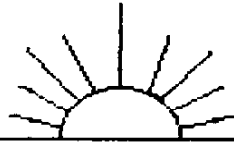


VOL. 8 No. 05

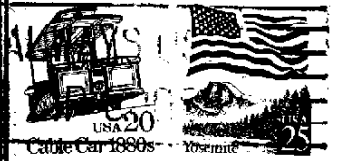
JUNE 1990

NH

NEW HORIZONS
NEWS LETTER



NORTHWEST OHIO COMPUTER CLUB FOR THE TEXAS INSTRUMENTS 99/40
AND THE NYARC GENEVE 9640 PERSONAL AND HOME COMPUTER



ATT. EARL H. HOFFSIS
N.W. OHIO 99'S USERS GROUP
7 FIRST CHURCH UNITY
3535 EXECUTIVE PARKWAY
TOLEDO OHIO 43606

Dallas TI Home Computer GP
PO Box 29853
Dallas, TX 75299

**PRESIDENT'S
CORNER**
Bill Tiep

This is junes newslett-
er and there too many
different subjects to be
covered in this artical,
but i'll touch on a fue.

First off there were
many different pieces of
software at Lima Ohio so
many in fact I have
decided to bring the disks
I copied from there club
disks and will let those
who wish to review them
through the summer break
and can then demo or re-
port back to the club.
At the September meeting
what they found. Don't
forget these can also be
a source of some very
interesting newsletter art-
icales . So on this count
come to the June meeting
to find out more.

As usual there will club
disks on sale and also
MICROpendium at the disk
sells table, also blank
disk. The usual 50/50
drawing for some lucky
winner and the regular
drawing.

So please come to the
June meeting and see what
we brough back from Lima
ohio for the membership
to use. You won't be sorry.

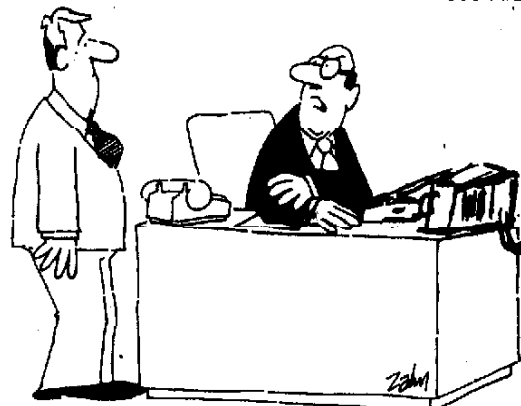
I understand in December
we will be loosing
Roger as our newsletter
Editor so is there anyone
who would be interested in
filling this position start-
ing in January of 1991.
I will close with this.

THIS NEWSLETTER IS PUBLISHED
BY NEW HORIZONS TI-99/4A
HOME COMPUTER USERS GROUP.
MATERIAL MAY BE REPRODUCED
WITHOUT PERMISSION PROVIDED
THAT THE AUTHOR AND SOURCE
ARE ACKNOWLEDGED.

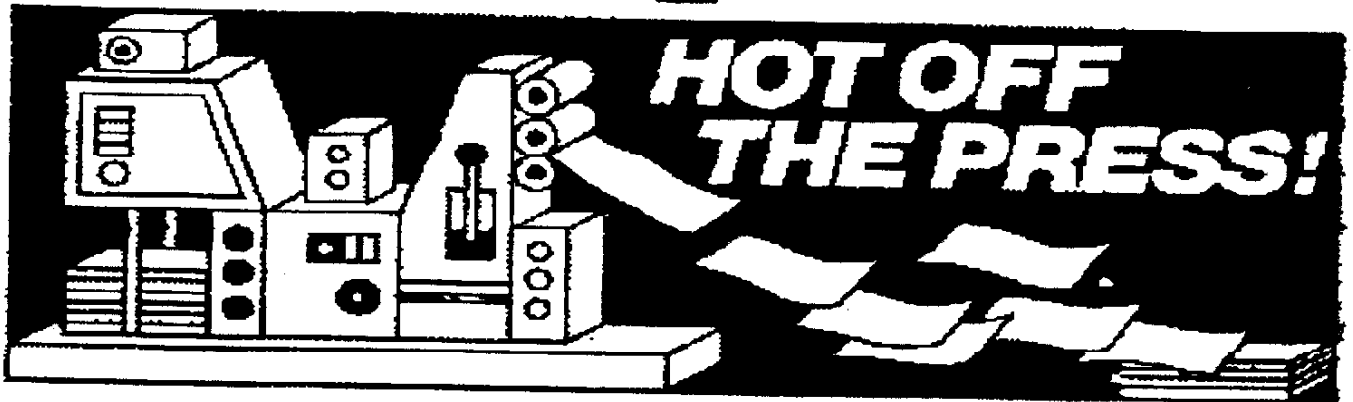
FOR MORE INFORMATION PLEASE
ONE OF THE FOLLOWING
THIS MONTHS MEETING SAT
JUNE 9, 1990 AT 12:30 PM.
BEHIND MENDO'S OFF SECUR
ROAD ON EXECUTIVE DR.

TI-CON BBS
1-419-385-7484
Bill Ttiep
1-419-475-1775
Charles Strobell
1-419-892-3527
Narilyn Schafstall
1-419-882-6870
Earl W. Koffsis
1-419-475-0461
John & Chris Dewey
1-419-475-3871
Burr Mallory
1-419-882-6769
Roger & Judy Feinauer
1-517-263-6144

THE WALL STREET JOURNAL



"Your report isn't worth the disk it's saved
on!"



NEW HORIZON MINUTES

The meeting of the New Horizon Computer Club was called to order at 12:50 p.m. on May 12, 1990, by president, Bill Tiep.

Due to the absence of the secretary the minutes of the April meeting were not printed in the newsletter; therefore, they were read. Roger Feinauer moved, and Dan Block seconded the motion that the minutes be approved.

Earl gave the Treasurer's report: Balance as of the April meeting was \$405.89; the current balance in the treasury is \$390.94. Report accepted.

Under Old Business the questionnaire was discussed. Only two or three more people remain to be contacted by a member of the group. Those we are not able to reach will receive notes printed in the newsletter.

Earl reported that the printing company, J. Brett's, that has been printing our newsletter and giving us a discount, has been sold. This month he had a great deal of trouble trying to contact the new owners, and picking up the finished newsletter at a time the business was opened. Therefore, he asked for permission to do the printing at the church until such time as a more suitable printer can be found. Roger Feinauer moved; Margaret Dixon seconded; motion passed.

At the Lima TI computer show, to be held on May 26th, our club has reserved a table so that we can sell some disks, hoping to make money. Bill Tiep was talking about going down on Friday night for the copy session to be held for club officers from

6:00 to 9:00 p.m. Since last year's meeting Lima has added about 176 different disks, so there will be a lot of new material available this year. Several of our members are planning to go.

Bill Tiep has 2 disks that are tutorials on Extended Basic, which will be available at the next meeting. Peggy Strobell will demo a disk with 2 games by Kean next month.

The 50/50 winner was Earl Hoffsis, \$12.00. Earl declined the money, giving the entire amount to the club, but did take a silver mug instead.

Other winners were: 1) Dan Block - disk holder; 2) Margaret Dixon - club disk; 3) Peggy Strobell - mug; and 4) Brian Block - bar of soap shaped liked a computer.

The meeting adjourned about 1:30 p.m. Peggy Strobell demonstrated two games: Burger Time; Gromo and Black Box.

Schafstall, Secretary

Marilyn

ROGER FEINAUER

This is the last news letter before summer. With it I hope to convey some of the new and exciting software and hardware now available to the 99/4A and Geneve.

First let me say that I enjoyed the trip to Lima Ohio, last week. While there I had a long talk with Beery Miller the author of 9640 NEWS. Some of the items we discussed were the so call printer drivers that came with Mdos 114, which I found out didn't work. The author hadn't finished them but they were released anyway. Beery showed me a pre-beta 160 column word processor, and I bought his multi-tasking system which by the way had three Advance Basic programs running on the screen at the same time. Each in its own window on the screen. By the way this software has a new twist, when you buy the software Beery places your name in the program. So when you boot it up your name appears. If you take your name out of the program it won't work. Whos going to be dumb enough to give out software to other people that tells the world who it belongs to. Also I bought all of Vol 1 of 9640 NEWS which is packed with Mdos software. Right know this artical is being written with a word processor on one of his disks. The program has most of the commands of TI-WRITER plus has an assembler built in also, an the buffer capability of GDE, plus Macros. At this time most of the softwre I got I haven't had time to look at yet so will let know latter on Mdos.

GPL side I got Pix Pro 99, Page Pro 99 ver 1.6, two disks of page pro- fonts. Also a program call Rock Runner. Were should I start well lets start with Pix Pro 99. Pix programs are really a set of 3 EAS files that are broke down as follows PIX which is the freeware program to convert Rle, Ti-artist, Graphx, Pix, and Pix128 files and also to print these picture to printer of your choice. Pix Pro is similar except it allows for windowing of pictures with the arrow keys and a clipping function to save portions of any picture to ether Page Pro 99 pictures or Ti-Artist instances. Mcpix the last of these programs this one will allow you to convert Macpaint pictures to ether PIX or PAGE PRO 99 and also has print picture function.

Page Pro 99 ver 1.6 the main improvements are the the speed up of the columniser program, which has been speed up 2 to 3 times, and the main program which is one set of 5 EAS programs that run on both the TI and Geneve in GPL mode. The program now supports clipping part or all of a page to a picture file which allows the need of only one file for a page instead of a lot of little files. An example would be the front page of this news letter which is a 189 sector Page Pro picture file. Also it allows you to do directors of floppy and hard drives and also sub-directories of the hard drive. the options are simple type the drive and number ending with a period then preses control C, at this point you are prompted for space for next file or enter. This is a single line directory, to see more just press the space bar. If you find the file you need press enter, this will lock this file for loading. Press enter one more time and the picture will load starting at were you had the curser. One last file will allow the use of a 24 pin printer to print Page Pro pictures.

Lastly Rock Runner is a very new game for the 99/4A sorry can't see to get it to run on the Geneve it has screens A thru O. Requires the E/A cart to run as the program is so large uses a lot of internal rom routines to run so you need this cart to run it I even got it to run from a TI with a hard drive out of the hard drive. But at times requires



me to turn off the system to run something else because as I said before the program uses so much memory . This program is a new level of power programming which for 14.95 is an excellent buy. This one may be at the club meeting. See you at this months meeting. Roger Feinauer 116 S. Mckenzie St. Adrian, MI 49221.



EDWARD S. MACHONIS
82-23 261st STREET
FLORAL PARK, NY 11004

May 6, 1990

Dear Roger:

Contrary to what seems to be a fixation in the TI Community, I, and not Charles Ball, am the author of Type-writer. The article was part of a paper I presented at TIMARC (TI Metropolitan Area Regional Conference) in April 1986, copy enclosed. It was subsequently reprinted in the QB Monitor for April 1987. A copy of this issue was sent to PUNN, NW Ohio 99ers, (Bill Sager, Maumee OH), CONNI and Bluegrass Computer Soc. as a trial exchange.

Only NW Ohio and Bluegrass accepted the exchange. (CONNI joined years later.) Evidently PUNN, while declining the exchange saw fit to reprint the article. In February 89 both CONNI and Bluegrass reprinted the article, Bluegrass without attribution and CONNI naming Charles Ball as author. I wrote both requesting correction. Jean Hall graciously responded and followed up with a correction a couple of months later.

Again in Feb, it appears in New Horizons. (Beware the Ides of February?) I suspect your source for the article was CONNI and you missed the subsequent correction. In an effort at damage control, I would appreciate a correction in New Horizons. You can see how these things spread.

If you recall my article, "In Praise of Editors", QB Monitor, May 89, I am getting used to it. No apologies are necessary. I admire the job you are doing and no criticism is intended. (But if next Feb. my Editor reprints Type-writer by Charles Ball, he gets shot, I guarantee it!)

VIVA AA

*Dear Ed
Am very sorry for the mistake in the Feb. Newsletter hope this makes up for it
As viva the TI Editor Feinauer
Geneve*



STYLE A LINE

A TINYGRAM

by Ed Machonis

TINYGRAM: A short program which can be typed in its entirety on one screen without any program lines scrolling off the screen. (REM statements can scroll off.) Popularized, I believe, by Mike Stanfill of the Dallas TI Home Computer Group.

First of all let me make clear that this is not a novelty program. It is a work horse, provided you have the work for it. What kind of work? Do you ever have to print just a line or two, such as a page header, an article or picture title, a title for a data base printout, a credit line for a reprinted newsletter article, etc., etc. Further, would you like to print this in an Expanded Compressed Italicized Double Strike Underlined type style? Yes all the same time! If so, this program is for you.

What no printer? I will try to have something for you next month. (A TINYGRAM - NOT a printer!)

Many of you are familiar with my 10 Line basic programs, PRINTSTYLE and PRINTALINE. (Both TINYGRAMS, written before I knew the name existed.) I often use both of them in titling data base printouts or copy for the Newsletter but it got to be a pain to change between the two every time I wanted to change a type style. Finally the light dawned! Why not marry the two?

STYLE A LINE is the result of that marriage. One major revision was to change an INPUT statement in PRINTALINE to a LINPUT. No more need to enclose in quotes any text lines containing commas or leading spaces

Using LINPUT required that the program run in extended basic. After some streamlining by deletion of unneeded features from PRINTALINE and the consolidation of statements into multi-statement lines, we wound up with 9 Lines of code. (After merging TWO TEN Line programs. The power of extended basic!)

Don't let its brevity fool you. You can select any of the 128 type styles available on the Epson RX-80 and many compatibles. With line spacing and margin variations, over 2000 different selections can be had. (Half line spacing and condensed superscript will let you tack on several lines of comment onto a photocopied article.)

Although there are better ways of doing it, you can even produce a right margin justified letter. (THIS is novelty!) Using Emphasized Pica, set

Left Margin at 13, and enter text. Two screen lines will print text 54 characters wide (LINPUT uses two character spaces.) Justify text by inserting spaces between words so that second line ends at screen edge. But it will NEVER replace TI-Writer!

Using the program is very easy. When RUN, a menu is displayed for programming the printer. It is always best to select "1" to clear the printer. If your printer doesn't support a master reset code, turn it off then on to clear it. Combine styles by successive selections. Select Option 10 to input text.

If you wish to change the type style, or do repeated printings of the same text, typing "ZZZ" or "zzz" will return you to the menu. Option 9 will do repeat printing of the same text and styles can be changed as required. To input new text, select Option 10 again. When in text mode, pressing ENTER with no text input will print a blank line.

Watch those commas in Line 10. The next to last data item is a lower case "L", not the figure 1.

BRAIN TEASER: Where is the data to set the left margin at column 13?

```

1 : *** STYLE A LINE ***
  a TINYGRAM by Ed Machonis
  QB-99ers, Bayside, NY

2 DIM P$(15):: FOR I=1 TO 15
  :: READ P$(I):: NEXT I

3 OPEN #1:"PIO.LF",VARIABLE
  132 :: L$=CHR$(10)

4 CALL CLEAR :: PRINT "1 PIC
A/RESET","9 PRINT TEXT","2
ELITE","10 INPUT TEXT","3 EX
PANDED","11 SUPERSCRIP","4
COMPRESSED","12 SUBSCRIP"

5 INPUT "5 EMPHASIZED 13 L/
2 LINE SP6 ITALIC 14 L
MARGIN 137 D'BLE STRIK 15 R
MARGIN 678 UNDERLINE ?":I

6 P$(9)=" "&TEX$ :: PRINT #1
:CHR$(27)&P$(I):: IF I=4 THE
N PRINT #1:CHR$(27)&CHR$(15)

7 IF I<>10 THEN 4

8 PRINT "INPUT TEXT OR 'ZZZ
' FOR MENU" :: LINPUT TRY$

9 IF TRY$="ZZZ" OR TRY$="zzz
" THEN 4 ELSE TEX$=TRY$&L$ :
: PRINT #1:TEX$ :: GOTO 8

10 DATA @,M,W1,,E,4,G,-1,,,S
0,S1,1,1,QC
    
```

INSTANCE CONVERTER

by WESLEY R. RICHARDSON
NORTHCOAST 99ERS, CLEVELAND, OH, JUNE, 1989

INSTANCE-X is an Extended BASIC program which converts TI-ARTIST instances to a file suitable for TI-BASE version 2.01 to use as a database. Martin Smoley demonstrated the method of using TI-BASE for printing graphics in his article in the May, 1989 issue of the CLEVELAND AREA TI-99/4A USER GROUPS NEWSLETTER. He also had the idea of converting standard graphic data from other programs into TI-BASE. With the help of Deanna Sheridan, who suggested ARTIST instances as an extensive source for graphics, TI-BASE can now import and print graphics.

FILENAME	TYPE	DESCRIPTION
INSTANCE-X	PROGRAM	Extended BASIC program
INSTANCE/S	DIS/VAR 80	Assembly source code
INSTANCE/O	DIS/FIX 80	Assembly object code
INSTANCE/C	DIS/VAR 80	TI-BASE command file
TI-WORLD_I	DIS/VAR 80	TI-ARTIST instance
TI-WORLD_H	DIS/FIX 255	Hex code instance data
TI-WORLD/D	INT/FIX 254	TI-BASE instance data
TI-WORLD/S	INT/FIX 255	TI-BASE structure file

The source code listing INSTANCE/S is for the assembly routines which are called from the Extended BASIC program. To assemble this program, type in the INSTANCE/S file and save it. Then assemble the file giving INSTANCE/S as the source code file and INSTANCE/O as the object code file. Use only the R option because BASIC cannot access object files which use the C compress option.

When running the INSTANCE-X program, it will look for INSTANCE/D on drive 1. After loading, select option 1) to print an instance directly to the printer. Give the input file such as DSK1.TI-WORLD_I and the printer name PIO.CR.LF or the equivalent. Selecting option 2) will convert an instance to hex code for use by TI-BASE. Give an input file name like DSK1.TI-WORLD_I and an output file like DSK1.TI-WORLD_H. Option 3) is for a brief information summary, and option 4) is to end.

Load TI-BASE 2.01 and enter the date in the form MM/DD/YY, for example-05/26/89. For the following instructions, the notation (ENTER) will indicate to press the Enter key. Once in TI-BASE, type SET DATDISK DSK2. (ENTER). If you are using only one drive, the files listed above will need to be on

your TI-BASE disk, and substitute DSK1 for DSK2 in these instructions. With your hex instance file, TI-WORLD_H in drive 2, type CONVERT TI-WORLD_H TI-WORLD_GO (ENTER).

Type HEXCODE (ENTER) under "descriptor", X (ENTER) under "type" and 254 (FCTN 8) under "width". The disk drive should run for a while, and then you will get the "." prompt. Type CATALOG DSK2. (ENTER) to confirm that files TI-WORLD/D and TI-WORLD/S were created. Type USE TI-WORLD (ENTER). Then type RECOVER (ENTER). Finally, type CLOSE (ENTER). If you have made it this far, you now have the instance in a database form which can be manipulated.

To print the instance, we must create a command file to tell TI-BASE what to print. Type MODIFY COMMAND INSTANCE (ENTER). When the screen clears, type in the file shown as INSTANCE/C in this article. Press (FCTN 8) when you are done to save the file and return to the command mode. With your printer turned on, type DO INSTANCE (ENTER) and the instance will print on your printer.

Instances up to 14 character positions wide, and any length up to 32 character rows long may be printed with this method. There is one problem, however. If the instance which you are printing has a long row of blank spaces, TI-BASE may not send these to the printer, and the printer will seem to print garbage, beep, eject paper, and so forth. If this happens with the particular instance you are printing, then you must add a few pixel positions (dots) which are printed to act as place holders.

NOTE: DO THE FOLLOWING STEPS ONLY IF YOU HAD PROBLEMS AT THE PRINTING FROM TI-BASE PROCEDURE.

If that is the problem for your instance, then do the following, type USE TI-WORLD (ENTER). Type MODIFY STRUCTURE (ENTER). With the cursor on HEXCODE, press (ENTER) and change the "type" from X to C, then press (FCTN 8). Type SET HEADING ON (ENTER). Type SET RECNUM ON (ENTER). Then type PRINT ALL (ENTER). When the printer stops, type MODIFY STRUCTURE (ENTER). And press (ENTER) to move the cursor from the word HEXCODE to the "type" C. Press X and (FCTN 8).

If you examine the printout of the hex codes you will find a 20 followed by 00's near the end of each record. The 00's after the 20 are ignored, however long sequences of 00's prior or to the left of that 20 can be removed by TI-BASE when it is sending to the printer. Type EDIT (ENTER) and then use (FCTN 5) to go to higher record number or (FCTN 6) to go to lower record numbers. When you

...INSTANCE 2

have located the record with a long string of 00's to the left of the final 20, use the (FCTN 5) or (FCTN 0) to put the cursor on a 0 in the middle of a string of 00's. Change a 0 to a 1 and then press (FCTN 8) to keep the change and move the the next record. Do this for each record with a long string of 00's. After each record is changed, press (FCTN 8) and when you are done with all of the records, press (FCTN 9) to return to command mode. Type CLOSE (ENTER) to close the database.

Type DO INSTANCE (ENTER) to print the instance to your printer. when you are finished with TI-BASE, type QUIT (ENTER) to return to the TI title screen.

The more that I have used TI-BASE, the more impressed that I am with the power and flexibility of this program.

If you wish to get a copy of the files listed previously, send either a disk (SSSD or DSSD) with some of your favorite programs or copies of your club's last three newsletters to me at 27440 Pergl Road, Solon, OH 44139. I cannot send you any portions of TI-BASE.

-----890526WR-----

Filename: INSTANCE/C

```
* INSTANCE/C
* TI-BASE INSTANCE PRINTER
* WESLEY R. RICHARDSON MAY, 1989
* NORTHCOAST 99ERS, CLEVELAND, OH
* CONVERT FILE USING FORMAT:
* FIELD DESCRIPTOR TYPE WIDTH
* 1 HEXCODE X 254
SET TALK OFF
SET RECNUM OFF
SET HEADING OFF
USE TI-WORLD
WHILE .NOT. (EOF)
PRINT HEXCODE (LF) (CR)
MOVE
ENDWHILE
PRINT (LF) (LF) (CR)
CLOSE
RETURN
```

-----Filename: INSTANCE-X

```
100 REM INSTANCE-X V 1.0 TI-ARTIST INST
. ANCES CONVERTER
110 REM WESLEY R. RICHARDSON MAY, 1989
120 REM TI-99/4A EXTENDED BASIC
130 REM NORTHCOAST 99ERS, CLEVELAND, OH
140 DIM A(8),AS(8),CS(32,32)
150 GOTO 170 :: CALL CLEAR :: CALL INIT
```

```
:: CALL KEY :: CALL LINK :: CALL L
OAD :: CALL SCREEN :: CALL SOUND
160 BS :: DS :: ES :: I :: J :: K :: PS
:: W :: WS :: X :: XS :: Y :: YS
170 !@P-
180 DS="DSK1.INPUT_I" :: PS="PID.CR.LF"
190 CALL CLEAR :: CALL INIT
200 DISPLAY AT(6,4):"LOADING DSK1.INSTA
NCE/D" :: CALL LOAD("DSK1.INSTANCE/
O")
210 REM MAIN MENU
220 ON ERROR 800
230 CALL CLEAR :: DISPLAY AT(4,4):"INST
ANCE-X CONVERTER FOR" :: DISPLAY AT
(6,4):"TI-ARTIST INSTANCES TO"
240 DISPLAY AT(8,4):"PRINTER OR HEX FOR
MAT" :: DISPLAY AT(10,4):"by WESLEY
R. RICHARDSON"
250 DISPLAY AT(14,6):"1=PRINT INSTANCE"
:: DISPLAY AT(16,6):"2=CONVERT TO
HEX CODE"
260 DISPLAY AT(18,6):"3=INSTRUCTIONS" :
: DISPLAY AT(20,6):"4=END"
270 CALL KEY(0,K,J):: IF J=0 THEN 270 :
: K=K-40 :: IF (K<1)+(K>4)THEN 270
280 ON K GOTO 650,420,290,040
290 REM INSTRUCTIONS
300 CALL CLEAR :: DISPLAY AT(2,3):"INST
ANCE PRINTING IS TO" :: DISPLAY AT(
4,3):"EPSON TYPE PRINTERS."
310 DISPLAY AT(6,3):"HEX CODE OUTPUT IS
16" :: DISPLAY AT(8,3):"CHARACTERS
PER RECORD IN"
320 DISPLAY AT(10,3):"THE FORM 91FDC320
78605AE4" :: DISPLAY AT(12,3):"TYPE
OF STRINGS."
330 DISPLAY AT(16,3):"FURTHER INSTRUCTI
ONS ARE" :: DISPLAY AT(18,3):"IN FI
LE INSTANCE-D."
340 DISPLAY AT(22,7):"PRESS ANY KEY"
350 CALL KEY(0,K,J):: IF J=0 THEN 350 E
LSE 210
360 REM INPUT FILE
370 CALL CLEAR :: DISPLAY AT(14,6):"DSK
8 TO EXIT"
380 DISPLAY AT(6,4):"INSTANCE INPUT FIL
E?" :: DISPLAY AT(8,6):DS :: ACCEPT
AT(8,6)SIZE(-15):WS
390 IF SEOS(WS,4,1)="0" THEN 210 ELSE 0
S=WS
400 ES=SEOS(DS,1,LEN(DS)-1)6"MH"
410 RETURN
420 REM CONVERT TO HEX
430 GOSUB 300
440 DISPLAY AT(10,4):"OUTPUT FILE?" ::
DISPLAY AT(12,6):ES :: ACCEPT AT(12
,6)SIZE(-15):WS
```




```

...INSTANCE 3
450 IF EGCS(WS,4,1)="" THEN 210 ELSE I
    F WS=05 THEN 440 ELSE ES=WS
460 GOSUB 610
470 IF X>14 THEN CLOSE #1 :: DISPLAY AT
    (16,4):"INSTANCE IS MORE THAN" :: D
    ISPLAY AT(18,4):"14 CHARACTERS WIDE
    " :: GOTO 340
480 FOR J=1 TO 8 :: A(J)=0 :: NEXT J
490 A(7)=X :: CALL LINK("SIZE",A(),BS):
    : CS(0,0)=BS
500 FOR J=1 TO Y
510 DISPLAY AT(16,5):X;Y;J
520 FOR I=1 TO X :: GOSUB 630
530 CALL LINK("CODE",A(),BS)! CONVERT 8
    NUMBERS IN A() TO HEX STRING IN BS
540 CS(I,J)=BS :: NEXT I :: NEXT J :: C
    LOSE #1
550 DISPLAY AT(14,4):"WRITING TO DISK"
    :: OPEN #2:ES,DISPLAY ,FIXED 255
560 FOR J=1 TO Y :: DISPLAY AT(16,5):X;
    Y;J
570 BS="1B41081B4B"6SEGS(CS(0,0),13,4):
    : FOR I=1 TO X :: BS=BSGCS(I,J):: N
    EXT I
580 W=LEN(BS):: BS=BS$"20"CRPTS("0",252
    -W):: PRINT #2:BS
590 NEXT J :: CLOSE #2 :: GOTO 210
600 REM SUBROUTINES
610 DISPLAY AT(14,4):"READING DISK" ::
    OPEN #1:05,INPUT
620 INPUT #1:XS,YS P: X=VAL(XS):: Y=VAL
    (YS):: RETURN
630 INPUT #1:AS(1),AS(2),AS(3),AS(4),AS
    (5),AS(6),AS(7),AS(8)
640 FOR K=1 TO 8 :: A(K)=VAL(AS(K)):: N
    EXT K :: RETURN
650 REM INSTANCE PRINT
660 GOSUB 360 :: DISPLAY AT(14,6):"XXX
    TO EXIT"
670 DISPLAY AT(10,4):"PRINTER NAME?" ::
    DISPLAY AT(12,6):"PIO.CR.LF"
680 ACCEPT AT(12,6)SIZE(-28):WS :: IF S
    EGS(WS,1,3)="XXX" THEN 210 ELSE PS=
    WS
690 GOSUB 610
700 OPEN #2:PS :: PRINT #2:05;CHRS(10);
    CHRS(10);CHRS(27);CHRS(65);CHRS(8)
710 FOR J=1 TO Y
720 DISPLAY AT(16,5):X;Y;J
730 PRINT #2:CHRS(27);CHRS(75);CHRS(8*X
    );CHRS(8)
740 FOR I=1 TO X :: GOSUB 630
750 CALL LINK("NUMB",A())
760 FOR K=1 TO 8 :: PRINT #2:CHRS(A(K))
    :: NEXT K
770 NEXT I :: PRINT #2:CHRS(13);CHRS(10
    ):: NEXT J
    
```

```

780 PRINT #2:CHRS(27);CHRS(65);CHRS(12)
    ;CHRS(10);CHRS(10);CHRS(10)
790 CLOSE #1 :: CLOSE #2 :: GOTO 210
800 REM ERROR
810 CALL SCREEN(9):: FOR I=1 TO 200 ::
    NEXT I :: CALL SOUND(500,110,0):: C
    ALL SCREEN(8)
820 RETURN 210
830 !@P+
840 REM END
850 PRINT "STOP"
860 END
    
```

filename: INSTANCE/S

```

*****
*
* INSTANCE-X EXTENDED BASIC FILE *
* INSTANCE/S ASSEMBLY SOURCE FILE *
* INSTANCE/O ASSEMBLY OBJECT FILE *
*
* WESLEY R. RICHARDSON *
* MAY, 1989 *
* NORTHCOAST 99ER'S - CLEVELAND, OH *
*
*****
    
```

```

DEF CODE,NUMB,SIZE
*
NUMASG EQU >2008 WRITE NUMBER
NUMREF EQU >200C NUMBER GET
STRASG EQU >2010 STRING ASSIGN
FAC EQU >834A FAC ADDRESS
STATUS EQU >837C STATUS REGISTER
GPLWS EQU >83E0 GPM WORKSPACE
*
MYREG BSS 32 MY REGISTERS
BUF1 BSS 18 BUFFER 1
BUF2 BSS 18 BUFFER 2
SAVE DATA >8000 RETURN ADDRESS
*
    
```

```

* CODE - A() CONVERTS TO BS
*
* HEX CODE IN PRINTER FORMAT
*
* CALL LINK("CODE",A(),BS)
*
    
```

```

CODE MOV R11,@SAVE RETURN ADDRESS
LWPI MYREG SET UP REGISTERS
BL @GETA A() -> BUF1
BL @SCRPT BUF1-> CVT-> BUF2
BL @HEXSTR BUF2 -> BS
B @OONE BACK TO BASIC
*
    
```

```

* SIZE - A() CONVERTS TO BS HEX CODE
*
* HEX CODE IN NUMBER FORMAT
*
* CALL LINK("SIZE",A(),BS)
*
    
```

```

SIZE MOV R11,@SAVE RETURN ADDRESS
    
```

...INSTANCE 4

```

        LWPI MYREG      SET UP REGISTERS
        BL  @GETA      A() -> BUF1
        BL  @TOBUF2    BUF1 -> BUF2
        BL  @HEXSTR    BUF2 -> BS
        B   @DONE      BACK TO BASIC
    
```

```

*
* NUMB - A() CONVERTS TO A() NUMBER
*       NUMERIC IN PRINTER FORMAT
*       CALL LINK("NUMB",A())
*
    
```

```

NUMB   MOV  R11,@SAVE  RETURN ADDRESS
        LWPI MYREG      SET UP REGISTERS
        BL  @GETA      A() -> BUF1
        BL  @SCRPT     BUF1-> CVT-> BUF2
        BL  @HEXNUM    BUF2 -> A()
        B   @DONE      BACK TO BASIC
    
```

```

*
* SUBROUTINES *
*
*****
    
```

```

GETA   CLR  R0          LOOP COUNTER A()
        LI  R1,1        VARIABLE NUMB A()
        LI  R2,>1000    LENGTH OF BS = 16
        LI  R5,BUF1    BUFFER FOR BS
        MOVB @R5,R5+   STORE LENGTH OF BS
LOOP1  INC  R0          POINT TO ELEMENT
        BLWP @NUMREF   GET NUMBER
        LI  R2,>4041    OFFSET FOR 0, 100
        CB  @FAC,R2    IS IT < 100?
        JEQ V99        YES JUMP TO 00
        LI  R2,100     VALUE 100 TO 9999
        CLR  R3          SET UP CONVERSION
        CLR  R4          SET UP CONVERSION
        MOVB @FAC+1,R3  HIGH ORDER VALUE
        SWPB R3          RIGHT SIDE OF R3
        MPY  R2,R3      MULT BY 100, -> R4
        CLR  R3          SET UP LOW ORDER
        MOVB @FAC+2,R3  LOW ORDER VALUE
        SWPB R3          RIGHT SIDE OF R3
        A   R3,R4       R4 IS HEX 00 - FF
        SWPB R4          PUT IN LEFT SIDE
        JMP  STD1       JUMP STORE IN BUF1
V99    CLR  R4          SET UP FOR 0 TO 99
        MOVB @FAC+1,R4  GET VALUE
STD1   MOVB R4,@R5+    PUT IN BUF1
        CI  R0,0        FINISHED WITH 0?
        JNE LOOP1     NO, DO AGAIN
        RT              YES, RETURN
    
```

```

SCRPT  LI  R1,>0000    POSITION VALUE
        LI  R2,BUF1+9  END POSITION
        LI  R6,BUF2+1  POINTER FOR BUF2
    
```

```

        LI  R9,>7FFF    BIT MASK
SLOOP2 LI  R0,>0000    BIT VALUE
        LI  R5,BUF1+1  POINTER FOR BUF1
        CLR  R7          OUTPUT BYTE
SLOOP1 CLR  R3          INPUT BYTE
        CLR  R4          INPUT BYTE
        MOVB @R5+,R3    GET BYTE
        SZCB R9,R3      GET ON BIT
        SWPB R3          PUT IN LOW BYTE
        MPY  R0,R3      ADJ FOR BIT
        DIV  R1,R3      ADJ FOR POSITION
        SWPB R3          MOVE TO HIGH BYTE
        AB  R3,R7       ADD TO OUTPUT
        SRC  R0,1       POINTER TO RIGHT
        C   R5,R2       LOOP DONE?
        JNE SLOOP1     NO, DO AGAIN
        MOVB R7,@R6+   SAVE BYTE
        SRC  R9,1       ADJUST MASK
        SRC  R1,1       ADJUST POINTER
        CI  R6,BUF2+9  LOOP DONE?
        JNE SLOOP2     NO, DO AGAIN
        RT              RETURN
    
```

```

HEXSTR CLR  R0          LOOP COUNTER
        LI  R5,BUF1+1  POINTER FOR BUF1
        LI  R6,BUF2+1  POINTER FOR BUF2
LOOP2  INC  R0          INCREMENT COUNTER
        CLR  R4          SET UP REGISTER
        MOVB @R6+,R4    GET VALUE
        SWPB R4          TO LOW POSITION
        MOV  R4,R3      COPY VALUE
        ANDI R3,>00FF   GET LEFT VALUE
        SLA  R3,4       IN HIGH ORDER BYTE
        CI  R3,>0A00    VALUE < 10?
        JL  CONLL       JUMP IF LOW
        AI  R3,>0700    ADJUST FOR ABCDEF
        AI  R3,>3000    ADJUST FOR STRING
        MOVB R3,@R5+    STORE LEFT VALUE
        ANDI R4,>000F   RIGHT VALUE IN BS
        CI  R4,>000A    VALUE < 10?
        JL  CONRL       JUMP IF LOW
        AI  R4,>0007    ADJUST FOR ABCDEF
        AI  R4,>0030    ADJ. STRING VALUE
        SWPB R4          MOVE TO HIGH BYTE
        MOVB R4,@R5+    RIGHT VALUE IN BS
        CI  R0,0        FINISHED A() ?
        JNE LOOP2     NO, DO AGAIN
        CLR  R0          BS IS NOT AN ARRAY
        LI  R1,2        BS IS VARIABLE 2
        LI  R2,BUF1    LOCATION OF BS
        BLWP @STRAS0    WRITE BS STRING
        RT              RETURN
    
```

...INSTANCE 5

```

TOBUF2 CLR R0      LOOP COUNTER
        LI R5,BUF1+1 POINTER FOR BUF1
        LI R6,BUF2+1 POINTER FOR BUF2
LOOP3  INC R0      INCREMENT COUNTER
        MOVB *R5+,*R6+ MOVE VALUE
        CI R0,B     FINISHED WITH A()?
        JNE LOOP3   NO, DO AGAIN
        LI R6,BUF2+7 SIZE VALUE
        CLR R3      CLEAR REGISTER
        MOVB *R6,R3  GET SIZE
        SLA R3,3    MULTIPLY BY 8
        MOVB R3,*R6 STORE SIZE
        RT

*
*****
*
HEXNUM CLR R0      LOOP COUNTER A()
        LI R1,1     VARIABLE NUMB A()
        LI R5,BUF2+1 BUFFER FOR B$
        CLR R6      ZERO
LOOP4  INC R0      POINT TO ELEMENT
        CLR R3      SET UP WORK AREA
        CLR R4      SET UP WORK AREA
        MOVB *R5+,R4 GET NUMBER
        CI R4,>5400 GREATER THAN 100?
        JL N99      NO, LESS THAN 100
        LI R2,>4100 VALUE 100 TO 9999
        MOVB R2,@FAC STORE X100
        LI R2,>006.1 DIVISOR = 100
        SWPB R4     IN LOW POSITION
        DIV R2,R3   R3=R4/R2
        SWPB R3     PUT IN HIGH BYTE
        MOVB R3,@FAC+1 STORE HIGH BYTE
        SWPB R4     PUT IN HIGH BYTE
        MOVB R4,@FAC+2 STORE LOW BYTE
        JMP ST02   GOTO WRITE
N99    LI R2,>4000 SET FOR 0 TO 99
        MOVB R2,@FAC STORE X1
        MOVB R4,@FAC+1 STORE NUMBER
        MOVB R6,@FAC+2 BALANCE IS ZERO
        MOVB R6,@FAC+3 BALANCE IS ZERO
        MOV R6,@FAC+4 BALANCE IS ZERO
        MOV R6,@FAC+6 BALANCE IS ZERO
        BLWP @NUMASG WRITE A() NUMBER
        CI R0,B     FINISHED WITH B?
        JNE LOOP4  NO, DO AGAIN
        RT         YES, RETURN

*
*****
*
DONE   LWPI GPLWS   GPL WORKSPACE
        CLR @STATUS CLEAR STATUS REG.
        MOV @SAVE,R11 RETURN POINT
        B *R11     TO EXTENDED BASIC
        END
    
```

filename: TI-WORLD_I

```

5,10
0,0,63,21,31,20,28,24
0,0,199,255,109,56,40,56
0,0,249,82,243,80,112,48
0,0,227,187,54,20,20,28
0,0,192,32,96,0,0,0
18,32,0,0,0,0,0,0
40,56,40,56,40,56,40,56
16,8,0,0,0,0,0,0
20,28,20,28,20,28,20,28
0,0,0,0,0,0,0,0
12,16,0,7,0,0,0,0
40,56,108,215,0,0,0,0
96,19,34,193,0,0,0,254
20,20,119,235,0,0,0,0
0,96,32,192,0,0,0,0
0,0,0,0,0,1,2,2
7,24,33,66,132,8,16,17
125,186,85,84,146,146,146,17
192,40,0,132,66,33,16,16
0,0,0,0,0,0,128,128
4,4,4,0,0,0,15,8
33,33,34,66,66,66,255,66
17,17,18,18,18,16,255,18
8,8,136,132,132,132,255,132
64,64,64,32,32,32,224,32
8,8,4,4,4,2,2,1
66,66,34,33,33,17,16,8
16,16,18,17,17,17,146,146
132,132,138,8,8,18,18,33
32,32,64,64,64,128,128,0
0,0,0,0,0,0,0,0
132,66,33,24,7,0,0,0
146,84,85,186,125,254,0,0
66,132,8,48,192,0,0,0
0,0,0,0,0,0,0,0
7,24,48,112,80,112,80,112
252,16,6,3,1,1,3,7
1,8,140,220,84,220,84,220
255,3,1,0,0,0,0,1
0,192,168,240,80,112,208,240
56,15,0,0,96,48,56,40
13,249,1,1,1,1,2,3
78,195,64,192,88,204,142,10
3,254,0,0,0,0,0,0
80,112,80,112,80,112,160,192
60,103,0,0,0,0,0,0
14,248,0,0,0,0,0,0
15,25,0,0,0,0,0,0
3,254,0,0,0,0,0,0
128,0,0,0,0,0,0,0
    
```



KIDS PAGE

This month we will use page pro to write this article and see what happens. This month's topic will be call sound or how can I make my computer make music. So here goes in order for our computer make music we need a device to organize certain commands given by us to our computer. On the 99/4A it is the 9919 music processor. Yes another little computer that works with or main processor but lets just call it the MP.

TI makes most of the process of the MP transparent to us which makes music and sounds that we need to make in our computer very easy.

The command we use in Basic or XB is the 'call sound' command. To this command we pass along a list of numbers, one other important item, we must follow a set of rules.

CALL SOUND(duration,frequency,volume)

duration=is an integer in the range from 1 to 4250 or -1 to -4250 and can be a numeric literal, numeric variable, or numeric expression. It is measured in milliseconds or 1000th of a second, so a value of 1000 here equals one second. We are allowed only one duration number per call sound command.

frequency=determines whether a sound is musical or a noise. Musical tone can be an integer from 110 to 44733 that corresponds to the desired frequency. For noise it is an integer from -1 to -8 depending on the type of noise.

volume=an integer from 0 to 30. Zero is the highest volume and 30 is the lowest or softest.

Now with this let's type in basic the next line as follows!

```
CALL SOUND(1000,110,5)
```

By changing the values in the above call sound command you can make new sounds of your very own.

As I wrote earlier you can only have one duration but, you can have up to 4 frequencies and 4 volumes. Three musical durations and volumes, and 1 noise duration and volume in one command of call sound.

```
CALL SOUND(3000,262,0,330,0,392,-3,0)
```

Try the above routine on the TI in Basic or XB.

Now with what we know lets use it to make a little program to make a clock bell.

```
100 REM CLOCK BELL
110 FOR L=1 TO 6
120 FOR L2=0 TO 30
STEP 3
130 CALL SOUND(-50,400,
L2,1200,L2,2400,L2)
140 NEXT L2
150 FOR D=1 TO 200 :
NEXT D
160 NEXT L
```

As you can see with a little practice you too can make almost any sound.