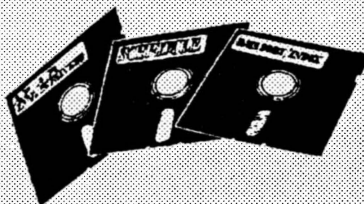
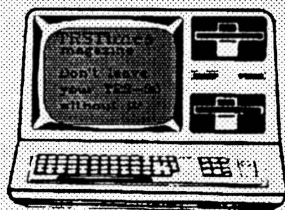


THE INTERFACE

NEWSLETTER OF SAGATUG

THE SAN GABRIEL VALLEY TANDY USER'S GROUP

THE CLUB FOR TRSDOS, MS-DOS, CPM AND LAPTOP COMPUTERS



Volume 12 - Number 11

Price \$1.50

November 1991

ORCHESTRA-90 FOR TRS-80 MODEL 4

Most people think that the Model 4 is an antiquated machine not capable of most of the things we are so used to with our PC's. To prove how wrong this is, we have in the past demonstrated the machine's hi-rez graphics abilities; at a future meeting GRAHICS-90 will show how well the Model 4 can handle low-resolution graphics and animation. This month, our illustrious president, Roy Beck, will tell us about ORCHESTRA-90, a great program that turns our 'old piece of junk' into a music machine, almost rivaling a PC with a midi board installed. Yes indeed, the Model 4 playing music with 5 voices - don't miss it!!

The November SAGATUG Meeting will be held Friday, November 8, 1991 at 7 PM
in the Arcadia Park Senior Citizen's Center
405 South Santa Anita Avenue, Arcadia

- SAGATUG meets the second Friday of every month -

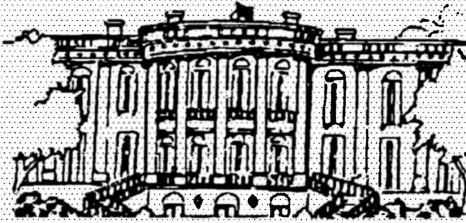
CLUB OFFICERS

President	Roy Beck
Vice President	John Phillipp
Sec'y/Treasurer	Fred Berg
Disk Librarians	Mike Winterer, Lance Wolstrup
Members-At-Large	Allen Jacobs, Rob Malone
Interface Editors	Roy Beck, Lance Wolstrup

INTERFACE Submissions: SAGATUG, 2153 Cedarhurst Dr. Los Angeles, CA 90027

Contribute to the INTERFACE and Club Disk Libraries

THE PRESIDENT'S
COLUMN



BY
ROY BECK

This has to be one of those good news/bad news columns

The good news is that we have a good-looking slate of officers nominated for Sagatug's upcoming 1992 year of operation. I don't even know how many years we have existed, but it really is phenomenal in this declining world.

The slate for election for 1992, as nominated last month, is as follows:

President	Allen Jacobs
Vice-president	Eric Dunne
Treasurer	Fred Berg

I am sure these worthy fellow members will do us proud in 1992, so I say, let's get out the vote for them! In addition, Bob Miranda has agreed to serve as Newsletter Editor, one of the more responsible jobs in our club. Please help him and the club by contributing material for the INTERFACE.

And there is some bad news, also.

The MISOSYS Quarterly, in it's 6th year of publication by Roy Soltoff, will not continue after this year's schedule is complete. The Summer 1992 issue will be the last, so there are three issues to go. MISOSYS is making the transition to the MS-DOS world, (after all, Roy Soltoff does have to eat, too!), and the recently upgraded LDOS 5.3.1 for both the Model I and III may be his last major software effort for the Z-80 world as we know it.

But there is some good news from MISOSYS, also. Roy announced he is seriously contemplating preparation of a new DOS manual. This would be a single manual covering all the DOS commands for LDOS for the Model I, LDOS for the Model III, and LS-DOS for the Model 4. Apparently RS is no longer selling manuals for the Model 4 DOS, (nor the Models I and III) and this combined with the recent improvements in LDOS, has created a significant need for a manual for all the TRS-DOSes.

Since a major part of Roy's effort was to make the three DOSes behave as nearly identically as possible, the production of a common manual is probably relatively simple at this time. I am looking forward to the proposed manual, we need it!

Another piece of good news. The most recent issue of MCTRUG, the Mid Cities TRS User's Group in Arlington, TX, notes that Dave Dalager is alive and kicking! He wrote an article, and made a presentation at a recent meeting. Those of you who know him or of him may have wondered about his status. His phone still doesn't work, so I guess he is still on the outs with Texas Bell, or whatever name the phone company uses there, but Dave is among us!

compiled by Lance Wolstrup

Page 3

Note that both lines have a single space before the "\$D". These PROMPT sequences require that you load ANSI.SYS in your CONFIG.SYS file. ANSI.SYS, the extended keyboard and display driver, comes with DOS. The proper line to add to CONFIG.SYS is:

DEVICE=ANSISYS

Try either prompt, but remember that you should have only one PROMPT statement at a time in your AUTOEXEC.BAT and that you must reboot your computer (press Control-ALT-Delete) to activate the new prompt. Also, remember that DOS updates the time only when it displays a new DOS prompt. The easiest way to see the current time is to display a new DOS prompt by pressing <ENTER>.

WORD FOR WINDOWS

Change .DOC Default

When you click on File Open in Word for Windows, the dialog box comes up with the wildcard filename *.DOC already in place. Word for Windows assumes that you give all documents a .DOC extension - not a good assumption considering all the more informative three-character extensions possible. If you aren't always looking for .DOC files, you have to change the default manually each time.

Change the default so that the dialog box, for example, appears with *.* each time, or displays any other extension that works better for you. First open any file, then click on Macros and Edit. Look for a Macro called FileOpen; this is the code that controls the action of the File Open dialog box. If FileOpen doesn't appear in the list of available macros, click on the box next to "Show All" or just type the name FileOpen. The code that makes up the macro will appear in the macro editor. It should look like this:

```
Sub MAIN
Dim dlg As FileOpen
GetCurValues dlg
Dialog dlg
Super FileOpen dlg
End Sub
```

If you want to change WinWord so that you are presented with a *.* option in the dialog box, insert three lines so that the macro listing reads as follows:

```
Sub MAIN
On Error Goto ENDER
Dim dlg As FileOpen
GetCurValues dlg
dlg.NAME = *.*
Dialog dlg
Super FileOpen dlg
ENDER:
End Sub
```

If you want to use a wildcard filename other than *.* , include the pattern you want within the quotes in the line beginning with dlg. The lines beginning with On Error and ENDER eliminate an error box that appears if you cancel the dialog without choosing a file. Note the colon at the end of ENDER on the next to last line.

Now close the file. You'll be asked if you want to save it; click on Yes. Then exit Word for Windows. You'll be asked if you want to save global changes to the glossary and commands; click Yes again. When you re-enter Word for Windows and open a file, the dialog box gives you the filename pattern you want.

dBASE

Finding Empty Fields

Finding records in which certain fields are empty can be a useful way to locate empty records or those missing important information. However, building the formula to accomplish this task can be tricky. Say your database contains a text field called City, which is 12 characters long. To find all the records in which the City field is blank, the condition

for City=""

won't work. When characters are displayed in the string on the right side of the equation, dBASE tries to match the first n characters of the string on the left side. Since there are 0 characters in the string "", dBASE tries to match the first 0 characters of the City field - in other words, dBASE doesn't test for a match at all. Every record passes the test.

The condition

for ""=City

won't work, either. dBASE tries to match a string of 12 characters contained in the City field - in this case the string you want to isolate is 12 blanks - to a string with no characters. Every record will fail this test.

But the condition

for ""=trim(city)

works. This condition matches the null string with the contents of the City field minus the trailing blanks.

LOTUS 1-2-3

Force a Page Break

1-2-3's Worksheet Page command does two things: First, it inserts the symbol 1:: (only the :: portion is visible in the spreadsheet), and then it inserts a blank row. When 1-2-3 encounters this symbol in the leftmost

column of a range you are printing, it inserts a page break and won't print the row of your print range in which the I:: symbol appears.

This is fine if your print areas are laid out like tiles, with each one below the previous one. In this case, inserting a blank row won't affect the other ranges. But print ranges at different horizontal locations in your spreadsheet may begin and end in different rows. The blank row inserted by the Worksheet Page command to end one print range may appear as an unwelcome gap in another range.

The solution to this problem is simple. Just go to the row where the page break should occur and manually type the three-keystroke symbol I:: (the pipe followed by two colons) in the leftmost column of the range.

PC TOOLS

Launching Applications

To easily navigate through your hard drive, even from within another application, just use PC Shell's Locate, View, and Launch functions.

During installation, when PC Setup asks if you want to search your drive for applications and add them to PC Shell's Application pull-down menu, select I for Install Applications. PC Setup is also copied to your hard drive at this time.

To use PC Shell to launch an application, type PCSHELL at the DOS prompt and choose APPLICATION from the top menu. Then, if you need to jump to another application, it is easy to do.

For example, suppose you are working in Lotus 1-2-3 and need to get some information from a WordPerfect document. Press Ctrl-Esc to hotkey into PC Shell. Choose Locate File from the File menu (or just press F7.) Since you configured PC Shell to search your hard drive for applications, there will be an entry in the Locate dialog box for your WordPerfect files. Select that entry. You are then prompted for the text you want to find in the WordPerfect documents. Enter the optional text. All WordPerfect files that contain the text will appear in a window. To view the files, press F2. To view the next file, press F9. Once you find the right file, press F4 to launch WordPerfect and load the file. When you exit WordPerfect, you'll return to where you left off in 1-2-3.

DESQVIEW

Running Out of Files

If you have trouble getting a program to run under Desqview 386 even though you've allocated as much memory as the program requires, the problem may be in the number of files your system can open at one time. When Desqview opens a program, MS-DOS "grabs" as many file handles as it needs - often five

or more. This can cause the program to run out of files. You can fix this by setting the FILES= statement in your CONFIG.SYS file to FILES=50.

WINDOWS

Better DOS Performance

Windows by default includes support for a wide range of applications and hardware platforms, but if you choose to take advantage of some of those options, it might slow you down. Try the following tip if you want to improve the speed of DOS applications running under Windows' 386 enhanced mode.

The first thing you should do is load the PIF Editor and open the PIF file of the DOS application whose performance you would like to improve. Then click on the Advanced... button; in the section entitled Display Option, be sure that none of the options alongside Monitor Ports are checked. The check boxes are used for application compatibility, but most applications do not in fact require any special settings.

WORDPERFECT

Mousing Around

Even longtime WordPerfect users might be surprised by the extensive mouse support in Version 5.1. Set up your mouse properly, and it will fly through menus and text. You don't even need to load a mouse driver first, as long as your mouse is on the list of supported pointing devices. Choose Shift-F1, M to bring up the Mouse Setup menu. Press T, and choose your mouse type from the nearly 20 listed; if your model isn't supported, choose the generic option, Mouse Driver (MOUSE.COM).

You can fine-tune the mouse's acceleration and the interval between double clicks, but the most important setting is 7 - Assisted Mouse Pointer Movement. Set it to Yes and exit.

Now, every time you click the right mouse button, the cursor will jump straight to the first choice on the top menu. Whenever WordPerfect asks for input, the cursor will jump straight to the status line. Clicking the right mouse button has the same effect as pressing the Exit (F7) key. If you have a three-button mouse, the middle button acts like the Cancel (F1) key; on a two-button mouse, you can mimic the Cancel key by pressing and holding one mouse button and then pressing the other.

From the List Files screen, mouse movement is snappy. Just point to a directory name, double-click with the left mouse button, and click once with the right button to navigate at full speed without ever touching the keyboard.

SETDATE

Model I/III & 4 MULTIDOS

programming technique - BASIC

by Jim King, MSEE, C10

A few months ago I had the opportunity to get acquainted with a nice TRSDOS 6.2 machine language utility called SETDATE. (I believe this program was written by Jack Decker of 'The Alternate Source' fame.) Boy, this is the way to enter the date, much better than typing it in from the keyboard. In essence, the program presents the previous date entered, which is then brought to the present date by pressing the right arrow.

I thought that this would be a nice feature to have in my favorite operating system (MULTIDOS), so I decided to attempt writing a BASIC version of Mr. Decker's idea. Rather than have the program save the new date to a disk file, I wanted to have the program modify itself to retain the new date in a variable and then write the entire program back to disk, thus saving valuable disk space.

Well, this proved to be somewhat tougher than I anticipated! Let me explain what I found myself up against: If line 10 is, for example: 10 A\$="Friday" and then somewhere later in the program A\$ is changed to: A\$="Saturday", when the program is saved back to disk, line 10 does not change. It remains 10 A\$="Friday". This, of course, was the problem. I wanted line 10 to modify itself to read, for example: 10 A\$="Saturday". I knew that by using VARPTR and PEEK to find the address of the variable in question, and then POKEing a new value to it, should modify the code in the line where the variable first appears. However, I just could not get the program to work; no matter where I PEEKed and POKEd, the program either retained the initial value of my variable, or it would simply crash. At this point I decided to get help, so I brought the problem with me to next meeting of VTUG where, after a lively discussion, the gang came up with the solution. SETDATE/BAS, because it is written in BASIC is slower than the TRSDOS 6.2 machine language version, but it certainly does the job.

The program brings up the title, and the statement that it uses the DEFine FuNctions of Lewis Rosenfelder in his book Basic Faster & Better. These DEF FN's are from line 81 to 88. Line 89 is one of my own that takes a 1 or 2 digit integer and converts it to a 2 digit \$string.

Line 200 displays instructions on how to use it: Right or Left arrows to increase or decrease one day, > and < to increase or decrease 10 days. In addition

to the number of the day, it displays the day of the week as a check.

The program works by storing a number that the FN's use to find that date, and the day of the week in A\$. This happens in line 80.

The calculation is done in line 210 and 220 to print the day of the week and the date in line 230.

Lines 3 to 9 are subroutines that I use constantly: 3 prints Z centered.

7 changes a lower case string to upper case.

8 is a one character Inkey\$, including a call to line 7.

9 causes the cursor to Jump Up 1+JU lines.

Lines 240 through 290 perform the addition to or subtraction from the date number, jumps to line 300 after you press <Enter> for the correct date, or cycle back to line 230 for any other key. If a new date is set by one of the arrows, flag K is set =1, and used later.

In line 300 the new value of the date number (A) is put into \$string AA and used after the time is entered. Time\$ is displayed to show what the computer now thinks is the date and time. The time input is then requested in 24 hour clock, 4 digits.

The IF tests in lines 310 and 320 prevent entering illegal times.

The chosen 4 digit time is put into TI\$ in line 330.

Line 340 peeks the address of the first date number, A\$.

Line 350 peeks the address of the new date AA\$.

Line 360 pokes into the address of A\$ the number in AA\$.

Note that the address of the new value is +1 because in line 300 STR\$(A) includes the leading blank.

Line 370 takes the numbers for the month (MO), day (DY), and year (IR) found in line 210 and puts them into DA\$ in the proper order. Then IF the date has been changed, and K=1, the whole program is saved, and the date in memory is corrected.

Line 380 saves the time, then again displays Time\$ to show if it all took.

I incorporated this routine into a shell program that I use all the time. Now, instead of entering the date and time at boot-up, I set it from within the shell program.

SETDATE/BAS

```

0 CLEAR 99:DEFINT B-J,L-Y:DEFSTR Z:
PRINT LEFT$(TIME$,8)" ";
Z="SETDATE - Date Setting Using Routines
from":
GOSUB 3: Z="BASIC FASTER AND BETTER
& OTHER MYSTERIES, IJG Inc.":GOSUB 3:
Z="Copyright (C) 1981, by Lewis Rosenfelder":
GOSUB 3:GOTO 80'-SetDate
3 PRINTTAB(38-LEN(Z)/2)Z:RETURN
7 FOR H=1 TO LEN(Z):
O=ASC(MID$(Z,H,1)):
IF O>96 THEN MID$(Z,H,1)=CHR$(O-32):
NEXT:RETURN
ELSENEXT:RETURN
ELSEReturn
8 Z=INKEY$:IFZ=""THEN 8
ELSE IF Z=CHR$(31) THEN END
ELSE GOSUB 7:RETURN
9 PRINT CHR$(29)STRING$(JU+1,27)CHR$(31);:
JU=0:RETURN
20 A=FNDN!(YR,MO,DY):ZA=STR$(A):
RETURN'date #
21 PRINT'Day # ="A:RETURN
22 DA$=FNZ2(MO)+"/"+FNZ2(DY)+"/"+FNZ2(IR):
RETURN
23 TI$=FNN$(HR)+":"+FNN$(MN)+":00":
RETURN
24 YR=FNRY%(A):J=FNRJ%(A):
MO=FNRM%(J,YR):DY=FNRD%(YR,MO,J):
RETURN 'not used
25 DA$=FNN$(MO)+"/"+FNN$(DY)+"/"+FNN$(IR):
RETURN 'was line22
30 K=PEEK(VARPTR(A$)+2)*256+
PEEK(VARPTR(A$)+1):IF K>32767 THEN K=K-
65536
31 RETURN
80 A$="727179":A=VAL(A$)
81 DEF FNDN!(Y%,M%,D%)=Y%*365+INT((Y%-
1)/4) +(M%-1)*28
+ VAL(MID$("000303060811131619212426",
(M%-1)*2+1,2))-((M%>2)AND((Y%ANDNOT-
4)=0)) +D%'-Day# p110
82 DEF FNRY%(N!)=INT((N!-N!/1461)/365)
' - Year p111
83 DEF FNRJ%(N!)=N!-
(FNRY%(N!)*365+INT((FNRY%(N!)-1)/4))' - p111
84 DEF FNRM%(J%,Y%)=-((Y%ANDNOT-4)>0)*
(1-(J%>31)-(J%>59)-(J%>90)-(J%>120)-(J%>151)-
(J%>181)-(J%>212)-(J%>243)-(J%>273)-(J%>304)-
(J%>334))-((Y%ANDNOT-4)=0)*(1-(J%>31)-(J%>60)-
(J%>91)-(J%>121)-(J%>152)-(J%>182)-(J%>213)-
(J%>244)-(J%>274)-(J%>305)-(J%>335))
85 DEF FNRD%(Y%,M%,J%)=(J%-(M%-1)*28
+VAL(MID$("000303060811131619212426",
(M%-1)*2+1,2))))+((M%>2)AND((Y%ANDNOT-4)
=0))' - p111

```

```

86 DEF
FNN$(N)=RIGHT$(STR$(N),LEN(STR$(N))-1)
' #->$
87 DEF FNDY$(N!)=MID$("Friday Saturday Sun-
day Monday Tuesday WednesdayThursday ",
(N!-INT(N!/7)*7)*9+1,9) ' - p110
88 'MO=VAL(LEFT$(TIME$,2)):
DY=VAL(MID$(TIME$,4,2)):
YR=VAL(MID$(TIME$,7,2)):
HR=VAL(MID$(TIME$,10,2)):
MN=VAL(MID$(TIME$,13,2)):
YR=YR+1900:
GOSUB 40:GOSUB41
89 DEF FNZ2(N)=RIGHT$(STR$(N),2)
90 'No Error Handling yet
99 MO=0:DY=0:YR=0:
PRINT"PRESS: Right Arrow to Advance 1 Day,
Left Arrow to Back Up 1 Day, > or < to incre-
ment by 10 days. <Enter> when Date is Correct."
190 'PRINT" A","K","K+1","K+2","K+3
200 YR=FNRY%(A):
J=FNRJ%(A):
MO=FNRM%(J,YR):
DY=FNRD%(YR,MO,J):
IR=YR-1900
205 GOSUB 30
210 PRINT FNDY$(FNDN!(YR,MO,DY));'day of
week
220 GOSUB 22:PRINT" "DA$
230 GOSUB 8:
IF PEEK(14400)=64 THEN A=A+1:
GOSUB 9:GOTO 200
240 IF PEEK(14400)=32 THEN A=A-1:
GOSUB 9:GOTO 200
250 IF Z=">"THEN A=A+10
260 IF Z="<"THEN A=A-10
270 IF Z=CHR$(13) THEN 285
280 GOSUB 9:GOTO 200
285 AA$=STR$(A):
KK=PEEK(VARPTR(AA$)+2)*256
+PEEK(VARPTR(AA$)+1):
IF KK>32767 THEN KK=KK-65536
287 GOSUB 30:
FOR X=0 TO 5:
POKE K+X,PEEK(KK+X+1):NEXT
295 PRINT"A="A
300 'GOSUB 30
310 CMD"DATE "+DA$:
PRINT"Time$ = "TIME$
400 PRINT"Time: Hour in '24'";:INPUT HR
410 PRINT TAB(28)CHR$(27);:
INPUT"Minute";MN:
MN=MN+1:GOSUB 23
420 PRINT FNDY$(FNDN!(YR,MO,DY))
" "DA$" "TI$
430 YR=YR-1900:GOSUB 22
440 CMD"DATE "+DA$
450 PRINT"Time$ = "TIME$:
END

```

IN THIS ISSUE:

PRESIDENT'S COLUMN	2
PC HINTS & TIPS	3
SETDATE	6
SETDATE/BAS	7

SAGATUG (San Gabriel Valley Tandy User's Group)
P.O. Box 6818
Burbank, CA 91510