



MID ILLINOIS COMPUTER RESOURCE
ORGANIZATION
P. O. BOX 766 BLOOMINGTON, IL
61701-766

BLOOMINGTON - NORMAL MICRO NEWSLETTER
JUNE 1985, VOL. 3, No. 6

Welcome to the age of the computer in the home! Future meetings will be held the third THURSDAY of each month at 7 P.M. at the IAA Building in Bloomington. Future dates include JUNE 20, JULY 18, and AUGUST 15.

The program for JUNE will consist of:

- * Our normal business meeting.
- * Program demonstration by Ray Hinrichsen
- * Electronic mail demo by Sherwood Smith and Aubrey Johnson.
- * Special Interest Group (SIG) Meetings

***** PRESIDENTS NOTES *****

May was our first meeting in the IAA building. I would to thank Ray Fisher and Aubrey Johnson for getting the room for our group. The room itself is the right size for us. There is seating for about 30 people around the tables. Unfortunately, only 8 members showed up. I understand some had commitments with school activities. Actually, with having only a few people being there, it was enjoyable. We spent considerable time talking about our TI's and demonstrating programs other than those planned. Hopefully, more will attend this months meeting as our new room is very nice.

Steve Maupin brought a visitor to last months meeting. His name is Bob Roth and is a member of a TI user group in Pekin. They are newly formed and have about 15 members. One of their members runs a TI bulletinboard. I plan on attending one of their meetings in the near future.

Out of room. See everyone June 20. Brian McFeeters

Our new library and membership cards have been distributed. If you did not receive yours, please see Ray Hinrichsen at the June meeting. You will need a new 1985 library card to check-out disks or cassettes at the meeting.

Sam Shank has completed the library program trade with the W.W.99'ers user group in Champaign. The programs will have to put on SS/SD format for library use. Volunteers will be needed to help download the new programs to disk and cassette. See Sam if you would like to help.

C.ITOH is offering a toll-free, technical hotline, that allows you to speak directly with a technician regarding any questions or problems that you may have with their products. They are the manufactures of the PROWRITER line of printers.

1-800-423-0300

Hook Up Your Computer To The TV Without Interference by Dale Mann

Your TV works fine. So does your computer. But connect one to the other and your screen display goes haywire.

A cheap and easy solution: Replace the cable that connects the TV to the computer with another that's the same type but long enough to allow 24 inches of slack. Wind it in a coil around a cardboard tube (such as a paper towel core) until you reach 10 or more turns. Make sure the turns are even and tightly spaced, and lie flat (don't overlap one turn of the cable onto another). Wrap a sturdy layer of tape or foil around the coil to prevent it from coming unwound.

Place the coil near the back of the TV next to the antenna lead-in terminals. Connect the ends of your cable to the TV and the computer in the usual manner.

Voila! The coil acts as a directional antenna, which you can aim to minimize the effects of broadcast interference. You will have to play with angle to perfect screen display.

Thanks again to Jim Peterson for his TIPS FROM THE TIGERCUB. Also, thanks to Steve Maupin for his review of Spinnaker's Facemaker cartridge.

Unisource, Tex-Comp, and Triton are selling TI cartridges for as low as \$4.95. Quantities on some are limited so if want any order soon.

The International Users Group has filed for protection under Chapter 7 of the Federal Bankruptcy Act. Their current library as well as some 400 new programs will be marketed by a newly formed company which will be announced 6/1.

* REVIEW *
* OF *
* FACEMAKER *
* BY *
* STEVE MAUPIN *

One of the main reasons I got my computer was for an educational tool for my children. Computers are great teachers. They have infinite patience and never get mad. Spinnaker Software Corporation makes one of my children's favorite programs. Not only is it my kids favorite, but also the neighborhood kids too. The recommended ages are 4-12. This cartridge is a great way to introduce a child to the computer as there is very little reading involved.

Facemaker provides three activities: 1) build a face, 2) program a face, and 3) play game. My kids always start out with building a face. In this activity, you are shown a screen with a blank face on it and to the side is a picture of five items; a mouth, eyes, ears, nose and hair. Select any number next to a facial feature and a new screen appears. This screen is the option list. For example, if you press the number next to the mouth, a screen of 8 different mouths appear for you to select from. Each mouth is different and some are just plain funny. After selecting the mouth you want, the screen with your blank face appears with the mouth you've selected. You repeat this with the other facial features until you complete your face.

Now that you have built a face, what do you do with it? You can program the face, in the next activity, to frown, wink, smile, cry, wiggle it's ears and stick out it's tongue. This can be done in any order and up to 25 facial movements in a row. This is my children's favorite part of the program.

The last option, the game, is a kind of matching test. The computer builds a face and then the face does a facial movement along with a corresponding sound. If the child presses the right letter key identifying the movement, a new movement is added. A perfect game is matching all 25 movements.

Facemaker is my children's favorite game. It's easy to see why, it's fun and lets a child create something he can instantly see. Facemaker can be ordered from most software dealers for \$9.95.

Tip from the Airport Area Computer Club March85 newsletter which searches for a program on any disk drive if you know the disk name. Can be used in your program or in the command mode.

RUN "DSK.diskname.program name"

TIGERCUB TIPS FOR BEGINNERS

by Jim Peterson

TIP #1 - Don't use EDIT! There are two ways to bring a program to the screen. You may type EDIT, the line number, and press Enter...or you may type the line number and press FCTN together with the down-arrow or up-arrow keys. When you graduate to Extended Basic, you will find that you can only use the second method. Then while you are trying to break the EDIT habit, you are apt to become confused, type the line number, hit Enter...and delete the line completely!

TIP #2 - Don't waste time typing LET. The computer understands I=1 as readily as LET I=1. You do not have to leave a space after the line number - the computer will do it for you. In Extended Basic, you don't have to leave a space before and after the :: statement separators - the computer will do that for you too, moving the statements to additional lines if necessary.

When programming in Basic, don't use character sets 15 and 16 (ASCII codes 144-159) unless you really need them for colors and redefined colors. And if you use multiple colons to scroll the screen, put a space between them : : :(Or better yet :;:;:;). Then your program will usually also run without modification in Extended Basic.

TIP #3 - If you have the Extended Basic module, why not leave it plugged in and select the Extended Basic option even when you are programming in Basic. This will allow you to type 5 lines on a line number (unless you put too many short items in a DATA statement), and the Extended Basic will accept input of your program lines much faster, especially when the program gets long. It also accepts changes and deletions much more quickly, and is more useful when you want to delete a large number of lines. It will quickly tell you how much memory you have left with the SIZE command (but you'll have more in Basic). It will bring your rejected input back to your screen for correction with FCTN 8. It will also run your program, if you stay away from character sets 15 and 16. Watch those double colons. In Basic, you can write: 100 PRINT "HELLO":::~::~:"GOODBYE" to scroll between the lines of print. In Extended Basic, the double colon :: is used to separate multiple line statements. When you load a Basic program in Extended Basic, you will usually find that the computer will rearrange the colons "HELLO" : : : : "GOODBYE" and run the program properly. But sometimes, especially if you put colons in front, as 100 PRINT :::"HELLO", the computer may become confused and give you some very puzzling messages. On rare occasions it may even rearrange the colons into strings of double sets :: :: and that will lock it up completely! So, even when programming in Basic it is a good idea to separate your colons : : : or better yet, ;:;:;. To get the computer to read the CALL KEY input as upper case letters, even if the ALPHA lock is up, just use key-unit 3 - CALL KEY(3,K,S). To cancel it use key-unit 5.

TIP #4 - I have found frequent use for Craig Miller's routine CALL INIT :: CALL LOAD(-31888,63255). This shuts down all of the disk files and makes it possible to load programs over 12K long from tape, and copy them back to tape, without having to physically disconnect the drives.

USING "PRINT USING" WITH YOUR PRINTER by Rich Roberts D/L From SOURCE (CRAFT)

One of the more obscure statements available with TI Extended BASIC is one called PRINT USING. Even more obscure is the fact that this statement can be used to format variables and constants that will be dumped to your printer. The Extended BASIC manual, on page 150, shows several examples of how PRINT USING can be used to format data for screen display, but nary a word of how to do the same with open files. It can be done, and is much more powerful than you may realize.

Any discussion of PRINT USING will require an understanding of the IMAGE statement, so if you are not familiar with it, you better brush up on it first. The PRINT USING statement uses IMAGE in one of two ways, either with a string expression, or a line number reference. I prefer the latter, as it allows for more flexibility, but since these different methods are explained in the manual, I will limit this to a few simple examples that are not shown in the manual.

```
100 TCOST=19.55
110 IMAGE ##.##
120 OPEN #1:"PI0"
130 PRINT #1,USING 110:TCOST
```

Running this sample program will effectively show how the PRINT USING statement will work with an open file. Of course, there are many other variations of IMAGE that can be used, so experiment with them and watch how it performs when line 130 dumps it to the printer. Shown below are a few more examples for use with an open file.

```
110 IMAGE "##.##    ##.##"
130 PRINT #1,USING 110:COST1,COST2
```

This IMAGE statement will allow you to print two (or more) variables at a pre-determined spot on the same line. The length of the string expression in the IMAGE statement can be as long as you wish, up to the limit of an Extended BASIC line.

```
110 IMAGE "#####    ##.##"
130 PRINT #1,USING 110:"TOTAL COST",TCOST
```

This version shows how you can format the printed line for string data as well as numerical data. A string variable could be used in place of the string constant, as below.

```
105 A$="TOTAL COST"
110 IMAGE "#####    ##.##"
130 PRINT #1,USING 110:A$,TCOST
```

It is also possible to place the IMAGE statement inside the PRINT USING statement, as shown below. First, delete line 110.

```
130 PRINT #1,USING "##.##":TCOST
or
130 PRINT #1,USING "#####    ##.##":A$,TCOST
```

A few other points to remember include the fact that IMAGE and PRINT USING can be used to round off calculated variables. A single string expression such as "#####.##" will round off and decimal align numbers as small as .01 up to 999999.99, and print the number at any designated location. This function could save many hours of algorithm development for accomplishing the same thing. So, in the long run, the PRINT USING statement is one that any programmer should be very familiar with, and use as much as possible.

(hope this helps,.....chuck craft)

TIPS FROM THE TIGERCUB

020

Copyright 1985

TIGERCUB SOFTWARE
156 Collingwood Ave.
Columbus, OH 43213

Distributed by Tigercub Software to TI-99/4A Users Groups for promotional purposes and in exchange for their newsletters. May be reprinted by non-profit users groups, with credit to Tigercub Software.

The entire contents of Tips from the Tigercub Nos. 1 through 14, with more added, are now available as a full disk of 50 programs, routines and files for just \$15.00 postpaid!

Nuts & Bolts is a diskfull of 100 (that's right, 100!) XBasic utility subprograms in MERGE format, ready for you to merge into your own programs. Contents include 13 type fonts, 14 text display routines, 12 sorts and shuffles, 9 data saving and reading routines, 9 wipes, 8 pauses, 6 music, 2 protection, etc., and now also a tutorial on using subprograms, all for just \$19.95 postpaid!

And I have about 140 other absolutely original programs in Basic and XBasic at only \$3.00 each!(plus \$1.50 per order for cassette, packing and postage, or \$3.00 for diskette, PPM) Some users groups charge their members that much for public domain programs! I will send you my descriptive catalog for a dollar, which you can then deduct from your first order.

Come on now, folks, don't you support your local schools? And don't you support those who support

you? There are thousands of schools which have TI-99/4A computers in the classroom, usually without disk drive and without Extended Basic. They could use some educational programs in Basic on cassette. They could probably use some of the public domain software in your library. Maybe they could use some of the educational programs I sell for just \$3 (and I authorize schools to copy them for use within the school). There is probably such a school in your area - is your group supporting it? In the last Tips, I asked the members of 101 users groups to give me the addresses of schools that had TIs, so I could send them a free catalog. How many addresses did I get? Zero to the power of zero times zero!

More on the pestiferous asterisk bug in TI-Writer. Dr. Guy-Stefan Romano has confirmed and explained it. If you are printing out of the Formatter mode and your text contains an asterisk followed by two or more numeric digits - the asterisk and two digits will disappear! For instance, A#256 becomes A6, and I've noticed that A6 in programs published in several newsletters recently.

The TI-Writer program misinterprets the asterisk and two digits as an instruction to input data from a "value file" (see Alternate Input on p. 111 of the manual).

The solution to this bug is to type two asterisks followed by two dummy digits, then the actual digits. For instance, instead of A#256 type A##25256. Trouble is, the bug usually shows up in a program which has been LISIed to disk and then

MERGED into TI-Writer, and is usually not noticed. The solution? Run the program through my 28-Column Converter (see Tips #18!).

Dr. Romano informs me that there is an even worse bug in the Transliterate command coding, erratic and sometimes destructive. It is triggered by certain sequences of characters, but these have not been documented.

Dr. Romano says that he does not use transliteration.

I would suggest that you also avoid the use of the & and @. The & will only underline a single word, unless you tie words together with the ^ sign. If you tie words together, the Fill and Adjust will leave gaping blanks in your lines and if you tie too many together the line will extend beyond the right margin! Also, the underlining is a broken line. It is better to use the escape codes CTRL U, FCTN R, CTRL U, SHIFT -, CTRL U, SHIFT A, CTRL U, which will give a solid underline until you turn it off with CTRL U, FCTN R, CTRL U, SHIFT -, CTRL U, SHIFT @, CTRL U.

The @ is handy to emphasize a single word, but if you want to double-strike a whole sentence or paragraph it is better to use the escape code CTRL U, FCTN R, CTRL U, SHIFT G, and turn it off again with CTRL U, FCTN R, CTRL U, SHIFT H.

The period bug is another killer - the Formatter thinks that any line which begins with a period is a formatter command, and deletes the whole line! If your text contains a decimal value such as .11 and the wraparound puts it at the beginning of a line, the

line disappears! There are two ways around this - put a 0 in front of all your decimals, as 0.11, or transliterate all your periods.

In all, the TI-Writer formatter is a temperamental and unpredictable piece of software, prone to unwanted line feeds and unexpected paper-wasting form feeds. I like to use it to right-justify text back to the disk, but from then on I prefer to print it out of the editor mode, or out of my own program.

Designing downloadable characters for the Gemini printer (see page 115 of the manual) is a bit tricky because it is hard to visualize how the expanded pattern will appear in print. The following program will enable you to experiment with designs, dump them directly to the printer for viewing, then save them as a file. When you later dump this file into printer RAM for use, you must activate the download characters with the escape code -
CHR\$(27);CHR\$(36);CHR\$(1).

```
100 CALL CLEAR :: CALL SCREE
N(4):: CALL CHAR(128,"FFB1B1
B1B1B1B1FF",129,RPT$( "F",16)
):: CALL COLOR(13,2,16)
110 FOR R=9 TO 15 :: CALL MC
HAR(R,11,128,9):: NEXT R
120 X=1 :: FOR R=9 TO 15 ::
DISPLAY AT(R,7)SIZE(2):STR$(
X):: X=X*2 :: NEXT R :: FOR
C=9 TO 17 :: DISPLAY AT(8,C)
SIZE(1):STR$(C-B):: NEXT C
130 DISPLAY AT(2,9):"TIGERCUB
B'S" :: DISPLAY AT(4,1):"GEM
INI CHARACTER DOWNLOADER" !p
rogrammed by Jim Peterson fo
r the Public Domain
140 DISPLAY AT(17,1):" Move
cursor with W,L,R,S,D,";"2,X
and C keys. Toggle on;"and
off with Q key. Press;"Ent
er when finished." :: "Pres
```

```

s any key"
150 CALL KEY(0,K,ST):: IF ST
=0 THEN 150 :: CALL MCHAR(17
,1,32,224)
160 R=9 :: C=11 :: CH=12B
170 CALL MCHAR(R,C,32):: CAL
L MCHAR(R,C,CH):: FOR D=1 TO
10 :: NEXT D :: CALL KEY(3,
K,ST):: IF ST=0 THEN 170
180 ON PUS("MWERDCXZS"&CHR$(
13),CHR$(K),1)+1 GOTO 170,31
0,230,220,210,200,190,260,25
0,240,330
190 R=R+1
200 C=C+1 :: GOTO 270
210 C=C+1
220 R=R-1 :: GOTO 270
230 R=R-1
240 C=C-1 :: GOTO 270
250 C=C-1
260 R=R+1
270 R=R-(R<9)+(R<15):: C=C-(
C<11)+(C<19):: IF CH=12B THE
N 300 :: CALL GCHAR(R,C-1,6X
):: CALL GCHAR(R,C+1,6Z):: I
F (6X<>129)&(6Z<>129) THEN 30
0
280 DISPLAY AT(22,1):"You ca
n't have two in a row":"horiz
ontally!" :: FOR D=1 TO 50
:: NEXT D :: DISPLAY AT(22,1
):" -"
290 CH=CH-1
300 CALL MCHAR(R,C,CH):: BUT
D 170
310 CH=CH+1+(CH=129)&2 :: IF
CH=12B THEN 320 :: CALL GCH
AR(R,C-1,6X):: CALL GCHAR(R,
C+1,6Z):: IF (6X<>129)&(6Z<>
129) THEN 320 ELSE 280
320 CALL MCHAR(R,C,CH):: GOTO
D 170
330 FOR C=11 TO 19 :: X=1 ::
FOR K=9 TO 15 :: CALL GCHAR
(R,C,b)
340 IF b=129 THEN A=A+X
350 X=X+2 :: NEXT R
360 FOR J=1 TO LEN(STR$(A)):
: CALL VCHAR(15+J,C,ASC(SEE$(
b)TR$(A),J,1)):: NEXT J ::
M=M&CHR$(A):: A=0 :: NEXT
C :: A=0
370 DISPLAY AT(20,1):"Print?
Y/N Y" :: ACCEPT AT(20,12)V
ALIDATE("YN")SIZE(-1):Q$ ::
IF Q$="N" THEN 470
380 IF F=1 THEN 390 :: F=1 ::
: DISPLAY AT(20,1):"Printer
name?" :: ACCEPT AT(20,15):F
$ :: OPEN #1:P$

```

```

390 DISPLAY AT(20,1):"ASCII
to redefine?" :: ACCEPT AT(2
0,20)VALIDATE(DIGIT)SIZE(3):
CH
400 DISPLAY AT(20,1):"Descen
der (0 or 1)? 0" :: ACCEPT A
T(20,21)VALIDATE("01")SIZE(-
1):D$ :: D=VAL(D$)
410 M$=CHR$(27)&CHR$(42)&CHR
$(1)&CHR$(CH)&CHR$(D)&M$
420 PRINT #1:M$ :: PRINT #1:
CHR$(27);CHR$(36);CHR$(1);
430 PRINT #1:RPT$(CHR$(CH),7
2):: PRINT #1:CHR$(14);RPT$(
CHR$(CH),36)
440 DISPLAY AT(20,1):"Save (
Y/N)? Y" :: ACCEPT AT(20,13)
VALIDATE("YN")SIZE(-1):Q$ ::
IF Q$="N" THEN 470
450 IF F3=1 THEN 460 :: F3=1
:: DISPLAY AT(20,1):"File na
me? DSK" :: ACCEPT AT(20,14)
: F$ :: OPEN #2:"DSK"&F$
460 PRINT #2:M$
470 M$="" :: DISPLAY AT(20,1
):"Another (Y/N)? Y" :: ACCE
PT AT(20,16)VALIDATE("YN")SI
ZE(-1):Q$ :: IF Q$="Y" THEN
100
480 CLOSE #1 :: CLOSE #2 ::
END

```

Micropendium ran a contest to improve on a brief ingenious organ program. The winner was Michael Christianson, who wrote a superb program. You'll have to buy the January issue of the magazine to get it (you should be subscribing, anyhow!). I didn't enter the contest, of course, and my version is not nearly as good, but have fun -

```

90 CALL CLEAR
95 PRINT TAB(5):"MICROPENDIU
M ORGAN" :: : : : : "Pl
ay bass with left hand": "o
n left side of keyboard," :
: "melody on the right" : :
100 REM - MICROPENDIUM ORGAN
modified by Jim Peterson
110 OPTION BASE 0
120 DIM NOTE(20)
130 FOR A=0 TO 20
140 READ NOTE(A)
150 NEXT A

```

```

160 DATA 40000,220,247,262,2
94,330,349,392,440,494,523,5
87,659,698,784,880,988,1047,
1175,1319,1397
170 CALL KEY(1,K1,S)
180 CALL KEY(2,K2,S)
190 CALL SOUND(-1000,NOTE(K2
+1),0,NOTE(K2+1)*.01,5,NOTE
(K1+1)*.75-ABS(K1+1=0)*1100
00,30,-4,0+ABS(K1+1=0)*30)
200 GOTO 170

```

A sprite routine that doesn't do anything but look pretty. I call it Patches.

```

50 CALL CLEAR :: CALL SCREEN
(5)
100 A$=RPT$( "AAS5",16):: B$=
RPT$( "F",64):: CALL MAGNIFY(
4):: RANDOMIZE
110 FOR CH=40 TO 136 STEP 8
:: CALL CHAR(CH,A$,CH+4,B$):
: NEXT CH
120 C=2 :: S=40 :: R=1 :: FO
R T=1 TO 24 STEP 2 :: COL=15
0&RND+50 :: CALL SPRITE(#T,S
,C,R,COL,#T+1,S+4,C+1,R,COL)
:: S=S+8 :: C=C+1 :: R=R+15
:: NEXT T
140 FOR T=1 TO 50 :: CALL CO
LOR(#INT(24&RND+1),INT(16&RN
D+1)):: NEXT T :: GOTO 120

```

This is one that I fancied up, based on a sprite routine written by a youngster named Andrew Sorenson, published in the Sydney Newsdigest from Australia.

```

100 ! WILL D' WISP
by Jim Peterson
based on
Andrew Sorenson's
sprite routine
110 CALL CLEAR :: CALL SCREE
N(2):: CR=4B
120 FOR CH=4B TO 63 :: FOR L
=1 TO 4 :: RANDOMIZE :: X=IN
T(16&RND+1)&2-1 :: X$=SEE$(
"0018243C425A667E8199A5BDC3DB
E7FF",X,2):: B$=B$&X$ :: C$=
X$&C$ :: NEXT L :: CALL CHAR
(CH,B$&C$):: B$,C$="" :: NEX
T CH
130 FOR N=1 TO 2B :: CALL SP
RITE(#N,CR,INT(14&RND+3),B$N
+20,120,S,0):: NEXT N :: IF

```

```

CR=64 THEN CR=4B :: T=T+1+(T
=2)&2 :: CALL MAGNIFY(T)
140 X=(INT(3&RND)-1)&4 :: Y=
(INT(3&RND)-1)&4
150 IF INT(10&RND+10)<>10 TH
EN 170
160 CR=CR+1 :: GOTO 130
170 FOR N=1 TO 2B :: CALL MO
TION(#N,-Y&20,X&20):: NEXT N
:: GOTO 140

```

Here are a few more enhancements to my Menu Loader, published in Tips #15. Delete line 150 and add

```

101 OPTION BASE 1 :: DIM P$(
127):: ON WARNING NEXT :: G
OTO 110
105 @,A$,A$,B,C,D$,FLAG,1,J,K
,KD,KK,M$,NN,P$,P$(1),Q$,S,S
T,T$(1),TT,VT,X
CALL INIT :: CALL LOAD :: CA
LL LINK :: CALL PEEK :: CALL
KEY :: CALL SCREEN :: CALL
COLOR :: CALL CLEAR :: CALL
VCHAR :: CALL SOUND :: !@P-

```

The pre-scan will speed up run time by a worthwhile amount. The warning default will prevent a screen scroll on an erroneous Enter.

When you're finished printing strip labels, cut off the strip BEHIND the platen and roll it FORWARD! You'll waste a few labels that way, but if you try to roll backwards and get a gummy label stuck in the works, you've got trouble!

MEMORY FULL

Jim Peterson

the Tigercub

MID ILLINOIS COMPUTER RESOURCE ORGANIZATION
P.O. BOX 766
Bloomington, IL 61701-0766



EDMONTON 99'ERS USER SOCIETY
P.O. BOX 11983, EDMONTON
ALBERTA, CANADA T5J-3L1

```
*****  
*      MMM  MMM  IIIIII  CCCCCC  RRRRRRRR  00000000  *  
*      MM M M MM    II    CC      RR      RR    00    00  *  
*      MM M M MM    II    CC      RRRRRRRR  00    00  *  
*      MM  M  MM    II    CC      RR      RR    00    00  *  
*      MM      MM    II    CC      RR      RR    00    00  *  
*      MM      MM  IIIIII  CCCCCC  RR      RR  00000000  *  
*  
*  
*      The MID ILLINOIS COMPUTER RESOURCE ORGANIZATION  *  
*****
```