TANDY
ELECTRONICS
The biggest name
in little computers

MICROCOMPUTER NEWSLETTER

Volume No. 9 May-June, 1980.

Prices may vary at individual Stores and Dealers

FREE CATALOGUE!

Included with this Newsletter is a copy of the new RSC-3 Computer catalogue. It contains the latest prices and information on both Model I and Model II TRS-80 computers. The catalogue is also available at no charge at any Tandy store.

NEW PRODUCTS

SCRIPSIT

At last! Word processing for the TRS-80 Scripsit is a powerful full-featured system designed to allow you full editing and formatting capabilities. The system includes Global Replace, Delete, and Find capabilities. Scripsit utilizes a unique audio tape teaching system to familiarise you with the features of the program. You simply listen to the tape and follow directions. We have provided sample text to demonstrate the many capabilities of Scripsit.

By defining a special key as a "control" key it is possible to have single key control over such functions as inserting, deleting, exchanging or moving words or paragraphs. We provide stick-on lables which identify the dual function keys.

Scripsit gives you automatic page numbers, page headings and footnotes and makes it easy to indent paragraphs, change line lengths and centre your test. It also allows single page printing if you wish to use company letterheads (requires a friction feed printer). If you use a serial printer Scripsit includes software to allow its use. Requires 16K Level II plus printer.

Scripsit. Disk. 269-1563 .... 139.95
Tape. 269-1505 .... 99.95

Accounts Payable:

One of the most important tasks in any accounting department is monitoring the flow of cash into payable accounts. Some invoices should be paid immeditely to take advantage of discounts. Other invoices may be held for longer periods of time. The Tandy Accounts Payable System (APS) will help you watch all vendor accounts and invoices, so that you may keep as much cash on hand as possible while still retaining good credit.

The APS will print cheques, cash requirements, aging reports, and totals for your company's General Ledger. The APS will help you decide when to write your cheques for maximum discount, and how those cheques will affect your cash position.

Reports products include:—

Complete or selected vendor file
Complete or selected invoice listing
Complete or selected posting report.
General Ledger recap report
Cash requirement report
Complete or selected aging reports
Discounts lost report

APS will handle 200 vendor accounts with 1000 invoices per month. An option allows for 75 vendor accounts and 1500 invoices per month. This is for a 32K Business System (2 disk drives) by adding a 3rd drive the capacity is increased to 500 vendor accounts and 2000 invoices a month. (3 drives requires 48K RAM).

Accounts Payable. 269-1554 .... 199.95

Real Estate IV

The 4th volume in our Real Estate series has just arrived. It contains the following programs:—

Ellwood Analysis — for analysing mortgages with known and unknown ratios.

Overall Rate — to calculate a maximum price to be paid for a property, to return maximum yield.

Ellwood Graphs — gives a graphical representation of equity yield. It will plot a graph with 5 different yield rates as a percentage of original price or value.

Sensitivity Analysis — helps you to see the effect of changes to different variables in most real estate situations i.e. mortgage, equity yield, cash flow, etc.
Requires 16K Level II

Real Estate IV. 269-1574 .... 59.95

CORRECTIONS:

Prices:— The price for the Accounts Receivable (269-1555) is $199.95 not $139.95 as stated in the previous newsletter. The price of Business Mailing List (269-1558) is $139.95. The price of the Accounts Receivable Series I (269-4554) for the Model II is $249.95 not $349.95 as previously advertised.

Accounts Payable (269-1554):

A point not mentioned in the manual for this program is that you require 48K of RAM when using the three disk option.

Rounding problems ; With early versions of Accounts Payable an intermittent rounding problem could occur. To fix this make the following change to the invoice program.

```
350 N#=VAL(IN$): W#=ABS(N#)*100: V$="": X=W#/D1#:
    W#=W#-X*D1#:
    V$=V$+CHR$(X-(N# < 0)*128): X=W#/D2#:
    W#=W#-X*D2#:V$=V$+CHR$(X):
    X=W#/D3#:W#=FIX(W#-X*D3#): V$=V$+CHR$(X)
     +CHR$(W#): RETURN
```

Accounts Receivable (269-1555):

The Accounts Receivable program has a similar problem. To correct this make the following changes.

"ENTRANS" program, add the following line.

```
1365 W#=FIX(W#)
```

A further problem occurs in the way that multiple page statements are printed. "PROCESS" program, change the following line:

```
2820 LPRINT LE$: POKE 16425,0: LF=7-PEEK (16425): GOSUB1590
```

FORTRAN (269-2201)

For those of you who are running Fortran on a single drive system, here is the way to do it without swapping the files on your diskettes.

Following steps 1-5 for the sample session beginning on Page 6 of the TRS-80 Fortran Users Manual.

STEP 6. Load and execute the program. To load the program into memory and execute it, put diskette #2 into the drive and type:

```
L80
```

This command tells TRSDOS to load and run LINK-80. When LINK-80 has been loaded, the command prompt (an asterisk) will appear:

```
*
```

Now remove disk #2 and re-insert disk #1. Type the file name of your program:

```
TEMP
```

As soon as the object file has been loaded into the correct memory locations the command prompt will reappear. Remove disk #1 and insert disk #2. Type the run command which is —G:

```
-G
```

LINK-80 will then search the system library to resolve any undefined references and execute the program. In this case, LINK-80 will not create a command file. Figure 3 shows a sample run.

STEP 7: Save the object code. The object file, once loaded by LINK-80, is in a form that can be executed by the TRS-80 computer. To save a copy of this file, type:

```
-N,-E
```

after the prompt. This pair of commands creates a command file that runs directly under TRSDOS. —N tells LINK-80 to save the file using the file extension /CMD. —E tells LINK-80 to exit to TRSDOS after searching the system library to satisfy any undefined globals.

APPLICATION SOFTWARE NOTE

If for any reason you hit the 'BREAK' key, DO NOT use 'RUN' to restart the program. 'RUN' can cause some applications programs to 'bomb'. Instead of 'RUN', use 'CONT'. The 'CONT' command allows the program to continue from the point at which the 'BREAK' key was pressed.

PROGRAMMING TIP
ARRAYS

Implicitly dimensioned arrays in Level II BASIC, Disk BASIC, and Model II BASIC are dimensioned to a maximum of eleven (11) elements (0-10) in a maximum of 3 dimensions.

SLOW LIST FUNCTION

If you find the scrolling speed during a LIST command is too fast, the following routine will slow it down.

```
10 FOR X = 16863 TO 16865
20 READ A . POKE X.A
30 NEXT X
40 FOR X = 32754 TO 32767
50 READ A : POKE X,A
60 NEXT X
70 DATA 195,242,127,58,128,56,31,208,197
80 DATA 1,0,32.205,96,0,193,201
```

You must protect 16 bytes of memory and you must run this program before loading your program to LIST. To slow down the LIST just press SHIFT. (For 16K TRS-80's).

SORT ROUTINE

One factor that can hamper string operations is the 'housekeeping' routine. This is a computer controlled function which allows the TRS-80 to recover previously used string space in memory. If you are sorting a large number of strings, this routine can take a significant portion of your sorting time.

We conducted a series of tests using three sort routines. The three routines were a standard bubble sort, a shell sort which exchanged strings and a shell sort which exchanged pointers. To make things as equal as possible, we used the same basic program, changing only the method of sorting. Here are some times we obtained (times in seconds):

Number of strings	Bubble Sort	Shell Sort	Shell Sort (with VARPTR)
10	2	3	5
50	36	17	62
100	181	50	165
500	14396	978	1164
1000	?????	9763	2560

The system used for this test was a 48K Model I, single drive, Disk BASIC and a line printer. Each sort was run only once so the times given are indicative but may not be representative. Each sort was used with randomly generated strings.

The skeleton program used was as follows:

```
10 CLS:CLEAR11000
20 DIMA$(500)
30 READ LAST
40 FOR I=1TOLAST
50 A$(I)=STRING$(RND(30),RND(26)+64)
55 NEXT I
60 QQ=QQ+1
70 PRINT" TEST IN PROGRESS - PASS ";QQ;" OF 5"
80 A1$=TIME$:PRINT A1$
400 FOR I=1TOLAST
410 PRINTA$(I)
420 NEXT
430 LPRINT" N= ";LAST,A1$,TIME$
440 GOTO30
450 DATA 5,25,50,250,500
```

The sort routines were entered beginning at line 100 . The following routines were used :-

BUBBLE SORT

```
100 M=LAST
110 FOR I=MTOM-1
120 FOR J=I+1TOM
130 IF A$(I)<=A$(J)THEN150
140 T$=A$(I):A$(I)=A$(J):A$(J)=T$
150 NEXT J
160 NEXT I
```

TRS-80 Classroom

PRINT USING:

The PRINT USING statement is a very powerful addition to your PRINT statements. With this statement you have the ability to format output using literals, alphanumeric variables and numeric variables.

We would like to examine one other aspect of the PRINT USING statement which is frequently misunderstood. Specifically we are referring to the ### modifer.

Consider the following:

```
10   A = 10.3: B = 10.4
20   PRINT USING "###"; A,B,A + B
```

Your output will be: 10 10 21

This is NOT an error on the part of the part of the TRS-80. The ### modifier uses a method of rounding called 4/5 rounding. When you ask the computer to print a number with a decimal part in a print field which contains fewer decimal positions than your argument, rounding is automatic. In the above example, both A and B are printed as 10. Out print field does not have a decimal column. When rounding occurs, the computer notes that both .3 and .4 are less than .5 and truncates them. However, when we add A and B, the result is 20.7. Now when the computer rounds the value, it notes that .7 is greater than .5 and rounds up, that is increases the one digit by 1, giving 21. In all three cases, the number printed on the display is the number CLOSEST to the actual value. If one of our values had been .5, the rules of 4/5 rounding would have added one to the ones column.

TRS-80 Progrmaming Consultants:

As you may know, our stores keep a list of programming consultants for those customers who require either a special program written or an existing program modified. If you are a programmer and would like your name placed on this list, please send the following details :—

Name or if trading as a company, your Company name, Address (P.O. Boxes are acceptable), Telephone number, a brief resume of your experience and background, the languages in which you propose to write for the TRS-80, an example of your work, written in BASIC.

Lower Case Instruction Modification:

The program which comes with the lower case modification (ULCBAS) needs a change to overcome a potential problem.

When a BASIC program uses a sufficient amount of memory, it writes over a call in ULCBAS. This causes the computer to freeze up or revert to MEMORY SIZE.

Only the cassette tape version is affected. To prevent this overwrite from occuring, use the following procedure to load ULCBAS :

```
>SYSTEM          <ENTER>
*?ULCBAS         <ENTER>
*?<BREAK>
>READY
>POKE 28829,242 :POKE 28830,125 <ENTER> ' (FOR 16K)
                      28830,189            (FOR 32K)
                      28830,253            (FOR 48K)
>READY           <ENTER>
>SYSTEM          <ENTER>
*?/              <ENTER>
```

```
170 A2$=TIME$
```

SHELL SORT (swapping strings)

```
100 N=LAST
110 M=N
120 M=INT(M/2)
130 IF M=0 THEN A2$=TIME$:GOTO 400
140 J=1:K=N-M
150 I=J
160 L=I+M
170 IF A$(I)<=A$(L)THEN 220
180 T$=A$(I):A$(I)=A$(L):A$(L)=T$
190 I=I-M
200 IF I<1THEN220
210 GOTO160
220 J=J+1
230 IF J>KTHEN120
240 GOTO150
```

SHELL SORT (swapping pointers)

For this sort, DELETE line 180 in the shell sort and insert the following lines:

```
177 FOR Z=0TO2
178 P=VARPTR(A$(I))+Z:IF P>32767THENP=-1*(65536-P)
179 W=VARPTR(A$(L))+Z:IFW>32767THENW=-1*(65536-W)
180 A1=PEEK(P):A2=PEEK(W)
181 IF Z=0ANDA1>LEN(A$(I))THEN200
182 IFZ=0ANDA2>LEN(A$(L))THEN200
183 POKE P,A2
184 POKEW,A1
185 NEXT Z
```

The effect of this last routine is to move string pointers instead of the strings themselves. This avoids the problem of creating new strings and avoids the 'housekeeping' routine.

The VARPTR routine is actually slower for up to 500 strings. Above 500 it is very much faster.

IMPORTANT NOTE FOR MODEL II USERS

After you have turned the power off to your Model II, wait AT LEAST 30 seconds before you turn it back on. Failure to do so could cause damage to the computer's circuitry.