TI-ARTIST WILL SUPPORT. WITH CLIPIX, YOU CAN PAGE DOWN AND GET PICTURES THAT WERE TOO LARGE BEFORE. HOWEVER, THE PICTURE MUST BE IN PAGE PRO FORMAT BEFORE IT CAN BE CLIPPED. THAT MEANS THAT IT WAS HAVE TO PREVIOUSLY BEEN CONVERTED FROM ANOTHER FORMAT, USUALLY PCX OR MAC BY ANOTHER CONVERSION PROGRAM. IT WOULD HAVE BEEN MUCH MORE CONVENIENT IF IT WOULD HAVE LOADED A PCX OR MAC FORMAT AND SAVED IT FORMAT AND SAVED IT TO PAGE PRO.

As I have seen so little written about J. Peter Hoddie's "SORT/EXP" (how is it more people do not know of it or use it?) that I decided to devote this month's column to a beginning introduction to it. It is FAIRWARE! (Fairware Disk #425).

This remarkable sort program by J. Peter Hoddie, was released (1988) under the title: "SORT/EXP". (The disk name I have is "JPH/SORTS"). It is a very short 13 sector program but is one of the fastest sorts around.

The requested Fairware fee is only $10. but as Hoddie has dropped out of the TI community, we no longer have a valid address for him. If and when an address is ever made known again, we will publish it. "SORT/EXP" alone is a bargain at this price! (Also included on this same disk is "MEGASORT96" (for the Geneve) and "ENCODE-IT", complete with docs).

SORT/EXP is capable of sorting on more than one field but to keep this article simple, I will step through its use for sorting on one field. After loading SORT/EXP (it loads almost instantly) with BOOT or any utility-type loader (TIWriter, etc.), the first screen you will see is the title screen "Sort Experiment" but this program is not an experiment!-- it is a very fast working sort program for any DV/80 file.

After pressing Enter to pass up the title screen, you will be asked for "File Name". Enter your file name. If you enter an invalid file name or disk number, the program defaults back to the Title Screen. If your file name and disk number are valid, you are taken to the next 3 prompts: "Field # Start Length". For these inputs, enter "1" for Field #. For Start, you input the column number you wish the sort to act on. For Length, you enter the ending column number of your column to be sorted. Here is an example, assuming a file of 5 columns of data:

```
A Column  B Column  C Column  D Column  E Column
0         1         2         3         4
1.........1.........2.........2.........2........
```

Repeating instructions so far, using the above 5 Column example you:

Enter Field # as "1", press Enter

Enter Start # as "22" (assuming you wish to sort by column C which sets sort pointer to the first Character of column C).

Next you are asked to enter the desired length. This depends upon the maximum number of characters in the C Column. The above example sets the column for a width of 9 characters, leaving one free space between columns. Though a space does not have to be between columns, full print-outs will be easier to read! So enter 9. (If you enter "11", for example, the sort will consider this 11th character (the first character in Column D) in sorting any similar items in Column C. Enter desired Length and press Enter.

Now the "Field 1" prompt appears. Enter "1" for Field 1 and as soon as you have pressed one, a "2" appears directly below the "1". Ignore this for now and press Enter.

Next you will be asked "Is this correct?". If you see you have entered what you wish, then press Enter again. ("N" allows you to re-enter your choices).

The next prompts are: "1) Descending 2)Ascending" (A Descending sort sorts in Z to A order whereas Ascending sorts A-Z. You will probably elect "2" for Ascending.

Now you are asked to select: "1) Shell 2) Shuttle" If you don't know what to choose it doesn't matter unless you have very large files to sort. They both sort quickly. I believe (if I've got it right) the Shuttle is faster for very mixed up unsorted items whereas the Shell sort is faster faster on files with many items already in order. Try timing both sorts on the same file and see which works faster. (I never had time to fill my pipe and light it--let alone go for coffee--they sort so fast).

Once you have selected the sort choice, the sorting begins. During the sorting, there is no sign of any activity, so WAIT! Your computer has not locked up. In usually less than 30 seconds there suddenly appears: "SAVE Filename:" near the bottom of your screen. This tells you the sort is done. Now enter the name you want your file saved under. *NOTE*: BE CAREFUL to have a good disk in the correct drive and a file name that does not exceed 10 characters. The sort will abort (and be lost) if you make a mistake here. If you want to keep your unsorted file, be sure to use a different filename! The file you sorted will be over-written if you use the same name. Incidentally on short files, sorting is so fast you may wonder if it "took". Don't worry-it did.

Once the sort has been saved, the program goes back to the title screen. Press FCTN/Quit to leave the program.

A word about non-columnar DV/80 files. For sorting purposes, these files are limited to 80 character widths. The sort acts on the first character of each line so your Start # will always be "1" unless you want to see what happens with other numbers. If your DV/80 file lines exceed 80 character length, the sort will sort upper-case (A-Z etc.) to the top of your list and lower case (a-z etc.) will be pushed to the end of the list. Spill-over (from lines exceeding 80 characters) will then be separated from their line beginnings, and you'll have no way to recover their original continuity. Because the data in such files have no columns to constitute separate regular fields to sort by, you are limited to a single field sort. If you enter 80 at the Length Option, you will get a complete sort through all 80 characters on the line. One way to categorize such data (before sorting) is to edit your file by entering code marks at the beginning of each line. If the line is a full line, remember you will lose the ending character by doing so. Remember to avoid using any characters like the period, which will send a signal to your formatter you didn't intend. If you are not using a formatter to print out your sort, you can use these characters if you use Text or Program Editor or other file reader to obtain your printout.

USING THE TI-WRITER MAIL MERGE
By Chick De Marti

WHAT IS MAIL MERGE?

A Mail Merge is a convenient way to mail a form letter to many people without manually typing in the names, addresses, etc. This is extremely useful when various forms of communication must be sent to the same group of people monthly, quarterly, etc. So the obvious thing then would be to create the mailing list. Use this...

Example:

```
1 Mr                      (Press ENTER> after each line.
2 Tom                     Yes! the numbers 1,2,3,4,5,6
3 Jones                   must be entered. Notice the '*'
4 2341 Any Street         asterisk separating each group of
5 Somewhere, CA 91123     members, TIWriter recognizes it
6 Jones family            as a separator.)
*
Ms
Jane
Smuthers
7777 Lucky Lane
Gameplays, CA 91119
*
1 Mrs
2 Eunes
3 Somuch
4 2468 'Preciate Ct.
5 Fairytale, CA 91121
6 Somuch family
*
```

Save as DSK2.MYFILE/1

Next, let's create a form-letter. This is the one I created for our August meeting. A typical heading would look similar to this:

```
.FI;AD;LM 8;RM 72   <enter>
.IN +32               "     (the +32 will print the following
Chick De Marti        "     lines near the center of the page)
18028 Falda Ave.      "
Torrance, CA 90504    "
<enter>               "
August 26 1991        "
.IN +0              <enter> This cancels the .IN on Line 2
<enter>
<enter>
*1* *2* *3*         <enter>   = Title, F-name, L-name
*4*                   "       = Street address
*5*                   "       = City, State, Zip code
<enter>
Dear *2*,           <enter>   = F-name
```

(now you would type the body of the letter)

Now load your FORMLET/1

Normally, this would probably be a different heading, but for the sake of expediency, let's use what we have. Go to the line with Dear *2*, and change it to Dear *1* *3* (Mr Jones). Now to the letter itself. Type in:

Thank you for your recent payment. Your balance is *6*. If you have any questions, please give my office a call.
.IN +32
Yours truly,
Ida Dungood
SMALL LOAN CO.

Again using the FCTN-3 key, delete the rest of the letter and save it as: DSK2FORMLET/2
(Did you make a copy using PF?)

TI-Writer matches all numbers within the asterisks (*1* *2*, etc.) with the respective lines in your mailing list. If it does not find the corresponding number, it halts the printing and asks for your input. Now print your letters answering the prompts as you did for FORMLET/1. The printer will stop and at the message "Enter input:" Enter a dollar sign and any amount you want ...
Example: $1150.90. When you read the letter, the second sentence will now read:
        "Your balance is $1150.90.

Ther doesn't seem to be any limit to the number of inputs you may have. With that in mind, another opportunity to use the Mail Merge might be at Golf tournaments. Still using the mailing list MYFILE/2, type in this form-letter:
Dear *2*,
<enter>
Thank you for participating in the Junior/Senior Golf Tourney. Your official handicapped score was *6* earning you *7* as your portion of the 'Winner's Booty'. Looking forward to seeing you again next year.
.IN +32
Bobby Sox, tres.
Good Guys Golf Gang

The inputs in the above example are obvious, and as you can see, the occasions to use a Mail Merge are endless. So try it out for kicks...I did and I learned how to Mail Merge.

    -**- Chick De Marti -**-

Example:

        This year we are going to have many interesting demonstrations,
and surprise guests. We expect some of the leading programmers in the
TI community to be displaying the wares, as well as their expertise.
There will be many bargains available in software as well as some
'steals' in hardware.
        It would be my pleasure to greet the *6* at the door personally.
So do come to the meetings and enjoy yourself as well as learn many
interesting things.
<enter>
.IN +32
Yours truly,
Chick De Marti   Save the letter as DSK2.FORMLET/1

This is about as simple as it can be. By doing the mailing list in
the TI-Writer editor, you can make changes, corrections, as well as
additions with little or no trouble.

NOTE: when you are through typing your form letter, before leaving
        the editor, make a quick copy of it using the PF command
(Print-File). The resulting copy will include all 'dot' commands and
will allow you to check the finished copy with your original notes.

        To make a 'Print-File' copy, while still in the editor:
                press Fctn-9 to get to the command line,
                type PF   <enter>
                      PIO   (No! LF (line feed) is not needed)
                            (You may be using BA-etc. etc. In any case,
                            just press <enter> and a copy of your
                            formletter will be printed.
PRINTING THE LETTERS

        This is even easier. Get into the Formatter and follow the
prompts:

        Program   DSK2.FORMLET/1   <enter>
        Printer   PIO.LF   <enter>
        Use mail list?   Y   <enter>
        Page(s)/All   A   <enter>
        Pause after each page   N   <enter>
        Mail list name   DSK2.MYFILE/1   <enter>

        That's all there is to it. When the letters are printed.
TI-Writer will replace *1* *2* *3* with Mr Tom Jones and the *4* with
his address. *5* will be replaced by the City, State etc. and at *6*
will insert "Jones family". Neat huh?
        Now TI-Writer has another feature you might like to try. It is a
manual insert option. Using this option you could insert personal
information anywhere within the letter. This could be used by a
business for billing purposes, or for a club to make it's members
aware of changes, awards etc. Here's a quick 'how to'.
        Let's make a minor change to our mailing list. Load MYFILE/1
into your editor and the delete line sixes (Jones family, etc.).
Place the cursor on the '6' and press FCTN3. When finished:

                Save as DSK2.MYFILE/2

XBASIC MISCELLANY #1
By Earl Raguse

One of the things I learned while reworking TIPS 1.6, to 1.6/ER and writing TIPSLABEL was that TI didn't tell us in the XB Manual, or the later addendums, all we should know about XB. At least in what I could find. You may recall in my article on TIPS 1.6, that I found that after I had converted TIPS to using CALL KEY(3,K,S), DISPLAY AT, and ACCEPT AT, I could not enter lower case in ACCEPT AT.

I had some recollection that I had done it once upon a time, but I was not sure about it. Then I remembered that XB does not have a command to restore the lower case character set once they have been redefined. CHARSET does not do it, it only restores the uppercase set. That presumably was because early XB did not have a lowercase set. I then reasoned that since that was true, it made sense that ACCEPT AT would only take uppercase.

I had plans for writing an assembly routine to LINK that would do it. I had once written an assembly program to take keyboard text input, and further I knew that Adrian Robinson had written in the ROM, a very detailed ACCEPT AT routine in assembly. My problem was that I didn't know how to get into Irwin Hott's LOADER program for TIPS. That is where there assembly routines are hidden, submerged below the XB.

How wrong I was! I did not know until I got a call from Adrian (Robbie) Robinson, that the problem was not with ACCEPT AT, but the fact that I had used CALL KEY(3,K,S) to insure that all entrys to CALL KEY would be upper case, instead of running them all through Ron Wolcott's assembly routine for converting inputs to upper case. I

didn't recall where I learned that CALL KEY(3) did that, surely not in the XB manual, but I knew it. It turns out, it was the Users Refence Guide.

What I didn't know was that once you do a CALL KEY(3,x,y) all, and I mean ALL, keyboard input thereafter, for CALL KEY, INPUT, LINPUT, ACCEPT AT, ect, etc, is restricted to upper case. I had used that fact for CALL KEY in my DIRectory programs. I didn't know that it stayed that way until you returned to the Title Screen. Or that you must do a CALL KEY (5,x,y), to restore normal upper and lower case, before any statement that calls for keyboard input. It matters not what x & y are so long as they are legal numeric variables. Lower case character redefinition has nothing to do with this. That is another story, later alligator. where again Robbie used his assembly knowledge to help me out of an of an XB problem with CHARSET.

After that phone call, I searched everything I had on XB, to no avail, I could find nothing to tell me this. The best source on the keyboard is the User's Reference Guide (the "Green Book"), but it does not even imply that it works that way. About two days later, I got a letter from Australia, from the Hunter Valley Assembly Guru, Ross Mudie, telling me the same thing Robbie did. I then got suspicious, why are the only people who know this the Assembly guys, I then scoured the TI Editor Assembler Manual. Firstly, I found a reference to the User's Reference Guide. There was however, a discussion, see page 250, about the fact that the Keyboard "device" was selected by placing a number, they discuss only numbers 0-3, into >8374. (Hex numbers are indicated by preceding with > as in >8374). Now this discussion makes no reference to CALL KEY, it is generic, and therefore refers to all keyboard

input. Also, once a number is loaded into location >8374, it stays there until changed. I can now assume that the XB CALL KEY does among other things, a CALL LOAD of the key number into >8374 which requires a new CALL KEY or CALL LOAD statement to change >8374 to a new number. I have tried to test this theory in XB, but to no avail, Robbie says it works, but it won't for me. If I were working in assembly this would be rather understandable, but to the average XB Manual reader, TI left it quite totally unexplained. I wrote a very interesting program called LOAD/PEEK to test CALL LOAD and CALL PEEK. Next time I may publish it, you could use it to learn a lot about how this TI computer works.

So what does this all mean? If you wish XB to return upper case only, do a CALL KEY(3,x,y); to restore lower case, do a CALL KEY(5,x,y); and to keep the previous state (ie don't disturb the keyboard device previously selected) use CALL KEY(0,x,y). I note that most XB programmers use CALL KEY(0.x,y) almost exclusively. They are then not taking advantage of the computer's (and XB's) capabiliies. I hope after this you will.

In the following months I plan to write about some of the other things I have learned about XB in recent days. Also including the above program LOAD/PEEK. One thing I have looked into is error trapping, and some things I have thus learned are not documented in the XB Manual. I have also learned some some otl.'r helpful things which I will get around to talking about. I will have something to say about the power of ACCEPT AT, that even my High Priced Spread computer does not have. Until next time, may your 99/4A's never do a hang-up in 1991.

Future Developments:
The opinions of one 99/4A developer

By Chris Bobbitt

OVERVIEW

With the apparent final demise of Myarc, and most likely the Geneve with it (not-withstanding the efforts of those trying to salvage the remains), the TI community is at an impasse.

For the last 6-7 years a substantial portion of the software and hardware developers among us have been focusing on the Geneve. While some interesting work has been done, the frustrations in developing for the machine has driven more developers out of the community than anything since the 99/4A was discontinued. The Geneve is the only computer that could make what little TI published about the 99/4A look like an embarrassment of riches.

In fact, considering how everything turned out (the intentions and hopes of all parties involved aside, including myself), I would say the Geneve was probably the worst thing that could have happened to the TI-99/4A. It siphoned off developers who would otherwise have worked on addressing the shortcomings of the 99/4A itself. The Geneve itself was a radical answer to those same shortcomings, and the fact that it was a total solution in one neat package goes a long way towards explaining the efforts and passions it inspired.

However, any reasonable assessment of the situation would conclude that it is too late to make lemonade out of this lemon. Even if the machine were readily available today. we would be addressing ITS shortcomings by now. In 1985 512K of RAM and 128K of Video RAM was still something to talk about. But today, the capabilities of the machine are about as relevant to current computing requirements as the IBM PCjr, Atari 520ST or Amiga 1000. Further, the basic problems with the Geneve's design would insure that the task of updating the machine would be just as great as that of updating the 99/4A.

While I am not trying to discourage all of the developers who have spent years working on the machine and in some cases are still at it, the simple fact of the matter is that the window of opportunity for it has long been shut. The Geneve just can't compete against the 80386 and 68030 computers of the world, much less computers using the 80486, 68040 and RISC processors.

ASSESSING THE 99/4A

The basic problems with the TI-99/4A have been discussed to death, but its important to restate the obvious sometimes.

1. Memory

The 99/4A never had enough of it. There probably isn't a single PC program available today that will fit in 32K (or even the 40K available with a Supercart).

Granted, we've gotten a lot of mileage out of virtual memory techniques, the 9900 processor's more efficient use of memory, and programming in straight assembly. But the fact of the matter is we are quickly coming to the end of the road on that. Who would use a virtual-memory word processor that had to go to disk to get a bit of code every time you did a search and replace? Putting all of the features people expect from modern software, not to mention modern graphical user interfaces, in 99/4A programs is very difficult to impossible with current memory constraints.

Oh, people can still write games and utilities and some types of application programs, but when is the last time that a major new application (like TI-Base or Page Pro 99) has been released? How can someone improve much on those programs when they've run out of memory and can't make the program any smaller? As programs for PCs and Macs improve they inevitably get larger and larger because developers are loath to remove features found in previous versions. Hence, programs like WordPerfect that have been through 5 major revisions have everything but the kitchen sink, and 1200 page manuals to prove it.

Developing software for the 99/4A has always been about tradeoffs, but you eventually get to a point where the tradeoffs cost too much, and so new software development stops. We are about at that point with 99/4A software. Without more memory we will not see any more advances in databases, spreadsheets or desktop publishing, much less newer applications like computer faxing.

2. Speed

The 99/4A is slow, especially by today's standards. Personal computers that perform 20 million instructions per second are found on the desks of secretaries. The 99/4A can just about manage 1/100th of that. The only thing that has kept the 99/4A competitive for so long is that the vast majority of the power is wasted on PCs and Macs by abominable software that needs 32K of code to read the keyboard.

The proliferation of graphical user interfaces (GUI) has put a premium on speed like nothing before in the PC world. While Mac users have been enjoying their benefits for years, the PC world has only recently awakened to the fact that they go a long way towards making PCs less user-hostile (one of the big things that kept people using the 99/4A for so long). While Windows is still a pile of you-know-what compared to (say) the NeXT or the Mac interface, it almost makes the PC as intuitive to use as a circa 1979 TI-99/4A.

To keep up with the Jones' a GUI for the 99/4A is inevitable. However, to produce a practical GUI for the 99/4A you'll need more speed and a lot more memory. Please note that the latter goes quite a bit of the way towards mitigating the lack of the former because effective speed is still sometimes more a product of logical design and efficiency than raw power. However, all programs (especially efficient ones) benefit from increased speed.

3. Video

Of all the areas that we have tried to keep up with other machines, video display has probably been one of our more successful attempts. Despite occasional supply problems, since the mid-80's we've had a steady stream of new video products first based on the 9938 and more recently the 9958. We've been extremely fortunate that the video processor line used in the 99/4A found a wider commercial following than the 9900 itself did.

However, this is not to say there isn't room for improvement. Desktop publishing, GUIs and more advanced applications like Multimedia and Desktop Video make the 9938/58 adequate at best. Despite the fact that the improved resolution offered by these processors is still largely unexploited, within a couple of years it will seem as antiquated as, well, the 9918a.

However, the interest in advanced video that exists today is enough to insure that we will see continued products using 9990s and perhaps even 34000 series processors. However, there is no point in tying a state-of-the-art video system to the 99/4A if there isn't enough CPU RAM to hold a single screen of data (there isn't even enough now with the 9938).

4. Sound and Speech

This is one of the least painful shortcomings of the 4A because compared to much of the opposition, we STILL have competitive sound and speech capabilities.

There is nothing like our speech system available in the PC or Mac world today which relies on hideously inefficient digitizing to achieve comparable sound quality (10 seconds of digitized speech can take 1000 times more storage than 10 seconds of speech from our synthesizer). While our sound system has aged less gracefully (it was second-best even to the Commodore 64), the 99/4A still sounds better than many PCs sold today. However, Desktop Video and Multimedia - granted both cutting edge applications that won't be widespread for a while yet - both require tremendously enhanced sound and speech capabilities. The emergence of these applications has also started to drive PC developers to improve sound and speech on the PC - its a safe bet that in the next year or two virtually every PC sold will have a Sound Blaster-like sound board as standard equipment (the Mac always had it).

Any expansion of sound and speech technology for the 99/4A will require an increase in memory and probably speed - especially if it includes input of speech and sound as well as output. As with advanced video, there is no point to improving sound and speech capabilities if you have no place to put this type of data, or you can't get the data to the

hardware fast enough.

5. Storage

As with video, the 99/4A development community has tried hard to keep up with the latest in storage technology. This has been as much out of necessity as virtue - because of the limited RAM of the 99/4A the only way to increase the functionality of 99/4A programs has been to use disk space as program storage space (virtual memory).

The HFDC brought the 99/4A up to the level of an IBM AT, and the promised IDE and SCSI controllers would bring the 99/4A into the modern age - on par with the latest from the clone makers and Apple. The SCSI controller, in particular, would give the 99/4A access to the wide range of storage peripherals outside of disk drives - CD-ROM players, tape-backup units and so forth.

While increased storage is always useful, it won't permit any new types of software application development, however, beyond what we can do already do today with the HFDC, RAM-disks and the like. However, speedier storage WILL make living with virtual memory techniques a lot easier. If it only takes a half-second for the "search and replace" function to load off of your SCSI hard disk, then you can probably live with a virtual- memory word processor.

6. Input/Output Devices

Along with storage and video, the variety of input and output devices has for the most part kept up with technology. You can attach the latest printers and serial mice to the 99/4A as easily as you can with a PC. We can thank the fact the we were fortunate TI supplied us with standard RS232 and Centronics-compatible Parallel ports for this (not necessarily a given with older computers).

There is, of course, always room for improvement. Many PCs nowadays sport second-generation parallel ports that can handle much higher speeds and two-way communications, and Apple Macs have featured the RS422 interface for years and years (which also works faster). Further, our serial interface is not quite compatible with some applications. MIDI has pushed the 99/4A to the limit in this respect - the technical problems related to properly exploiting this technology are great because of shortcomings with the interface.

Also, as with improving most other aspects of the 4A, an improvement in memory is also important to optimize improvement in input/output devices. Additional memory would permit better buffering of data (which would facilitate faster I/O), and make supporting MIDI a lot easier.

SUMMARY

As it stands the 99/4A could stand major improvement in memory and speed, and substantial improvements in storage, and modest increases in speech and sound, video and I/O.

This isn't meant to indict the 99/4A - the machine will still be useful for years to come. But lack of improvement in these areas will mean that TI-99/4A applications will seem ever more and more primitive by comparison with what can be done on other computers. Without improvement it will be impossible to even contemplate many up-to-date applications.

PRESCRIPTIONS

While the situation is not good, it is far from hopeless. Despite the thinning of the ranks caused in part by the Geneve, there still exists a dedicated group of people interested in developing new hardware and software for the 99/4A. Chances are any substantial improvement in what the 99/4A can do will bring more of these people out of the woodwork as it did in the past with the introduction of GRAM devices, higher capacity drives, new programming languages, and so forth.

Lately small groups of individuals have tried to help solve the problem by defining it better in conferences and meetings. The best known of these efforts is the 99/4A Standards discussions that have been held on various electronic networks and at TI conventions.

1. Memory

The 99/4A Standards committee (the NTISC) has evolved into a general discussion on addressing the memory shortcomings of the 99/4A - essentially by defining common protocols for accessing these types of devices (physically and through software). While there is a lot of utility in such discussions, I also have some concerns about them. I believe that in the short run they unnecessarily raise expectations; the great disparity in the way currently available memory devices work will make what they are trying to do very complex; and, actually developing devices and software to pre-designed specifications sometimes points out inadequacies in the specifications more than anything else. In this way they can be counter-productive.

Further, trying to write software that interfaces to a wide variety of memory devices promises to be most difficult. Optimizing a program to work with a variety of different page sizes and interfacing methods, even smoothed over by common access routines, may be impossible. One program we worked on at one time, Press, became impossible to complete on this point alone.

The best way to promote a standard memory device for the 99/4A is to let the community vote with its dollars. The most popular memory device becomes the de facto standard. If there are shortcomings in products that are currently available, anyone who makes a better one can probably sell a lot of them if they address those shortcomings.

This isn't to say that a standards committee can't help the matter along. While it may be impossible to set a standard access method for currently existing memory devices, a standards committee could certainly design an "ideal" memory system that addresses the problems with current systems. This design could then be licensed to multiple vendors - better to insure steady supply and price competition.

2. Speed

Speed is a much harder commodity to come by. The only really compatible speed improvement currently available is to increase the clock speed. While this improves speed up to 30% or so, even this causes problems with some programs.

Recent efforts in this area (besides the Geneve) include a 99105 based daughter-board that would plug into where the 99/4A's CPU currently resides. While this would certainly meet the need for speed, it has evidently hit a stone-wall during development.

A better long-term answer may be to simply replace the entire motherboard - in essence what Myarc did with the Geneve. While this would probably be more costly than a daughterboard, the technical problems might actually be less - particularly if the designers put all of the common peripheral ports on the motherboard and didn't worry about access to the P-Box.

The cost of such a thing would certainly be an issue, particularly if the designer used one of the few remaining available 9900 compatible processors. A 99105 costs around $225 EACH in quantity (making the COST of a full-blown system built around it $500-1000 depending on the components and peripherals included).

The designer of such a system may want to consider using another microprocessor and emulating the 9900. The upside to this is the cost per processor may go down dramatically. The downside is the technical problem of emulating a 9900, and potential compatibility problems that may result from a buggy emulation. Of course any emulation done depends on the microprocessor that is running your emulation - and it may be easiest to do a 9900 emulation with the new TI SPARC chips or the TI 34020. These chips have a few 9900 genes floating around in them and have enough raw speed to make a emulator that is 20 times faster than the original. Of all the problems facing the 99/4A, the speed problem is probably the hardest.

3. Video

Hold the course! More 99/4A software is becoming 80-column aware all of the time. If only we could convince more people to buy the upgrade - that would insure both a steadier supply and future development of both high-resolution software and hardware.

4. Sound and Speech

There have been several efforts to improve both of these things on the 99/4A - ironically both by TI.

The MBX system is STILL just as good or better than the typical speech recognition device for the PCs and Macs, and TI speech synthesis technology is peerless. It is rumored that TI has also married the two with speech recognition devices that interpret the spoken word into the Linear Predictive Coding system used by TI synthesizers (including our own Speech Synthesizer). A board built around this technology would be on the cutting edge for all computers!

Also, TI reportedly was behind the design for the FORTI music card, which expanded our system to 12 voices with stereo output.

All we would need to bring the 99/4A into modern times would be a Sound Blaster for the 99/4A with an expanded version of the speech technology we already use.

5. Storage

This area promises to be one of the first addressed, particular if the new IDE and SCSI hard disk controllers are ever released. Both IDE and SCSI offer enough speed and capacity to meet the 99/4A's storage needs for years to come. They could also facilitate cutting edge applications like multimedia, and current hot topics such as GUIs, desktop-publishing and so forth.

6. Input/Output

The only thing the 99/4A needs is a device offering 1-2 true Centronics parallel ports and 2-4 improved RS232 ports. With the former it may be possible to attach all of those peripherals designed to work on a parallel port that are available for the PC. With an improved version of the latter, some of the technical problems of MIDI would be solved.

CONCLUSIONS

The 99/4A has come a long way in the 13 or so years since its first incarnation as the 99/4. It is reliable, useful and addictive. However, if it is to become more useful, and stay as challenging and addictive, it is going to have to be improved.

My purpose in writing this was not to deride the 4A, and certainly not to offend anyone in the past or present who was/is working towards the common goal of improving the machine. Everyone's effort in this respect has been invaluable to the cause. I also didn't write this to quibble with anyone's approach to the problems at hand. While I disagree with some aspects of the NTISC, as a programmer I can certainly empathize with their aims.

The point of this article was to ask a few question of myself and others - "What's wrong with what we have now?", "What can be done to correct what's wrong?" and "What hardware would I like to write programs for?". I look forward to reading other answers to these questions.

Please send all comments to:

Chris Bobbitt
c/o Asgard Software
P.O.Box 10306
Rockville, MD 20849-0306

--WP♦