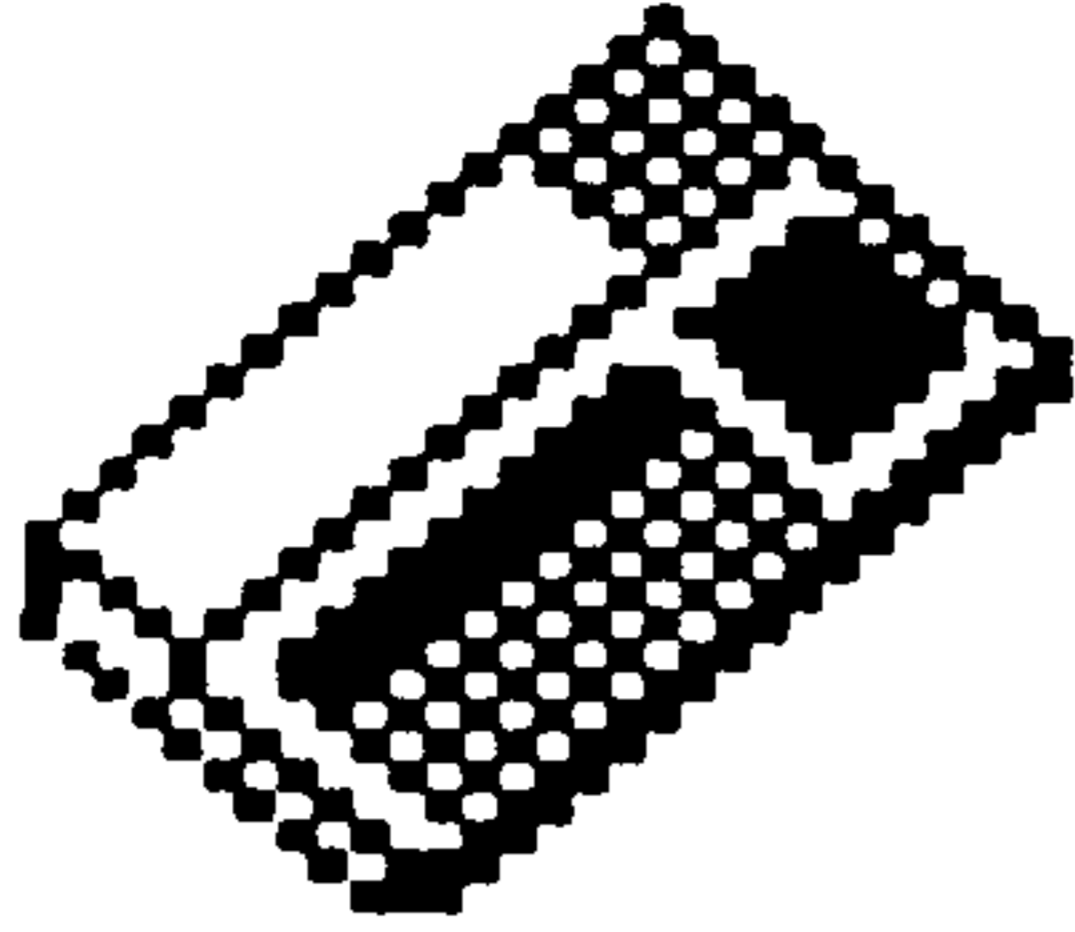


PROP. of HUG
SET "A"
c/o R. Lumpkin
Houston Texas
713-469-5089



HUG

HOUSTON USERS GROUP

AUGUST
1985

MEETING SCHEDULE

FIRST SUNDAY OF EVERY MONTH
(2nd Sunday if 1st Sunday
is on a holiday weekend)

HUG TIBBS - (713) 487-5530
24-hour BULLETIN BOARD

ANNUAL PICNIC AT THE NEXT MEETING

SUNDAY, AUGUST 14, 1985 2:00 P.M.

Come join us for the annual picnic of the Houston Users Group in Hermann Park. Bring your family. Enjoy an afternoon with your fellow members, or enjoy the planetarium, museums, zoo, and pond that are located within the park. HUG has obtained a permit to use the Hermann Park pavilion. The pavilion is located over on the west side of the park close to Fannin Dr. Look for the HUG sign on the pavilion. You can get to Hermann Park by from Loop 610 south by taking the Fannin Rd. Exit and going north past the Astrodome and the Texas Medical Center. You can get there from the west by taking Holcombe Blvd. (changes to Bellaire Blvd at Stella Link) to Fannin. From the northwest, go into town on 54 westbound. Take the Greenbriar exit, go south on Greenbriar and then left on Sunset Blvd to Fannin. Right on Fannin to the pavilion. Please bring your own food and drinks. Please, no alcoholic beverages.

IN THIS ISSUE

FORTH TUTORIAL
HUG MINUTES
CABLE CONSTRUCTION

FIELDING RECORDS
SCREEN UTILITY
TI-WRITER TIP

1985 HUG OFFICERS

President --- BILL KNECHT ... 473-5713
VP/Membership DON LEWIS 353-5295
VP/Program -- SANDOR KARPATY 955-1138
VP/S.I.G. --- MARK CRUMP 467-2541
Exec. Asst. - TOM JAY 850-0222

Secretary - CHIA GREER ... 668-4500
Treasurer - DAVID MATHER . 941-1497
Librarian - LARRY PIPKIN . 499-9991
TIBBS/SysOp CECIL CROWDER unlisted
Editor ---- ROGERS MILLS . 930-0810

THIS NEWSLETTER IS PUBLISHED MONTHLY BY THE HOUSTON USERS' GROUP. ANY OPINIONS OR ENDORSEMENTS ARE THOSE OF THE AUTHOR, AND MAY NOT NECESSARILY REFLECT THE OFFICIAL OPINION OF 'HUG'. PERMISSION TO REPRINT GRANTED TO OTHER USER GROUPS. SUBSCRIPTION IS FREE WITH MEMBERSHIP.

THE EDITOR'S CORNER

Well here is the second issue under my direction. I am sure that there is something for everyone or at least I hope there is. The last issue was mailed the day that I left for vacation. As always, the printing must go on schedule if at all possible. I'm giving Cecil Crowder twoatta'boys for the help that he gave me. Much thanks to him for his efforts.

In this issue I have articles from Mark Crump, Bill Knecht (El Presidente), a FORTH program with docs, from Jeff Stanford, and Cecil Crowder. That is worth anatta'boy for each of these fellows.

While on vacation, I visited the North West Fla 99 Users Group in Pensacola. They were happy to see the goodies that I have acquired while here in Texas and we all enjoyed some good fellowship. The supplies for the TI 99/4A have all but dried up in that part of the country. I was able to get at least one good deal while I was there so the trip was successful in more ways than one. The same day that Disk Manager 1000 arrived here was the same day that I received it in Pensacola! I have had no trouble in keeping up with the latest and greatest even when I am away from what appears to be the best source in the country.

There is a lot to be said about the support that is available here in the Houston area. Many things that are almost common place here are unheard of in many parts of the country. The exchange center is one advantage which I have taken several items to after I had a catastrophic event here a while back. That is not always readily available in other places. True, you can ship it off, but that takes extra time.

I literally computed across half of the country, while I was on my trip. The only place that I did not take the computer was the Florida keys. I spent the entire time in the Keys devoted to diving and snorkeling here. The water was crystal clear and the slight breeze kept the nights very pleasant indeed. If you have the time to go the trip is more than worth it. There are several avenues to take depending on your budget. From camping at the various locations, from Key Largo on down to Key West to the finest of hotels. You can bare-boat a cruise or fly out the Dry Tortugas. The reefs are the only tropical reefs on this continent. They are absolutely magnificent.

Back to computing. While in Pensacola, I had the chance to exchange a few programs and to bring the NW 99ER UG library up to date. As a member of both groups, I can say that they really stick together for there is no other way that they will survive unless they do. The HUG has it made in the shade in comparison but we must also beware of the same fate that could befall us.

K. Mills

The Editor

PRESIDENT'S REPORT

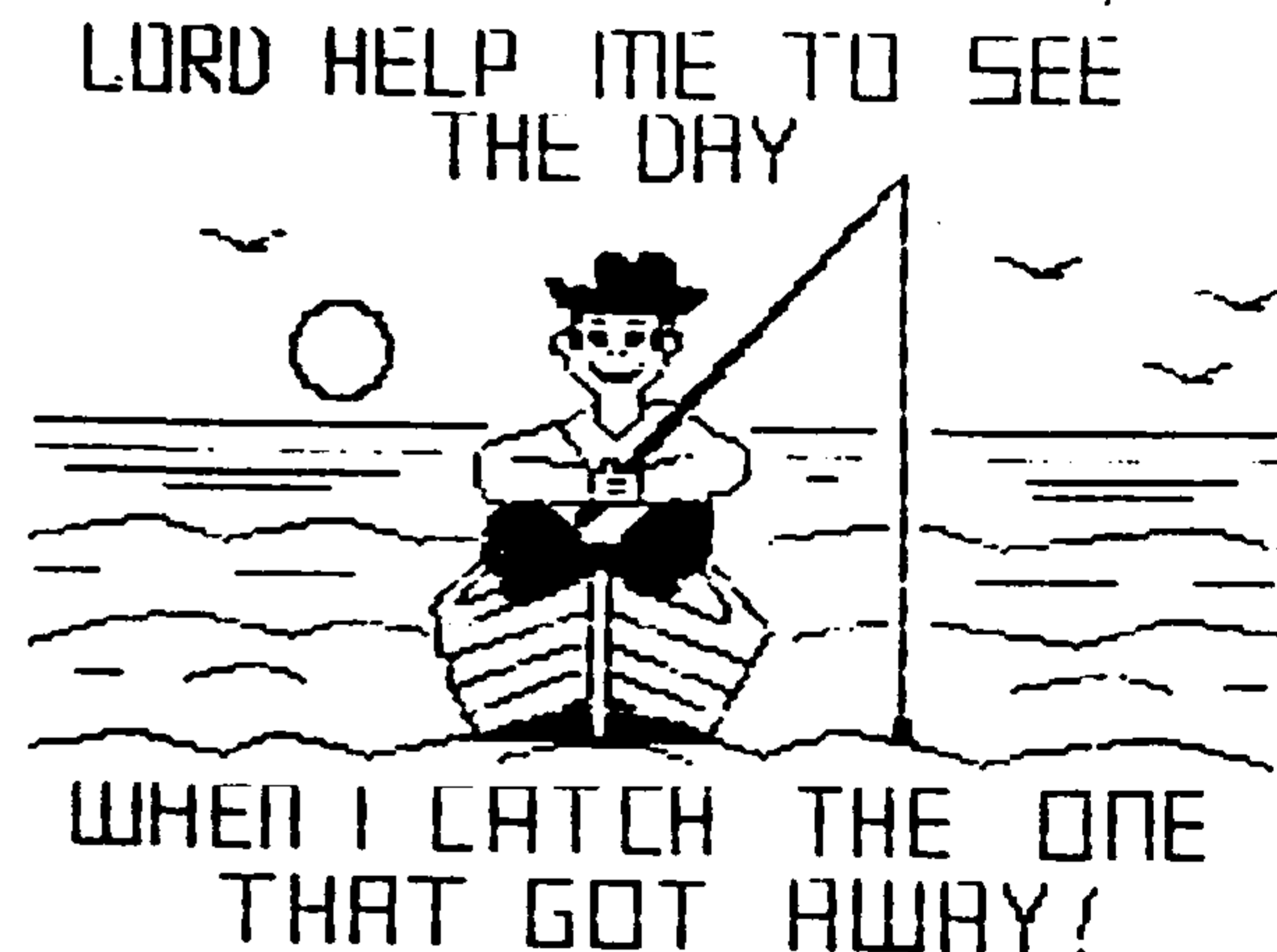
I would like to start this report by congratulating Don Lewis on an excellent program presented at the last HUG meeting. From comments I have heard, I feel every gained some knowledge and understanding from his presentation on array handling. As Sandor will agree, this program resulted from just one person's suggestion. How about some more suggestions to him.

Looking over the Library Addendum in this newsletter, I am sure you will realize that we are getting some fantastic programs in our library. Programs like TI Pilot, Itod, Fast Tern, Xmodem and Disk Manager 1000 would cost several hundred dollars if they were commercial programs and a lot more if you were buying them for another brand computer. Judging from the programs we have been adding, I believe that the members are missing out on a lot if they do not request at least one program every month. The library is there for you, so why not benefit from it.

The Board of Directors met this past month and I would like to thank them for their support. Since I have been a member of HUG, we have never had that good a turnout for a Board meeting. Eight of our ten members were on hand. Some of the things we discussed included the picnic, the constitution & by-laws, upcoming programs for meetings and any problem areas. I feel we really accomplished much in this hour and a half.

Hope to see all of you and your families at the picnic. Remember to bring enough food for yourself and the President.

Bill W. Knecht



-BwK

Fielding Fixed Length Records
by Mark A. Crump

Perhaps the thorn of Basic programming is the use of data files. The most important function of our computers is to process large amounts of information with incredible speed and 100% accuracy. Yet many novice programmers are stymied when they deal with data files. Many of these problems arise because the programming techniques are not covered in the manuals that come with the computer and cartridges and because TI Extended Basic does not provide organic support for fielding a record. I have resolved this problem by taking advantage of Extended Basic's powerful subprogram support. This article introduces to several ground rules to follow and two subprograms that you can merge to your own basic program to easily implement those rules.

If you think you do not have the necessary background to understand this article I suggest that you read the following remedial materials:

TI 99/4A User's Reference Manual: File processing pp. II-118,136.

Disk Memory System Manual: pp. 30-36 Extended Basic Reference Manual: User Written Subprograms, pp. 23-24, p. 113, p. 55, pp. 180-184. The fundamental unit of information is the file. A file is a collection of related information. Files are further subdivided into records. A record can be accessed in two modes. The first is sequentially, where you have to start reading a file with its first record and each record in the order that it appears in the file. The second mode is random access. In this mode, you can indicate the number of the record you want and read or write to it directly. Sequential records are useful for certain types of information such as the text produced by a word processor. Most programs that deal with information management will do so by random access. Random access is much faster although it can use more disk space than a sequential file holding the same amount of information.

Records are further subdivided into fields. A field has a fixed length and is identified within a record by its position in the record. Most programming problems occur because the programmer fails to properly define the fields within a record. Consider the following statement:

```
PRINT #1, REC $:A$,B,C
```

This statement prints a record that has three fields. The first field is a text field, the next two are

numerical fields. The problem is that the lengths of these fields have not been fixed. This may or may not cause problems later on when you try to retrieve this record.

Here are four rules that establish the way we do about fielding a record:

1. Read a record the same way you write a record.
2. Each record is divided into fields. Each field has a fixed length. The sum of the field lengths must be equal to the record length.
3. Numbers are to be rightset in their field. The last digit will be the rightmost character in the field. If the number of digits is less than the field length then the remainder of the field must be filled with zeros.
4. Text will be leftset in a field. The first character of text will occupy the leftmost position in the field. If the text length is less than the field length, then the remaining field space is to be filled with blanks.

The HUG library contains a file called fields. The fields file when merged with your extended basic program will provide you the leftset (LSET) and rightset (RSET) functions. LSET is accessed as:

```
CALL LSET(FL,X$) where FL is the field length of X$
The subprogram will return X$ with the information leftset
in it and LEN(X$) = FL.
```

RSET is accessed as:

```
CALL RSET(FL,X,X$) where FL is the field length .
is the number to be stored rightset in X$. the length of
X$ upon return is FL.
```

Here is an example of LSET and RSET being used in program to record amount paid by person into a record.

```
100 RL = 40 !RECORD LENGTH
```

```
110 NL = 30 !FIELD LENGTH OF THE NAME FIELD NAME
```

```
120 AL = 10 !FIELD LENGTH OF THE AMOUNT FIELD DOLLAR
```

```

130 FNS = "DISK.JOURNAL" FILE NAME
140 OPEN #1:FNS. DISPLAY. FIXED NL, RELATIVE
150 INPUT "AMOUNT PAID: ":DOLLAR
155 IF LEN(STR$(DOLLAR)) > AL THEN 150
160 INPUT "PAYEE: ":NMS
165 IF LEN(NMS) > NL THEN 160
170 CALL RSET(AL,DOLLAR,DOLLAR$) : FIELD THE RECORD
175 IF DOLLAR$="" THEN 200 ! FIELD TOO SMALL ABORT
PROGRAM
180 CALL LSET(NL,NMS)
190 PRINT #1. REC 1:NMS,DOLLAR$ : WRITE THE RECORD
200 CLOSE #1
210 END

```

Lines 100-120 are in fact constant declarations. The lengths of the fields and the record are established here. If I need to modify the structure of the file later on I can do so simply by changing the values in these lines.

After opening the file in line 140, the information is obtained from the keyboard. I filter the information in lines 150 and 165 to make sure it will fit the field length. Unpredictable results can occur if the information is larger than the field. RSET returns its a null string if the field is not large enough. Line 175 shows how to check for a null string.

I write the record to the disk buffer in 190. The record length is 40. The NMS field has 30 characters, the DOLLAR\$ field has 10 characters satisfying the requirement that the sum of the field lengths equal the record length.

Later retrieval of the record can be done in one of two ways:

1. Use input the same way you used print

Example: INPUT #1. REC 1:NMS,DOLLAR\$

2. Use Linout. Linout causes the entire record to be

read into a string. It will be easy to identify the fields within the string because each field has a fixed length. You can use the SEG\$ function to retrieve the individual fields.

Example:

```
LINPUT AS
```

```
NMS = SEG$(AS,1,30)
```

```
DOLLAR = VAL(SEG$(AS,31,10))
```

It would not be possible to extract the fields out of the record using LINPUT unless we had adhered to the four rules above.

I reserve record zero for a special purpose. I store information on the field lengths within the other records of the file in zero. All files have record zero fielded the same way. (Note that record zero is usually fielded differently than the other records in the same file). With record zero reserved for this purpose, I can have a standard "dump" program that can read any file and print its contents.!

I would like to close this article by commenting on TI's standard approach to handling fields within a record. Each field is preceded with its field length. When the Disk Memory System looks for the field, it reads the first byte as length of the following field. From that value it determines how many characters to read and the position of the next field. I would conclude that simply reading a record in the way it was printed is all that is necessary to guarantee that the record is retrieved. However I have had bad experiences depending on the Disk Memory System to identify the fields in the record. I have had several experiences where I had processed large amounts of information into a file but then unable to retrieve the information intact later on. I have experienced no retrieval problems when adhering to these conventions. I also get one advantage with this approach. The sorting of disk files is easily facilitated by right setting numbers in a field and left setting text in a field.

SCREEN CLEAR ROUTINES
by Bill Knecht

Here is the listing for the field procedures:

Many times you have heard me stress the need to clear your screen and not leave garbage at the end of your programs. Just adding CALL CLEAR where your program terminates will do the trick. But I have come up with a couple of routines which you can use instead of CALL CLEAR and add more effectiveness to the screen clearing. Note that the program commands are separated by double colons, but these routines work as well in BASIC by placing each command on a separate line.

The first routine will clear the screen like a curtain. The clearing starts from the edges of the screen and clears toward the center. CHR% 32 represents a blank space, but this characters may be either the blank or one that you redefine yourself.

```
J=33 :: K=0 :: FOR I=1 TO 16 :: J=J-1 :: K=K+1 :: CALL
VCHAR(1,J,32,24):: CALL VCHAR(1,K,32,24):: NEXT I ::
STOP
```

The next routine will clear the screen in a crunching effect. It starts clearing from all four sides toward the center.

```
J=25 :: K=0 :: L=33 :: FOR I=1 TO 12 :: J=J-1 :: K=K+1 ::
L=L-1 :: CALL HCHAR(K,1,32,32):: CALL VCHAR(1,L,32,24)::
CALL HCHAR(J,1,32,32):: CALL VCHAR(1,K,32,24):: NEXT I ::
STOP
```

If you want to include a demo with these routines, you can type in the following lines to fill the screen. Type them in before the routines.

```
CALL CLEAR :: CALL HCHAR(1,1,42,768):: FOR I=1 TO 1000 ::
NEXT I
```

These routines are not limited to using at the end of your program. They can be used anytime you need to clear the screen by using either a GOSUB or a CALL Subroutine. I hope you can use these as much as I do.

TI-WRITER TIP

To produce a header for an article is no real problem. The commands to do this task are very simple. All you need to do is set the printer to double width printing and have the overstrike produce the bold print.

The command for the double width print is ' control U shift N. The control U will give you an underline cursor and the 'N' gives the character. Then add CONTROL U FCIN R followed by a capital E to produce the for the overstrike.

```
1000 SUB LSET(L,A%)
1005 IF L<1 THEN 1070
1010 IF LEN(A%)>L THEN 1070
1020 FOR I=LEN(A%)+1 TO L
1030 A%=A%+ " "
1040 NEXT I
1050 A%=SEB$(A%,L)
1060 SUBEXIT
1070 DISPLAY AT(24,1):"WARNING: LSET MODIFICATION"
1080 GOTO 1050
1090 SUBEND
2000 SUB RSET(L,X,X%)
2005 IF L<1 THEN 3000
2010 X%=STR$(X)
2020 IF LEN(X%)>L THEN 3000 ELSE IF LEN(X%)=L THEN
SUBEXIT
2030 FOR I=LEN(X%)+1 TO L
2040 X%="0"X%
2050 NEXT I
2060 SUBEXIT
3000 DISPLAY AT(24,1)BEEP:"WARNING: RSET SUBPROGRAM.
NUMBER EXCEEDS FIELD LENGTH OUTPUT IS NULL."
3010 X%=""
3020 SUBEND
```

LSET is for fielding text. It requires the field length as the first parameter and the text string as the second. The text string is returned fielded.

RSET is for fielding numbers. It requires three parameters: the field length, the number to be fielded, and a return string. The return string will contain the number rightset upon return.

The major benefit to subprograms is that they give you a tool chest of procedures that can be added to your program. Therefore it will not be necessary to redesign these same procedures for every program. The problem comes with trying to prevent an error from occurring in the subprogram itself. Even though these subprograms are free of syntactical error they can suffer from logical errors if their input parameters are not correct. Logical errors occur in these programs if (1) the fieldlength passed is less than one or the information to be fielded is longer than the field length. If either occurs the subprograms will return a null string. Your main program should insure that the parameters passed are valid.

FORTH TO YOU. TOO

BY

LUTZ WINKLER

Introduction:

According to our source there are quite a few people out there who got the TI-FORTH disk and documentation when TI made them available to user groups. But not very many do much with it. Why? Well, the TI manual is not a tutorial, it assumes that you know something about FORTH. Though packed with useful information there are no instructions on how to begin. We will try to get you started from the very beginning. Hopefully we'll strike a happy medium, somewhere in between teaching and providing info that's useful to you.

What is FORTH?

There was much hype when it became available to TI users, some of it was overdone, but it is faster than BASIC and there are some advantages which will not immediately be evident to a beginner. FORTH is a TIL (Threaded Interpretive Language) and it'll be hard for you to believe that there is no GOTO command. If that is hard to swallow, there is more: It uses RPN or post-fix notation (RPN = Reverse Polish Notation). In other words, it's not $2 + 2$ that equals 4 but $2 2 +$. We'll find out more as we go along, for now let's just say that FORTH is very powerful, quite a bit faster than BASIC, compact, but perhaps more difficult to learn than BASIC. As a matter of fact, knowing BASIC may make it harder on you, because you'll be thinking BASIC until you get the hang of FORTH.

Setting started.

Before you do anything with your FORTH disk, get out the DISK MANAGER and make a backup copy. Do all your work and experimenting with this copy unless you are prepared to get a new FORTH disk. Now plug in the EDITOR/ASSEMBLER, opt for 3 (LOAD AND RUN) and enter DSK1.FORTH. After a moment the screen shows "BOOTING..." which is soon replaced by a menu. These are the LOAD options. For right now you need to concern yourself with only 2 of them: the normal or the 64-column editors. Your choice will depend on several factors: 1) your eyesight, 2) your monitor, and 3) how well you have adapted to using 'windows'. So jump right in and enter -64SUPPORT. After your disk drive is through you will see a tiny 'ok', meaning the 64 column editor has been booted. To see what your screen will look like type 34 EDIT (enter). If you can read what is displayed on your screen, you'll want to stay with -64SUPPORT. If it's hard on your eyes, settle for the 40 column editor. To get an idea what it looks like, hit FUNCT-9(ESCAPE), then enter TEXT COLD. FORTH will re-boot and when it is done, enter -EDITOR. (From now on, 'enter' will mean to type in the word followed by the ENTER key.) Again enter 34 EDIT to see what your 40 column editor looks like.

Programming in Forth consists of editing SCREENS,

such as that number 34 screen you called up for editing. But we are not ready for that, yet. Hit ESCAPE (F-9) and enter FLUSH and do this: Make yourself an overlay strip so you can edit easily. Keys and their functions are explained on page 5, chapter 3, of the TI-FORTH manual. Now here is another thing you might want to find out right now: a display color that suits you. Since you are still in the so-called 'interactive' mode of FORTH (no program is running) you can type this little ditty:

```
: SEE 252 22 DO I DUP . 7 VVTR KEY 2 = IF ABORT
ENDIF LOOP ;
```

After you get the ok type SEE. Don't worry if you can't read anything, at times the FG and BG colors match and there's nothing to be read. When you see a combination which gives you a good screen display, write down the last number (bottom of the screen) and continue to step through the loop (or exit via FUNCT-2).

You have accomplished 2 things:

- 1) you know the editor you want,
- 2) you have chosen a screen color END SESSION 1

FORTH READY TO RUN PROGRAMS

SCR #204

```
0 ( Diamond Draw- An Original FORTH program by J. Volk 4/2/84 )
1 : AT GOTOXY ; : INSTR CLS 1 1 AT ." Diamond Draw " 1 3 AT ." by
2 John J. VOLK" 1 7 AT ." USE E,S,D,X,W,R,Z,C TO MOVE DIAMOND" 1 9
3 AT ." 'Q' TO CHANGE DIAMOND COLOR" 1 11 AT ." 'O' FOR DRAW ON--
4 'F' FOR DRAW OFF" 1 13 AT ." '.' TO CHANGE BACKGROUND COLOR" 1 2
5 0 AT ." ----- HAVE FUN ! -----" 1 23 AT ." ANY KEY TO
6 START" BEGIN ?KEY 0 > UNTIL ; 1 VARIABLE STAT 2 VARIABLE SCOLR
7 124 VARIABLE YPOS 1 VARIABLE SCRCOLOR 94 VARIABLE XPOS HEX : SET
8 UP GRAPHICS2 3800 ' SATR ! 2000 SDDT 20 DCOLOR ! 1028 4482 8244
9 2810 0 SPCHAR YPOS @ XPOS @ SCOLR @ 0 0 SPRITE ;
10 : DELAY 500 0 DO I DROP LOOP ;
11 : STATEON 1 STAT ! ; : STATEOFF 0 STAT ! ;
12 : SOCHANGE 10 DCOLOR +! DCOLOR @ FO > IF 00 DCOLOR ! ENDIF 1 SCO
13 LR +! SCOLR @ F > IF 0 SCOLR ! ENDIF SCOLR @ 0 SPRCOL DELAY ;
14 DECIMAL : BCHANGE 1 SCRCOLOR +! SCRCOLOR @ 15 > IF 0 SCRCOLOR !
15 ENDIF SCRCOLOR @ SCREEN DELAY : -->
```

UNDER CONSTRUCTION

There comes a time when the computer enthusiast expands his computer out to the ultimate system. (more than one disk drive) He chooses the route that he wishes to take and then proceeds to read the latest adds on disk drives, power supplies and cabinets. The choices are staggering! There is so much to chose from now! Just last year the cost of a single sided drive meant taking out a second mortgage on your house. Now for what it costs to fill your gas tank up twice, you can purchase a brand new, never out of the box with new smell even, disk drive.

So now comes the decision time. Just what kind of a disk drive do you want? You can get single sided drives from those who are doing a complete upgrade of their system or you can get a double sided drive. Just remember one thing, what ever you chose, keep your back up disks in the format of the least sophisticated drive and spare disk controller if you have one. This allows you to use the programs when you main hardware is down for repairs. The next choice is double density if you wish to go the change of disk controllers. Then there is the selection of 1/2 heights, full height, or 2/3 height drives. Also there are the 3 1/2 inch drives that are compatible with the 5 1/4 drives. Ok, now you have made up your mind, spend the money, and the box has arrived in the mail. But you do not have a cable!

No problem, for that is just what this article is about. Building the cables. There are two cables that you will need for the second drive. The ribbon cable and the power cable. If you have purchased a disk drive cabinet, you will not need to make a separate power cable. If you are building a cabinet or you are putting a set of half heights in the P-BOX then you will need the power cable.

To make the disk controller ribbon cable you will need at least two connectors and a piece of ribbon cable. The connectors can be purchased at Radio Shack. They do not stock the right ribbon cable. You will need a ribbon cable that has 34 conductors in it. You can get this cable from a local electronics supply house (other than Radio Shack). The connectors should be the type that you press on. The solder on type is more like work! Now all you need is a vice (sinking is not what I am referring to) to squeeze the connector on to the ribbon cable. Start with the large edge connector first. Carefully align the ribbon with the two halves of the connector and place the cable, sandwiched between the connector parts, in between the the jaws of the vice. Now tighten the vice carefully until the two parts of the connector are seated. Now remove the cable from the vice. This completes this half of the cable.

The next operation is to put the 34 pin dip connector on. This is done the same way as the larger connector was done, except there is a bit of caution here. You must now get the second connector on in the right manner. There are 4 different ways to do this of which only one

will be right! Take a careful look at the 34 pin dip connector. It is constructed with a guide key on one side of the connector. If you look directly at this side, pin 1 and pin 2 will be on you right and pin 33 and 34 will be on your left. Look at the large cable connector and you will see that the pins are numbered. You must match these up electrically or the cable will not work! Just remember MURPHY'S LAW applies at this very moment! Follow the diagram below and you should not have any problems. Once you have pressed the second connector on the cable will look like figure 2. Now the part left over is the strain relief. Install it by bending the cable back over the connector as illustrated and press the strain relief on. The cable will now appear like figure 3. This cable will not a a guide pin on the drive side unless you install one so pay attention as to how you put it on the drive's edge connector.

The power cable must be modified if you are going to use two drives in the P-BOX. You have three options to choose from. First you may wish to just purchase a 'Y' cable from D C Electronics.

The second choice is to buy a power cable from RADIO SHACK and splice it in with a crimp-on connector like you would use to connect up a boat trailer. This is not as neat of an installation but it will work quite well. One of my own systems has this modification.

The third way is to get the connector and build the cable from the ground up. I had some difficulty in obtaining the correct connector but I found a substitute that is working very well. It started on the day I received my second drive. At that time I did not know of the TI vendors in this area. So I picked up a 4 pin solex connector that was the same physical size as the one used in the P-BOX. One of the corners had to be trimmed off to allow for a proper fit. I then soldered the power wires to the connector and then used the crimp splice to add the cable to the P-BOX. The last thing to do was to add the second 34 pin edge connector to the existing ribbon cable. This was accomplished in a matter of minutes. The total time to convert the cable system was 1 hour and 45 minutes, to include the time it took to go to the Shack and get the parts. All that was left was to drill two holes for the screws to secure the disk drive in its' new home. (I started with a 1/2 height in anticipation of the addition of a second drive to my P-BOX.)

PLEASE, if you plan to follow these directions, read and look at the drawings CAREFULLY! If you have any doubts, call someone who has done this before. I as an electronic technician so some things are easy for me to do than it would be for the novice. As with all construction projects that are listed in this newsletter, this is a build it at YOUR OWN RISK! When in doubt ask questions. I can not be responsible for the work done by others. I hope that this article will help you.

R. MILLS

SCR #207

```

0 ( Diamond Draw- Screen 2 ) DECIMAL
1 : CHECK XPOS @ DUP 0 < IF 1 XPOS ! ENDIF 180 > IF 180 XPOS ! END
2 IF YPOS @ DUP 0 < IF 1 YPOS ! ENDIF 250 > IF 250 YPOS ! ENDIF ;
3 : UP -1 XPOS +! CHECK YPOS @ XPOS @ 0 SPRPUT STAT @ 1 = IF YPOS
4 @ 3 + XPOS @ 8 + DOT ENDIF ;
5 : DOWN 1 XPOS +! CHECK YPOS @ XPOS @ 0 SPRPUT STAT @ 1 = IF YPOS
6 @ 3 + XPOS @ 3 + DOT ENDIF ;
7 : RIGHT 1 YPOS +! CHECK YPOS @ XPOS @ 0 SPRPUT STAT @ 1 = IF YPO
8 @ 5 @ XPOS @ 3 + DOT ENDIF ;
9 : LEFT -1 YPOS +! CHECK YPOS @ XPOS @ 0 SPRPUT STAT @ 1 = IF YPO
10 @ 5 @ 3 + XPOS @ 3 + DOT ENDIF ;
11 : LUP -1 XPOS +! -1 YPOS +! CHECK YPOS @ XPOS @ 0 SPRPUT STAT @
12 1 = IF YPOS @ 3 + XPOS @ 8 + DOT ENDIF ;
13 : RUP -1 XPOS +! 1 YPOS +! CHECK YPOS @ XPOS @ 0 SPRPUT STAT @ 1
14 = IF YPOS @ 3 + XPOS @ 8 + DOT ENDIF ;
15 -->
    
```

SCR #208

```

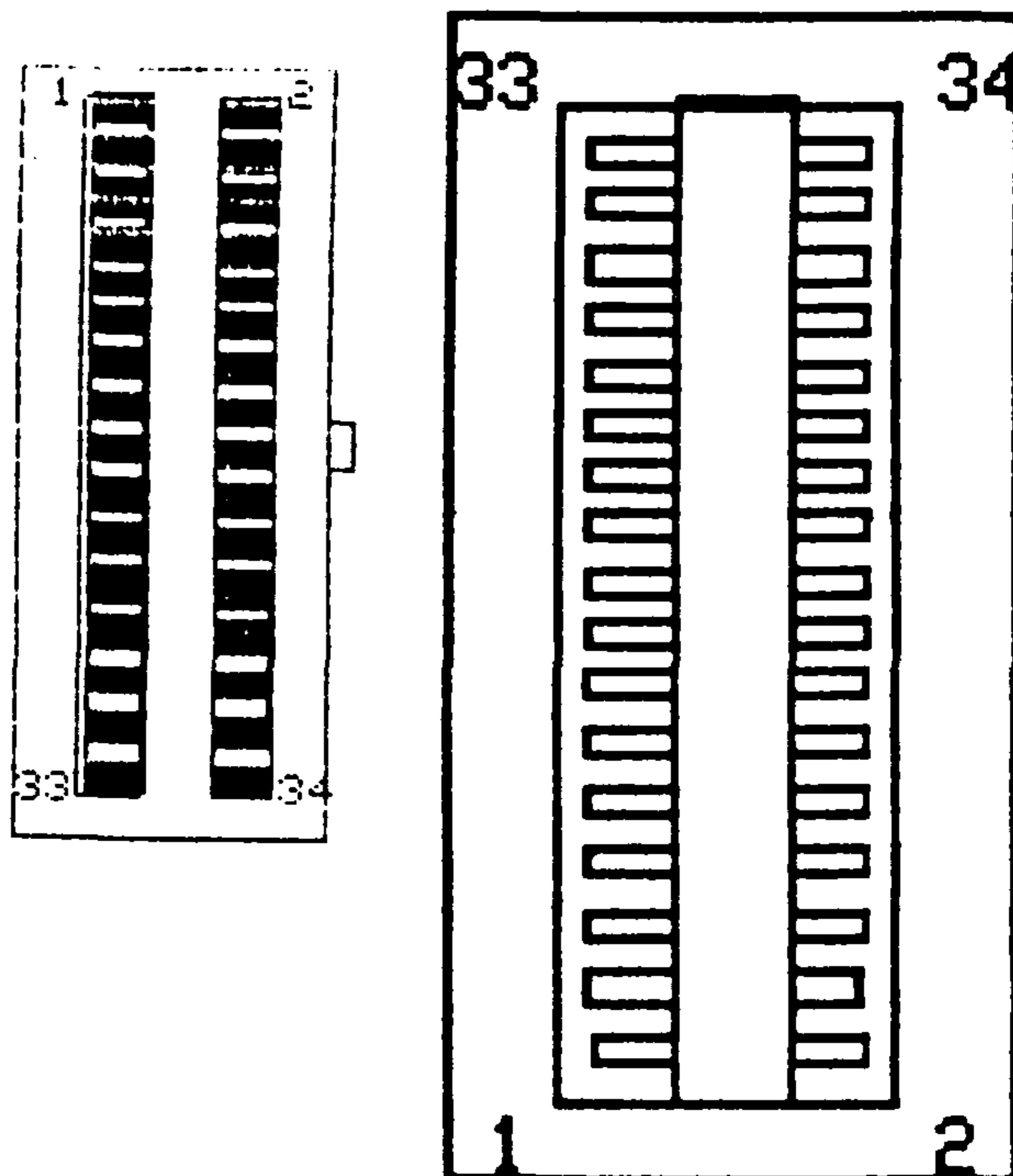
0 ( Diamond Draw-3rd Screen)
1 : LDOWN 1 XPOS +! -1 YPOS +! CHECK YPOS @ XPOS @ 0 SPRPUT STAT @
2 1 = IF YPOS @ 3 + XPOS @ 8 + DOT ENDIF ;
3 : RDOWN 1 XPOS +! 1 YPOS +! CHECK YPOS @ XPOS @ 0 SPRPUT STAT @
4 1 = IF YPOS @ 3 + XPOS @ 8 + DOT ENDIF ;
5 : INIT INSTR SETUP ;
6 : KEYIN3 CASE 69 OF UP ENDOF 88 OF DOWN ENDOF 68 OF RIGHT ENDOF
7 83 OF LEFT ENDOF ENDCASE ; : KEYIN2 CASE 87 OF LUP ENDOF 82 OF R
8 UP ENDOF 90 OF LDOWN ENDOF 67 OF RDOWN ENDOF ENDCASE ; : KEYIN1
9 CASE 79 OF STATEON ENDOF 70 OF STATEOFF ENDOF 46 OF BCHANGE ENDO
10 F 81 OF SDCHANGE ENDOF ENDCASE ;
11 : ENDALL TEXT CLS 1 1 AT ." ENTER 'FORGET AT' TO SAVE MEMORY" ;
12 : RUN INIT BEGIN ?KEY DUP DUP DUP 0 > IF KEYIN1 KEYIN2 KEYIN
13 3 ENDF SP! ?TERMINAL UNTIL ENDALL ;
14
15
    
```

SCR #97

```

0 ( CLOCK - TO START ENTER HOUR 1 - 24 AND MINUTE AND TIME )
1 0 VARIABLE XX 8 ALLOT 58 XX 2 + C! 58 XX 5 + C! 0 VARIABLE TT
2 : !UPDATE 1 TT +! TT @ 59 > IF 0 TT !
3 XX 7 + 1 OVER C@ + DUP 58 < IF SWAP C! ELSE DROP 48 SWAP C!
4 XX 6 + 1 OVER C@ + DUP 54 < IF SWAP C! ELSE DROP 48 SWAP C!
5 XX 4 + 1 OVER C@ + DUP 58 < IF SWAP C! ELSE DROP 48 SWAP C!
6 47 TT !
7 XX 3 + 1 OVER C@ + DUP 54 < IF SWAP C! ELSE DROP 48 SWAP C!
8 XX 1 + 1 OVER C@ + DUP 58 < IF SWAP C! ELSE DROP 48 SWAP C!
9 XX DUP C@ 1 + SWAP C! ENDF XX C@ 50 = XX 1 + C@ 52 = + 2 = IF
10 48 48 XX C! XX 1+ C! ENDF ENDF ENDF ENDF ENDF
11 XX 22 8 VMBW ENDF ;
12 : TIME 10 /MOD 48 + XX 3 + C! 48 XX 6 + C! 48 XX 7 + C!
13 48 + XX 4 + C! 10 /MOD 48 + XX C! 48 + XX 1+ C!
14 INTLNK @ ' !UPDATE CFA ISR ! -31804 ! ;
15 : STOPCLOCK 0 -31804 ! ;
    
```

PRESS ON CABLE
CONNECTORS
PIN ARRANGEMENT



HUG PICNIC

Come join us for the annual picnic of the Houston Users Group in Hermann Park. Bring your family. Enjoy an afternoon with your fellow members, or enjoy the planetarium, museums, zoo, and pond that are located within the park. HUG has obtained a permit to use the Hermann Park pavilion. The pavilion is located over on the west side of the park close to Fannin Dr. Look for the HUG sign on the pavilion. You can get to Hermann Park by from Loop 610 south by taking the Fannin Rd. Exit and going north past the Astrodome and the Texas Medical Center. You can get there from the west by taking Holcombe Blvd. (changes to Bellaire Blvd at Stella Link) to Fannin. From the northwest, go into town on 59 westbound. Take the Greenbriar exit. Go south on Greenbriar and then left on Sunset Blvd to Fannin. Right on Fannin to the pavilion. Please bring your own food and drinks. Please, no alcoholic beverages.

D.C. ELECTRONICS
 5206 KINGSMILL RD.
 FRIENDSWOOD, TEXAS 77546
 (713) 482-0186

ADVANCED DIAGNOSTICS.....\$17.95 (hardware tool)	STAR SG-10.....\$257.95 (factory auth. supp)
4A/TALKS (V1.2).....\$17.95 (with autodial)	LEGEND 808..... \$169.95 (100cps dot matrix)
THE EXPLORER.....\$21.95 (advanced programing aid)	VOLKSMODEM 12.....\$229.95 (with cable)
BIT MAC.....\$31.95 (superior graphics)	SIGNALMAN MKIII.....\$74.95 (300 baud manual)
NAVARONE DBM.....\$54.95 (with support)	SIGNALMAN EXPRESS..\$309.95
MINI-WRITER 1.....\$17.95 (cassette word processor)	PRINTER STANDS.....\$15.95

TI-MMS MINUTES

JULY 14, 1985

Pres. Bill Knecht called the regular monthly meeting of the Houston Users' Group to order at 2:02 p.m. at the St. John's School, 2401 Claremont, Houston, Texas. Vistors and new members were asked to introduce themselves after which the President introduced the officers.

Treas. David Mather presented the Treasurer's Report.

The chair entertained a motion to table the approval of the minutes. Motion entered and seconded. MOTION CARRIED.

Newsletter Library Committee Chairaan Mary Boyd called for volunteers to help with the cataloguing and distribution of information from newsletters.

Motion was made by Don Lewis: That the Board of Directors be elected annually by the membership and be entrusted with the decision making for the HUG with the membership reserving the right to rescind any decision. MOTION TABLED.

The program portion of the meeting commenced at 2:35 with Don Lewis conducting "Array Handling, a HUG Tutorial" after which hand-outs were provided. Then Bill Knecht illustrated his means of clearing the screen, a program he said he would submit to the newsletter editor. Tony Johnson deonstrated a second prototype stand-alone keyboard he has made for the 99/4A.

Bill Knecht announced that the next issue of the newsletter would give the location of the August picnic-meeting to be held on August 11. The meeting closed at 3:34 p.m..

Respectfully submitted,
 Lucia C. Greer, Secretary

The biggest selection of

COMPUTER FURNITURE

In Texas

From \$49

EFFICIENT

ERGONOMIC

PRACTICAL

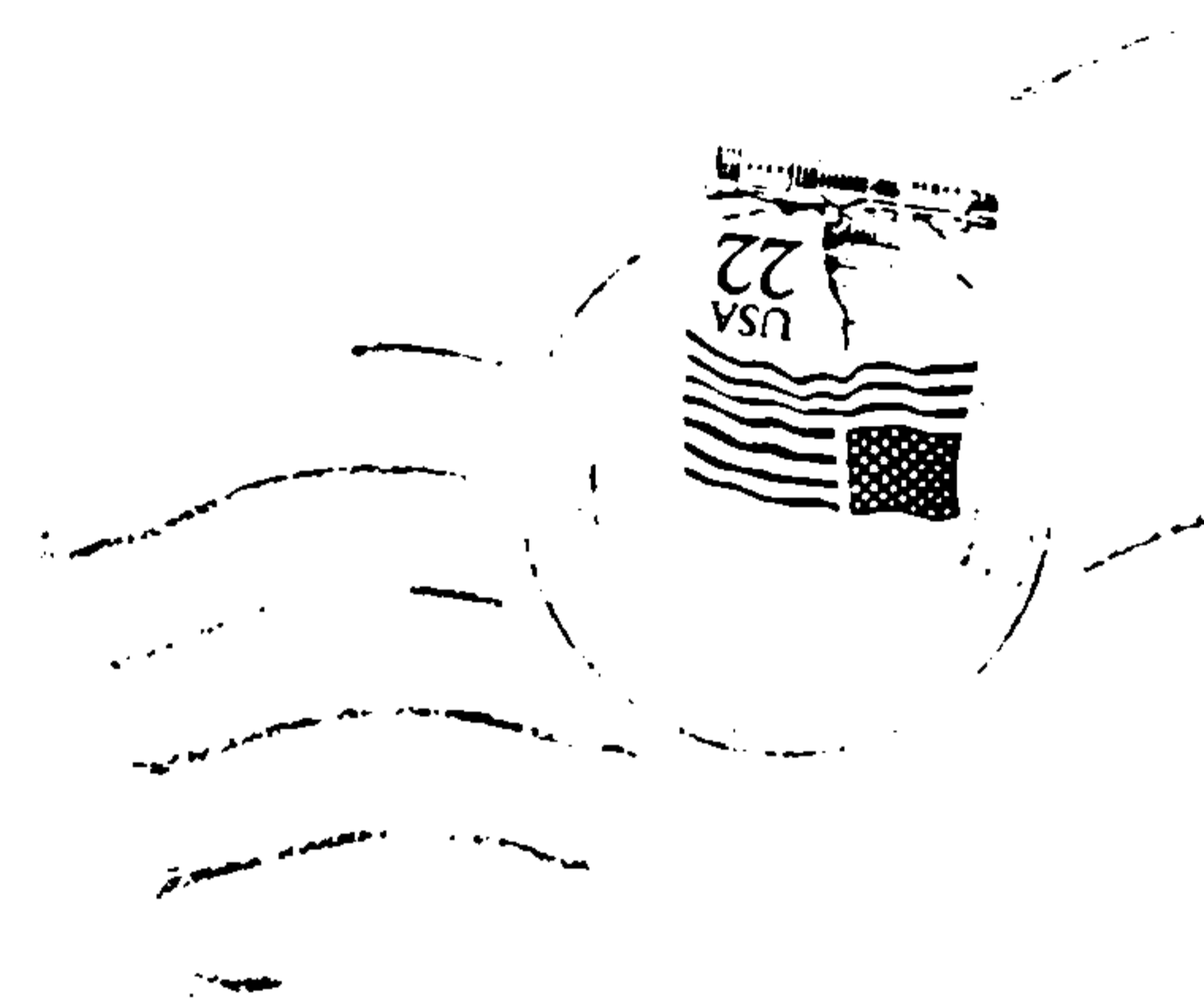
AT:
PORTABLE STANDS & FURNITURE, INC.

10968 WESTHEIMER AT WILCREST (NEXT TO RANDALLS)

MasterCard VISA Discover

Open Monday — Saturday 10 am-9 pm
781-8060 Free Gift with Purchase

ROGERS MILLS
P.O. BOX 5310
PASADENA, TX. 77508-5310



Sanford & Ruth Herman
6219 Sanford Road
Houston Tx 77096