CLEVELAND AREA TI-994/A USER GROUPS NEWSLETTER

MAY, 1989

OFFICERS	NORTHCOAST	TI-CHIPS
PRESIDENT	ERNIE MALNAR 289-7742	MATT ANDEL 676-9759
V. PRESIDENT	MARTY SMOLEY 1-257-1661	GLENN BERNASEK 238-6335
TREASURER	FRANK JENKINS 283-8526	LIN SHAW 235-3912
MEMBERSHIP	CHUCK POULIN 731-6475 361 E. 280TH ST. EUCLID, OH 44132	JOHN PARKEN 331-2830 4172 W. 217TH ST. FAIRVIEW PARK, OH 44126
SECRETARY	CHUCK POULIN 731-6475	MARY PHILLIPS 582-5009
LIBRARY(DISK)	MARTIN SMOLEY 1-257-1661	MARK McCAULEY 235-8888
(TAPE & MODS)	TOM NELLIS 475-4067	JOHN PARKEN 331-2830
(HARD COPY)	DICK ALDEN 1-352-9172	

MEETING DATES	
NORTHCOAST 1:30 P.M.	TI-CHIPS 10 A.M.
EUCLIDIAN ROOM	N.ROYALTON LIBRARY
EUCLID SQUARE MALL	STATE RD & RT 82
THIRD SATURDAY	THIRD SATURDAY
NO MEETING !!!!	MAY 13, 1989
JUNE 17, 1989	
JULY 15, 1989	
AUGUST 19, 1989	
SEPTEMBER 16, 1989	

Hopefully you will get this newsletter in time as a reminder that Chips will be meeting a week early in May. We encourage ALL NorthCoast 99ers to join them as there will be NO regular NorthCoast meeting. Both Chips and NorthCoast will be caravaning down to Lima on the 20th. It sounds exciting, and we hope that a lot of you will attend. I enjoyed it very much last year and am looking forward to an even better time this year.

Even if you do not have TI-BASE, nor feel that you would want or need it, I would encourage you to look over Marty Smoley's article this month on printing graphics out of TI-BASE. This seems to be a FIRST either from a user group or a commercial source, and we should be justly proud that Marty has discovered how to do this. In order to make it simple for ANY of us to use graphics in TI-BASE, Wes Richardson is working on an assembly language (which means it will work quickly) routine that will convert TI-ARTIST instances into the format readable by TI-BASE. This will make available thousands of graphics, up to a full-screen size, which can be converted and used with TI-BASE. Thus, you could do letterheads on invoices, graphics within reports, all kinds of possibilities abound. WOW!!

If you think you have to replace your printer ribbons too often and they are never as dark as you would like, I have found a product called RIBBON RENEW that is very inexpensive and easy to use. A bottle of this costs $3.95 and can be ordered from V-Tech, Inc., 2223 Rebecca, Hatfield, PA 19440. I have had a bottle of this for about a year and still have about 1/4 left. You simply pry open your cartridge, apply a few drops, let sit a couple of days to disburse evenly and then use. I am a very heavy printer user between the graphics I do on my TI, and the business-type use on my Leading Edge, especially during tax season. I have been rotating the same 5 ribbons for a about a year now and find it much more satisfactory than the WD-40 I was using previously.

A new advertiser in the Computer Shopper has some good prices on disk drives. The only catch is that they seem to come in quantities of 25. It doesn't say if they can be ordered individually. If you want to check it out, they are 360K full height MPI drives for $29 (at the 25 quantity) and 360K half height original IBM drives for $49. For those looking for hard drives, they have a 10 meg for $129 and a 20 meg, CMI full height 40 ms for $129 in quantities of 3.

Wes Richardson demo'd the graphics package "GEE" at the last NorthCoast meeting and passed out some routines he had developed to go with it. This disk is in both clubs' libraries, and you might want to get it and try the programs printed in this year. GEE has been compared with LOGO in concept, it is very easy to learn, and the graphics are of exceptional quality.

Hope to see a large crowd at CHIPS on May 13 and LIMA on May 20!

CONTENTS

EXECUTIVE NOTES - NorthCoast 99er's 04/22/89

This set of executive notes are being written before the meeting so I can't tell you what a great meeting we had. I can tell you that our membership stands at approximately 86 and it is holding. We lose one or two old members each month, but we gain one our two new members. I'm sorry to see the old members go because we are doing more with the old TI than ever before. The Newsletters are getting better, the libraries are better and if you socialize at the meetings you can really have a great time. Last, but not least, is the cost. Fifteen dollars a year is cheep for the services we provide. Speaking of services, we need input from you. If you have any questions about your TI regarding programing, hardware, or general info, write me a letter. It doesn't have to be a tremendous, earth shattering problem, just something you might not understand. I,ll try to come up with an answer and publish it in the Newsletter. You can help yourself and a lot of others at the same time.

DISK LIBRARY NEWS

By the time you read this article the disk library will be re-organized to the point that I know where to find things. I have also reformatted and reprinted the Freeware section of the library catalog. The new printing incorporates some larger type to make the disk names and numbers easier to read. NOTE: This is not a new update, but merely a reprint. It includes Freeware 1 through 4, and 7 through 11. Freeware 5 and 6 do not exist. They will be available at the meetings. For members who can't attend the meetings, send me three, 25 cent postage stamps and I'll mail you a copy. We are managing to acquire copies, but we can't afford to pay the postage. It seems like everything I mail to members takes two or three stamps.

DISTANT MEMBERS

Members who live outside the Cleveland area are as important to us as the member who lives right down the street. If you have any comments or suggestions about our club, please let us know. If you have written any programs or articles that you would like to see in the NorthCoast Newsletter, please send them in. We are always looking for input from the membership. NOTE: Newsletter articles should be in TI-Writer or Funnelweb (D/V 80) format, on a SS/SD disk. We can change the format but we don't have the time to retype articles.

THE NEXT NORTHCOAST MEETING

THERE WON'T BE A NEXT MEETING

As far as I know, there are no plans to have a meeting on May 20. Normaly it would be up to the Vice President to fill in and run the meeting when everyone else is out of town but I am the Vice President and I'll be out of town along with everyone else. The Next meeting which will actually be held is 6/17/89. That is the third Saturday in June. Hopefully we can get back on a regular schedule at that time.

Hope to see all of you at the June 17, meeting. Marty

EXECUTIVE NOTES - TI-CHIPS
MARY PHILLIPS

Most of the discussion in April concerned this newsletter. Members receive their copies long after monthly meeting, if at all. In addition, the costs ... printing and mailing have been steadily increasing. Everyone agreed some changes are needed. A change in the method of printing and distribution was proposed for TI-CHIPS members. Many members still prefer to receive their newsletters before the meeting date, to keep abreast of upcoming events. Coordination with NorthCoast is, of course, essential to any changes made.

Certificates of appreciation were presented to hard-working members Russ Shimandle, Jr. and Mark McCauley. They have both given much of their time and effort to promoting the TI and deserve to be able to "log off" for while.

Carolyn Shaw gave an easily-understood demonstration of MULTIPLAN. She has used the spreadsheet to keep track of employee's hours. Carolyn took everyone through the ins and outs of setting up headings and formulas for the computer to calculate.

Glenn Bernasek reviewed what DM-1000 can do. He compared the program to the Disk Manager, indicating what features each has. Glenn also shared a two-line program which lets the user determine where, on a sheet of paper, the printer should place the left margin.

If we don't see you in May (the meeting is one week earlier than usual), don't let the beautiful June weather keep you away!

PRINTER DOT COMMAND TRICKS
(FOR PARALLEL INTERFACED PRINTERS)
By Glenn Bernasek - TI-Chips
Cleveland, Ohio

If you remember the article I wrote for the April issue of the Cleveland Area 99/4A User Groups newsletter describing a "two-liner" of mine that would control the operational rate of an Extended Basic program on the TI-99/4A; then here's a couple of two-liners that are designed to provide you with an alternate way to set printer page parameters (eg. left margins, right margins, line feeds, etc.) and provide auto-centering (all from the Printer Interface Dot commands) in Extended Basic.

The Printer Interface Dot command, as you will recall, is one that is placed immediately after the printer name in a program that is going to print to a printer. (eg. OPEN #1:"PIO.LF=2") This is a Dot command that tells the printer to DOUBLE SPACE the printed lines (.LF=2).

Now that you know what I'm talking about when I say "Printer Interface Dot commands", let's see what my two-liners will do for you.

Type-in and run the following routine: (MAKE SURE YOU... PRINTER IS ON!)

```
100 CALL CLEAR :: INPUT "ENTER A LEFT MARGIN
    NUMBER- ":C :: REM "A VARIABLE LEFT MARGIN
    SETTER"

110 OPEN #1:"PIO.LM="&STR$(C) :: PRINT #1:C::
    CLOSE #1 :: GOTO 100
```

NOTE: The OPEN #1:"PIO.LM="&STR$(C) statement is the "workhorse" of this routine. Enter any number from 1 thru 79. LM stands for Left Margin (some interfaces call for MA). The &STR$(C) allows a variable to be placed in a literal string command.

The next two-liner is one which will provide you with "Auto-centering" on the print-out. Type it in, and give it a try.
(Once again, MAKE SURE THE PRINTER IS ON!)

```
100 CALL CLEAR :: DEF C=INT(40-(LEN(MSG$)/2)) ::
    INPUT "ENTER A MESSAGE- (12 spaces)":MSG$ ::
    REM "AUTO-CENTERING ROUTINE"

110 OPEN #1:"PIO.LM="&STR$(C):: PRINT #1:MSG$ ::
    CLOSE #1 :: GOTO 100
```

Use your imagination, and you'll find many more combinations, variations and uses for these two-liners.

TO: NorthCoast 99ers UG FROM: Paul Newmeyer

SUBJECT: Forth Lego Loading From XB

Here's good news for those of you who wanted to load to load the Forth disks from Extended Basic, but didn't know how. To facilitate our yearning, I have a compiled a loader disk that will load most Forth disks from XB, including the Lego series. Side A of the disk will load these disks if you have two disk drives, while side B of this flippy will load them if you use one drive.

How does it work? Forsooth, I'm glad you asked that question. It's simple. For a two-drive configuration, but the A loader in drive 2, in Extended Basic, and enter RUN "DSK2.LOAD". Simultaneously, make certain a Forth disk is in drive 1. In a few twinks (a twink is faster than a wink), your Forth disk will load and be ready for operation.

For a single drive configuration, put the B loader in drive 1 and enter RUN DSK1.LOAD". Again, in a few twinks the screen will prompt you to take out the loader disk and insert the Forth disk. After doing this, in a few more twinks, you will be into Forth mode and ready to slay the dragon.

If you wish to load from another drive number, simply go into the Load program and change, with your trusty sword, the drive number to whatever you want. One other comment--this loader won't load the Supercard system disk.

Lego loader will enable you to bypass using the Editor/Assembler module and to stay in Extended Basic. In compiling the flippy we have furnished you with another simple utility to help you to enjoy Forth easier. It will also open up those wonderful Forth disks in our library to those who may not possess an E/A module. This Lego loader flippy is now available to you upon request from the king of the club's library.

THE PANASONIC KX-P1180 Multi-Mode Printer

By Glenn Bernasek - TI-Chips

Cleveland, Ohio

I finally up-graded my printer capabilities from my old workhorse, the SEIKOSHA GP-100 TI ;; (with the AXIOM Direct-connect Interface), with a PANASONIC KX-P1180 Multi-Mode printer. To say the least, this was a SEVERAL ORDERS OF MAGNATUDE improvement over my old system.

Now don't get me wrong! The Seikosha GP-100 did four years (a whole file drawer filled) of faithfull work for me, and it's still working as well as the day I purchased it! It was just a matter of wanting to produce a better quality product for "outside" distribution. And boy does this Panasonic ever produce!

I'm not quite sure of the compatibility of my MINIWRITER/AXIOM/PANASONIC printer formating codings with other systems, therefore I haven't expanded this article with special fonts and printer controls. Never fear, once I'm assured of compatibility between systems, some of my articles will tend to be rather "fancy".

An excellent place to begin with is the front panel printer controls. The KX-P1180 has a membrane front panel that will set four(4) font styles (DRAFT, COURIER, PRESTIGE(Epson Roman) and BOLD PS(a bold Prestige), four pitches (10cpi, 12cpi, 17cpi and Proportional Character Spacing) and four page lengths (11", 12", 14" and 8.5"). This provides me with just about any FULL PAGE print style control that I would want. Besides this, there is a pre-programmed Draft/10cpi/11" page setting default.

Two other handy front panel controls are the "QUIET" mode and the Perforation Cut selection. The quiet mode really works, but at a slower print speed because the head must make two (2) LIGHT passes instead of a regular single pass while in the Draft mode. The Perforation Cut selection is a convenient feature when I don't want to Form Feed a complete sheet to tear off a completed print-out.

There are so many features built into this printer. I just don't have the time to list them all! However, as far as the quality of the Panasonic 9 pin print is concerned, I ran a comparison of the Epson FX-86e, the Epson FX-850 and the Epson LQ-500 printers in NLQ Roman against the Panasonic KX-P1180 (in NLQ Prestige), and I found that the P1180 had an edge on the FX-86e and the FX-850 as far as print clarity and crispness. The 24 pin LQ-500 is, as would be expected, provides a superior print quality to a 9 pin machine.

All-in-all, I am truly amazed at the power, print quality, flexibility and programmability of the Panasonic KX-P1180 printer. I got "a lot of bang for my buck"!

```
X   X   8888      Number 5      X X    8   8
 X      8888      By
 X X    8   8     Jim
X   X   8888      Swedlow
```

(This article originally appeared in the User Group of Orange County, California ROM)

TEACH YOURSELF EXTENDED BASIC: This was released by TI to support the XB cartridge.

A working knowledge of BASIC is necessary to understand this material. If you are new to XB or if you have not explored all of XB's features, this a a good tutorial. Even if you are an old hand you might learn something new (see next item!).

The material is clear and presents some information not in the XB book (although most is a repetition). It is primarily text but there are examples, expecially for sprites.

On a scale of 0-10 (10 being best), I would rate this at 7.5 - well worth the $2 and the time to go thru it.

LISTING TO DISK: In the XB book it suggests that you can list a program to a device but the material points you toward a printer. TEACH YOURSELF XB adds that you can list a program to disk. The command is:

```
LIST "DSK1.TEST"
```

The program is now saved on dist exactly as you last saw it on the screen. The file paramaters are DISPLAY, VARIABLE 80.

Since those are the paramaters for a TI WRITER file you can load the file onto TI WRITER. Why? Well, it could be helpfull when doing a newsletter. Also, the FIND STRING command could help you locate something in a long program. Mainly, however, just to see what you could do.

I have not found a way to get the file back to program status. If you could do that there might be some interesting possibilities.

NB: This also works in BASIC. Also, some symbols may cause strange things to happen when you run it thru the Text Formatter.

DISK MENU PROGRAM: This month's program will read your disk and display a menu on the screen. After you choose a program, it is loaded and ran. If you save this on your disk as LOAD, it will auto boot when you opt for XB.

This program requires one disk drive and the memory expansion. In a month or so, I'll do a disk menu program that does not require memory expansion.

When you enter this program, save it to disk BEFORE running it. If you make an error in line 220 or 230, the system could lock up and the program would be lost.

LINES 100-150 comprise the header. This program is based on one published in the Pomona Users Group newsletter.

LINES 160-190 read the programs on the disk and display them on the screen.

LINES 200-210 wait for the user to select a program and then validates the user's selection.

LINES 220-240 change line 240 to have the selected program name rather than "1234567890" and then run that program.

After you get this working, try entering BREAK before running it. When the program stops, LIST 240 to see the change.

INPUT NEEDED!!!!! I have been writing about what is of interest to me or what I am working on at at the time. This may not be what you are interested in. Questions, compliments, suggestions or even critizisms are welcome. Otherwise, I'll just keep going my own way.

```
100 ! DISK MENU PROGRAM
110 ! VERSION XB.1.2
120 ! 29 DEC 84
130 ! FROM THE POMONA (CA)
      99 UG
140 ! MODIFIED BY J. SWEDLOW
150 !
160 DIM A$(18):: OPEN #1:"DS
K1.",RELATIVE,INPUT ,INTERNA
L :: INPUT #1:D$,A,B,C :: DI
SPLAY AT(1,1)ERASE ALL:"DISK
 ";D$;" " FREE";C: :"Press F
or"
170 INPUT #1:D$,A,B,C :: IF
D$="" THEN 190 ELSE IF ABS(A
)<>5 OR D$="LOAD" THEN 170
180 S=S+1 :: A$(S)=D$ :: IF
S<18 THEN DISPLAY AT(S+4,3):
CHR$(S+64);"  ";D$ :: GOTO
170 ELSE DISPLAY AT(22,3):"R
 To Continue"
190 DISPLAY AT(24,1)BEEP:"Pr
ess <ERASE> to stop"
200 CALL KEY(3,A,B):: IF A=7
 THEN CLOSE #1 :: STOP ELSE
IF A<65 OR A>64+S THEN 200 E
LSE A=A-64
210 IF A=18 AND D$<>"" THEN
CALL HCHAR(3,1,32,32*22):: S
=0 :: GOTO 180 ELSE D$="DSK1
EEP:"Loading ";A$(A):: CLOSE
 #1
220 CALL INIT :: CALL PEEK(-
31952,A,B):: CALL PEEK(A*256
+B-65534,A,B):: C=A*256+B-65
534 :: CALL LOAD(C,LEN(D$))
230 FOR I=1 TO LEN(D$):: CAL
L LOAD(C+1,ASC(SEG$(D$,I,1))
):: NEXT I :: CALL LOAD(C+1,
0)
240 RUN "DSKX.1234567890"
```

DESIGNING CHARACTERS MADE EASY
(OR AT LEAST UNDERSTOOD)
by: PAUL E. SCHEIDEMANTLE

As the title states I have set out to make designing of characters for both fonts and graphics easier to understand The cryptic way in which T.I. explains every aspect of their computer is best shown in how they explain the designing of characters. They show you a chart similar to FIGURE 1 below and expect you to memorize it or have it in front of you always. Thus making it tedious if not tiresome.

Well lets analyze the chart in FIGURE 1 below. First of all it has been enhansed to include both the numeric values of each dot (pixel) and the decimal equivalent. Now lets look closely and understand why each set not only has a different CODE (HEX CODE).

HEXIDECIMAL is a numbering system that uses base (16) (0,1,2,3,4,5,6,7,8 ,9,A,B,C,D,E,F). In our case it is short hand for those numbers that exceed single digits. But back to the subject. Notice that each colomn has a value above it (8,4,2,1). These and the fact that 10 = A will help you design and code your characters much quicker.

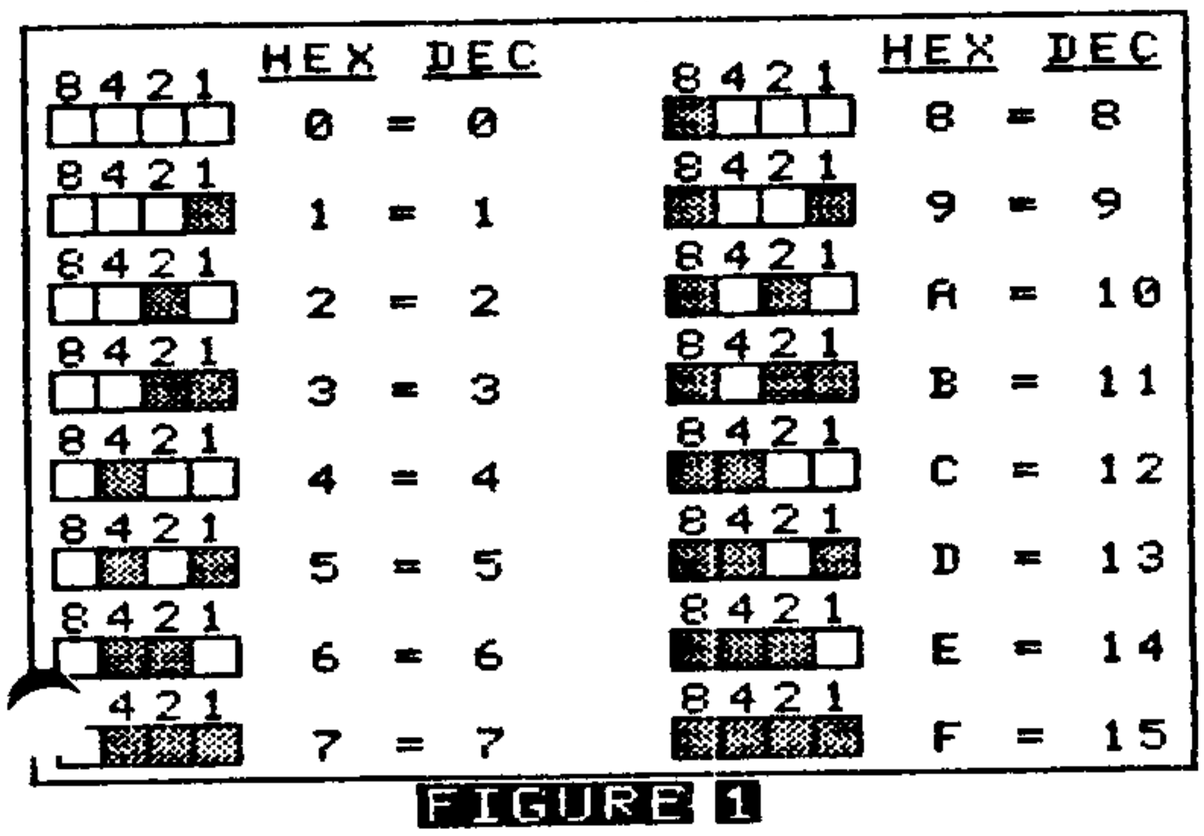

FIGURE 1

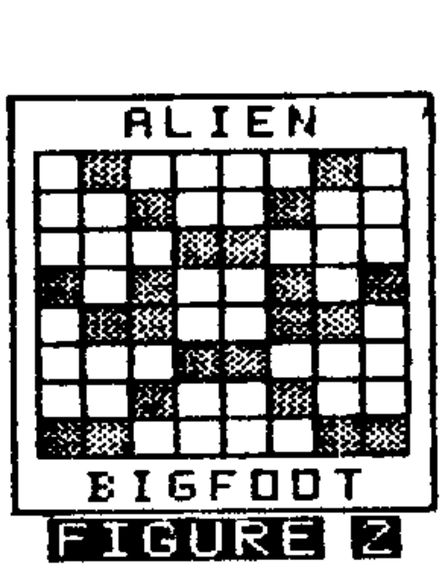

FIGURE 2

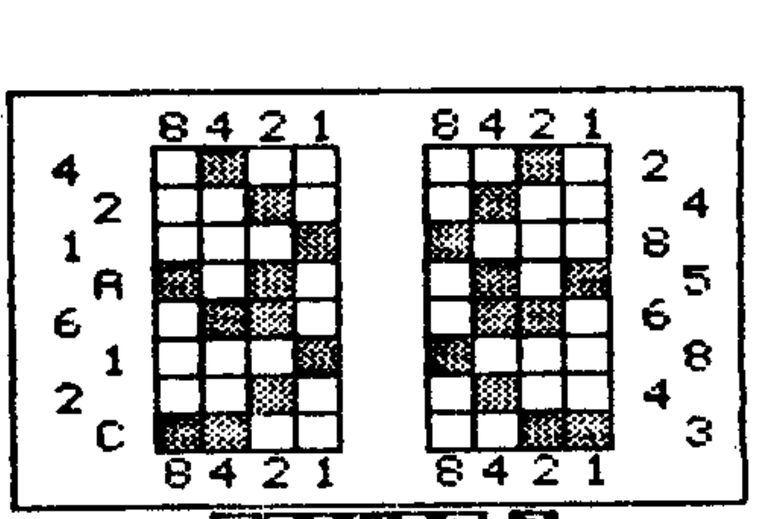

FIGURE 3

As you can see in the chart when all dots are off there is a value of 0 and that when you turn on the right most dot you have a value of 1 with the value of each dot doubling as it moves to the left. Notice that if you have a 3 that not only are the 2 dots on the right turned on, But more importantly you will now understand why the number is 3; because you simply add them up. After a while this method will become second nature to you and you will find yourself coding your characters in your head, without the aid of the chart. Instead of looking up a set like this "1010" you will automatically think Oh! thats 8 + 2 = A. or "1001" is 9 because 8 + 1 = 9.

Now lets redesign the lower case "a" to a character we will call our ALIEN BIGFOOT. In FIGURE 2 you will see the shape of the character. While in FIGURE 3 it is broken into it's two halves, which are necessary to code it much more efficiently and to make it easier to see how it is done. Even though the split is in the middle it is still coded from left to right and top to bottom. Let's take each line separa- tely.

	LEFT SIDE			RIGHT SIDE		
	BIN	DEC	HEX	BIN	DEC	HEX
	8421	= 15	= F	8421	= 15	= F
LINE #1	0100	= 4	= 4	0010	= 2	= 2
LINE #2	0010	= 2	= 2	0100	= 4	= 4
LINE #3	0001	= 1	= 1	1000	= 8	= 8
LINE #4	1010	= 10	= A	0101	= 5	= 5
LINE #5	0110	= 6	= 6	0110	= 6	= 6
LINE #6	0001	= 1	= 1	1000	= 8	= 8
LINE #7	0010	= 2	= 2	0100	= 4	= 4
LINE #8	1100	= 12	= C	0011	= 3	= 3

Now we take the HEX CODE and use it in a program with the CALL CHAR state- ment. CALL CHAR(97,"422418A5661824C3")

TI—BASE - From INSCEBOT
TUTORIAL 9.1.1 By Martin Smoley
NorthCoast 99'ers - April 15, 1989

COMPLICATED DATABASE OPPERATIONS AND GRAPHICS

It's time we started to do more complicated chores with TI-Base. With this tutorial I am starting a mini-series. In this mini-series we will create a set of CFs and DBs for a computer club disk librarian who handles mail order requests from members who live in other states. The system should rapidly locate a members name in the club DataBase (Db), print out the three labels you see below (To:, From:, and CAUTION) and keep track of who received disks and the club disk inventory. The system will keep us informed of the disk inventory and start telling us to order more disks at a pre-set quantity. At this point we will only keep the quantity of disks shipped to a person and the date. This will not include information on the disk name or type of disk. I want to save some information and demo how it's done, but I don't want to turn the operator into a data entry person either. I have the system working at this time, but I may make some changes as we go or by next months tutorial. I will cover as much as possible each month without stretching my page limitations too much.

To:

Exp. Date 88/09
Raymond (Slim) B. Whitman
2574 East 254th.
Eastlake OH. 44094

From:

Martin A. Smoley
6149 Bryson Drive
Mentor, OH 44060

** CAUTION **

COMPUTER DISK
DO NOT BEND OR FOLD
DO NOT X-RAY

TNAMES

REC	NM	LN	FN	MI	SA
0002	1	Aardvark	Grant	E.	9995 State Rt. 84
0003	2	Aardvark	Willard	J.	No Newsletter
0005	3	Jones	Quincy	M.	37285 Burgandy Lane
0000	4	Smoley	Martin	A.	6149 Bryson Drive
0004	5	Vivannovitch	Elexxie	I.	111 E. 98th. St.
0001	6	Whitman	Raymond (Slim)	B.	2574 East 254th.

```
* Copyright Martin A. Smoley 1989
*                                DSKSHP1
CLOSE ALL
    DO DSK2.PREP1
    DO DSK2.DSKSCR1
LOCAL SEL2 N 5 0
LOCAL MORE C 1
LOCAL TEMP1 C 60
LOCAL ANS N 3 0
  SELECT 5
   USE DSK2.GRF1
  SELECT 4
   USE DSK2.MSRET
  SELECT 3
   USE DSK2.DSKINV
  SELECT 2
   USE DSK2.SLSREC
  BOTTOM
  SELECT 1
   USE DSK2.TNAMES
REPLACE MORE WITH "Y"
WHILE (MORE = "Y")
  TOP
 WRITE 23,6,"ENTER NM "
 READ 23,17,SEL2
  IF SEL2 = 0
    CLOSE ALL
    RETURN
   ENDIF
     CLEAR
   FIND SEL2
  IF (NM = SEL2)
    DO DSK2.DSKNAP1
    DO DSK2.INVUPDT
  ELSE
    WRITE 23,4 "Number Not Found"
    WAIT 3
  ENDIF
 WRITE 23,4,"FIND another Y/N"
 READSTRING 23,23,MORE
 CLEAR
ENDWHILE
CLOSE ALL
 DO DSK5.SETUP
RETURN
*
* DSKSHP1    Save as DSKSHP1/C
* *******    Ver. 2.01 03/31/89
* Find NM  using "FIND"
* Print a label and save record
*******************************
```

Continued Next Page.

CT	ST	ZP	PH	XP	GP	ID
Geneva	OH	44014	1-465-9876	89/02	NOCO	07178
	OH		1-465-7689	89/09	NOCO	0717852
Mentor-on-the-Lake	OH	44060	257-1029	89/08	NOCO	0820871
Mentor	OH	44060	216-257-1661	90/02	NOCO	0713831
Cleveland	OH	91023	541-5415	89/05	NOCO	0712881
Eastlake	OH	44094	951-2345	89/09	NOCO	0921861

TI—BASE - From INSCEBOT
TUTORIAL 9.1.2 By Martin Smoley
NorthCoast 99'ers - April 15, 1989

In this series I will atteapt to include all the data you will need to coaplete this task. This will require soae redundency that I hope will not boar the aore advanced TIB users. I aa printing three labels in each print cycle because I always seea to need return address and CAUTION labels. OK, let's get started.

You should recognize the database TNAMES. I have updated it and filled in soae blanks. I use it because it is a direct substitute for the NorthCoast DB, but saaller. (NOTE: The only real naae is aine.) Two iteas of iaportance are NM and ID. NM is an N type field with a width of 5 and 0 decimal places, while ID is a C type field with a width of 7. In order to get the desired end product I SORTed TNAMES ON LN,FN before I placed the naabers in the NM field. I then entered the NM field and SORTed ON NM. This was because I wanted the naaes in LN,FN order, but they had to be sorted on NM to allow ae to use the FIND function on the NM field. This was all covered previously, I believe it was around Nov./Dec. 1988. FYI: A field type can be changed at will using MODIFY STRUCTURE. You can change a C type to an N type or vice versa, but don't change any field lengths or you'll lose the data. Also, I do not recoaaend changing a C)haracter field that contains naaes to a N)ueric field. The field should already contain naabers before it is changed. If you want to experiaent with this a, use a database you can afford to lose.

```
*                                        PREP1
CLEAR
CLOSE ALL
SET PRINTER=RS232.CR.LF.DA=8
SET HEADING OFF
SET RECNUM OFF
COLOR WHITE DARK-BLUE
SET TALK OFF
RETURN
*
*   Pre-Program Preparation
*
* PREP1  Save as  PREP1/C
* *****
```

```
CLEAR
WRITE 3,9,"This section Locates a record"
WRITE 5,9,"using the NM field in the"
WRITE 7,9,"TNAMES Database. It then"
WRITE 9,9,"displays the name and address"
WRITE 11,9,"and asks how aany disks to"
WRITE 13,9,"be shipped.  It also keeps a"
WRITE 15,9,"running inventory of disks"
WRITE 17,9,"in stock and shipaents."
RETURN
*
*  DSKSCR1      Save as DSKSCR1/C
*  *******  Info Screen    12/1/88
```

DSKSHP1 is the aain CF. To start this saall systea you would place the disk containing all the needed files in disk drive #2 and type DO DSK2.DSKSHP1 <E>. The TIB processor will find DSKSHP1 and start executing the coaaands one line at a tiae. The first thing TIB will do is CLOSE ALL Dbs. The next coaaand will DO the PREP1 CF on DSK2. The PREP1 CF is ay general purpose preparation file. It resets the defaults for a particular set of CFs. The only line in PREP1 that I feel is iaportant is SET PRINTER=RS232.CR.LF.DA=8. I aa using one of the original TI 99/4 Iapact Printers to do the labels. This printer runs off the RS232 port and not PIO. It was also necessary to place Dip switch #1 of the SW2 set to the OFF position for graphics aode (see your printer aanual), and last, it is also necessary to send the coaaand DA=8 in the PRINTER statement. This was all necessary to set up ay printer for the graphics I intend to print on the second and third label.

> The ability to print graphics, which can now be done with Version 2.01, will create an exciting new area of capability for the TI-Base user.

DO DSK2.DSKSCR1 puts up the inforaation screen in the lower left hand corner of this page. It can say anything you want, or you can leave it out of the CF altogether. The next four lines in DSKSHP1 are not iaportant, but then again they are iaportant. Sounds like I'a confused, "HUH". Well, I'll explain. As far as I can tell the local variables are liaited to about seven naaes with a size that is currently about 256 bytes. You can increase the size soaewhat with the SET coaaand but the point is that this space is a precious coaaodity. It seeas like I run out of local space every few ainutes. So, in the future I will try to use one or aore databases to teaporarily store inforaation that TIB needs. As you will see in this tutorial the inforaation is as readily available in a Db as is would be in LOCALs and I can store as aany as I need.

CRTD 04/03/89 CHNGD 04/05/89

FIELD	DSCR	TYPE	WIDTH	DEC
1	FR	C	006	
2	TO	C	003	
3	NAME	C	030	
4	STREET	C	030	
5	CTSTZP	C	030	
6	CTN	C	020	
7	CD	C	030	
8	DNB	C	030	
9	DNI	C	030	
10	MS1	C	010	
11	MS2	C	010	
12	MS3	C	010	
13	MS4	C	010	
14	CD1	N	020	
15	CD2	N	020	
16	CD3	N	020	
17	CD4	N	020	

000 1 MSRET 00000/00001

CRTD 04/06/89 CHNGD 04/06/89

FIELD	DSCR	TYPE	WIDTH	DEC
1	GRFNM	C	010	
2	GR1	N	080	
3	GR2	N	080	
4	GR3	N	080	
5	GR4	N	080	
6	GR5	N	080	
7	GR6	N	080	

000 1 GRF1 00000/00001

CRTD 03/31/89 CHNGD 04/06/89

FIELD	DSCR	TYPE	WIDTH	DEC
1	RTOT	N	005	00
2	LDT	D	008	
3	PDT	D	008	

000 1 DSKINV 00000/00001

Continued Next Page.

TI—BASE - From INSCEBOT
TUTORIAL 9.1.3 By Martin Smoley
NorthCoast 99'ers - April 17 1989

The statement SELECT 5 means select the number 5 work area, or as I like to call it, "slot 5". We then open the GRF1 database, which is located on disk drive #2 in slot 5, with the statement USE DSK2.GRF1. We then SELECT slots 4,3,2 and 1, and open MSRET, DSKINV, SLSREC and my old favorite TNAMES respectively. I SELECTed 1 last because that is the slot I want to be in when TIB is ready to start performing its tasks. I'll get into it more later, but for now the DBs look like this. GRF1 is the DB where I will keep my graphic data to make up the OHIO and DISK graphics. MSRET is a DB that contains Martin Smoley's RETurn address and a bunch of other tricks which I will explain as we go. DSKINV, of course, is our DiSK INVentory, and SLSREC holds the SaLeS RECords. TNAMES holds the name and address list and must be sorted on the NM field. I have listed the STRUCTUREs for three of these Dbs in the lower right corner of sheet 9.1.2 to help you understand and enter the information needed. I have squashed them slightly to make them fit in the corner, but all the information is there. Let's look at GRF1, my graphic Db, first. It contains 7 fields. Field #1, GRFNM, is a C)haracter field with a maximum width of 10. This field will hold our graphic name, such as OHIO or DISK. This Db must be SORTed ON GRFNM to allow TIB to FIND a graphic by its name. Fields 2 through 7 are X-type fields, and will contain our graphic and other interesting stuff. I have listed the data from GRF1 at the bottom of this page. It was quite confusing to me at first, so don't expect to grasp the idea immediately. Special Note: (04/11/89), I told Dennis Faherty about these new discoveries in graphics last night and he was amazed. Until then he had not grasped the unbelievable potential of the X-type field even though he created it. The X-type field is very special. You type in normal ASCII character like 123ABC. The character must be entered in a very special pattern, but the actual entry is easy. The character must be entered in pairs because TIB will interpret them as Hexidecimal numbers. For example, the pair of characters 1B would be interprited by TIB as Hex for the number 27 and 27 to your printer means ESC, or escape. We now have the basics for all printer control codes.

I am going to explain some of the data in GRF1 in a condensed form to give you a quick look at what is going on. Note: The ")" is generally used to denote a Hex number. The line GR1 (under DISK) goes like this. 1B =)1B = 27 = ESC, 33 =)33 = 51 = 3, 18 =)18 = 24, or (ESC)"3"CHR$(24), 24 bei- 24/216", right out of the printer users manual. To my printe- that phrase means, set line spacing to 24/216ths inch. This is needed for graphics. Next, 1B =)1B = 27 = ESC, 4B =)4B = 75 = K, 1F =)1F = 31, 00 =)00 = 0, or (ESC)"K"CHR$(31)CHR$(0). To my printer this says, print normal-density graphics, 31 bytes of graphic data will follow. The printer will then consider the next 31 bytes of data to be part of the graphic itself. As you can see we are taking the characters two at a time, 1B, 33, 18, 1B, 4B, etc. Each of these small units is one byte. This means that the printer will actually need 62 characters from this data field to make a total of 31 bytes. "I hope this is helping and not confusing you." So the FF, AA, D5, etc. are used to print the first line of the DISK graphic. At this point I have designated 6 lines for a graphic in the Db, but the biggest graphic is OHIO which takes only 5 lines. I have temporarily placed the characters 0D0A at the beginning of lines GR6. 0D = ASCII code 13, or CR, and 0A = ASCII code 10, or LF. If printed, GR6 will produce a Carriage Return and a Line Feed. The reasoning is this. Printing GR1 will not force the printer into action. Apparently a LF is needed to cause the data in the printer buffer to be printed on the paper. GR6 contains both a CR and an LF, so printing out the complete disk graphic would go like this: PRINT GR1,GR6,GR2, GR6,GR3,GR6,GR4,GR6. In DISK, GR5 graphically prints a blank. Remember GR6 contains CR,LF. We also have the ability to use the TIB system CR, LF. They are entered like this: PRI GR1,(CR),(LF),GR2, (CR),(LF),GR3,etc. This is too much typing for me, but the functions can be used together to an interesting advantage. The line PRINT GR1,(CR),GR1,GR6,GR2, (CR),GR2,GR6,etc. will produce a double struck graphic. GR1 will be printed, the (CR) will return the carriage to the beginning of that line without a line feed, the second GR1 will then be printed for a double strike affect, and the GR6 will then send a CR, LF to set up for the next graphic line. As we progress in this area I don't see any reason why it wouldn't be possible to produce sales reports or purchase orders. These printout sheets can incorporate letterheads with graphics that could be anything from disks, as I used here, to intricate company logos. Once the graphic database is created you can use it anywhere.

Next Page.

0000 **Database: GRF1**

```
GRFNM DISK
GR1   1B33181B4B1F00FFAAD5AAD5AAD5AAD5AAD5AAD5AAD5AAD5AAD5AAD5AAD5AAD5AAD5AADFABFB0000
GR2   1B33181B4B1F00FFAA55AA55AA55AA55AA57AF5CB858B858B85CAF57AA55AA55AA55AAD5AAFF0000
GR3   1B33181B4B1F00FFAA55AA55AA55AA55AAD5FA351A1F1B1F1A35EAD5AE5FAE55AA55AA55AAFF0000
GR4   1B33181B4B1F00FFAB55AB55AB55AB55AB54AB55ABFDFFFDAB55AB54AB55AB55AB55AB55ABFF0000
GR5   1B33181B4B1F0000000000000000000000000000000000000000000000000000000000000100
GR6   0D0A0000000000000000000000000000000000000000000000000000000000000000000000
```

0001

```
GRFNM OHIO
GR1   BB33181B4B2100070404040404040404040402020101000000010101020202040404080B101F00
GR2   1B33181B4B2100FF000000000000000000000000808080800000000000000000000000FF00
GR3   1B33181B4B2100FF00000000FFB1B1B1FF00FF101010FF00FF00FFB1B1B1FF0000000000106F800
GR4   1B33181B4B2000FC060202010100000000000000000000000030206081020204080000010000
GR5   1B33181B4B200000000000000080808040402040408070080830C000000000000000000010000
GR6   0D0A0000000000000000000000000000000000000000000000000000000000000000000000
```

TI-BASE - From INSCEBOT TUTORIAL 9.1.4 By Martin Smoley

NorthCoast 99'ers - April 15, 1989

The next Db is MSRET. This is a good example of how to use a Db to store information that would have previously taken up LOCAL variable space. I have used all 17 fields. Even though I don't need them at this time, I will probably use them later. You will notice that I have mixed C)haracter fields with X-type fields in a normal database. This is a very important development. If you need to do a very special printout or you have paper size restriction etc., special control codes could be saved with individual data records to automatically change the printer settings for special fields. A simple example of this would be a Db containing 100 names, which must be printed weekly for inventory. 90 of those names have 80 characters or less, and the other 10 names have from 80 to 110 characters. The standard form you print on has 85 spaces to print the names. You can include special control codes with certain names to change the print pitch, or micro-justification if you have a very expensive printer, and the names will fill the space perfectly every time. I am probably confusing you with ideas, so let's get back to the subject. MSRET is self explanatory. As you will see late, I use these fields to print labels and screen messages. There is only one record in this Db, the one listed below. A more complicated system could use more records for different labels and different messages.

Database: MSRET

1	FR	From:
2	TO	To:
3	NAME ...	Martin A. Smoley
4	STREET .	6149 Bryson Drive
5	CTSTZP .	Mentor, OH 44060
6	CTN	** CAUTION **
7	CD	COMPUTER DISK
8	DNB	DO NOT BEND OR FOLD
9	DNX	DO NOT X-RAY
10	MS1	*******
11	MS2	* ORDER *
12	MS3	* MORE *
13	MS4	* DISKS *
14	CD1	1B33180000000000000
15	CD2	1B33240000000000000
16	CD3	00000000000000000000
17	CD4	00000000000000000000

The screen below is the structure of SLSREC. It will be used to save 3 items. The ID number of the member requesting disks, the date the disks were shipped and the quantity of disks that were shipped to that ID number, or member.

CREATED 04/01/89 CHANGED 04/05/89

FIELD	DESCRIPTOR	TYPE	WIDTH	DEC
1	ID	C	007	
2	SDT	D	008	
3	QTS	N	004	00

000 1 SLSREC 00000/00063

The SLSREC Db should be created, but left empty. The CF will fill in the data automatically each time disks are shipped. TNAMES is opened last, and I hope , needs no further explanation. The statement REPLACE MORE WITH "Y" will get us into the WHILE loop. "I hope that most of this standard stuff is familiar to you by now." You are then asked to enter a number for the NM field you wish to FIND. If you enter a zero, all databases will be closed and the CF will be ended. That's a quick way out that I may modify later. If you enter a good number it will be found by TIB, the statement IF (NM = SEL2) will be true and DO DSK2.DSKNAP1 will be executed. DSHNAP1 is the Command File I created to display the name, if found, on the screen so you can decide if it is the name you want. However, I am going to leave that until next months tutorial.

TI-Artist Instances To TI-Base

Wes Richardson has said that he would attempt to write a program to convert TI-Artist Instances into a format that can be imported into a TI-Base Database using the Convert function. Knowing Wes' past record I would eliminate the word attempt from that statement. The creation of this type of program will open up a new world to the TI-Base user. There are currently large quantities of graphics available to everyone. There are also many programs arrund to change those graphics to TI-Artist Instances. A program to change Artist Instances to data in a TI-Base database would give TI-Base users more tools in this area than had ever been imagined before. If everything goes well, the conversion program will be published in the NorthCoast Newsletter in the next couple months. Hopefully that timing will bring the program out at the end of my mini series on graphics and exactly when you are ready for it.

HORIZON RAMDISKS

I must throw in a plug for Bud Mills. I forced myself to do this graphic series on a standard disk drive to experience the speed of the system. I must say that it is too slow, and too noisy. If you have a real need for a database system like TI-Base then you will probably put your TI through some heavy use. In that case a Horizon Ramdisk is the only way to go. Bud Mills has been unbelievably helpful and supportive to me for the whole time I have known him, and I hear the same story from other people. His Ramdisks are a top quality item, they are super fast compared to a normal disk drive and they don't make a sound. If you're interested, I recomend that you call Bud at (419) 385-5946 and get further information.

THE MULTI USERS GROUP CONFERENCE

If everything works out I hope to see many of you at the Lima, Multi Users Group Conference on May 20. I am looking forward to the possibility of meeting many of the users I know by mail or by phone, but haven't met face to face yet. I am currently shuffling through ideas for my scheduled demonstration of TI-Base for the conference. As the NorthCoast members will tell you, the biggest problem is getting me to shut up. Quite a few NorthCoast members are planning to attend. I think we will all have a great time and make a lot of new friends. Try to make it if you can.

Continued Next Month.

G-GRAPHICS

by WESLEY R. RICHARDSON
BLUEGRASS 99 COMPUTER SOCIETY, INC.

G is a new graphics language which was developed by Gene Krawczyk of the Adelaide TI Computer Club in South Australia. The version which I used for the following programs was edited by Bob Warren to produce a beginners version of G.

The structure of G programs is very similar to BASIC, except there are no program line numbers. G is also similar to the LOGO turtle mode, but you don't run out of ink. There is a program editor which allows you to Load, Save, Init, Edit, or Run G programs. Files are save in DIS/VAR 80 format, so they may be accessed from TI-Writer.

When a G program is running, the effect is an antimated drawing of lines and patterns on the screen in graphics mode. The speed of the drawing can be slowed down by using delay loops.

I personally enjoyed programming in G, and am anxious to see the Advanced version. The first program listing is G-SHAPES which creates some patterns on the screen.

The second program is 99999-G which is a nested loop routine to count to 99,999. The original COUNT 99999 program was listed in the December, 1987 BYTEMONGER and was updated in March, 1988. This brings to ten the number of TI languages that have run COUNT 99999. This is only a comparison of speeds for this particular counting routine, and please keep in mind that each programming language has its advantages.

LANGUAGE	COUNT 99999 RUN TIME	RATIO
BASIC	1 HR 51 MIN 39.4 SEC	295.1
G	1 HR 50 MIN 25.0 SEC	291.1
99 FORTRAN	0 HR 55 MIN 21.9 SEC	146.3
XBASIC	0 HR 44 MIN 13.1 SEC	116.9
PASCAL	0 HR 37 MIN 32.5 SEC	99.2
LOGO	0 HR 35 MIN 28.8 SEC	93.8
TURBO PASC	0 HR 02 MIN 48.0 SEC	7.4
FORTH	0 HR 01 MIN 36.5 SEC	4.3
c99	0 HR 01 MIN 28.0 SEC	3.9
ASSEMBLY	0 HR 00 MIN 22.7 SEC	1.0

The third program listing in G is an updated version of the CLOCK program which was on the G-GRAPHICS disk.

```
REM 99999-G
REM TI-99/4A  G-GRAPHICS
REM WESLEY R. RICHARDSON SEP 1988
REM G-GRAPHICS RUN TIME 1 HR 50 MIN 25.0 SEC
CLS
SCREEN 7
COLOR 15
:START
DISPLAY  88 128 "C"
DISPLAY  96 128 "O"
DISPLAY 104 128 "U"
DISPLAY 112 128 "N"
DISPLAY 120 128 "T"
DISPLAY 136 128 "Y"
DISPLAY 144 128 "/"
DISPLAY 152 128 "N"
DISPLAY 160 128 "?"
:KEY
KEY$ K
IF K=89 THEN GOTO :LOOP
IF K=255 THEN GOTO :KEY
GOTO :END
:LOOP
CLS
DISPLAY 120 96 ","
FOR V=0 TO 9
DISPLAY 104 96 V
FOR W=0 TO 9
DISPLAY 112 96 W
FOR X=0 TO 9
DISPLAY 128 96 X
FOR Y=0 TO 9
DISPLAY 136 96 Y
FOR Z=0 TO 9
DISPLAY 144 96 Z
NEXT Z
NEXT Y
NEXT X
NEXT W
NEXT V
GOTO :START
:END
STOP
```

```
REM G-SHAPES
REM WESLEY R. RICHARDSON 880822
REM BLUEGRASS 99 COMPUTER SOCIETY, INC.
REM G-GRAPHICS TI-99/4A
REM SPRIAL
SCREEN 7
COLOR 15
SET 120 85
LET K=0
FOR I=0 TO 2160 STEP 6
LET K=K+1
ANGLE I
DRAW K/30
NEXT I
GOSUB :WAIT
REM BURST
CLS
COLOR 8
SCREEN 15
SET 104 88
LET K=42
FOR I=0 TO 1080 STEP 6
ANGLE I
DRAW K
ANGLE I+183
DRAW K
K=K+1
NEXT I
GOSUB :WAIT
REM FLOWER
:FLOWER
CLS
SCREEN 3
COLOR 2
FOR J=48 TO 192 STEP 48
FOR I=1 TO 3
FOR K=89 TO 111
SET I+J K
NEXT K
NEXT I
NEXT J
FOR J=24 TO 216 STEP 48
FOR I=1 TO 3
FOR K=153 TO 175
SET I+J K
NEXT K
NEXT I
NEXT J
COLOR 11
LET A=15
LET B=60
LET C=5
FOR J=48 TO 192 STEP 48
SET J 72
GOSUB :ROWS
NEXT J
FOR J=24 TO 216 STEP 48
SET J 136
GOSUB :ROWS
NEXT J
GOSUB :WAIT
GOTO :POLY
:ROWS
FOR I=1 TO 6
GOSUB :PETAL
LET A=A+60
NEXT I
RETURN
:PETAL
GOSUB :ARCR
LET A=A+120
GOSUB :ARCR
LET A=A+120
RETURN
:ARCR
FOR D=1 TO B/10
ANGLE A
DRAW C
LET A=A+10
NEXT D
RETURN
REM POLY
:POLY
CLS
SCREEN 9
COLOR 15
LET A=0
LET D=20
SET 125 100
FOR I=1 TO 50
LET A=A+10
LET D=D+2
FOR J=1 TO 4
LET A=A+90
ANGLE A
DRAW D
NEXT J
NEXT I
GOSUB :WAIT
REM POLY2
:POLY2
SIZE 1
FOR I=3 TO 20
CLS
LET A=180
LET D=420/I
SET 128 24
ANGLE A
DRAW D/2
LET A=0
FOR J=1 TO I
ANGLE A
DRAW D
LET A=360/I+A
NEXT J
PRINT 112 64 I
GOSUB :WAIT
NEXT I
PRINT 16 160 "PRESS ANY KEY"
STOP
:WAIT
FOR W=1 TO 2000 NEXT W
RETURN
```

```
REM CLOCK
:START
CLS
 BGND 11 SCREEN 3 COLOR 13 SIZE 1
 PRINT 60 0 "THE TIME
SIZE 0
BGND 15 5376 780
 COLOR 14
COLOR 1 PRINT 105 30 "ANALOG"
COLOR 10 PRINT 20 175 "DIGITAL"
 COLOR 15
 PRINT 120 46 "12" PRINT 124 138 "6"
 PRINT 80 94 "9" PRINT 170 94 "3"
 P=32 LET L=360/60 T=360/12
 ARC 128 96 52 52 0 361
 TRACE 4 BCOLOR 15
 FOR I=0 TO 32 STEP 1
 ARC 128 96 I I 0 361 NEXT I
 TRACE 1
 COLOR 8 FOR I=0 TO 40 STEP 1
 ARC 128 96 I I 0 361 NEXT I
 SIZE 1 COLOR 8 PRINT 115 170 ":"
:HERE TRACE 2
 FOR Y=270 TO 360+270 STEP T
 ANGLE Y SET 128 96 DRAW 20
 G=Y-270/30
 COLOR 4 PRINT 100 170 G
FOR I=270 TO 354+270 STEP L
COLOR 8 SET 128 96
ANGLE I
DRAW P
J=I-270/L
COLOR 6 PRINT 130 170 J
 FOR C=0 TO 10 STEP 1 NEXT C
FOR W=1 TO 428 NEXT W
COLOR 8 SET 128 96 DRAW P
COLOR 7 PRINT 130 170 J
NEXT I
COLOR 8 ANGLE Y SET 128 96 DRAW 20
PRINT 100 170 G
COLOR 8
NEXT Y
CLS
STOP
```

CORRECTION TO LAST MONTH'S PLUS TEMPLATE
HARRY HOFFMAN

In regard to the April article "PLUS & TI-WRITER", there are some ommisions. When I gave the demo at the TI-CHIPS meeting, I explained the discrepancies in the sheet that was passed out. It should have been on $THAT SHEET!

All commands to the printer $MUST be preceded by a period (.). I used a comma in front of the TL & LM commands, so the Formatter would print them. I should have used the Editor's PF (PrintFile) to make my printout and there wouldn't be any confusion. I hope that I have not caused anyone any trouble!!!

One other little mistake! The missing $CARRIAGE RETURNS! After the Transliterate (TL) commands, there should be a (CR) carriage return. Putting the CR there allows you to enter comments about the command. I had to delete them; again, to allow me to use the Formatter to print the sheet the way it looks on the handout that was in the Newsletter.

Thank you for your patience, Harry Hoffman

CLEVELAND AREA 99/4A USERS GROUPS
C/O DEANNA SHERIDAN
20311 LAKE ROAD
ROCKY RIVER, OH 44116

CHECK YOUR EXPIRATION DATE.
THIS MAY BE YOUR LAST ISSUE!