SAC-TUG
SACRAMENTO
TANDY USERS' GROUP.

VOLUME 2 NUMBER 1   FEBRUARY 1987

The President Speaks.....

It's a "NEW YEAR" and SAC-TUG is ready to to lead you to new adventures on your TANDY computer. SAC-TUG is building it's PUBLIC DOMAIN library to better serve you, so come on out and support us.

At the February meeting you can expect an "OLE TIME BULL SESSION". This means that you can ask questions about hardware, software or anything else, there might even be some answers. There will also be time to work with the club librarys (MS-DOS, TRS-DOS and C/PM).

This new year also finds SAC-TUG with some important goals, like increasing the membership, starting some club projects and building a bigger club library. Bring your ideas to the meeting and share them, after all, SAC-TUG is what ever you make of it.

.....Richard Larry Ward

Yes we will have a February meeting, the first meeting of the NEW YEAR. Bring a friend and enjoy a great evening with our second love (first love, if you're not married).

The SAC-TUG NEWS is published monthly by the Sacramento Tandy User Group. Membership in SAC-TUG is $12.00 annually. The membership fee includes a one year subscription to the SAC-TUG NEWS.

Non-profit organizations (e.g. computer clubs) are free to reprint material from this newsletter, with the appropriate credit given. Profit organizations must obtain permission from the editor to reprint material.

SAC-TUG maintains a newsletter exchange with other clubs. To join the exchange program, send us a newsletter.

The deadline for the SAC-TUG input will normally be the 25th day of the month. Anything received after this date will be held for publication in the following month's issue. Any questions should be directed to the editor.

Your mailing label will tell you when it is time to renew your membership. Renew promptly to avoid missing a single issue of the SAC-TUG NEWS.

SAC-TUG OFFICERS
President
        Richard Larry Ward
Secretary
        Bill Gregory
Treasurer
        Louis Valenta
Librarian/BBS
        Gary Peterson

****
****

The following is a list of the SAC-TUG MS-DOS library. To get a copy of any disk, you must be a current SAC-TUG member and have paid the $5.00 library fee. Copys can be made at the monthly meeting.

----

0. The Library Index (3 disks), guide to the entire library
1. PC-Write, Quicksoft's word processor, v, 4.0
2. CHASM & NUMZAP by David Whitman, Cheap Assembler v. 4.0
3. PC Picture Graphics by Eugene Ying (color/graphics requires)
6. FRED, etc., the FRee EDitor, Full Screen EDitor, etc.
7. LadyBug, Logo's Turtle Graphics, v. 1.0D
8. FREECALC, a spreadsheet modeled on Multiplan, v. 1.0
9. Letus A-B-C (8 disks), magazine index
10. BBS-TALK, modified version of Fluegelman's PC-TALK III
11. Utilities, includes a 160K RAM drive and BASIC XREF
13. PC-File III, Jim Button's

excellent database program, v. 4.0

14. DOS 2 Utilities, includes various RAM drives & directory sorts

15. EPISTAT, Tracy Gustafson's statistical analysis program, v. 3.0

16. Utility 1-2-3, macros and tutorials for Lotus 1-2-3

18. Pascal programs, a sampler from Capital PC

19. Designer, sprite graphics for BASIC by Jan Young

20. ExpressCalc from ExpressWare, excellent spreadsheet, v. 3.0

21. DiskCat by Nelson Ford, disk cataloging program, v. 4.0

22. ECSTAT, statistical analysis package by Robert Dohner, v. 1.1

23. KINETICS, linear programming, the simplex method

24. Genesis Finance Manager, general ledger program, v. 4.0

25. PC-DIAL, Button's communications program (incl jr-Dial), v. 1.5

26. Ultra Utilities 4.0, FreeSoft, reads/writes nonstandard formats

27. SCREEN, screen editor from Basic Business Software, v. 3.0

28. Newkey & Defkey, keyboard redefinition programs

29. Desktop, Lotus 1-2-3 templates -- calandar, scratchpad, etc.

30. File Express (2 disks), ExpressWare database, v. 3.7

31. PC-Prof and ZOO SORT, BASIC and sorting tutorials

33. Sketch-A-Color by Mark Bridger, color/graphics sketchpad

34. Word Processing for Kids, large font wp by Sidney Nolte

35. PC-Graph, Jim Button's graphics add-on for PC-File III, v. 1.0

36. The Nimbus Anthology (2 disks), Bill Juhl's utility kit, v. 1.01

37. RPN Calculators, scientific RPN calculators, RAM resident

38. PC/Pad, Fraundorf's word processor & spreadsheet, v. 1.31

39. File Commando, DOS menu, v. 1.30 (includes BURNOUT)

40. Extended Batch Language, better batching in DOS, v. 2.04

41. Draw and DrawIT, graphics tools (color/graphics required)

42. DOS HELP, Chris Bailey's HELP screen for DOS

43. Time and Money, home finance program from MPW

44. Qmodem, Forbin Prject modem program, v. 2.3

46. Turbo WHEELS, forty Turbo Pascal routines, Neil Rubenking

47. Personal Accounts Manager (2 disks), S. E. Button, v. 1.1

48. Fas-Type by Trendtech, typing tutor, v. 2.1 (c/g required)

49. PC-DBMS, relational database from KWare, v. 1.2

52. FANSI-CONSOLE by Hersey Micro, fast console driver, v. 1.11N

53. Power Worksheets, Lotus 1-2-3 templates

54. Diskette Librarian by Don Mankin, v. 7k, plus ARC 5.1, etc.

56. ShortCut by Mossy Rock Sytems, resident DOS shell, v. 2.03sw

57. PC-DeskMates, resident notepad, dialer, etc., v. 1.00

58. PC-Key-Draw by Oedware, high-power drawing tool, v. 3.08

60. PC-Type, Jim Button's word processing program, v. 1.00

61. ProComm, Datastorm Technologies modem program, v. 2.42

62. Stock Manager, CompTalk, by Jim McClellan, and PC-Stock

63. Turbo Windows, Turbo Pascal window managers and more

64. QSYS 3.00 by Dennis Lee.

personalized menu system
65. Prolog, AI programming
language & XLISP 1.4
66. newBASE, BASIC database
manager (with source)
67. Turbo Lessons, turbo Pascal
tutorial by Leslie Faurot
68. Small-C, C compiler by
Hendrix (requires assembler!),
v. 2.1
69. The Draftsman graphics
program by David Stang
70. The Still River Shell, DOS
shell program, v. 1.21
71. Icon Maker by Sidney Nolte
and Don Mankin's Slide show
72. TMODEM from P & M (with
Turbo Pascal source code!), v.
2.0
73. Tshell, BTM Software's DOS
shell and encryption, v. 1.10
74. Turbo Calc by P & M
Software, v. 4.1
75. PIBTERM, Philip Burn's
modem and terminal program, v.
3.2.5
76. ABC Design, graphics editor
by Automatic Micro Systems
77. CED, DX & DISKIT, DOS
command editor and extended dir
78. FreeWord, stillwell
Software's word processor, v.
1.0
79. FreeFile, Stillwell
Software's database program, v.
1.0
80. MandelZoom, Sintar
Software's fractal graphics
81. 3by5, File-card emulating
database program
82. PC-GL/AR/PR; Medlin's
ledger, acc'ts receivable, &
payroll
83. Assemble, mini-assemble;
ASMGEN disassembler; Res86
debug
84. EZ-Forms, forms generator
by MaeDae, v. C.2
85. PC-Outline by SoftWorks
Development, v. 1.08
86. PC-Style, word processing
utility from ButtonWare, v. 1.0
87. ZIP, integrated file

handler by Edward V. Dong, v.
0.86C4
88. DOSmatic operating shell by
Marin Pacific Software, v. 2.00
89. EDIT, text editor by Thomas
Thurston, v. 1.20
90. WORDFLEX word processing
program by Nemco, v. 1.31
91. Video Chem, high school
chemistry tutorial by Andy
Schoals
92. D-Cube, full-function modem
program by Dorn Stickle, v.
2.02
93. ChiWriter tech word
processor, Horstman Software,
v. 1.12
94. pBASE relational database
from 4B Associates, v. 1.02
95. PibCalc programmable
calculator by Phil Burns, v.
1.02
96. Minihost by Don Mankin,
host program for remote access
97. NAMEGRAM anagram generator
by Neil Rubenking, v. 1.1
98. Pc/370 Assembler, produce
IBM 370 programs on your PC
99. Grade Guide, data entry and
grade computing system
100.The SLIC Works, intergrated
package
101.A86 macro assembler by Eric
Isaacson, version 2.11
103.Stock Trader investment
program from DSF Associates, v.
2.00
104.Ctr1Alt resident utility by
Barry Simon and Rick Wilson, v.
1.0
105.As-Easy-As, 200 * 52
speadsheet clone of 1-2-3,
version 1A
106.Disk Organizer 1.01a, Soft
GAMS, and STUFIT 0.6
107.Dupes!, PC-File utility
from Reflections Software,
version 1.00
108.QEDIT text editor from
SemWare, version 1.25
109.WYSIWYG character-based
graphics editor (no CGA
required)

THE PROGRAMMERS WORKSHOP
BY
Richard Larry Ward
Copyright October 1984

PRINTING IN A VERTICAL POSITION

Have you ever had a table to print that could have been printed in several vertical columns? Did you print it horizontally instead? If so, you are not alone. Many programmers have done that. Printing your listing to read in multiple vertical columns on a page requires quite a bit of planning in advance. However, there is a rather simple trick to it which I shall show you in this article.

I first did this in FORTRAN V on a UNIVAC 1108. I have since had two other occasions to use this routine in COBOL on an IBM 370. The first program was used to list the coordinate points making up the polygons in the design of MOS chips such as those of the micro-processor chips found in your computer. This was part of an error checking system used to verify the correctness of the raw input data used in the design and manufacture of MOS microcircuitry.

The second occurrence was while I was working for CALTRANS. I wrote a program that lists the status of all ongoing contracts between CALTRANS and private contractors throughout the state. The listing included a brief description, the estimated cost at completion, and the cost to date of each contract. An index at the end listed every contract and the page number on which that contract could be found.

The last occurrence was a report for the Legislative Council Bureau's Keyboarding unit listing all requests and bills in the data entry file. A cross reference at the end of the report identifies which requests became which bills and shows the page number of the report where the request ID is found.

Prior to my first article, I put a message to ALL in the Programmers Base in which I mentioned that every programmer should maintain a notebook of programming techniques. This was the routine which prompted me to start my own notebook. When I wrote the program for CALTRANS, I remembered having written a program which printed vertical columns across a page, but not exactly how to do it. After much effort in re-developing the routine, I decided never to let that

situation arise again. My notebook was the result.

This technique is so useful that I fully expect to have many other uses for it in the future. I have been thinking of writing an Index program which would create and print an Index to a manual or other documentation. This would be another excellent use for printing vertical columns on a page.

The first step in this procedure is to determine exactly how many lines of how many columns you wish to print on a page. Lets say that you are going to print a report consisting of five columns of fifty lines. This means that you will print 250 entries on a page. If you are going to store all of your data in a table in memory, then you will step through it in steps of 250 entries at a time. If your data is stored in a file, you will have to define a table of 250 entries and read data into it for each page that you print. In either case, you will have to have as many indexes as you have columns. The first index is varied from the current value of the page pointer to the page pointer plus the column length minus one. The second index is set to the column length plus the first index. The third index is set to 2 times the column length plus the first index. The fourth index is set to 3 times the column length plus the first index, etc.

If there is less than a page of data to be printed, or when the last page of a report contains less than a full page

worth of data, the column length must be adjusted. The column length should be computed to be the number of items left to be printed divided by the number of columns on the page. If there is a remainder, add one to the column length. This will insure that all but the leftmost column will be of equal lengths.

```
CLS: CLEAR 20000
1100 DIM DA$(1000), I2(1000)
1200
AL$="ABCDEFGHIJKLMNOPQRSTUVWXYZ"
1300 T1$="MULTIPLE VERTICAL
COLUMN"
1400 T2$="SAMPLE PROGRAM"
1500 GOSUB 3200
1600 GOSUB 2100
1700 GOSUB 4100
1800 GOSUB 2100
1900 GOSUB 6600
2000 END
2100 FOR P1=1 TO 1000 STEP 250
2200 IF P1 > I1 THEN
P1=P1+1000: GOTO 2900
2300 IF P1+250 <= I1 THEN GOTO
2700
2400 P2=I1-P1: CL=P2/5:
X1=INT(P2/5)
2500 IF CL > X1 THEN CL=CL+1
2600 GOTO 2800
2700 CL=50
2800 GOSUB 5600
2900 NEXT P1
3000 RETURN
3100 END
3200 REM INITIALIZE TABLE TO BE
PRINTED
3300 I1=RND(1000): PRINT@1,I1
3400 FOR I=1 TO I1
3500 FOR J=1 TO 6
3600
DA$(I)=DA$(I)+MID$(AL$,RND(26),1
)
3700 NEXT J
3800 I2(I)=I: PRINT@65,I;DA$(I)
3900 NEXT I
4000 RETURN
4100 REM. SORT ROUTINE THIS IS
```

```
                              PROBABLY THE MOST INEFFICIENT
4200 REM   WAY OF SORTING THAT
           THERE IS
4300 REM
4400 FOR I=1 TO I1-1
4500 FOR J=I+1 TO I1
4600 IF DA$(J) < DA$(I) THEN
     GOTO 4800
4700 GOTO 5100
4800 SW$=DA$(I): SW=I2(I)
4900 DA$(I)=DA$(J): I2(I)=I2(J)
5000 DA$(J)=SW$: I2(J)=SW
5100 NEXT J
5200
     PRINT@128,"SORTING";I;J;DA$(I)
5300 NEXT I
5400 RETURN
5500 END
5600 REM PRINT PAGE
5700 LPRINT TAB((80-
     LEN(T1$))/2);T1$
5800 LPRINT TAB((80-
     LEN(T2$))/2);T2$
5900 LPRINT STRING$(80,"="):
     LPRINT" "
6000 FOR C1=P1 TO P1+CL-1
6100 C2=CL+C1: C3=2*CL+C1:
     C4=3*CL+C1: C5=4*CL+C1
6200 LPRINT I2(C1);
     DA$(C1),I2(C2)
     ;DA$(C2),I2(C3);DA$(C3),I2(C4);
     DA$(C4),I2(C5); DA$(C5)
6300 NEXT C1
6400 FOR I=1 TO 12: LPRINT" ":
     NEXT I
6500 RETURN
6600 REM END OF JOB
6700 CLS: PRINT"DONE!!!"
6800 RETURN
```

The mainline is the string
of GOSUBS in lines 1500-1900.
The subroutine at line 3200
loads the table with data. The
subroutine at line 2100 prints
the table. The Gosub at line
1600 prints it in the order in
which it was generated. The
subroutine beginning at line
4100 sorts the table. Please

note, this is the worlds worst
sort algorythm. More on
sorting in another article.
Finally, we print the sorted
table.

Lines 2100 through 3000
control the processing of the
pages. I have determined that
I want to print 250 entries per
page in five columns of fifty
entries. Therefore, I use the
STEP parameter of the
FOR...NEXT statement step
through the table in one page
increments. However, I must
check for short pages (line
2300). Please note that the
test on line 2200 is for a
situation which should never
occur. Lines 2400 and 2500
adjust the column length to
approximately one-fifth of the
remaining data in the table
when there is less than 250
entries remaining.

Lines 5600 through 6500
actually print the page. Note
the setting of the column
subscripts in line 6100. The
column 1 subscript is
controlled by the FOR...NEXT
loop starting at line 6000.
The remaining subscripts are
set to the appropriate multiple
of the column length plus the
value of the first subscript.

As I mentioned earlier,
the sort routine that I used in
this program is one of, if not
the very worst of sort
algorythms in the world. One
of my test runs of this program
ran for over three hours
sorting 551 entries.

-- END --

Model 4 features in Model III Mode
By Steve Board

From the Nothern Illinois Computer Owners League DIME and NICOL Newsletter, March 1986. Reprinted from the December 1986, NCTCUG Newsletter.

Those of us who cut owr teeth on the Model I or III have found the Model 4 to be only a partial upgrade. It provides features like sound, 80-character screen, a ramdisk, and reverse video. And it's twice as fast for calculations. But its BASIC loads in slower and we all have lots of programs in Model III formats into which we would like to splice these features rather than convert them to Model 4.

Increasingly I have come across methods to add the 4's features to III mode. Speedup programs are available everywhere. I have put on the WACO BBS (312-351-4374) a program called SOUND43/BAS which pokes into memory the proper code for creating sound internally from the Model 4 while in Model III mode. It is adapted from Hardin Brothers' machine language program in the January 1984 80 Micro. Here are the key lines:

```
20 PRINT "DEMONSTRATION OF
SOUND IN MODEL III MODE FOR
MODEL 4 TRS80S"
30 PRINT "FIRST WE POKE THE
ROUTINE INTO HIGH MEMORY:"
40 READ A: IF A<>0 POKE
&HFFDC+B,A ELSE GOTO 170
50 PRINT PEEK(&HFFDC+B); 'THIS
IS TO SHOW WHAT'S HAPPENING
60 B=B+01:GOTO 40 'STEP UP THE
POKE ADDRESS BY ONE
70 DEFUSR=&HFFDC 'DEFINE WHERE
THE USR ROUTINE HANGS OUT
80 DEFFNT=USR(K) 'THIS PUTS THE
USER FUNCTION INTO A DEFFN
VARIABLE "T"
90 ' THE FOLLOWING LOOP
DEMONSTRATES THE PITCH &
DURATION FOR THE VALUE
100 'OF "K" INDICATED. YOU CAN
GO TO 32000
110 PRINT "NOW WE WILL VARY THE
VALUE OF 'K' TO SHOW VARIED
PITCH"
120 PRINT "AND DURATION"
130 PRINT "K IS NOW: "
140 FOR K=10 TO 220 STEP 5
150 PRINT FNT; 'YOU COULD ALSO
USE T=USR(K)
160 NEXT K
170 PRINT
180 DATA
205,127,10,243,76,69,62
190 DATA 1,211,144,16,254,69,62
200 DATA 2,211,144,16,254,13,32
210 DATA 239,251,201,201,0,0,0
220 PRINT:PRINT"YOU CAN DEFIN
'K' AS A VALUE AND USE IT
ANYWHERE":PRINT"YOU WANT IN
YOUR PROGRAM AS 'PRINT FNT'"
230 PRINT:PRINT"FOR EXAMPLE,
YOU COULD DROP IN 'K=50:PRINT
FNT' AND GET THIS:":K=50:PRINT
FNT
240 PRINT "IF YOU WANT NO
SCREEN INDICATION OF K, USE
'K=50:T=USR(K)' AS IN THIS:":
FOR TM=1 TO 900:
NEXT:K=50:T+USR(K)
250 'ADD YOUR OWN RETURN IF YOU
MAKE THIS A SUBROUTINE
                -- END --
```

─── SAC ~ TUG ───

SACRAMENTO TANDY USER'S GROUP
(SAC-TUG)
MEMBERSHIP APPLICATION

(____) Visitor (Complimentary SAC-TUG Newsletter)
(____) New Membership
(____) Renewal
(____) Information Change

------ PLEASE PRINT CLEARLY ------

Name: _____ Phone: _____

Mailing Address: _____

City: _____ State: _____ Zip: _____

(____) I DO NOT wish my name be released to others for mailing lists, etc.

Signature: _____ Date: _____

To access club library, pay an additional one time fee of ($5.00)
One year membership to SAC-TUG ($12.00)
Please mail this form and remittance to:

SAC-TUG
3910 RENICK WAY
NORTH HIGHLANDS, CA. 95678
Attention: Secretary

The public is invited to attend the monthly SAC-TUG meeting. Members will
receive our monthly newsletter, "THE SAC-TUG NEWS" and other special
privileges. SAC-TUG meets at 7:00 pm the first Thursday of every month at
the SMUD training building 59th ST. and Folsom Blvd. directly behind Corti
Bros. store. Please stop in and join us.

~THE SAC-TUG NEWS~

FEBRUARY 1987

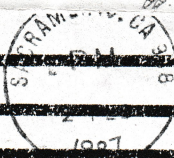### THE OFFICIAL NEWSLETTER OF THE SACRAMENTO TANDY USER'S GROUP

---------- IN THIS ISSUE OF THE SAC-TUG NEWS ----------

PART FIVE OF "THE PROGRAMMERS WORKSHOP"

MORE ABOUT MODEL 4 FEATURES IN MODEL III MODE"

--------- FEBRUARY SAC-TUG MEETING ---------
TO START OUT 1987 WE WILL HAVE A GOOD OLE BULL
SESSION. THERE WILL BE TIME FOR QUESTIONS AND A
CHANCE TO GET BACKUPS OF THE CLUB LIBRARYS.......

SAC-TUG NEWS
3910 RENICK WAY
NORTH HIGHLANDS, CALIFORNIA
95660

DON MCDERMOTT 3/87
8405 BENNINGTON WAY
SACRAMENTO, CA. 95826

*** IT'S GOING TO BE A GREAT YEAR FOR EVERYBODY ***