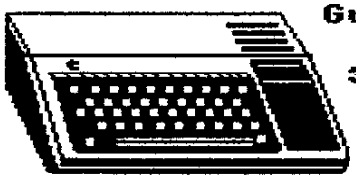
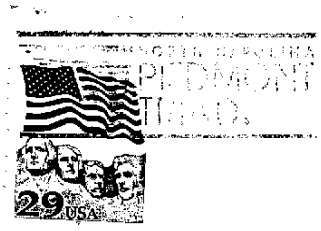
GUILFORD 99'ERS NEWSLETTER

SUPPORTING THE TEXAS INSTRUMENTS TI-99/4A COMPUTER



GUILFORD 99'ERS UG
3202 CANTERBURY DR
GREENSBORO NC
27408



TO:

George von Seth, Pres. (292-2035) Bob Carmany, Newsletter Ed (855-1538)
Tony Kleen, Sec/Treas (924-6344) Bill Woodruff,Pgm/Library (226-1092)
BBS: (919)621-2623 --ROS

The Guilford 99'er Users' Group Newsletter is free to dues paying members (One copy per family, please). Dues are $12.00 per family, per year. Send check to:Tony Kleen c/o 3202 Canterbury Dr., Greensboro, NC 27408. The Software Library is for dues paying members only. (Bob Carmany Ed)

OUR NEXT MEETING

DATE: Sept 3, 1991 Time: 7:30 PM. Place: Glenwood Recreation Center, 2010 S. Chapman Street.

Program for this meeting will be the unveiling of the latest (and best) version of FUNNELWEB -- version 4.40. If you thought the the earlier versions were something, wait until you see this one! We'll also take you through the complete configuration process.

MINUTES

The August meeting of the Guilford 99'ers Users' Group was held on Tuesday the sixth, at Glenwood recreationn center. There were six people present. We enjoyed the company of two guests Joe and Robbie Strebele from Greensboro.

The meeting opened two minutes early (according to my watch) at 7:28 led by Director George von Seth, whose watch was two minutes fast. The Sec/Treas reports were read and approved. As of this writing, the club has $168.29.

There was no old business. New business was depressing. June is the end of the Hunter Valley club from 'down under'.

There was a lot of discussion prior to the monthly presentation. The latest news on the 12 Mhz 'accelerator' chip was tossed around. Software for printer chips was discussed. TIBase, Multiplan, you name it. After 30 minutes of general topic discussion, we decided to get on with the program.

Bob Carmany's topic for the evening was on Wycove and TI FORTH programming. I was thoroughly intrigued and informed. I thought I knew a little bit about FORTH programming, but now realize I'm a greenhand novice. I can see it takes a lot to be as proficient as Bob. Bob, thanks again for the introduction.

Respectfully Submitted,

Tony Kleen

F'WEB 4.40

Some years ago, the standard joke arounf Funnelweb Farm (Tony McGoverns name for his place of abode) was when the next 'final' version of F'WEB was going to come out. If memory serves me right, there were several 'final' versions at least! Anyway, version 4.40 is the latest of these definative versions.

Tony finally squashed the last remaining bug inherent in the original TI-Writer package. Formerly, if you did a (R)ecover(E)dit on an empty file, the system would lock up. As the result of the jiggling and prodding, he finally found enough space to make the correction in this latest version. There are still a cpuple of insignificant bugs but they are all explained in the bug report accompanying the program.

Basically, everything is just as it was in previous versions with the exception of the clean-up of some minor internal

bugs in DR/DS (DISKREVIEW) and some of the other files. The functions in the text editor have been streamlined and are crisper and quicker. There are some other 'housekeeping' functions done as well.

One of the more interesting additions to the F'WEB package is the multi-listing capability of the new ML file. It is the same size at the UL (Utility List) file and serves a similar function. It collects two UL-type files plus the Disk Utility (D1) file from the central menu and displays them side-by-side on the screen for program selection. It is a simple matter to re-name the ML file UL and then create a couple of other UL-type files to be displayed under the names of UM, UN, etc. In fact, the lists could be chained to infinity with almost instant loading of virtually any A/L program. You will see a demo at the meeting.

The files comprising DM-1000 and DISKD (DISK-PATCH) are no longer supplied with the F'WEB package since their functions have been taken over by DISKREVIEW. If you have Vn 4.3x, however, they can be added and loaded as a GPL-type file (DISKD) or program image (DM-1000). DISKREVIEW has become a full-featured disk manager that doubles as a superb disk sector editor. Tony even added a function to view the comments that you can add to the file headers with John Birdwell's DISK UTILITIES.

The features that Tony has added through the years to the editor and formatter are still there along with a new feature (since Vn 4.31). (S)wap(T)abs allows a second set od tabs to be appended to a document and brought in to change the margins when needed. In additon, the margin release now takes care of both right and left margins.

Tony describes the editor/formatter as a "notebook editor" when compared to the larger PC's but I have to disagreen with him. I've used the ones on the IBM PS/2 and F'WEB is easier to use and the functions are a lot more convenient as well.

There are rumblings in both the documentation and Tony's correspondence about a re-write of the entire F'WEB program to tidy up the code and re-do the editor completely. I understand that will be the next 'final definitive' version ---Vn 4.5. It's just 'terrible' that we have to 'make do' with this version until then (please don't throw me in that briarpatch!!).

FORTI CARD REVISITED

Ever since I first heard about a P-Box card that would let you have 12 voices for music, I've wanted to get my hands on one. As you know, the 4/A console is limited to 3 notes and one noise. According to Ron Albright in the Orphan Chronicles, the FORTI card was developed by TI employed engineers, but TI elected not to buy the rights to it and market it. (The card was developed as part of a TI incentive program for employees to create new products for the 4/A; if the company liked the ideas, they would purchase the rights to them for later sale.)

(I understand that at the big CES show in Las Vegas in 1983, the FORTI card was playing an impressive rendition of "Chariots of Fire" in the TI booth.) After Black Friday, the rights to the FORTI system were purchased by Don Bynum, former Chief engineer of the Consumer Products Group. Anyway, the FORTI card has 4 sound chips on it and is driven with FORTH-like software which is supplied when you buy it.

The card is not housed in the usual metal case; instead the case is made of a special plastic similar to what is used for photographic underwater cases....very durable. Curiously, the TI logo is embossed on the outside of the case. Documentation is quite thorough, coming with two demo disks and about 36 pages bound in a 3-ring binder.

To begin, open up your P-Box and put the card anywhere your heart desires. There are RCA jacks on the back for connection to a stereo (even quad if you were suckered in on that a few years ago). I would suggest that you use the stereo connections as it makes the sound considerably better (depending upon the quality of your stereo) and the sound coming through your monitor's speaker doesn't sound right anyway.

To run the software, use E/A #3 and load the program just like you would load FORTH. When the menu screen is displayed, you will see a number of FORTH-like commands. To hear the sample tunes, just type in ALBUM and it will play the selections that have been pre-created. Some sound quite nice, others are a bit weak, but all tend to show off some of the different capabilities of the card. While the music is playing, several colored bars are displayed in an interesting manner (depending upon the music being played) which are a graphic representation of what the sound chips are doing.

Creating your own music is not too difficult, although I would recommend some musical knowledge. Even if you know nothing about music, you should still be able to convert sheet music if you study the docs a bit.

Actually, the FORTI docs start with a bit of an introduction to time signatures, key signatures, octaves, and all that other musical stuff. (At this point, I should probably note that I play guitar and a bit of synthesizer, so I do have some musical knowledge.)

First, you have to create a voiceline with the program's editor. In reality, this is the 64 column editor of FORTH, so you will need to have a monitor with good enough resolution to handle the small letters (I did not find it to be a problem unless you're using a TV).

A voiceline is just FORTI's way of noting what you want one of the chips to play (they call each voice a "musician"). If you are playing a melody, this would be one voiceline. While creating the voiceline, the user can program in a slur or even change the envelope (attack and decay characteristics). This is all done quite simply in the editor.

Once you have saved the voiceline to disk, you can go back and play it so you're sure it sounds the way you want it. It's a bit tricky to get all the notes worked out in the proper time signature, but it just takes a little persistence. Next, you start to create the conductor. This is what determines how everything comes together.

You can assign a voiceline to several voices to fatten the sound if you want. Three of the "musicians" can act as drums or extra low bass parts to fill up your sound also. The documentation has information on how to get fancy with your music such as grace notes, accents, and some of the other things that make music interesting.

In all, it's a fairly easy system to use. It can be a little complicated if you're not transposing something out of a book or from a printed page, but it's still not too hard to get the hang of it. One complaint I have is that you cannot do chorusing. (Chorusing is where you add a slightly delayed, slightly out-of-tune voice to fatten up the sound and give it a liquid sound) I suppose you can create a similar effect using the envelopes though.

Once you have created several tunes you can set up the software so that it will play them all by just typing in ALBUM. One thing I really like is that since the computer's sound is going through your stereo, that means that you favorite games will also go through the stereo. My favorite is BLASTO....those explosions are really neat sounding.

The presence of the card does seem to mess up some of the sounds when running other programs. Whenever I use P-TERM, I usually get a buzzing sound when I set instead of tones, using the stereo eliminates this problem.

In all, it seems more like a "gee- whiz" piece of hardware rather than anything really useful, but if you're into music it does sound nice through a good stereo. The price ranges from $150 to $200 depending upon the dealer. My feeling is that the card should be priced lower, say around $100. I am told, however, that costs do not allow such a low price at this time.

For further information, contact: Texas Peripherals Rt. 1 Box 15C Lancaster, Texas 75146 There is a demo tape available (or at least there used to be), but you really need to hear one through the stereo to appreciate the sound.

I should point out that the sounds produced by the FORTI card are the same tones you get in your regular programs. That is, you can't get violins or horns out of it, you are limited to whatever our sound chips can produce.

SPEECH MOD

Before I got a Triple Tech card, one thing that plagued my system was the speech synthesizer hanging on the side. Most programs would crash due to the poor connection. My solution was to just take the thing out of there and not be bothered. But some programs require that the speech box be hooked up and will not operate without it. Well, you can buy a Triple Tech card if you want the clock and buffer, but I realize that all of you are not as affluent as myself (HA). There has been much talk about putting the speech card in the P-Box, but it's not as simple as just plugging in the card (although I have seen breadboard jobs done).

Here's how to put the speech synthesizer where it belongs...in the console. First, take the circuit board out of the shell and remove the big female connector. This is done by carefully prying up each contact as you hit it with a low-wattage soldering iron. Some type of solder-sucker would be very useful whether it is braid or a solder-sucker.

Next, you'll need to take your console apart so that you can mount the card on top of the motherboard case. I won't tell you how to take the console apart since you've probably done it by this time. It is pretty easy if you just pay attention. Get some thin cardboard (like what they package with shirts) and some double-stick tape so you can insulate the circuit board from the shield on the motherboard.

You'll need a piece of 8-conductor ribbon cable and another wire of equal length. Cut the length to whatever is needed

to fit the speech board inside where there will be no obstructions when the case is reassembled. You will need to split the ribbon cable a couple of inches or so to do the soldering. Make sure you have neat ends on the cable....neatness counts. Now, refer to the diagram below for the pin designations on the speech board. The pins are labeled as if you were viewing the circuit board from the side, looking at where you took the connector off. The top pins are even numbered and the bottom ones are odd numbered.

Connect the ribbon cables to the following pins: TOP 2 12 34 36 38 40 42 44 BOTTOM 1 3 5 19 35 37 39 43 Connect the odd wire to #21 which is the big pad on the bottom where there were 4 wires going to it. This is the earth ground. Now all you have to do is connect these wires to the side connector of the console in the places they would have gone if the speech box were still there. REMEMBER, neatness counts!!!!

Once you're through, use the double-stick tape to mount the circuit board to the motherboard shield and make sure you use some kind of insulation so that it doesn't short out. Reassemble the console and give it a try. Everything should work as if the speech synthesizer were plugged in the side. NOTE: I take no responsibility for any screw-ups anybody makes doing this..it does work, but don't blame me if you fry your console from sloppy workmanship.

BASIC TIPS

1. HOW TO DISABLE THE "FUNCTION-QUIT" HARDWARE RESET: TI Basic and Extended Basic has two ways to exit, one by typing in "BYE" which will properly close all files, or by pressing "Function=(QUIT)". The latter method really should not be used at all since files will not be closed and unpredictable things can happen if function quit is pressed while files are open. Unfortunately, many of us had the nasty experience of accidentally hitting "Function Quit" with the result that everything in memory was lost and files were scrambled. IF you have Extended Basic and 32K memory, the following will disable "Function Quit": CALL INIT::CALL LOAD(-31806,16). This can be typed in as a direct command, or could be the first line of an extended basic program.

2. HOW TO SPEED UP EXTENDED BASIC. While XB offers faster execution speed for some applications compared to console basic, XB can be speeded up even further by disabling sprite graphics (naturally this works only if the program does not use sprite graphics). The program statement is: CALL INIT::CALL LOAD(-31878,0). There are several different releases or versions of Extended Basic and the speed-up effect will be more pronounced with some versions than with others. 32K memory is required.

3. HOW TO RECOVER MEMORY IN TI BASIC/EXTENDED BASIC WITH DISK DRIVE ATTACHED. The TI operating system automatically sets aside memory to serve three concurrent open files. A minimum of 534 bytes of memory are taken up by general expansion overhead plus 518 more bytes for each of the three files opended up by default, or a total of just about 2K. If you know that you will have only one file open, key in the following DIRECT COMAND: CALL FILES(1)(Press ENTER) NEW (Press ENTER). This sequence will recover 1K of precious memory. Please note that this sequence can be keyed in as a command only and cannot be used as a program statement. Don't forget the NEW or results will be unpredictable. This prodecure can be used with both TI Basic or Extended Basic. With TI Basic and attached disk this is more essential than ever since TI Basic will only address 16K and you can ill afford to lose much of that.

TIBase Topic - TUTORing #01
by Tony Kleen, Guilford TI99er Users Grp

Article 08; Copyrighted August 1991
Reprints are encouraged! Let's use it.

I have been mulling over various approaches to tutor a novice, and have decided that Dennis Faherty's TUTOR diskette does a marvelous job. I plan to utilize his approach (and his words), and intend to add additional text and examples to what I've plagurized from Dennis.

When you receive the TIBase software, you also receive a diskette labeled TUTOR III. To activate the lesson plan, you simply insert this diskette and enter DO TUTOR at the .DOT prompt. Dennis's 'README' file aptly describes what to do:

TI-BASE TUTOR II DISK

This is a supplementary disk to version 2.0 of TI-BASE. It contains information which is organized into logical categories to help in understanding the functions of TI-BASE.

This disk also contains the sample data-bases which are desscribed in the users manual.

To access the data on this disk you should execute a command file named TUTOR from TI-BASE. Proceed as follows:

- o Load TI-BASE
- o Enter the date at the prompt
- o Using the SET directive, set DATDISK to the device that contains the TUTOR disk. i.e. SET DATDISK=DSK2.
- o Invoke TUTOR by entering DO TUTOR
- o Select the category you wish information on and press ENTER
- o Repeat as desired

(*)

Ignore the reference to 'version 2.0'. This simply hasn't been edited to read 'version 3.02'. Also, this is 'TUTOR III' diskette, not 'II'.

Notice that your first activity is to load the TIBase. TIBase may be loaded using the Editor/Assembler, Extended Basic, or Mini-Memory modules. Also, Funnelweb, RAVE OS, and Horizon OS can load TIBase. I've used all but the Horizon OS and have had no problems.

Loading is EXTREMELY slow from the mechanical disk drives. By all means, if you are fortunate to own RAMdisks, place the TIBase files on your RAMdisk and load from there. It takes 2-3 seconds from RAMdisk and 20-30 seconds from diskette.

Your easiest method for loading? Use the XB cartridge, place the TIBase software disk in drive #1, and then select the XB cartridge from your screen. This way, you automatically use the DSK1.LOAD file which runs the TIBASEB file.

There is a benefit to loading with the mini-memory cartridge. You get the use of an additional 4K of RAM memory. TIBASE will utilize this memory. As a novice, this won't benefit you, but when you start using multiple databases, with each database using their maximum 17 fields, you may need the extra 4K memory. More on mini-memory, later, when we talk about buffers and memory usage.

When TIBase has loaded, the prompt;

'enter date (mm/dd/yy)'

is displayed. Respond by entering the date in the mm/dd/yy format. I always enter all six digits, ie., '01/01/91' instead of '1/1/91'.

In order for TUTOR to function properly, the disk drive that the TUTOR III diskette resides on must be designated as the DATDISK. The DATDISK is TIBase's default device for the data files and the command files. If you have a two drive system, you could leave the TIBase software diskette in DSK1 and place the TUTOR III diskette in DSK2. If you do place TUTOR III in DSK2, you need to inform TIBase by entering:

'SET DATDISK DSK2'.

By default, the DATDISK is set to DSK1. So, alternatively, after entering the date at the prompt, you could instead remove the TIBase diskette and insert the TUTOR diskette. Then, invoke the TUTOR by entering DO TUTOR. If you have only a one drive system, this, of course, is your only alternative!

Okay, let's say we got TUTOR invoked. Here's what you get as a menu:

T U T O R

0= Exit	D= Printing Data
1= Introduction	E= Sorting Data
2= Status display	F= Finding data
3= Create data-base	G= Summing data
4= Use a data-base	H= Command files
5= Appending data	I= Modify commands
6= Editing data	J= Converting data
7= Modify structure	K= Disc functi
9= Replacing data	M= Install area
A= Literal data	N= Macro's
B= Math notation	O= Reports
C= Display	P= Control keys

Select choice and ENTER

TUTOR 019 1 00000/00000

(*)

Simply select the choice you wish to review, and be prepared to read faster than I can. The paging of the text is somewhat faster than I can read/comprehend. You can solve this problem relatively easily. Just revise the command file to use a longer wait state (which, if you are a novice, I don't recommend at this point); or you can read along in this tutorial. I've simply printed each topic for your review. So, if you're wondering what's going to be covered in next month's article, just scan down the TUTOR menu.

Here's the introduction:

INTRODUCTION

WELCOME TO TI-BASE

First a few words about the screen.

When you are prompted by a . and a flashing cursor, you are on the main entry screen and any directive may be entered.

If you are prompted by a ?, it is in response to a prompt which needs to be answered.

All directives are in UPPER case.

The hi-lited info line contains data from left to right as follows:

- -command filename and line number.
- -select slot number
- -data-base name,rec.num.,total rec.
- -end of file indicator
- -insert character indicator
- -activity indicator:
  - O = file being opened
  - R = file being read
  - C = file being closed
  - W = file being written
  - S = file sending status
  - * = system in pause state

Most sequential operations such as displays, command file operations, etc. may be suspended by depressing the space bar until an * appears on the info line. It may be restarted by depressing the "S" key or aborted by the ESCAPE key (F9).

TI-BASE is designed to permit the collection and manipulation of a wide variety of data. Basically the user is allowed to CREATE data-bases and define the STRUCTURE of each record in the data-base.

Once the data-base has been defined, data may be added by APPENDing records. After the desired data has been entered, it may be displayed or manipulated as necessary.

There may be up to 5 data-bases active by selecting slots 1-5.

Data may be freely interchanged via the REPLACE directive.

A command language has been provided so that repetitive operations may be performed.

----------(*)

An example of the 'info line' is given on the last line of the menu that I've displayed earlier.

'TUTOR' is the command file name.

'019' is the line number of the command file. This number and a listing of the command file can help you determine which directive you're executing.

The data base name is left blank, because, at the time of the menu snapshot, no database was in use.

'00000' is displayed as the record in use. Zeroes are displayed if no database is in use. Zero, however, is a valid record number if a database is in use. Record numbering starts at zero!

'/' is a delimiter.

'00000' is displayed as the number of records in the database.

The 'end of file indicator' is either blanks or 'eof'.

The 'insert character indicator' is either blanks or 'ins'.

The activity indicator is the last character on the line. The valid values are listed above in the INTRODUCTION topic.

One more category this article:

STATUS

The system status may be viewed by entering DISPLAY STATUS. The items displayed are default values and may be modified via the SET directive.

SET (keyword)=(value)

SET DATDISK=DSK2.

DATDISK = default device for data files and command files.
PRGDISK = default device for program segments and help screens.
PRINTER = printer attributes.
PAGE = printer lines/page
HEADING = display heading on/off
TALK = command echo on/off
SPACES = space between display field
RECNUM = display rec # on/off
LSPACE - local variable space
CURSOR = cursor speed (0-99)
DATE = current date
FORMAT = (blank) default format
INVERSE = write inverse on/off

----------(*)

Here's an example of doing a DISPLAY STATUS on my system. Notice that I do not use disk numbers but instead use disk names. Also note that I do not have the default system statuses as provided by TIBase. I have previously altered these statuses with the SET directive.

```
.DISPLAY STATUS                          014
-----------------------------------------
DATDISK = DSK.DATA
PRGDISK = DSK.BOOT
PRINTER = RS232.BA=9600.PA=N.DA=8.TW
PAGE    = 056
HEADING = ON
TALK    = OFF
SPACES  = 01
RECNUM  = ON
LSPACE  = 0512
CURSOR  = 02
DATE    = 08/11/91
CRLF    = ON
INVERSE = OFF
-----------------------------------------
.SNAP                                    004

        000 1         00000/00000
-----------------------------------------
```

That's it for this article. I know it's not much coverage, yet, but it does show you that there's nothing to getting TIBase up; and nothing to getting the TUTOR diskette usable. SO, lets dust off that copy of TIBase and start learning how to use this fine DBase software.

FOOTNOTE: (*) These sections were copied from Dennis Faherty's TUTOR diskette, or the TIBase Users Guide.

```
100 REM ********************
110 REM * BARS AND GRAPHS *
120 REM ********************
130 REM BBY JOHN GUNTER
140 REM HOME COMPUTER MAGAZI
NE
150 REM VERSION 4.3.1
160 REM TI BASIC
170 CALL CLEAR
180 PRINT TAB(7);"BARS AND G
RAPHS": : : : : : :
190 INPUT "USE A PRINTER (Y/
N)?":P$
200 IF (P$<>"Y")&(P$<>"N")TH
EN 190
210 IF P$="N" THEN 290
220 PRNT=1
230 INPUT "PRINTER DEVICE NA
ME?":DEV$
240 GOTO 290
250 CALL KEY(0,K,S)
260 IF (S=0)+(K<>13)>THEN 25
0
270 CALL CLEAR
280 RETURN
290 PRINT TAB(10);"PLEASE WA
IT": :
300 PRINT TAB(12);"GOTTA RES
T"
310 DIM CO(32)
320 DIM A$(11)
330 A$(1)="00000000000007E"
340 A$(2)="0000000000007E7E"
350 A$(3)="00000000007E7E7E"
360 A$(4)="000000007E7E7E7E"
370 A$(5)="0000007E7E7E7E7E"
380 A$(6)="00007E7E7E7E7E7E"
390 A$(7)="007E7E7E7E7E7E7E"
400 A$(8)="7E7E7E7E7E7E7E7E"
410 A$(9)="0101010101010101"
420 A$(10)="0101010101010FF
"
430 A$(11)="FF0101010101010
1"
440 CALL SCREEN(13)
450 CALL COLOR(9,5,1)
460 CALL COLOR(10,9,1)
470 CALL COLOR(11,11,1)
480 CALL COLOR(12,15,1)
490 COL(1)=5
500 COL(2)=9
510 COL(3)=11
520 COL(4)=15
530 FOR X=95 TO 158 STEP 8
540 FOR CH=1 TO 8
550 CALL CHAR(X+CH,A$(CH))
560 NEXT CH
570 NEXT X
580 CALL CHAR(42,"935539FF39
5593FF")
590 CALL CHAR(91,"0000000000
0000FF")
600 CALL CHAR(92,A$(9))
610 CALL CHAR(93,A$(10))
620 CALL CHAR(94,A$(11))
630 CALL CLEAR
640 SET=0
650 CALL VCHAR(1,7,92,21)
660 FOR X=1 TO 21
670 CALL HCHAR(X,8,91,25)
680 NEXT X
690 CALL HCHAR(21,8,93,24)
700 FOR TIC=5 TO 20 STEP 5
710 CALL HCHAR(TIC,7,93)
720 NEXT TIC
730 CALL HCHAR(1,7,94)
740 ST$="MAXIMUM DATA VALUE
? "
750 GOSUB 770
760 GOTO 1190
770 D1$=D$
780 D$=""
790 FOR ST=1 TO LEN(ST$)
800 CALL HCHAR(24,2+ST,ASC(S
E6$(ST$,ST,1)))
810 NEXT ST
820 KI=1
830 CALL SOUND(100,800,2)
840 CALL HCHAR(23,7,42)
850 CALL KEY(0,K,S)
860 IF S=0 THEN 850
870 IF K=13 THEN 1140
880 IF K=7 THEN 1080
890 IF K=8 THEN 980
900 IF K=9 THEN 930
910 IF K<>32 THEN 1030
920 IF (ST$="SIDE LABEL ? ")
+(ST$="BOTTOM LABEL ? ")+(ST
$="LEGEND 1")+(ST$="LEGEND 2
")THEN 1030
930 IF M+1>32 THEN 840
940 CALL HCHAR(21,M,93)
950 M=M+1
960 CALL HCHAR(21,M,42)
970 GOTO 840
980 IF M-1<8 THEN 840
990 CALL HCHAR(21,M,93)
1000 M=M-1
1010 CALL HCHAR(21,M,42)
1020 GOTO 840
1030 IF KI>25 THEN 840
1040 CALL HCHAR(23,7+KI,K)
1050 D$=D$&CHR$(K)
1060 KI=KI+1
1070 GOTO 840
1080 CALL SOUND(300,300,2)
1090 FOR ES=KI TO 1 STEP -1
1100 CALL HCHAR(23,7+ES,32)
1110 NEXT ES
1120 D$=""
1130 GOTO 820
1140 CALL HCHAR(23,1,32,32)
1150 CALL HCHAR(24,1,32,32)
1160 IF D$<>"" THEN 1180
1170 D$=D1$
1180 RETURN
1190 FOR I=1 TO LEN(D$)
1200 IF (ASC(SEG$(D$,I,1))<4
8)+(ASC(SEG$(D$,I,1)))>57)THE
N 1240
1210 NEXT I
1220 IF (LEN(D$)<1)+(LEN(D$)
>5)THEN 1240
1230 GOTO 1260
1240 GOSUB 1870
1250 GOTO 740
1260 MD=VAL(D$)
1270 SCA=(INT(MD/20)*20)+20
1280 IF INT(MD/20)<>MD/20 TH
EN 1300
1290 SCA=SCA-20
1300 DIV=100
1310 LA$=STR$(SCA)
1320 FOR MK=1 TO 21 STEP 5
1330 FOR LAB=1 TO LEN(LA$)
1340 CALL HCHAR(MK,2+LAB,ASC
(SEG$(LA$,LAB,1)))
1350 NEXT LAB
1360 DIV=DIV-25
1370 SCA1=SCA*(DIV/100)
1380 LA$=STR$(SCA1)
1390 NEXT MK
1400 ST$="SIDE LABEL ? "
1410 GOSUB 770
1420 IF LEN(D$)<25 THEN 1440
1430 D$=SEG$(D$,1,24)
1440 FOR LAB=1 TO LEN(D$)
1450 CALL VCHAR(LAB,2,ASC(SE
G$(D$,LAB,1)))
1460 NEXT LAB
1470 ST$="BOTTOM LABEL ?"
1480 GOSUB 770
1490 R=22
1500 Y=7
1510 GOSUB 1530
1520 GOTO 1570
1530 FOR LAB=1 TO LEN(D$)
1540 CALL HCHAR(R,Y+LAB,ASC(
SEG$(D$,LAB,1)))
1550 NEXT LAB
1560 RETURN
1570 M=8
1580 ST$="COLOR?1-BLU,2-RED,
3-YEL,4-GY"
1590 GOSUB 770
1600 IF (ASC(D$)<49)+(ASC(D$
)>52)+(LEN(D$)>1)THEN 1580
1610 C=VAL(D$)-1
1620 IF SET=0 THEN 1770
1630 CH=127
1640 CH1=135
1650 SE=SET-1
1660 CST=13+SET-1
1670 IF CST<17 THEN 1790
1680 CST=CST-1
1690 SE=SE-1
1700 CALL SOUND(300,300,2)
1710 D$="COLOR FULL-ENTER ER
ASE OR END"
1720 Y=1
1730 R=24
1740 GOSUB 1530
1750 ST$=""
1760 GOTO 1800
1770 CH=95
1780 CH1=103
1790 ST$="ENTER DATA"
1800 CALL HCHAR(21,M,42)
1810 CALL HCHAR(21,M-1,93)
1820 GOSUB 770
1830 IF D$<>"C" THEN 1900
1840 IF SET=0 THEN 1580
1850 SET=SET+1
1860 GOTO 1580
1870 CALL SOUND(300,300,2)
1880 ST$="BAD DATA-TRY AGAIN
"
1890 RETURN
1900 IF D$="END" THEN 2330
1910 IF D$<>"ERASE" THEN 196
0
1920 CALL HCHAR(21,M,93)
1930 CALL VCHAR(2,M,91,19)
1940 CALL HCHAR(1,M,95)
1950 GOTO 1770
```

```
1960 IF (D$="P")$(PRNT=1)THE
N 2560
1970 IF D$="NEW" THEN 630
1980 FOR X=1 TO LEN(D$)
1990 D=ASC(SEG$(D$,X,1))
2000 IF (D<47)+(D>57)THEN 20
30
2010 NEXT X
2020 GOTO 2050
2030 GOSUB 1870
2040 GOTO 1820
2050 D=VAL(D$)
2060 IF D>SCA THEN 2080
2070 GOTO 2100
2080 GOSUB 1870
2090 GOTO 1820
2100 BAR=D/(SCA/20)
2110 HT=INT(BAR)
2120 RE=BAR-HT
2130 TP=1+INT((RE*8)+.5)
2140 IF TP<9 THEN 2160
2150 TP=8
2160 SE=C
2170 CALL CHAR(CH+TP+(SE*8),
A$(TP))
2180 CALL CHAR(CH1+(SE*8),A$
(B))
2190 CALL VCHAR(21-HT,M,CH1+
(SE*8),HT)
2200 CO(M)=C
2210 IF HT>19 THEN 2250
2220 CALL HCHAR(20-HT,M,CH+T
P+(SE*8))
2230 CALL VCHAR(1,M,91,19-HT
)
2240 GOTO 2260
2250 CALL HCHAR(21-HT,M,CH1)
2260 IF M<32 THEN 2310
2270 CALL SOUND(300,300,2)
2280 ST$="DATA FULL"
2290 GOSUB 770
2300 GOTO 1790
2310 M=M+1
2320 GOTO 1790
2330 CALL SOUND(500,800,2)
2340 ST$="LEGEND 1"
2350 GOSUB 770
2360 D2$=D$
2370 ST$="LEGEND 2"
2380 GOSUB 770
2390 D3$=D$
2400 D$=D2$
2410 R=23
2420 Y=7
2430 GOSUB 1530
2440 D$=D3$
2450 R=24
2460 Y=7
2470 GOSUB 1530
2480 CALL KEY(0,K,S)
2490 IF S=0 THEN 2480
2500 IF K=80 THEN 2560
2510 IF K=78 THEN 630
2520 IF K=83 THEN 2540
2530 GOTO 2480
2540 CALL CLEAR
2550 END
2560 OPEN #1:DEV$
2570 FOR X=1 TO 24
2580 FOR P=1 TO 32
2590 CALL GCHAR(X,P,Z)
2600 IF Z<96 THEN 2620
2610 Z=35
2620 IF (Z<91)+(Z>94)THEN 26
40
2630 Z=95
2640 PRINT #1:CHR$(Z);
2650 NEXT P
2660 PRINT #1:""
2670 NEXT X
2680 CLOSE #1
2690 GOTO 1820
```

```
100 REM CHARACTER DEFINITION
110 DIM B(8,8)
120 CALL CHAR(100,"")
130 CALL CHAR(101,"FFFFFFFFF
FFFFFFF")
140 CALL COLOR(9,2,16)
150 CALL CLEAR
160 M$="AUTO CHARACTER DEFIN
ITION"
170 Y=3
180 X=4
190 GOSUB 770
200 M$="12345678"
210 Y=8
220 GOSUB 770
230 GOSUB 820
240 M$="0=OFF=WHITE"
250 Y=22
260 X=4
270 GOSUB 770
280 M$="1=ON=BLACK"
290 Y=23
300 GOSUB 770
310 FOR R=1 TO 8
320 CALL HCHAR(8+R,5,100,8)
330 NEXT R
340 FOR R=1 TO 8
350 FOR C=1 TO 8
360 CALL HCHAR(8+R,4+C,30)
370 CALL KEY(0,KEY,STATUS)
380 IF STATUS=0 THEN 370
390 IF (KEY<>8)+(KEY<>9)=-2
THEN 420
400 GOSUB 870
410 GOTO 360
420 KEY=KEY-48
430 IF (KEY(0)+(KEY)1)(=-1 T
HEN 370
440 B(R,C)=KEY
450 CALL HCHAR(8+R,4+C,100+K
EY)
460 NEXT C
470 NEXT R
480 HEX$="0123456789ABCDEF"
490 M$=""
500 FOR R=1 TO 8
510 LOW=B(R,5)*8+B(R,6)*4+B(
R,7)*2+B(R,8)+1
520 HIGH=B(R,1)*8+B(R,2)*4+B
(R,3)*2+B(R,4)+1
530 M$=M$&SEG$(HEX$,HIGH,1)&
SEG$(HEX$,LOW,1)
540 NEXT R
550 CALL CHAR(102,M$)
560 CALL HCHAR(8,20,102)
570 FOR R=0 TO 2
580 CALL HCHAR(12+R,20,102,3
)
590 NEXT R
600 Y=16
610 X=12
620 GOSUB 770
630 M$="PRESS 0 TO QUIT"
640 Y=18
650 X=12
660 GOSUB 770
670 M$="PRESS ANY OTHER"
680 Y=19
690 GOSUB 770
700 M$="KEY TO CONTINUE"
710 Y=20
720 GOSUB 770
730 CALL KEY(0,KEY,STATUS)
740 IF STATUS=0 THEN 730
750 IF KEY<>81 THEN 140
760 STOP
770 FOR I=1 TO LEN(M$)
780 CODE=ASC(SEG$(M$,I,1))
790 CALL HCHAR(Y,X+I,CODE)
800 NEXT I
810 RETURN
820 FOR I=1 TO LEN(M$)
830 CODE=ASC(SEG$(M$,I,1))
840 CALL HCHAR(Y+I,X,CODE)
850 NEXT I
860 RETURN
870 CALL HCHAR(8+R,4+C,100+8
(R,C))
880 IF KEY=9 THEN 960
890 C=C-1
900 IF C<>0 THEN 1020
910 C=8
920 R=R-1
930 IF R<>0 THEN 1020
940 R=8
950 GOTO 1020
960 C=C+1
970 IF C<>9 THEN 1020
980 C=1
990 R=R+1
1000 IF R<>9 THEN 1020
1010 R=1
1020 RETURN
```