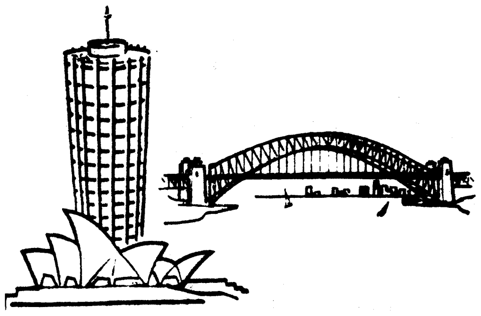

FREE Cassette
Next Issue
1Game/1BusinesS

Registration Sought for Posting
as a Publication CATEGORY 'B'
P.O. Box No. 105
MARRICKVILLE, N.S.W. 2204.

8th BIT VOL. 8 TANDY and SYSTEM 80 NEWSLETTER -- FEB 1980

Note the new title ,this is to cater for Dick Smith's new System 80 which is very similar to the Tandy TRS-80 ,and which will be generally available from about March,1980. Also note the new official style advertising layout to suit Australia Post.

" 8th. Bit" is to be the main medium in which the software exchange members can keep a tap on what is happening in this country. Its success will rely on the contribution from members.

It is proposed that "8th Bit" should contain the following-
1. It will give information on what is for sale and where it's for sale (locally).
2. Contributions from members.
3. Any information of general significance.

Programming hints
Useful application Inspirations

If you wish to offer programmes for sale they should be sent to the shop for evaluation and testing. Prices will be determined on complexity and usefulness.

The best medium for transporting programmes is by posting cassettes. The hassles of hand listing a programme without mistakes are great (presuming no one as yet has a printer), and then to re-enter by key board at the other end and to have to debug it might make a receiver wonder why he ever bothered at all especially in longer programmes.

It is an inherent problem with the limitations on memory in microcomputer that programmes will be badly documented and quite unreadable .Programmes for useful applications offered for sale should have documentation in the form of flow charts and comments accompanying the cassette so that if sold, buyers can modify them fairly easily and will be less likely to bounce back with problem bugs.

There are four or five Tandy owner groups meeting in Sydney , and other groups in other States and also in America, so we will attempt to pass the information around; since our members are all over Australia, New Zealand -and even in America.

PLEASE SEND IN YOUR PROBLEMS, AND HOPEFULLY THE ANSWERS ALSO IF YOU HAVE THEM.

This Newsletter will be issued every 3 months (4 times/year).In addition specials may be issued.

CONTENTS:

Dan Ellsweig U.S.A. Computronics
Another Print--LPrint LPrint--Print Programme. Does not work properly with TAB( ).

```
10 CLS
20 INPUT"Do you want HARD COPY---Y/N";A&
30 IF A&="y" GOSUB 10000
40 FOR X=1 To 10
50 PRINT"TEST"
60 NEXTX
70 IF A&="Y"GOSUB 1000
75 INPUT"TO CONTINUE PRESS ENTER";AA&
80 GO TO 10
90 END
1000 REM-SUBROUTINE TO CHANGE PRINT TO LPRINT.
1010 Y=0:Z=0
1020 FOR x=16422. TO16423:Y=Y+1:A(y)=PEEK(x):NEXTX
1030 FORX=16414 TO 16415:2=2+1:B(Z)=PEEK(X):NEXTX
1040 POKE 16422,B(Z-1):POKE 16423,B(Z)
1050 POKE 16414,A(Y-1):POKE16415,A(Y)
1060 RETURN
```

Computronics(RS Computer Services)
VIDEO DISPLAY onto LinePrinter Does not work with Graphic Symbols

```
1000 C=15360 :D=C+63:FOR A=1 TO15: FOR B=C TO D
1010 E=PEEK(B):LPRINT CHR&(E);
1020 NEXT B
1030 C=C+64:D=D+64
1040 LPRINT
1050 NEXT A
```

A. Sangster
Purchasing Goods from the U.S.A.

By chance I stumbled on a good test prior to sending money to U.S.A. Phone up the company concerned and say that an assoceate is flying over to the U.S.A. and would like to come in and see the product running. It is surprising how many companies reply by saying,"well they are not quite finished and ready for sale or demonstration" even after, in some cases , 6 to 12 months of advertising.

DOS 2.3 and 2.2 -
The 2.2. DOS gives faults when used with Random access files , thes has been overcome in 2.3 DOS . The fault appears to be the spoiling of the end or the RANDOM ACCESS FILES.
The RENUMBER facility is a beauty, however you can only renumbe from a nominated point, to the end of the programme. It is nicer to use REMODEL-PROLOAD from Racet , which enables you to renumber any chunk of programme you wish, and also to put it anywhere you wish.
Basic* is very handy, as you can get back to your programme in memory after having pressed reset.This assumes the power to memory did not disappear, because then everything disappears. Changing Random access file from 255Bytes-256Bytes seems a waste of time, although you can now evenly divide it exactly into equal chunks, thus saving one byte in every 255Bytes.Big deal!!! Perhaps we are missing the point. Anybody have any comments?
The system check is handy but not conclusive.
Unfortunately ,Tandy keep on changing the "top of memory" usage. This is most annoying. We put handy machine language subroutines up in the top or memory , and everytime Tandy DOS changes, we have to rewrite our machine routines. It seems that New Dos+ from Apparat is much better if you can run your programmes with it.

ON ERROR

Appendix B of the Level 11 Basic Manual lists all the errors that you can control.

eg.,100 ON ERROR GO TO 450

If at any time after this statement in your programme, one of the 23 errors listed in Appendix B occurs ,you will go to line 450. In line 450 you can send the programme back to anywhere you wish.

Without this statement ,you EXIT the programme completely and you have to start again. (Most confusing for customers trying to use your programme)

Every ON ERROR --should have an accompanying ---RESUME.

eg , 450 RESUME 500
will continue from 500

eg., 450 RESUME NEXT
will continue from line following where error occured

eg., 450 RESUME
will continue executing with the statement at which the error occured.

If for example further down in the programme yor wished to automatically save on cassette if no disc is present, simply insert another on error.

```
1000 ON ERROR GOTO 2000
1010 OPEN "O",1,"Disc file"

2000 RESUME 2010
2010 PRINT#1,"Cassette File"
```

Finally you can selectively trap errors, by checking the value of the error

```
450 IF ERR=2 THEN
```

Don't Pull Out Cassette Recorder Jacks

from (Computronics)
Box 149 New City,
New York,10956

Press Fast Forward OR Rewind
Type PRINT "1,A (Enter)
Tape will move approx. 40 units(according to the cassette tape counter)
Any Programme or Data in memory will not be affected.
To stop in less than 40 units press cassette STOP button.
To move more than 40;
Type PRINT "1,A:PRINT"1,A.(ie. move 80 units

Printer Control

LPRINT CHR$(11) or LPRINT CHR$(12)
moves paper to top of page.
PRINT PEEK (16424) [ENTER|
gives 67 ie. 67-1=66 lines/page.
For example if you have printed 10 lines then
LPRINT CHR$(11) moves (66-10)=56 spaces .
POKE 16424,7 then LPRINT CHR$(11) moves the balance of 6 lines to next address/statement etc.
PRINT PEEK 16425 shows how many lines have been used in a page.

16 Digit Square Root accuracy

```
5 REM-Routine to find square root of a number
10 REM-with at least 16-Digit accuracy
7  REM-by
8  REM-Houard Y Gosman
10 CLS:INPUT"ENTER NUMBER":X"
20 Y"=SQR(X")
30 PRINT Y#
40 Y"=(X#/Y"+Y"/Z
50 PRINTY"
60 IF Y"=A GOTO 80
70 A"=Y" :GOTO 40
80 PRINT :PRINT :PRINT"The square root of":X";"is";Y":PRINT
90 INPUT"To Find Sq root of another number Press Enter";X
100 GOTO 10
```

WORD PROCESSING SIMULATION

```
10 CLEAR1000
20 CLS
30 A$=1NKEY$ : IFA$=" "GOTO 30
```

Alan Smith (Comments)

1. Anyone wanting to buy the Microsoft Macro Assembler / Linking Loader / Text Editor for TRS-80 can get it from:-

Lifeboat Associates,
2248 Broadway,
New York,N.Y.,10024.

They sell it for $US80(eighty dollars),which is at least fifty dollars cheaper than anywhere else in the U.S.(Tandy in Australia say they'll be marketing it here for about $130 next year). the package is superb for those wanting to do some serios Assembler work. They charges me an extra $US10 for airmail(insured) postage, but that was more than reasonable as they had $18 worth of stamps on the parcel. The documentation is pretty professional, although not orientated to beginners; manual includes all the documentation for their FORTRAN package as well, but the FORTRAN software is another $80. ($150for both FORTRAN and the Macro Assembler).....Both packages require a DOS system with a minimum of 32K RAM.

2. Other software I've bought recently (mostly from the U.S.):-

* Scott Adams' "ADVENTURE" games - superb
* William Engel's "Stimulating Simulations" (from Computerland) - a waste of money; if only the programs were as sophisticated as the documentation!
* Small System Software's "RSM-2" monitor - worth the money for the Disassembler alone; very useful
* Small System Software's "DCV" for executing just about any machine-lantguage program from disk, even if it loads in the same area as DOS, is easily worth the $10; with it you can put (e.g.) the Editor/Assembler onto disk and load it in seconds instead of minutes ..... with a bit of fiddling, I even managed to get Microchess 1.5 onto disk.

Just a thought for those who have wondered what sort of hassles there are in ordering software from the U.S.-in my experience ,virtually none at all.One thing ,however-if you wan the goods to arrive in 3 or 4 weeks instead of 3 or 4 months ,then emphasize AIRMAIL in your letter ,preferably in two-inch-high letters, and enclose an extra amount to cover it (usually $US2 O.K. unless it's a big package).

3. Contrary to the club's experience, I was not impressed with the Dick Smith "TRS-80 Compatible" disk drives.I had a lot of trouble getting accurate technical information from the store personnel, who seemed more interested in emphasising how cheap the drive was compared with Tandy's (I already had one Tandy drive, and was looking for asecond.)The drive gave me more disk errors in any half-hour period than I had had in months of using the Tandy drive, and I was told that was because I was trying ro read disks written on my Tandy drive!!!!...I gave up,got my money back, and have had no problems since .

4. Have sometimes found it annoying that BASIC forces the cursor to the start of the next line after use of an INPUT statement ,since precludes outputting anything to the screen onthe same line as the typed input. The following gets round it by effectively "backspacing" the cursor the required number of positions.....

Level 11-

```
PRINT@PEEK(16416)+PEEK(16417)*256-15360-nn
```

where nn is the no. of positions you want to "backspace"

TRSDOS

The Level 11 code will ,of course ,work,but you can save space if you want to do this in a number of spots in the program by using:-

```
DEFFNPP(BS)=PEEK(16416)+PEEK(16417)*256-15360-BS
```

at the start of the program, then each time using:-

```
PRINT@FNPP(nn)
```

where nn is the no. of positions you want to "backspace"

Of course all of the above is only necessary if the line you're "INPUTTING" on is variable; if it was always the same line, you would of course position the cursor simply by doing a PRINT@ with an absolute value.

COMMENTS Mr Smith must have been lucky importing softward, as approx 1 in 3 packages are scrutinized by Customs (Redfern Mail Exchange) and Sales Tax of 15% and Duty of 2-35% are then payable. Also you may have problems with magnetic distortion in the transport. Pitt St. Microcomputers have overcome these problems by obtaining licences for software and copying from Master Copies held in Australia.

PATTERN FUN.

```
10 REM 1. RUN THE PROGRAM
20 REM 2. PRESS ANY COMBINATION OF KEYS
30 REM 3. THIS WILL START A PATTERN
40 REM 4. PRESS ANY OTHER KEY TO CHANGE THE PATTERN
50 POKE 16396,23:CLS
60 FOR N = 14336 TO 15360
70 PRINT (CHR$(PEEK(N)));" ";
80 NEXT N: GOTO 30
```

VIDEO SCREEN ON-LINE PRINTER (Do not use with graphic characters)

```
5 CLEAR 200
10 FOR X = 15360 TO 16359 STEP 64: A$=" "
20 FOR Y = 0 TO 63: A$=A$+CHR$(PEEK(X+Y))
30 NEXT Y: LPRINT A$: NEXT X: END
```

MERGE TWO PROGRAMS USING CLOAD

1. First progrsam is in computer memory.
2. Second program is on cassette and has line numbers larger than any used in the first program
3. PRINT PEEK (16633), PEEK(16634)
4. If the contents of 16633 is 2 or greater,
   POKE 16548,PEEK(16633)-2
   POKE 16549,PEEK(16634) then go to 6
5. If the contents of 16633 is zero or 1,
   POKE 16548,PEEK(16633)-254 and
   POKE 16549,PEEK(16634)-1
6. CLOAD second program then
   POKE 16548,233
   POKE 16549,66
7. LIST, RUN or SAVE the merged program.

DETECTING IMPROPER SHIFT KEY USEAGE (Courtesy Computronics)
If you accidentally type a letter of the alphabet with your finger on the shift key, it will appear on the TV screen exactly the same as a normal letter. However, to the computer it is different. Thus you may have typed SMITH J. (with the shift key down for the letter 'I'). When you try and find SMITH J. (typed correctly without shift key), the computer cannot find it.
Here is a little routine to find such mistakes.
F$ is the name of the sequential file on disk that you wish to check.

```
5 CLS: CLEAR 500: INPUT"FILESPEC";F$
10 OPEN "I",1,F$
15 PRINT@219,"CHECKING"
20 IF EOF(1)PRINT"REACHED END OF FILE":CLOSE:GOTO 100
30 LINEINPUT#1,B$:C%=1: POKE 14305,0
40 FOR I% = 1 TO LEN(B$): Q$ = MID$(B$,I,1): POKE 14305,0
50 IF  (ASC(Q$).95) AN (ASC(Q$),128) THN 70
60 NEXT I% : IF C% = 0 THEN PRINT: GOTO 20 ELSE 110
70 IF C% = 1 THEN PRINT B$: C% = 0
80 PRINT Q$;" ";: POKE 14305,0
90 D% = 1:: GOTO 60
100 IF D% = 1 PRINT "PLEASE NOTE SHIFT KEY ERRORS": END
110 PRINT"NO LOWER CASE ERRORS FOUND"
```

CREATIVE GRAPHICS by Dennis Meixsell

```
5  CLS
16 POKE C,32
15 A$=1NKEY$
20 PRINTA$;
30 A=PEEK(16417)
40 B=PEEK(16416)
50 C=A*256+B
60 POKE C,42
70 FOR T=1 TO201:NEXT
80 GOTO 10
```

Graham Jackson

Has tried the following to overcome power fluctuation problems when the Refrigerator switches on.

Line Filter;
SNZ3326-01
6A MaxLoad (plenty)
$25.20 + Sales Tax
From Westinghouse
It is a M.O.V. and a low pass filter in a screened case.90% successful.

A. Sangster (Sorting etc.)

The problem with moving large numbers of strings,(the computer keeps taking 10 minute rest breaks) has been overcome.The solution was not to move strings, but to nominate a number to each string which represents its correct sorted position. Thus sorting or inserting are carried out by means of changing the nominated number associated with the string. In Basic this is shown below

Alphabetising without replacement by Howard y Gosman

```
10 CLEAR 10000:DIM A$(50),N$(50):I=1:CLS
20 INPUT"NAME";N$(I):IF N$(I)="9999" GOTO 50
30 I=I+1
40 N=N+1:GOTO 20
50 FOR I=1 TO N:A$(I)=N$(I):NEXT:CLS:K=0
60 I=1:FOR J=2 TO N:IF A$(I)<A$(J) GOTO 80
70 I=J
80 NEXT:IF N$(I)="9999" GOTO 90   ELSE PRINT N$(I)
90 A$(I)="ZZ":K=K+1:IF K=N GOTO 100  ELSE GOTO 60
100 PRINT:INPUT"ADD NAME TYPE 1..END TYPE 0";Q
110 IF Q=0 PRINT"END OF PROGRAMME":END
120 N=N+1:I=N:GOTO 40
```

Mr D. Bates :- has converted this into machine language giving 1000 Items "sorted" in 12 seconds.

However the in memory machine language "sort" from RACET is 1000 Items in 9 seconds , and is very nicely documented at about $20 . Contact PITT ST MICROS for these programmes.

Machine Language Book

We have found TRS-80 Assembly-Language Programming by William Barden Jr.,available from Tandy stores to be an excellent book on Machine Language Programming .Among the topics covered are the instruction set of the Z-80,it's different addressing modes ,movement of data within memory, arithmetic and compare operations,logical operations,including shifting and bit-wise operations, and of course assembly language programming.

Courtesy Tandy Assembly U.S.A.

Double Precision

When specifying a double precision constant follow it with a D(double precilion exponent) ie.,without the D

```
10 A#=12.34
20 PRINTA#
RUN
```

gives 12.34000015258789

However

```
10 A#=12.34D
20 PRINTA#
RUN
```

Gives 12.34

Double Sized Characters

May be obtained by pressing Shift key and Right Pointing Arrow or by using CHR$(23).

Print @ 64 will print at location 64 on the screen normally, and PRINT @ 65 at location 65.
However with double sized characters
Print @ 64 will print at location 64 & 65
Print @ 65 will not print.

BACKUP.
It is most important to BACKUP data, particularly business records,using the BACKUP command. Beware because some business systems place some of the data on the program disk as well as other data on the data disk. Unfortunately in these cases you have to BACKUP both disks. Also it means the program disk must remain unprotected ... a little dangerous!!!

TANDY SOFTWARE.
Below are listed some of the software for which there is a current addendum. You might call in your local Tandy store and see if you need any of these corrections.

In-memory Information System Version I
Payroll Program - Level I - II Conversion
Statistical Analysis (26-1703)
Personal Finance Level I - II Conversion
Math I Level I - II Conversion
Algebra I Level I- II Conversion
Disk Mailing List

COMPACT AND SPEED UP YOUR INPUT/OUTPUT TO CASSETTE ON LEVEL II
Placing numeric information onto cassette using a string B$ (aproximately 230 characters at a time).

```
10 REM ENCODE AND OUTPUT DATA
20 CLEAR 1000:B$=" ":A$="/"
30 REM "N" IS THE NUMBER OF DATA ELEMENTS
50 FOR I = 1 TO N
60 IF LEN(B$+STR$(D(1)+A$)|230 THEN PRINT#-1,N,B$:B$=" "
70 B$=B$+STR$(D(1))+A$
80 NEXT:IF B$ [| " " THEN PRINT#-1,,N,B$
```

The program above is to output numbers onto cassette. First the program writes onto cassette the number of numbers that wil be output. Then the program converts the numeric data contined in array D to string form, and saves the result in string variable B$. When B$ is about 230 characters long (or the last number has been converted) it writes the string to cassette.. Note that the "/" (slash) is placed between the numbers in B$.

```
100 REM INPUT AND DECODE DATA
110 A$="/": B$=" ": J = 1
120 REM 'NN' IS THE NUMBER OF NUMBERS TO BE INPUT
140 INPUT#-1,NN,B$: N= LEN(B$): LF = 1
150 FOR I = 1 TO N
160 IF MID$(B$,I,1) = A$ THEN NC = I - LF ELSE GOTO 200
170 D(J) = VAL(MID$(B$,LF,NC))
180 LF = I + 1: J = J + 1
190 REM CHECK FOR LAST NUMBER
200 IF J | NN THEN220
210 NEXTI : GOTO 140
220 REM YOUR NEXT STATEMENT
```

The second program is used to input data from cassette that has been encoded and output by the first program. This input program reads the number of numbers that are on tape (line no. 130) so that it will know when all data has been read. Variable B$ contains the string data read from tape. Array D contains the data converted to numeric form.. Notice that line 200 branches out of the input loop when all the numbers have been converted to numeric form and stored in array D.

Comments From Geoffrey B. Kent,
These comments are the result of the trouble I've experienced with the RS-232-C Board.

Initially I had trouble with my chips.First,#2-8 went bad - one of the ones I installed in the interface. Then, some months later during a Test 1/CMD (from TRS-DOS) I found that 2-16 was bad also.I thought something wasn't proper if two RAM's went bad, but replaced 2-16 and used the computer successfully anyway.

However, later-after 3 continuous hours with no problems, I began getting numerous Syntax errors. As a check I ran Test 1/CMD to see if any more RAM's had gone. The test was positive.In fact, it told me that the entire bank that came with the interface, 2-9 through 2-16, were all bad!!!

I immediately called Mal Williams and told him about it. I asked him if it were possible for the RS-232-C board to cause malfunctions of this order and he said yes. I took the board out of the interface, tested again, and all is well. I suspect the fault lies in the connector instead of the board itself, but I'm sending it, interface and all down to Mal to have him solder each and every connection from the interface to the connector and from the connector to the board. If that doesn't solve it, I don't know what will.

Up until this time, I thought I had solved the connector problem in this way: I removed the connector itself entirely from the printed circuit board in the interface and, using a small pointed object (a straight pin), I very carefully bent back each spring contact further than it origionally was. I reasoned that perhaps a bit more pressure would help. I did this to both sides. I also cleaned everything with spirits.

After I put it back together, the board seemed to work quite well----until now. My interface is not subject to any vibration (other than when I bang the desk with my knee) so how those contacts are becoming inoperative, I don't know. In any case, I'm sending the interface, board and all, to Sydney to get the board replaced, if necessary, and all the contacts soldered,as I said before.

Other users having trouble with their 232-C's may wish to try my old method before sending everything to Sydney.

Another comment: purely operational. Anyone having a disc drive hooked up as well as a tape deck must get into the habit of powering up, getting into Disc Basic and then--before anything else--Enter CMD"T". I know it's something very simple, but I've caught myself that many times, it's not funny.

Here's something else that other's may find useful. You remember I've been trying to unite a program to keep information on machines for a week at a time. I origionally wanted to keep 12 weeks per machine so I could refer back, etc. and then print it out with my line printer. I had no difficulty (or very little in comparison) in identifying each machine and placing it's number into a filespace thereby creating seperate files for each machine. Where I came totally up against a wall was finding a particular machine file, but a particular week as well.

FOR-NEXT loops may work, as you suggested, but I'm afraid they're still way beyond me. I have, however successfully solved my problem, by giving each file-spec the machine number and the first four digits of the "week ending". Don't think, though, that Iv'e got a list of 96 "OPEN filespec" statements. hope. Instead, I have only one. And that is good for any number of machine (or machines) and any "week ending".

I've happened to find on page 7-35 of the TRS-DOS manual, a method that allows the operator to supply the mode, buffer#, filespec and drive# while the program is running. The result? Only one "OPEN" statement. (Actually I included the drive specification in the filespec.) My filespecs now look like: W(week ending ie. 2411)M(machine# ie.24):1 or,

```
W2411M24:1
```

This gives each machine for each week, a separate file. Not bad, eh?

The program is fully operational now and if you want a "listing" let me know. I won't be able to do it for a while, of course, as my interface is going to Sydney.

I hope these comments have some value to you.

PITT ST. MICROCOMPUTERS Phone: 569 8228(2-6 pm)
Box A 344, Sydney South P.O. A.H. 560 6677
Pitt St., Sydney 2000

SOFTWARE

* = Machine Language

Microcomputer Applications
T.S.E.

Level II Cassette

Programme Title	
Cribbage	9-00
Checkers	8-00
Fastgammon*	16-00
Galactic Blockade	9-00
Pork Barrell	11-00
Round The Horn	11-00
Santa Paravia	9-00
Taipan	11-00
Ten Pin Bowling	9-00
Kentucky Derby	7-00
Tarot Cards	11-00
Accounts Receivable	26-00
General Accounting	16-00
Personal Finance	11-00
Mortgage Calculator	9-00
Inventory Control	13-00
Basic Statistics	21-00
Ham Radio	11-00
Electronics Assistant	11-00

TORANTO & ASSOC. CONVERSION OF OSBORNE & ASSOC. COMPLETE BUSINESS SYSTEM.

Written to conform exactly with the two Osborne Textbooks describing their successful Wang Computer System.

Accounts Receivable	150-00
Account Payable	150-00
General Ledger	150-00

SMALL SYSTEM SOFTWARE

Electric Pencil)*Cassette	105-00
Word Processor) Disc	155-00
Air Raid*	16-00
Barricade*	16-00
RSM-2 Monitor*	28-50
Basic IP Converts LII to LI *	26-00

PITT ST. CENTRE

Level II Games(Startrek etc.)	12-00
Level II Basic Instruction Course	32-00
Keno & Game of Life	14-00

Available DISC.

Final Approach & Space Battle	26-00
Advanced Personal Finance	26-00
Amateur Radio System	26-00
Inventory 2.2	60-00

10 Verbatim Discs Incl S.T. 46-00

Programme Title	
Adventure(Land) *	16-00
Adventure (Pirates)*	16-00
Amaz'in Mazes	9-00
Atlantic Balloon Race	11-00
Backgammon	9-00
Baseball	9-00
Bingo	7-00
Batter Up	7-00
Breakaway	7-00
Bridge Challenger	16-00
Dogstar	11-00
End Zone	9-00
Hamurabi	7-00
Hangman	7-00
Kamikaze	9-00
Lunar Lander	9-00
Mastermind*	9-00
Mission Impossible*	16-00
Othello	7-00
Pentominoes	11-00
Remainder	9-00
Robots	7-00
Safari	9-00
Sink 'um	7-00
Space Battle	16-00
Startrek III 3.3	16-00
Time Bomb	6-00
Treasure Hunt	9-00
Tycoon	9-00
X-Wing Figher II	9-00
Warfare!	9-00
Appointment Log	11-00
Astronomy 1	9-00
Biorythms	6-00
Calendar Functions	9-00
Calculator	4-00
Cash Register	11-00
Complex Mathematics	9-00
ESP Tester	7-00
File Handling	9-00
Finance (Loans & Invest.)	9-00
Fourier Transforms	9-00
Home Finance	11-00
Inventory (Mod)	21-00
Inventory (FP)	25-00
Keyboard 80*	11-00
KVP Extender*	26-00
Linear Programming	9-00
Micro text Editor	11-00
Math Drill	6-00
Numerical Integration	9-00
Pre Flight	21-00
Renumber *4,16,32, OR 48	15-00
Root Finder	9-00

16K Memory Kit Incl S.T. 119-50

Continued Overleaf..

****

TAILOR MADE SOFTWARE FOR TANDY:-CASSETTE AND MINI DISK TO 10 MEGABYTE DISK

****

Inventory 2.3	80-00
KVP Extender	30-00
Mail List II	99-00
Newidos +	99-00
ST-80 D	80-00
Text - 80	60-00
Level I Relocated in Level II	26-00

BOOKS

Osborne McGrahill

ACCTS REC/ACCTS PAY	20-00
GEN LEDGER	20-00

Business System Price Includes Books

Pennington

TRS-80 DISK & OTHER MYSTERIES	25-00

Statistics II & III	11-00
Star Finder	9-00
Trig Package	9-00
8800-Z80 Conversion	16-00
RACET	
REMODEL	25-00
REMODEL & PROLOAD	35-00
GSF	25-00
DOSORT	35-00
COPSYS	15-00
TIMSER	15-00
COMPROC	20-00
SMALL SYSTEM SOFTWARE	
DCV-1	10-00
TORANTO & ASSOC	
CASH JOURNAL	50-00
INVOICING	100-00

-:- THIS IS A PAID ADVERTISEMENT -:-

FOR THE SERIOUS DISK SYSTEM USER - DISK OPERATING SYSTEM SORT MERGE "DOSORT"

SORT/MERGE multi-diskette sequential files. DOSORT utilizes a control program written in BASIC with very efficient machine language routines for the time-critical tasks of sorting and data comparisons. An extended GSF is included which is used as an interface between the BASIC and machine language programs. The resulting system is both a versatile and effective sort merge package for the TRS-80 system.

- DOSORT can process data files that are read or written using standard TRS-80 input/output commands.
- DOSORT sorts input files accouding to user specified sequences utilizing multiple ascending or descending sort keys. User exits are provided during initial input and final output processing allowing specialized processing.
- TWO MODES of sort variable specification may be used - multi-mixed-array mode and character string mode.
- MULTI-VOLUME (diskette) files may be sorted or merged on a minimum two-disk 32K system. System performance is improved with a 48K or three- or four-disk system.
- PRESORTED files may be included in a given sort/merge application. Data files may be spread over several diskettes.
- DOSORT is self-instructing - supported by comprehensive user documentation. Specification of sort parameters has been designed to be easy - even for the inexperienced user.

ORDER 32 OR 48K VERSION OF TS26C AT $34.95

GENERALIZED SUBROUTINE FACILITY "GSF"

- A SYSTEM for incorporating multiple machine language programs in a unified structure that provides easy access by TRS-80 BASIC users. Easily expanded by the user to include additional functions. Instructions furnished describing conventions used.
- GSF INCLUDES eighteen machine language subroutines providing the TRS-80 with much greater flexibility and extended capabilities.

SORT 1000-ELEMENT ARRAYS IN 9 SECONDS!!! Two in-core sort subroutines are provided for sorting data in memory. The first subroutine sorts records consisting of corresponding elements of up to 15 arrays (mixed string, floating point, and integer) using multiple ascending or descending sort keys. The second subroutine sorts records consisting of elements of a character string array using multiple substrings as ascending or descending sort keys. These sort systems are very fast, versatile, and easy to use. Sort times average 30 TIMES faster than the fastest BASIC sort routines.

READ AND WRITE ARRAYS TO TAPE. Two subroutines are provided to read and write data to cassette tape. This provides the user with the capability of reading or writing an entire array or screen image with one command. No leaders are written between data items. This significantly reduces read/write times making data tapes much more practical. Data validity checking is performed to ensure correct data is read, thus eliminating many data input/output errors.

COMPRESS AND UNCOMPRESS DATA. Two subroutines are provided which compress data in memory by removing repeated characters with the ability to uncompress the data in the minimum space possible with subsequent regeneration. This, coupled with the Read and Write Tape Data Subroutines, provides an efficient method of data storage and retrieval.

MOVE ARRAYS IN MEMORY. This subroutine moves data from one location in memory to another. This can be used to rapidly set one array of data equal to another or to move data into protected memory. The latter option provides a "common" area that can be passed from one BASIC program to another.

DUPLICATE MEMORY. Two subroutines are provided to duplicate a byte in memory. This is useful for setting arrays to zero or rapidly placing rows or columns of repeated characters on the screen.

FAST HORIZONTAL AND VERTICAL LINES. This two subroutines draw graphic lines of any length and location on the screen. These routines dramatically decrease times necessary to generate graphical displays containing lines.
DISPLAY SCREEN CONTROL. Five subroutines are provided for scrolling the screen up, down, left, right, and for generating inverse graphic displays. This can add impact to screen displays.

WORD PROCESSING - REVERSING LOWER & UPPER CASE WITH THE SHIFT KEY

Some people who use the TRS-80 for word processing may have had difficulties using the keyboard becuase the upper-lower case arrangement is opposite of the standard typewriter keyboard arrangement. Here is a simple program you may use to everse upper and lower case for letters and arrows. If you havean upper-lower case printer, this program will allow you to use it more effectively.

```
10 REM ** REVERSE KEYBOARD (REVKEY) **
20 A$=INKEY$:IFA$=""THEN 20 ELSE 30
30 A = ASC(A$)
40 IF A [ 65 THEN A = A: GOTO 70
50 IF A [ 91 THEN A  = A + 32: GOTO 70
60 IF A [ 128 THEN A = A - 32
70 LPRINT CHR$(A);
80 GOTO 20
```

Your video display, of course, will still be in upper case.

TRS-80 Users Group Newsletter

by Micro Computer Consultants
Florida 32019

Some ideas on attractrive menu borders,
Programme listing called "Patterns"

```
10 CLS
20 FOR N=129TO191
30 FORZ=1TO500:NEXTZ
40 PRINT"ASCII CODE#";N
50 FORX=1TO64
60 PRINTCHR$(N);
70 NEXTX
80 NEXTN
90 END
180 CLS
190  FOR X=0 TO 127
194 RESET(X,16)
196 RESET(X,46)
200 SET(X,0)
205 SET(X,15)
210 SET(X,47)
220 NEXTX
230 FOR Y=0TO47
240 SET(0,Y)
245 SET(1,Y)
250 SET(127,Y)
255 SET(126,Y)
260 NEXTY
270 PRINT@655,"PRESS BREAK KEY TO END DISPLAY";
280 GOTO 280
290 CLEAR 500                                   ;+CHR$(24):NEXT
300  CLS:FORI=1TO15:PRINTTAB(0)CHR$(150);:PRINTTAB(62)CHR$(154)
310 PRINT@0,STRING$(62,179);:PRINT@321,STRING$(61,131);:PRINT@960,STRING$(63,131);
320 PRINT@655,"PRESS BREAK KEY TO END DISPLAY";
330 GOTO330
```

LEVEL II MANUAL ERROR

On page 8/8 of some of the Level II Reference Manuals, the follwing appears near the middle of the page:

```
   POKE 16526,0: POKE 16527,208
      (208 DECIMAL = 7D HEX)
```

This should read:

```
   POKE 16526,0: POKE 16527,125
      (125 DECIMAL = 7D HEX)
```

MODIFYING MODEL I SOFTWARE TO MODEL II

Model I BASIC programs may be transferred to Model II, however they will need some modification before they will run properly. The main items which will require attention are:

1. 'IF-THEN' Statements - Model II requires the use of 'THEN' where it is optional on Model I.
2. 'PRINT@' Statements - Most 'PRINT@' statements will need to be changed to properly format screens on the larger Model II video.
3. Drive Specifications - At least for a while, most Model II users will be running single disk systems. This requires that you change any drive specs which refer to drives other than drive 0(zero).
4. All references to PEEK, POKE, INP or OUT must be removed. These commands are not needed in Model II BASIC. Instead, TRSDOS supervisor calls are available which will allow you to pform the same functions.

Other changes can be made which will enhance your programs and take better advantage of Model II's capabilities, but the four mentioned above are required if your software is to run properly.

Transfer of programs between Model I and Model II requires a special cable and terminator plug. These items are described in the Model II manual.

**NEXT ISSUE

FOLLOWING REQUESTS FROM NEWSLETTER SUBSCRIBERS OUR NEXT ISSUE WILL DESCRIBE AS MANY PROGRAMMES AS WE CAN FIND. IF YOU CAN ASSIST US ,PLEASE SEND US COMMENTS.IT WILL BE ISSUED AS SOON AS POSSIBLE ,PREFFERABLY BEFORE THE END OF MARCH POSTAL INCREASE.

GENERAL

IT MAY INTEREST SOME THAT THIS WHOLE NEWSLETTER WAS TYPED USING ** THE ELECTRIC PENCIL ** DIABLO DAISY WHEEL AND TELETYPE 43 DOT MATRIX.