HOCUS

Home Computer
Users Spotlight

a monthly publication of the
Milwaukee Area 99/4 Users Group

MARCH—1989

MILWAUKEE AREA TI99 GROUP
4122 GLENWAY WAUWATOSA WI 53222

President	D.Walden	5292173
Vice-Pres	J.Schroeder	2644735
Coordinator	G.Hitz	53[illegible]
Treasurer	P.Norton	4[illegible]
Secretary	J.Stepanovich	7[illegible]67
Disk Librarian .	R.Ziolkowski	6[illegible]05
Tape Librarian .	D.Dann	5[illegible]33
9640 Librarian .	M.Viesselmann	2[illegible]578
Newsletter	G.Hitz	[illegible]
Publicity	D.Walden	[illegible]73
S.I.G.	Schroeder/Walden/Hitz	

Next Group Meeting - 2rd Saturday
April 8, 1989 - 12 noon t 4 PM
Wauwatosa S & L - 7500 West State

North Sub-Meeting - 1st Tuesday
April 4, 1989 - 7 PM til 1 PM
Security S & L - 5555 N Pt Washington

South Sub-Meeting - 3rd Tuesday
March 21, 1989 - 7 PM til 10 PM
Franklin State Bank - 7000 So 76th

Membership Dues $10 - Family $15

<<<<< HOCUS NEWSLETTER INDEX >>>>>

8903

SWAP MEET
SWAP MEET
SWAP MEET

Our Milwaukee Area 99 User Group will be holding the annual swap meet next month at our regular monthly meeting. That's April 8, the second Saturday in April, from 1:00 til 4:00 PM at the Wauwatosa Saving & Loan. Tables will be available free to all Group members and any and all types of electronic or computer related items: hardware software, firmware but no underware. All are invited so bring your interested friends or even enemies to take advantage of the myriad of bargains rummage garbage and whatnots. Everyone's money will be accepted.

In Memoriam

Phillip Weiss

Active in Group for over 4 years

Passed away last month

Below is the listing for a FAIRWARE Basic program strictly for cassette users. By using this program one can load 10 programs on a 60 minute tape and later again using it, locate the exact program position with the tape running at fast forward. The program also will create a catalog of the tape to save at the start of the tape before the 10 programs. This catalog can be output to a printer if one is available.

I took the liberty of editting the program and reduced it to about half the original size to allow it to be loaded into minimem for constant use. A counter is not necessary on the tape recorder. If it has a STOP control, the program ahould automatically stop at the correct position. To use it with out a STOP control one must watch the screen. It turns yellow as a warning as the position approaches and red at the STOP position. Enjoy!

```
100 REM    CS1-FINDEX
        TI-BASIC  FAIRWARE
        Joseph E. Bartle
        16 S&E Trailor Ct
        Parish, NY 13131
110 CALL CLEAR
120 _$="__________"
130 DIM P$(19)
140 PRINT : :"  ############
############   #
             #   #  FINDEX  O
PTION LIST   #   #
             #   ############
############"
150 PRINT "  #
         #   #   0. LOAD   C
ATALOG   #   #   1. SEARCH T
APE      #   #
         #   ################
##########"
160 PRINT "  #
         #   #   2. CREATE C
ATALOG   #   #   3. CHANGE C
ATALOG   #   #   4. PRINT  C
ATALOG   #   #   5. SAVE   C
ATALOG   #"
170 PRINT "  #   6. QUIT
         #   #
         #   #      Enter  O
 - 6     #   #
         #   ################
##########": :
180 CALL KEY(0,O,S)
190 IF (O<48)+(O>54)THEN 180
200 ON O-47 GOTO 210,300,330
,400,380,720,1030
210 OPEN #3:"CS1",INTERNAL,I
NPUT ,FIXED 192
220 INPUT #3:G$
230 INPUT #3:H$
240 F$=G$&H$
250 CLOSE #3
260 FOR X=0 TO 19
270 P$(X)=SEG$(F$,X*12+1,12)
280 NEXT X
290 GOTO 140
300 CALL CLEAR
310 PRINT TAB(12);"Location.
..";
320 GOTO 400
330 CALL CLEAR
340 FOR X=0 TO 19
350 P$(X)=_$
360 NEXT X
370 GOTO 400
380 U=1
390 OPEN #U:"PIO"
400 PRINT #U:"  Programs
  Remarks": :
410 FOR X=0 TO 9
420 PRINT #U: ;STR$(X);" ";P
$(X);"  17);P$(X+10)
430 NEXT X
440 IF O=49 THEN 830
450 IF U=0 THEN 490
460 CLOSE #U
470 U=0
480 GOTO 140
490 PRINT
500 INPUT "Any Changes? Y/N
  ":X$
510 IF X$<>"Y" THEN 140
520 PRINT
530 INPUT "Program No. (N=No
ne) ":X$
540 IF X$="N" THEN 140
550 CALL CLEAR
560 PRINT : : :" Just Press
<ENTER> to Keep. 'N' to Quit
"
570 I=VAL(X$)
580 PRINT : : :" Program Nam
e ^Limit 12 CHR^"
590 PRINT :"Old/ReName #";I+
10$(I>9);P$(I): :
600 INPUT X$
610 IF X$="" THEN 660
620 IF X$="N" THEN 520
630 IF LEN(X$)<13 THEN 650
640 X$=SEG$(X$,1,12)
650 P$(I)=X$&SEG$(_$,1,12-LE
N(X$))
660 IF I<10 THEN 690
670 I=I-9
680 GOTO 580
690 I=I+10
700 PRINT : : :" Description
  ^Limit 12 CHR^"
710 GOTO 590
720 F$=""
730 FOR I=0 TO 19
740 F$=F$&P$(I)
750 NEXT I
760 O$=SEG$(F$,1,180)
770 H$=SEG$(F$,181,180)
780 OPEN #4:"CS1",INTERNAL,O
UTPUT,FIXED 192
790 PRINT #4:O$
800 PRINT #4:H$
810 CLOSE #4
820 GOTO 180
830 INPUT "Location Number 1
-10  ":L
840 IF (L<1)+(L>10)THEN 830
850 L=L-1
860 CALL SCREEN(2)
870 CALL CLEAR
880 PRINT "  Press Any Key T
o Continue": : :"At First Be
ep...": :"* REWIND CASSETTE
TAPE   CS1  THEN PRESS ENTER
": : : :"At Second Beep...":
 :"* PRESS FAST FORWARD
CS1"
890 PRINT "  THEN PRESS ENTE
R": : : :"At Third Beep...":
 :"* PRESS CASSETTE STOP
CS1  THEN PRESS ENTER": :_$;
_$
900 CALL SCREEN(15)
910 CALL KEY(0,K,S)
920 IF S=0 THEN 910
930 OPEN #2:"CS1",DISPLAY ,O
UTPUT,FIXED 192
940 FOR I=0 TO L
950 IF I<L THEN 970
960 CALL SCREEN(11)
970 PRINT #2:I
980 NEXT I
990 CALL SCREEN(7)
1000 CLOSE #2
1010 CALL CLEAR
1020 PRINT "Program ";P$(I-1
);" located": : : :"Enter 'O
LD... or 'SAVE CS1'": : : :"
   BUT DO NOT REWIND !!!": :
 : :
1030 PRINT "@1983. @1984, 91
985": : :
1040 STOP
```

A DIFFERENT APPROACH TO SPEECH

by

KEVIN COX

The program opposite does not look much like the normal 'speech program', as it consists mainly of CALL LOADS, but this method works much quicker than the usual CALL SAY method.

When the computer encounters a CALL SAY(" "), it stops execution of the program until it has completed the CALL SAY subprogram, while in the CALL LOAD(" ") method the computer continues on with the program, not waiting for the subprogram to be completed.

The phrases are listed in the Editor/Assembler manual on page 422. The 2 bytes following the phrase are noted and the digits reversed and 64 is added to each digit. After inserting the numbers it must finish with 64, and then 80 is needed at the end to tell the computer to speak that line.

The first program will do all that for you. All you have to do is insert the numbers as they appear in the manual. This program runs in either Extended BASIC or in BASIC with the Mini-Memory module.

(Thanks to Kevin Cox and the Hunter Valley newsletter Aug 1988)

```
10 REM ********************
20 REM *SPEECH CONVERSION*
30 REM *    NUMBERS       *
40 REM *  by kevin Cox    *
50 REM *  USING THE E/A   *
60 REM *      MANUAL      *
70 REM *  9th July 1988   *
80 REM ********************
90 CALL CLEAR
100 PRINT "INPUT 4 HEX NUMBERS"
110 INPUT "SEPARATE BY
    COMMAS -":A$,B$,C$,D$
120 IF A$="A" THEN A$="10"
130 IF A$="B" THEN A$="11"
140 IF A$="C" THEN A$="12"
150 IF A$="D" THEN A$="13"
160 IF A$="E" THEN A$="14"
170 IF A$="F" THEN A$="15"
180 I=VAL(A$)
190 I=I+64
200 IF B$="A" THEN B$="10"
210 IF B$="B" THEN B$="11"
220 IF B$="C" THEN B$="12"
230 IF B$="D" THEN B$="13"
240 IF B$="E" THEN B$="14"
250 IF B$="F" THEN B$="15"
260 H=VAL(B$)
270 H=H+64
280 IF C$="A" THEN C$="10"
290 IF C$="B" THEN C$="11"
300 IF C$="C" THEN C$="12"
310 IF C$="D" THEN C$="13"
320 IF C$="E" THEN C$="14"
330 IF C$="F" THEN C$="15"
340 J=VAL(C$)
350 J=J+64
360 IF D$="A" THEN D$="10"
370 IF D$="B" THEN D$="11"
380 IF D$="C" THEN D$="12"
390 IF D$="D" THEN D$="13"
400 IF D$="E" THEN D$="14"
410 IF D$="F" THEN D$="15"
420 K=VAL(D$)
430 K=K+64
440 PRINT K;J;H;I;64;80
450 OPEN #1:"PIO"
460 PRINT #1:K;J;H;I;64;80
470 CLOSE #1
475 PRINT
480 PRINT "ANOTHER SET OF
    NUMBERS (Y/N)"
490 CALL KEY(0,K,S)::
    IF S<1 THEN 490 ELSE
    IF K=89 THEN 100 ELSE END
```

HOUSEHOLD BUDGET MANAGEMENT

by Jim Seitz

(Ed Note: This is a reprint. This article appeared in the April 1985 issue of Spirit of 99 followed by SUCCESS, another article in March 1986 issue of Spirit of 99. Both relate to the HOUSEHOLD BUDGET MANAGEMENT program. Jim's address listed in the first article has been changed to his present address.)

I bought my computer during the Great Computer Sale of November 1983. One of the first programs I bought was "Household Budget Management". We put this program on line January 1984 and we are using it this year also. The program consists of 99 preselected categories of which 34 can be active at any one time; thus you can customize your budget to your needs. The categories are classed as either income or expenses. After choosing your categories, you can assign a budgeted amount to the category or enter a full amount. One of my compaints about the program is you cannot rename any of the categories; for instance we wanted to keep track of pet care expenses but there is not a pet care category. We had to use the "Dry Cleaning" category and remember it was really "Pet Care".

As with any budget the biggest problem is keeping track of your records so they can be entered. We had to develop several ideas to improve our record keeping which I would like to share with you. We keep a small box on our desk to put receipts in after we go shopping. It helps to label the items purchased on the receipt to insure they are entered into the proper category. We put paystubs near the box to be entered into the computer also. I do as much entering from the individual receipts as I can; you cannot keep an accurate budget if you only use the monthly statements or your charge cards. After entering the receipts and paystubs I look at the checkbook for other expenses that might have been mixed.

When I finish with the checkbook I draw a line under the last item entered into the computer so I know where I left off. I do the same for the notebooks we keep in the cars to record the car expenses. The last place I look for expenses is on the calendar we keep in the kitchen. We record baby sitting and other cash expenses that generally do not issue a receipt there. It also helps to keep a menu of which category you enter hard-to-define items. For example: are computer expenses "Household" or "Education"? I only enter our budget about 2 times a month and this whole process takes about 30 minutes.

After entering your data you can analyze your budget using several different options. I use monthly and the year-to-date options the most. There are also options to change your budgeted amount or to correct mistakes. You can add or substract categories as you choose, but remember to go back and update your entries! The monthly and year-to-date options also include graphs and projections that can be helpful also.

I have only two chief complaints against this program. The first is-it treats Savings as an expense, I wish the program operated under three main classes: Income, Expenses and Savings. My second complaint is the program does not include the option of a printout. If anyone knows how to get a printout please contact me: Jim Seitz at 2167 Keller Pl W., Grove City, OH 43123 or call (614)875-5532.

I think this is a good program and is worth the investment for the person who does not have access to the more expensive spreadsheet programs.

##

SUCCESS!

by Jim Seitz

In April 85 issue of "The Spirit of 99" I wrote an article entitled HOUSEHOLD BUDGET MANAGEMENT reviewing the module of the same name. In the article I mentioned one large drawback to the program was the lack of a printout. Last October I received a letter from Mr. Bob Lawson of Houston, TX, stating he had written a program, available as "freeware", to print out the HBM files. Would I be interested?....YOU BET! In late November I received my copy of HBMPRINT and used it right away.

You will need the following to use the program: 99/4A console, 32K memory, disk drive(s), RS232 and printer, Editor/Assembler, and your data disk. The program is easy to run; just turn on the hardware, insert the E/A module, put the program disk in the disk drive, select "Load and Run" from the E/A menu and load the program. After the program loads you are walked through a hardware checklist to identify the hardware being used. After identifying the name of your data file disk, place it in the disk drive; press any key; and the file will be read. After answering a few questions and setting up the printer the printout menu will appear. You can choose from the following printouts: 1) All Categories for One Month, 2) All Categories Year to Date, 3) All Categories Total Year, 4) One Category by Month, 5) All Categories by Month, 6) All Income by Month, 7) All Expense by Month. I use the first printout monthly and the rest of the printouts as needed. You can also customize your printout by using this program in conjunction with TI-Writer. I consider the program to be the missing link needed to complete my monthly budgeting. This program is available through the library, and let's support the author of this great program.

TIPS FROM THE TIGERCUB

#53

TIGERCUB SOFTWARE
156 Collingwood Ave.
Columbus, OH 43213

Over 120 original programs in Basic and Extended Basic, available on cassette or disk, NOW REDUCED TO JUST $1.00 EACH!, plus $1.50 per order for cassette or disk and PP&M. Minimum order of $10.00. Cassette programs will not be available after my present stock of blanks is exhausted. The Handy Dandy series, and Color Programming Tutor, are no longer available on cassette.
Descriptive catalogs, while they last, $1.00 which is deductable from your first order.

Tigercub Full Disk Collections, reduced to $5 postpaid. Each of these contains either 5 or 6 of my regular catalog programs, and the remaining disk space has been filled with some of the best public domain programs of the same category. I am NOT selling public domain programs - they are a free bonus!
TIGERCUB'S BEST, PROGRAMMING TUTOR, PROGRAMMER'S UTILITIES, BRAIN GAMES, BRAIN TEASERS, BRAIN BUSTERS!, MANEUVERING GAMES, ACTION GAMES, REFLEX AND CONCENTRATION, TWO-PLAYER GAMES, KID GAMES, MORE GAMES, WORD GAMES, ELEMENTARY MATH, MIDDLE/HIGH SCHOOL MATH, VOCABULARY AND READING, MUSICAL EDUCATION, KALEIDOSCOPES AND DISPLAYS

NUTS & BOLTS DISKS
These are full disks of 100 or more utility subprograms in MERGE format, which you can merge into your own programs and use, almost like having another hundred CALLs available in Extended Basic. Each is accompanied by printed documentation giving an example of the use of each.
NUTS & BOLTS (No. 1) has 100 subprograms, a tutorial on using them, and 5 pp. documentation. NUTS & BOLTS No. 2 has 108 subprograms, 10 pp of documentation. NUTS & BOLTS #3 has 140 subprograms and 11 pp. of documentation.
NOW JUST $15 EACH, POSTPAID.

TIPS FROM THE TIGERCUB
These are full disks which contain the programs and routines from the Tips from the Tigercub newsletters, in ready-to-run program format, plus text files of tips and instructions.
TIPS (Vol. 1) contains 50 original programs and files from Tips newsletters No. 1 through No. 14. TIPS VOL. 2 contains over 60 programs and files from Nos. 15 thru 24. TIPS VOL. 3 has another 62 from Nos. 25 through 32. TIPS VOL. 4 has 48 more from issues No. 33 through 41.
NOW JUST $10 EACH, POSTPAID.

```
**********************************
*          NOW READY             *
* TIPS FROM TIGERCUB VOL.5       *
* Another 49 programs and        *
* files from issues No. 42       *
* through 50. Also $10 ppd       *
**********************************
```

TIGERCUB CARE DISKS #1,#2,#3 and #4. Full disks of text files (printer required).
No. 1 contains the Tips newsletters #42 thru #45, etc. Nos. 2 and 3 have articles mostly on Extended Basic programming. No. 4 contains Tips newsletters Nos. 46-52. These were prepared for user group newsletter editors but are available to anyone else for $5 each postpaid.

Someone wanted a program to teach how to make change.

```
100 CALL CLEAR
110 DEF I$(X)="$"&SEG$(STR$(
X),1,POS(STR$(X),".",1)+2)
120 CALL COLOR(1,2,8,2,2,8,3
,2,8,4,2,8,5,2,8,6,2,8,7,2,8
,8,2,8,9,2,8,10,2,8,11,2,8,1
2,2,8)
130 CALL SCREEN(5):: D$=RPT$
(" ",112)!programmed by Jim
Peterson Apr. 1988 for the p
ublic domain
140 GOTO 180
150 CALL KEY :: CALL CLEAR :
: CALL SOUND
160 A,P,C,G,T,F,D,Q,D,N,S,TT
,X,B$,QQ,K,M$,J
170 !@P-
180 DISPLAY AT(2,8)ERASE ALL
:"MAKING CHANGE";;:"     by
Jim Peterson" :: RANDOMIZE :
: CALL KEY(3,K,S)
190 DISPLAY AT(6,1):"Do you
want to":::" (1) Input amoun
ts":" (2) Use random amounts
" :: ACCEPT AT(6,16)SIZE(1)V
ALIDATE("12")BEEP:A :: CALL
CLEAR
200 IF A=2 THEN 240 :: DISPL
AY AT(2,1):"Price of item? $
";;;;;; :: ACCEPT AT(2,17)V
ALIDATE(NUMERIC)BEEP:P :: IF
 P<0 THEN 200
210 DISPLAY AT(4,1):"Amount
offered by customer?":"$" ::
 ACCEPT AT(5,2)VALIDATE(NUME
RIC)BEEP:C
220 IF C<>INT(C)AND P=INT(P)
<>C-INT(C)THEN DISPLAY AT(23
,1):"Even dollars please!" :
: GOTO 210
230 IF C<P THEN DISPLAY AT(2
3,1):"Not enough!" :: GOTO 2
10 ELSE IF C=P THEN DISPLAY
AT(23,1):"No change needed!"
 :: GOTO 200 ELSE GOTO 260
240 RANDOMIZE :: P=20*RND+.0
1 :: P=INT(P*100)/100 :: DIS
PLAY AT(2,1):"Price of item
" :: DISPLAY AT(2,16):I$(P+.
001)
250 C=INT(4*RND+1)*5 :: IF C
<=P OR C=15 THEN 250 :: DISP
LAY AT(4,1):"Customer offers
 $" :: DISPLAY AT(4,19):STR$
(C)
260 G=C :: C=C-P :: T=INT(C/
10):: C=C-T*10 :: F=INT(C/5)
:: C=C-F*5 :: D=INT(C):: C=C
-D :: Q=INT(C/.25):: C=C-Q*.
25 :: D=INT(C/.1):: C=C-D*.1
 :: N=INT(C/.05)
270 C=C-N*.05 :: X=C*100 ::
TT=0
280 DISPLAY AT(7,1):"OK, sta
rt adding change from the pr
ice until you reach the cus
tomer's amount." :: TT=P+.00
1
290 IF X=0 THEN 330 :: GOSUB
 640 :: IF B$<>"P" THEN M$="
No, give pennies to reach
"&I$(TT+X*.01):: GOSUB 670
300 DISPLAY AT(11,1):"How ma
ny pennies?" :: ACCEPT AT(11
,19)VALIDATE(NUMERIC)BEEP:QQ
310 IF QQ=X THEN 320 ELSE IF
 TT+QQ*.01>G THEN GOSUB 680
:: GOTO 300 ELSE GOSUB 690 :
: GOTO 300
320 DISPLAY AT(7,1):"You hav
e reached "&I$(TT+X*.01):;:;
:;: :: TT=TT+X*.01
330 IF N=0 THEN 380 :: GOSUB
 640
340 IF B$<>"N" THEN M$="No,
if the cents portion is .10
or .20 or .35 or .45 or .60
or .70 or .85, add a nick
el" :: GOSUB 670
350 DISPLAY AT(11,1):"How ma
ny nickels?" :: ACCEPT AT(11
,19)VALIDATE(NUMERIC)BEEP:QQ
360 IF QQ=N THEN 370 ELSE IF
 TT+QQ*.05>G THEN GOSUB 680
:: GOTO 350 ELSE GOSUB 690 :
: GOTO 350
370 DISPLAY AT(7,1):"You hav
e reached "&I$(TT+N*.05):;:;
:;: :: TT=TT+N*.05
380 IF D=0 THEN 430 :: GOSUB
 640
390 IF B$<>"D" THEN M$="No,
add dimes to reach .25 or .
50 or .75 or .00" :: GOSUB 6
70
400 DISPLAY AT(11,1):"How ma
ny dimes?" :: ACCEPT AT(11,1
9)VALIDATE(NUMERIC)BEEP:QQ
410 IF QQ=D THEN 420 ELSE IF
 TT+QQ*.1>G THEN GOSUB 680 :
: GOTO 400 ELSE GOSUB 690 ::
```

```
 GOTO 400
420 DISPLAY AT(7,1):"You hav
e reached "&I$(TT+D$,1):;;;:
;: :: TT=TT+D$.1
430 IF Q=0 THEN 480 :: GOSUB
 640
440 IF B$<>"Q" THEN M$="No,
add quarters to reach an eve
n dollar." :: GOSUB 670
450 DISPLAY AT(11,1):"How ma
ny quarters?" :: ACCEPT AT(1
1,20)VALIDATE(NUMERIC)BEEP:Q
Q
460 IF QQ=Q THEN 470 ELSE IF
 TT+QQ*.25>G THEN GOSUB 680
:: GOTO 450 ELSE GOSUB 690 :
: GOTO 450
470 DISPLAY AT(7,1):"You hav
e reached "&I$(TT+Q*.25):;;;
;;: :: TT=TT+Q*.25
480 IF D=0 THEN 530 :: GOSUB
 640
490 IF B$<>"D" THEN M$="No,
add dollars to reach a  mult
iple of five dollars." :: GO
SUB 670
500 DISPLAY AT(11,1):"How ma
ny dollars?" :: ACCEPT AT(11
,19)VALIDATE(NUMERIC)BEEP:QQ
510 IF QQ=0 THEN 520 ELSE IF
 TT+QQ>G THEN GOSUB 680 :: G
OTO 500 ELSE GOSUB 690 :: GO
TO 500
520 DISPLAY AT(7,1):"You hav
e reached "&I$(TT+D):;;;;;;
:: TT=TT+D
530 IF F=0 THEN 580 :: GOSUB
 640
540 IF B$<>"F" THEN M$="No,
add a five dollar bill  to r
each a multiple of ten." ::
GOSUB 670
550 DISPLAY AT(11,1):"How ma
ny fives?" :: ACCEPT AT(11,1
7)VALIDATE(NUMERIC)BEEP:QQ
560 IF QQ=F THEN 570 ELSE IF
 TT+QQ*5>G THEN GOSUB 680 ::
 GOTO 550 ELSE GOSUB 690 ::
GOTO 550
570 DISPLAY AT(7,1):"You hav
e reached "&I$(TT+F*5):: TT=
TT+F*5
580 IF T=0 THEN 620 :: GOSUB
 640
590 IF B$<>"T" THEN M$="No,
add ten dollar bills to reac
h"&I$(6):: GOSUB 670
600 DISPLAY AT(11,1):"How ma
ny tens?" :: ACCEPT AT(11,16
)VALIDATE(NUMERIC)BEEP:QQ ::
 IF QQ=T THEN 620
610 IF QQ>T THEN GOSUB 680 :
: GOTO 600 ELSE DISPLAY AT(1
4,1)BEEP:"That's not enough!
" :: GOTO 600
620 DISPLAY AT(7,1)BEEP:"You
 gave the correct change!":"
":"":""
630 DISPLAY AT(14,1):"
 PRESS ANY KEY" :: DISPLAY A
T(14,1):"       press any ke
y" :: CALL KEY(3,K,S):: IF S
=0 THEN 630 ELSE 200
640 DISPLAY AT(11,1):"Will y
ou now give":" (P)ennies":"
(N)ickels":" (D)imes":" (Q)u
arters":" (O)ne dollar bills
":" (F)ive dollar bills":" (
T)en dollar bills"
650 DISPLAY AT(19,1):"":"":"
":"":"":""
660 ACCEPT AT(11,19)SIZE(1)V
ALIDATE("PNDQOFT")BEEP:B$ ::
 DISPLAY AT(11,1):"":"":"":"
":"":"":"":"" :: RETURN
670 FOR J=1 TO 5 :: DISPLAY
AT(20,1):D$ :: DISPLAY AT(20
,1)BEEP:M$ :: NEXT J :: RETU
RN
680 CALL SCREEN(7):: FOR J=1
 TO 15 :: CALL SOUND(-99,110
,0,-4,0):: DISPLAY AT(14,1):
"" :: DISPLAY AT(14,1):"You
gave too much change!" :: NE
XT J :: CALL SCREEN(5):: RET
URN
690 DISPLAY AT(14,1):"No, th
at's wrong!" :: RETURN
```

And here's a one-screen tinygram - you could convert this to speech and it would do a better job of making change than most clerks do nowadays!

```
1 !*************************
  !*      CHANGEMAKER      *
  !*       TINYGRAM        *
  !*    by Jim Peterson    *
  !*************************
100 CALL CLEAR :: S$(1)="S"
110 DEF P$(X)="$"&SEG$(STR$(
X),1,LEN(STR$(X))-1)
120 DATA 10,5,1,.25,.1,.05,.
01,CENT,NICKEL,DIME,QUARTER,
DOLLAR BILL,FIVE DOLLAR BILL
,TEN DOLLAR BILL
130 PRINT TAB(8);"CHANGEMAKE
R":::: :: INPUT "PRICE? ":P :
: INPUT "AMOUNT OFFERED? ";B
 :: IF B<P THEN 130
140 PRINT :: C=B-P :: FOR J=
1 TO 7 :: READ X(J):: A(J)=I
NT(C/X(J)):: C=C-A(J)*X(J)::
 NEXT J :: Z=P+.001
150 PRINT P$(P+.001);" OUT O
F ";P$(B+.001)::: :: FOR J=7
 TO 1 STEP -1 :: READ A$ ::
Z=Z+A(J)*X(J):: IF A(J)>0 TH
EN PRINT "AND";A(J);A$&S$(-(
A(J)>1))&" IS ";P$(Z)::;
160 NEXT J :: PRINT :: RESTO
RE 120 :: GOTO 130
```

The Extended Basic Manual did a very poor job of showing us how to use USING. My thanks to Karl Romstedt for telling me how to do it with DISPLAY AT - put a semicolon directly before USING -

```
100 CALL CLEAR :: DISPLAY AT
(12,5):USING "$##.##":1.23
```

Other commands can go either before the AT or after the parameters -

```
110 DISPLAY ERASE ALL BEEP A
T(12,5):USING "$##.##":1.23
120 DISPLAY AT(12,5)ERASE AL
L BEEP:USING "$##.##":1.23
```

However, to output to a printer, put a comma before USING -

```
120 OPEN #1:"PIO" :: PRINT #
1,USING "$##.##":1.23
```

The trouble with PRINT USING "$##.##" is that it will print nothing but asterisks if the integer contains more digits than the number of # left of the decimal, and will leave blanks between the $ and the first digit if the integer contains less digits than the number of ## left of the decimal. This algorithm will correctly print dollars and cents values of ANY size, rounded off to the nearest cent and with the dollar sign directly before the first digit or decimal.

```
100 INPUT A :: PRINT USING "
$"&SEG$(RPT$("#",LEN(STR$(IN
T(A)))),1-(INT(A)=0),255)&".
##":A :: GOTO 100
```

Thanks to Ed Machonis for some of these improvements to the Printall program which was published in Tips #45 -

```
171 DISPLAY AT(8,12):"V.1.2"
190 DISPLAY AT(18,7):"TURN P
RINTER ON!":::"SET TOP OF FO
RM HALF INCH  BELOW PERFS"
200 DISPLAY AT(23,9):"PRESS
 ANY KEY" :: DISPLAY AT(23,8)
:"press any key" :: CALL KEY
(0,K,S):: IF S=0 THEN 200 EL
SE CALL CLEAR
330 IF P=1 AND SS$<>"Y" THEN
 DISPLAY AT(12,1):"EMPHASIZE
D? (Y/N) Y" :: ACCEPT AT(12,
19)VALIDATE("YN")SIZE(-1)BEE
P:E$ :: IF E$="Y" THEN PRINT
 #1:CHR$(27);"E";
390 IF NC=1 THEN 410 :: AV=I
NT(TA/(NC-1)):: DISPLAY AT(1
2,1)ERASE ALL:"COLUMN SEPARA
TION?":"MINIMUM 2":"MAXIMUM
 "&STR$(AV)&" AVAILABLE ":"2"
400 ACCEPT AT(15,1)VALIDATE(
DIGIT)SIZE(-2)BEEP:CS :: IF
 CS<2 OR CS>AV THEN 400 ELSE
 S$=RPT$(" ",CS)
450 LSP=12 :: DISPLAY AT(10,
1):" ":" ":"LINES PER PAGE?
60":" ":" ":" " :: ACCEP
T AT(12,17)VALIDATE(DIGIT)SI
ZE(-3):LP :: IF LP<61 THEN 4
90
460 !DELETE
470 !DELETE
480 LSP=72/(LP/10):: PRINT #
1:CHR$(27);"A";CHR$(LSP)
510 DISPLAY AT(15,1):STR$(LP
)&" lines per page":"with "&
STR$(INT(LSP))&"/72 line spa
cing"
640 ! DELETE!
650 IF LEN(M$(IP))<=CW THEN
670 :: T$=SEG$(M$(IP),1,CW):
: CALL SOUND(1000,110,0,-4,0
):: DISPLAY AT(12,1):M$(IP);
" OVER";CW;"CHARACTERS":"TRU
NCATED TO ";T$;"OK?"
660 CALL KEY(3,K,S):: IF S=0
 THEN 660 ELSE IF K<>89 THEN
 STOP ELSE M$(IP)=T$
```

MEMORY FULL!

Jim Peterson

File Identification

Should you ever run across an unknown file that you don't know where it may belong or how to run or read it, here are the normal identifications of the most common files. A program can save a file in almost any format, however, but it may prove wise to first assume that it fits these easily recognised patterns.

```
Program ... BASIC program
            Extended Basic program
            PRK file
            33 or 34 sector length
               Program Image Assembly
                  to load
                E/A Option 5
                TIW Option 3
                FUNLWEB Option 3
                FUNLWEB Option 2 GPL
                X-B Image Loader
            25 sector with _P or _C
               TI-ARTIST file
            54 sector
               GRAPHX screen
               Scott Adams Adventure

IV 254 .... X-Basic >48 sectors

DV 163 .... X-Basic MERGE file

DV 80 ..... Standard text file (TIW)
            MAX-RLE file
            PICASSO file

DF 80 ..... Assembly LOAD & RUN
               to load
             E/A Option 3
             FUNLWEB Option 4

                if uncompressed
             X-Basic or E/A Basic:
               CALL INIT
               CALL LOAD("DSKn.file")
               CALL LINK("program")

DF 128 .... Archived file
IV 128 .... Archiued file

IF 128 .... Multiplan file
```

To find DF80 programname (LINK), load program and run this in X-B:

```
100 FOR X=16128 to 16383
110 CALL PEEK(X,Y)
120 PRINT CHR$(Y)
130 NEXT X
```

Among assorted garbled mish-mash you will find the START name.

TELCO QUICK REFERENCE

TELCO Editor Key Functions

Fctn-1 Delete Character at cursor
Fctn-2 Insert Character at cursor
Fctn-3 Delete line
Fctn-5 Clear Input
Fctn-8 Insert line
Fctn-S Left Cursor
Fctn-D Right Cursor
Fctn-E Cursor up
Fctn-X Cursor down
Ctrl-A Clear all tabs
Ctrl-B Set Bell
Ctrl-C Clear Tab
Ctrl-P Place Tab
Ctrl-R Set Right Margin
Ctrl-S Show Tab Line
Ctrl-T Tab

TELCO Terminal Function Keys

Fctn-1 Auto Dialer
Fctn-2 Print Spooler Toggle
Fctn-3 Window Left
Fctn-4 Download files (Page Down on Geneve)
Fctn-5 Window Right
Fctn-6 Upload files (Page Up on Geneve)
Fctn-7 Help
Fctn-8 Review Buffer
Fctn-H Hangup
Fctn-M Macro Select
Fctn-Y Screen Setup Options
Fctn-N Full/Half Duplex toggle
Fctn-L Log open/close
Fctn-/ Log Hold
Fctn-J Window Lock toggle
Fctn-V Status line toggle
Fctn-. (function-period) Conference Mode
Fctn-9 Reset Clock
Ctrl-2 Clear screen locally

TELCO Review Buffer Function Keys

Fctn-1 Top of review buffer
Fctn-2 Bottom of review buffer
Fctn-3 Window Left
Fctn-4 Window Down
Fctn-5 Window Right
Fctn-6 Window Up
Fctn-7 Help
Fctn-8 Screen Dump to a device
Fctn-P Purge review buffer
Fctn-E Line up
Fctn-X Line down
Fctn-S Column left
Fctn-D Column right

TELCO Auto Dialer Function Keys

To view the list of numbers use:

Fctn-4 Page down
Fctn-6 Page up
Fctn-X Line down
Fctn-E Line up

by Joe Nollan.

How and Why Intergrated Circuits Work

A sheet of paper crossed my desk the other day, and as I read it, a basic truth came over me. So simple! So obvious we couldn't see it! The author, of unknown origin, I think has discovered what makes integrated circuits work. He says that smoke is the thing that makes IC's work., because every time you let the smoke out of an IC it stops working. I was flabbergasted! Of course! Smoke makes all electrical things work.

Remember the last time the smoke escaped from the Lucas voltage regulator on your car? Didn't it quit working? I sat and smiled like an idiot as more of the truth dawned on me. Its the wiring harness that carries smoke from one device to another in your machine, and when the harness springs a leak, it lets the smoke out of everything all at once, and then nothing works! The starter motor requires large quantities of smoke to operate properly, thts why the wire going to it is so big.

There is more. Feeling very smug, I continued to expand my hypotheses. Why are Lucas electrics more likely to leak smoke than say, Bosch? Hmmm. Aha!! Lucas is British. Things British make always *leak!* British convertible tops leak water, British engines leak oil, the British government leaks defense secrets. Naturally British electrics leak smoke.

So, in view of all of this, do everything possible that you can to keep the smoke in your computer.