NATIONAL TRS-80 & GENIE USER GROUP

Newsletter

V4 #7

JANUARY 1983

CONTENTS

2. Information on the Group
3. Editorial
3. Guidelines for contributors
4. Members' letters
10. London (South West) Sub-group programme
11. Problem section
12. Loading system programs through the second cassette port on the Genie
13. LPRINTing CHR$(0), (11) & (12) revisited
14. A simple monitor program
19. Cutting down on garbage
20. Passing parameters from DOS
21. RS-232 interface by Eric Hartley
22. In reply to C.L. Lovell - Page 4, October 1982 issue
23. Prestel 300 baud interface
25. Communications conference
25. Microline 82A line feed switch select
26. Using Prestel at 300 baud
27. Using an ex British Telecom modem type 2A
31. A use for the expansion interface relay
32. New hardware and software from Tandy
32. Using macros
33. Slower and faster
34. Milton Keynes workshop report

NATGUG

INFORMATION ON THE GROUP

Membership of the Group is by subscription to the Newsletter, which is published monthly. Membership details are obtainable from the Group Secretary. Membership of the Group is open to anyone with an interest in the TRS-80 range of microcomputers, and compatible systems such as the Video Genie.

Details of the Group accounts and the constitution of the Group are obtainable from the Secretary.

Members requiring assistance with problems related to the TRS-80/Video Genie may call the Secretary. An attempt will be made to put them in touch with a member who can help with the problem.

Workshops are arranged from time to time in various parts of the country.

Sub-Groups exist in many areas. A list is provided at the back of the Newsletter from time to time. Sub-Groups also exist for medical and educational users.

The Group runs 80-NET, a computerised bulletin board service that is available to all users of microcomputers.

The Group maintains three software libraries (Models I, II and III) which are free to members. library lists are obtainable from the Secretary. The medical and educational sub-groups maintain their own libraries.

A membership list is available from the Secretary on disk.

Back numbers of the Newsletter are available.

Please send all contributions for the Newsletter to the Editor.

Chairman and Newsletter Editor:-

Leon Heller,
8 Morris Walk,
Newport Pagnell,
Bucks. MK16 8QD.
Tel: (0908) 613004

Secretary and Newsletter Publisher:-

Brian Pain,
24 Oxford Street,
Stony Stratford,
Milton Keynes.
Tel: (0908) 564271

Librarians:-

Model I: Leighton Davies, 105 Caerau Road, Caerau, Glamorgan. Tel: (0656) 738337
Model II: Jim Hutton, 25 High Street, Stroud, Glos. Tel: (045 36) 4423
Model III: Nick Clay, 6 Rectory Lane, Byfleet, Surrey KT14 7LL. Tel: (09323) 42348

EDITORIAL

I am sorry for the lateness of this issue. Pressure of work and the January workshop were responsible.

The workshop was extremely successful, with a total of over 100 members attending over the three days. Everyone seemed to enjoy themselves, and it would appear that although membership of the Group is falling off, those members we do have are extremely enthousiastic.

A full description of the proceedings appears elsewhere in this issue, but one thing should be mentioned here, and that is the new name of the Group. It was decided at the AGM to henceforth call the Group the National TRS-80 & Genie Users Group, or NATGUG. John Carne and his wife have designed a new front sheet for the NATGUG newsletter; I hope you like it.

From now on I would prefer contributions to be sent in printed form, if possible. A large proportion of members now have printers, and it will save me a lot of work. If you do not have a printer, you may still send material on disk or cassette, of course, or handwritten, if you must.

Leon Heller

GUIDELINES FOR CONTRIBUTORS

1. If possible, send material in printed form.
2. Scripsit or Pencil disks/cassettes are acceptable.
3. Ensure your ribbon is in reasonable condition.
4. Printer output should be on A4 paper if posssible.
5. If you send in handwritten material, write legibly.
6. Do not fold the sheets when posting them. Use a large envelope and keep them flat.

Leon Heller

BOOKS RELATING TO TRS-80 MACHINES FROM McGRAW PUBLISHERS

AUTHOR	TITLE
ANBARLIAN	TRS-80 SPREADSHEETING
DWYER	BASIC AND THE TRS-80: A Structured Approach to Creative Programming (Byte)
KATER	TRS-80 GRAPHICS FOR THE MODEL I AND MODEL III (BYTE)
KATER	TRS-80 WORD PROCESSING WITH SCRIPSIT (BYTE)
POOLE	PRACTICAL BASIC PROGRAMS: TRS-80 LEVEL II EDITION (OSBORNE)
SKIER	TOP-DOWN BASIC FOR THE TRS-80 COLOUR COMPUTER (BYTE)

MEMBERS' LETTERS

From D.N. Evans

Referring to our telephone conversation on Maxi Manager. There have been a number of problems with this program. It is very comprehensive in some ways, but in other ways it has had a sad lack of facilities. It is also very slow, with slow string handling if one has a lot of data to enter.

One very bad fault with it has been, that it has been impossible to change a report file quickly if mistakes are made in formating. It has by experience been far quicker to develop one with a wordprocessor, than use the software written in the program.

One other severe limitation, agravated by the above point is that if discs are taken out before files are closed, (and this can all too easily happen, due to frustration and the length that sub-programs take to load), then the key files that are set up to access data are non-usable because of them not being closed, and eventually if this happens to all of the key files (only 4 available), then your data becomes inaccessible.

Realizing that this has been happening, the programmers have come to the resuce with a Utility program called MMUP-1, which is a package of 3 programs to perform the following functions.

> RECOVERY: A Key file restoration program. EXTRACT: To select data from an existing Database. MERGE: To transfer data from the discs produced by EXTRACT, to normal Maxi Manager data files.

The frustrations of Maxi Manager lead me to invest in the new Data Writer program, which I find exellent. Data Entry is both easy to set up, and extremely fast. I have in quite a short time written a 4000 item inventory, and also a 4000 item price list.

With the purchase of this new program, then the MMUP-1 which I originally bought for Maxi Manager, becomes redundant, and is available to anyone having these sort of problems, for the sum of £25.00.

Due to the natural progression from cassette to Disk, I also have a Scripsit Cassette word processor, available for £20.00 if anyone requires it.

D.N. Evans

90 Belvedere Road
Bexleyheath Kent

13th. January 1983

From Steve Taylor

I have just read Marshall Barry's letter in the November '82 Newsletter. While understanding both his anger born of frustration and his reticence due to "shyness or pride" I would like to redress the balance somewhat.

I too have purchased from "within the group". If I had purchased something from Boots or Tandy then I would expect a "fully finished" product, designed for instant easy use ("unwrap, plug in and watch it go"). In fact what I very promptly received was what I expected. An unboxed module, fresh-off-the-Epson instructions and a disc. "It" didnt work.

Embarrased by MY own ignorance I telephoned the "supplier". I received sympathy (balm to my dissappointment), understanding (solace to my vulnerable pride) and above all, real help.

The disc was verified and returned within 4 days. Still no joy. Another difficult phone call. More of the same. Patience is a virtue.

The hardware was returned for checking. Faulty chip found. But whose fault? Whose home-made power supply could be called into question? No questions asked. Within 5 more days "it" is now up and running. I have spent little money and gone no further than my corner letter-box.

That level of service cannot possibly be "commercially" worthwhile. But it IS available within the group. So I am now awaiting delivery of another un-boxed bare board add-on from a "group" source. I dont expect written guarantees, nor do I think they will be necessary. The originator (author or inventor) presumes the user has some "nouse". He (she) will not be hiding behind anonymous packaging or distribution networks and will be only a phone call away if/when help is needed.

Meanwhile, thanks to the User Group, experience grows and one day some more of us may be able to put back in what most of us take out.

Keep up the good works... Many thanks,

Steve Taylor.

I would like to help a primary school with setting up a computer system and would therefore wish to see what software is available. Who do I contact on this matter?

Yours Sincerely,

T. Schweiger

24 Langdale Drive,
Nuneaton,
Warks.
CV11 6EY
24/1/83

From David Washford

OK, so the TRS-80 owners are dwindling as a percentage of the total Micro-computer world. This fact disturbed me until I realised just how many were still in regular use and how they are still very popular with users who want more from a computer than just Pacman or Invaders; it is my contention that this drop in %share of the total is due more to the increase in total 'market' size than to any failing on the part of the TRS-80! I believe that I have already voiced my thoughts on the publication of a Tandy orientated magazine, and the position of the National User Group being well suited to being publishers of same.

Obviously, from your position, you have a much clearer picture of the overall situation than I do; on the other hand I can speak both for myself as an individual, and as Secretary of a Local Group. As an individual I need far more help,tuition,and news than my Local Group can provide, and as a Local Group Secretary I need the news, guidance, and direction that the National User Group provides. Thus do I justify the need for your continued existence! I see no harm (nor benefit - but that does not matter) in sharing workshops with other User Groups; in my case both Birmingham and Southall were such good 'value' that I never even noticed the Apples!

Paradoxically, it does concern me that both I and 99% of my Group are Model I users - the equivalent to 'old bangers' in my trade. Yet the Model III seems to be a far less popular machine than the Model I, which seems to be lending itself to fairly widespread modification; using the same trade comparison, perhaps it represents the Custom Car? All I can say is, there a lot of Users out here who WANT THE BENEFITS OF A NATIONAL GROUP!

Let us also consider one other important but illegal and immoral aspect of Workshops. As I sit writing this letter I have a Member of our Local Group enquiring as to travel arrangements for Mk - he must go 'to see what programs he might acquire.' I am not suggesting for one moment that you should condone or even connive at 'ripping-off', but I am saying that you should at least recognise the value of that activity to your Members.

Your notes in the current Newsletter suggest that it is only the workshops that are threatened at the moment, so perhaps I should point out that most of the above comes from a fear that that could be the thin end of the wedge. Are the workshops really so badly attended? Apart from the two mentioned above, I also had [illegible]ston and even Blandford, both of which were pure TRS[illegible], and I got the impression that all had a very good turnout. I suppose that Gloucester was a

disappointment, attendance wise, but might that have possibly been partly due to another of my pet hobby-horses, poor or late publicity? I don't know, I can't remember that far back!

Brian, HONESTLY, YOU ARE DOING A GREAT JOB! Nobody would pretend that any Club Secretaryship is fun, and I expect that too many people are like me and forget to thank you for the services provided and so gratefully received.

I leave this subject with the thought "As ye SOW, so shall ye reap" - in this case SOW stands for "Save Our Workshops" !

David Washford

From C. Philip

Following our telephone conversation the other day concerning TRS-80's and Typesetting, of which more anon, I thought it might be of some interest to your readers for me, a middle-aged owner of my own business to describe his experiences of using a TRS-80.

I am both a TRS-80 user and a member of the TRS-80 User Group but I find that approximately 95% of the Newsletter is, to me at anyrate, incomprehensible.

This leads me to the conclusion that either I am at fault or, perish the thought, the Magazine is at fault; perhaps it is not intended for the likes of me.

My business and my main interest in life is Fireworks and Firework Displays and I have recently been compiling a Bibliography on the subject. It soon became obvious that some form of word-processor could be of help in this venture so that I could assemble all the hundreds of snippets of information into a coherent whole. I therefore started making enquiries about a suitable machine and was at once confronted by adverts in Computer Magazines that appeared to be written in some kind of foreign language.

I then tried one or two shops that sold computers and was now confronted by teenagers who spoke this same incomprehensible language and who were quite unable to convey to me whether or not the particular machine that they were selling would do the job I wanted it to.

It was then that I came into contact with Andrew Tollett of the Bishopsgate branch of Tandy Computers and at once he appreciated my problem and explained in simple English words which I had learnt at school some 40 years ago what a Tandy with Scripsit could do. I bought one; and to this day I have no idea if this particular system is better or worse than any other for the purpose .

I put the Instruction Tapes on my Cassette recorder, "sat comfortably whilst we have some fun with your friendly TRS-80", and duly ploughed through the lessons trying to ignore what I felt at the time were not essential items such as putting headers and footers to articles on Hippopotumusses. Soon I was 'Word-processing'. Now came the tricky part.

I was not too happy with the quality of the Tandy Printers that were available at the time but had noticed the high quality and various type-faces that were obtainable on the Epson. I bought one and plugged in, all was well except I had no idea how to obtain the various type-faces that I had hoped for. Tandy initially could not help, "it is not one of our machines", the firm I bought the printer from could not help, "we only deal with Sharp computers". Tandy then suggested Molimerx but at the time no enhancement was available, then they suggested B & H Micros who, after a considerable amount of work in my home with their programme and my TRS-80 and Epson eventually obtained the various types I was hoping for.

All was now plain sailing, Scripsit was under control, and it did everything I wanted it to. I even taught the basic rudiments of the system to a friend of mine in an hour who can hardly change a fuse and is incapable of reading instructions.

It soon began to dawn on me that I could probably use this expensive equipment to assist with my company records such a Stock Control. All the systems that I was offered appear to do a great number of things I did not particularly want and I did not wish to invest in an expensive item of software without being able to try it out in practice. To give them their due, Tandy did offer to let me try it out and were prepared to take the programme back and refund me if it did not do what I wanted it to.

I was then introduced to the principles of Visical and it seemed to me that I could use this programme to format my own Stock Records. Again I sat down with the Manual and had soon grasped the basic concept and had by now learnt to ignore a number of items (i.e. Mathematical functions) that were of no concern. Soon I had devised my own format for my records. Now came the tricky bit again. I gathered from the instructions that it was possible to 'Setup' the Printer, but how? I wanted to obtain 132 columns which I felt could be done by using the 'Condensed' mode, the Epson instruction book told me to use 'SI', but this had no effect. I will not bore your readers by telling them in detail the trials, tribulations and time that was spent making the enquiries necessary to discover that what was wanted was 'CO'. I have no idea why this should be but judging by the highly technical articles in your Newsletter it would be second nature to your readers. NOT TO ME HOWEVER!

At this stage I felt that by trying to learn BASIC I might understand more about what I was doing as I have never been too happy about doing things parrot fashion. I got as far as about Chapter 8 and then asked myself, "What on earth am I doing trying to unravel the mysteries of String Variables, how will it help me?". The answer I fear is that it won't.

Which brings me back to the beginning and my query about Type-setting. I now have my Bibliography nearly ready for printing and have been led to believe that it should be possible to feed my Model III Scripsit discs into a Typesetting machine, however it appears that no company that advertises this service can cope with a Tandy Model III, any other system, "Yes", but Tandy, "No". Can any of your readers help?

This raises another issue. A Tandy owner seems to be out in the cold from the start. A glance through some of the Computer magazines and one would think only of cider, there being so many references to Apple or indeed any other machine but Tandy. Software and Hardware appears compatable with anything but Tandy.

Are there any others out there like me? I find it difficult to believe that I am unique. There must be many others who do not know their 'hex' from their 'byte' yet feel, as I do, that having been able to come to terms with as much as we have already, we should be able to progress, but how? At the present, the ultimate in magic is still the result of pressing 'BREAK', 'P', 'ENTER'

C.Philip
19 Coolhurst Road,
London N8

P.S. Has anyone else come across the SCRIPSIT BUG? If you accidentally press 'BREAK', 'V', 'ENTER' the screen display will freeze and any attempt to clear it will result in the Disc Drive performing a fandango and the Display will clear (destroying all your work) and filling with garbage!

From Steve Alderton

May I take this opportunity to make one comment on the Newsletter contents. I have observed that a large proportion of the articles/letters relate to either disks or word-processors. Obviously you can only publish material that is sent to you, so am I in the minority by having no interest in word-processing or having no hope of being able to afford disk drives? Also, there is only rarely a mention of the Model III. Does this mean that I am in the minority again? I don't intend to belittle the Newsletter at all, I am sure the contributions are of great value to those posessing these products, but I do hope the group isn't turning into one for business users only! I would be most interested in your comments and also those of any other members who are 'Model III cassette bound'.

Steve Alderton.

(* Most of the Model I material in the Newsletter is applicable to the Model III. If you would let me know what sort of articles you would like to see, I am sure some of the members would oblige. We mainly get disk orientated articles - perhaps most of the members now have disks. What about some articles by those of you who are still using cassettes? LFH *)

From J. Keneally

The Newsletter does not seem to contain many references to that Aladdins Cave of goodies at the bottom of your Model 1, known as the ROM (Rotten Old Mess?). Books on the subject must be a mystery to all except a fortunate few, so how about a review of some of the most useful ROM routines and how to use these in BASIC?

Do we have to have our language besmirched by all this Americanese which seems so popular in computing. I particularly object to the term 'garbage collection' used to describe the recovery of free string space. None of my BASIC programs produce any garbage whatsoever, it is all good, clean and wholesome English text. It would be different if Yankee software was of a higher standard. In my experience, if it works at all (unusual), it is impossible to use without a trunk-full of User manuals by your side. Come to think of it, perhaps 'garbage' is not so inappropriate!

On the problem of A. Robinson in the last issue, shouldn't the cassette commands read:-

```
CLOAD£-1,filename
```

etc., rather than:-

```
CLOAD£1,filename
```

or have I missed the point?

LONDON (SOUTH WEST) SUB-GROUP PROGRAMME

17 February	Computer Games
1 March	BASIC techniques
17 March	Z80 operation
5 April	No meeting
21 April	Machine code
3 May	CP/M
19 May	The ROM and useful addresses
7 June	Word processing
16 June	High resolution displays

All meetings at Stowford College, Sutton, at 8 PM.

Ron Everitt

FOR SALE

Holmes 48K expansion board (less RAM). New and unused. Sell for £40, or swap for a cassette or, preferably, floppy tape-based version of Scripsit or other word-processor, plus documentation.

C.T. Dempsey,
Flat 2,
57 Woodlands Road,
Ilford,
Essex IG [illegible]JN.

PROBLEM SECTION

MACRO-80

Does any user of Macro-80 know the fix for the fault in EDIT which scrambles the index file produced for long source files? Having experienced this and found the only answer was to delete the index file, I then read in Byte of a fix supplied by Radio Shack. When I wrote to Tandy about it I was told the only 'anomaly' they knew about was a scrambled index file, but that this was not needed for the TRS-80, and the cure was to kill the index file. Helpful people! The index file is, in fact very useful since it cuts down the time taken to load a very long source file.

Henry Shepherd

My copy of Super Scripsit is throwing up problems, fortunately of a minor variety so far - like @H does not give help when the cue at the front of the screen is telling me to type it, and <CLEAR> with or without one of the top row keys gives ↑ and apparently not the character coded.

Tony ??? (* He did not give his surname. LFH *)
21 Sandalwood Road,
Loughborough.

LNW EXPANSION INTERFACE - 32K ram,Disc controller,Clock,Printer driver, RS232 driver,Bus Connector,Power supply,Cable.Built and tested,but uncased and needs FDC and UART chips.....£120

ANDERSEN JACOBSON AJ211 Accoustic Coupler.New and Unused.....£80

Contact Mike Seaford.....01-953 8642

ORGINAL CASSETTE-BASED SOFTWARE FOR MODEL 1 LEVEL II

One of each:

	£
Scripsit + manual (Tandy catalogue 26-1505)	29
Molimerx Scripsit patch for Epson	5
Various games (Sargon II chess, U-boat strike, Battle of Britain, Amazing maze-Touchdown, Invaders) Price for the lot	10

Contact Bob Wiggins Hassocks (07918-2430), 'Dillhaven', North Bend, Hassocks, W.Sussex.

LOADING SYSTEM PROGRAMS THROUGH THE SECOND CASSETTE PORT ON THE GENIE

This Christmas not only shall I not have the disk and expansion I'd promised myself (DAMN KRAMM!), but the number 1 cassette is on the blink. Now, on the Genie, of course, it is built-in, so I have to fit my daughter's cassette player on to the £-2 port. For BASIC programs, OK - CLOAD£-2,"name" and so on. But SYSTEM programs are called to cassette £-1 by the ROM.

So, the following utility program willload a system program through cassette £2. At the end it gives "*". Enter"/" and the program will run. If you wish to load two programs, however, you can't.

Note: this program will not work with a DOS up, for it overlays the DOS entries. But, if you have a DOS, why do you need this utility? And by the way, it won't load some of the commercial software with its own loader. These call cassette £-1.

```
5  REM: TO LOAD SYSTEM TAPE ON CASSETTE £-2
10 DEFINT A-Z:XX=16722:FOR N=0 TO 16:READ C:POKE
   XX+N,C:NEXT
20 DATA 33,231,65,49,136,66,175,61,205,18,2,205,
   150,2,195,209,2
30 CLS:PRINT@ 960,"READY":PRINT@ 960,">SYSTEM":
   PRINT@ 960,"    ":PRINT@ 960,"*? ";
40 XX=16872:B$="":FOR N=0 TO 5
50 A$=INKEY$:IF A$="" THEN 50
60 C=ASC(A$):IF C=13 THEN 80
70 POKE XX+N,C:B$=B$+A$:PRINT@ 963,B$;:NEXT:GOTO 90
80 FOR M=N TO 5:POKE XX+M,32:NEXT
90 PRINT@ 980," ":POKE XX+6,0:POKE 36526,82:POKE 36527,65:
   XX=USR(N)
100 END
```

In assembly language the USR routine is:-

```
        LD      HL,41E7H    ;ADDR. OF INPUT BUFFER -1
        LD      SP,4288H    ;SYSTEM STACK
        XOR     A           ;THE BOOKS ARE WRONG HERE. TO
        DEC     A           ;CALL CASSETTE £-2, A MUST
                            ;CONTAIN 0FFH ON ENTRY TO 0212H
                            ;AND 0 FOR CASSETTE £-1
        CALL    0212H       ;SELECT CASSETTE, START MOTOR
        JP      02D1H       ;TO ROM, TO FINISH JOB
```

A. Hitchcock

HULLFORTH

HULLFORTH, originally produced for the Nascom, is now available for the TRS-80 and Genie, at £25, from A.F.T. Winfield and P.S. Cain, 148 Goddard Avenue, Hull, HU5 2BP.

P C DREW
Markyate

LPRINTING CHR$(0)...(11) & (12) revisited

First I must point out that the idea of how to solve this problem is not my own but came from Don Bannister (see news letter Sept 82)
My contribution is WHERE to put the code in memory.
I have been playing with string packing ever since Mark Leduc showed just how easy it was at the Luton Workshop last year,and the other night I recived a phone call asking how to turn off a function on a EPSOM printer which required a CHR$(0) to be sent to the printer , I was able to help but it set me thinking,as there must be a better method than the one I arrived at over the phone,This is it.
I enclose a short program that is commented and will allow you to do all sorts of nice things such as STRING PACKED HIRES GRAPHICS as you can now LPRINT all the ASCII codes

```
5 DEFINT A-Z
10 A$="0123":           REM SET's A$ up MUST not be reset later on as
it will reset your machine
20 V=VARPTR(A$):        REM Find the pointer to A$
30 LSB=PEEK(V+1):       REM Get and store in LSB the Least significant
byte of A$'s location
40 MSB=PEEK(V+2):       REM Get and store in MSB the Most significant
byte of A$'s location
50 AM=LSB+MSB*256:      REM Work out the Actual Memory loacation of A$
60 FOR X=0TO3:READ A:POKEAM+X,A:NEXT:
                        REM Now we know the address for A$ ,lets POKE
the Machine code in to redirect the lineprinter routine back on its
self
70 POKE 16422,LSB:      REM Tell the operating system where the new bit
of code is (LSB)
80 POKE 16423,MSB:      REM Tell the operating system where the new bit
of code is (MSB)
100 DATA 121,195,180,5,
1000 LPRINTCHR$(27);"K";CHR$(0);CHR$(1);:FOR X=0TO 255:LPRINT
CHR$(X);:NEXT:          REM Test Line
```

First Type in RUN 1000 and the printer will not resopnd correctly.
Then run the whole program and Lo and behold it work's a treat.

You can, once run delete lines 50,60 and 100 , when the program is saved it will retain the PACKED code in A$.
Dont alter or edit line 10 in any way once run as the system will crash

If you must touch or alter line 10 first POKE 16422,141:POKE 164423,5 :(D Bannister).It may be a good idear to do this on exit from any program that uses this,as you can somtimes leave other programs,ie Scripsit/Copyart in the lurch, a reboot is also just as effective.

Pete Drew

A SIMPLE MONITOR PROGRAM

One of the problems encountered by anyone who wishes to learn assembly language programming is finding a suitable application. This simple monitor program for the Model I should provide a suitable starting point. The monitor uses the following commands:-

Exxxx - Examine/modify memory at location xxxx (Hexadecimal)
Gxxxx - Goto address xxxx (Hexadecimal) and execute the program at that address

Command E first displays the contents of the memory location, and then waits for a keyboard entry. Pressing the space bar steps forward one byte, and pressing the minus key steps backwards one byte. Entry of two valid Hex characters enters that value into the selected memory location. An invalid character returns you to the monitor prompt - >.

The program is based on Neil Harrison's Minimon that was published in the first issue of PCW. Neil based his program on the Amsat 8080 monitor published in an early issue of Byte. I have modified Neil's program to make use of a JUMP table, making it very easy to add your own commands. This would be an excellent way to learn assembly language programming. Some suggestions are:-

A dump command to display memory contents 16 bytes to a line
A trace/single step command
Breakpoints (Hint - use a restart instruction) and register dump
Disassembly of instructions
In-line assembly (difficult)

The code is very modular and well-commented and should be self-explanatory, with the possible exception of the few lines following FOUND: in the main program loop. Crummy, obsolete processors like the Z80 do not have instructions of the form CALL (HL), that is, a register indirect subroutine call. I have simulated this instruction by pushing the return address onto the stack, and then JUMPing to the required subroutine. On encountering a RET instruction in the subroutine, the return address is popped off the stack into the program counter. The return address is always the monitor start point (BEGIN). The stack pointer is reset and things start again from scratch. Rather messy but the best I could do. Modern processors such as the 8088/8086 and the 68000 have a decent instruction set that includes register indirect subroutine calls.

```
        ASEG
;
CHROUT  EQU     33H
CHRIN   EQU     49H
SPA     EQU     20H             ;SPACE CODE
CR      EQU     0DH             ;ENTER CODE
START   EQU     7000H           ;START OF PROGRAM
;
        ORG     START           ;START HERE
        DI                      ;DISABLE INTERRUPTS
        LD      A,0EH           ;DISPLAY CURSOR
        CALL    CHROUT
BEGIN:  LD      SP,START        ;INITIALISE SP
        CALL    CRLF            ;NEW LINE
        LD      A,'>'           ;DISPLAY PROMPT
        CALL    CHROUT
        CALL    GETE            ;GET A COMMAND IN A
        LD      B,A             ;PLACE COMMAND IN B
        LD      HL,TABLE        ;POINT TO START OF COMMAND
                                ;TABLE
SEARCH: LD      A,(HL)          ;GET COMMAND FROM TABLE
        CP      -1              ;ERROR IF END OF TABLE
        CALL    Z,ERROR
        CP      B               ;COMPARE WITH INPUT COMMAND
        JR      Z,FOUND         ;JUMP IF MATCH
        INC     HL              ;IF NO MATCH - BUMP POINTER
        INC     HL              ;THREE TIMES TO NEXT COMMAND
        INC     HL              ;IN TABLE
        JR      SEARCH          ;AND KEEP SEARCHING
FOUND:  INC     HL              ;COMMAND MATCHES
        LD      E,(HL)          ;FORM COMMAND ROUTINE ADDRESS
        INC     HL
        LD      D,(HL)
        EX      DE,HL           ;IN HL
        LD      DE,BEGIN        ;RETURN ADDRESS
        PUSH    DE              ;PLACE ON STACK
        JP      (HL)            ;JUMP TO ROUTINE
;
;SUBROUTINES FROM NOW ON
;
;               * * * EXAM * * *
;
;EXAMINE MEMORY LOCATION AND MODIFY
;
EXAM:
        CALL    GXW             ;GET ADDRESS FOR EXAMINE IN HL
        CALL    C,ERROR         ;ERROR IF CARRY SET
ADDR:   CALL    CRLF            ;NEW LINE
        CALL    TXW             ;PRINT ADDRESS IN HEX
        CALL    SPACE           ;PRINT SPACE
        LD      A,(HL)          ;GET DATA AT CURRENT ADDRESS
        CALL    TXB             ;PRINT IT IN HEX
        CALL    SPACE           ;PRINT SPACE
        CALL    GXB             ;GET NEW DATA OR COMMAND CHAR.
        JR      C,EXAM1         ;COMMAND IF CARRY SET
```

```
        LD      (HL),A          ;PUT NEW DATA IN MEMORY
        CP      (HL)            ;MAKE SURE IT GOT THERE
        CALL    NZ,ERROR        ;ERROR IF NOT
NEXT:   INC     HL              ;BUMP ADDRESS POINTER
        JR      ADDR            ;AND GO ROUND AGAIN
EXAM1:  CP      SPA             ;IF SPACE DO NEXT ADDRESS
        JR      Z,NEXT
        CP      '-'             ;TEST FOR '-'
        RET     NZ              ;IF NOT '-' RESTART
        DEC     HL              ;IF IT IS DO PREVIOUS ADDRESS
        JR      ADDR            ;GO ROUND AGAIN
;
;               * * * GOTO * * *
;
GOTO:


        CALL    GXH             ;GET GOTO ADDRESS IN HL
        CALL    C,ERROR         ;ERROR IF CARRY SET
        PUSH    HL              ;PLACE GOTO ADDRESS ON STACK
        CALL    CHRIN           ;WAIT FOR <ENTER>
        CP      CR              ;IS IT CR?
        RET     Z               ;"RETURN" TO GOTO ADDRESS
;
;               * * * ERROR * * *
;
;ERROR ROUTINE. PRINT '*' AS ERROR INDICATION, AND RESTART
;MONITOR.
ERROR:
        LD      A,'*'
        CALL    CHROUT
        JP      BEGIN
;
;               * * * CRLF * * *
;
;PRINTS NEW LINE
;
CRLF:
        LD      A,CR
        CALL    CHROUT
        RET
;
;               * * * SPACE * * *
;PRINT SPACE
;
SPACE:
        PUSH    AF
        LD      A,SPA
        CALL    CHROUT
        POP     AF
        RET
;
;               * * * GXN * * *
;GETS HEX ASCII CHARACTER FROM CONSOLE AND RETURNS
;HEX VALUE IN LOWER NIBBLE OF A. IF ILLEGAL CHARACTER
```

```
;THE CARRY BIT IS SET AND CHARACTER RETURNED IN A
;
GXN:
        CALL    GETE            ;GET CHAR. FROM KEYBOARD
                                ;AND ECHO TO DISPLAY
        CP      '0'             ;LESS THAN '0'?
        RET     C               ;YES - RETURN WITH CARRY SET
        CP      '9'+1           ;LESS THAN OR EQUAL TO '9'?
        JR      C,GX1           ;YES - JUMP
        CP      'A'             ;LESS THAN 'A'?
        RET     C               ;YES - RETURN WITH CARRY SET
        CP      'F'+1           ;GREATER THAN 'F'?
        CCF                     ;COMPLEMENT CARRY
        RET     C               ;RETURN IF CARRY SET
        SUB     7               ;CHAR. BETWEEN 'A' AND 'F'
GX1:    SUB     30H             ;CONVERT TO BINARY
        RET                     ;AND RETURN
;
;               * * * GXB * * *
;
;RETURNS TWO HEX CHARACTERS AS A BYTE IN A. IF ILLEGAL
;CHARACTER CARRY BIT SET AND ILLEGAL CHARACTER RETURNED IN
A
;
GXB:
        CALL    GXN             ;GET A NIBBLE
        RET     C               ;RETURN IF CARRY SET - ERROR
        PUSH    BC              ;SAVE BC
        RLCA                    ;SHIFT TO UPPER NIBBLE
        RLCA
        RLCA
        RLCA
        LD      B,A             ;PLACE IN A
        CALL    GXN             ;GET ANOTHER NIBBLE
        JR      C,GX2           ;JUMP IF CARRY SET - ERROR
        ADD     A,B             ;ADD UPPER NIBBLE
GX2:    POP     BC              ;RESTORE BC
        RET                     ;AND RETURN
;
;               * * * GXW * * *
;RETURNS WITH FOUR HEX ASCII CHARACTERS IN HL. IF A NON-HEX
;CHARACTER IS FOUND THE CARRY BIT IS SET AND THE ILLEGAL
;CHARACTER RETURNED IN A
;
GXW:
        OR      A               ;RESET CARRY
        PUSH    AF              ;SAVE AF
        CALL    GXB             ;GET HIGH BYTE
        LD      H,A             ;PLACE IN H
        JR      NC,GX3          ;JUMP IF OK
        POP     AF              ;RESTORE AF
        LD      A,H             ;BAD CHAR. IN A
        SCF                     ;SET CARRY FOR ILLEGAL CHAR.
        RET                     ;AND RETURN
GX3:    CALL    GXB             ;GET LOW BYTE
```

```
        LD      L,A             ;PLACE IN L
        JR      NC,GX4          ;JUMP IF OK
        POP     AF              ;RESTORE AF
        LD      A,L             ;BAD CHAR. IN A
        SCF                     ;SET CARRY FOR ILLEGAL CHAR.
        RET                     ;AND RETURN
GX4:    POP     AF              ;RESTORE AF
        RET                     ;RETURN WITH VALID CHAR. IN A
;
;               * * * TXB * * *
;
;PRINTS CONTENTS OF A AS TWO HEX CHARACTERS
;
TXB:
        PUSH    AF              ;SAVE AF
        RRCA                    ;TOP NIBBLE TO BOTTOM NIBBLE
        RRCA                    ;OF A
        RRCA
        RRCA
        CALL    TXN             ;PRINT NIBBLE
        POP     AF              ;RESTORE AF
;TXN PRINTS FOUR LSBS OF A AS A HEX CHAR.
TXN:    PUSH    AF              ;SAVE AF
        AND     0FH             ;MASK OFF TOP FOUR BITS
        CP      0AH             ;NUMBER?
        JR      C,TX1           ;YES - JUMP
        ADD     A,7             ;NO - ADD OFFSET FOR LETTERS
TX1:    ADD     A,30H           ;CONVERT TO ASCII
        CALL    CHROUT          ;DISPLAY IT
        POP     AF              ;RESTORE AF
        RET                     ;AND RETURN
;
;               * * * TXW * * *
;
;PRINTS CONTENTS OF HL AS FOUR HEX CHARACTERS
;
TXW:
        PUSH    AF              ;SAVE AF
        LD      A,H             ;HIGH BYTE OF HL IN A
        CALL    TXB             ;DISPLAY IT
        LD      A,L             ;SAME FOR LOW BYTE
        CALL    TXB
        POP     AF              ;RESTORE AF
        RET                     ;AND RETURN
;
;               * * * GETE * * *
;GET CHARACTER FROM CONSOLE AND ECHO IT
;
GETE:
        CALL    CHRIN           ;GET CHAR. IN A
        CALL    CHROUT          ;DISPLAY IT
        RET                     ;AND RETURN
;
;JUMP TABLE (ADD YOUR OWN COMMANDS HERE!)
;
```

******MKSS******
24 OXFORD STREET,
STONY STRATFORD,
MILTON KEYNES
0908--564271

FOR SALE

SOUND BOARD(LFH)
EPROM BOARD(LFH)

LOWER CASE KIT(EH)
RS232 KIT (EH)
RS232 KIT M3 (EH)

COLOUR GENIE(SB)
DISCS 5 OR 10 (SB)

EDGE CONNECTORS (JK)
NOTES ON ORCH-80 (CK)
NOTES ON SCRIPSIT (CK)

****

```
TABLE:
        DB      'E'         ; insert your own commands here!
        DW      EXAM
        DB      'G'
        DW      GOTO
        DB      -1

        END     START
```

Leon Heller

CUTTING DOWN ON GARBAGE.

Seeing my early notes on garbage collection reprinted prompts me to add a footnote or two. A program was published in Byte a while ago to print a large C on the screen during garbage collection so that you knew why the computer had stopped. I tried this out and although it worked, it doubled the time the garbage collection took, as well as slowing down all string operations; on balance I preferred it without!

Two ways exist of preventing garbage arising in the first place, and if it doesn't arise it doesn't need collecting! Both are only available in Disk Basic, unfortunately. They rely on never using an expression of the sort A$= ..... after the first time in any program.

First rule - when you first allocate a string variable make it as long as it will ever need to be.

Second rule - if it is written in the program text write it in two bits with a + sign in between thus:

```
10 A$="This program "+"is mine!"
```

Alternatively, define it as a blank string thus:

```
10 A$=STRING$(" ",63)
```

By doing this you ensure that it will be stored in the reserved string area, rather than in the program text.

Third rule - whenever you need to change the content of A$, do it EITHER using MID$(A$,x,y)= (new string), OR using LSET A$= (new string). The second method is much the easier as it doesn't require you to count letters to find the values of x & y. Whichever method you use, the result is the same, your original string is amended where it exists in memory, and no garbage is created. The book doesn't say you can use LSET for ordinary strings, but you can. LSET pads strings with blanks so if you are printing them on the screen or a printer you may have to resort to tactics like inserting marker characters to show how much of the string you want to print. For example, if you have set up A$ initially to be 63 blanks, and you want to print a shorter string than this, then:

```
LSET A$=B$ + C$ + CHR$(1)
```

will give you a 63-byte string consisting of A$ and B$ joined together, followed by a non-printing byte (01 hex), with the remainder of the 63 bytes all blanks (20 hex). To print only the business part of this, use:

```
PRINT LEFT$(A$,INSTR(A$,CHR$(1)));
```

Because CHR$(1) is a nonprinting character it doesn't matter whether you print it or not - it won't affect the display. Don't use CHR$(0) as it may upset Basic.

Neither method will work for strings in program text, hence rule two!

It's not always possible to do things this way, but even if you only prevent some garbage collecting, every little helps.

Henry Shepherd

PASSING PARAMETERS FROM DOS

A little used feature of TRSDOS is the ability to pass parameters to a program from DOS. The TRSDOS manual states that location 4318H is the start address of a 64 byte buffer containing the last TRSDOS command (usually a filename) that was entered. The following routine places the word following the file name into a second buffer. The second word could be the name of a file to be input to the main program, and the second buffer could be a DCB. For instance, COUNT FRED/DOC would load a word count program which would count the words in the file FRED/DOC. The routine can easily be extended for additional parameters. Some commercial programs such as MACRO-80 and LINK-80 use this technique; M80 FRED=FRED will load MACRO-80 and assemble the source file FRED/MAC, producing the relocatable object file FRED/REL. It would be nice if this feature were added to Scripsit - SCRIP FRED/DOC would load Scripsit and a text file called FRED/DOC, saving a few keystrokes. How about one of you trying it!

```
GETPARM:
        LD      HL,BUFF1        ;POINT TO START OF BUFFER
                                ;(4318H)
G1:     LD      A,(HL)          ;GET CHAR IN A
        CP      CR              ;CARRIAGE RETURN?
        JR      Z,DONE          ;IF CR - DONE
        CP      SPACE           ;SPACE?
        JR      Z,G2            ;IF SPACE - JUMP
        INC     HL              ;NOT SPACE - GET NEXT CHAR
        JR      G1
G2:     INC     HL              ;SKIP SPACE(S)
        LD      A,(HL)          ;
        CP      SPACE
        JR      Z,G2
;HL NOW AT START OF SECOND WORD - MOVE TO BUFF2
;HL=SOURCE
        LD      DE,BUFF2        ;DE=DEST
G3:     LD      A,(HL)          ;MOVE WORD UNTIL CR OR SPACE
        LDI
        CP      CR              ;DONE IF CR OR SPACE
        JR      Z,DONE
        CP      SPACE
        JR      Z,DONE
        JR      G3              ;NEXT CHAR
DONE:   RET
```

The above techniques can be used wherever there is a requirement to extract words from a sentence, in Adventure type games and spelling checkers, for instance, or tokens from statements when writing assemblers and compilers.

Leon Heller

PS 232 INTERFACE by ERIC HARTLEY
From Doug Irvine-VG owner

I am sorry that this review has taken so long but apart from being snowed under at work I've had great difficulty in obtaining a driver program for testing the RS 232

The board that I was given to build was made by Eric Hartley , whose lower case mod I have been using for some time now The board is double sided but NOT through hole plated & this caused a few problems in the building of the board Construction of the board is based around the use of Soldercon socket pins, which join the two sides of the board at the I.C. legs
This job is quite tricky as too much solder will fill the socket & too little will give a dry joint.
The total time taken to fit the Soldercon pins was 90 mins & a further 20 mins taken to check for shorts & dry joints It is essential to check for solder bridges at this point as some tracks pass very close to IC legs under the IC itself On the power supply side I chose not to use an external power supply to A) save space & B) cut cost I therefore added a -12 volt regulator to the inside of my Genie & took this,& the internal +12 volt lines out via a seperate socket
Eric suggests using a 40 way DIL plug to connect between the expansion bus & the RS232 , I now agree, but at the time I hard wired my cables direct. After checking the supply voltages on the board I plugged in all the ICs,switched on,& checked for correct computer opperation , all o.k It was then that I stated having problems with software Eric suggests buying a copy of the Tandy RS232 manual ,to use the TERM program it contains I tried for almost six weeks to get one but in the end I gave up ! Tandy just don't want to know! Eventually I aquired a copy & was back in business.
Unfortunately my software prowess is not all it might be & the TERM Program refused to assemble for me so I typed it in in hex , success , the screen cleared & a cursor appeared at the top left hand corner of my T.V. but even with pins 2 & 3 of the interface plug linked the keyboard was dead A dry joint on the xtal was the culprit & at last the keyboard echoed back,all keys except the alpha characters worked , suggesting bit 7 low this proved to be an incorrect setting of the hardware links (which set parity , baud rate etc) So at last my RS 232 is working & although I have not at present had it linked to a printer or a modem I am sure that it will perform as well as my lowercase does I would not recomend this construction project to people who have no electronic experience as it is quite a delicate job Eric Hartley is again to be congratulated on a well made product, produced at a reasonable price I would be happy to offer any help over the phone to anyone building this board or adding an internal power supply to their Genie I can usually be reached in the evening on 0296 661315

Doug Irvine

```
10 'EXAMPLE OF USING CMD'0' FROM ND/80 2 0, SENT IN BY:-"
20'STEVE HALL FROM 1 MEADOW VALLEY, ALWOODLEY, LEEDS17, LS17 7RF.
30 CLS
40 PRINT"A Newdos 'MF' file comprising 500 strings and 500
        integers is now being loaded from disc
50 CLEAR 10000 DEFINT I
60 DIM A$(500),B%(500),S%(500) 'THE ARRAY S%(500) IS FOR USE BY    CMD"0"
70 OPEN "I",1,"MFDATA/DAT","MF",10  'THIS IS AN 'MF'FILE WITH      LRL=10
80 FOR I=1 TO 500:'I IS SET AS AN INTEGER IN LINE5 0
90 PRINT@ 450,"The Record being Loaded is;"
100 PRINT@478, I:'THIS ADVISES THE USER HOW MANY ITEMS HAVE BEEN READ FROM DISC
110 GET 1,,,A$(I),B%(I); 'A$(I),B%(I), IS AN IGEL:THE"," IS IMPORTANT
120 'NOTICE THE ND80 JARGON - IGEL.
130 NEXT
140 CLOSE:'ONCE READ FROM DISC CLOSE BEFORE DOING ANYTHING ELSE.
150 FOR I=1 TO 500
160 PRINTA$(I),B%(I),:'THESE ARE THE ITEMS READ FROM THE DISC.
170 NEXT
180 PRINT"Press any Key to sort files (sort is made on strings)
190 Z$=INKEY$ IFZ$=""THEN 190 'GIVE A PROMPT FOR TIMING SORT
200 CMD"O",500,*S%(1),A$(1),B%(1) '*S%(1) IS DUMMY ARRAY,AND VALUE WILL BE OF
210 PRINT"Sort is now complete!                                    SORTORDER.
220 PRINT"PRESS ANY KEY TO LIST SORTED FILE"
230 PRINT"The sort took about 6 seconds
240 Z$=INKEY$:IFZ$="" THEN 240
250 FOR I =1 TO 500
260 PRINTI;A$(S%(I)),B%(S%(I)), 'THE VALUE OF S%(I) BEING USED TO "
270 'THE A$ IS OBVIOUSLY AN ALPHA ELEMENT, AND THE B% A NUMERIC ELEMENT/ITEM
280 'ACCOUNTANTS ARE HAPPY WITH ITEMS, PROGRAMMERS WITH ELEMENTS.
290 'PRINT OUT TO THE DISPLAY THE ITEMS NOW SUCCESSFULLY SORTED.
300 NEXT
310 PRINT"The line with the sort command is      200
```

IN REPLY TO C.L.LOVELL - PAGE 4 ,OCTOBER 1982 ISSUE

THE TECHNIQUE USED IN THE LINE :-

```
IF Z AND 8 Y=Y-1 ELSE IF Z AND 16 Y=Y+1 ELSE IF Z AND 32 X=X-1 ELSE
IF Z AND 64 X=X+1
```

IS ALL TO DO WITH THE KEYBOARD SCAN ROUTINE WHICH MAY BE CALLED BY THE STATEMENT :-

```
Z=PEEK(14400)
```

ON CALLING THIS ROUTINE DURING A PROGRAM THE VALUES OF CERTAIN RESPONSES TO THE KEYBOARD MAY USED IN THE CONTROL OF THE PROGRAM. IN THE EXAMPLE GIVEN THE VALUES BELONG TO THE ARROW KEYS WHERE:-

(8) = UPARROW
(16) = DOWNARROW
(32) = LEFTARROW
(64) = RIGHTARROW

OTHER KEYS AND THEIR VALUES ARE:-

(1) = ENTER
(2) = CLEAR

(DOES ANYONE OUT THERE KNOW IF ANY MORE KEYS ARE AVAILABLE FOR THIS PURPOSE INSTEAD OF USING "INKEY$"?. IF THERE ARE ANY MORE KEYS HOW ABOUT A SHORT PROGRAM TO CALCULATE THEIR INPUT VALUES!!)

TO CLOSE I OFFER A SHORT PROGRAM (PRACTICAL BUT NOT ELEGANT) TO DEMONSTRATE THE ABOVE TECHNIQUE IN DIFFERENT CIRCUMSTANCES THAN A "DRAW WITH THE CURSOR" ROUTINE

CLIVE KEECH

P.S. -- BY THE WAY - THE ADDRESS EFFECT AT THE TOP OF THIS LETTER CAN BE OBTAINED BY USING THE PRINT PARAMETER - FR=Y - IN SCRIPSIT WHEN BOTH MARGIN VALUES ARE SET.

(* The address was right justified. LHH *)

```
10 CLEAR 5000
12 G=PEEK(14400):GOSUB 20:GOSUB 110
15 A$=STRING$(30,32)
20 CLS:PRINT A$,"MENU":PRINT A$,"====="
30 PRINT TAB(5)"TO LIST THIS PROGRAM HIT <ENTER>"
40 PRINT TAB(5)"TO PRINT A LISTING OF THIS PROGRAM HIT <CLEAR>"
50 PRINT TAB(5)"TO BREAK THIS PROGRAM HIT ";CHR$(93)
55 FOR X=1 TO 1000:NEXT
60 RETURN
100 '
102 '****** KEYBOARD DECODE ******
105 '
110 IF G AND 1 THEN LIST ELSE IF G AND 2 LLIST ELSE IF G AND 16 STOP ELSE 12
120 RETURN
```

PRESTEL 300 BAUD SERVICE

Dear Computer User Group

Prestel are launching a 300 baud service for microcomputer and teletype terminal access in January 1983.

The new service will initially be available on Prestel's London computers (Dryden and Kipling) and will allow connection from any microcomputer/teletype terminal using a 300 baud modem currently approved by BT.

The full Prestel database of some 250,000 pages of information are offered. For example:

GENERAL INFORMATION on News, Weather, Amusements, Buying and Booking, Travel, Sport, What's On.

BUSINESS INFORMATION on Stock & Shares, Commodities, Securities, Exchange Rates, Company Accounts, Shipping and Government information.

LOCAL INFORMATION by Town & Region.

TELESHOPPING and other interactive services.

ALLADIN'S CAVE Programs for micros that can be downloaded direct onto your machine.

MICRONET 800 Coming soon a completely new microcomputer database on Prestel specifically designed for micro-users.

Prestel Charges (excluding telephone call)

SUBSCRIPTION	Business User	£15 per quarter
"	Residential User (Includes micros used at home)	£ 5 per quarter

TIME CHARGE

As a special incentive to the Residential User, Prestel will not be making any time charge when the system is used in the evenings, Saturday afternoons and Sunday.

i.e. 8-6pm Monday-Friday) 5p per minute
8-1pm Saturday)

ALL OTHER TIMES FREE

How to apply

If any of your members are interested in joining the new service why not complete one of the attached forms and sent it to

Andy Anderson
Prestel 300 Service
Prestel Headquarters
Room 419
Telephone House
Temple Avenue
LONDON
EC4Y OHL

I am interested in subscribing to the new Prestel 300 baud service opening in January 1983. Please send more information when available.

Name .....

Address .....

.....

.....

.....

Tel No. .....

Type of Micro .....

Type of Modem (eg BT Model 2, etc) .....
(if known at this point in time)

COMMUNICATIONS CONFERENCE

NATGUG, in conjunction with BASUG (the British Apple Systems Users Group) will be running a one day communications conference at Churchill College, Cambridge, on Saturday 12 March. Topics to be covered will include Prestel, Micronet 800, bulletin boards, and direct connection between micros. The cost should be under £5, including a snack lunch. This is not a workshop, so you will not be very welcome if you bring your system. Contact Brian Pain on (0908) 564271 for further details.

Leon Heller

MICROLINE 82A LINEFEED SWITCH SELECT

Depending on whether I'm using my Microline 82A printer as a listing device or a w.p. printer I frequently need to either enable or disable the linefeed when a CR is received. This function is controlled by Switch 6 in the DIP switch package mounted on the Operation Panel Circuit Board.

DIP SW6

ON	OFF
As CR is received, printer prints, returns carriage, and moves paper up one line.	As CR is received, printer prints and does carriage-return only.

Unfortunately the switch can only be accessed by removing the complete upper cover of the printer, which is rather inconvenient, so my idea was to wire a miniature toggle switch in parallel with SW6 (which is then left in OFF position) and the function is then controlled by the toggle switch. The easiest way to get at the pins of the DIP switch is to remove the Operation Board by undoing the single retaining screw which holds it on the sub-base, and unplugging the ribbon-connector. SW6 is the 3rd switch down from the top of the board. Solder a 6" wire to each of the pins of SW6 (on the back of the p.c. board) and connect the other ends to the toggle switch, which can be conveniently located in the slot in the sub-base to be found just to the right of the Operation Board fixing screw. Replace the Operation Board and connector, and refit the printer's upper cover. The function of SW6 can now be controlled completely by the toggle switch which is easily reached simply by removing the ribbon access cover.

Note that in operation the desired toggle position must be set before switching the printer on, because this sets the logic state; any operation of the toggle after switch-on will have no effect. If you wish to change the logic state you must switch off, re-select the toggle position, then switch on again.

John Sollis tel: Ruislip 37289

USING PRESTEL AT 300 BAUD

A 300 baud port is now officially available on Prestel (see the note from Andy Anderson of Prestel eleswhere in this issue) and I have now registered with Prestel, enabling me to use this facility.

Actually registering was very simple. I just sent off the form to Andy Anderson (photo copy the one on page 24 if you do not wish to mutilate your copy of the Newsletter) and a few days later received a phone call from Prestel. I gave all the details to them over the phone and received an information pack and application form a couple of days later. The tricky bits on the application form had already been filled in for me, so all I had to do was fill in a few other details, sign it, and send it off. A couple of days later I received a letter confirming my registration and giving my Customer Number, Password, and the telephone number of the Prestel computer with the 300 baud interface.

The next day I loaded ST80III, connected my acoustic coupler to my Model I, and dialled the number. On hearing the tone from the modem at the other end I placed my telephone handset in the acoustic coupler and the Prestel signing-on message appeared. However, the Prestel computer did not like my Customer Number. This was a Sunday, so the next morning I dialled a Freephone number mentioned in the registration letter, and explained my problem to a very helpful chap at Prestel. He thought it likely that a mix-up had ocurred, said he would sort it out, and register me on all the Prestel computers into the bargain, so I could also use the standard 1200/75 baud service if I had access to a Tantel or similar adaptor. As he suggested, I tried again in an couple of hours and this time the Prestel computer accepted me as a valid user.

The Prestel format is 40 columns and 24 lines. The columns are OK, as the Model I has a 64 column display, but the 16 line display on the Model I means that the top part of the Prestel page scrolls off the screen. I intend to write my own communications program that buffers the Prestel page in memory and then allows viewing of both parts of the page on demand. Alternatively, with ST80III one can save the incoming pages to disk, and view them later with Scripsit. I also tried a Model II, with my own communications software, and found that this was much more satisfactory, as it has a 24 line display.

The 300 baud interface is a bit slow, but I found that if I pressed "X" as soon as I got the information I wanted, the page that was being transmitted would be aborted, and I could then go directly to the next page I wished to view.

Leon Heller

USING AN EX BRITISH TELECOM MODEM TYPE 2A.

I have just acquired a modem type 2A and finally managed to make it work correctly in all its functions. This article may however also be of use to anyone else planning to use a modem in an auto answer-mode.

The modem 2A is a large two tone grey box weighing about 6 kgs. It is a mains powered unit designed to run for long periods without overheating. It has a standard 25 pin 'D' socket on the rear for connection to a computer RS-232 output.

The description auto-answer on the modem is misleading. The modem is not meant to answer the phone by itself. It is normally used by connecting up (details of this later) the various wires and leaving the modem in the not connect-modem-to-line status. The phone may then be used normally for speech. If an incoming call is received the modem will signal to the computer via the RS-232 ring-detect status that there is an incoming call. The modem will answer the call on receiving the computer instruction to set connect-modem-to-line.

Should a call out be required the terminal program will normally tell the modem to connect to the line.

The modem does its own switching between answer and originate mode. The modem can be kept covered and sealed, with no switch adjustments in normal use.

To set the modem up remove all power from the modem and do not connect it to the telephone lines. Open the modem by unscrewing the corner screws in the base and removing the lid.

Facing the front you will now find a row of test sockets on the edge and immediately behind them a series of pins sticking up. The pins have to be joined in various combinations to preset the modem options:

Join C/F LOCK COM to MK
Join C/D TIMING COM to 50
Join HOLD COIL to the pin just below it
Join the pin between 'OUTPUT LEVEL' and 'COMMON' to -5
Join RFS TIMING COM to 20
following the unbroken line on the right marked normal
Join 1 to 2; 3 to 4; and 5 to 6.

Set the left hand front switch to the centre position 'OFF'. Set the right hand front switch to the back position 'NORM'.

The front and the switch positions are now set. At the rear of the unit remove the white label saying 'DANGER 250 VOLTS' and connect a power cable with a plug on the other end to the terminals beneath. Replace the label. Do not yet apply power.

Connect a second telephone with dial (if desired) to terminals 5 and 6 of the rear block. Connect the telephone line to terminals 11 and 12.

Check all connections. Connect the RS-232 connector to the computer, switch the power on and test the system by pressing the small button at the rear next to the RS-232 socket. The modem should echo back to the computer any characters sent to it.

At this stage the modem is an originate only modem. The answer mode may be forced by turning the front left switch to ANSWER.

To set the auto-answer mode you will need a 1 mfd capacicitor with a working voltage of at least 160 volts. Do not use electrolytics. Tandy sell suitable capacitors.

You will also need a type 1A1 thermistor. This is obtainable from friendly (!) B.T. (i.e. G.P.O.) engineers. I am sure there is an equivalent for it commercially but I don't kow the serial number.

Join 13 to 15 on the rear terminal block. Connect the thermistor between 7 and 12. Connect the capacitor between 8 and 11. Replace the case and power up as before.

The modem is now in full auto-answer service.

Don't worry if the right hand two slots for circuit boards are empty. This is normal. They were only used for the earliest series of the modems.

DO NOT repeat DO NOT twiddle the potentiometers on the boards especially that on the fourth board from the left.

If anyone has a source for the fuses for the modem please let me know. They are a non standard imperial size.

It will be seen from the above that the modem is an unintelligent slave device which has to be controlled from the computer via the RS-232 line. If you are using a Tandy RS-232 (all the instructions up to now have applied to any computer) there are a few special points worth noting. The first and most important is that the RS-232 seems to have a memory for its old settings. I have found that if the modem is disconnected from the line by means of

not setting pin 20 of the RS-232 then that position is remebered by the system after the whole computer installation has been powered down.

To use the modem in an auto-answer mode the software must include the following stages:-

check for an incoming ring
connect modem to telephone line
check for a carrier
if no carrier shut down
if carrier found enable transmission.

All these checks are done by testing and setting the various RS-232 lines. These are fairly well explained in the manual which comes with a Tandy RS-232. It should be noted (and the manual does tell you) that the logic for the RS-232 is the reverse to that expected. That is to say to set pin 20, "connect-modem-set-to-line", a value of zero is sent to bit 1 of port EAH. In BASIC this could be written OUT 234,253 or OUT 253,0 ; in fact any number OUT which does not have a binary 2 in it. This information is only needed for those who plan to write their own software for auto-answering.

The Tandy RS-232 board has been much maligned. It's reliability can be much improved if instead of using the two washers supplied under the mounting screws a metal bar is used. I have taken a piece of brass cut to the size shown on the board as the area for contact, and drilled it with two holes for the screws. When this is used as a washer the RS-232 board is held down along the entire contact length. The bar prevents any slight warping of the board which combined with dirt can give bad contacts.

The most common part to fail on the board is the 5 volt to +12,0,-12 volt inverter. This is the black rectangle near the contacts. Note that when this is about to fail it behaves erratically first and will give rise to reboot problems. This is your cue to have it checked. I have modified my RS-232 by breaking the +5 volt supply to the board and fitting a switch in series. This enables me to switch my board off when not required, thus prolonging its life and eliminating any problems it might bring. To set your mind at rest I am using an RS-232 with double density on my Model 1 and have had no problems.

If you have a modem but have no RS-232 do not give up!-

Fred Brown in Hull sells an RS-232 adaptor which connects to the keyboard unit for about £45.
The National TRS-80 U.G. sell a circuit board to build a full RS-232 on the bus expansion: about £10 for the bare board.

I have (and by now the library will have) a program from '80 Microcomputing (June/July 1982) which gives a software RS-232 through the cassette port. This is only suitable for use as a terminal, not as a bulletin board. About £10 of parts will be needed. This will work on a 16K up Model 1 with or without expansion interface and disks.

You can leave messages for me on my modem where I am running a bulletin board in the evenings.

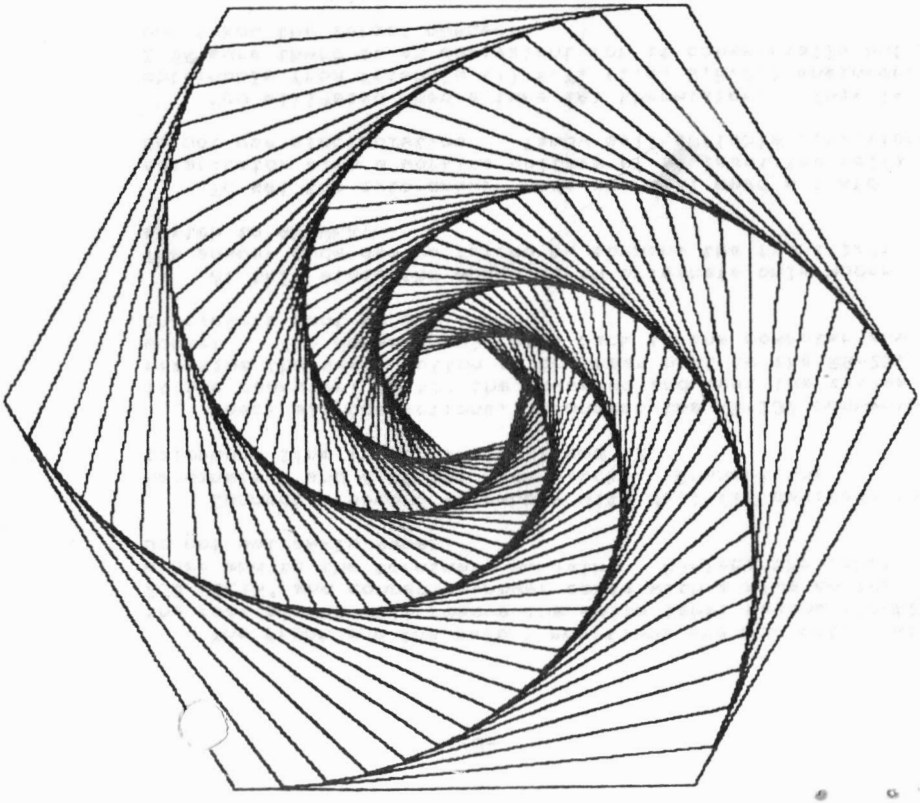
Mondays & Fridays 22:00 to 01:00
Sundays 21:00 to 01:00

Other days between 21:00 and 01:00 I will put on the bulletin board if I am in and anyone phones asking for it. The telephone number is 01-348-6518.

I run my bulletin board with 300 baud, one stop bit, even parity enabled and a word length of 7 data-bits and 1 stop bit. These are the switch settings as described on page 8 of the Tandy RS-232 manual.

I would like to give my thanks to all those who helped me get going, especially Jim Eccleston for technical know-how, Nick Robinson for the software RS-232 and Jim Erskine of Belfast for many test calls to check my system really worked.

John Newgas.



A USE FOR THE EXPANSION INTERFACE RELAY.

If, like me, you have an expansion interface and use disks, you probably never use the relay in the expansion interface which changes from cassette 1 to cassette 2. It is a fairly robust job and seems rather wasted just sitting there. Since I am never going to use two cassettes, I use it as a buzzer by calling the following subroutine whenever I want a buzz:

```
FORB%=0TO5:POKE&H37E4,1:B%=B%:POKE&H37E4,0:B%=B%:NEXT:RETURN
```

All this does is to close the relay, wait a bit, then open it again, wait and repeat the cycle, in this case 5 times. The wait-a-bit is provided by the two B%=B% statements which give the relay time to operate, but don't tie up another variable; if it doesn't buzz try using a slower statement like A=SIN(1). It's not a very elegant buzz, but it does draw attention to the fact that the computer is waiting for you to do something. The limits of B% may need to be altered to give the length of buzz you want, since the time the computer takes to find variable B% will depend on the complexity of your program. I wouldn't try it if you do use two cassettes, as it may not do the relay much good.

Henry Shepherd

SOFTWARE FOR SALE.

I have the following software software for sale - in all cases the original disks/cassettes and manuals (used only to produce working copies):

Radio Shack Editor/assembler (cassette)	£5.00
Microsoft Editor/Assembler-Plus (cassette) (much enhanced, with Macros and a very good monitor/debugger, Z-Bug)	£10.00
IRV (disk) (this is a keyboard enhancer program for use with TRSDOS, NEWDOS 2.1, or NEWDOS80 v.1.0. It provides auto-repeat, a flashing cursor, and allows you to type phrases and key words with one keystroke. Also includes an onscreen Editor for Basic)	£7.00
LDOS 5.0.1 (with patches to 5.0.3) (Supports double sided and double density disks. Not economically upgradeable)	£35.00

PLUS POSTAGE!
Offers considered.

H.H.Shepherd,
3, White House Close,
Shippon,
Abingdon,
OXON OX13 6LP
Tel: 0235 21921

NEW HARDWARE AND SOFTWARE FROM TANDY

Tandy have announced that they will shortly be supplying the XENIX operating system for the Model 16. XENIX is the Microsoft implementation of UNIX, about the best operating system for programmers that there is. I have my doubts about whether it is suitable for end users, but it is relatively easy to write a special interface for UNIX which insulates the user from the tricky bits. It will probably be rather expensive: into four figures, I would think!

The Model 12 has been announced. This is a Z80-based system in a similar box to the Model II, but with 128K of RAM and a built-in hard disk. Someone from Tandy told me it will sell for about £1800 over here. If that is true it will be extremely good value.

Tandy will shortly be supporting LDOS for the Model I/III, and I was told by a member of staff at Seacoal Lane that they will also be supporting CP/M for the Model II/16. I would take the latter with a pinch of salt, although quite a large proportion of Model II users have CP/M.

Leon Heller

USING MACROS

Laurie Shields' pontifications notwithstanding, macros can be very useful. I recently needed Z80 instructions of the form LD HL,(HL) for the HL, BC and DE registers. This form of indirect register load is not in the repertoire of the Z80, so I simulated it using the following macro:-

```
ILD     MACRO   R1,R2           ;;R1=H, B OR D. R2=L, C OR E
        PUSH    R1&R2           ;;REGISTER CONTENTS IN IX
        POP     IX
        LD      R2,(IX)         ;;LOW BYTE IN REGISTER PAIR
        LD      R1,(IX+1)       ;;HIGH BYTE IN REGISTER PAIR
        ENDM
```

ILD H,L will now simulate the (non-existent) instruction LD HL,(HL). Similarly for BC and DE. This saved me quite a lot of typing, besides making the program much easier to follow and debug. Note the use of the "&" to concatenate the symbols R1 and R2, representing the dummy parameters, and the double ";"s to suppress the comments each time the macro is expanded.

Macros can be tested quite easily by writing a simple test program, and creating a listing file when assembling. Use the form M80 FRED,FRED=FRED to create a listing file FRED/LST. Examination of the listing file will show whether the macro expansion behaved as expected. For instance, here is a portion of the listing file produced when the above macro was tested:-

```
                                        ORG     7000H
                                        ILD     H,L
7000    E5                              PUSH H&L
7001    DD E1                           POP     IX
7003    DD 6E 00                        LD L,(IX)
7006    DD 66 01                        LD H,(IX+1)
                                        END     7000H
```

In the manual for the latest CP/M version, Microsoft admit that it is only for experienced programmers, so if you are a beginner it would be advisable to start off with a simple assembler like ZEN or EDTASM, and then graduate to MACRO-80 when you feel you are ready for a proper assembler.

Leon Heller

SLOWER AND FASTER

If anyone is using Profile on the Model 3 then beware the Sort. No, stop laughing, this is deadly serious. If a two disc database is set up on the Model 3 with Tandy's latest version of Profile then all seems OK until live records reach the second disc. At that point although the records can be displayed and modified the moment one tries to sort them Radio Shack's disc non-operating system denies access and the program bombs out with something like 'Password Protection'. Whether this is some kind of Texas computer joke or Tandy's new approach to shortening the time their sort takes has yet to be established.

F A S T E R : At Luton John demonstrated a little utility called Faster (written by Pro-soft and obtainable from Wordworth). This program quite cleverly intercepts the Basic interpreter's accessing of all the variables being used by your program and on request displays them in tabular form together with the total number of accesses to each. To make things easy they are also sorted into descending frequency of use. This information can now be put to good use as by by making a few additions to your program the most often used variables can be located at the start of Basic's variable look-up table and so minimise the amount of non-productive time spent in the program's execution.
Whilst this will obviously save some time, it might at first seem rather marginal and not worth the effort but this is far from the case. The Profile sorting in Trsdos Basic, mentioned elsewhere, now runs in under two-thirds of the time of Profile's sort and the Newdos80 version has

decimated a sort of 178 records of 227 bytes each from over 40 minutes down to under 4, and Nigels production same now list all 800 plus products, sorted by customer, sorted by description or part number, in two hours as opposed to four and a half.

Laurie Shields

(* My apologies to Laurie. I just came across this article of his on one of my disks. He gave it to me ages ago at the Luton workshop. LFH *)

MILTON KEYNES JANUARY 1983 WORKSHOP

This was a three day event and in terms of numbers and input was the best to date. Nearly 150 turned up, lectures went until Saturday evening 9.30 and Sunday till 5.30 pm. The food was by a local caterer, and accomodation either at a local youth hostel or at a local bed & breakfast home in a former's doctor's large house.

Friday saw members arrive eventually at the rearranged Place in the city centre, where we used the Library until tea. Derrick Rowe arrived a little late and Leon kindly filled in for him, and Jim Norris followed Derrick up to tea. Derrick covered 'Programming for a database application' and Jim very kindly gave two case studies on current applications he is working on. In the evening Jim continued on the current equipment in the business sphere and some of their potential uses. Leon gave an overview of CP/M. Colin Ashford explained the range of software in the CP/M Users library and Nigel Hyde finished with a forum on the current Tandy range of equipment.

Saturday saw chaos! We have never had so many systems together all arriving at the same time. By tremendous team spirit, and me getting rather bossy, the systems room looked great and sounded even better. Our guests arrived from Belfast the previous day and one in fact used public transport, and walked from the Central Milton Keynes station to the library - and met a member from his home town. Another guest was a wheel chair bound person, Beryl, who kindly came along and kept order in the canteen, attended the lectures and met some very friendly members. She thanks everybody. The programme ran according to the published agenda with Laurie switching with Mack Leduc from Nottingham, the AGM in the morning lasting two hours, followed by lectures till late Saturday evening.

The lecture I sat in on was that by Clive Keech on using OPCH-80 and going through a score of music, explaining the simple steps along with a 13 page handout. Once again I was enthralled, and the handout was excellent enabling progress to be made at a faster rate than possible in the past sessions. The handout is available at 75p, and depending on demand there is no reason why Clive cannot goahead and produce another 13 pages on a variety of scores. The score I have in mind is 'Mood Indigo' by Duke Ellington with perhaps several working on this, at another meeting spread over a whole day. If we can do 8 hours on Machine Code with Laurie Shields, then we can have 8 hours on OPCH-80 with Clive & John Bedford. I would not miss that one.
to b ntinued Brian Pain

BALROG ADVENTURES **Games by mail**

We have the franchise to run the products of a leading US adventure gaming Company. This gives us the sole UK rights to market and run their play by mail adventure and science fiction games. As a result, a unique opportunity exists for a TRS 80 III owner to earn cash in his/her spare time.

T A N D Y C O M P U T E R U S E R (P A R T T I M E)

We are looking for one or two TRS 80 model III owners to run custom-written programs on a spare time basis. Anything from one day a week, or even less, could be quite sufficient. No programming expertise is required because we supply only well tried and tested software, but an interest in hobby gaming is absolutely essential.

The positions are flexible to meet the spare time needs of candidates, but reliability must be assured. For the right people a partnership situation is envisaged in the long term. However, apart from having the hardware and standard consumables (paper, discs, etc) no further capital outlay would be expected.

Please apply in writing in the first instance, giving us brief details of yourself, your interests and your spare time availability, together with a description and ages of your equipment. Or send sae for more information...

BALROG ADVENTURES **39, Chapman Avenue, Maidstone, Kent, ME15 8EJ**

CAMEL MICROS

33A COMMERCIAL ROAD, WEYMOUTH, DORSET

SOFTWARE

DISCOPT

Adds 10 extra BASIC commands for disc access (see August ad). Personalise discs, find disc free space, lock-out tracks, etc. all with a single Basic command. Also suitable for developing your own 'ZAP' programs in Basic. Only works with Model 1 single dens. at present
Price £9.90

FILEMASTER

For disc users, this program can map almost any disc file format (fixed record size, variable record size, part fixed part variable, or serial files) into any other. Data on two input files can be merged to the output file, and arithmetic relationships can be set up. Ideal for linking data across programs, or producing listing files or command files to drive other programs. Capable of producing automatically, for example, a file which can be listed on a printer, used in Scripsit, or used as a command file, direct from a Visical /DIF file. All redundant data is stripped, full control over format of listing is possible, and titles (or commands) etc. can be included horizontally or vertically. Also has a manual mode for examining file structure, and manually extracting data onto a different file.
Price £11.90

FLOPPY TAPE BACKUP

This allows wafer backups to cassette. All file types are handled (Data, BASIC, and M/C). Options include to/from cassette, verify, triple speed cassette baud rate, double saving of sectors, and printer listings of progress.
Price £9.50 (32K minimum)

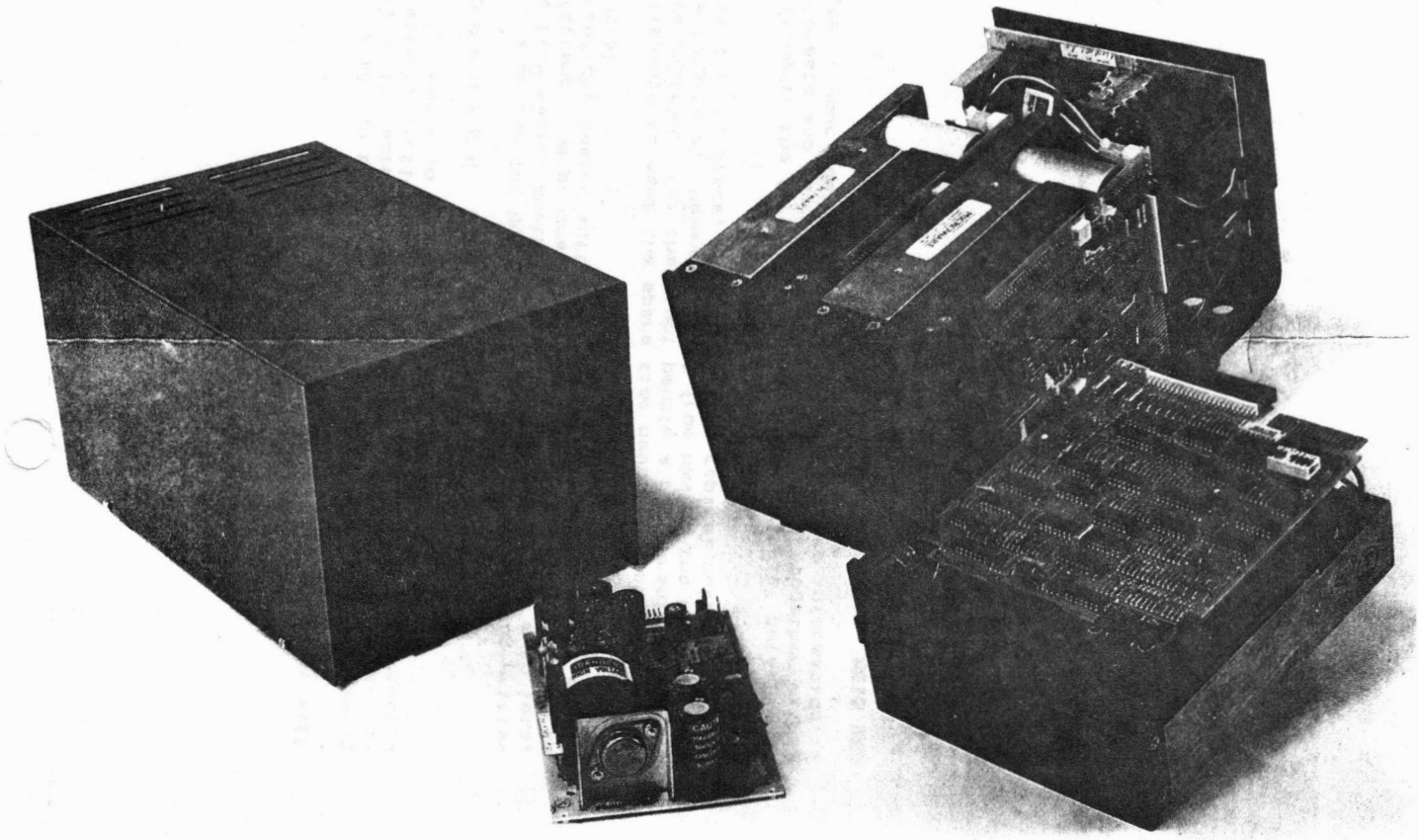
HARDWARE

OKI M80A PRINTERS	£255
SMITH/CORONA DAISYWHEEL (Sale price)	£460
(State whether RS232C or Centronics style required)	
EPSON MX80 FT/3 Special ffs for User group	£362
GP100A	£223
PRINCE HIGH-RES (24MHZ) MONITORS	£100
QUALITY BLANK-LABEL CASSETTES. 5 for	£2.50
(C15, using HM-1260/1290 tape)	

ALL PRICES INCLUDE V.A.T. AND UK CARRIAGE.

Microware (London) Ltd.,
637a Holloway Road, London N19 5SS.
Telephone 272 6398/6237

MICROWARE ANNOUNCE THE Z/L RANGE OF FLOPPY DISC SUB SYSTEMS



The Control Data Family of Flexible Disk Drive is a single- or double-sided, random-access, low-cost storage device. Maximum storage capacity 1 megabyte on a 133.4-millimetre (5.25-inch) interchangeable disk. This disk drive is interchangeable with comparable products, providing industry compatibility in size, mechanical mountings, electrical interface, power requirements and physical appearance. ...® FDD operates in single- or double-density formats. Single-density ...ration is achieved by using Frequency Modulation (FM) encoding, and in double-density operation uses Modified Frequency Modulation (MFM).

OPERATIONAL CHARACTERISTICS

Capacity	9408	9409	9409T
Unformatted			
Per disk	250.0 kbytes	500.0 kbytes	1 megabyte
Per track	3.1 kbytes	6.2 kbytes	6.2 kbytes
Formatted (16 sectors, 128/256 bytes)			
Per disk	163.84 kbytes	327.68 kbytes	655.36 kbytes
Per track	2.1 kbytes	4.1 kbytes	4.1 kbytes
Code	MFM	MFM	MFM
Transfer Rate	125 kbits/s	250 kbits/s	250 kbits/s
Average latency	less than 100 ms	less than 100 ms	less than 100 ms
Seek Time			
Track to track	less than 5 ms	less than 5 ms	less than 5 ms
Average Access	less than 80 ms	less than 80 ms	less than 132 ms
Setting time	less than 15 ms	less than 15 ms	less than 15 ms
Head Load Time (OPT)	less than 50 ms	less than 50 ms	less than 50 ms
Media	hard/soft sector	hard/soft sector	hard/soft sector
Rotational Speed	300 r/min		300 r/min
Track Density	48 TPI		96 TPI
Flux Reversal Density (track 39, side 1)	5876 FRI		5922 FRI
Number of Tracks	40		80
Inner recorded radius (side 0)	1.437 in (36.50 mm)		1.385 in (35.2 mm)
Outer recorded radius (side 0)	2.250 in (57.2 mm)		2.250 in (57.2 mm)
Inner recorded radius (side 1)	1.354 in (34.39 mm)		1.344 in (34.1 mm)
Outer recorded radius (side 1)	2.167 in (55.0 mm)		2.167 in (55.0 mm)

THE Z/L RANGE

Type	Capacity	Tracks	No. of Drives
ZL141	250 k	40	1
ZL142	500 k	40	2
ZL241	500 k	80	1
ZL242	1 Mb	80	2
ZL291	1 Mb	160	1
ZL292	2 Mb	160	2

Microware Disk Drive Subsystems are plug compatible with the following:–
TANDY, VIDEO GENIE, BBC MODEL B, NABCOM, SUPER BRAIN, IBM/PC. and many more.

Microware also supply floppy drives, Winchesters and sub systems to OEM's and the trade at very aggressive prices, offering the best price performance ratio available. Call and ask for details.

CABINETS AND PSU

Cabinets of steel, aluminium or plastic are available in a choice of colours, and each unit is fully guaranteed.

AC 8151 40 WATT SERIES

The AC 8151 Switch-Mode Power Supply has been designed for use in small terminals and other similar equipment. The AC 8151 has dual line inputs and regulated outputs of: +5V at 2.5A, +12V at 2.0A, −12V at 0.1A. This compact unit features low magnetic radiation and is built to conform to International Safety and RFI Regulations.

Name _____

Address _____

_____

YOUR LOCAL DEALER IS:–

NATIONAL TRS-80 & GENIE USER GROUP Newsletter

VOLUME 4 ISSUE 8 FEBRUARY 1983

CONTENTS

2. Information on the Group
3. Editorial
3. Guidelines for contributors
4. Members' letters
8. TRS-80 Educational Users Group
9. Tandy to support CP/M on Model II, 12 and 16
10. Thanks again Brian and all for another great weekend
15. Cassette-based mail list
19. Relative calls to assembly language routines
20. FORTRAN tutorial
25. A review of "FASTER"
26. Keyboard input
27. More keyboard input
28. Graphics editors
31. MK Workshop report (Part 2)

NATGUG

INFORMATION ON THE GROUP

Membership of the Group is by subscription to the Newsletter, which is published monthly. Membership details are obtainable from the Group Secretary. Membership of the Group is open to anyone with an interest in the TRS-80 range of microcomputers, and compatible systems such as the Video Genie.

Details of the Group accounts and the constitution of the Group are obtainable from the Secretary.

Members requiring assistance with problems related to the TRS-80/Video Genie may call the Secretary. An attempt will be made to put them in touch with a member who can help with the problem.

Workshops are arranged from time to time in various parts of the country.

Sub-groups exist in many areas. A list is provided at the back of the Newsletter from time to time. Sub-groups also exist for medical and educational users.

The Group runs 80-NET, a computerised bulletin board service that is available to all users of microcomputers.

The Group maintains three software libraries (Models I, II and III) which are free to members. Library lists are obtainable from the Secretary. The medical and educational sub-groups maintain their own libraries.

A membership list is obtainable on disk from the Secretary.

Back numbers of the Newsletter are available.

Please send all contributions for the Nesletter to the Editor.

Chairman and Newsletter Editor:-

Leon Heller,
8 Morris Walk,
Newport Pagnell,
Bucks. MK16 8QD.
Tel: (0908) 613004

Secretary and Newsletter Publisher:-

Brian Pain,
24 Oxford Street,
Stony Stratford,
Milton Keynes.
Tel: (0908) 564271

Librarians:-

Model I: Leighton Davies, 105 Caerau Road, Caerau, Glamorgan. Tel: (0656) 738337.

Model II: Jim Hutton, 25 High Street, Stroud, Glos. Tel: (045 36) 4423.

Model III: Nick Clay, 6 Rectory Lane, Byfleet, Surrey. KT 14 7LL. Tel: (09323) 42348.

EDITORIAL

Sorry about the missing details from the front page of the last issue. They were there when I gave the front page to Brian. He somehow chopped the top off my sheet when pasting it up.

I was rather short of contributions for this issue, hence the somewhat reduced size. This also accounts for the lateness - I was hoping for some more material to arrive.

The launch of Micronet, the Prestel-based system for microcomputer users, has been put back again. TRS-80 users should be able to get their adaptors by the middle of the month. I have asked the Micronet people to place our software library in a closed section accessible only to group members. This section can also be accessed by members of other user groups, but I don't think that matters too much.

The Southern Group were very keen for us to hold a workshop in Swindon. This commences on Friday 18 March and lasts all weekend. Contact Brian for further details.

Leon Heller

GUIDELINES FOR CONTRIBUTORS

1. If possible, send material in printed form.

2. Scripsit or Pencil disks/cassettes are acceptable.

3. Ensure your ribbon is in reasonable condition.

4. Printer output should be on A4 paper if posssible.

5. If you send in handwritten material, write legibly.

6. Do not fold the sheets when posting them. Use a large envelope and keep them flat.

Leon Heller

MEMBERS' LETTERS

From Dr. M. J. Molloy

It has been good to have been kept in touch via the Newsletter.

I thought the UK group may be interested in my first impressions of a MODEL 16.

I spent 2 months in America on my way home to New Zealand and through spending the majority of my time on business I had a good opportunity to sample not only great American hospitality but the relative advantages that some consumer products have in price over England.(Not all,I might add).

Tandy prices however make one drool.The Mod.16 with 128K RAM,2 Thinline Quad Density double sided disks drives and the dual procesors cost just under $5000:00 (say #2800:00).To get the good bits over,the design seems clever with the Z-80 handling I/O and the 68000 chip supposedly doing all the internal processing at 6 MHZ.The advantage to the user is that the the range of Mod 2 software can be used so Tandy have the drop on the market with the first 16/32 bit machine available that actually has programs that can be run.

To me,brought up on my beloved Mod 3,it is an awesome experience to see the 153rd track formated on Drive 0,to see 4093 SECTORS shown after <FREE> (approximately 1,257,000 bytes) and know there is the same vast amount of space needing filling on Drive 1.

Most commands will be familiar to Mod 2 users,with the options of running programs under TRSDOS 2.0 (they are converted to run with the Thinline Drives or can be run as is,provided Dr.0 contains THINLINE) or TRSDOS 11 (which is effectively the 4.1 operating system written for Hard Disks and is much more space efficient than 2.0.)or TRSDOS 16 can be used.This occupies some 48K of memory and uses both processors.

SCRIPTSIT must be run under 2.0 but most other programs can be FCOPY'ed to TRSDOS 11 formated disks to take advantage of the enhanced operating system and double sided disks.

That is a summary of the good bits but unfortueately there is more to be said.

This computer is S-L-O-W.Unbelieveable for a 6 MHZ 68000.Booting takes forever and FORMATing is virtually a full days work (that's a little exagerated maybe).It appears that the 68000 is just sitting there waiting for some software to be written.Tandy's defensive comments about the multiple VERIFIES of everything,don't take into account the at least equally reliable and superfast DOSPLUS for Mod 2/3 or the speed and Power of LDOS for Mod 1/3.

Servicemen (with whom I have had a lot to do -more later) offer the opinion that this is just a cheap and not particularly elegant upgrade of a Model 2.).TRSDOS 16 may be powerful but the only software is an Editor-Assembler (supplied).There is no BASIC though rumours abound in the States and a request for BASIC in Mod 16 mode brings up a

PASSWORD PROTECTed message.After the Model 3 under DOSPLUS and LDOS I find the TRSDOS 11 BASIC clumsy,particularly in regard to Editing and have found both DOS and BASIC to be non-user-friendly.(It's about time Tandy's Error message format was altered from it's unhelpful present "ERROR xx" for one thing).

RELIABILITY:(The marks on the paper are tears).In the 4 months I have had the MOD 16 I have had the use of it for maybe 4 weeks.The Drives were badly out of alignment,there have been 4 faulty chips found and (to quote my first servicing account) ".....some PCB boards had a strange coating ...Head load pressure faulty both drives,...fault found in Video,Keyboard interface " and a link between Drives 0 and 1 was missing completely.Later the Drives refused to read disks formated on the other and were again realigned at crazy costs for a new computer.

The pricing policy of the Tandy agents in New Zealand is even more iniquitious than in the UK and,having purcahsed the computer in America,even the short and nasty Tandy warranty was refused here.

On an unrelated matter,I think it has been generally announced now so I can tell you that Logical Systems (LDOS) are to provide the operating system for Tandy for the Model 3 Hard drive.After playing with one of these babies I can tell you that the improvement in operating efficiency is of the order of magnitude between cassette and disk.A Model 3 and a Hard Drive with a good operating system - who could

ask for anything more ?

Dr.M.J.Molloy

What are members paying for their listing paper? I buy from D.W. (Direct Wholesale) Ltd., Media House, Kingsbury Trading Eastate, Kingsbury Road, London NW9 8SL, tel. 01-206-3476. They charge #3.78 + VAT for 1,000 sheets of 1-part 60 gsm tractor mounted 12" * 9 1/2", sold in minimum quantities of 2,000 sheets. If you are buying in small quantities, you order by phone and collect yourself; in large quantities, they deliver. They also sell a wide range of computer stationery, multi-part forms, etc. Good quality A4 typing paper is around #8 per 1,000. I should be grateful to hear from members who can obtain paper at lower prices.

Christopher Currie

From Don Bannister

Having unfortunately had to rush off at the end of the AGM without saying a quick farewell, I am just writing a little note to thank you very much for all your efforts in making our first AGM/weekend so enjoyable. It was nice meeting the faces for the names that we knew, and having so many experts available. In addition, both Trevor & myself wanted to include a thank-you for the use of your "eprom washer" - we really honestly hadn't expected so much interest.

For the record, we are getting on fairly well with the Am7910 modem, with a couple of the PCB set already laid out, built and working, and most of the rest of the design work complete. All being well, the arrival of some full spec chips in a several weeks time will enable us to produce a few samples for field testing. Hopefully there will be an article for the magazine, and maybe something for 80-NET. I'll probably be in touch by phone shortly to get the new phone number for the service.

Don Bannister

Dear Leon,

I am giving up my membership of the group as I am now completely saturated with information and ideas which I have no time to explore. My machine has become little but a (most useful) word processor.

I should particularly like to thank you, Brian, Os House and John Carne plus the rest of the members in general for for all the help and fun I have had.

(* I hope no-one else is in a state of informational overload! LFH *)

FOR SALE

ACULAB FLOPPY TAPE + 40 WAFERS + SOFTWARE.
£130 OR OFFER.

JIM DUNLOP
01- 940 1615

From Dick Gays

Keith Bedford's machine code listing for the Genie LE - 18 Hires screen dump in the November newsletter was most useful, though with no previous experience in changing machine code programmes, it took a long time to get it sorted out to run on my Epson MX - 80 printer.

If any other member has the Lowe Hires and Epson printer I will gladly supply copies or listings to the two programmes that give a hard copy from the screen in either 3.3" x 2.6" or 6.6" x 5.2", or can forward these to the Library if required. Printing time for the full screen dump in the larger size is 1 min 51 sec, less for the smaller size.

Dick Gays,
55 Conaglen Road,
Aylestone,
Leicester LE2 8LE

From J.R. Jones

Just a brief note to go with my annual subscription. I find your Newsletter very helpful in learning more about my TRS80 Mod 3.

I've recently bought a Microfirm High Resolution Graphics Kit which I installed in my Model 3. It is rather fiddly work and took me several hours, and phone calls to the producers before it was functioning. The firm was very helpful however and able to direct me in finding the fault over the phone. The graphics produced are fairly good and there are some programs that come with the kit on a disk and assist in making high resolution characters, and in line drawing, point setting, etc. One limitation I found was number of character produced was restricted to 128, which is the number of character codes added, which quite severely restricts the complexity of screen patters generated. For instance, in drawing random lines on the screen, the program stops after a short time, with an 'out of Memory' error message on the screen. This is because all the available new characters have been used up. On the positive side it seems quite good at drawing line graphs, and is not too slow at this.

I hope this has been helpful to anyone considering upgrading their Model 3 to high resolution.

I would also like to ask whether it would be possible connect a smaller micro. such as a Sinclair Specrum to a TRS-80 via an RS232 interface and use the internal disk drives of the TRS-80 Mod 3 with the Spectrum. I would be interested to hear if anyone has any experience with this.

J.R. Jones

TRS-80 EDUCATIONAL USERS GROUP

Latest Information on the Educational Users Group

February/March 1983.

The Educational Users Group grew out of a meeting of half a dozen TRS-80/Video Genie Users at the first National Primary School Computer Conference at Exeter in April 1981. It is part of the National TRS-80 Users Group.

The basic aim of the group is to support the use of the TRS-80 in education. This is done in a number of ways.

HELP AND ADVICE.

Help and advice are normally just a telephone call away. Within the group there is tremendous technical expertise covering interfacing, sound, graphics, add-on, word-processing, database handling, software design and many other things.

We have members from all fields of education - Nursery, Infant, Junior, Secondary, Further, and Higher. Plus of course many home users interested in the educational potential of their machine.

SOFTWARE LIBRARY AND EXCHANGE.

An extensive library has been established and members are able to choose programs from this library. It contains many hundreds of Primary and Secondary programs many of which are in use regularly in schools accross the country. The library contain computer assisted learning packages, simulations and programs for administration. Programs are only accepted into the library on the stipulation that they may be copied and exchanged with other users. No commerical or copyright material is allowed.

SOFTWARE DIRECTORIES.

The Educational Users Group also publish a number of software directories listing programs available from the extensive library of educational software available to members of the group.

DIRECTORY FOUR is about to be printed and will include the usual range of programs that will be of great interest to members. To give some idea a few of the programs included are listed below.

Element Bingo
Astruc
Word Learn
Guess the Missing Word
Abacus
Length Area and Volume
Sound Effects Creator
Picture Book
Equilibrum
French Verbs
A word Processing Pack
PILOT
Balancing Forces
Atomorb
Word Fill
Domino Count
Bracket Sums
Tell Time
ComPass
ElemPhys
Home Economics
French Phrase
LOGO
A Maths Adventure

DIRECTORY THREE was Published in September 1982, and listed a full range of Program created by members, it included some excellent new Primary and Secondary material, Plus some statistical and utilitity Packages. Altogether 96 Programs were available through this directory.

DIRECTORY TWO was Published in February 1982.and It lists 160 Programs.

DIRECTORY ONE was Published in October 1981 and reprinted in July 1982. It is about to be reprinted again. It lists a further 120 Programs.

Each directory is available to members of the Group for two Pounds each Plus 25P. Postage and Packing.

Programs can be supplied on tape to members of the Group and the range available in the library covers.
i. Computer Assisted Learning
ii. Simulations and Strategy Games
iii. Computer Studies demonstration material
iv. Utility Programs for Programming

You can Join the Educational Users Group and obtain these directories by contacting

Dave Futcher
Beaconsfield Combined School,
SOUTHALL,
Middlesex, UB1 1DR.

TANDY TO SUPPORT CP/M ON MODEL II

Tandy recently announced that they will shortly be supplying CP/M Plus on the Model II, 16 and 12. CP/M Plus is the latest version of CP/M that supports bank-switched memory ib a single-user environment.

Leon Heller

Thanks again Brian and all for another great weekend.

It seems ages since I last wrote anything for the magazine. I suppose I could blame Leon for not being his old provocative self but that would be unjust (and anyway we don't know what he might be up to in the next issue).

Believe it or believe it not I penned that back in January and then never got round to finishing it off.

Less than one month later appears Leon's undated broadsheet plugging his usual outdated CP/M 8080 style of programming. Hasn't he learnt yet that whenever one writes general purpose routines, and this applies even more when macros are involved as their details tend to get forgotten, that any registers used ought to saved first and restored on completion. Now we all know that CP/M was written a long time ago before the Z80 was born and hasn't noticeably improved since, but there are other systems about that do use the IX and IY registers, and merrily jumping in waving macros in both hands like Maggie in the Falklands isn't going to do any good if those registers are being used by the system.

Just for the record Leon's macro ILD routine uses 9 bytes and clobbers the IX register's original contents, the same operation can be achieved in conventional style with just 6 bytes and all registers used restored on completion. In other words whilst recognising that as far as some people are concerned 'never mind the quality ... feel the width' seems to be the fundemantal parameter for their basis of judgement I prefer to be with those who appreciate the merits of simplicity in their philosophy. Now back to whatever it was that I was writing about before being so rudely interrupted (a proper assembler indeed . . . he ought to be strung up by his relocatables with his modules plastered with Bdos errors).

In the last magazine but one, someone (if only I could find my copy) was asking how to go back and forth between @Zen and Aculab Basic. Well it couldn't be easier; whilst in @Zen key G (for GO), 114 or 72H (for Address), just <enter> for breakpoint and there you are with Basic's READY > message. To get back from Basic key SYSTEM <cr> and then /23456 <cr>. It couldn't be easier and it is in the manual.

I was shamed the other week when Maxwell Stampe said he had to dig right back to a 1981 edition of the mag for the price of Zen so if Leon will pardon the advertising:

@Zen	for the Aculab	£22.50
Zen	for Level 2 cassette Model 1,Model 3 or Genie	£22.50
Zen	for the Colour Genie (with colour and sound)	£22.50
Zen83	for disk Model 1, Model 3 or Genie (all Dos's)	£44.50

All versions include a comprehensive monitor for manipulating memory, capability to assemble direct to memory and a debugging facility including breakpoints.
The latest cassette and disk versions include system tape reading editing and copying routines, global text replacement, cross reference listing, etc.

For the benefit of anyone out there whose TRS80 hasn't got lowercase may I suggest that they only have half a computer. All program titles and prompts on the screen look infinitely better with lower case as does any printer output. However once lower case is fitted the problem arises of your stock control program not recognising keyboard input like 1( )yz for the part number 123XYZ. This can be most annoying as a routine in Basic to check input and convert any lower to upper can be rather slow and will invariably invoke the string garbage collection freeze up.

Well that was the problem I faced and as a result the following USR routine emerged. In machine code it is easy, once we know where abouts in memory the character string is located and also how many bytes are involved. Microsoft would appear to have had just this sort of problem in mind when they put their Basic together as the function VARPTR tells us all that we need to know by simply including it within the USR brackets.

To get the value enclosed in the USR brackets, in our case the Varptr of the possibly lower case string, we have to call a Rom routine at 0A7FH and the information returns in the HL registers. The information is in fact the address of the first byte of three that give details about the string variable and this one is simply the length of the string. If it is zero then we have a null string and the largest would be 255, which is why the longest string we are allowed is 255 bytes as any more would require more than one byte for the size. The next two bytes give us the address of somewhere different in memory where the string is located and this address is stored in the usual Z80 format of Least Signicant Byte first followed by the Most Signicant Byte.

If we wanted to be bullet proof then we ought to include a test on whether or not the varptr supplied from Basic belongs to a string or some other variable, and this we could do by looking three bytes before the size and checking that the value there is 3, if not then the Varptr belongs to something other than a string. Once we we have checked that the size isn't zero and the actual address has ben loaded into the 'D' and 'E' registers then we go round a simple loop loading the 'A' register with each character in turn, checking if it is outside the range 'a' to 'z' and if not then re-setting bit 5 to 0 by ANDing with 5FH, putting the result back where we found it. On completion we simply RETurn to Basic.

```
 1                  ; LOWCASE
 2
 3                  ; Program designed to be called
 4                  ; as a USR from Basic to convert
 5                  ; any lower-case characters in a
 6                  ; string variable to upper case.
 7
 8                          ORG  8800H  ; Just for testing
 9
10 8800 CD7F0A   START:     CALL 0A7FH  ; To get string pointer
11 8803 7E                  LD   A,(HL) ; Get length of string
12 8804 B7                  OR   A      ; Test for null string
13 8805 C8                  RET  Z      ; If zero bytes then return
14 8806 47                  LD   B,A    ; Put byte count into Reg B
15 8807 23                  INC  HL     ; Point to LSB of string address
16 8808 5E                  LD   E,(HL) ; Load in E register
17 8809 23                  INC  HL     ; Point to MSB of string address
18 880A 56                  LD   D,(HL) ; DE now has address of string
19 880B 1A       LOOP:      LD   A,(DE) ; Get byte from string
20 880C FE61                CP   "a"    ; If less than "a"
21 880E 3807                JR   C,NEXT ; ignore it
22 8810 FE7B                CP   'z'+1  ; If greater than 'z'
23 8812 3003                JR   NC,NEXT ; ignore it
24 8814 E65F                AND  01011111B ; Mask out Bit 5
25 8816 12                  LD   (DE),A ; Put back into string
26 8817 13       NEXT:      INC  DE     ; Point to next char.
27 8818 10F1                DJNZ LOOP   ; Go round the loop
28 881A C9                  RET         ; All done, return to Basic
29                          END
```

To get this routine within a Basic program we convert all the bytes into decimal numbers, eg LD = 12*16 + 13, 7F = 7*16 + 15, etc and write them into a Data statement, but as my system of putting them back into memory requires an even number of bytes we must add a zero at the end giving 28 bytes in total. We are going to store these in an integer array where each value in the array holds 2 bytes so for 28 we need to dimension 13 as we will use the 0 one.

To get the values into the array we multiply the second one of a pair by 256 and add it to the first but checking that the result is not greater than 32767 before allocated it to the integer array. If the result is greater than 32767 then subtract 65536 first. However I don't like these big numbers and as the only way to end up greater than 32767 is for the second number of a pair to be greater than 127 in the first place then this is simpler to sort out.

The result, in Basic, of the statement (M > 127) will evaluate to -1 if true and 0 if not true even without an IF command before it. The expression 256 * (M > 127) will result in either -256 or 0 for true and false. Hence M + 256 * (M > 127) will, in one statement, result in the value M-256 if M is greater than 127 and just M otherwise. So by reading from the Data statement pairs of values into variables L and M (least and most significant) we calculate the combined value as:

256 * (M + 256 * (M > 127)) + L
without an IF or ELSE in sight.

If you are fortunate enough to be using Zen for the assembler then include a LOAD 8800H command just after the ORG statement and then Peek at memory addresses 8800H to 881AH and get the decimal values of the machine code bytes. Convert these into a Data statement and your finished program looks like this:

```
10  DEFINT J,L,M,N: DIM JL(13)
20  FOR N = 0 TO 13
30  READ L , M
40  JL(N) = 256 * (M + 256 * (M > 127)) + L
50  NEXT
60  LINEINPUT"Text string ? ";IN$
70  DEFUSR = VARPTR( JL(0) )
80  M = USR( VARPTR(IN$) )
90  PRINT IN$
100 GOTO 60
110 DATA 205, 127, 10, 126, 183, 200, 71, 35, 94, 35, 86, 26, 254, 97,
     56, 7, 254, 123, 48, 3, 230, 95, 18, 19, 16, 241, 201, 0
```

It is necessary to re-define the USR routine each time before use since introducing other simple variables into the running of whatever application you put this routine to, could cause the actual bytes of memory allocated by Basic for this array to be changed.

Now for the routine that had to be rushed through at Milton Keynes as Leon was pressing to show everybody how useful macros were at gobbling up bytes faster than Pacman. We wrote a simple program to Open an existing disc file for input, Read the file into memory, crashed into the Dos's end-of-file marker and then closed the file. I have since added a call to the Rom video display routine so that as the file is read into memory it is also shown on the screen.

Everything one needs to know about disc accessing via the Dos from a machine code environment is described in Tandy's Dos manual - technical section.

Before a file can be Read, it must be Opened, if it isn't there in the system then we get an error condition.

To Open a file we must give the Dos the Filename, allocate a 256 byte buffer for the Dos to use and we must also tell the Dos how many bytes of data we want to transfer between our program and the Dos at each Read. The Dos of course works in chunks of 256 when physically Reading the disc.
All that information must be collected together with the address of the buffer in HL, the Filespec left justified in a 32 byte block of memory called a File Control Block, (an FCB or sometimes called a DCB, for Device Control Block) and the Record Length in the B register, with DE holding the address of the FCB so that the Dos knows where to find it.

To Read a record then we simply load DE with the DCB, load HL with wherever we want the bytes to go, usually this is called the UREC for User RECord and of course must have as many bytes available as were in the B register when we opened the file, (do not confuse this with the 256 byte buffer we had to start with, that's for the Dos's own mysterious operations). Thus armed we call the Dos's Read routine and wait expectantly. Anything can go wrong of course, such as we hadn't opened it, disc I/O error, we've passed the end-of-file, something has nobbled the data in the DCB, etc, etc. If the Dos isn't happy with whatever we asked it to do then the Flag register returns NZ, not zero, if all OK the it returns Z, zero. So after any calls to the Dos we must always check on the Zero flag.

In our example we will carry on reading the file, moving the UREC higher in memory after each read until we get an error, then we must ask the Dos to please expain the error for us poor mortals and thank the Dos very kindly by Closing the file.

One of the jobs that we don't want to confuse the issue with is that of accepting and validating a filespec from the keyboard, so to save time we will include the Filespec 'TEST' in our program and make sure that there is a text-file caled Test on the disc when we run the finished coding. This is what we ended up with at MK:

```
 1                          ORG   9000H
 2                          LOAD  9000H;    Not possible with M80.
 3                BUFFER:   DEFS  256  ;    For Dos's use.
 4 9100 54455354  DCB:      DB    'TEST'
 5 9104 0D202020            DB    13,"                          "
 5 9108 20202020                            ;  32 byte FCB but Model 3
 5 910C 20202020
 5 9110 20202020                            ;  TRS Dos needs 50 bytes.
 5 9114 20202020
 5 9118 20202020
 5 911C 20202020

 6                UREC:     EQU   0A000H ; Input characters to go here
 7 9121 00A0      ADDR:     DEFW  UREC   ; Keep track of UREC position

 8                OPEN:     EQU   4424H  ; Dos routines.
 9                CLOSE:    EQU   4428H
10                READ:     EQU   4436H
11                DERROR:   EQU   4409H

12                START:    EXEC  START
13 9123 210090              LD    HL,BUFFER ; Prepare to Open.
14 9126 110091              LD    DE,DCB
15 9129 0601                LD    B,1       ;  Logical Record = 1
16 912B CD2444              CALL  OPEN
17 912E 2015                JR    NZ,ERROR
```

```
; If we have got this far then there is a file called Test
; waiting to be Read, so go round a loop
; Reading 1 byte at a time and moving UREC 1 byte higher
; in memory before each Read

   18 9130 2A2191   LOOP:    LD   HL,(ADDR) ; Get Urec address
   19 9133 23                 INC  HL        ; move it up 1
   20 9134 222191             LD   (ADDR),HL ; save for next time
   21 9137 110091             LD   DE,DCB    ; Dos needs this
                                             ; HL => Urec
   22 913A CD3644             CALL READ      ; anything there ?
   23 913D 2006               JR   NZ,ERROR
   24 913F 7E                 LD   A,(HL)    ; Yes so prepare to
   25 9140 CD3300             CALL 33H       ; dispay on video.
   26 9143 18EB               JR   LOOP      ; Keep it up !

; Eventually we end up in trouble here so to get an error message
; from the Dos we must Set bits 6 & 7 by ORing with 0C0H

   27 9145 F6C0     ERROR:    OR   0C0H
   28 9147 CD0944             CALL DERROR
   29 914A 110091             LD   DE,DCB
   30 914D CD2844             CALL CLOSE
   31 9150 00       BRKPNT:   NOP            ; Useful for de-bugging
   32 9151 C32D40             JP   402DH
   33                         END
Exec Addr 9123
```

There is one potential error situation that could arise if this program was tried using Model 3 TRSDos on a file that was created with other than a logical record length of 1. What happens is that the Dos quite happily opens the file ignoring our value in B and then processes the file as it thinks fit rather than the way we want to do it.

We developed an interesting fault on one of the Model 1's the other day. The machine would quite happily boot up the Dos and read any program from the disc. Trouble started as soon as the one used a Basic program that did any disc I/O with the basic program itself becoming garbaged, (for the want of a better word). Visicalc was an absolute disaster but Scripsit worked without any problems and all the various memory testers passed 100 %. The problem seemed to be associated with using the Rom when the discs were turning; any program, such as Scripsit or Dos modules like Format or Copy that do not use Rom coding worked quite well.

Eventually we pinned it down to an inadequacy in the single power supply that fed the whole system. The mains transformer had lost some power and when the discs were drawing juice the voltage to the computer and expansion interface fell. Whilst the fall was insufficient to trouble the Ram, the more sensitive Rom degenerated and sent some duff info to the Z80. Replacing the little black boxes to drive the computer has put everything working again until we get the transformer replaced.

I've had two phone calls from people with vaguely similar difficulties but possibly very different causes. The first chap had a straight forward standard TRS80 and suddenly found that his LDOS wouldn't merge or load Basic programs stored in Ascii format. The machine simply accepting the command and returning with Ready just as if all had been well, but in fact nothing had been loaded into memory. Needless to say both commands worked OK in TRSDOS.

The other problem was with an LNW expansion ard TRS80 that worked OK with TRSDOS but Newdos80 wouldn't format a disc or do a copy, each time packing up on sector 99 with a verify error or the like. Needless to say the same Dos disc using the same drives connected to a different computer worked without a hitch.
So far we don't have an explanation for either problem.

At Milton Keynes there was an interesting set-up of a Colour Genie with connected to an Aculab drive. Once again I can't remember the owners name but please would he give details of the hardware mods required (in very simple language) and also what went into his special connector that linked the Colour Genie output to his standard Tandy monitor.

In a probably futile attempt to stop Leon's stroppy derogatory remarks about Zen I've just spent the last couple of hours writing a macro pre-processor that will allow Zen to do everything, and a bit more besides, that Leon has ever bragged about his preferred assembler. At the moment the program is somewhat slow and cumbersome, a bit like M80 I suppose, but that's the way with macros. It could be easily adapted to disc Edtasm format files and when time permits I hope to get round to it.

Laurie Shields Software. 151 Longedge Lane, Wingerworth
Tel. 0246 75568 Chesterfield. S42 6PR.

(* What a reaction to an innocent little macro! I do not understand Laurie's reference to an 'outdated CP/M 8080 style of programming'. There is nothing particularly inefficient or outdated about macros, they are merely a form of short-hand which can save a lot of typing, and make your programs more readable. My ILD macro was somewhat inefficient because I wrote one single macro to cover indirect register loads for all three register pairs. Laurie would have to type his code out, in slightly different form, for each of the HL, DE and BC register pairs. As for saving the IX register, it is not usual to save registers willy nilly when writing in-line code; macros are NOT sub-routines. Of course, if I need to save and restore the IX register, I merely write the code for this once, in the macro body, rather than repeating it several times through the program, as Laurie would have me do. Moreover, that macro was specifically written for use with Microsoft FORTRAN, which uses 8080 code exclusively, as does Microsoft BASIC, incidentally. LFH *)

CASSETTE-BASED MAIL LIST

This is a cassette based menu that will,
1) Save Addresses and phone numbers.
2) Retrieve the above from tape.
3) Print on VDU or Lineprinter the list.
4) Find individual name. Initials not required.
5) Print on line printer:-
   a) List of addresses.
   b) Individual address.
   c) Envelopes. Address in centre. Tab function.
   d) Labels. Xerox type. Same address, any number. or one address on each label.

```
470 IF J1$=LEFT$(F(A),V) THEN 500
480 NEXT A
490 PRINT"No   surname ":GOTO 60
500 F1$=RIGHT$(F(A),Y-V)+LEFT$(F(A),V):PRINTT(A);F1$
510 PRINTG(A):PRINTH(A):PRINTI(A):PRINTJ(A):PRINTK(A):PRINTQ(A)
520 PRINT"Are these the right initials (Y/N)":INPUT Z$:IF Z$="N" THEN 480
530 IF E=1 RETURN
540 INPUT"Another? (Y/N) ";D$:IF D$="Y" THEN 400 ELSE 60
550 INPUT"When ready press newline key";X
560 E=1:GOSUB 410
570 INPUT"Do you wish to change the Name!(Y/N)";Z$
580 IF Z$="Y" THEN PRINT"Surname First :-":INPUT F(A)
590 INPUT"Change the Title Mr Mrs etc (Y/N)";Z$
600 IF Z$="Y" THEN PRINT"Title :-";:INPUT T(A)
610 INPUT"Change Address :- (Y/N)";Z$
620 IF Z$="N" THEN 670
630 INPUT" 1st Line:-";G(A)
640 INPUT" 2rd Line:-";H(A)
650 INPUT" 3rd Line:-";I(A)
660 INPUT"4th Line:-";J(A)
670 INPUT"Change the Postcode:-(Y/N)";Z$
680 IF Z$="N" THEN 700
690 INPUT"Postcode:-";K(A)
700 INPUT"Change the Telephone number (Y/N):-";Z$
710 IF Z$="N" THEN 720 :ELSE INPUT "Phone :- Numbers only";Q(A)
720 INPUT"Another Name? (Y/N)";Z$:IF Z$<>"Y" THEN E=0
730 IF E<>1 THEN 60 ELSE 550
740 INPUT"When ready press NEWLINE KEY";X
750 PRINT"Load the tape recorder and rewind, dont forget leader.
Put into record mode. Put counter to zero."
760 INPUT"Are you ready (Y/N)";Z$
770 IF Z$="Y" THEN PRINT "What are you going to call tape":ELSE60
780 INPUT"Name of tape";G$
790 PRINT#-1,G$,P1
800 FOR A=1 TO P1
810 PRINT#-1, F(A),T(A),G(A),H(A),I(A),J(A),K(A),Q(A)
820 NEXT
830 PRINT"Recording Finished, Note the counter mark the cassette"
840 INPUT"Back to Menu?  (Y/N)";Z$:IF Z$<>"Y" THEN STOP
850 GOTO 50
860 INPUT"When ready press newline key";X
870 PRINT"Place tape into recorder, rewind to start.
Put into the play mode."
880 INPUT"Ready";Z$:IF Z$<>"Y" THEN 870
890 INPUT#-1,G$,P1
900 PRINT G$+"   ";:INPUT "Is this the tape (Y/N) ";Z$
910 IF Z$="N" THEN PRINT"Find the right one":GOTO 860
920 FOR A=1 TO P1
930 INPUT#-1,F(A),T(A),G(A),H(A),I(A),J(A),K(A),Q(A)
940 NEXT
950 GOTO 50
960 CLS:PRINT"1) For printing on Envelopes"
970 PRINT"2) For printing list of labels 1 each."
980 PRINT"3) For printing  labels, one address."
990 PRINT"4) For printing list on paper."
1000 PRINT"5) Return to menu"
1010 P$=INKEY$:IF P$<"1" OR P$>"5" THEN 1010
1020 P=VAL(P$)
1030 ON P GOTO 1040,1150,1320,1390,60
1040 PRINT"Place envelope in printer.
```

```
10 REM MENU PROGRAM FOR NAMES AND ADDRESS AND TELEPHONE NUMBERS.
20 CLEAR 1000:DEFINT A-D:DEFSTR F-L,T,Q
30 DIMA(50),Q(50)
40 DIMF(50),G(50),H(50),I(50),J(50),K(50),E(50),T(50)
50 CLS
60 PRINTTAB(30)"MENU":PRINT
70 PRINTTAB(5)"1) To Build a File"
80 PRINTTAB(5)"2) To See all Files"
90 PRINT TAB(5)"3) To See Individual Name"
100 PRINTTAB(5)"4) To make Corrections"
110 PRINTTAB(5)"5) To Save Current File to Tape"
120 PRINT TAB(5)"6) To Input File from Tape"
130 PRINT TAB(5)"7) Printing on line printer."
140 PRINT:PRINT"Which do you require 1-7"
150 E$=INKEY$:IF E$<"1" OR E$>"7" THEN 150
160 M=VAL(E$):PRINT M;" ";
170 ON M GOTO 190,340,400,550,740,860,960
180 INPUT"When ready press NEWLINE KEY";X
190 CLS:PRINT"To close File type in FINISH. Instead of name"
200 PRINT"NAMES Have to be printed SURNAME. First  Fullstop.
 then Initials with a fullstop after each one.
 Note! There must be NO! Commas. Take care."
210 PRINT"Addresses have only (4) lines plus
          1) Line for the post code
          1) For the telephone Numbers only"
220 PRINT"For lower case hold down 'SHIFT' key"
230 PRINT:PRINT:FOR A=1 TO 50
240 INPUT"Name:-";F(A):IF F(A)="FINISH." THEN 310 :ELSE INPUT"Title:- Mr & Mrs. Etc)";T(A)
250 INPUT"1st line address:-";G(A)
260 INPUT"2nd line address:-";H(A)
270 INPUT"3rd line Address:-";I(A)
280 INPUT"4th line address:-";J(A)
290 INPUT"Postcode:-";K(A)
300 INPUT"Telephone:-";Q(A):GOTO 320
310 P1=A:A=50:GOTO330
320 IF FRE(X$)<100 THEN 310
330 NEXT:CLS:PRINT"File closed":PRINT:GOTO 60
340 INPUT"When ready press NEWLINE KEY";X
350 CLS:FOR A=1 TO P1
360 PRINTF(A);T(A):PRINT"    "+G(A):PRINT"      "+H(A):PRINT"        "+I(A):PRINT"          "+J(A)
370 PRINT"Telephone:-";Q(A)
380 INPUT"Next press newline";X:NEXT
390 INPUT" Press newline for the next one";X
400 INPUT"When READY press newline key";X
410 CLS:PRINT"Enter the SURNAME 'ONLY' With a Fullstop.Then newline"
420 INPUT J1$
430 FOR A=1 TO P1
440 Y=LEN(F(A))
450 FOR V=1 TO Y:Y$=MID$(F(A),V,1):IF Y$="." THEN 470
460 NEXT
```

```
Make sure the first line
One third down the envelope.
TAB Stops are 8 letters apart
1050 PRINT"!    !       !       ! etc
Normal will be 1 or 2 "
1060 INPUT"How many tab stops are required ";U
1070 U=U*8
1080 E=1: GOSUB 400
1090 LPRINTTAB(U);T(A);" ";F1$
1100 LPRINTTAB(U+4);G(A)
1110 LPRINTTAB(U+6);H(A):LPRINTTAB(U+9 );I(A)
1120 LPRINTTAB(U+12);J(A):LPRINTTAB(U+12);K(A)
1130 INPUT "Any more :-";Z$:IF Z$="N" THEN E=0
1140 GOTO 50
1150 INPUT"Label strip in printer at Left Edge (Y/N) ";Z$
1160 IF Z$<>"Y" THEN 1150
1170 LPRINTCHR$(18)
1180 FOR A=1 TO P1
1190 Y=LEN(F(A))
1200 FOR V=1 TO Y:Y$=MID$(F(A),V,1):IF Y$="." THEN 1220
1210 NEXT V
1220 W$=RIGHT$(F(A),Y-V)+LEFT$(F(A),V)
1230 LPRINT"   "+T(A)+" "+W$
1240 LPRINT"       "+G(A)
1250 LPRINT"          "+H(A)
1260 LPRINT"            "+I(A)
1270 LPRINT"              "+J(A)
1280 LPRINT"               "+K(A)
1290 LPRINT" ":LPRINT" "
1300 IF W=1 RETURN
1310 NEXT A
1320 PRINT"Are the labels to the left of printer"
1330 INPUTZ$:IF Z$<>"Y" THEN 1320
1340 E=1:GOSUB 410
1350 INPUT"How many labels ";Z
1360 W$=F1$:FOR X=1 TO Z:W=1 :GOSUB 1230
1370 NEXT X
1380 GOTO 50
1390 PRINT"Paper in printer, printer turned on ":INPUT"(Y/N) ";Z$
1400 IF Z$<>"Y" THEN 1390
1410 LPRINTCHR$(15)
1420 FOR A=1 TO P1
1430 LPRINT F(A)+T(A);
1440 LPRINT G(A);LPRINTH(A);LPRINTI(A);LPRINTJ(A);LPRINTK(A);LPRINTQ(A);LPRINT" "
1450 NEXT A
1460 LPRINTCHR$(18)
1470 GOTO 50
```

Mr. T.J.Rowe.
57 Somerstown.
Chichester.
W.Sussex.

PO 19 4AL.

TAB can be changed to give different envelope address.

RELATIVE CALLS TO ASSEMBLY LANGUAGE ROUTINES.

REF JOHN COUCH'S ARTICLE IN THE FEBRUARY NEWSLETTER (P.22), THERE IS IN FACT A ROM SUBROUTINE AVAILABLE IN THE TRS-80 (AT 13E5H) WHICH ENABLES THE SAVING OF THE HL REGISTER WHILE SIMULTANEOUSLY GETTING THE PROGRAM COUNTER. IT OCCURS ONLY ONCE IN THE ROM AND WAS HARD TO FIND. THE S/R EXCHANGES HL AND CONTENTS OF SP THEN RETURNS TO THE NEXT INSTRUCTION IN THE PROGRAM (REL JUMP).

THE ADVANTAGE IS THAT THE HL REGISTER CAN BE USED FOR OTHER PURPOSES AND IS AUTOMATICALLY PROTECTED (SAVED). THE FOLLOWING LISTING ILLUSTRATES A SIMPLE USE OF THE PRINCIPLE (BUT DOESN'T USE HL FOR ANYTHING ELSE, ALTHOUGH IT COULD HAVE DONE).

MAC RANDLE.

```
00140 ;-------------------------------------------------------
00150 ; * RELCAL *                                   MJR  1-12-81
00160 ;
00170 ; TO CREATE A 'CALL RELATIVE'.
00180 ; FOR RELOCATABLE M/C PROGRAMS.
00190 ;
00192 ;METHOD :-     (MAIN PROGRAM)
00194 ;              (      ''      )              NOTE :-
00200 ;              CALL     13E5H               CALL TO ROM.
00210 ;              JR       SUB                 ----------------
00230 ;               (CONTINUE)                   13E5H  EX (SP),HL
00240 ;               (   ''   )                   13E6H  JP (HL)
00250 ;                                            ----------------
00260 ;    SUB       INC      HL
00262 ;              INC      HL
00270 ;              EX       (SP),HL
00280 ;               (SUBROUTINE)
00290 ;               (     ''     )
00292 ;              RET
00300 ;
00310 ;-------------------------------------------------------
00320 ;
00330 ; EXAMPLE :  TO PRINT CHRS  1  2  AND  3  ON SCREEN.
00340 ;
00350 ;   (ASSEMBLE IN MEMORY THEN RUN VIA 'EDTASM-PLUS' ETC)
00360 ;
00370 START   PUSH     DE              ;SAVE DE
00380         LD       DE,3D30H        ;VIDEO
00390         LD       A,31H           ;ASC 1
00400         LD       (DE),A
00410         CALL     13E5H           ;HL TO STACK.  PC TO HL.
00420         JR       SUB             ;REQUIRED SUBROUTINE
00440         INC      DE              ;CONTINUE PROGRAM
00450         LD       A,33H           ;ASC 3
00460         LD       (DE),A
00470         POP      DE              ;RESTORE DE
00480         JP       4386H           ;TO EDTASM+
00490 SUB     INC      HL              ; ) POINT TO RET ADDR
00500         INC      HL              ; )   IN MAIN PROGRAM.
00510         EX       (SP),HL         ;PC TO STACK. RESTORE HL.
00520         INC      DE              ;NEXT VIDEO
00530         LD       A,32H           ;ASC 2
00540         LD       (DE),A
00550         RET                      ;TO MAIN PROGRAM
00560         END      START
00570 ;-------------------------------------------------------
```

1.0 INTRODUCTION.

FORTRAN is the daddy of the high level languages and has been around since 1957. It was I.B.M.'s first attempt at a high level language and it is a tribute to the engineers, lead by John Backus, that it is still in use and flourishing today. There is not suprisingly a considerable amount of software around in the language but it is almost exclusively in the scientific and engineering fields (though it is interesting to note that the original Adventure was written in FORTRAN). The reason for this is quite simple; FORTRAN stands for FORmula TRANslation and this is what it was primarily aimed at. That is the substitution of numerical values into algebraic expressions and either displaying them on a terminal, listing them to a printer or saving them to a file.

There have been a number of versions of FORTRAN over the years. The standard FORTRAN version being FORTRAN II, this was improved upon in 1966 and this was adopted as a standard to become FORTRAN IV. This is the standard used in the TRS-80 and other Z-80 FORTRAN systems with minor improvements. However there has been a further standard adopted in 1977 which has become known as FORTRAN 77. This is a particularly nice version and the one I have had most experience in. It is a great pity that it is not available for the TRS-80 (as far as I know it is only available under the USCD-p system). All versions are upwards compatable, that is you can take a FORTRAN IV programme and run it on a FORTRAN 77 compiler.

1.1 The Language.

FORTRAN is a compiler based system. That is it takes the source code (the original typed in stuff) and compiles it into the object code (i.e. a /CMD file). The latter can be run from "Dos Ready". This takes a long time, compared to BASIC, to perform, but results in faster code. It also results in a lack of line numbers, instead numeric labels are used for transfer through the program. However it does mean only one statement per line although overflow is allowed onto any number of further lines. Lastly it means an END statement is required as the very last statement to tell the compiler when to stop.

2.0 DATA TYPES.

There are several data types in FORTRAN most of which will be familiar to the BASIC programmer.

2.1 Double.

This gives double precision accuracy to the numbers and is signified by the letter D. It is exactly equivalent to the BASIC interpretation, i.e to 16+ significant digits.

e.g

1.6D+2 =160.0000000000000 (notice the D)

345=345.0000000000000 (this will have to be declared)

2.2 Real.

These are the single precision accuracy version and are accurate to 7 significant digits. They again are equivalent to their BASIC (normal, non integer numbers) interpretation.

e.g.

1.6E+2 =160.0000 (notice the E)

345.=345.0000 (notice the decimal point)

2.3 Integer.

These again are exactly the same as their Basic counterparts.
e.g.
345=345 (notice NO decimal point)

2.4 Logical.

Something different at last.This numeric type can either have a value of TRUE (255 decimal)or FALSE (0 decimal) This is a very economic arrangement.

2.5 Complex.

This is the last numeric data type and is NOT implemented in FORTRAN for the TRS-80.This is a major drawback from the scientific point of view.Incidentally if you are wondering what it is then try to take the square root of -1 on a pocket calculator.It will give you an error and this is a data type that overcomes that small problem.(This is a wild oversimplification but this is FORTRAN not maths).

2.6 Hollerith.

This is like strings in BASIC.That is a Hollerith type is a string of any number of characters from the computers character set with blanks being significant.These will be dealt with in the next tutorial as it is not that simple!

3.0 VARIABLES.

These can be of all the types listed above with the exception of Hollerith which are usually treated as integers and have no decaration statement in FORTRAN IV.

The variable name can have a maximum 6 significant letters and as with everything in FORTRAN upper - case letters only.The variables begining with the letters I,J,K,L,M,N are already (or implicitly) defined as integers the rest being considered real.In order to involve the other data types a declaration statement is required (c/f DEFINT or similar in BASIC)
e.g

```
INTEGER FRED,BILL  (makes FRED & BILL integer variables)
REAL JEFF,IAN      (makes JEFF,IAN real variables)
DOUBLE BOSS        (makes BOSS a double precision variable)
LOGICAL DAVE       (makes DAVE a logical variable having a
                    value of true or false)
```

There are variations on the theme but for the moment let leave it at that.

4.0 Expressions.

4.1 Numeric.

These are evaluated with exactly the same order as Basic and there are very few pitfalls here.However you must remember that exponentiation in BASIC is not the up arrow but the double asterix **.
e.g
2 up arrow 3 = 2 cubed =2 ** 3
Also although some forms of FORTRAN allow mixed mode arithmetic (i.e mixing reals with integers or reals and doubles etc.in the same expression) it is bound to cause your downfall some day.

4.2 Logical and Relational.

In contrast to the previous section this is "all change".The usual logical expressions are allowed but they all

have dots around them.This gives rise to the expression that you need to be dotty to learn FORTRAN which is totally unfounded - just mad!This will be summerised in the following table.

MEANING	FORTRAN EQUIVALENT
true	.TRUE.
false	.FALSE.
logical NOT	.NOT.
logical AND	.AND.
logical OR	.OR.
logical XOR	.XOR.
less than	.LT.
less than or equal to	.LE.
equal to	.EQ.
not equal to	.NE.
greater than	.GT.
greater than or equal to	.GE.

The logical operator .NOT. can be combined with the others to produce NAND i.e. .NOT. .AND. but the others can not be combined.The usual order of preference or hierachy is followed.

5.0 CONTROL STATEMENTS.

These are very similar to BASIC and so no head scratching should be caused by these few:-

```
GO TO label       where label is a number from 1 to 99999
GO TO (label 1,label 2)j   dependent on j (integer) it
                           will go to label 1 or label 2 i.e
                           if j=1 goes to label 1
                           if j=2 goes to label 2
                           ANY number of labels may be used.
IF (A) label 1,label 2,label 3   i.e A .LT. 0 = label 1.
                                     A .EQ. 0 = label 2.
                                     A .GT. 0 = label 3.
                                   THREE labels maximum.
If(......)statement      If the expression inside brackets
                         is true then executes statement.
DO label 1 integer variable = start,end,step size
This is a loop instruction
e.g.
DO 10 I=1,30,2  is equivalent to: 5 For I=1 to 30 step 2
                                  10 Next I
```

It is convention in FORTRAN to have a statement called CONTINUE as the statement at the label and to indent loops like PASCAL.CONTINUE is a statement that does absolutely nothing.

e.g.

```
       DO 10 I=1,10
       ..............
       ..............
       ..............
10      CONTINUE
```

This will execute the statements between the DO and the CONTINUE ten times,I going up by one each time until it reaches 10.

6.0 INPUT & OUTPUT STATEMENTS.

These are the most different parts of FORTRAN to BASIC and perhaps the most difficult to appreciate.

6.1 Output Statement.

This is the general WRITE statement put in its simplest form(the more difficult version will be taught later).

WRITE (LUN,Label) variables.

e.g.

```
WRITE ( 5,10 )FRED,JIM
```

Some explanation is required as this probably looks like a load of gibberish.The LUN stands for Logical Unit Number and is that which decides where the output goes .The standard Logical Unit Numbers are :

1,3,4,5 for the keyboard or display (whichever is correct)
2 is for the line printer (if one exsists)
6-10 are for disk files.

The label denoted the place where a FORMAT statement exsists,see 6.3 if you are impatient.This is where life gets complicated and only simple FORMAT will be dealt with here.The variables print out their value as specified by the format statements.

6.2 Input Statements.

These go under the simplified general form of :

READ (LUN,label) variables

e.g

```
READ (5,10)FRED,JIM
```

The Logical Unit number points to the place where the information is being received and the label the Format statement denoting the type of information to be read in by the variables.Incidentally the information in a list should be separated by commas e.g for the above input 2,3 <enter>

6.3 Format Statements.

These are going to be dealt with in the simplest way possible in this tutorial so you can output and input stuff with no formatting etc.

A typical output format is:

```
10      FORMAT (' MARY HAD ',I2,' LITTLE LAMBS')
label           literal   Integer  literal
number          output    number   output
                          2 digits
```

This would output :-

```
MARY HAD 3 LITTLE LAMBS
```

please notice the gap between the single quote and the first letter and that the integer has a 2 digit range.

A typical input format is:

```
10      FORMAT (I2)
```

This will read in an integer of a maximum of two digits with the negative sign counting as one digit.

7.0 FORTRAN ON THE TRS-80.

The source file (the stuff we have been refering to) is created using the EDIT part of the package by typing EDIT <enter> and then the filename/FOR <Break>.Proceed by typing I and 100 will come up.Type in the code in the following form :-

```
-----&.....................................................
label                   statement
```

The & is where the continuation mark should be placed i.e if the line is too long to fit onto one line then put any character,as long as its not a zero,and the line will be continue from the previous.If there is no label number then use spaces to go past the label & continuaton columns.
C in the first position or column denotes a Comment line (c/f REM in BASIC).

Don't forget that the END statement should be the last statement in the source file.

When this is finished type E <enter>
To check whether there are any errors type :-
F80 =filename

If there are no Fatal errors (stops compilation) or Warnings (does NOT stop compilation but shows you something is wrong)

Then type
F80 filename,filename=filename
L80 filename-G (if you would like to see it run)
L80 filename-N,filename-E
You can now run the program from "Dos ready"by typing the filename.
For an explanation of what is happening I suggest you refer to pages 6-8 of the TRS-80 FORTRAN Manual (not the Microsoft bit).

8.0 QUESTION.

I am going to pose a question whose answer will be given in the next tutorial to see if you have grasped this tutorial:-

Suppose you have two girlfriends, Mary and Jane who have established minimum standards for the men they date.
Mary's standards: 6 foot tall and decent looking or earns 30000 per year and owns a Jaguar or is at university and is very good looking
Jane's standards: Decent looking and owns a car or isn't broke and is very good looking or is an electronics student.
The relevent data are height (in inches),looks (on a scale of 0 to 10),money (£'s a year),car (0=none,1=ordinary, 2=new,3=Jaguar),type of student (0=dropout,1=0 level student, 2=university student,3=electronics student,-1=post graduate student)
The Question is: Write a program to decide whether a man with a given characteristics can date Mary or Jane?
(HINT use logical variables ..... it can be done in 40 lines)

A REVIEW OF "FASTER"

Programmers in the Basic language work hard at trying to speed up their programs. As more is learned of the way the TRS80 handles Basic programs, so more utility aids are available to the programmer to assist in optimising his program for space and speed. Prosoft, of the USA, market a program called "FASTER" which can, they claim, generally speed up a Basic program by 15 - 50 %. It is 'how' Faster does this which is as interesting as the program itself.

The Basic interpreter has to search a variable table each time a variable is referenced in a program. It starts at the beginning of the table and if it does not find the variable, it is added onto the end. To cut down the 'search time' for variables referenced the most, they can be placed at the start of the table. Finding out which are the most commonly referenced variables in a program by analysing it visually is difficult to say the least. This is where Faster comes in, it counts the number of times the variable is referenced and produces a list from most to least referenced.

The program comes on cassette for use with 16K or greater cassette or disk systems. Transfer of the program to disk is very straight forward and it is self relocating. It operates with TRSDOS, NEWDOS NEWDOS80 1&2 LDOS and DOSPLUS. I found no difficulty in operating the program along with other Basic programming utilities such as Compak or with keyboard drivers etc as found in LDOS.

With Faster in memory, and active, load and run a Basic program through it's normal sequences. Pressing three keys (567) together places a mini-menu at the bottom of screen. From this menu you can display the variable table to either video or printer, the latter being more useful. Faster also highlights certain program lines, those with DIM and CLEAR statements. To allocate the variables in their order of 'most referenced' in the program, one line of DIM statements is added to the program. This could resemble something like ---- DIM A$,X%,I,AD,AF$,C(10),N$(10) etc.

Two other important things are the placement of CLEAR and DIM statements in the program. The latter is necessary if you are going to 'DIM' the variables, as in the example line above, because a 're-dim' can not be used. More than one CLEAR statement can also cause problems, but a little program reorganisation can usually surmount the difficulty.

Does it work ? The simple answer is yes, but the time savings may not be all that apparent. Reducing the time of some program operations makes such very small savings that they can only be seen with the aid of a stopwatch. There are operations however, such as sort routines, where the savings are not only apparent but quite obvious. Added to other good programming techniques such as: placing initialisation routines at the end, using multi statement lines and putting the most used routines at the start of a program, can make for greater efficiency.

Faster is an easy to use program with a well written manual and does what it claims. At a price of £17.95 (inclusive) it is quite expensive but if time is important in your program then the price may be worth paying.

Ken Grey

Keyboard Input

There has been recent correspondence (Oct '82 & Jan '83) relating to keyboard input, namely the use of AND when PEEKing address 14400, & are there any more available. In giving some further information I would first acknowledge William Barden Jnr, who in his book - PROGRAMMING TECHNIQUES for LEVEL II BASIC, really describes the keyboard scan facilities in a simple yet effective way.

I would refer to the following diagram which gives the memory address, against which the value of the bit set accordingly determines which key has been pressed.

Memory Address	Bit Set Value 1	2	4	8	16	32	64	128
14337	@	A	B	C	D	E	F	G
14338	H	I	J	K	L	M	N	O
14340	P	Q	R	S	T	U	V	W
14344	X	Y	Z					
14352	0	1	2	3	4	5	6	7
14368	8	9	:	;	,	-	.	/
14400	Ent	Clr	Brk	↑	↓	<--	-->	Sp.
14464	Sft							

On each & every address it is whether a particular bit is set that determines the key pressed, & the program written must take the necessary testing into account:-

```
10 Z = PEEK(14400)
20 IF Z AND 2 PRINT"Pressed Clear Key"
```

this coding is specifically testing for a clear key being depressed. However it is not possible to test whether two keys on the same address are being pressed :-

```
20 IF Z AND 3 PRINT"Enter & Clear Keys Pressed"
```

this is invalid as there will be a response if only one key is pressed.

Testing for two pressed keys is possible provided that the keys are on two different addresses e.g.:-

```
10 Z = PEEK(14400): Y =PEEK(14464)
15 IF Z=2 AND Y=1 PRINT"Shifted Clear pressed"
```

(note that I have used Y=2 & Z=1 rather than the faster BOOLEAN functions, because I have found that (Y AND 2) AND (Z AND 1) not to work even with using brackets) (another note is that you can test for the break key to have been pressed, & the routine will work before it breaks the basic program - so disable the break key by POKEing 16396,165 which still permits "shift break" to operate as "break".)

Before leaving the keyboard it is worth noting the use of INKEY$ in repetitive mode. Normally INKEY$ returns a one character string for a key depressed & will not repeat the key unless it is released & then repressed. Repetition can be achieved by zeroing the appropriate key buffer. There are seven such buffers as per the following:-

Memory Address	Keys
16438	@ - G
16439	H - O
16440	P - W
16441	X - Z
16442	0 - 7
16443	8 - /
16444	ENTER - SPACE

Thus in order to have all the keys H to O repetitive, then after the assignment of the string variable in the INKEY$ function all that is needed is for there to be a POKE 16439. This can be shown as:-

```
10 A$ = INKEY$: IF A$ = "" GOTO 10
20 IF A$ < "H" or A$ > "O" GOTO 10
30 PRINT A$ : POKE 16439,0
40 GOTO 10
```

Rod Bullard Southern TRS-80

MORE KEYBOARD INPUT

Regarding the letter in the current newsletter by Clive Keech regarding the keyboard scan, the following chart printed in William Barden's excellent book "PROGRAMMING TECHNIQUES for LEVEL 2 BASIC", may be of help.

ROW ADDRESS									BUFFER
14337	@	A	B	C	D	E	F	G	16438
14338	h	i	j	k	l	m	n	o	16439
14340	P	Q	R	S	T	U	V	W	16400
14344	X	Y	Z						16441
14352	0	1	2	3	4	5	6	7	16442
14368	8	9	:	;	,	-	.	/	16443
14400	ENT		BRK	<-ARROWS->			SPC		16444
	CLR								
14464	SHIFT								
COLUMN VALUE	1	2	4	8	16	32	64	128	

PEEKING at any of the ROW ADDRESSES will produc a value corresponding to the COLUMN VALUE.

The bUFFER addresses on the right also contain the COLUMN VALUES. This allows a useful KEY REPEAT function, as in the following demo program:-

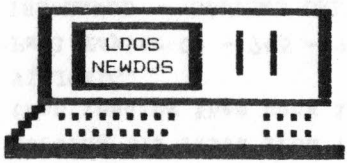
```
10 CLS
20 a$=""
30 I$=INKEY$:IF I$="" THEN 30
40 A$=A$+I$:PRNT@128,A$;
50 POKE 16441,0:GOTO 30

RUN
```

Hold either the X,Y or Z keys & a continuos line of characters will be printed. Another useful aid, regarding the INKEY$ function is POKE 16537,0 which zeroes INKEY$ & prevents previous entries being read.

PETE BAMFIELD
FLAT 5, 24 WINDLESHAM GARDENS
BRIGHTON
EAST SUSSEX

GRAPHIC EDITORS



For the TRS8Ø and GENIE

A review of Powerdraw and Zgraph

Creating good graphics for your programs can be tiresome without a graphics editor. Printing CHR$(and poking characters to screen takes time and it is often difficult to obtain the result you want. Your program may end up a mass of SETs, POKEs and PRINT @ statements. When it comes to assembly language program graphics the situation is even more frustrating. It can take many hours of re-editing to achieve the effect you want. The two programs in this review will allow you to create graphics for use in both Basic and Assembler. You may not end up creating a "Dancing Demon" but your programs will be all the better for a bit of graphic artistry.

POWERDRAW - by Kim Watt - from Powersoft.

Most of us are acquainted with the Kim Watt name from his best known work Super Utility. He has lived up to his reputation for programming ability with Powerdraw. The program is supplied on a 35 track single density disk and operates with all known major DOS. Transfering the program to your own operating system is easy.

Typing PDRAW from the dos ready prompt enters the program with a beautifully drawn logo, followed by a "help" page. The list of commands is extensive ---

Set angles	Access buffers	Clear buffer	Draw mode
Erase mode	Flash speed	Get a file	Home
Up/Left	Display grid	Down/Left	Down/Right
Menu	Flip sides	Up/Right	Put file
Quit	Reverse screen	Skip mode	Text entry
Cursor locate	Video size	Watch buffers	Mirror R=L
Mirror T=B	Move		

You can draw on the screen, directing the cursor with the arrow keys and for diagonal lines the 4 special keys are available. If you have the Alpha joystick, this can also be used to control the cursor. Text can be mixed with graphics and it is easy to toggle between the two modes. The angle of a line to be drawn can be set with a choice between solid or dotted. A grid can be toggled on and off the screen to assist in correct alignment of your drawing.

Once the "masterpiece" has been created it can be stored in a buffer for subsequent dump to disk or tape. There are seven formats to save the screen data.

(1) Compressed, for use by Pdraw
(2) Edas format, using the DB opcode with 8 data bytes per line
(3) Edtasm format, using DEFW with 2 data bytes per line
(4) Basic array, comprising a DIM and 16 line array
(5) Basic string, having 16 lines of print"
(6) Basic data, with 8 bytes per line
(7) Load module format exiting to 402D

Reverse screen, three types of screen print, merging and moving of buffers can make your graphics "talk" and even give you moving pictures. This program has to be used to be appreciated and your screen may never look the same again. My only criticism is the occasional key bounce when in the text enter mode, but it is one inconvenience worth putting up with for such a fine piece of software.

ZGRAPH - by Karl A Hessinger - from Misosys.
==========

Misosys support the LDOS operating system with a number of programs and Zgraph is a new addition to the collection. Like Powerdraw it will provide you with the opprtunity to use the graphics capabilies of the TRS80 to their maximum. Unlike Powerdraw, Zgraph is dedicated to one operating system, namely LDOS, and it also differs as regards facilities.

The program is in two parts, ZGRAPH/CMD and ZGRAPH/BAS, the first part loads the machine languge support and calls the Basic program automatically. There are two command modes, primary and secondary. Their commands are ---

PRIMARY

Draw	Home cursor	Function select	Erase
Help	Reverse screen	Text insert	Locate marker
X-flip	Move mode	Set marker	Y-flip

SECONDARY

Get file	Circle	Blank	Exit	Save file
Duplicate	Abort	Help	Merge	Line
Translate	Position	Exchange	Rectangle	Window
Query				

The cursor is controlled by means of the numeric keypad, or in the case of the Genie by the numbers on the top row of the keyboard. The outstanding commands are CIRCLE, which will draw a circle of any number of pixels radius, RECTANGLE which will draw a rectangle between the marker and current cursor position and LINE which will draw a straight line between the marker and current cursor position. It is these three commands which make this program a dream to use, saving use of the cursor control keys and it "gets it right first time".

The file saved by Zgraph can be converted by the utlity program BINCONV, to give four file formats.

(1) Load module with user defined start address
(2) Packed Basic strings in array format
(3) Basic data, 16 bytes per line, user defined line numbers
(4) Edas format, DEFB with 16 values per line

Reverse screen, a screen print utility and sample screens go to make up an excellent package. The main drawback of the program is that it only operates with LDOS but it has a ntages if used with the other Misosys package Grasp in providin it-image print on an MX80 of graphics and text.

I can thoroughly recommend both programs if you enjoy graphics and even if you don't, they make life much easier.
Ken Grey

FOR SALE

Model One Keyboard
Lineprinter VII
Printer Interface
£390 the lot
Mr Knight Dudley 235712

2/83

DRAGON USERS.
DAVE BUCKINGHAM
LUTON 570125

2/83

FOR SALE

Tandy Quick Printer
Paper & Manual Cost £199
Bargain At £50.

Phone Alan Toplis
0922-28798

FOR SALE

Aculab Floppy Tape
XBAS, SCRIPSIT, FORTH
DATABASE, TAPEACU, ETC.
£100.00

Phone Bob Moore
01-940-2916

FOR SALE

Computer Consumables
Discs & Sundries

CONSUP 01-670-9886
Unit 112,
Lambeth Ind. Estate,
62 Tritton Road,
London SE21 8DE.

PROBLEM

How do I hook up to Paper Tiger with my Video Genie?

Richard Stockwell,
c/o St. Peterport School,
Les Ozouets St. Peterport,
Guernsey.

PROBLEM

Has anybody put their golf club handicap system on their TRS-80 Model I system.
Hubert Butler 0748 - 3108

2/83

PROBLEM

Can a CREED terminal be hooked to Video Genie
S.Casarez 0394 - 3051

2/83

PROBLEM

Connecting the old Micropolis Drives to the BBC micro seems to be a problem.
Mr Gill of Kentish Town.

PROBLEM SOLVED !

Creating random numbers using machine code.
Robin Telkman 0532 - 795977

PROBLEM SOLVED

Tandy to Apple transfer - dead easy without packages.
Anthea Dore 0245 - 400230

2/83

PROBLEM

Sorting DIF files from VISICALC then reading them back into VISICALC.
Paul Taylor 01 - 743 - 1035

INSURANCE - SYSTEMS ONLY

Brian Pain can help on the above

MILTON KEYNES WORKSHOP

For Sunday morning there was a rich offering:-
Leo Knaggs & Laurie Shields on ussing an assembler for the first time, Leon Heller using MACRO-80, Dusty Miller & Clive Keech on ELECTRIC PENCIL & SCRIPSIT, with John Newgas on LAZYWRITER and the more recent word Processing Programs.

I would dearly like to offer recordings of the sessions, but they always seem to be the last thing on our priority list. If enough wrote in I am sure Laurie & Leon could be persuaded to just chat on their favourite topics. Leo Knaggs ought to be shackled to a system and given a disk to repair. Please write in & give them a nudge.

On the above topics please do not hesitate to ring in - Dusty & Clive can be very helpful on WP in the working office. John Newgas says he is available until very late, which I doubted until proved at midnight one Sunday evening. A caller from Germany chatted to John recently & promptly paid his sub !

The afternoon was preceded by a session on communications with Don Bannister from Belfast which ranged over hardware & software. For those who want some assistance, ring him & hang on - why not record the chat & send in a transcription? All weekend he was in the thick of some "heavy sessions", especially with Andrew Donald and John Taylor on BEEB to TRS-80 transfers.
Sunday Afternoon saw Chris Paradine doing a session on his DATABASE Program, much appreciateed by those who had specifically asked for this. Leon reviewed the hardware scene in usual masterly manner, thanked by those riveted to this display of expertise. Finally we had John Newgas on Compilers and Pete Drew on the EPSOM Printer. John has now put many members on the right compilers road - going over their uses, drawbacks & strengths - and Pete has assisted over the phone and at this session on the printer's features.

For the record the weekend was the best attended & showed the most enthusism. Let us try & maintain this sort of meeting as there is no comparison with the one days

Brian Pain

```
10 '   A MODEST LITTLE PROGRAM FOR A MODEST LITTLE PRINTER,
                THE  TRS-80  LINE-PRINTER  VII

20 '*************************************************************
     From Chris Driver, 7 Chambres Road, SOUTHPORT. PR8 6JG.
   *************************************************************
30 A$="AN ENHANCED TITLE.":B$="AN ORDINARY TITLE."
40 LPRINTCHR$(31)A$;    'LPRINTS TITLE IN LARGE LETTERS
50 LPRINTCHR$(26);CHR$(18);CHR$(128);CHR$(31);A$
60 '     CODE 26  =  PRINT WITHOUT LINEFEED;
         CODE 18  =  SET GRAPHICS MODE;
         CODE 128 =  MOVES PRINTHEAD BY 1 GRAPHICS SPACE;
         CODE 31  =  SET CHARACTER (LARGE LETTERS) MODE
70 LPRINT B$
80 ' A RUN FOLLOWS :-
```

AN ENHANCED TITLE.
AN ORDINARY TITLE.

CAMEL MICROS

33A COMMERCIAL ROAD, WEYMOUTH, DORSET

SOFTWARE

GRASP

General purpose graph plotting program with facilities for scale change, axis inversion, histograms or graphs, multiple graphs, and screen dump to MX80 printer with graphics, etc.
Price £6.50

FILEMASTER

For disc users, this program can map almost any disc file format (fixed record size, variable record size, part fixed part variable, or serial files) into any other. Data on two input files can be merged to the output file, and arithmetic relationships can be set up. Ideal for linking data across programs, or producing listing files or command files to drive other programs. Capable of producing automatically, for example, a file which can be listed on a printer, used in Scripsit, or used as a command file, direct from a Visical /DIF file. All redundant data is stripped, full control over format of listing is possible, and titles (or commands) etc. can be included horizontally or vertically. Also has a manual mode for examining file structure, and manually extracting data onto a different file.
Price £11.90

ACCOUNTS SYSTEM

An accounts system intended for retail shop accounts, but useful for many small business applications. Sales, expenses (analysed), equipement, stock purchase, bank balance reconciliation, VAT, profit & loss account, weekly summaries of gross/net profit, etc. Complete system, s.a.e. for details.
PRICE £35 (48K single disc minimum)

HARDWARE

OKI M80A PRINTERS	£255
SMITH/CORONA DAISYWHEEL (Sale price)	£460
(State whether RS232C or Centronics style required)	
EPSON MX80 FT/3 Special ff's for User group	£362
GP100A	£223
PRINCE HIGH-RES (24MHZ) MONITORS	£100
QUALITY BLANK-LABEL CASSETTES. 5 for	£2.50
(C15, using HM-1260/1290 tape)	

ALL PRICES INCLUDE V.A.T. AND UK CARRIAGE.

NATIONAL TRS-80 & GENIE USER GROUP Newsletter

VOLUME 4 ISSUE 9 MARCH 1983

CONTENTS

2. Information on the Group
3. Editorial
3. Guidelines for contributors
4. Members' letters
6. Storing date and time data
8. Word-4-Word patch to allow printing CHR$0, 11 and 12
9. PEEKing the keyboard
10. Cassette problem - a solution from Tandy
12. Problem section
12. C/80 compiler
13. An aid for the handicapped
14. Mixed drives...backing up the directory
15. Exatron memory expansion
17. Microfiles
19. Electric Pencil, LDOS, and Epson printers
21. RS-232 test program
25. In reply to the reply to C. Lovell
25. Orchestra-80
33. Saving GSF on Floppy Tape
35. Video Genie, Epson, and Tiny Pascal

NATGUG

INFORMATION ON THE GROUP

Membership of the Group is by subscription to the Newsletter, which is published monthly. Membership details are obtainable from the Group Secretary. Membership of the Group is open to anyone with an interest in the TRS-80 range of microcomputers, and compatible systems such as the Video Genie.

Details of the Group accounts and the constitution of the Group are obtainable from the Secretary.

Members requiring assistance with problems related to the TRS-80/Video Genie may call the Secretary. An attempt will be made to put them in touch with a member who can help with the problem.

Workshops are arranged from time to time in various parts of the country.

Sub-groups exist in many areas. A list is provided at the back of the Newsletter from time to time. Sub-groups also exist for medical and educational users.

The Group runs 80-NET, a computerised bulletin board service that is available to all users of microcomputers.

The Group maintains three software libraries (Models I, II and III) which are free to members. Library lists are obtainable from the Secretary. The medical and educational sub-groups maintain their own libraries.

A membership list is obtainable on disk from the Secretary.

Back numbers of the Newsletter are available.

Please send all contributions for the Newsletter to the Editor.

Chairman and Newsletter Editor:-

Leon Heller,
8 Morris Walk,
Newport Pagnell,
Bucks. MK16 8QD.
Tel: (0908) 613004

Secretary and Newsletter Publisher:-

Brian Pain,
24 Oxford Street,
Stony Stratford,
Milton Keynes.
Tel: (0908) 564271

Librarians:-

Model I: Leighton Davies, 105 Caerau Road, Caerau, Glamorgan. Tel: (0656) 738337.

Model II: Jim Hutton, 25 High Street, Stroud, Glos. Tel: (045 36) 4423.

Model III: Nick Clay, 6 Rectory Lane, Byfleet, Surrey. KT 14 7LL. Tel: (09323) 42348.

EDITORIAL

Thanks for all the excellent contributions. The Newsletter is back to its former size. Do keep it up.

Some interesting rumours in a recent issue of Electronics - a Model IV, and a new Colour Computer to be made in South Korea, might be in the pipeline.

Computing had an article on a possible deal between Tandy and GEC Computers concerning the Model 16 and Model 12 machines. GEC might sell them under their own name.

The Swindon workshop was highly successful. I must confess I had reservations about whether it would prove viable, but I am very pleased to have been proved wrong. Can someone write a report for the Newsletter, please.

Leon Heller

GUIDELINES FOR CONTRIBUTORS

1. If possible, send material in printed form.
2. Scripsit or Pencil disks/cassettes are acceptable.
3. Ensure your ribbon is in reasonable condition.
4. Printer output should be on A4 paper if possible.
5. If you send in handwritten material, write legibly.
6. Do not fold the sheets when posting them. Use a large envelope and keep them flat.

Leon Heller

MEMBERS' LETTERS

I am writing to say that for the first time since joining the Group I have received a Newsletter which I thoroughly enjoyed. I particularly found C. Phillip's article very amusing and he is most definitely on the same wave length as me.

I also found Steve Taylor's letter illuminating and have to say that without the constant support and help from the Group I would long ago have decided that "computers are not for me'. I have to thank Adrian Morant for his help and his constant nagging for me to use the computer and not to leave"this expensive piece of equipment solely for my son's enjoyment". Adrian and I joined the local group at about the same time, both beginners and now enthusiastic supporters. I have also to thank Jeff Smith for his help with Scripsit - after several phone calls I managed to link the printer properly to the computer. Without his help I would never have found out what to do. I must also thank Eric Frankland and Eric Ericson for their help with Visicalc.

Referring next to Steve Alderton's letter I also found that the newsletter devoted too much space to those fortunate people who have discs and printers. Thanks to my "flexible friend" I have discs and a printer now but still use casettes and will continue to do so. As you know my son Joseph is a keen member of the Group - could not the children of the members contribute more: perhaps meet occasionly. If the Dads wish to hog their machines on Sunday mornings I would be very happy to arrange for children to come to my home on another day.

I cannot find a copy of TRS-80 Word Processing with Scripsit (McGraw) anywhere. Can you perhaps tell me where I can find a copy?

Finally Joseph has applied to be an Operator in response to the advert put in by Balrog Adventures. I will let you know how he gets on.

Do you have any news on machine code over 4-6 Saturday mornings at Seacoal Lane. We benefited greatly from the Basic course.

Unfortunately this time I could not "show off" my skills by using Scripsit to write to you as Michael is using the computer. He has become most skilled at Visicalc and is doing projections at the moment.

Once again many thanks to you all.

Miriam Pollard

This letter is mainly to thank all those concerned with the organisation of the M K Workshop in January. It was most beneficial to me, mainly from the lectures and advice recieved.

I have a few points which may be worth considering:

a. Coffee: I think it would be a little better if the group purchased some dispensers for the "makings".

b. Machine Code: Has Laurie considered putting on a formal course, perhaps during the School Holidays. I know that some commercial organisations do this sort of thing for profit and that we can't expect Laurie to provide his services free but I would have thought if he is willing something could be arranged. I for one would be willing to pay a reasonable sum for a weeks course, but I am in no position to pay the type of fees expected by the commercial companies who I think aim at the professional programmer.

c. Accommodation: I must say that I was most impressed by the Youth Hostel, the service, to my mind, was first class it was better than some hotels that I have used.

All in all it was a thoroughly enjoyable weekend and I am looking forward to the next one.

If there are any members out there who would like to form a local group for Mid-Norfolk then please drop me a line or give me a bell.

Paul Golder
Tel: (036 283) 491

Encouraged by the response to my last letter, one caller said that "it struck a sympathetic chord", here is another on the theme of the poor ignorant end USER!

At your suggestion I obtained a copy of "Electric Accountant" as I was looking for a simple basic accountancy programme for my company. I duly obeyed the instructions to "RUN INITACC"; nothing happened. I tried again; again no luck. Greatly daring I tried "DIR" and found there was a programme listed called "INITACC9". Success at last. A few key strokes later and the manual told me to "enter the first three letters of the month"; no luck. This was hardly suprising as the screen told me to enter the "MONTH NUMBER".

At this point that I threw the Manual out of the window, followed the screen prompts and completed my accounts in record time.

It was as a result of this experience that, with a blinding flash of inspiration, I formulated my panacea:-

SAM

I hereby formally announce that I appoint myself President-for-life of the "Society for Abolishing Manuals".

This resolve was greatly strengthened after having received a copy of "DATA-WRITER". Molimerx's blurb told me that it had one of the best manuals they had come across. I don't like to think what the others are like. Parts of it are, I admit, very good but other sections, such as REPORT, are completely devoid of any screen prompts even though four separate sections have to be formatted. In addition it frequently gets upset with me and either hangs-up and refuses any instruction or sulks and returns to DOS without any explanation. Furthermore, attempts to EDIT with the ACCESS programme often produces the extremely rude reply, "Illegal function call in Line 29299". On another occassion it said to me "?Redo from Start" but any attempt on my part to be pleasant to it by politely saying "Y" or even "N" or anything else was met with silence. A Reboot was the only solution. The Manual was silent on all these responses.

As a result of these and other experiences during the last few months I have formulated the first few rules of the SAM society. Any programme wishing to obtain the coveted Golden Disc award must, at the very least, comply with the following:-

1. The Disc as supplied must self-load immediately from Drive 0 on pressing the orange button. No "CONVERT"ing, "TRANSFER"ing or "BUGGER"ing is allowed.

2. The Manual is only permitted to explain in general terms WHAT the programme does, not HOW TO DO IT.

3. All the information required should be made self-evident from the screen.

4. The programme may not sulk, hang-up, return to DOS or make rude comments without a full explanation of what one has done wrong.

5. Any text editing commands incorporated in the programme must obey the same general rules as the word-processor used by the customer, in my case SCRIPSIT. When inserting a letter I expect to use "CONTROL" "S" not finger-twisting "SHIFT" "DOWN ARROW" "F".

6. The key repetition reaction time should also match the customers word-processor. The one incorporated in DATA-WRITER is faster than SCRIPSIT and results from time to time in a string of unwanted letters.

Additional rules will be published from time to time.

All submissions should therefore be made for use on a TRSDOS MODEL III. I should add that it can be on any DOS so long as it runs on my machine and complies with Rule 1 above. I don't wish to know!

So all you "String-compressors", "Garbage-collectors" and other masters of the arcane, order your bytes better and sent me your Discs.

I should perhaps add that at this stage ELECTRIC ACCOUNTANT would be well in line for the first of SAM's golden discs if it were not for the fact that it does not self-load. It is polite, and forgiving and its prompts are tactful and always available. It has a few minor quirks but I am sure Mr. Miller can put them right.

Chris Philip

STORING DATE AND TIME DATA

A couple of things I have found out recently but probably most people knew years ago.

A String takes up a lot of memory and if you have a lot of strings, you quickly use up all your memory. I used strings to hold dates and times because they are convenient and simple ie 21/06/32 for a date and 01:59.12 for a time. If you only have a few then there is no problem and time can be sorted quite easily as string as long as you stick rigidly to the format. It is no use having one as 01:59.10 and another as 1:59.1 and even 01.59.10 causes problems because the ":" sorts differently to ".". Dates are not easy to sort because we want to sort by year first then month. This makes the format YY/mm/dd which is fine for computers but difficult for humans.

To save on memory it is better to see if you can change the method of storing the data. A date really consist of 6 figures and two seperators. If we remove the seperators, we come out with a six figure number which will not exceed 311299 this century. If we store this number as an Integer we would save a lot of memory because integers take less memory than strings. If we get it to a number not exceeding 32767 we can get it into two bytes. We know that when we express dates commercially we can often forget the "19" and reduce the year to "82". It is not likely to be 1882 nor 2082 and sometimes we can do the same with the "8". Thus a simple

"2" can be taken as 1982. If we know the date was recent then it cannot have been 1972 and certainly not 1992. T' assumption is not valid for all date applications but it suff. s for a lot of situation.

How do we use this? First define a variable as an integer using a DEFINT statement. Then write a short routine to strip the seperators from the date string.

```
        DEFINTD
        INPUT" Date (dd/mm/yy) ";D$
  IF LEN(D$)<>8THEN PRINT"ERROR"
        IF MID$(D$,7,1)<>"8"THEN PRINT"DATE OUT OF RANGE"
        DD$=MID$(D$,1,2)+MID$(D$,4,2)+MID$(D$,8,1)
        D=VAL(DD): PRINTD
```

Hopefully you will now have reduced your date to two bytes and can store it in an array

To reconstruct the date you can use a similar method and put back the missing "198". Of course if by mistake you had allowed a "197" date to creep in then you will not come back to the same date. Therefor you need an error trapping routine to make sure that all dates are in the 1980's.

This will not help you sort dates corectly as they are still in the dd/mm/yy format and the 01/12/82 will preceed 31/01/82 but it is possible to use the MID$ function to reverse the date for sorting.

```
D1=MID$(D$,7,2)+MID$(D7$,4,2)+MID$(D$,1,2)
```

Should give you the date in yymmdd order. A simple loop can read this into an array and the whole thing sorted. It is important to realise the difference in storing in ddmmy order. In 1984 the abbreviated date will read 40101 on the first of Jan and 40101 exceeds the 32,767 limit for a two byte integer.

I also found a lot of interesting things when looking at times. In this context I mean performance times for competive events. At first we looked at the worst case and soon found races that would take us over 32,767. for instance any time over 3min 27.67 sec. We can ignore the decimal places as long as trailing "00" are kept in. We can ignore the fact that 3min is not the same as 327 secs because we are not confined to the normal numeric system which specifies X10 for each place etc. We cannot have times that exceed our top limit but we can manipulate them to fit.

One way round the problem is to reduce the figure by a set amount. In many races there as a top and a bottom limit to the range of times recorded. No runner has run a mile in less than 3 minutes nor are they likely to in the immediate future. Therefor you could reduce the input time by 3 minutes and add it back on output. This method can be very uselful. Basically you only store the difference between two numbers which will be smaller than either of them.

The general principle is to work in the best format for a computer rather than use human methods. On the other hand do not expect humans to appreciate the raw computer data. Convert a name to a number by all means but convert it back before you expect 810230049 to answer you.

The b difference between computer storage of data and our own is in s conversion principle. A few lines of basic will reduce a lot of information into a little memory and a few more will put it back. Also a few lines of code takes up a lot less space than the unconverted data. If you have only 8ot 16· of memory then you have got to use it eficiently if you have 64k you still cannot afford to waste it.

Derek Trayler

Word-4-Word Patch To Allow Printing Chrs 0, 11 and 12

Here is a simple patch to Word-4-Word which will allow the sending of control codes 0, 11 and 12 to your line printer. Codes 11 and 12 are normally blocked by the TRS-80 ROM which converts them into a series of linefeed or formfeed instructions. Code 0 gets blocked by both the ROM and W4W, the latter changing it to a question mark (code 63). The patch here is for the early cassette based program and it works on my system which does not include disks.

The first part of the patch is a bypass of the test in ROM, and the idea comes from an article by Don Bannister in the September 1982 Newsletter. In the initialisation routine of Word-4-Word is a short section of seemingly redundant code which goes as follows:-

```
43803E00     LD   A,00H          ;zero in register A
438232003C   LD   (3C00H),A      ;load into first VDU location
43853A003C   LD   A,(3C00H)      ;load VDU contents back into A
4388FE00     CP   00H            ;check if A holds zero
438A200D     JR   NZ,4399H       ;jump to main program if not
```

It may not be obvious how the A register can hold anything except zero, reading a bite from a memory location immediately after loading zero into it. However, if you don't have a lower case modification the video RAM consists of only 7 bits and the missing bit will be read as a 1. The section of code is simply a test for a lower case modification; if you know whether your system is modified or not you can omit the test and 12 bites become available. The new code to put in here is:-

```
4380218843   LD   HL,4388H       ;Relocate printer driver
4383222640   LD   (4026H),HL     ;  address in DCB
43861804     JR   438CH          ;jump to main program
438879       LD   A,C            ;replace instruction in ROM
438AC3B405   JP   05B4H          ;back to ROM driver
```

The first two instructions change the print driver address from 058DH (in ROM) to 4388H. The instructions at 4388H load A with the value in C (replacing the first instruction in the original) and jump to 05B4H, in the ROM driver after the tests for 0, 11 and 12. The relative jump at 4386H puts you back into the Word-4-Word program at the appropriate place for a system with a lower case mod installed. If your system doesn't have lower case, change this line to:-

```
43861811     JR   4399H          ;jump to main program
```

The second part of the patch is to prevent the program stopping zeros going through. The relevant test for zero is at 4994H:-

```
4994FEC0    CP   00H          ;is value zero ?
49962002    JR   NZ,499AH     ;if not, continue
49983E3F    LD   A,3FH        ;if so, change to 3FH "?"
```

The patch here is simple; stop zero being changed to 3FH. This is done by changing the contents of 4998H and 4999H each to 00H, which represents NOP (No OPeration), or do nothing.

I made the changes and saved the modified version using the memory edit and punch to cassette features of Mon-4. Any other monitor which does not reside between 4350H and 527DH could be used i.e. not TBUG which sits between 4380H and 497FH.

Jim Bartholomew

PEEKING THE KEYBOARD.

I am writing in response to Clive Keech's question in the latest edition of the NATGUG Newsletter, for information about addresses for PEEKing the keyboard. Clive mentions address 14400 which is the most useful, as it PEEKs the arrow keys and space bar, but all of the keys are PEEKable, as the list below shows:

PEEK	Value =	1	2	4	8	16	32	64	128
A	14337	(@)	(A a)	(B b)	(C c)	(D d)	(E e)	(F f)	(G g)
D	14338	(H h)	(I i)	(J j)	(K k)	(L l)	(M m)	(N n)	(O o)
D	14340	(P p)	(Q q)	(R r)	(S s)	(T t)	(U u)	(V v)	(W w)
R	14344	(X x)	(Y y)	(Z z)					
E	14352	(0)	(1 !)	(2 ")	(3 £)	(4 $)	(5 %)	(6 &)	(7 ')
S	14368	(8 ()	(9))	(: *)	(; +)	(, <)	(- =)	(. >)	(/ ?)
S	14400	(ENTER)	(CLEAR)	(BREAK)	(UP.AR)	(DN.AR)	(LT.AR)	(RT.AR)	(SPACE)
=	14464	(LSHFT)	(RSHFT)	(BothS)					

The differing PEEK values for the shift keys (14464) only work with the Model III; on the Model I, 1 is returned when either shift key is depressed.

It should be reaslised that pressing a combination of keys returns their combined values at any particular address: therefore, pressing X, Y and Z in unison will return a value of 7 when address 14344 is PEEKed; very handy for a deliberate escape from a routine.

Also, (using 14400 as an instance) if you AND your argument, (IF PEEK 14400 AND 64), then both 64 AND 16 (or 64 AND 8) are acceptable arguments when the right arrow AND down arrow (OR up arrow) are depressed which enables, for instance, diagonal lines to be drawn or diagonal steering to be achieved. However using EQUALS, (IF PEEK 14400 = 64) as an argument, then only the depression of the right arrow key will be recognised.

Another interesting item that some members may not be aware of, is that the keyboard buffer can be clear[illegible] when using INKEY$, so that a constant key depression can be [illegible]ecognised. Try the following:

```
10 IF INKEY$ = < "X" THEN 10
20 PRESS = PRESS + 1 : PRINT PRESS;" ";
30 GOTO 10
```

You will see that PRESS is incremented each time an X,Y or Z key is pressed, and that you have to release the key for the next press. Now change line 30 to read:

```
30 POKE 16441,0 : GOTO 10
```

Holding down X,Y or Z now, will increment PRESS continually

POKING 16441,0 clears the X,Y and Z print buffer. The other print buffer addresses are:

```
16438 = @ to G          16442 = 0 to 7
16439 = H to O          16443 = 8 to /
16440 = P to W          16444 = ENTER to SPACE
16441 = X to Z
```

It may interest you to know that PEEKing almost any address in the 14300 to 15359 range will detect a key press; Type: 10 PRINT PEEK(nnnnn);:GOTO10. Running this with differing values for nnnnn (between the addresses already mentioned) and pressing different keys, or combinations of keys, will give interesting results. Try 14336 or 14337, then press the 'S', Down Arrow and 'A' keys simultaneously; another deliberate abort routine?

I hope this answers your question, and is of some use to you.

Michael J. S. Barley

CASSETTE PROBLEM - A SOLUTION FROM TANDY

The problem with your CLOAD and INPUT functions lies not with the equipment but with the change in ROM which apparently was not documented in the manual you received. These changes and their impact on the dual cassette function have been documented in several places by Tandy/Radio Shack as well as in other media publications. I have excerpted two such articles from our own TRS-80 Micro-Computer Newsletters and enclosed them for your information. As a matter of interest to your Rheindahlen Computer Club, all the TRS-80 Micro Computer Newsletters from first edition through December 1980 (20 issues) have been reprinted into one document and issued as Catalog Number 26-2115. The twelve issues printed in 1981 have been reprinted in one volume and issued as Catalog Number 26-2240. These are valuable reference documents and would be decided assets to you as an individual and the club collectively. There is probably more information on our computers, including dozens of tips and programs to enhance your computing and your enjoyment of the TRS-80 line of computers than any other document on the market today. These should be available locally through the U.S. Armed Forces Audio Photo Club at Mainz Kastel near Wiesbaden/Mainz or through the Tandy outlets throughout Germany. I believe there is a Computer Center right there in Monchengladbach as a matter of fact. Look into it, it is definitely worth your while.

As for the mismatch of your VDU to the EI, I don't have much in the way of good news. When Tandy stopped manufacturing the Model I here in Fort Worth, we licensed a Japanese firm to continue production for a period of time. Due to the change in vendors, our U.K. and Belgian offices procured Video Units locally to satisfy the 220V 50Hz power problem. (See "Frequently Asked Questions", page two of the attached) T Video Units were 'off the shelf' rather that custom designed for ndy Corporation, therefore the mismatch to the EI. For that I apologize, but have no solution save the suggestion to build a platform to house the EI and support the VDU.

In any case, I hope that these answers will solve some of your problems and get you on the road to more fun with computing. Thank you very much for writing and for your interest in Tandy products.

Sincerely

Thomas M. LaCourse
Manager, Computer Support

For Your Information...

The following information is provided to help eliminate "concern" that because a recent Level II machine does not behave exactly the same as an earlier one, that one of them is "malfunctioning."

We are currently shipping 26-1054, 26-1056 and 26-1120 (Model I Level IIs and Level II Kits) with a new ROM. There are a few differences between this ROM and what we have used in the past.

1. The power-up query is: MEM SIZE?
2. The BASIC sign-on message is: R/S L2 BASIC
3. The amount of free memory space available was reduced by two (2) bytes. With a 16K machine, no memory protected, ?MEM will give 15570.
4. You can CLOAD and CLOAD? only from tape drive #1. You may not specify a tape unit when using these commands. The Level II manual indicates that these operations can be done using either of the two data recorders when you have an Expansion Interface. You can still CSAVE#—n and save and input data from both recorders.
5. You can now output control characters. Press <SHIFT> then <down-arrow>. This combination is "control" while you have both keys pressed, press the alphabetic key (A-Z) to send the desired control code. For example:

   <SHIFT><down-arrow>Z

   is the sequence for control-Z.
6. Do NOT use the KBEIX keyboard driver with these keyboards. The driver is NOT needed and will not work properly.
7. The cassette load modification for the keyboard is not needed with the new ROM and will not be installed.
8. This ROM is not interchangeable with the old ROM. A complete NEW Level II kit (26-1120 $150.00) must be installed to use this ROM. This kit is only available installed by the service centers. (Installation now extra.)
9. Some older tapes, specifically:

   26-1146 RS-232 Comm. Pack
   26-1181 VOXBOX Tapes
   26-1504 Tape Payroll
   26-1707 Level II Algebra I
   26-1802 Quick Watson
   26-1803 Blackjack/Backgammon
   26-1901 MicroChess
   26-1906 Invasion Force
   26-1907 Checkers 80
   26-2002 Editor/Assembler

   may not load properly using the new ROM. The problem has to do with the timing of the material on the tapes. If you get a tape which does not load, or which loads with the old ROM, but not the new one, you can either have someone make a copy from an old-ROM TRS-80 (this will correct the timing problem), or request your local Radio Shack to replace it.
10. You can now perform multiple PRINT @s using a single PRINT statement. For example:

    PRINT@10,"A",@74,"B",@138,"C"

    will not work with the old ROM, but will work with the new one.
11. Under Disk BASIC, you can now specify a file name when using CLOAD. In the past, under Disk BASIC, you could not CLOAD using a filename.

MODEL I LEVEL II ROM

Another subject we have had many questions on is the old and new Mod I Level II ROM's. This was handled in an earlier newsletter (July 1980, page 2) but we shall go over it again. The new ROM's can be distinguished from the old by the power up messages.

OLD ROM:	NEW ROM:
MEMORY SIZE? RADIO SHACK LEVEL II BASIC	MEM SIZE? R/S LII BASIC

Another difference is that a PRINT MEM at power up on the old ROM 16K unit will return 15572 whereas the new ROM will return 15570. New ROM's also cannot CLOAD from 2 cassettes, but can output control codes, and already has the KBFIX program built in.

The only problem we've had with the new ROM has been that some machine language tapes will not load. This is caused by the timing of the material on the tape. If you get a tape which does not load, or which loads with the old ROM, but not the new one, you can either have someone make a copy from an old-ROM or request a replacement. If the tape is made by another Mod I Level II, old or new ROM it should work fine. So if you have one of our machine language tapes, and it does not load, then have your nearest Radio Shack store exchange the tape for a new ROM version. An alternate solution would be to copy the program from an old ROM Mod I using T-BUG or DEBUG on tape. If you have programs from other companies either contact the manufacturer or use the second method described above.

FREQUENTLY ASKED QUESTIONS:

Question 1: I have a Radio Shack computer and am moving to Europe. Will it work overseas?

Answer: In general, if the destination country has 60 hz power, you can use a voltage converter to obtain 110-117 volts. Since most countries outside of the United States are 50 hz this procedure will not work. For the Model I only, you can purchase new power supplies overseas for the Keyboard/CPU and the Expansion Interface so they will work. You must purchase a new video and desired peripherals overseas to obtain 50 hz operation. No 50 hz equipment of any kind is available for sale in the U.S.

Question 2: I would like to have a schematic of my computer, where can I get one?

Answer: You can purchase a Service Manual for any of our computer devices using the stock number of the device preceded with an MS and the 26 followed by a zero. For example, the Service Manual for the video on the Mod I, which has a stock number of 26-1201, would be MS260-1201. The Mod III is MS260-1061/62/63. (The Mod III has the trailing numbers because there are three different kinds of Mod III's.) These Manuals have complete schematics, diagrams, port descriptions with bit allocations, etc. Essentially, these Manuals will contain as much information on the device as you can get in documented form. Additional information is contained in the owner's manual which comes with the device.

PROBLEM SECTION

I was interested in J R Jones's suggestion and query regarding joining up a Sinclair Spectrum with the TRS80. I was similarly intrigued in the possibility of linking up with the BBC micro and wrote to the designers, but the answer seemed very negative. My son has managed to get hold of a second hand terminal keyboard Memorex 1377 with its beautiful screen with 80 characters and 24 lines. It is possible to key in any characters to the screen but it is dumb - I suppose it has only a rudimentary CPU, despite the fact that its internals are packed with many more chips of different kinds than the TRS80 has. These terminals designed for use with main frame computors are likely to come of the second hand market increasingly, and it would be marvellous to be able to use them in the same building as a terminal to the TRS80, or in some other useful way. But I guess that some hard thinking and design work would be needed. Any comments?

Bill Cowhig

C/80 COMPILER

Although I like programming in assembly language, a more efficient way to produce utilities and systems software is to use a systems language like BCPL or C. The MDBS data base management system that I am currently using on several projects is written in C, and Digital Research now write their CP/M implementations in C. Code produced by such languages is not as efficient as machine language produced by a skilled programmer, but takes a fraction of the time to develop, and is easily transported to other processors.

I recently purchased the C/80 compiler, having seen a favourable review of it in the latest PCW. It also did quite well in some benchmark tests published in Byte recently. It only cost £50!

GOTO p16

AN AID FOR THE HANDICAPPED

I have been prompted to write by your short article on programming for the handicapped, a subject that I have been interested in for some time now but a lack of interest on the part of the "helpers" of some of the handicapped has almost destroyed my motivation. However, shortly before the end of the last school term I became involved with a local special school who had just been given 3 Genies by their P.T.A. but didn't know what to do with them.

I had written a rather crude but effective communication program some time ago and I pulled this out and updated it a bit and we got it into use just before the end of the term with a spastic boy of about thirteen. He is unable to speak and has very poor control of hands and feet and previously had been using a Possum with a three position joystick. We have now got him using the Genie instead of the Possum and he is controlling it with a four position joystick. By the end of the first week he was able to produce text on the printer as fast or faster than using the Possum but it had become apparent that a joystick was not the best input device for him. This term we will fit him up with some push switches. Tandy sell some nice footswitches for under three quid which for them is exceptional value! These switches can be given a softer action by removeing some of the springs.

The program (also on this tape) at present can be used for communication, recreation or as a simple calculator as follows:-

CLOAD and RUN in the normal manner the enter the response speed when prompted. This alters the repeat rate and speed of response to inhibit inadvertent entries.

Using the four cursor control keys (we have paralled switches with thes) move the cursor left, right or down to the appropriate character. You will note that the character over the cursor will also be shown on the screen at the bottom of the control area. Selecting the "UP" key will select the character an display it in the text area after any previous characters. A space may be generated by selecting one of the spaces at the end of the line or in the middle.

Selecting a character on the second line will generate a whole word and one on the third line will generate a whole phrase. These can of course be added to the text.

Selecting a < character will erase the previous character and the one below that gives a line feed.

Selecting the X on the far left will erase the text area whilst the P below it will give a screen print of the text area then erase it.

Selecting the left hand $ symbols switches the top line between upper and lower case.

If the word DRAW (no spaces or other characters) is put in the text area and then the left hand E is selected, the program will escape to a simple drawing routine. The four corner characters will erase the line, draw a line, erase the screen or return to the main program. Simple, crude but for a child who can't hold a pencil wonderfull.

Escaping to MOUSETRAP takes you into a game from Computing Today with a way out back tho the main program.

If a simple calculation is built up in the text area and then the left hand C is selected the Calculator mode will give the result. Again, simple and crude but if your fingers won't work ---------

You will note that I have used four input switches rather than two as Possum usually use. This is because if a severely handicapped person can operate two switches they can usually manage four - these could of course be a mixture of foot, hand, head or mouth controls. By using four controls the potential speed of operation is exponentially faster than one, two or three.

This program is in a fairly early stage of development and was started before I knew how to use arrays but has been modified somewhat. As a result of this it is rough, unstructured but it works!

I hope that you find this of interest - include any of this letter in the newsletter if you wish. Please let me know how yo

JOHN TAYLOR

Mixed Drives .... Backing up the directory

I've finally got the TRS to fully support my rag-bag of disk drives under LDOS. At the moment I have 1-Tandon 41-track, 1-BASF 41-track (which is reluctant to work double density) and an old 8" Shugart SA800. All are single sided and I've bodged a V.G. DD controller so it will handle both drive sizes in both densities. (8" DD needs a 4MHz clock though ... next project.)

Newdos 80v2.0 won't cope with this mixture (I'm relieved to say!). Ldos is sufficiently well documented to allow patches to be made to the disk I/O primatives in the PDUBL driver relatively easily (an extra 12 bytes when you know where to look!). Of course its impossible to get this code in the right place until the mandatory 25 directories have been sacrificed ,so after only 3 lost disks I thought it would be a good idea to copy the directory and hide it in a lonely corner of the disk.(Before someone asks, it is of course impossible to take proper disk backups whilst pushing back the frontiers of knowledge at 3am in the morning!.)

Hence this program. Not very elegant but it does work. It copies the directory to another file on the same or another disk using the diskette pack-ID as its name (no embedded blanks please) and "/BAK" as an extension. It then does it all again so it includes its own entry. You can get round this by CREATE-ing a suitable file first, but I never remember. Besides all my drives need different length files:

5" SD	103KB	10 sector directory
5" DD	185KB	18 sector directory
8" SD	308KB	16 sector directory
8" DD	591KB	30 sector directory...some day

All sizes double for double sided operation.
All sectors are 256 bytes.

Heres the program...

```
100 CLS
110 GOTO150
120 INPUT "WHICH DRIVE 0-->2";X%
130 IF X%<0 OR X%>2 THEN PRINT"INVALID DRIVE !!":GOTO120
140 RETURN
150 PRINT"TARGET DISKETTE":GOSUB120
160 INFILE$="DIR/SYS.SYSTEM:"+RIGHT$(STR$(X%),1)
162 PRINT"DESTINATION DISKETTE":GOSUB120
170 OPEN"RO",1,INFILE$
180 FIELD 1,255 AS SI$,1 AS S1$
190     GET 1
200 BACFILE$=MID$(SI$,0,8)+"/BAK:"+RIGHT$(STR$(X%),1)
```

```
220 OPEN"R",2,BACFILE$
230 FIELD 2,255 AS SD$,1 AS S2$
232     LSET SD$=SI$:LSET S2$=S1$
234     PUT 2
236 N%=2
240 FOR I%=N% TO LOF(1)
250     GET 1,I%
260     LSET SD$=SI$:LSET S2$=S1$
270     PUT 2,I%
280 NEXT
290 CLOSE
300 IF N%=1 THEN END
310 N%=1
320 OPEN"RO",1,INFILE$
330 OPEN"RO",2,BACFILE$
340 GOTO240
```

NOTES

Line 190 gets the first sector of the directory and line 200 abstracts the diskette name and makes it into a file name.

On the fist time round line 234 writes the first sector to the new file with the loop in lines 240-280 copying the rest.

On the second time this loop starts one secto earlier and copies the whole file.

"RO" in the OPEN statement is a means of ensuring that the file already exists. You've got problems if it doesn't!

Al Redhouse (07593-582)

EXATRON MEMORY EXPANSION

We had some memory problems with our LNW board and decided to install the EXATRON expansion into our keyboard. This is a new way of expanding the memory of the Model 1 by inserting an additional board into the keyboard unit. This is not unique as many have reported using the Holmes Expansion board which does a simlar job. The difference is in the type of chips.

The EXATRON board is produced by the makers of the Stingy Floppy and uses the new 64K chips. It is available as a bare board or as made up unit. Ours was fully made up and only required insertion. Fitting involves taking out the Z80 and the memory chips and putting the new board in its place. This sounds like a difficult alignment job but it was done without problems. No tracks are cut but two wires have to be soldered in. This is reasonably straight forward but I will leave to my son to explain the details.

The new RAM has 64k which overlaps all the ROMS. It is in one continuous block but the sections covered by the ROMS are not used. The ROMS can be switched out under sofware control which has great possiblities (not much use but great possiblities). A word processor which can use the extra space is in the pipeline as are other languages, CPM etc.

It is possible and simple to load the contents of the ROMS into the RAM. This allows the contents to be altered. You can get a lower case drive and keyboard inversion so that the shift key works like a typewriter.

The new memory takes lees current than the old type so no extra power supply is used. If you add a Stringy Floppy it makes a good portable unit that will go into a small case. The additional memory makes a considerable difference to word processing. With 16k and Scripsit I could only get one and a half A4 sheets but with expansion I get sixteen pages.

Was it worth it? Is it a good buy?

I already had a LNW Expansion with Disc controllers, RS232 and a printer interface, therefor I have not gained much and I still need the expansion for the disc controllers and double density. The only real difference is that I can take the expansion off and still have 48K. I have the "state of the art chips" and less money.

If this had been available before I had the LNW board it would have been a very sensible way to go up, especially as I had Stringy Floppy. I also feel that with ENHBAS, I could have had a system that did most of the things I wanted to do including the SORT. It is strange but despite the facilities on NEWDOS and LDOS, I still find ENHBAS easier to use. After all it is easier to use GSUB "SORT" than GOSUB1000 because you may change the position of the GOSUE when developing the software GSUB in ENHBAS will find the subroutine where ever it ends up.

I have heard that a seperate add on disc controller will be produced. It was not produced at the same time because the ESF claim that you do not need discs if you have a Stringy Floppy. This is partially true but since the cost of discs has come down people are looking in that direction.

How does it compare with the Holmes board? Well for a start it cost us a lot more. It contains a speed up mod which at present we do not have working. It allows the ROMS to be switched out which might be useful when software is available to exploit the facility but otherwise you do not get much extra for your money. The bare board is quite cheap and well worth considering if you can get the chips. The problem we found on cost was not the basic price but the extras this end. Most things have come through airmail and the customs forms come at the same time. The whole thing can be done by post. This was well packed and came Air freight. I had to pay #50 and a further #40 customs and handling. It would have been a lot more If I had not collected from Heathrow. Most of the freight cost was the weight of the packing and the manual.

Regards

Derek Travier.

(* You should ALWAYS have items purchased from the States sent Air Mail, never by air freight. It is just about as quick and very much cheaper, and the goods are delivered by the postman. LFH *)

C/80 COMPILER (contd.)

C/80 comes on an 8" CP/M disk, and I am running it on a Model II under the Lifeboat CP/M. I have only had it a couple of days, but I am really impressed with it. It is a development of the Small C compiler published in DDJ a few years ago, and has most of the features of the standard C that is supplied with the Unix operating system. There is no floating point, but I don't need FP anyway for the sort of programming I intend to use it for. It compiles to 8080 assembly language code which is then put through an absolute 8080 assembler which comes with the package. One can also use it with Macro-80 to produce .REL files, which can then be linked and loaded in the usual way. The fact that it produces 8080 code does not really matter, as compilers like C/80 only use a couple of registers in conjunction with the stack. A Z80 version would produce exactly the same set of machine language instructions. The

Cont p2

I have used MICROFILES for certain filing jobs and find that it has advantages and disadvantages. It is an old peice of software but I find it useful. I know many people can write better things for themselves but to the novice with discs it is a lot better than some packages.

One of its advantages is that it is easy to add on fields, this makes it a good starting point. Once you know the information you will require it is easy to use other programs but when first starting out with a new project it is easy to find that a piece of information is not available. You may have started out with a Husband/Wife column but later wished you had a "not speaking to each other col". MICROFILES allows that. You may want to fill in certain fields but not all. It is time consuming to keep <ENTERING> an empty field. MICROFILES allows optional fields. You can also print or display some or all fields in any order. This is particularly useful in certain applications. MICROFILES will sort very rapidly and I have no complaints in that direction but it will only sort on one field at a time and will not retain a sorted order.

This whole question of sorts is very complex and I have found many programs which appear to be useful but are less applicable when you come to use them. For instance one program will sort by several fields in any order while another will do the same but only if the fields are laid down in the correct order in the first place. Some will only sort on one field but it can be any field and the file will stay in that order. Therefor you can carry out the first sort and save to disc. A second sort takes it one stage further. There are often ways round problems and I have found a few.

MICROFILES will only SORT on one field so it is necesary to set up small files and sort within the file. In other words you do the first sort yourself. Instead of having all the members in one file you can have seperate male and female files. This prevents you working across the files but sometimes this is not actually necessary. In competitions there can be different age groups and boys and girls catagories. However boys never compete against girls so they can be in seperate files. I can load a file quicker than some so called Data base progams can sort.

If you do use seperate files you can run into problems when getting a composite printout. MICROFILES is a little fiddly when it comes to printing out sections of a list. You cannot specify records 140-239. You have to FIND 239 then go back to the PREVIOUS record and print the REST. If 239 is not the last record you have to wait for the right moment and <shift> <clear>. If you get it right you get the exact print out but often you get an extra line. I have found a useful trick that works with this and some other programs.

MICROFILES comes on TRS-DOS and is configured for a 35 track single density. There is no way that it can be altered unless you can find a machine code enthusiast. You can put it onto LDOS and it works on a doble density 40 track double density disc. It still only uses 35 tracks and the rest of the space is empty. Once you have LDOS as an operating system the you can do some extra things.

LDOS has a command ROUTE. This allows you to direct certain actions away from their intended destination. You can send the printer information to the screen or to a MODEM. You can also send the printer information to a disc file. Thats exciting you say, I can take my data off the disc and put it back again instead of printing it on paper. Well thats what happens but the new file on disc is not the same as it was. The new file is in ASCII format and is therefor readable by a word processor. If you use SCRIPSIT you can manipulate blocks, add headers, footers and page numbering. You can use the Global changes to extend fields. Abreviations can be expanded and lower case used if necessary. Using my example of the print out from MICROFILES, I can chop off those unwanted lines and delete all the blank lines so that my final copy is clean without resorting to cut and paste plus liberal use of Snowpake. Whats more I can use the control codes for my printer and there is nothing more annoying than having got the printout configured to 130 cols and realise that you printer is in normal print mode. You have break out and start all over again and there is no way you can use expanded lettering without a wordprocessor.

NEWDOS 80 has ROUTE command but it does not include ROUTE to DISC. Anyone with LDOS that wants to try this out can find it explained fully in the manual but briefly you have to ROUTE *PR filespec. I use only one text file at a time and therefor stick to ROUTE *PR PRINT/PRF:0 This opens a file "PRINT/PRF" on drive 0 but that is mainly because I know MICROFILES will occupy less than half my double density disc. If the file is very long it is better to ROUTE to drive 1. A wise precaution is to use the DEVICE command which should confirm that *PR = PRINT/PRF:0. I then load MICROFILES and load the files that I want to use. From then on I proceed as with normal print out even to sending the TURN PRINTER ON command. On the command PRINT the disc drives activate and in a few seconds the computer returns to the normal state. However the file is still open and the printer is still ROUTED. I then type DONE which is microfiles for return to DOS and get back to LDOS. It is vital to RESET *PR as this closes the file. I tend to keep SCRIPSIT on drive 1 and load my text file by LOADing PRINT/PRF and checking its contents. I usually SAVE it again under a diferent file name because despite the fact that LDOS is supposed to APPEND subsequent files. I find that it is safer to start again and kill PRINT/PRF and let the ROUTE command start a new one. It is easy enough to chain all the files together in SCRIPSIT and to form one composite file.

This technique should work with INSTANT SEARCH and SORT but the return to DOS reboots so the file remains open and is lost. I did try to modify ISS to go back to DOS READY instead of rebooting but I got the address wrong. Any help would be appreciated.

Another advantage in generating a text file is that the file becomes accessible with BASIC. It is possible to load the file back under BASIC instead of SCRIPSIT. A few extra sorting routines and you can manipulate the file still further. The modifiled file can be reSAVED and then loaded into SCRIPSIT.

Why not do this all under a BASIC program from the start? Well that is the best way if you know how and have the time. Also provided you thought of it in the first place. At least by my method you start off with a ready made set of input and file saving routines. You can get started and find out the problems as you go along. Also by using a word processor you can see what is happening and rectify faults as you go. I see my method as being like buying a ready made suite and having it altered to fit. If the alterations are extensive it would have been better to have had it made to measure but you cannot see whether it suites you before you start.

Derek Trayler

ELECTRIC PENCIL, LDOS, AND EPSON PRINTERS

By David Hawgood

A note by Alan Redhouse in your March 1982 issue (p53) pointed me in the right direction for solving Electric Pencil's awkward habit of dropping characters. I bought LDOS, used the Fix supplied with LDOS to patch my Pencil, and now have all the advantages of a type-ahead buffer.

On Electric Pencil, I had become familiar with using <CLEAR> for control functions: e.g. <CLEAR> <D> deletes a character. LDOS also uses <CLEAR> for its own control, and part of its Pencil Fix moves Pencil's control to <SHIFT> with <DOWN ARROW>. I didn't like this change, and Molimerx could not suggest any solution.

LDOS has a nice Key Stroke Multiply facility, which lets you print a standard paragraph by depressing <CLEAR> and a letter key. The standard paragraph can be built from Hex characters. It occurred to me that I could build a standard paragraph equivalent to keystrokes <SHIFT> (DOWN ARROW> and <D> together, file it in KSM under "D", and - Hey Presto! - I am back where I started. <CLEAR> <D> by this elaborate rigmarole deletes a character in Electric Pencil under LDOS.

This doesn't work for H, X, Y, and Z - the appropriate codes do things like moving the cursor about. I was reduced to odd successions of commands: to make <CLEAR> <Y> delete a line I had to construct a sequence "Insert a line, move cursor up, move cursor down, delete from cursor to end of line". But at least with KSM I could do all this sequence by keying <CLEAR> <Y>. While I was about it, I made a few improvements to the Pencil user interface: I have made <CLEAR> <I> as well as the oiginal <CLEAR> <F> turn on insertion mode, and <CLEAR> <S> now inserts a single space.

This technique may be useful with other packages. It enables you to alter and enhance the control sequences of software without any assembler programming or patching.

Another problem with the version of Electric Pencil I use is that it has no provision for special printer control characters. For a long time, before I loaded the word procesor I used to run a little program which left my Epson MX80F/T printer in Enhanced mode, or Condensed mode. I then thought of a technique which I thought would be tedious to operate, but actually is easy.

It involves building an Electric Pencil text file (name EPSDUBL/PCL) containing Epson control characters with tags to show what they do. If I want to turn on double width, I append EPSDUBL to the text I am writing, and use a block move to put the start-doubling character into the text. This is particularly useful for a layout like a letterhead as the result complete with control characters can be stored.

A convenient way to create the Pencil file with control characters is with a BASIC program:

```
5 CLEAR 500' PROGRAM "DOUBLER" FOR EPSON IN ELECTRIC PENCIL
10 OPEN "O", 1, "EPSDUBL/PCL"
20 DE$ =  "CHARACTERS BETWEEN !! SIGNS START !"  + CHR$(27)
   + "E! ENHANCING. "
25 DB$ ="THIS ONE !" + CHR$(14) + "! STARTS DOUBLING"
30 PRINT£1, DE$
35 PRINT£1, DB$
40 PRINT£1, CHR$(0) 'End Of File for Pencil file
50 CLOSE
60 PRINT "PROGRAM HAS RUN" : END
```

This program (using a method described in "TRS-80 Disk and other Mysteries" by J.C. Pennington) creates a file which prints like this:

CHARACTERS BETWEEN !! SIGNS START !! ENHANCING.
THIS ONE !! STARTS DOUBLING

The escape character displays as upwards arrow on a Model 1, CHR$(14) displays as N. Don't be put off by this: just remember that it's you fooling the Tandy this time, not the other way around.

Again, this technique could be useful on other software packages. For example you can float in a half line space, or some graphics characters. Of course, it will upset any justification: you will have inserted non-printing characters and may have doubled or condensed as well. I had in any case stopped using right justification with the Epson printer, so this did not bother me.

I have incidentally moved the listing of the Basic program above into Electric Pencil, using another technique described by Pennington: the Basic program was saved as ASCII with program name DOUBLER/PCL. This gives it the correct extent for an Electric Pencil file. Then I zapped the last character of the Basic program to change it from 0D to 00. I murmured the magic incantation "Take that, stupid Tandy" and found the file transformed into one I could load as text in the middle of this article.

C/80 COMPILER (contd.)

compiler uses a small run-time system which is included with the code produced, and this would probably gain from being written in Z80 code, however.

In addition to some of the usual features such as I/O re-direction, £define, and source file £include, one can use in-line assembler code for time-critical portions of programs. I think that I could even use Z80 code here, if I use Macro-80 for assembly, as the compiler passes the assembler code through unchanged. I will just have to use the .Z80 and .8080 switches to tell M80 how the code that follows is to be treated.

Since I have the source listing of the original Small C (written in C, of course), I can compile it on the Model II using C/80, and produce my own C compiler, with my own features, such as a run-time system using Z80 code. I could even modify the I/O to suit the Model I, compile the code on the Model II and download the assembler code to the Model I for assembly using M80. I will then end up with my own C compiler for the Model I. I could then download the

GOTO P27

RS-232 TEST PROGRAM

after reading the review of my rs232 board plus the fact that I've been writing a RTTY program in basic has given me some experience ! I have put together a simple RS232 test program in basic suitable for either my board or tandy's.

Future orders will be dispatched with this program but for those original orders here it is.
One point not mentioned by the reviewer was that the board is fully compatible with ALL Tandy software.

Further to his trouble achieving an RS232 manual I quote from page 56 of the model III technical manual 'the TRS80 RS232C Interface hardware manual has a good discussion of the RS232 standard and specific programming examples (catalog number 26-1145) ',so I am sorry for misleading him but it seems it should be available.

A friend 'phoned last week (Jim Roden I think) and he made some positive noises about the Maplin modem , his worked first time with no setting up!! It's competitor the one from Ambit semms ok but don't think it will modify to be an answer type very easily (I don't mean auto answer).
STOP PRESS! a friend has just built the ambit one and it worked first time.

What with telephones being so dear I've turned to transmitting programs over the air . I have used the audio tones from csave with success (volume very critical),my standard modem again ok and using the standard RTTY boxes built from bits (ST5 is the type);all using st80 iii but you need disk, unless anybody has that program that feeds incoming data through the basic interpreter ,some US mag I am told.

The ST5 is basically an american design for RTTY , radio teletype.Two audio tones are often transmitted,very similar to modem tones,these tones represent 1 or 0 ,the code is not ASCII but a clever 5 bit code which is toggled between two character sets with a shift character .It is called Murray code, no doubt those using a Creed printer will know exactly what I mean. The ST5 using tuned inductor / capacitor circuits converts the tones into voltage pulses of either +12v or -12v, which are usually fed to two mangets via a couple of transistors , hence a particular character is printed for each 5 bit code .
Anyone who has access to such a terminal unit and posesses an rs232 could decode the signals with the listed program if the transmission is 50 baud its very simple feed the signal from the op amp(IC2 pin 10)straight into the RS232 input

Transmission is easy too just feed the output from the rs232 into the tones board (pin number 33).

This program powers up in receive mode to go to transmit <shift> T ,back to receive with <shift> R .In Tx mode several shifted characters send some standard phrases.
I don't claim that the program is well written or bug free but it works well enough for me!

Eric Hartley

```
0 'THIS BASIC PROGRAM HAS THREE PARTS ENTER RUN 10 -RUN 1010 -RUN2010 DEPENDING
ON WHAT YOU WANT TO CHECK  RUN 10 TESTS THE MAIN PARTS OF THE RS232 IN VARIOUS S
ETTINGS
10 REM THIS BASIC PROGRAM WILL ENABLE CHECKING OF AN RS232 BOARD
20 REM JOIN TOGETHER THE SERIAL INPUT AND OUTPUT LEADS OF THE RS232
30 REM A NUMBER ENTERED IN  N WILL BE ECHOED TO THE SCREEN PROVIDING IT IS < OR
= TO THE WORD LENGTH
50 INPUT"WORD LENGTH ? (0=5BITS,1=6,2=7,3=8)";W
60 INPUT"STOP BITS (0=1 BIT,1=2 BITS)";S
70 INPUT"PARITY ENABLE (0=YES,1=NO)";P
80 INPUT"PARITY (1=EVEN 0=ODD) ( REPLY EITHER IF PARITY IS DISABLED)";E
90 X=E*128+W*32+S*16+P*8+4
97 OUT232,0
100 OUT234,X
110 OUT233,0'SETS BAUD TO 50
120 INPUT"ENTER THE NUMBER TO SEND";X
130 A=INP(234)
140 IF(A AND 64)=0 THEN 130
150 OUT235,X 'SEND THE NUMBER
160 A=INP(234)
170 IF (A AND 128)=0 THEN 160
180 A=INP(235)
190 PRINTA
200 GOTO120
210 END
1000 ' TO CHECK RTS,DTR,OUTPUTS "RUN 1010"WILL SET THEN RESET  EITHER OUTPUT DEP
ENDANT ON LINE 1020 CHECK WITH A VOLTMETER
1010 OUT232,0
1020 OUT234,1'SETS RTS OUT234,2 SETS DTR EDIT TO ALTER
1030 GOSUB1080
1040 OUT234,0
1050 GOSUB1080
1060 GOTO1020
1080 FOR X=1TO500:NEXT
1090 RETURN
1100 END
2000 'TO CHECK THE CTS,DSR,CD,RI INPUTS SET THEM TO A KNOWN VALUE (CTS,DSR,CD,RI
 CORRESPOND TO THE 4 MOST SIGNIFICANT BITS OF INP(232) THEN "RUN 2010
2010 OUT232,0
2020 PRINTINP(232);'THE VALUE RETURNED HERE WILL TELL WHICH OF CTS ETC HAVE BEEN
 SET
2030 FOR X=1TO200:NEXT
2040 GOTO2010
2050 END
```

C/80 COMPILER (contd.)

compiler source code to the Model I and then do any further development work for the Model I version on the Model I. This will all take quite a lot of time. The code generation part of this new compiler can easily be modified for other processors, so a cross-compiler for the 68000, say, could easily be produced. Assembly of the 68000 code would be tricky, so I will need a cross-assembler for the 68000 written in C. These are available (at a price).

More on C and C/80 in the next issue.

Leon Heller

LOST/STOLEN AT SWINDON

Someone accidentally removed/purloined a box of disks belonging to David Washford at some time during the Swindon week-end. Please give Brian a ring if you know anything about it.

```
20 REM RTTY/ASCII TX RX PROGRAM 15/2/83 EH V1-19
40 DIM  C$(32) ,D(127),U$(32)
60 C$(1)="E":C$(16)="T":C$(6)="I":C$(28)="M":C$(3)="A":C$(12)="N":C$(5)="S":C$(2
4)="O":C$(10)="R":C$(21)="Y":C$(14)="C":C$(23)="Q"
80 C$(7)="U":C$(4)=" ":C$(9)="D":C$(15)="K":C$(13)="F":C$(18)="L":C$(26)="G":C$(
20)="H"
100 C$(22)="P":C$(25)="B":C$(19)="W":C$(11)="J":C$(30)="V":C$(29)="X":C$(17)="Z"
120 D(65)=3:D(66)=25:D(67)=14:D(68)=9:D(69)=1:D(70)=13:D(71)=26:D(72)=20:D(73)=6
:D(74)=11:D(75)=15:D(76)=18
140 D(77)=28:D(78)=12:D(79)=24:D(80)=22:D(81)=23:D(82)=10:D(83)=5:D(108)=31:D(10
2)=27
160 D(84)=16:D(85)=7:D(86)=30:D(87)=19:D(88)=29:D(89)=21:D(90)=17:D(32)=4
180 D(45)=3:D(63)=25:D(58)=14:D(51)=1:D(37)=13:D(36)=20:D(56)=6:D(40)=15:D(41)=
8:D(46)=28:D(44)=12:D(57)=24:D(48)=22:D(49)=23:D(52)=10:D(33)=5:D(53)=16:D(55)=
:D(61)=30:D(50)=19:D(47)=29:D(54)=21:D(52)=10:D(64)=26:D(13)=8:D(10)=2
200 U$(1)="3":U$(16)="5":U$(6)="8":U$(24)="9":U$(10)="4":U$(21)="6":U$(23)="1":U
$(7)="7":U$(4)=" ":U$(22)="0":U$(19)="2":U$(29)="1"
220 U$(28)=".":U$(3)="-":U$(12)=",":U$(5)="!":U$(14)=":":U$(15)="(":U$(13)="%":U
$(18)=")":U$(26)="@":U$(20)=CHR$(96):U$(30)="=":U$(17)="+"
240 CLS
260 OUT 232,0
400 OUT 234,156
420 OUT 233,0
540 PRINT:PRINT:PRINT"*********  NOW IN RX MODE --- <SHIFT> T TO GO TO TX ******
** "
560 POKE15423,108:A=INP(234):IFPEEK(14356)=16ANDPEEK(14464)=1THEN760
570 IFPEEK(14400)=2:CLS
580 IFPEEK(14400)=8THEN660
600 IF(A AND 128)=0 THEN 560
620 R=INP(235):IFR=27THEN660
640 PRINTC$(R); :GOTO 560
660 POKE15423,102:A=INP(234):IFPEEK(14356)=16ANDPEEK(14464)=1THEN760
680 IFPEEK(14400)=8THEN560
700 IF(AAND128)=0THEN660
720 U=INP(235):IFU=31THEN560
740 PRINTU$(U);:GOTO660
760 PRINT:PRINT:PRINT"  ********* NOW IN TX MODE ----<SHIFT> R TO GO TO RX *****
**":L=2
770 Z=0
780 T$=INKEY$: IF T$="" THEN 780
800 IF T$= "r" THEN 540
820 IFT$="1"THEN1000
840 IFT$="c"THEN1260
860 IFT$="q"THEN1300
880 IFT$="y"THEN1280
900 IFT$="a"THEN1320
920 IFT$="f"THEN1140
930 IFT$="i"THEN1330
940 IFASC(T$)>32ANDASC(T$)<65THEN1120
960 IFL=0THEN1060
980 IFL=1ANDASC(T$)<33THEN1060
```

```
1000 C=INP(234):IF(CAND64)=0THEN1000
1020 OUT235,31
1040 L=0:POKE15423,108
1060 C=INP(234):IF(CAND64)=0THEN1060
1080 OUT235,D(ASC(T$)):Z=Z+1:IFZ>59ANDT$=" "GOSUB1480
1090 IFZ>69THENGOSUB1480
1100 PRINTT$;:GOTO780
1120 IFL=1THEN1200
1140 C=INP(234):IF(CAND64)=0THEN1140
1160 OUT235,27
1180 L=1:POKE15423,102
1200 C=INP(234):IF(CAND64)=0THEN1200
1220 OUT235,D(ASC(T$)):Z=Z+1:IFZ>59ANDT$=" "GOSUB1480
1230 IFZ>69GOSUB1480
1240 PRINTT$;:GOTO780
1260 SS$="1CQ  CQ  CQ  CQ  CQ   DE Gf61XXX    CQ  CQ  CQ  CQ  CQ  CQ  CQ  CQ  I
 Gf61XXX  ":GOTO1340
1280 SS$="1RYRYRYRYRYRYRYRYRYRYRYRYRYRYRYRYRYRYRYRYRYRYRYRYRYRYRYRYRYRYRYRYRY
YRY  ":GOTO1340
1300 SS$="1THE QUICK BROWN FOX JUMPS OVER THE LAZY DOG f01234567891 DE Gf61XXX
":GOTO1340
1320 SS$="1  DE  Gf61XXX  BLACKBURN LANCASHIRE  ":GOTO1340
1330 SS$="1THE NAME AT THIS END IS ERIC I AM USING A STf51 FED INTO A TRS f801
IA AN RSf2321C INTERFACE  THE RIG IS A YAESU f4801R":GOTO1340
1340 Z=1
1360 M=1
1380 T$=MID$(SS$,M,1)
1400 PRINTT$;
1420 C=INP(234):IF(CAND64)=0THEN1420
1440 OUT235,D(ASC(T$)):Z=Z+1:IFZ>60GOSUB1480
1460 M=M+1:IFM=LEN(SS$)THEN770ELSE1380
1480 C=INP(234):IF(CAND64)=0THEN1480
1500 OUT235,8
1520 C=INP(234):IF(CAND64)=0THEN1520
1540 OUT235,2
1560 Z=0
1580 RETURN
1600 'THE f AND 1 IN THE Tx STRINGS CALLED SS$ ---
 f SENDS FIGURE SHIFT
 1 SENDS LETTER SHIFT
```

IN REPLY TO THE REPLY TO C.LOVELL. (P22.Jan83)

Clive Keech has answered my query as to the meaning of "if Z and 8 Y=Y-1" etc. I had thought it might be a matter of Logical Operands, but in the light of knowledge gained whilst awaiting an answer I know that he is correct, but why isn't it written " If Z=8 then Y=Y-1" for that is what it means?

Clive himself asks a question that I can now answer thanks to a program given to me by John Gale of Hemel Hempstead. This little prgram shows how and which memory position is affected when various keys are pressed. It is easy to draw a chart for reference, and simple rules can soon be discovered which govern what happens if several keys are pressed at once.

Run, and then press various keys and observe what happens. Line 200 is a delay so that one can have time to absorb what happens.

```
100 A$="fffffff": CLS
110 PRINT @0,"MEM LOC";
120 PRINT USING A$; 14337; 14338;14340;14344;14352;14368;
    14400; 14464
124 PRINT @192,"PEEK(16537)   K$=INKEY$       ASC(K$)"
130 PRINT @64, "PEEK";
140 PRINT USING A$; PEEK(14337);PEEK(14338);PEEK(14340);
    PEEK(14344);PEEK(14352);PEEK(14368);PEEK(14400);
    PEEK(14464)
160 P=PEEK(16537):K$=INKEY$
170 IF K$="" THEN A=0 ELSE A=ASC(K$)
180 IF A>32 THEN PRINT @256,P,K$,A
    ELSE PRINT @256,P," ",A
200 IF A > 0 THEN FOR I=1 TO 500:NEXT
999 GOTO 130
```

How about an article by someone on the pros and cons of this method of Inkey$ versus the usual method? How does one obtain what might be called a "Permanent Inkey$" so that if, say, one presses a shift+up-arrow one exits to a sub-routine at any time in a program.....obviously it is not done with an Inkey$ on every line, but how?

Chris Lovell

ORCHESTRA-80

ORCHESTRA-80 IS A DIGITAL MUSIC SYNTHESIZER USING HARDWARE IN THE FORM OF A PRINTED CIRCUIT BOARD AND SOFTWARE BEING THE COMPILER FOR THE MUSIC CODE.

THE HARDWARE IS A PRINTED CIRCUIT BOARD BASED ON A MUSIC CHIP WHICH APPEARS TO BE ONE OF THE TI SYNTHESIS CHIPS.THIS BOARD PLUGS ONTO THE BUS EXPANDER ON THE BACK OF THE KEYBOARD OR ONTO THE SCREEN PRINTER CONNECTOR ON THE EXPANSION INTERFACE.THE BOARD HAS A PHONO SOCKET ON IT INTO WHICH YOU PLUG A RCA-TYPE PHONO PLUG WHICH IS PLUGGED INTO THE AMPLIFICATION YOU ARE USING.THE BASIC FUNCTION OF THIS PART OF ORCHESTRA-80 IS AS A PRECISION DIGI) TO ANALOGUE CONVERTER WHICH RECEIVES CODE FROM THE COMPILER AND CONVERTS IT INTO INFORMATION TO BE RECEIVED BY THE AMPLIFIER.

THE SOFTWARE CONSISTS OF A DIGITAL SYNTHESIZER WHICH IS CAPABLE OF A SIX OCTAVE RANGE WITH UPTO FOUR SIMULTANEOUS NOTES OR VOICES.EACH OF THESE VOICES CAN BE ASSIGNED TO ANY OF FOUR USER-DEFIGNED TONE COLOUR REGISTERS.THE COMPILER HAS FOUR PRESET TONE REGISTERS AND THESE ARE TRUMPET,OBOE,CLARINET & PIPE ORGAN.THESE REGISTERS MAY BE CHANGED TO IMITATE OTHER INSTRUMENTS OR TO CREATE NEW EFFECTS.THE TONE REGISTERS MAY BE CHANGED IN THE INITIALISATION OF THE COMPILER AND THEY MAY BE SAVED TO DISK OR CASSETTE TO GIVE A PERSONAL VERSION OF ORCHESTRA-80.

THE MUSIC LANGUAGE COMPILER ALLOWS DIRECT TRANSCRIPTION OF VIRTUALLY ANY WRITTEN MUSIC INTO A SYMBOLIC FORM USED BY THE COMPUTER.THIS METHOD MAKES IT RATHER EASY FOR NON-MUSICIANS TO USE AS WELL AS GIVING ALL THE FACILITIES ADVANCED MUSICIANS NEED.

THE COMPILER SUPPORTS ANY KEY OR TIME SIGNATURE;ANY NOTE FROM A BREVE TO DEMI-SEMIQUAVERS (WHOLE NOTES TO 1/64 NOTES) AND THESE MAY BE DOUBLE OR EVEN TRIPLE DOTTED;THE USE OF ACCIDENTALS IN MUSIC (WHERE A NOTE IS SHARP OR FLAT AND THIS DOES NOT COME INTO THE KEY OF THE PARTICULAR PIECE OF MUSIC).THE COMPILER ALSO SUPPORTS TWO FORMS OF ARTICULATION;THE USE OF TRIPLETS AND THE ABILITY TO USE MODULATION (THE CHANGING OF A PIECE OF MUSIC FROM ONE KEY TO ANOTHER) IN TRANSCRIPTIONS.

THE COMPILER IS MADE UP OF TWO PARTS BEING THE EDITOR AND THE COMPILER ITSELF.THE EDITOR IS A FULL FUNCTION/FULL SCREEN TEXT EDITOR AND ALLOWS UPTO FOURTEEN LINES OF MUSICAL CODE TO BE DISPLAYED ON THE SCREEN AT ONE TIME.THE TOP TWO LINES OF THE SCREEN ARE USED FOR THE EDITOR COMMANDS.THE EDITOR MODE IS INDICATED BY A FLASHING BLOCK CURSOR WHICH MAY BE MANOEUVRED THROUGH THE TEXT USING THE ARROW KEYS.FUNCTIONS AVAILABLE WHEN YOU ARE IN THE EDITOR MODE ARE INSERTION/DELETION OF CHARACTERS/LINES;GLOBAL CHARACTER SEARCH AND OTHERS I WILL MENTION LATER.

ONE OF THE MAIN FEATURES OF ORCHESTRA-80 WHICH MAKES IT SUCH A FINE PROGRAM IS THE FILE MANAGER WHICH IS INCORPORATED INTO THE COMMAND MODE OF THE EDITOR.THIS FILE MANAGER PROVIDES AN ORDERLY STORING AND RETRIEVAL OF NAMED PROGRAM FILES ON TAPE AND DISK.FEATURES OF FILE MANAGER INCLUDE THE AUTOMATIC LOADING,COMPILATION AND PLAYING OF FILES FROM ONE COMMAND AND THIS CAN BE EXECUTED FOR A STRING OF FILES ON THE SAME TAPE/DISK.THIS FILE MANAGER IS BACKED UP BY A UTILITY PROGRAM SUPPLIED WITH ORCHESTRA-80 CALLED ORCHCOPY/CMD WHICH CAN TRANSFER MUSIC FILES FROM TAPE TO DISK AND VICE-VERSA.

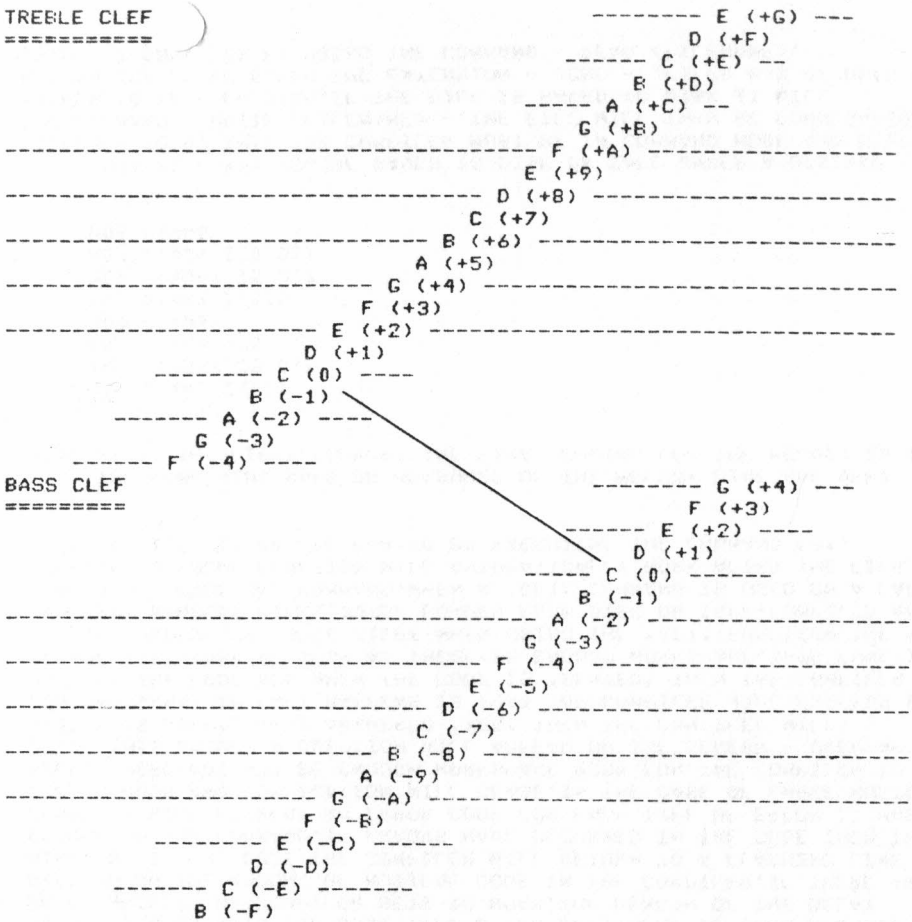
ON POWER UP YOU ARE ASKED FIRST OF ALL IF YOU REQUIRE THE STANDARD FOUR - VOICE SYNTHESIZER OR THE HIGHER RESOLUTION THREE - VOICE SYNTHESIZER.YOU MAY ALSO CHOOSE BETWEEN A STANDARD CLOCK (1.77 MHZ) OR AN ENHANCED CPU CLOCK RATE (2.6 MHZ) IF YOU HAVE THIS MODIFICATION FITTED.THIS INITIALISATION IS ALSO THE POINT WHERE YOU MAY MODIFY THE TONE REGISTERS IF YOU SO WISH.

YOU ARE NOW READY TO BEGIN ENTERING YOUR MUSICAL CODE.THE SCREEN WILL CLEAR TO A BLINKING LINE CURSOR ON THE FIRST LINE OF THE SCREEN.THIS IS THE COMMAND LINE.TO ENTER THE EDITOR MODE TYPE <E>.YOU ARE NOW READY FOR CODING THE MUSIC YOU HAVE CHOSEN TO TRANSCRIBE.IN THE EDITOR MODE YOU HAVE SEVERAL CONTROL KEYS.THESE ARE AS FOLLOWS :-

<BREAK>	STOPS CURRENT PROCESS AND E RS COMMAND MODE
<CLEAR>	ERASES CHARACTER AT THE CURSOR AND EVERYTHING TO THE RIGHT OF THE CURSOR.
<--	CURSOR LEFT
-->	CURSOR RIGHT
UPARROW	CURSOR UP
DOWNARROW	CURSOR DOWN
SHIFT <--	DELETES CHARACTER UNDER CURSOR AND SHIFTS THE REMAINDER OF THE LINE TO THE LEFT.
SHIFT -->	A SPACE IS INSERTED UNDER THE CURSOR AND THE TEXT IS SHIFTED TO THE RIGHT.
SHIFT UPARROW	THIS JOINS THE CURRENT LINE TO THE LINE ABOVE MAKING ONE LONG LINE.THIS IS ONLY EFFECTIVE AT THE BEGINNING OF A LINE.
SHIFT DOWNARROW	FORMS A NEW LINE BEGINNING AT THE CURSOR POSITION.
QUESTION MARK	MAKES A COPY OF THE CURRENT LINE AND INSERTS IT AT THE END OF THE FILE.
EXCLAMATION MARK	MOVES THE CURRENT LINE UP ONE POSITION TOWARDS THE TOP OR BEGINNING OF THE FILE.

THE COMMAND MODE OF THE COMPILER IS THE PART WHICH DOES ALL THE CONTROL WORK AND MANIPULATION OF THE CODE AND FILES.INSTRUCTIONS IN THE COMMAND MODE ARE NORMALLY A KEYWORD FOLLOWED BY ONE OR TWO OPERANDS.THE ACTUAL KEYWORDS MAY BE ABBREVIATED (I.E. <E> INSTEAD OF <EDIT>).THE KEYWORDS IN THE COMMAND MODE ARE AS FOLLOWS:-

KEYWORD	OPERAND	INSTRUCTION
NEW	NONE	DELETE CURRENT FILE IN MEMORY,RESET ALL POINTERS AND BEGIN NEW FILE.
/	<STRING>	SEARCH FOR NEXT OCCURENCE OF STRING.
BOTTOM	NONE	POSITION FILE POINTER AT END OF FILE.
DIR	0,1,2,3	LIST ALL FILES WITH EXTENSION /ORC ON DISK NAMED IN OPERAND.
EDIT	NONE	GO INTO EDIT MODE.
GET	FILENAME	READ,SCORE & PLAY EACH OF THE FILES NAMED IN THE OPERAND.
LIST	NONE	PRINT THE CURRENT FILE ON THE PRINTER.
PLAY	PART NO OR BLANK	PLAY THE CURRENT FILE STARTING WITH THE PART NUMBERED SPECIFIED OR PLAY THE WHOLE FILE.
QUIT	NONE	RETURN TO DOS OR LEVEL II.
READ	FILENAME	READ THE NAMED FILE INTO MEMORY.
SCORE	NONE	COMPILE THE CURRENT FILE.
TOP	NONE	GO TO TOP OF FILE.
WRITE	FILENAME	WRITE THE CURRENT FILE WITH THE NAME IN THE OPERAND.



ORCHESTRA-80 USES A CODING METHOD FOR MUSIC BASED UPON ONE PARTICULAR NOTE BEING A STARTING NOTE.THIS NOTE IS IN FACT THE NOTE 'C' BETWEEN THE TREBLE AND BASS CLEFS.EVERY NOTE ABOVE THIS NOTE IS DEEMED TO BE A 'PLUS' NOTE WHILST EVERY NOTE BELOW THIS IS A 'MINUS' NOTE.THE ACCOMPANYING TABLE SHOWING THE ACTUAL NOTE AND ITS' CODE NUMBER.

EVERY NOTE IN MUSIC HAS A VALUE ATTACHED TO ITS' LENGTH IN RELATION TO THE NUMBER OF BEATS OR COUNTS PER BAR.ORCHESTRA-80 SUPPORTS NOTES FROM A WHOLE NOTE DOWN TO A SIXTYFOURTH NOTE.EACH VALUE OF NOTE HAS A CODE AS WELL WHICH ARE SET OUT BELOW:-

NOTE	LENGTH	CODE
S/BREVE	WHOLE	W
MINIM	HALF	H
CROCHET	QUARTER	Q
QUAVER	EIGHTH	I
SEMI-QUAVER	SIXTEENTH	S
DEMI-SEMI-QUAVER	THIRTYSECOND	T
DEMI-DEMI-SEMI-QUAVER	SIXTYFOURTH	X

NOW FOR AN EXAMPLE SHOWING A PIECE OF MUSIC CODED.THE PIECE IS 'SWANEE RIVER' BY STEPHEN FOSTER WHICH HAS BEEN ARRANGED BY A COMPUTER PROGRAM GIVEN THE BASIC RULES OF HARMONY.WE WILL DEAL FIRST OF ALL WITH THE MELODY LINE I.E. THE TOP LINE OF NOTES ON THE MUSIC.

IN THE MELODY LINE THE FIRST SECTION UP TO THE LINE,KNOWN AS A BAR OR MEASURE READS AS FOLLOWS :-

MINIM E ; QUAVERS D E C D

THE TRANSCRIPTION OF THIS IS :-

```
M01 V1* H2 I1021
```

WHERE :- M01 STANDS FOR MEASURE OR BAR 1
V1 STANDS FOR VOICE 1
* TELLS THE COMPILER THAT THE STRING OF CODES THAT FOLLOW ARE ALL 'PLUS'.IF ALL THE NOTES WERE 'MINUS' THEN AN 'AT' SIGN WOULD PRECEDE THE STRING.
H2 TELLS THE COMPILER THAT THE FIRST NOTE IS A HALF NOTE OR MINIM AND THAT IT IS NOTE 'E'.
I1021 TELLS THE COMPILER THAT THERE ARE FOUR QUAVERS OR EIGHTH NOTES OF NOTES 'D,C,E & D'.

ALL MEASURES HAVE A TOTAL VALUE OF BEATS OR COUNTS AND THIS VALUE MUST BE MADE UP OR ELSE THE COMPILER WILL COMPILE THE CODE INCORRECTLY.THIS PARTICULAR PIECE OF MUSIC HAS 4 COUNTS OF QUARTER NOTES TO THE BAR AND THIS BAR ADDS UP CORRECTLY AS H= 1/2 AND Q=1/8 THERFORE 1/2 + 4*1/8 = 1.IF THERE ARE NOT ENOUGH NOTE VALUES TO COMPLETE THE VALUE OF THE BAR THEN RESTS MUST BE ADDED TO THE BAR.IN ORCHESTRA-80 CODE THE SYMBOL FOR A REST IS '$'.A REST IS WRITTEN IN THE SAME MANNER AS ANY OTHER NOTE SO IF A REST FOR A QUARTER NOTE IS NEEDED IT WOULD BE WRITTEN:- Q$.

NOW ON TO THE SECOND BAR OF THE MELODY WHICH READS:-

CROCHETS C C ; QUAVER A ; DOTTED CROCHET C

THE CODE FOR THIS IS:-

```
M02 V1* Q07 I5 0.7
```

THIS CODE IS STRAIGHT FORWARD ENOUGH EXCEPT FOR THE LAST NOTE WHICH IS ANOTHER FEATURE OF ORCHESTRA-80.THE LAST NOTE OF THE BAR IS A DOTTED CROCHET.A DOT WHEN PLACED AFTER A NOTE INCREASES THE NOTE BY HALF ITS' ORIGINAL VALUE AGAIN.IN ORCHESTRA-80 T[illegible] SAME OCCURS BUT THE DOT MUST ALWAYS BE PLACED AFTER THE VALUE S[illegible] (W,H,Q,I,S,T,X) OF THE NOTE.THE SCORING FOR

THE MELODY LINE FOR THE FIRST TWO BARS NOW LOOKS LIKE THIS:-

```
M01 V1*H2 I1021
M02 V1*Q07 I5 Q.7
```

AND TO COMPLETE THE FIRST PHRASE WE CODE THE TWO FOLLOWING BARS AS FOLLOWS :-

```
BAR 3 : - MINIM G ; CROCHETS E C   = M03 V1*H4 Q20
BAR 4 : - SEMIBREVE D              = M04 V1*W1
```

WE CAN NOW LISTEN TO OUR MELODY LINE IF WE SO WISH.TO DO THIS WE MUST GET BACK INTO THE COMMAND MODE BY HITTING THE <BREAK> KEY.THE COMMAND FOR PREPARING THE CODE FOR OUTPUT IN ORCHESTRA-80 LANGUAGE IS 'SCORE'.TO DO THIS YOU MUST TYPE S <ENTER>.THERE IS THEN A SMALL DELAY WHILST THE COMPILER GETS TO WORK.THE LENGTH OF THE DELAY DEPENDS ON THE AMOUNT OF MUSICAL CODE IN THE COMPILER.IF THERE ARE NO MISTAKES IN THE CODE THE COMPILER WILL RETURN TO A FLASHING LINE CURSOR ON THE COMMAND.IF ERRORS HAVE OCCURRED IN THE CODE THEN THE COMPILER WILL DISPLAY AN ERROR CODE FOR EACH TYPE OF ERROR IT HAS COME ACROSS AND COMPILATION WILL CEASE.IN THE CASE OF LARGE MUSIC FILES THERE MAY NOT BE ENOUGH WORKSPACE ROOM FOR THE COMPILER TO WORK IN.IF THIS OCCURS A QUESTION WILL APPEAR ON THE SCREEN - OVERLAP? -.IF THIS QUESTION IS ANSWERED 'YES' THEN THE COMPILER WILL CONTINUE.WHAT IN FACT HAPPENS IS THAT THE COMPILER WILL PROCESS SO MUCH OF THE CODE AND WHEN THE CODE IS 'PLAYED' THEN THE COMPILER WILL RESUME ITS' TASK AS SOON AS THERE IS ENOUGH WORKSPACE.THIS FUNCTION IS VERY USEFUL ON LARGE FILES WHEN USING THE 'GET' FUNCTION.THE 'GET' FUNCTION ENABLES FILES TO BE LOADED FROM DISK OR TAPE,COMPILED AND PLAYED ALL FROM ONE COMMAND.WHEN A 'GET' COMMAND IS USED ON A LARGE FILE THE OVERLAP FUNCTION WILL AUTOMATICALLY WORK.AFTER THE FILE HAS BEEN COMPILED IT MAY BE PLAYED BY EXECUTING THE COMMAND 'P'.

THE REMAINING BARS OR MEASURES OF THE MELODY LINE ARE VERY SIMILAR TO THE FIRST FOUR SO THE FINAL CODING FOR THE MELODY IS :-

```
M01 V1*H2 I1021
M02 V1*Q07 I5 Q.7
M03 V1*H4 Q20
M04 V1*W1
M05 V1*H2 I1021
M06 V1*Q07 I5 Q.7
M07 V1*Q4 I20 Q11
M08 V1*W0
```

THIS FILE MAY NOW BE SAVED TO DISK OR TAPE UNDER A DESIRED FILENAME.TO DO THIS THE COMPILER MUST BE IN COMMAND MODE AND GIVEN THE COMMAND - WRITE <FILENAME> -.THE FILE WILL THEN BE DOWNLOADED EITHER TO TAPE OR DISK.IF THE FILE IS SAVED TO DISK IT WILL AUTOMATICALLY BE GIVEN THE EXTENSION - /ORC - .FILES MAY BE READ BACK INTO THE COMPUTER BY USING THE COMMAND - READ <FILENAME>.

WE HAVE NOW TO PREPARE THE OTHER THREE PARTS FOR THE ARRANGEMENT AND THIS IS DONE IN THE SAME WAY AS FOR THE MELO WE WILL FIRST ADD THE BASS LINE,THEN THE TENOR LINE AND FINALLY THE ALTO LINE TO GIVE THE FULL ARRANGEMENT AS ATTACHED.TO COMPLETE THE ARRANGEMENT WE MUST SET A TEMPO MARKER,DESIGNATE THE VOICES AND INCLUDE ANY REPEATS AS NECESSARY.

TEMPOS ARE SET BY INCLUDING AN OPERAND LIKE THIS:-

NQ=70

THIS TELLS THE COMPILER THAT THE MUSIC IS TO PLAYED AT A SPEED OF '70 Q' BEATS TO THE MINUTE.ANY OF THE NOTE VALUES MAY BE USED AS THE SECOND LETTER OF THE TIME SIGNATURE EXCEPT FOR 'W' OR THE WHOLE NOTE.THE VALUE AFTER THE EQUALS SIGN IS NOT SEVENTY BUT THE HEX VALUE 70.THIS VALUE MAY BE SET TO HEX VALUES FROM 'FE' TO '80'.THESE VALUES EACH CORRESPOND TO THE WEIGHTED VALUES OF KEYS 1 TO 7 WHICH,IF PRESSED WHILST A PIECE OF MUSIC IS PLAYING,WILL ALTER THE TEMPO OR SPEED THE PIECE IS PLAYED AT.TO GET HIGHER TEMPOS THE TIME SIGNATURE MAY HAVE TO BE MODIFIED.IN THESE CIRCUMSTANCES A TEMPO MARKER FOR EXAMPLE 'NQ=' MAY HAVE TO BE CHANGED TO 'NI=' TO GIVE DOUBLE OR HALF THE SPEED DEPENDING ON WHETHER THE NOTE VALUE IS GETTING LARGER OR SMALLER.TEMPO IS NORMALLY SET AT THE BEGINNING OR A PIECE BUT ORCHESTRA-80 ALLOWS TIME SIGNATURES AT ANY POINT IN THE MUSIC SO THAT THE PIECE OF MUSIC CAN BE SLOWED DOWN OR QUICKENED AS IT PLAYS.

ANOTHER PARAMETER TO BE SET UP AT THE BEGINNING OF THE PIECE OF MUSIC IS THE VOICES DESIGNATED TO EACH PART.THE VOICES THAT CAN BE DESIGNATED ARE TRUMPET,OBOE,CLARINET AND PIPE ORGAN WHICH IS THE DEFAULT VOICE IF NO VOICES ARE DESIGNATED.A TYPICAL VOICE PARAMETER LINE WOULD BE:-

V1YA V2YB V3YC V4YD

THIS TELLS US THAT WE HAVE ASSIGNED THE TRUMPET VOICE TO PART ONE,THE OBOE VOICE TO PART TWO,THE CLARINET TO PART THREE AND THE PIPE ORGAN ON THE REMAINING FOURTH PART.THE VOICES MAY NOT NECESSARILY BE IN THIS ORDER AND THEY MAY BE ASSIGNED TO SUIT THE SCORING OF THE PARTICULAR PIECE OF MUSIC.A FURTHER ENHANCEMENT WHICH MAY BE SET WHILST IN THIS PARAMETER IS WHERE THE SCORING OF THE MUSIC IS FOR DIFFERENT INSTRUMENTS IN DIFFERENT CLEFS.AS A VOICE CAN BE ASSIGNED TO A PART THEN EACH VOICE CAN BE DEFINED AS PART OF A DIFFERENT CLEF.THIS IS DONE BY USING 'U' AND A NUMBER WHICH IS THE DISTANCE BETWEEN MIDDLE C (0) IN THE CLEF BEING DEFINED AND MIDDLE C (0) IN THE TREBLE OR ORIGINAL CLEF.AN EXAMPLE WOULD BE:-

V1 U-2 V2 U-6 V3 U-8 V4U-C

THIS DEFINES VOICES 1-4 AS - SOPRANO,ALTO,TENOR AND BASS.WE DO NOT NEED TO USE '*' OR 'AT' SIGNS AS ALL CLEFS ARE DEFINED RELATIVE TO THE TREBLE CLEF.ONE USEFUL TRANSPOSITION IS V4U-7 WHICH LOWERS VOICE 4 ONE OCTAVE TO GIVE A LOW BASS VOICE.

IN SOME CASES THE COMPUTER MUSICIAN MAY WISH TO MOVE THE WHOLE PIECE OF MUSIC UP OR DOWN IN THE TREBLE CLEF.TO DO THIS ANOTHER PARAMETER LINE IS ENTERED AT THE BEGINNING OF THE MUSIC FILE LIKE :-

<2

THIS PARAMETER TELLS THE COMPILER TO MOVE THE WHOLE PIECE OF MUSIC UP TWO SEMI-TONES (HALF-STEPS).TO TRANSPOSE UPWARDS WE USE '<' AND DOWNWARDS '>'.THIS IS MAINLY USEFUL IN PREVENTING THE DISTORTION

OF VOICES IN THE HIGHER RANGES OF THE SYNTHESIZER.ONE NECESSARY PARAMETER TO BE SET AT THE BEGINNING OF THE CODING IS THE KEY OF THE MUSIC BEING TRANSCRIBED.THIS IS SET TO A NUMBER OF FLATS (&) OR SHARPS (#) SO THE KEY OF 2 SHARPS WOULD BE CODED :- K2# AND THE KEY MAY BE CHANGED AT ANY POSITION IN THE MUSIC.

TO CONCLUDE I WILL QUICKLY RUN THROUGH OTHER MUSICAL NOTATION WHICH THE COMPILER WILL SUPPORT.

MUSICAL NOTATION	ORCHESTRA-80 CODE
PHRASE	EXAMPLE P01 - A NUMBER OF MEASURES ARE DEEMED TO MAKE A 'PART' WHICH IS DEFINED BEFORE THE GROUP OF MEASURES
REPEAT	EXAMPLE R01 - MEANING REPEAT THE PART NUMBERED 1
SHARPS	CODED # - RAISES A NOTE BY HALF A TONE OR STEP
FLATS	CODED & - LOWERS A NOTE BY HALF A TONE OR STEP
TRIPLETS	THREE NOTES IN THE TIME OF ONE - I.E CODE I:121 - MEANS PLAY NOTES D,E &D IN THE TIME OF 1 EIGHTH NOTE.
ACCIDENTALS	CODED % - WHERE A NOTE SHOULD BE SHARPENED OR FLATTENED IN THE KEY OF THE MUSIC BUT IS NOT TO BE
REITERATION	WHERE A PARTICULAR STRING OF NOTES REPEAT IN A BAR.CODED (Q1234)2 SHOWS FOUR QUARTER NOTES - D,E,F & G BEING REPEATED TWICE.
ARTICULATION	THE DETACHING OF ONE NOTE FROM THE NEXT BY INSERTING A SMALL SILENT SPACE BETWEEN THE NOTES.ORCHESTRA-80 SUPPORTS 2 KINDS OF ARTICULATION - SHORT AND LONG - CODED ' FOR SHORT AND * FOR LONG.
STACCATO	THE SHORTENING OF NOTES FOR EFFECT. AGAIN BOTH LONG AND SHORT STACCATO ARE SUPPORTED BEING CODED , AND ; RESPECTIVELY

THIS IS JUST A SMALL INSIGHT INTO THE WORLD OF ORCHESTRA-80 WHICH,IN MY OPINION,IS A WONDERFUL TOOL FOR THE ACCOMPLISHED OR BUDDING (COMPUTER!) MUSICIAN.ORCHESTRA-80 WILL JUST ABOUT DO ANYTHING YOU WANT IT TO DO AND WILL GIVE MANY HOURS OF PLEASURE WHETHER IT IS USED FOR JUST EXPERIMENTING IN MUSIC OR FOR MUSICAL ARRANGEMENTS.IF YOU ARE INTERESTED FURTHER THERE IS NOW A STEREO VERSION WITH AN OPTIONAL FIFTH VOICE TO SUPPORT PERCUSSION EFFECTS CALLED ORCHESTRA-85 (ORCHESTRA-90 FOR THE MODEL III) OR IF YOU HAVE DISKS AND A 100% SPEED UP MODIFICATION YOU CAN NOW HAVE AN ADD-ON PROGRAM TO ORCHESTRA-85 CALLED PIANO-85 WHICH ALLOWS YOU TO DETERMINE THE ACTUAL WAVEFORM OF THE VOICE YOU ARE PRODUCING.THIS WILL GIVE YOU ALMOST TRUE SYNTHESIZER CAPABILITIES ON A MICROCOMPUTER.

I HOPE THIS HAS INTERESTED YOU AND PLEASE DO NOT HESITATE TO CONTACT ME IF YOU WANT ANY HELP WITH ORCHESTRA-80 OR WISH TO SWAP 'ORCH' FILES

CLIVE KEECH
10 WOLVERTON ROAD
HAVERSHAM
MILTON KEYNES
MK19 7AA

That excellent set of machine code sub-routines from Racet Computes called GSF is supplied on cassette and the manual which comes with it describes how to save the routines to disk. However, life is not so easy for those of us who still persevere with floppy tapes. The routines cannot be saved using SYSTEM/CMD under XBAS as they are not written as a contiguous block, consequently, rather more subtle techniques have to be used.

What follows is largely the work of Laurie Shields, who devised this technique for me some months ago, and I am very grateful to him for it. As with David Washford's contribution in the June Newsletter (p.4), the writing up of this technique has been prompted by conscience pricking as a result of Laurie's letter published in the April Newsletter.

Machine Code programs can be saved to F-tape by the simple expedient of entering the following command from the keyboard:-

@SAVE"FILENAME/EXT",(sectors),(start address),(entry address)

(Where the entry address is optional, allowing the program to be @RUN if used.)

Each sector is 256 bytes long and the F-tape ALWAYS records a complete 256 byte sector, NEVER a part of one. If the end of memory is reached before a complete 256 byte sector has been recorded then the "infernal machine" goes back to the beginning of memory and starts recording the ROM until all 256 bytes have been used up. (With interesting consequences on reloading!) In order to prevent this, it is vital always to begin recording 256 x no. of sectors to be recorded, addresses BEFORE the end of memory.

Unfortunately, in the case of GSF, the program is not an integral number of 256 bytes long and thus simply using the appropriate MEMORY SIZE address as the start address in the @SAVE procedure results in chaos on trying to reload. To prevent this, the start address to use must be the next integral of 256 before GSF starts (16K=29696, 32K=45568, 48K=61952) i.e. 254 for 16K, and 216 for the others, addresses lower than the memory size addresses quoted in the GSF manual.

When reloading it is important to remember that, on power up, the appropriate one of the values quoted above should be used in response to the MEMORY SIZE? prompt rather than the GSF manual value. If the latter is used, the routines will partially load over the stack which is only 50 bytes below the MEMOk.' SIZE address on power up, again causing chaos and not recommended!

Provided all the above is followed, GSF can be saved to, and loaded from, F-tape successfully in the normal way. However, following the rest of the cassette loading procedure given in the manual, i.e. entering CLEAR, does not appear to cause GSF to autostart when called from a BASIC program. The reason seems to be that the appropriate entry address has not been located in the USR memory addresses (16526 and 16527 - as described in the LEVEL II BASIC MANUAL). These entry addresses for GSF are given in the manual on the last page in the instructions for saving to disk under "cccc".

There are various ways to overcome the problem, the easiest being to include a POKE statement line in the BASIC program BEFORE the first call to GSF i.e.

POKE 16526,LSB:POKE16527,MSB

The MSB and LSB values are calculated from the "cc " values given in

the GSF manual for the appropriate size of memory.

Alternatively, a small autostart machine code routine can be incorporated at the beginning of GSF before @SAVE-ing it using some of the spare 254/216 bytes created by having to save an integral number of 256 bytes. The assembler program for the routine is as follows:-

```
00100   ORG             ;Convenient starting point before
00110                   ;GSF according to size of memory.
00120   LD    HL,ccccH  ;Load HL registers with entry address
00130                   ;obtained from GSF manual
00140   LD    (16526),HL ;Load USR addresses with contents
00150                   ;of HL registers.
00160   JP    1A19H     ;Jump back to BASIC
00170   END
```

This can be converted into machine code using any of the assembler programs on the market. I do not own a copy of ZEN or MACRO-80 so I have no idea whether either, or both, is/are the "best thing since sliced bread". However, as part of the job lot which I acquired when I bought my TRS-80 second hand, was a copy of EDTASM which did the job perfectly well and joined the result to GSF.

Using:-

@SAVE "GSF",14,(start address),(entry address)

it is now possible to save GSF to F-tape and then auto start it using @RUN "GSF" before loading the BASIC program with which GSF is to be used. (N.B. In this case, "entry address"="start address"= ORG address in the above program.)

For those who do not have an assembler program at all, the machine codes corresponding to the above program are (in HEX):-

21 LSB MSB (of relevant entry address) 22 8E 40 C3 19 1A

This can be put into the appropriate memory locations prior to @SAVE-ing either by using a monitor program, or even the "M" facility of the humble T-BUG program (which I also acquired in my job lot). Or, again before @SAVE-ing, the data can be converted into decimal giving:-

33 LSB MSB (of entry address) 34 142 64 195 25 26

and the following BASIC program run:-

```
10 DATA (decimal values as above)
20 FOR I=1 TO 9
30 READ D
40 POKE ADDRESS+I-1,D
50 NEXT
```

(Where ADDRESS is the appropriate starting point for the size of memory being used.)

Laurie originally asked me to write this up so that even a beginner like me could understand it.

R. C. BAILEY

VIDEO GENIE, EPSON, and TINY PASCAL

Having had no response to my appeal in the December newsletter, I am now in a position to reply to myself. I attach a set of Pascal procedures for Epson printing, cassette read/write, CLS, and WRITE AT (equivalent to Basic PRINT @).

The cassette procedures are based on using Basic ROM calls and writing a byte at a time. This approach should enable the user to be able to reproduce any desired tape format - SYSTEM, or Basic CSAVE, or Basic PRINT£-1, or even a private format which can be read back only by the user's own loading routine!

The modus operandi is therefore to write a tape with one call of LDR, followed by repeated calls in a loop of BYTOUT(X) where X is the address of the memory location where each byte in turn is to be found, finally calling XMOT to stop the cassette motor.

Conversely, reading the tape is accomplished by one call of SKIP, a loop of BYTIN(X) where X is now the memory address to which each byte is placed, finishing with XMOT again.

Both BYTOUT and BYTIN pass bytes via location 16320, at the bottom left corner of the screen, which gives a comforting visual display of the progress of operations. Some care may be needed with timing - errors will occur if you write bytes away to tape faster than your reading routine is able to handle them on re-input. Hence, if the input loop contains any very significant processing instructions, the writing loop may need to contain dummy instructions giving an equivalent delay.

The procedure UTIL plants the necessary Z.80 code in high memory locations 49000-49098, and it follows that UTIL must be called once before any of the tape procedures are invoked.

AT(P) is a function to control the position of the cursor on the screen, and thus the layout of displays. It is used in a WRITE parameter list as in this example:

```
CLS; WRITE(AT(256),X1,AT(512),X2,AT(64),X3,X4,X5 ...etc).
```

Being a perfectly normal Pascal function however, it can also be called elsewhere eg: CURSOR:=AT(260); READ(A,B,C) would position the cursor prior to keyboard entry of data required by the program.

John Gale Hemel Hempstead 0442-53643

```
(*A package of Tiny Pascal utilities*)
(*-------------------------------------------------------------*)
PROC LP(CH);  (*sends a single chr from VG to printer*)
BEGIN WHILE INP(253)<>63 DO;
   OUTP(253,CH)
END;
(*-------------------------------------------------------------*)
PROC LLIST; (*lists a Tiny Pascal source *)
VAR LOC: INTEGER;
BEGIN LOC:=MEMW(16768); LP(15);
   REPEAT IF MEM(LOC)=9 THEN BEGIN LP(32);LP(32);LP(32) END
      ELSE IF MEM(LOC)<>10 THEN LP(MEM(LOC));
      LOC:=LOC+1
   UNTIL (LOC-1)=MEMW(16770)
END;
(*-------------------------------------------------------------*)
PROC CLS; BEGIN WRITE(28,31)  END; (*clears screen*)
(*-------------------------------------------------------------*)
PROC POK9(L,A,B,C,D,E,F,G,H,I); (*used by UTIL *)
BEGIN
   MEM(L):=A;MEM(L+1):=B;MEM(L+2):=C; MEM(L+3):=D;
   MEM(L+4):=E; MEM(L+5):=F; MEM(L+6):=G;
   MEM(L+7):=H; MEM(L+8):=I
END;
(*-------------------------------------------------------------*)
PROC LDR; (*ROM calls 530 and 644 to start motor/write leader*)
   BEGIN CALL(49026) END;
(*-------------------------------------------------------------*)
PROC BYTOUT(X); (*ROM call 612 to output one byte*)
   BEGIN MEM(16320):=MEM(X); CALL(49044) END;
(*-------------------------------------------------------------*)
PROC XMOT; BEGIN OUTP(255,0); (*turn off tape motor*)
   MEM(16445):=(MEM(16445 AND 248)) END;
(*-------------------------------------------------------------*)
PROC SKIP; (*ROM calls 530 and 662 to start motor and*)
   BEGIN CALL(49060) END;      (*skip over leader*)
(*-------------------------------------------------------------*)
PROC BYTIN(X); (*ROM call 565 to input one byte*)
   BEGIN CALL(49078); MEM(X):=MEM(16320) END;
(*-------------------------------------------------------------*)
PROC UTIL; (*poke Z.80 code into high mem*)
BEGIN WRITE(13,'LOADING TAPE UTILITIES');
   POK9(49000,241,197,213,229,221,229,253,229,245);
   POK9(49009,201,0,0,253,225,221,225,225,209);
   POK9(49018,193,241,19,19,201,0,0,0,245);
   POK9(49027,205,104,191,62,1,205,18,2,205);
   POK9(49036,132,2,195,116,191,0,0,0,245);
   POK9(49045,205,104,191,33,192,63,126,205,100);
   POK9(49054,2,195,116,191,0,0,245,205,104);
   POK9(49063,191,62,1,205,18,2,205,150,2);
   POK9(49072,195,116,191,0,0,0,245,205,104);
   POK9(49081,191,205,53,2,33,192,63,119,195);
   POK9(49090,116,191,0,0,0,0,0,0,0);
   WRITE(' COMPLETED',13)
END;
(*-------------------------------------------------------------*)
FUNC AT(P); (*positions cursor on screen*)
   BEGIN MEM(MEMW(16416)):=32;MEMW(16416):=15360+(P AND 1023);
      AT:=0 END;
(*-------------------------------------------------------------*)
BEGIN LLIST END.
```

ODDS & ENDS

WANTED

Printer to fit direct onto TRS keyboard.
Price Range £150 - £200

D. Burnett, Warrington (0925 67685)

WANTED

Did anybody buy a SOLOLOAD from a company in Brighton.

John Taylor from Gloucester wants to know.

FOR SALE

34 & 40 way Gold plated edge connectors

Richard Creak 0245 - 413725

40 way £3.20
34 way £2.80

Postage & Packing 30p.

FOR SALE

John Pomfret Atherton 884738

TRS80 16k L2, L/C &Keypad
Centronics P1 with cable.

£275 Keyboard & £120 Printer.

FLIGHT COSTING PROGRAM

A local Milton Keynes member has developed this over the last 3-4 years. Ring Brian Pain.

MODEL II & CP/M & DBASE II

Ken Dudman has the above in an ironmongery business (Kent)

VIDEO GENIE & CP/M & A PUB

Chas Phillips from Old Northampton would like some ideas & help.

SWINDON IS BACK IN NOVEMBER '83

Ideas for talks etc welcome or I'll decide !
Brian Pain.

FOR SALE

TRS80 Keyboard, Interface, and LP SKGP8 D/M Printer
Lots of software & cassette.

£550 ono, might split

Mike Russell 0908 - 604398

Tandy 4 Colour Printer to BBC

try Brian Sharrocks 04606 4444, or Andrew Donald London.

Model III & VISICALC

In London several members are helping each otheron the above.

Paul Taylor West London.

Model III & Comms.

Mike Morris from Bracknell can help anybody wanting to have a go.

JULY EL CHEAPO WEEKEND IN MILTON KEYNES

This is all Dusty Miller's fault. We have a large ex-school in Stony Stratford suitable for a weekend, as an alternative to the more expensive Swindon. A rest room can be used for light sleepers as well as the usual youth hostel, or bed & breakfast within 800 yds. Caravans and similar are no problem, with ample hard standing next to the systems room. I will set up 4 lecture rooms for small group work, and the weekend will have DOS as its theme. Brian Pain

TURN YOUR MICRO

INTO A **MACRO**

5 MEGABYTE	HARD DISC	£1650
10 MEGABYTE	HARD DISC	£1895
1 MEGABYTE	FLOPPY BACK UP	£ 600

OR

HARD DISC BACK UP
NO FLOPPIES
20 TO 40 MEGABYTES

5 MINS TO BACK UP 10 Mb

20 Mb. (10 FIXED/10 REMOVABLE) £4495

MULTI TERMINAL CAPABILITY

4 48K TRS 80's
20 MEGABYTES
£8850

DISKETTES	WABASH	BOX 10	POSTAGE & V.A.T.
LOWEST PRICE IN THE COUNTRY.	SSSD	18·50	INCLUDED !!
	SSDD	22·50	SEND CHEQUE WITH ORDERS.
	DSDD	31·50	

AMBASSADOR COMPUTERS LTD.

ASHLEY LANE WORKS
SHIPLEY
WEST YORKS.

TEL 0274 595941

NATIONAL TRS-80 & GENIE USER GROUP

VOLUME 4 ISSUE 10 APRIL 1983

CONTENTS

2. Information on the Group
3. Editorial
3. Guidelines for contributors
4. Members' letters
7. Problem section
8. Machine language addition in BASIC
10. Zap for EDTASM/CMD to give text recovery
11. Boot for Newdos+
12. Writing a smart terminal program in C
13. A TRSDOS/Newdos80 problem and other things
17. Edit
18. Using C
20. Dotprint: a brief review
21. FORTRAN tutorial Part 2
25. Building someone else's Maplin modem
28. LISP/80

NATGUG NEWSLETTER

INFORMATION ON THE GROUP

Membership of the Group is by subscription to the Newsletter, which is published monthly. Membership details are obtainable from the Group Secretary. Membership of the Group is open to anyone with an interest in the TRS-80 range of microcomputers, and compatible systems such as the Video Genie.

Details of the Group accounts and the constitution of the Group are obtainable from the Secretary.

Members requiring assistance with problems related to the TRS-80/Video Genie may call the Secretary. An attempt will be made to put them in touch with a member who can help with the problem.

Workshops are arranged from time to time in various parts of the country.

Sub-groups exist in many areas. A list is provided at the back of the Newsletter from time to time. Sub-groups also exist for medical and educational users.

The Group runs 80-NET, a computerised bulletin board service that is available to all users of microcomputers.

The Group maintains three software libraries (Models I, II and III) which are free to members. Library lists are obtainable from the Secretary. The medical and educational sub-groups maintain their own libraries.

A membership list is obtainable on disk from the Secretary.

Back numbers of the Newsletter are available.

Please send all contributions for the Newsletter to the Editor.

Chairman and Newsletter Editor:-

Leon Heller,
8 Morris Walk,
Newport Pagnell,
Bucks. MK16 8QD.
Tel: (0908) 613004

Secretary and Newsletter Publisher:-

Brian Pain,
24 Oxford Street,
Stony Stratford,
Milton Keynes.
Tel: (0908) 564271

Librarians:-

Model I: Leighton Davies, 105 Caerau Road, Caerau, Glamorgan. Tel: (0656) 738337.

Model II: Jim Hutton, 25 High Street, Stroud, Glos. Tel: (045 36) 4423.

Model III: Nick Clay, 6 Rectory Lane, Byfleet, Surrey. KT 14 7LL. Tel: (09323) 42348.

EDITORIAL

By a superhuman effort I have managed to put this issue together by the 17 April, so you should get it well before the end of the month.

I hope members will excuse my self indulgence in concentrating on the C programming language, but I do feel it is the best general purpose language available at present, although you need floating point for most applications programming. The more expensive versions have floating point, of course.

Lots of members seem to be getting into communications, now that relatively cheap modem kits are available, and new computerised bulletin boards are springing up all over the place. This is all to the good, as with more bulletin boards around the country, more users will be able to access them without having to make long-distance telephone calls. Eventually, most users should be able to get onto one with a local call, provided we have enough people willing to provide such a service.

Leon Heller

GUIDELINES FOR CONTRIBUTORS

1. If possible, send material in printed form.
2. Scripsit or Pencil disks/cassettes are acceptable.
3. Ensure your ribbon is in reasonable condition.
4. Printer output should be on A4 paper if possible.
5. If you send in handwritten material, write legibly.
6. Do not fold the sheets when posting them. Use a large envelope and keep them flat.

Leon Heller

All the front page was by John Carne, who is the contact for North London.

WORKSHOPS

MILTON KEYNES WEEKEND

JULY 29,30 & 31

SWINDON WEEKEND

NOVEMBER 18,19 & 20

MEMBERS' LETTERS

I think the time has come for me to put pen to paper, I have at last received the copy of the Newsletter that kept being sent back to Brian, as a matter of interest it has no month or volume reference on it.

I keep reading letters from people who complain that the Newsletter is too erudite for them, that is only because the only people that write letters now are those who have solved all the simple problems of a few years back & newcommers think they may be thought stupid if they write & ask for answers to simple questions, the ones we needed answers to some time ago.

It's no good complaining to you, you can only publish the letters you receive, if they want answers, they must ask the questions, there is always someone willing to give a helping hand especialy if they have carefully filed all their back copies of the Neswletters.

That's enough in Brians & your defence for the time being.

Here are a few notes about the OMIKRON package 'THE WORKS', Tim Bourne made some comments in the May issue last year & these are in addition to them.

CP/M is called an Industry standard, so are MBasic-80 & CBasic, Wordstar is supposed to be the last word in wordprocessers, well, if they are Industry Standards no wonder there is a recession.

I would have thought that by now Digital Research could have provided a better Editor considering how much money they must have raked in, no wonder programmers walked round with a dazed look on their faces, fancy trying to write a CBasic program with 'ED'. They haven't even re-written their manuals.

I also though Wordstar would solve the problem but that is so SLOW & has so many different instructions just to move the cursor let alone anything else that I have given up in disgust & gone back to my trusty 'PENCIL', (the 'Dosalum' effect has been cured with a patch from R.H.Schubert System Software in America called PENMOD).

Mbasic-80 syntax is much too pendantic, some places you MUST put a space others you MUST NOT, not at all user friendly, CBasic gives very missleading error messages when compiling, the errors are never where the messages say they are & that's just for starters, I could go on for pages.

All in all I am sorry to say it was a waste of money, stil we live & learn. I suppose I have been spoilt, writing programs using NEWDOS-80 V2 is so much easier.

When I eventualy received Wordstar there was no manual with it only a book called 'Introduction to Wordstar' which states that it is not a lot of use on it's own, the manual is esential, still it was easier to understand than Digital's ED manual & gave me enough information to try the package. Why Omikron couldn't supply a manual makes me wonder how they got hold of Wordstar in the first place although it had all the correct labels & registration documents.

John Kilpatrick.

Dear Group:

Enclosed is a check (cheque) in the amount of $15 (ca. 7 pounds) for a subscription to the monthly newsletter.

Obviously, I have good access to Model 1 programs on this side of the Atlantic; however, I assume there must be some interesting ones that are more British in flavor--running for the House of Commons, test match cricket, and whatever, which are not available here. I should be particularly interested in anything along those lines.

Jim Glanville,
P.O. Box 4591,
Roanoke,
Va. 24015,
USA

Many thanks for the use of your Password & account No. to give my friend and I access to Your Electronic Mailbox facility. It was very enlightening and gave us a taste of things to come. I must admit, however, that our success in linking up with the host system proved to be less than straightforward, nevertheless, we did eventually SUCCEED.

Equipment Used :- Tandy Model III
Accoustic Coupler
LDOS 5.1.3.

Whilst the keys are Tapping, I would like to give my heart felt thanks (& much belated) to a Gentleman named Laurie Shields. During my "L Plate" stages he was a great help in writing a Patch to enable my Genie to drive an ex Honeywell Terminet 300 Printer using RS232. Again Laurie Thanks.

On another subject, is there a circuit diagram in the house, for a Doubler suitable for my Genie 3013. I would be very gratefull for any help with this matter as I would like to try my hand at building one.

D.A.Morris
45 Oldstead
Crownwood
Bracknell
Berks
RG123JE
20/03/83

Having been a member for a few years, I thought it time to make a contribution, so please find enclosed a couple of items which should be of some interest. I hope the ribbon strength is good enough. It is for my use but may not be good enough for copying. If not, I can supply the text in scripsit form if you let me know.

I would like to confirm the opinion of most members that the newsletter is of great use - well done all concerned for their effort in turning out every issue.

I can supply more articles but what articles do members want to see? My good points are pretty good at basic and medium experience at machine code. I have written M/C programs that work which surprises me at least! although I usually modify programs from other people for my own use

My bad point is chronic slowness at actually doing something about it!

Keep up the good work.

G. Brittain.

Please find enclosed a short software review that may be of interest for the newsletter.

There seems to be a rather serious slant to the articles published, mostly on utilities & reviews, but rarely anything on the fun side of microcomputing.

Prior to joining NATGUG, I was involved in writing a regular column & articles for the now defunct 80 UK newsletter. The column, called 'THIS & THAT', covered any tips or hints & short routines for use in games etc., (do you rember the INKEY$ simulator for Level 1 at one of the early PCW shows ?).

If you think that something along these lines on a regular basis would be suitable for the newsletter, I would be pleased to oblige, perhaps bi-monthly at first. It would rely on feedback from members of course.

Pete Bamfield

Having recently pleaded for the continuance of Workshops, I was very disappointed in my fellow mankind at Swindon - a library case containing 9 DD,d/s disks of games somehow seems to have fallen out of my box which was on the floor. Since I don't have much time for games, I don't keep any backup copies of such disks, so I would be extremely grateful if the Member who has accidentally acquired this box would return it to me - I am fully prepared to refund postage and packing costs. The address is:

David Washford
1 Alexandra Road
Bournemouth
BH6 5JA

I do hope that someone out there will be kind enough to restore my faith in mankind! I don't want to bore the pants off you, but I only see my kids every three months.

Thank you

PROBLEM SECTION

Can anyone help me to get a copy of the TRS-80 Technical Manual suitable for the Model I, Level II ? I thought I should get one before they went off the shelf, but seem to be too late. My local Tandy shop seem very doubtful of being able to help, and suggested the Users' Group.

Also, may I ask if it is possible to read program lines from tape as a character string, complete with line number? I had in mind "reading" a program and reporting the line numbers which include a specified character string (or even report the whole line). It could be useful, for example, if one were making changes involving a particular variable - rather like the facility offered by a "global edit".

Another sigh over the editor in Level II - can one somehow make it move on to the next line, or the next line which is not a null string?

Many thanks if you are able to help, either directly or via another member or the Newsletter.

Peter Knewstubb,
175 Huntingdon Road,
Cambridge,
CB3 ODH

Lowe Electronics kindly gave me your adress and suggested that you may be able to help me with the more technical questions I have been raising with them. Firstly, I am eager to join the User Group so please send me joining instructions, literature or details of available literature, including past news letters, etc., as soon as possible.

Secondly, I would be grateful for some advice, but do not let this delay the above. I have been using a computer for three years, and now have a 16k Genie I with Lowe's high resolution graphics attachment. I am anxious to purchase a printer. Lowes tell me that the Print Screen function does not operate with the HIRES. My manual does not explain how to instruct a printer to print a normal graphics display, and certainly not how to print the HIRES display, nor whether it is even possible! Can you help? In particular, could you advise on a good printer for this purpose, that does not cost the earth? Lowes have recommended their own EG3085 (of course) which looks like a Seikosha and costs £357 + VAT. Is this the best value for money?

I have other problems, but these can wait until I am a member and have possibly made other contacts, and, who knows, might even be able to give advice! However, I really do want to get on with printing graphics.

B. Mason,
Highlands,
Bromsash,
Ross-on-Wye,
Herefordshire,
HR9 7PR.
Tel: (098 981) 353

MACHINE LANGUAGE ADDITION IN BASIC

This is a machine language program to add a number of integers stored in an array. I needed such a facility to rapidly add many small numbers and this routine does just that. The program cannot simply be relocated as it requires 2 values to be entered before running, so one must be poked into memory. However I will indicate the areas you will need to recalculate. Assembler users should of course have no problem! the program should work with model 1 and model 3. This was originally designed on a model 1 with disc basic.

```
00010          ORG   0E800H          ;Anywhere suitable
00020
00030
00040 ADDINT   CALL  0A7FH           ;Get total entries to count
00050          LD    (TOTAL),HL      ;Save
00060          LD    HL,0000H        ;Zero accumulator register
00070          PUSH  HL              ;Save it
00080          LD    HL,(S1)         ;Get contents of array
00090 NEXTAD   LD    C,(HL)          ;Into BC
00100          INC   HL
00110          LD    B,(HL)
00120          INC   HL
00130          EX    DE,HL           ;Save array position
00140          POP   HL              ;Get value so far
00150          ADD   HL,BC           ;Add array value to total so far
00160          PUSH  HL              ;Save it
00170          LD    HL,(TOTAL)      ;Get no of entries to do
00180          DEC   HL              ;1 more done
00190          LD    A,H             ;Is it zero?
00200          OR    L
00210          JR    Z,DONADD        ;Finished
00220          LD    (TOTAL),HL      ;No - so save it
00230          EX    DE,HL           ;Get next array seq
00250          JR    NEXTAD
00260
00270 DONADD   POP   HL              ;Get total back to HL
00280          JP    0A7FH           ;Return value
00290
00300 TOTAL    DEFW  0               ;Total entries to count
00310 S1       DEFW  0               ;Where to get them from
00320
00330          END   ADDINT
```

When entering from basic, the address of the array must be poked into memory which corresponds with 'S1'. The usr function is then issued with the total number of entries to count. If the array dimension is exceeded, the program will either crash or give false results.

```
100 CMD"LOAD ADDINT/ML"   REM For disc users only
110 DEFINT A-Z
120 DIM A(200):DEFUSR=&HE800
130 FOR X=1 TO 200:A(X)=X:NEXT
140 X=VARPTR(A(1)):POKE &HE829,PEEK(VARPTR(X))
150 POKE &HE82A,PEEK(VARPTR(X)+1):X=USR(200)
160 PRINT"TOTAL =";X


200 DATA 205,127,10,34,(39),(232),33,0,0,229,42,<41>,<232>,78,35
210 DATA 70,35,235,225,9,229,42,(39),(232),43,124,181,40,6,34
220 DATA (39),(232),235,24,234,225,195,154,10,0,0,0,0

<<Total of 43 bytes>>
```

Above are the decimal statements for basic users. The values with () are references to 'TOTAL' while <> marks the use of 'S1'. If the program is moved, these values must be changed. 'TOTAL' resides at E827H in the above program. To convert to decimal, treat as E8-27. E8H=232D (E*16+8 where E=14,). 27H=39D (2*16+7). Your new addresses must be calculated and stored in place of the marked values. 'S1' is exactly 2 more than the address of 'TOTAL'

If you don't know how to store the data values into memory, a quick look through some back issues should help.

Line 140 is where it all goes on. Don't use any undeclared values after the 2 pokes as by doing so will move array A thereby making the just poked values incorrect. The address of the first required array value is stored in memory, and then X=USR(200) Means count 200 values. The program can count any number of consecutive values in the array. To count values 146-152, use A(146) not A(1), and use X=USR(7) to count them.

The sum of the values cannot exceed 32676 as only an integer value may be returned, but otherwise the program can count approximately 5400 integers per second. I hope that is fast enough!

If you have problems, please let me know. I'm on the phone at odd times. 021-358-2757. Leave a message if I'm not there.

Gary Brittain, 45, Waddington Ave, Great Barr, B'ham, B43 5JF.

ZAP FOR EDTASM/CMD TO GIVE TEXT RECOVERY

Yes Laurie and I know that some of you NEWDOS 80 users (very wise) are using EDTASM (no comment), so here is a little something that might be of interest in case you ever "lose" or think that you have lost your text through a reboot. I can claim absolutely no credit for the clever bit that does the recovery, I think I came across it when reading back issues of the Newsletter. This is something I would strongly recommend - its amazing how much you learn in a short time, and when you come back a few months later, what was previously incomprehensible or of no use may now be just what you're looking for.

I'm afraid that I just cannot find the article in question so with apologies to the person who came up with the original idea, here goes. The actual recovery is done by making a jump to 58DA (hex), and the original suggestion was to go into Debug and simply issue the instruction G58DA. If you happen to use a DOS where Debug is not so easily entered, this is a bit of a problem, so my mind turned to trying to use the BASIC * idea. For those not familiar with this feature of NDBO, if you have a reboot or some other problem when in BASIC, you can often get the program back by typing BASIC * from DOS. It really does work, but do make sure there is a space before the asterisk. For those who are not interested in how these things are actually done, skip on to the Zaps at the end.

When you type any command in DOS, what you have typed is saved up in a buffer in memory, until you press <ENTER>, when DOS will start to examine this buffer to see what you've typed. It will look for letters and numbers making up a program name and as soon as it finds anything else (eg a space or some other symbol) it assumes that it has a complete name, which it will then look for in the disk directory(ies).

At this stage one of the Z80 registers is left pointing at the next character in the input buffer. Our little patch will look to see if the character happens to be an asterisk, and act accordingly. The code is as follows.

```
                ;:**** PATCH FOR EDTASM/CMD ****

                ORG  7660H

7660 7E         LD   A,(HL)        ;EXAMINE THE CHARACTER
7661 FE2A       CP   2AH           ;IS IT ASTERISK ?
7663 C2006F     JP   NZ,6F00H      ;NO - COLD START EDTASM

7666 CDC901     CALL 01C9H         ;TIDY UP - CLEAR SCREEN
7669 C3DA58     JP   58DAH         ;RECOVER THE TEXT
```

I have put the code at 7660H, a little bit that appears to be unused in EDTASM. All that has to be done now is to zap this in along with the new execution address. Don't forget Zapper's First Law - ALWAYS WORK ON A COPY !! I normally use SUPERZAP and the necessary instructions are as follows.

```
SUPERZAP <ENTER>        from DOS
DFS <ENTER>             This will display file sector(s)
EDTASMX/CMD <ENTER>     Filename you have chosen for the copy
32 <ENTER>              Sector in the file to zap
MODB1                   Should display a cursor at byte B1 in
                        the sector display (no <ENTER> needed).
                        This and the next 11 bytes should be 00

        Now type the following (without the spaces)
        7E FE 2A C2 00 6F CD C9 01 C3 DA 58 <ENTER>

Y                       To confirm the modification
<ENTER>                 As prompted


I                       This will step on to sector 33
MOD53                   This byte should be just after 02 02

        Type the following (without the spaces again)
        60 76 <ENTER>

Y                       To confirm
<ENTER>                 As prompted
EXIT                    No enter required - exits to DOS
```

If all is well that should be it. Run the new program, and put in some text. Press the reset button, and when DOS is ready, type EDTASMX 8 (or whatever name you've chosen) and the text should have returned. Type P. to discover it's even at the line number it was before the reboot !!

Don Bannister 41, Myrtlefield Park, BELFAST, BT9 6NF
Home Tel: 0232- 668899

BOOT FOR NEWDOS+

Hands up everyone using Newdos +. Type LIB. What on earth does 'DEVICE' do. Nothing, except to return to DOS. Well here is a use for this command. In Newdos 80, there is a command to reBOOT the computer, just like pressing reset. Here is a little modification to do just that for Newdos +.

The LIB commands are stored in SYS1/SYS and if one is typed in as a command, SYS1 loads SYS6/SYS and passes control to it. So to modify the system we have to alter both programs. I trust you are familiar with superzap. If not then spend some time learning to use it.

Take 1 fresh disc and make a backup of your original. Put your original disc out of reach and load the backup in a drive. Run superzap.

Modify SYS1/SYS.
================

Select track 10, sector 1. At the lower part of the right hand display you will see LIB command words, such as FREE, DIR, DEVICE etc. First modify the word 'DEVICE' to say 'BOOT'. Type MODA9 to modify from byte A9.

```
Was :- 44 45 56 49 43 45
Is  :- 42 4F 4F 54 20 20        The 20H are important!
```

Press enter and answer 'Y'. The display should now read 'BOOT' instead of 'DEVICE'. That's half the work completed.

Modify SYS6/SYS.

Select track 13, sector 6. Type MODD6.

Was:- C3 2D 40 7E
Is :- C3 00 00 7E

Press enter and reply 'Y'. This change simply changes the jump address for the DEVICE command. Previously, typing 'DEVICE' did a jump to 402DH (jump to DOS). Now the jump is called by 'BOOT' which does a jump to 0000H (same as a reboot).

Testing - load the disc in drive 0 and press reset. This is necessary to load the new system files. Type LIB and one of them should be BOOT. If not, reload superzap and check SYS1. If all is ok, typing BOOT should reboot the system. If not, check SYS6.

When modifying SYS1/SYS, you could change the other command names by typing the hex equivalent for characters in place of those already there and filling excess characters with spaces. For example, FREE could become F and DIR could become D. It is simply a matter of choice. However, make sure that you only modify a backup of your dos. Never modify the original!!!, after all, you may not like the changes.

G. Brittain

WRITING A SMART TERMINAL PROGRAM IN C

Here is the listing of a partially completed smart terminal program that I am writing in C, using the C/80 compiler. If anyone is thinking of learning a new programming language, I think that the best way of going about it is to write a few simple programs at first, just to make sure you have got the hang of the syntax, and the process of editing and compiling the code (assuming the language is compiled, of course), and then jump in at the deep end and write a fairly complex program in the new language. This technique works for me, anyway.

I don't claim that this program is particularly well written or efficient, but I have tried to use most of the features of C, and have laid it out in the approved fashion with indented lines to make the program easy to follow. I have not included any comments which is rather naughty of me. The program is very modular, and as written allows full duplex communication with the remote system, and the transfer of files to it. The modules not yet written, for opening and closing a buffer, and saving the contents to disk, are represented by dummy stubs, which just return to the main program.

GOTO P. 27

A TRS80/Newdos80 Problem and other things by Roy Barber.

In the Feb issue of the newsletter Laurie Shields referred to a problem I had discussed with him. As this may be of interest to others I will give the full story. The newsletter was short on articles last month so I will also provide some additional information on my system which I hope will be of interest. At least it will help if Leon is short of material again.

The problem I had was that TRSDOS worked OK on my system but Newdos80 which I intended to use would not do a disk copy function. It appeared to perform OK in most respects but when doing a backup copy I got a verify error on track 99 each time. I later found an additional fault when saving BASIC programs. A low number line was added to the end of each program saved. When trying to delete the added line after loading the program the system would crash. I had been able to back up the master disk using the TRSDOS backup command with no problems.

My expansion interface is the LNW expansion board with 16k memory, disk controller, real time clock and RS232 sections only incorporated. The LNW has a TRS80 expansion plug on it to which I had attached an additional expansion box of my own, more on this later.

To track down the trouble I borrowed a friends TRS80 expansion box which when connected to my system instead of the LNW allowed Newdos to copy quite OK. The problem appeared to be due to the LNW interface (I thought.).

After some experimenting the solution was found. Sections of my system were disconnected one by one until the problem disappeared.

In order to explain it is necassary now to describe my system in a little more detail.

I have seperated the keyboard and CPU board of my TRS80 and have the CPU board located in a box with the LNW interface. On top of these I have an additional matching box containing additional expansion boards as follows: Decoder board for EPROMS, R/W memory, Printer, Parallel I/O, Serial I/O, Battery maintained RTC and some other ports for leds etc. My eproms contain a number of utility programs such as a complex LC driver, A TC8 high speed cassette operating system, a renumber prog. and a driver for my RTC. The eproms are located at C000H upwards with about 8k currently in use. This is the reason I only have 16k of additional memory in the LNW. The eprom programs require some scratchpad memory during their execution and when I started on the system I decided I would not use the popular area at around 4050H just in case I ever wanted to use disks. (The original TC8 system used this area of memory). I configured the Eprom programs to use a 256 byte area of scratchpad memory at F800H. To provide this memory I incorporated a 256 by 8 R/W chip at F800H and redirected the eprom programs to use this new scratchpad area. As this new RAM is well seperated from the TRS80 RAM no memory protection was needed at powerup. I hate such requirements also the need to enter dates and times on powerup.

To get back to the problem with Newdos, it was found to be solely due to the extra block of R/W memory at F800H. It appears that Newdos looks at the top of each additional block of RAM at powerup to establish how much additional memory is

available and if it gets a reply from the top of the block of memory it assumes the whole block is there. As Newdos uses all available memory for a copy function it was trying to use memory that was not there. My 256 byte block at F800H had not been completely decoded and repeated itself every 256 bytes up to FFFFH hence Newdos assumed there was memory below this from 32k up to a full 48k.

Well, thats the story. Removing the 256 byte RAM proved the point conclusively. However I have many programs I needed to keep on my TC8 high speed cassette system. I had built a remote controlled cassette unit using the Hart Electronics SF925 solenoid controlled deck which was quite a fast system and about as good as you can get using standard cassettes and for this reason I was not prepared to dispense with the scratchpad memory. Also I have a 4k eprom emulator which I often plug in at E000h to provide a further 4k of protected memory for system programs. I had installed a switch on the LNW disk controller to disable the disk system when required and as I did not want any more switches I decided to use the same switch to disable the additonal RAM when disk operation was required. This was easy to do as spare chip enables were available on the memory involved. Obviously I can not go into much more detail here but I will happily discuss the subject on the phone or by letter.

If anyone is considering adding the disk disable switch to allow reset without a disk boot I can recommend it. The subject has already been discussed in the newsletter but details about adding the switch to the LNW appears incomplete. Also LNWs method requires removal of the FD1771 disk controller chip. This is not nececcary if a small mod is carried out to the board.
I can recommend the LNW expansion board as a low cost interface (if you are good at fine soldering). I did have some memory problems initialy but these were overcome by using very short connections between the CPU and the LNW board. AS mine is located about an inch below the CPU board I got my connections down to about two inches.

Some of you may be wondering why I seperated the TRS80. The reason was to allow me to use an intelligent(?) keyboard when required. I can also plug in the original TRS80 Keyboard as neceassary. My keyboard has its own Z80, Eprom and 2k of R/W memory. It is based on a good, comprehensive commercial keyboard with 106 keys. This allows me to have many single entry keys for frequently used commands such as: gosub, return, delete etc. Also a full number pad is incorporated with all math functions such as +,-,/,=,<,> etc on single keys unaffected by shift. I have also put the Newdos 123, DFG and JKL functions on single dedicated keys. The board has 24 programable keys which can be loaded from the keyboard independantly. The big advantage with this is that I can have any command, phrase or line of text on a single key whilst using a dedicated program such as a word processor or assembler. I can put frequently used commands such as :

```
LD        A,(HL)      <enter>  or:
NOP                   <enter>
```

on a key and call it in one key press. When equired it will also be repeated at will if the repeat function is in operation. Very handy if you want 20 NOPs in your text.

Well is that enough ? Maybe Leons got room for a little more...

No one has mentioned the Canon disk drives so far. I needed small drives so had a good look at most drives available

before settling on the Canons.The well proven TEACS and the like were simply too big for my small work areea in my lounge. I already have a largish TRS80 system and with a Selectric typewriter and an Epson printer, cassette drive and other extras normal disk drives simply would not go in. As I also have a rather expanded Hammond organ in the lounge expansion of my share of the room would have been too much to expect of my wifes tolerance. I just had to get the disk drives into the space occupied by my ESF string floppy drive. I did not quite succeed but only used about a further half inch all round the ESF space. The drives are 2/3 height and look something like the Seimens units. The construction of the drives looks to be of typical Japanese quality. They are qiuet and quite fast. I am using 4 millisec. track stepping time under Newdos with no problems. If anyone is considering these drives I can recommend them but get a service manual from the importers, A.M. Electronics at Chiswick as the link selectable options inside allow almost any imaginable configuration of drive options to be set up. Also discussions with A.M.Elect. gave me the impression they were capable and willing to help users. I got my drives from Premier Publications of Croyden who were not very knowledgeable about them and only under pressure from me at time of purchase gave me a copy of a handbook on a totally different unit to the ones I was purchasing.They assured me that the information applied to my drives. It didnt !.

I went to disks rather reluctantly (They are expensive and do need a lot of memory for operation) but even using the TC8 or ESF high speed cassette systems I could not cope with large data files. In all other respects I was quite happy with my cassette systems. If you are not handling large data files I still think disks are an expensive luxury although I consider a high speed cassette system an essential for the model 1.
I think that if I was stuck with only TRSDOS I would still be unhappy with disks. I must give credit to Newdos80. I consider this is a superb operating system and much prefer it to LDOS or DOSPLUS which my friends use. I must admit it will probably appeal most to those who are system meddlers rather than business users. I must admit to using SUPERZAP almost as soon as I got Newdos. I was able to Zap into my programs a number of alterations I wanted so easily. The inclusion of EDTASM has also saved me a lot of work patching my EDTASM plus for disk storage. I patched it for the TC8 system and did not want to have to do more work at once. At least I have a working assembler I can use on disk for no extra cost. Patching EDTASM+ can take a back seat while I learn more about Newdos files.

Now on another subject.

Epson MX printer users have been discussing the problem of the omitted CHR$(0) and the problem of sending CHR$(11) and (12) to the printer. Some suggestions for overcoming the problem have been provided but the solution I am using may be of interest. An Eprom can be programmed and fitted between the printer output data lines and the printer to convert an unused code to the (0) and any other two unused codes to (11) and (12). I put CHR$(0) on (3) and (10) and (11) on (7) and (9). Most eproms are large enough to allow a number of 256 byte conversion tables to be stored. these can be alternative sets allowing the codes 0,11 and 12 to be put on different codes, switch selected so that if one needs the code (3) say, one can switch to a table which puts it on say (5). The only cost for

this mod. is an eprom, holder and switch. The eprom is installed in the printer on my friends system so that other computers can be connected and the printer responds in the way he is used to. Of course one of the tables can be a one for one in out copy reverting to the original codes.

In practice the 8 data lines of the printer port are taken to the lower 8 address lines of the eprom. The upper address lines go to a switch, a small dil switch in my case allowing the conversion table option required to be selected. The 8 data lines of the eprom go directly to the printer data inputs. The eprom can be permanently selected. Programming the eprom will present some with a problem, I could do a few if anyone was really desperate. The 2716 eprom I used will drive my printers through about a yard of miniature multicore cable but if longer cables are used its output may need buffering. It could be put in front of the existing interface buffer but this reduces the simplicity of the method a little. As described the eprom could be in a small box between the printer port and printer.

On one more subject...

I have just got the system program 'Trashman' from Prosoft in the States. This program reduces the 'garbage collection' hang-ups during string handling for which Microsoft BASIC is renowned. The demonstration program supplied shows really dramatic improvements in hang up times. Processing 1000 strings in the demo program takes about 180 seconds due to hang-ups whilst garbage collection occurs. With Trashman activated the time is reduced to 4 seconds. This is some improvement. The program occupies only about 570 bytes but requires a further 2 bytes for each string or string array it has to handle. The program is self locating and will load automatically below any other machine code programs already loaded. Memory protection is automatic.

One line has to be added to ones basic program to invoke Trashman as follows:

```
10  DEFUSR-578:X=USR(1000)
```

This line 10 calls the Trashman program at location -578 which is its auto-load location in a 48k system. (1000) would be the number used to allow Trashman to handle up to 500 strings variables or arrays. This program has reduced the run times of some of my BASIC programs by about 30 minutes and I have been keeping string handling to a minimum already to reduce the problem. Trashman seems to work OK with the ACCEL3 compiler but I have not yet had it long enough to know if this will always be the case. The program costs $39.00 and appears to be very good value for money if you suffer those string clearing delays.

My apologies for not giving details at do-it-youself level but probably not many will be affected by my particular problems or want to do the mods I have described. Anyway, I will discuss this further with anyone by phone or letter.

I hope all this has been of some interest.

PS The word processor is Word 4 word Plus.

Roy barber,
44 Dallin Road,
Bexleyheath,Kent.
DA6 8EJ. Phone:01 304 3856

EDIT

Over the past few months, I have been looking at various Basic screen editors. All have their advantages & disadvantages so I finally settled on EDIT as the best.

This utility, available on disk or tape, comes with an excellent little manual of 9 pages, neatly bound and easy to read. The manual starts with a description of the two versions, how to load & how to relocate for your particular memory size. The cassette version loaded without any problems, but it is wise to make a working copy first using COPSYS or similar SYSTEM tape copier. The program takes about 3.3k plus a variable workspace of about .5k

Once loaded, the program can be called by typing /EDIT & then hitting ENTER. The screen is cleared & a frame appears on the two sides & bottom of the screen, together with a column of 5 digit line numbers starting at 100 with leading zeroes. A flashing cursor is on the first line. A Basic program can now be entered as normal, errors corrected by overtyping (the cursor is moved by the four arrow keys) & line numbers can be changes to suit in the same way. Care is needed when the end of a line is reached as there is a slight delay while a space is added after the line number, the display is opened up & any PRINT's that have been typed as ? are now shown in full. This means that the text will spill over to the next line & the cursor will not be at the last typed character.

There are some 30 commands available. Some of these cover deleting & inserting characters, lines or blocks of lines. Single or blocks of lines can be moved or copied either before or after an existing line. Any line numbers that are now out of sequence are shown clearly for re-numbering.

Other useful commands are Find which enables you to enter a string (up to 8 characters + 2 delimiters) along the bottom edge of the display. Pressing ENTER will now cause a search for the string from the current cursor position, the screen scroll stopping when the string is found. Find can be continuously repeated by holding down @ & F. In conjunction with Find is Global change which allows replacement of the string defined with a new version. Split is used to break up long lines into short ones while Join is useful for uniting several short lines into one. BREAK is used as a special command key, which includes return to Basic.

There is also a lower case driver routine. Basic commands can be typed in this mode, they are converted to upper case when the line is entered unless within quotation marks.

Perhaps the one thing I find EDIT lacks is the provision of macro keys for Basic commands, which save a lot of typing. However, it should not be too difficult to incorporate this without increasing the memory size too much. A minor

criticsm is the bind of typing /EDIT each time the editor is needed!

On the whole, I can thoroughly recommend this utility, a little expensive at £17.50 inc., for the cassette version, but 'once used, always used'.

EDIT is available from Southern Software.

Pete Bamfield

USING C

One of the first things I did when I got the C/80 compiler mentioned in the last issue was to write a program to count the number of words in a text file. Here it is:-

```
#include "printf.h"
#define EOF -1
#define BLANK ' '
#define TAB '\t'
#define NEWLINE '\n'
#define YES 1
#define NO 0

main(){
        char c,inword;
        int nw;

        nw=0
        inword=NO;
        while((c=getchar())!=EOF){
                if(c==BLANK|| c==NEWLINE||c==TAB)
                        inword=NO;
                else if(inword==NO){
                        inword=YES;
                        ++nw;
                }
        }
        printf("no. of words = %d",nw);

}
```

Using the I/O re-direction feature of C programs I just have to type WORDCNT <FRED.DOC to count the number of words in the text file FRED.DOC.

I am currently using C to write a smart terminal program, and when that is finished I shall write a bulletin board program.

I find that C/80 works extremely well in conjunction with Macro-80, and I am gradually building up my own library of useful C functions, which do things like clear the screen, position the cursor anywhere on the screen (like PRINT@ in BASIC, but using X,Y co-ordinates), make BDOS calls to CP/M, and perform some rudimentary string handling. I have also written a very useful analogue of the BASIC INKEY function. This was needed to get unechoed keyboard input for my terminal program.

These functions are all contained in one relocatable object module. Using the Macro-80 Library Manager I then create a single library containing my functions, the C/80 printf function, and the C/80 run-time library. I only need to

link to the one library when I produce an absolute load module, which saves a bit of time. I actually use the CP/M SUBMIT facility to perform the tedious sequence of operations required to compile the source code, assemble the intermediate assembly language file, and link the resultant relocatable object module with the library. Using the C I/O re-direction feature the compiler error messages are placed in a file. This works because the C/80 compiler is in fact a C program.

I have done some interesting experiments with C/80, which is running on a Model II under CP/M. I have been able to compile C programs written for the Model I on the Model II, convert the object code into Intel Hex ASCII format using the /X switch when linking with L80, download the Intel Hex file to the Model I via the serial ports, convert it to binary using a Macro-80 program I have written, and load it into memory for testing and subsequent saving to Model I disk using the TRSDOS DUMP command. Of course, I had to write some C functions for the Model I I/O, as they are completely different from those for the Model II under CP/M. This process should work for other micros which use the Z80 processor, such as the Spectrum, Colour Genie, and Lynx.

Molimerx sell a C compiler (LC) that is used in conjunction with the EDAS editor/assembler on the Model I/III. This product also requires you to have LDOS. It looks rather over-priced at about £115 when compared with the C/80 compiler for £50, especially as it lacks things like multi-dimensional arrays, structures and unions. It does provide floating point, by accessing the ROM routines, however, and a much better run-time library is supplied.

I think I can recommend C to anyone who wants a change from that dreadful language BASIC. How about some other users of this interesting language writing about it. I know at least three of you are using it.

Leon Heller

FOR SALE

Green screen monitor £40; Single drive cable £5; Printer cable (Centronics 739) £7.50; Cassette software £5; Disk software £10; Joystick £5; "The Book" Vol. 1 £2.50.

Patrick Thomson
01-777 3552

DOTPRINT : A BRIEF REVIEW.

Last year I purchased GEAP, a wonderful little program that allows one to compose graphics, draw, rotate, magnify, or shrink pictures to be printed out on either an Epson or an OKI. I have an MX80 to which I have recently added the Graftrax type chips.

I was very interested at the Swindon meeting to see the latest version (2.1) of GEAP which incorporated a secondary program called **DOTPRINT**. Dotprint basically consists of a series of fonts, or typefaces, which a Graftrax-equipped Epson can reproduce. These font programs apparently can be purchased, or alternatively, the user can create his very own letterset! I saw around thirty variations, all of which greatly impressed me, since in my other (full-time) occupation as a Sales Manager, I use sheet upon sheet of Letraset to compile printers proofs for mailing shots; Dotprint will allow me to create, and re-create, documents which can go straight to the duplicator!

Apart from an endless variation in the number of fonts available, one also has the option of either, or both, magnification and reversal - as in the heading above. If one wishes to use these options, then the Dotprint program is used, but more advantage is gained by using another program called Dotwriter, from which a 'Newscript' format document is written which can access as many of these fonts as necessary:

The example following this one is Greek!

ΑΒΧΔΕΦΓΗΙΘΚΛΜΝΟΠΡΣΤΤΥΩΞΨΖ

REMEMBER THAT

THIS PRINT IS SHRUNK

When producing the Newsletter

Which is why I didn't use the Microprint Font!

There are many, many other styles available, and - as I said above - you can create your own. I don't know if the program is available in this country since Wordsworth, from whom I purchased my original GEAP, aren't advertising it. It is advertised in 80-Micro by JF Software, the publishers.

Happy Printing!
David Washford

FORTRAN tutorial PART 2

9.0 ANSWER to Part 1 Problem.

There now follows the answer to the problem set in 8.0 of part 1.I hope you got somewhere near it !!

```
C ********************************************
C *        THIS PROGRAMME  IS  THE  ANSWER    *
C *        TO THE FORTRAN TUTORIAL PROBLEM    *
C *        IN   NATGUG   NEWSLETTER FEB 83    *
C ********************************************
C
C  This part sets up the type of variables (see 3.0)
C
          INTEGER HEIGHT,LOOKS,CAR,STYPE,MONEY
          LOGICAL TALL,VGL,DL,RICH,BROKE
          LOGICAL ANYCAR,JAG,COLL,ELEC
C
C  This writes to the screen the prompt in between the ' '
C  in format label 2.(see 6.1 and 6.3)
C
          WRITE(5,2)
  2       FORMAT(' TYPE IN HEIGHT,LOOKS,MONEY,CAR & TYPE ')
C
C  This reads the variables HEIGHT,LOOKS,MONEY,CAR,STYPE
C  from the keyboard according to the format of label 3.
C  The format tells the program to read in 5 variables,
C  all of which are Integer type.Further explanation is
C  given in this tutorial.
C
          READ(5,3)HEIGHT,LOOKS,MONEY,CAR,STYPE
  3       FORMAT (I3,I2,I5,I1,I2)
          TALL=HEIGHT .GE. 6*12
C
C  This chunk sets up all the logical variables.
C  Due to the nature of the problem these may differ
C  slightly from your values.For example your definition
C  of Very Good Looks could be 8 out of 10 instead of my
C  9 out of 10.(see 4.0)
C
          VGL=LOOKS .GE. 9
          DL=LOOKS .GE. 5
          RICH = MONEY .GE. 30000
          BROKE = MONEY .LE. 0
          ANYCAR=CAR .GT. 0
          JAG=CAR .EQ. 3
          COLL=STYPE .GE. 2
          ELEC=STYPE .EQ. 3
C
C  The best way to decide who should date whom is by
C  logical variables.This chunk does the rest.(see 5.0)
C
          IF(TALL .AND. DL
     +      .OR. RICH .AND. JAG
     +      .OR. COLL .AND. VGL) WRITE(5,5)
  5       FORMAT(' MAY DATE MARY')
  20          IF (DL .AND. ANYCAR
     +           .OR. .NOT. BROKE .AND. VGL
     +           .OR. ELEC) WRITE(5,7)
  7           FORMAT(' MAY DATE JANE')
          END
```

FORTRAN tutorial PART 2

10.0 The FORMAT statement

The first part of the tutorial introduced the FORMAT statement on a very limited scale. In fact only Integers could be read in or written out. This clearly is inadequate for many situations and so there are many more Format descriptors. The most commonly used ones will now be described while the rest will be left for you to look up at some future date. The general form of a format descriptor is Descriptor w or descriptor w.d ,where w stands for the width of the field and d the number of decimal places. In all cases a blank will be represented by a _. In the examples external is what is read in or written out while internal is what is actually in the computer.

10.1 Iw (INTEGER)

This is an easy one as we have come across it before. It means read in or write out an integer of width w. Blanks being ignored on reading and output on writing in any unused positions. For example I6

External	=>	Internal	¦	Internal	=>	External
__1042		1042	¦	3546		__3546
___-27		-27	¦	-86		___-86
-27___		-27	¦	-27000		-27000
__+__6		6	¦	6		_____6
-3__4_		-34	¦	-340404		******
______		0	¦	3.14		illegal
_-3+44		illegal	¦			
__3.14		illegal	¦			

10.2 Fw.d (REAL)

This is the first new format descriptor and has a catch to beware of. It means that the input or output should be treated as a real number of width w and to an accuracy of d decimal places. This means that F3.1 describes a field of the form _._ and NOT ___._ .The example is for F6.2

External	=>	Internal	¦	Internal	=>	External
__3.14		3.14	¦	3.14		__3.14
_-2.79		-2.79	¦	-2.79		_-2.79
+_90.1		90.1	¦	90.1		_90.10
0.1234		0.1234	¦	49.2**-3		__0.05
___372		3.72	¦	49.2**-6		__0.00
__372_		3.72	¦	482.143		482.14
9001+1		900.1	¦	55.2**6		******
2.1E-3		0.0021	¦	3927		illegal
_____5		.05	¦			

10.3 Lw (LOGICAL)

This is a really simple one.As its name suggests this is for Logical variables.The examples will be using L6.

External => Internal		Internal => External
__T____	.TRUE.	.TRUE. _____T
TANDY__	.TRUE.	.FALSE. _____F
_____F_	.FALSE.	
F_____	.FALSE.	
_TRUE__	.TRUE.	
FALSE_	.FALSE.	
.TRUE._	.TRUE.	
.FALSE._	.FALSE.	

10.4 nX (SPACING)

This is the last descriptor I am going to deal with and it simply adds n spaces to the output or input.For example:-
FORMAT (I3,4X,I3)
Used as an output Format would add 4 spaces between the two integers output.Thus the output would be e.g 123____678.It can be used as an input Format however it is not recommended. It can be done though, e.g. If 1234567890 would result in the first I3 being 123 and the last I3 being 890.

10.5 Repeating Formats.

By now you are probably thinking what a bore to type in loads of F6.2's and I3's.You are quite correct and in this version of FORTRAN their is no way round it.However repeating Format descriptors go some way to help by allowing you to repeat the same descriptor n times.For example 2I3 is the same as I3,I3 and 2F6.2 is the same as F6.2,F6.2.That's all there is too it!!

11.0 SUBROUTINES

If you have programmed in another computer language then you probably know that a subroutine is one of the most useful parts of a computer language.Unfortunately TANDY BASIC only allows you to GOSUB to a line number,the subroutine being a part of the main programme.FORTRAN allows a more powerful version and it is this that has helped keep the language flourishing today.

A subroutine in FORTRAN is a separate programme.That is it has its own declarations,statement labels,END statement etc.They are kept entirely separate from the main programme unit.Even if your main programme and your subroutine both have label numbers of 100,they won't be confused.

Let's look at the way the main programme communicates with the subroutine.That is how the main programme CALL's the subroutine.Naturally this is via the CALL instruction.For example:-
CALL NEWPGE(PGENUM)
Where the CALL is the keyword telling the programme what to do.NEWPGE is the name of the subroutine to be called and PGENUM is the arguement to be passed.This is fairly straight - forward

FORTRAN tutorial PART 2

in this example so let's look at the subroutine which is denoted by the keyword SUBROUTINE. For example:-

```
SUBROUTINE NEWPGE(PAGE)
```

Where the SUBROUTINE is the keyword telling the programme what to do. NEWPGE is the name of the subroutine, notice the correspondance with the NEWPGE in the CALL statement. Lastly PAGE; this is the formal parameter passed from the CALL. This need not be the same as the one in the call as it is only the value you are passing across. The concept I usually use is the umbilical cord between a mother and her child. The CALL being the mother, the SUBROUTINE the child with the arguement at one end of the umbilical cord and the formal parameter at the other. This reinforces the view that the main programme and the subroutine are entirely separate programmes.

Thus a typical programme using subroutines will look like this:-

```
C   Main programme
      ..................
      ..................
      CALL ROTATE(FLAG,NUMBER)
      ..................
      ..................
      END
C
C   Subroutine programme
      SUBROUTINE ROTATE(METHOD,NUM)
      ..................
      ..................
      END
```

This means that the values held by the variables FLAG and NUMBER in the main programme unit are passed to the subroutine as the variables METHOD and NUM. It must be emphasised that the variable TYPES MUST BE THE SAME. Thus for example FLAG must be declared an integer in the main programme unit or METHOD must be declared a real in the subroutine.

Since SUBROUTINEs are completely separate program units they can cause all manner of subtle errors. It would take a considerable amount of space describing the trials and tribulations possible. In fact an article as long as this one could easily be written on the pitfalls of CALLS and SUBROUTINES. It is even more frustrating that most FORTRAN compilers usually don't pick up these mistakes either. MORAL :- If you have a programme involving CALLs and SUBROUTINES and is not working look at these bits first.

Jeffrey Goldberg
458-3025

Building someone elses Maplin modem

The Maplin modem was ordered by David Washford in the dim and distant past and arrived after many 'phone calls and finally a letter to Maplin. David had waited about six weeks to get his hands on the beast and called me one Sunday evening to tell me that it had arrived. Later that evening another call revealed that the kit used close tolerance resistors and such were coded in a standard but not common 5 band colour code. The kit arrived on my doorstep not an hour later with the question, Can you identify these?...About a dozen resistors left over from the bulk that were neatly labelled on a piece of paper. David left me with the kit to identify the remaining components. (A call the next day put me in the position of component identifier and kit constructor).

A first comment must be that some of the resistors did certainly look odd with their 5 band colour codes on a light blue background. Having identified the remaining resistors with the occational help of a multimemter I was just about ready to start construction. I did find that a set of ten 10K resistors had be supplied as 6.8K but that was the only error I encountered during construction. Construction is straight forward as the two printed circuit boards supplied have screen printed legends on them making component loaction simple.

Tools required for construction are at small soldering iron (15-25 Watts), a pair of cutters and small pliers. Reading the article I noted that they recommend the use of sockets for the integrated circuits - these were NOT supplied and examining the I.C.s used, two are priced at £7.75 & £5.75 so use of sockets on these two I would recommend. The more expensive device is a 14412, a member of the CMOS family and as such is sensitive to static damage so caution should be exercised when handling this particular device. The remaining I.C.s should not be 'thrown' around but are a little more hardy. The use of sockets on the 8 pin devices (741s & similar) makes life easier, unless you are very good at desoldering!.
The 1MHz crystal has fairly stiff leads and care should be exercised when bending the leads to fit it to the board. It is a good idea to glue it to the board. to prevent it waving in the breeze!

Once all components have been fitted to the board it is always advisable to check for solder splashs and dry joints as with any electronic construction project. Holding the board up to a strong light is a quick and easy way of detecting any solder splashs. Following the diagram supplied I wired up the connector that plugs into the modem for connection to power supply and telephone line. The kit does not include any sort of enclosure for the modem so with the LEDs wired directly to the connector, a birds nest of wires connecting the power supply, Full-Test-Half Duplex switch and Originate/Answer switch together with the line isolating transformer I applied power to the system...WITHOUT the modem board plugged into the connector. I checked the power supply voltages, +5,-12,+12 were all present and correct. Once satified that the wiring was O.K. I switched off and plugged the board in, without the chips plugged in.(shows you how useful sockets are!). If the pin supply voltages are ok, then switch off, wait 5 minutes, plug the chips in, and repeat the test.With all connected I switched back on...no smoke', so with a pair of Hi-Fi 600 ohm headphones connected across the output line I listened - silence, looking back at the circuit it was immediately obvious that I wasn't going to hear anything. The GPO line has 50 Volts on it normally and this was being used to power a LED indicating that the modem was holding the line. The circuit uses a bridge of diodes to give correct polarity to the LED and with no 50V present, they were not conducting hence I heard nothing. Applying the headphones to the isolating transformer got more results, a single note was heard which altered when switching from Originate to Answer.

So with the main construction over I looked back at the text for

details on aligning the modem. Reading them through revealed nothing to tricky but the use of an Oscilloscope is needed. Do not let this put anyone off because local groups nearly always have someone who has such a device or at least can get their hands on one when the time comes. One error in the text is obvious when you sit on the emitter of TR1 expecting to see a sinewave, they mean TR2 and not TR1. Having followed the setting procedure in the text, I was almost ready to go off into the local GPO network to find someone to try this new toy on. One problem I did encounter with the system is that most communications software checks the RS232 card for silly little things like CD (carrier detect) and other signals which it is expecting to read. The connection of this modem to the computer is a three wire job (initially), Transmit data, Earth and Receive data. So to use the Tandy communication package and similar, you need to do something with those lines. A quick read of the Tandy RS232 manual revieled that by connecting CD,DSR & CTS to Vcc. The software was well and truly convinced that all was well. CD - Carrier Detect I will take from the phase locked loop lock output (i.e. the chip says - all Ok to receive). Since we have a Carrier Lock LED on the modem we can use this point to tell the RS232 card when things are all locked up and ready to go (See diagram). CTS - Clear to Send is much the same, the software will wait until this line goes high before it will send anything so by pulling CTS to Vcc the RS232 will speak to the modem. DSR - Data Set Ready must also sit at Vcc to enable data to flow between the RS232 card and modem.

With all three lines tied to Vcc I tried to 'phone Distel - the Display Electronics data base available free to anyone with the right equipement. No reply...A nice girl in 'sales' told me that their computer had blown a fuse and that it may be back within a week or so!. Next I tried REWTEL, run by Ambit International again certain parts are free. The number just wrang & wrang. After getting the right number - yes it had changed I got a tone on the end of the line.... I connected and bingo, my first introduction to expensive phone bills! The introduction below was transmitted to me at a nice reading pace:

```
WELCOME TO EXTERNAL REWTEL.

PLEASE TYPE IN YOUR SUBSCRIPTION NUMBER >>

PUBLIC SESSION

TYPE HELP <RETURN> FOR HELP AND
TYPE QUIT <RETURN> WHEN YOU HAVE FINISHED.
TYPE <RETURN> AT THE END OF EACH PAGE
BEFORE ENTERING NEW KEYWORDS, OR ANYTIME TO ABORT APGE

**** REWTEL 24hrs a day from 29th Nov ****
**** TELEPHONE NUMBER  (0277) 232628 ****

WRE Spring catalogue out soon; on REWTEL soon

Subscription Services required
for REWSHOP and BINGO facilities, but
you can always leave a BULLETIN.

Public session timeout after 8 minutes

13.49 REWTEL >>
```

And away you go into the massive data bank available.

A few points to note when buuilding & testing the modem are that that parts list describes one R36a as a 470 Ohm resistor - this is correct but

no R36a exists on the board, it should read just R36. If as mentioned earlier, the status lines CD, DSR & CTS are tied to Vcc the RS232C will receive and transmit to the modem even when nothing is connected to it because it has been told that a carrier is being detected (CD), it is Clear To Send (CTS) and data can be received (DSR) and the outcome of this is that occational rubbish will be printed on the screen if you are running a terminal program. One simple solution is to use the carrier lock LED to tell the RS232C card when a carrier is present. Looking at the circuit, the output of IC8 goes low to bring the carrier lock LED on, this signal is then inverted via IC10a. A connection to IC10a output can be taken to the CD line on the RS232C card (ie pin 8 of the RS232 connector) . The following table is a list of pin connections on the 25 way D-type connector as supplied with the Tandy RS232C card:

MODEM SIGNALS	25 WAY D-CONNECTOR PIN NO.
RS232 OUT (10)	3 - RD - Receive Data
RS232 IN (17)	2 - TD - Transmit Data
IC10a output **	8 - CD - Carrier Detect
+12V (PSU 5)	6 - DSR - Data Set Ready
+12V "	5 - CTS - Clear To Send
0V (PSU 8)	1 - PGND - Protective Ground
0V "	7 - SGND - Signal Ground

** - The pin numbers on IC10 do not correspond with the circuit diagram - The following table gives corrections to the circuit diagram:

IC10

Old Pin Nos.	New Pin Nos.
1	5
2	6
5	1
6	2

Should you not have access to an oscilloscope for alignment of the modem, you can use the self test mode (middle position of S2) to adjust the device:

Connect the modem to the computer and using any terminal program (preferably with a repeat key!) adjust RV1 and RV2 so that the modem echoes back the characters typed. Start in originate mode and adjust RV1 for correct echo, then switching to answer adjust RV2 for correct echo. This will allow you to use the modem although at some later stage correct alighment with the aid of an oscilloscope is recommended. RV3 may be left at mid tarvel as it seems not to have any 'great' effect on performance.

I hope that these comments will be of use to anyone considering buying one of these modem kits or to the constructor who may be having difficulties.

A list of Maplin Modem ownrers compiled at the Swindon Workshop is given below and we hope that Bulletin board operators may allow a Maplin Modem users group corner within their systems to allow owners to exchange ideas and to promote the use of these devices between users directly and not always via a Bulletin Board system.

David Washford
1 Alexandra Rd
Bournemouth
BH6 5JA

Mac Randle
17 Langdale Close
Leamington Spa
Warickshire CV32 7QB

Don Bannister
41 Myrtlefield Pk.
Belfast
BT9 5JA

0202 423064

Pete Smith
7 Wootton Cres.
St. Annes Pk.
Bristol 4

0926 39871

Dave Sampson
4 The Coots
Stockwood
Bristol 4

0232 668899

Nic Robinson
The Residency
Northwick Park Hospital
Harrow Middx.
01 864 5311 Ex 2588

Carl Rabe
40 Felton Road
Parkstone
Poole Dorset
0202 730617

Jim Roden(has built autoanswer unit)
36 Ribbesford Drive
Stourport
Worc's

Please excuse any errors here as most of the addrs. were very roughly scribbled onto a scrap of paper. Nic Robinson informs me that both John Newgas(01 348 6518, ringback modem)and Jim Roden have recently placed a Maplin Modem users corner on their bulletin board services so any comments, difficulties, and/or ideas about improving the modem could be left here for the benefit of all.

Carl Rabe (TOPIC)

LISP/80

I was so pleased with the C/80 compiler from The Software Toolworks that I ordered their LISP/80 interpreter. Here are some preliminary notes on it.

LISP/80 is written in C, and comes on a standard 8" IBM format SS/SD disk.

Most of the standard LISP data structures and functions are provided. 16 bit arithmetic is supported. There are the usual string functions such as PACK and UNPACK.

File I/O is available, and functions can be loaded from disk as required. The total LISP environment cannot be saved, however, so the language cannot gradually evolve to your own requirements. You have to load the new functions each time.

A simple editor is provided (written in LISP) and there are some useful debugging aids such as a trace facility, and optional break on errors. Machine-language functions can be loaded and called from LISP programs, so it would be possible to add things like floating point, with a lot of work.

A couple of example programs are provided; ANIMAL which tries to guess the animal that the user is thinking about, and a simple version of ELIZA, the simulated Rogerian psycho-therapist that Weizenbaum originally developed (and then wished that he hadn't).

LISP/80 seems very close to MACLISP and INTERLISP, and most of the programs in Winston and Horn's classic book, LISP, would probably work, with the exception of those using floating point.

It is rather slow, however, so it is not suitable for serious work. In fact, 8-bit micros are not really suitable for LISP anyway. Ideally, you should have a 32-bit machine, but a 68000 implementation wouldn't be too bad.

Anyway, LISP/80 is a good way to learn the language, and it only costs £42.

C/80 and LISP/80 can be obtained from:

System Science,
54 Enfield Cloisters,
Fanshaw Street,
London N1.
Tel: 01-739 0540

Leon Heller

Comtd. from p. 12

Note the use of the switch case C construct. This is like the case statement in Pascal, but is a lot nicer. Also note the use of in-line Z80 code where I have to interact directly with the Model II hardware. Most of this code could in fact have been written in C by using assembler input() and output(c) functions.

```
£include "printf.h"
£define ACMD 0F6H
£define ASTAT 0F6H
£define ADATA 0F4H
£define EOF -1
£define CR 13
£define ESC 27

main(){
        char c;
        cls();
        while(1){
                c=kbrd();
                if (c==ESC) cmd();
                else{
                        if (c!=0) serout(c);
                        c=serin();
                        if (c!=0) dsply(c);
                }
        }
}
```

```
cmd(){
        char c;
        c=getkey();
        switch (c) {
                case 'q':quit();
                break;
                case 'Q':quit();
                break;
                case 'o':open();
                break;
                case 'O':open();
                break;
                case 'c':close();
                break;
                case 'C':close();
                break;
                case 's':save();
                break;
                case 'S':save();
                break;
                case 't':xmit();
                break;
                case 'T':xmit();
                break;
                default:return;
                break;
        }
}

quit(){
        cls();
        exit();
}

open(){
}

close(){
}

save(){
}

xmit(){
        char c, fname[12];
        int chan;
        printf("\n<TRANSMIT FILE>");
        printf("\nFILENAME: ");
        gets(fname);
        chan=fopen(fname,"r");
        if (chan==0){
                printf("<OPEN ERROR>\n");
                return;
        }
        while (c!=EOF){
```

```
                c=getc(chan);
                if (c=='\n') serout (CR);
                else serout(c);
        }
        fclose(chan);
}

kbrd(){
        return (bdos(255,6));
}

getkey(){
        char c;
        do c=kbrd();
         while (c==0)
        return (c);
}

dsply(c) char c;{
        bdos(c,2);
}

bdos(DEparm,Cparm) int DEparm,Cparm;{
fasm
        .Z80
        POP     HL
        POP     BC
        POP     DE
        PUSH    DE
        PUSH    BC
        PUSH    HL
        CALL    5
        LD      L,A
        LD      H,0
        .8080
fendasm
}
serout(c) char c;{
        c;
fasm
        .Z80
S3:     XOR     A
        OUT     (ACMD),A
        IN      A,(ASTAT)
        BIT     2,A
        JR      Z,S3
        LD      A,L
        AND     7FH
        OUT     (ADATA),A
        .8080
fendasm
}
serin(){
fasm
```

```
          )Z80
S1:       XOR     A
          OUT     (ACMD),A
          IN      A,(ASTAT)
          BIT     0,A
          JR      NZ,S2
          XOR     A
          JR      S4
S2:       IN      A,(ADATA)
          AND     7FH
S4:       LD      L,A
          .8080
fendasm
}
```

Leon Heller

STOP PRESS

In last week's Computing it was reported that Onoly Enterprises Inc. are negotiating with Tandy Corp. for the purchase of 65,536 Model I 16K CPUs. They intend to connect the Model Is into a giant multi-processor system, with every processor connected to every other processor. This will give them a system about 10 orders of magnitude more powerful than the Cray I. In fact, it will use four Cray Is as front end processors. The system will be used for Artifial Intelligence research, and it is hoped that within 10 years it will have accumulated the entire corpus of knowledge of the human race. It will then be used as a super intelligent Expert System, able to answer questions on topics in every field of philosophy, science and technology, and medecine, such as the origin of the universe, the achievement of controlled nuclear fusion, a cure for the common cold, etc.

J. Torofex.

LOWER CASE FOR THE 779 PRINTER

One of the main drawbacks of the Tandy model 1 and Centronics 779 printer is its lack of lower case characters. In the past only three options have been available to overcome this problem: pay 95 dollars and send to America for a ready built unit, copy the article in 80 US and build it yourself costing about 70 pounds or sell the printer and buy a new one. None of these options appealed to me so I designed my own cheaper and more versatile unit to produce lower case.

I am willing to market this product but would like to know what form is preferable, from circuit diagrams and character generator to ready built unit and what sort of demand there would be. If sufficient demand exists I feel that a price of as low as 20 pounds could be achieved.

Please contact me with your views in the near future on the following number 06845-67494 (Malvern)

Paul Everington

CDDS AND ENDS

Milton Keynes Weekend will see Peter Tootill from Liverpool, and Alan Redhouse from York coming for the first time, Alan on LDOS and using mixed discdrives on the Model one, and Peter on DOSPLUS.

Laurie Shields and Leo Knaggs will be doing NEWDOS/80 with John Newgas on ULTRADOS. Brian Pain will do a proper session on CORPLAN, which will last around six hours, and will need 4 systems with 3/4 persons to each system.

The booking form and full programme will be sent out in the May issue. The date will be the last weekend in JULY.

During the summer holidays Brian Pain & Leon Heller will be found at MKMUG on Tuesday evenings, which is the local club in Milton Keynes. The meeting place is a factory unit laid out for micros, and members as a rule bring their systems.

John Silver on 0273-501901 has just acquired discs, and seeks help from those around Brighton area.

Geoff Henshall from Milton Keynes rang to advise "80-MICRO" is now available from major newsagents. I bought March in Euston the other day.

Richard Marks on 01-609-1340 has got an Olivetti Praxis 30 and is delighted, and solved a tricky out of paper problem. Richard can help over the phone.

Andy Cammack has moved to Plymouth, and can be reached on P 706935.

Mike Orr on 0908-612766 is interested in TRS - OSBOURNE transfers.

Luton College have their Tandy room now live, and welcome members to drop in. 0582 - 34111 Dept of Accounting AND Finance.

John Imam would like a copy of an article on TBUG for II 80-MICRO ARRIL 80 01-309-1486

Coming next month a new simple timetabling program using the ND80 SORT & RUN,V facility. It works but lecturing has come before getting it ready for the newsletter. BCP.

FOR SALE

Model I kekboard, Expansion interface SK GP8 printer,cassettes & manuals.

£500 ono Mike Russell 0908-604398.

As I told you my Model III is now fitted with disk drives and upgraded to 48K.I was under the impression that the cassette version of Scripsit and Edtasm could be made to run on the disk with some help.If you know of any suggestions please let me know. Another problem is that I am unable to produce different typesetting on my Epson 82FT,the type I see in your monthly newsletter.Could you suggest anything?

DR.C.F.A.J.SALDANHA
131 Thornbridge Avenue,
great Barr,
Birmingham, B42 2AP.

NATIONAL
TRS-80 & GENIE
USER GROUP
Newsletter

VOLUME 4 ISSUE 11 MAY 1983

CONTENTS

2. Information on the Group
3. Editorial
3. Guidelines for contributors
4. Problem section
9. 80-MICRO
12. To see or not to see...
12. Scripsit and Visicalc
14. In-line printer codes with Scripsit
18. Editing successive program lines
19. Review of GNOMIC's DP1000 disk and printer unit
20. PRESTEL via LDOS and acoustic coupler
29. Deflux your RAMs
30. NATGUG Swindon '83

NATGUG

INFORMATION ON THE GROUP

Membership of the Group is by subscription to the Newsletter, which is published monthly. Membership details are obtainable from the Group Secretary. Membership of the Group is open to anyone with an interest in the TRS-80 range of microcomputers, and compatible systems such as the Video Genie.

Details of the Group accounts and the constitution of the Group are obtainable from the Secretary.

Members requiring assistance with problems related to the TRS-80/Video Genie may call the Secretary. An attempt will be made to put them in touch with a member who can help with the problem.

Workshops are arranged from time to time in various parts of the country.

Sub-groups exist in many areas. A list is provided at the back of the Newsletter from time to time. Sub-groups also exist for medical and educational users.

The Group runs 80-NET, a computerised bulletin board service that is available to all users of microcomputers.

The Group maintains three software libraries (Models I, II and III) which are free to members. Library lists are obtainable from the Secretary. The medical and educational sub-groups maintain their own libraries.

A membership list is obtainable on disk from the Secretary.

Back numbers of the Newsletter are available.

Please send all contributions for the Newsletter to the Editor.

Chairman and Newsletter Editor:-

Leon Heller,
8 Morris Walk,
Newport Pagnell,
Bucks. MK16 8QD.
Tel: (0908) 613004

Secretary and Newsletter Publisher:-

Brian Pain,
24 Oxford Street,
Stony Stratford,
Milton Keynes.
Tel: (0908) 564271

Librarians:-

Model I: Leighton Davies, 105 Caerau Road, Caerau, Glamorgan. Tel: (0656) 738337.

Model II: Jim Hutton, 25 High Street, Stroud, Glos. Tel: (045 36) 4423.

Model III: Nick Clay, 6 Rectory Lane, Byfleet, Surrey. KT 14 7LL. Tel: (09323) 42348.

EDITORIAL

Once again, I have had too much for this issue, and have had to hold over a lengthy article for publication next month.

Although I finished the April issue on the 17th, I did not get it to Brian for a couple of days. I think the weekend intervened, holding up the printing, and we always use second class postage. Hence the delay in reaching you, which a couple of members have remarked upon.

We intend to use a booklet format (A5), starting with the first issue of Volume 5, in July, as our printer has now got a machine that collates and stitches the sheets. This should make the Newsletter much easier to handle, and will not involve me in any additional work, as the pasting-up technique will be the same. We should save some money on the envelopes, into the bargain.

The London Computer Fair will be upon us within a few weeks. We will have a stand, thanks to the ACC, and volunteers will be required to man it. Please let Brian know if you can help. Of course, such volunteers will get free passes to the show!

Leon Heller

GUIDELINES FOR CONTRIBUTORS

1. If possible, send material in printed form.
2. Scripsit or Pencil disks/cassettes are acceptable.
3. Ensure your ribbon is in reasonable condition.
4. Printer output should be on A4 paper if possible.
5. If you send in handwritten material, write legibly.
6. Do not fold the sheets when posting them. Use a large envelope and keep them flat.

Leon Heller

FORTHCOMING WORKSHOPS

MILTON KEYNES WEEKEND
July 29,30,31

Theme :- DOS
Speakers Alan Redhouse, Peter Tootill, Laurie Shields & Leo Knaggs etc
Location Stony Stratford, Milton Keynes.

SWINDON WEEKEND
November 18,19,20

Theme :- suggestions invited

LONDON COMPUTER FAIR at EARLS COURT
June 16,17,18,19. Thurs Leon Heller
Fri Brian Pain
Sat & Sun London Members

PROBLEM SECTION

My present word processor program is cassette based TEXPRO from Harding. It is slow loading and not compatible with my disk based operating system which is DOSPLUS or TRSDOS. To use it I have to boot up with BREAK to dislocate the disk drive. I am about to get a disk based word processor and am considering NEWSCRIPT which works with DOSPLUS. Any advise about a word process program on disk compatible with DOSPLUS from you or any member would be appreciated.

Eric Brandes.

46 Field Way, Chalfont St Peter, BUCKS. SL9 9SH
Tel:Gerrards Cross(0753)882982

I RECENTLY PURCHASED A GENIE SYSTEM AND FIND IT A LITTLE HARD TO UNDERSTAND .IT HAS TWIN DISCS WITH AN EPSON PRINTER. I HAVE BADGERED THE TANDY COMPUTER CENTRE LOWE'S AND THE DEALER AND NOW HAVE A LOWER CASE IN PLACE BUT NO METHOD OF TRIGGERING IT OTHER THAN TANDYMAIL 1 AND WORD 4 WORD. THE CASSETTE RECORDER IS WORKING I AM ASSURED BUT I WOULD LIKE TO GET THE PROGRAM ONTO DISC. ANY SUGGESTIONS WOULD BE WELCOME. THERE IS A PROGRAM IN THE NEWDOS+ WHICH ASKS FOR INFORMATION BEYOND MY CURRENT KNOWLEDGE.

DONALD M HARTLEY
35 DACRES ROAD
FOREST HILL
LONDON . S.E. 23. 2NS

Your editorial in the April issue said that you had finished it (bar the printing and collating, I assume) on 17th April, and I received it this morning (7th May). I couldn't read the postmark, so should I have a word with my friendly neighbourhood postman, or just sympathise with the fact that it took a little while to getting round to what must be the worst job of the lot - stapling it all up?

The foregoing is not really the reason for my putting pen to paper - (or, more precisely, word processor to buffer). I was prompted by John Kilpatrick's letter in the last issue. He said, quite rightly, if you want to know, ask. I have spent many many happy hours inventing the wheel, which is not such a bad idea: most of the entertainment my computer provides me with is the stimulation of finding ways to do things. However, when completely stumped, a lecture or a letter in this magazine has suddenly either provided the answer, or given me enough information to figure it out for myself. (I do read books, but they are either an ABC of Dartford Basic or are written in Chinese - well, at least, seem like that!) I've reached a point where I need to know something new, so, if anyone can help, I would appreciate it.

I should like to know how to find, in memory, a particular line of a BASIC program, and, having found it, how to change it, using, I suppose, POKE. In other words, I want a BASIC program to change itself while running.

Why on Earth, (I hear you cry), should anyone want to do that? About a year ago, at one of the workshops, Marc Leduc gave a very entertaining and supremely useful lecture on string packing. Since most of my programs use graphics in differing degrees of complexity, I have since spent many tedious hours working out the numbers to put into data statements to draw the graphics. Invariably I get a few wrong, and spend ages trying to find the bugs, and sort them out. In due course, it came to me that it must be possible to write a program which enables you to draw the graphics on the screen, and having done that, print out the numbers required for the data statement. Several happy man-hours later, I had that, but it was still imperfect, because I had to copy those numbers out again. I then extended the program so that it put graphics from A$(1) to A$(9) in lines 20-100 and then deleted everything except those lines. (Line 10 contains a memory setting routine, if necessary, a CLEAR statement, and various DIM & DEF statements, as required)

The program works beautifully, except the slight inelegance of ending with SYNTAX ERROR IN LINE 650 instead of the usual READY. (Line 650 no longer exists as it is the line which causes the deletion of everything except 10 - 100. However, I frequently don't want as many as nine graphics, and I would like the program to work out and put into line 10 the amount of memory to CLEAR, the number of A$'s to dimension, and to change the DELETE comand so that it deletes the lines up to 100 which no longer have any function.

Obviously, these refinements are not vital, as I've already taken out much of the head-banging in graphics, but it will satisfy my aesthetic sense if the program does everything. I should also like to know how to prevent that SYNTAX ERROR ending. Incidentally, I can only make it work with the NEWDOS-80 command CMD"F",DELETE n1-n2. Is it possible to use more than one DELETE command in a program with TRSDOS or LDOS or even without discs? (More than one because DELETE 1 - 9 and DELETE 101 - final line are needed).

Of course, the string packing routine itself changes the program while it is running, and I have tried to extend the changes it will make into the changes I want it to make, without success. It's entirely probable that when somebody points out how to do it, I shall wonder why I hadn't worked it out for myself - that happens a lot with me! I have a tendency to overlook the obvious!

If and when I get the program running to my complete satisfaction, I'll send a listing to the magazine or a copy to the library or both.

Question number two may be simpler, particularly if anyone else out there uses AJEDIT. I'm very happy with it, in general, but, every now and again it will suddenly spread everything out all over the screen, putting about four words on each line. The only way I have found of solving this irritation is to save the text to disc, then reboot and load it again. Is there an alternative? >
And why did that > suddenly appear?

And, finally, when I bought my printer, I made the mistake of buying at the Tandy shop. Not that there's anything wrong with the printer. It's beautiful. It's a Line Printer VIII. It's just that (a) it cost a lot, and (b) in spite of rereading the instruction book about fifteen times, I can't make it print graphics, which it is supposed to do. Many times, I have tried putting it in the graphics mode, as per the instructions, and then using JKL. Standard characters it will print out quite happily, but when it comes to a graphics character it prints
I. Can anyone help?

Paul Willey

I have just come across two problems and shall be grateful if you could cast some light on them.

The first one relates to Command O. I wish to do an indirect Sort and as a test I entered the two lines shown on the enclosed printout and then ran the program with the results that you can see displayed. It changed the order of the strings but did not put them into alphabetical order or anything approaching it. I have no doubt that I am doing something stupid somewhere, but at the moment I have been unable to see it and Os. House has been unable to help also.

The other problem is that I tried running one program out of another, retaining the variables, by the addition of ,V on the end of the command and lost the variables. I have a feeling that this must be due to the fact that I had to clear a string area for the program and presumably this cleared the variables at the same time. If so I would be glad to know whether there is any way I can get round it.

I would be most grateful for any help you can give me with either of these problems.

M.C. Matthews.

5A West Walks
Dorchester
Dorset

```
2010 M=10:A$(5)="Aaaaaaaaa":A$(7)="Ddddddddd":A$(3)="Wwwwwwwww":A$(4)="Uuuuuuuuu":A$(1)="Mmmmmmmmm":A$(6)="Sssssssss":A$(2)="Ttttttttt":A$(8)="Mmmmmmmmm":A$(9)="Fffffffff":A$(10)="Try this":FORI=1TOM:R3%(I)=I:NEXT
2020 CMD"O",M,*R3%(1),A$(1)(1,1):FORI=1TOM:PRINTA$(I),R3%(I):NEXT:STOP
```

```
>list2010-2020
READY
>run2010
Mmmmmmmmm        5
Ttttttttt        7
Wwwwwwwww        9
Uuuuuuuuu        1
Aaaaaaaaa        8
Sssssssss        6
Ddddddddd        2
Mmmmmmmmm       10
Fffffffff        4
Try this         3
Break in 2020
READY
>_
```

```
Break in 2020
READY
>cont
Uuuuuuuuu        1
Ddddddddd        2
Try this         3
Fffffffff        4
Mmmmmmmmm        5
Sssssssss        6
Ttttttttt        7
Aaaaaaaaa        8
Wwwwwwwww        9
Mmmmmmmmm       10
Break in 4999
READY
```

For Sale.
2 Orbis 8 inch disc drives.(uncased)
24 volt power supplies in situ.
5 volt dc supply.
Manual with circuit diagrams.
2 50 Pin sockets on short lead.
9 discs to suit.
The lot for 300 quid.
Reason for sale.Not enough technical know how to instal with my TRS 80.
Phone 0604 710391 HARVEY FRUISH — NORTHAMPTON

(* Can anyone help with the first problem?

Regarding the second problem. As you say, CLEAR zeroes the variables. Why not just use CLEAR in the main program, not in the program you are chaining. You should be able to re-arrange things if you do not have enough memory in the main program. Another technique would be to write all the variables to a temporary disk file, then pick them up in the chained program - rather slow and inelegant, but it should work. The latter would work with ordinary TRSDOS BASIC which does not support program chaining. LFH *)

I have an ICL terminprinter 7075, some circuit (logic) diagrams, together with a small extract from the service manual (now out of print).

I cannot get the wretched, but beautiful, machine to even work off-line, and would dearly like to employ it for letter quality printing on-line, once going. I need help from somebody with knowledge of this model, or a boffin who can tinker, when armed with a bit of documentation. Any offers of help, pretty please?

I also have a Diablo 530 hard disk, which appears to be mechanically and electrically sound, and complete with DEC compatible terminator box and PSU. Computer Warehouse still sell these and assure me that their clients have achieved this aim, but alas could not tell me how to do it, nor who has achieved it. Would any of our clan have the know-how, perhaps?

I have recently acquired, legitimately, a Post Office telephone coin box. It struck me that this robust piece of engineering has a three way peripheral potential:-

(a) fund-raising events by permitting games entry to the computer by depositing coins of the realm appropriately

(b) telephone dial allows choice of ten programs (users need not necessarily touch the precious keyboard!)

(C) the handset could be linked to voice synthesiser for instructions

Any ideas or project development notes from my colleagues, please?

Finally, thank you for providing a super monthly dose of 'Tandy Talk' to help the lesser mortal.

Steve Broderick,
5 The Croft,
Marsh Baldon,
Oxfordshire OX9 9LN.
Tel: Nuneham (086 738) 359

I have a problem which I hope either you or some member of NATGUG can help me with. I am trying to write a Software Package for my TRS80 Model 1. I want to be able to 'BOOT' the programs independantly of TRSDOS. To do this however I have to know what the level 2 Basic expects to find on disk.

All I have been able to ascertain so far is that BOOT/SYS must be at track 0 sector 0 and an executable program (normaly SYS0/SYS) at track 0 sector 5. I have not been able to find out what the TRS80s bootstrap loader expects to find in BOOT/SYS. I could of course attempt to disassemble the ROM but this would be a long and arduous task. Could you or any body else please help (either with the information or with a book reference).

The software package I am writing will include its own disk I/O primatives. I have found the book MACHINE LANGUAGE DISK I/O AND OTHER MYSTERIES by M J Wagner (IJG publications) very useful on this score. It is very well written and is packed cover to cover with useful information about controlling the FD1771 and FD179X floppy disk controllers. Rodnay Zaks PROGRAMMING THE Z80 I have found to be a very useful reference manual for the TRS80s 'heart'.

I have recently upgraded my system from 16K with Floppy Tape to 48K with Disks. I used a 48K internal from General Northern Microcomputers and their DP1000 disk/printer interface. After a few teething problems all seems to be working fine.

I hope the above two items might be of help or interest to some body and that someone can help me by answering my question.

Richard Jacobs,
9 Savernake Court
Wolverton Road
Stanmore
Middx
HA7 2RJ

John Symes of Microdeal tells me that, in view of all the hassles with freight agents and their charges, he is, reluctantly, ceasing to import this magazine after the April '83 issue. If anyone is interested, he does have a small stock of each of the last six or so issues, at £2.00 per copy plus postage (the Anniversary issue, if he has any left, was £3.00). His phone number is 0726 67676, and the new address is 41 Truro Road, St. Austell, Cornwall. They also handle the CLOAD magazine, and have just brought out a Joystick for the model III.

Cheers for now,
David Washford

REVIEW OF THE CREATOR - A BASIC PROGRAM GENERATOR

A listing of this program by Bruce Tonkin appeared in 80 Micro for January, and I have now typed it in and tested it for two applications. It is in Basic and is a generator for menu-driven data-file management programs. The files created by the object programs have maxima of 22 fields and 255 characters per record. Shorter record lengths are handled by packing. Only 1 key field is allowed for access to records, but you can update on any field. Access is by hashing and is quick; you can use the same key for different records and the program will search them one by one until you are satisfied. The maximum data file size is a diskful, and the generator program has to be modified for non-standard capacity drives. Different files can, however, be set up under the same name on different disks: hence a single database can be of unlimited length. The structure of an existing file is unchangeable. The usual range of data types can be handled. There is no sort facility, though I found it quite easy to write conversion programs which produced a sortable file from the hashed one.

The CREATOR could not be used by anyone who had no knowledge of computer jargon about file structure, but it requires no knowledge of programming. The prompts are not in a particularly logical order: they ask for the name of

the generated program, the drives on which the data files and the program are to be written, the maximum number of records, record length, displayed title of the program, exit codes to return to the menu, number and lengths of fields, key field, and for each field the data type, error checks and messages, and displayed title, and finally for an exit message to the operator. To understand what some of the prompts mean requires practice, and in one case I needed to modify the object program after generation. Half an hour to an hour would be a typical set-up time, allowing for any such modifications.

Tonkin claims that the generator and object programs work on any DOS for Model I or Model III. As they don't use variable-length records or special file types that is probably true. The snag is that the object program is written to disk with the line numbers out of order. Tonkin believes that the interpreter can cope with that, but it doesn't. I get round the problem by loading the object program into Scripsit, inserting

BASIC

at the beginning, saving it back in ASCII as a JCL file, and then telling the computer to DO it, in effect switching to Basic and auto-retyping the program in from disk. It can then be saved and run in the usual way. Those who haven't got NEWDOS80 or LDOS could use SCRIPSIT or Southern Software's EDIT to sort out the line numbers manually.

Failing that, it would be a tedious job getting the generated programs to work.

The package also includes a Report Generator Program which I have not found very useful. The generated programs allow limited manipulation of the data for print-out, but I should need to generate a new program for almost every report.

If anyone is interested I should be prepared to send copies of the programs to the library.

Christopher Currie

WANTED : OMIKRON Mapper for Model I. Peter Tootill (051)428 2733.

Please note my new adress: 7, Stockville Road, Liverpool L18 3EJ.
Telephone (051) 428 2733

(We still hold meetings of the Merseyside TRS 80/Genie Users Group in Liverpool

---

Liverpool Mailbox now has a new phone number;- (051)428 8924
Operating hours are basically 09.00 to 22.30, but this really means from when I get up to when I go to bed, so startup is earlier on weekdays and later at weekends! (You can ring anytime, and if the system is not on you won't disturb us. Mailbox may go 24 hours a day in the near future - watch for developments!)

WANTED : MX80FT or similar . Peter Tootill (051)428 2733

To see, or not to see
That IS the question!

sorry, Bill!

Not very many months ago I was complaining about invisible files on disk - how very easy it was to wipe them out in blissful ignorance - and I described how I overcame the password problem in order to restore visibility.

Now, I suppose that one could say that yet another wheel has turned full cycle - I WANT invisibility! This desire stems from the fact that now that I have double density, I prefer to let all my disks be system rather than data format, and let them all carry utilities such as D, DDSD, and Superzap. This of course is no problem - until I feed my disks into one of the directory programs; I waste three sheets of paper in repeatedly printing out those three file names!

The answer of course is to remove their Visible status. This is quite easily done with ATTRIB, but I didn't like losing another opportunity for 'zapping' - I get a much better kick out of changing bytes than I do from typing in in English! So, if anyone else is as daft as me, here is the procedure, based on the Newdos Superzap, but could be any zap program:

Find the directory entry to be altered, usually between sectors 172 and 179. The directory entry should start 10 - if it starts with 00 then it has been killed, and, sorry, changing it to 10 will NOT restore it! Simply change that 10 to 18 and the file is now invisible. If only I had known that twelve months ago I would never have learnt about 9642 as a password!

Cheers for now,
David Washford

SCRIPSIT AND VISICALC

A small tip for the people starting out with Visical and Scripsit. Use Scripsit to print out the Visical sheets instead of Visicals own print routine. The benefits are mainly cosmetic but the visual improvement can be considerable.

For instance you can use headers and footers and chain reports. You can expand abrieviations and use the global change command to correct those cramped lables. You can have all your print codes handled by Scripsit and page numbering.

The Visical column width can waste space if one column needs to be wide but with Scripsit you can remove surplus spaces. I find that for some calculations I end up with a single figure in a six figure column but now I can use those spaces for the overflow from the previous label. I can also add text to explain the figures or I can add Visical calulations in a Scripsit text file.

How do you do it? All is explained in the Visical manual but in case it is not clear, the following works for me. Use the /P option but instead of following with P for print try F for file. The edit line calls for a filename and I tend to use PR/PRF and I can specify a drive other than 0 by using PR/PRF:1. The edit line then asks for the lower right co-ordinates as with Print and as soon as these are entered the file is written to disc. All you have to do is to quit Visical and run Scripsit in the normal manner. The file is loaded as with any Scripsit file L PR/PRF.

A point to watch is that the out put from Viscal is dependant on two co-ordinates. The top left which comes from the initial cursor position and the bottom left. If you have a empty column in Visical to give a left margin then you will have an additional margin when it is printed in Scripsit. Also, if you inlcude an extra column on the right then you will have some unwanted blanks. I kept on getting an extra line feed giving me double line spacing. Eventually I realised that the line was too long and that there were six spaces in front of the end of line marker in Scripsit. I deleted the spaces and all was well. Eventually I realised it was quicker to go back and re print the file to disc and keep the co-ordinates tight around the figures.

An attractive feature is that you can see what you are doing and you can correct the position of labels to center them over the figures. It is much easier than tring to produce eloborate titles in Visical.

Regarding the comment in March newletter about connecting a Model 1 and the BBC computer. We have connected a UK101 and a Model 1 via the RS232 and we shall be attempting a link with the BBC model B now we have one.

For those that wonder why any one should want to link two different machines I can offer our reasons. The UK101 was our first machine and it lacks a good keyboard and we have no discs for it. Needing to develope and demonstrate some software for an exhibition we were faced with the having to type and debug the program. We also had the problem that the UK101 has only 8K of memory. This was just enough for the application but it is easier to get it working in a larger memory and then trim it to size.

Thus the program was written on the TRS-80 and saved to disc at the end of the sessions. When we were ready, we sent the program to the UK101 via the RS232 and saved it to tape in the UK101 format. The UK101 is designed for RS232 output and the last price for the whole computer kit was about the same price as the RS232 board for the TRS-80. We can also use the parallel printer interface on the TRS-80 instead of having to add a serial input to our Epson.

This work was carried out by my son Neil and technical queries should be refered to him 01 590 4721.

We are trying to build the Ambit Modem but the hardware is out of stock at the time of writing. We have also made enquiries about MicroNet but it is not ready. Any information on these would be appreciated.

We have upgraded our LDOS to 5.1.3 and used the Fix disc from Hardings to patch Scripsit. The very last patch LSCRIPTX is not working. We have checked the code and appears to be OK. Has anyone succeeded?. The rest of the enhansements are very good and make Scripsit better, for some purposes, than Superscripsit.

Derek Trayler.

P.S. Regarding the cost of paper for printers. This paper cost £1.50 inc VAT a roll from our local office suppliers. Our local newspaper pays .65p a roll excluding VAT but they buy paper in three ton loads (not all teletype rolls)

In-LIne Printer Control Codes with Scripsit.

Having just obtained a new set of Graphtrax Roms for my Epson MX80, I was faced with the prospect of either buying a new word processor that would allow me to use all these nice new facilities or hacking away at my old favourite, Scripsit. The first course would cost me money! - so I chose the second option, besides which, I am used to, and like Scripsit. The result of my hacking is presented below as a patch program which you can use to modify Model 1 Scripsit. For Model 3 users I suggest you carry out the modifications as per 80-Micro September 1981 to convert Model 1 Scripsit to work on the Model 3. The patch introduces a new control character CTL/P, which will print a 'spotty' character on the screen. This will be recognised by Scripsit as a printer control marker, and information placed between a pair of these markers will be sent to the printer as control codes. All of this can be done from anywhere within a line of text, and in addition it can also be used in Scripsit's justified mode, something that most commercial patches won't allow.

The dollar sign ,$, is used for the Escape character (ASCII 27). Alpha characters are converted to upper case if necessary and sent as their ASCII code. Numeric characters 0-9, representing a decimal number are converted to hex and then sent to the printer. Sequences of numbers can be separatd by spaces. For example, on the Epson printer to cause a word to be <u>underlined</u> you need to send :

[ESC],-,1 or 27,45,1 to turn it on
[ESC],-,0 or 27,45,0 to turn it off.

With this patch you would type:-

(@/P)$-1(@/P)underlined(@/P)$-0(@/P)

The text below illustrates some of the things you can make a Type 3 Epson do, having installed this patch.

This is ^{super}script and this one is _{sub}script. Now to see if the <u>underline</u> will work. This should turn on **emphasised**, this should be doubled and this one in **BOLD**. This word will be w i d e and now condensed. What versatility!.

There are 2 main ways you can install this patch depending on your capabilities and software tools. The program is given as an assembly listing, which should be typed in, assembled and given a suitable name such as PATCH/CMD. LOAD SCRIPSIT/LC and run PATCH. Save a new copy of your program with the DUMP command using 5200H, 7B8CH and 5200H as the Start, End and Transfer addresses.

Alternatively, you can load SCRIPSIT/LC and then use DEBUG or suitable monitor to enter the Hex code to generate the patched program in situ. This can then be saved as before. There is a fairly large amount of code to type in but I think it is well worth the effort. For those have neither the time nor patience give me a ring at the number below and on receipt of a disk and £5 I will send you a copy. If you think that this patch is useful please let me know. Then, if the user feedback says that there is a need, I will write another article showing you how to

extend this patch even further, to give the ability to print custom designed characters, such as Greek letters etc., from within Scripsit.

This patch IS compatible with the other modification that I wrote up re counting words in a text file. With respect to that, if you change the byte at 7AB9H from 2B => 43 , you will count the words according to where you place the cursor in your text. Placing the cursor at the end (shift-down arrow) will of course give the total words in the text as before.

```
;Patch for inline Printer Control Codes
;Using Model 1 Scripsit
;
;Jumps from Scripsit to new code
;
ORG 5333H
          JP SCAN1
ORG 6163H
          JP CTLP
ORG 721CH
          JP INIT
ORG 7227H
          JP CTLCNT
ORG 72A1H
          CALL ADDCTL
ORG 7A9EH
          JP TSTCTL
                    ;New code
;
ORG 7B00H
; * Additional code for line length formatting during
; printing to make printer control codes 'invisible'
;
SCAN1:    LD A,(DE)           ;See if
          OR A                ;end of
          RET Z               ;text
          BIT 7,(IY+34H)      ;See if we're
          JP Z,5336H          ;printing
          CP 7FH              ;See if print
          JP NZ,5336H         ;control character
SCANFW:   INC DE              ;Go along line 'til
          INC C               ;you find end print
          LD A,(DE)           ;control character
          CP 7FH
          JR NZ,SCANFW        ;'C' holds number of
          INC DE              ;characters to put into
          INC C               ;print buffer
          JR SCAN1
;
; * Adds extra control character CTL/P *
;
CTLP:     JP NC,6167H
          AND 9FH
          CP 10H              ;Is CTL/P pressed
          JP NZ,6167H         ;If it is load up with
          LD A,7FH            ;dotty character and
          JP 6179H            ;return
```

```
; * Decimal to Hex routine *
;
PLUS:       ADD HL,HL
            LD B,H
            LD C,L
            ADD HL,HL
            ADD HL,HL
            ADD HL,BC
            SUB 30H
            LD C,A
            LD B,00H
            ADD HL,BC
            RET
;
; * Zero counter used for number of printer control codes *
;
INIT:       LD HL,7ED9H
            XOR A
            LD (IY+0BH),A
            JP 721FH
;
; * Check for printer control codes and count them *
;
CTLCNT:     LD A,(DE)               ;Load 'A' with character
            INC DE                  ;Inc. pointer
            DEC C                   ;Dec. counter
            CP 7FH                  ; ? CTL/P
            JP NZ,722AH
LOOP10:     INC (IY+0BH)            ;Count 'til you reach
            INC HL                  ;the end one
            LD (HL),A
            LD A,(DE)
            INC DE
            DEC C
            CP 7FH
            JR NZ,LOOP10
            INC (IY+0BH)
            JP 722AH
;
; * Add total printer control codes to the line length *
; * the printer thinks its got, to get justification right
*
;
ADDCTL:     CALL 7558H
            ADD A,(IY+0BH)
            RET
;
; * See if printer will accept data and printit *
;
TSTRDY:     EX AF,AF                ;Save contents of 'A'
CHECK:      CALL 6004H              ;Check for CLEAR key
            LD A,(37E8H)            ;Test printer location
            BIT 07H,A               ;and loop if not ready
            JR NZ,CHECK
            EX AF,AF                ;Restore 'A'
PRINTIT:    LD (37E8H),A            ;Send contents of 'A'
                                    ;Model 3 substitute
                                    ;OUT (0F8H),A
            RET
```

```
; * Code for processing of printer control sequences *
;
TSTCTL:   CP 7FH              ;Is it Printer Control
          JR Z,PCTL           ;If it is - go process it
          CALL PRINTIT        ;If not just printit
          JP 5F74H            ;Return to main program
;
PCTL:     PUSH DE
          EX DE,HL            ;Swap buffer pointer into DE
          LD A,(DE)
WHAT:     CP 20H              ;Is it a space
          JR Z,PROCEED        ;Ignore used as a separator
          CP "$"              ;Is it an escape
          JR NZ,TRYNUM        ;Its not - see if its numeric
          LD A,1BH            ;Load with escape
DOIT:     CALL TSTRDY         ;Go and send to printer
          JR PROCEED
TRYNUM:   CP 3AH              ;It may be a number if its
          JR C,NUMBER         ;< '@'
          CP 40H              ;If its < 'A'
          JR C,DOIT           ;right away
          RES 5,A             ;Make upper-case
          JR DOIT
NUMBER:   CP 30H              ;see if < '0'
          JR C,DOIT
          LD HL,00H           ;Its a number
XTEN:     CALL PLUS           ;Call Dec. to Hex
          INC DE              ;Inc pointer in buffer
          LD A,(DE)           ;Check again for numeric
          CP 3AH
          JR NC,GO10
          CP 30H
          JR NC,XTEN          ;Its another numeric
GO10:     LD A,L              ;transfer value into A
          DEC DE              ;dec buff counter
          JR DOIT
PROCEED:  INC DE              ;Cont along buffer
          LD A,(DE)           ;checking for terminating
          CP 7FH              ;PCTL
          JR NZ,WHAT
          INC DE              ;Terminate & back to
          EX DE,HL            ;Normal processing
          POP DE
          JP 5F74H
          EXEC 5200H          ;402DH to go back to DOS
          END
```

Hex dump of the above assembly listing.
;Jumps from Scripsit to new code

Address	Change		To
5333	1A B7 C8	=>	C3 00 7B
6163	30 02 E6	=>	C3 1A 7B
721C	21 D9 7E	=>	C3 36 7B
7227	1A 13 0D	=>	C3 40 7B
72A1	CD 58 75	=>	CD 5A 7B
7A9E	32 E8 37	=>	C3 71 7B

```
                ;New code

7B00    1A B7 CB FD CB 34 7E CA 36 53 FE 7F C2 36 53 13
7B10    0C 1A FE 7F 20 F9 13 0C 18 E6 D2 67 61 E6 9F FE
7B20    10 C2 67 61 3E 7F C3 79 61 29 44 4D 29 29 09 D6
7B30    30 4F 06 00 09 C9 21 D9 7E AF FD 77 0B C3 1F 72
7B40    1A 13 0D FE 7F C2 2A 72 FD 34 0B 23 77 1A 13 0D
7B50    FE 7F 20 F4 FD 34 0B C3 2A 72 CD 58 75 FD 86 0B
7B60    C9 08 CD 04 60 3A E8 37 CB 7F 20 F6 08 32 E8 37
7B70    C9 FE 7F 28 06 CD 6D 7B C3 74 5F D5 EB 1A FE 20
7B80    28 2F FE 24 20 07 3E 1B CD 61 7B 18 24 FE 3A 38
7B90    08 FE 40 38 F3 CB AF 18 EF FE 30 38 EB 21 00 00
7BA0    CD 29 7B 13 1A FE 3A 30 04 FE 30 30 F3 7D 1B 18
7BB0    D7 13 1A FE 7F 20 C7 13 EB D1 C3 74
```

For Model 3, replace underlined bytes with D3 F8 00

Geof Smith. 01-950-6345 after 6pm.

EDITING SUCCESSIVE PROGRAM LINES

Peter Knewstubb wants to know how to make the Level II EDIT command move on to the next line of a program (Newsletter, 4,10,Problem section).

In an article in the February 1981 issue of 80 Micro, Dan Rollins presented a short routine (he called it Auto Edit) which allows editing of successive program lines. The 23 bytes of machine code are relocatable and in the following version for a tape-based Model I, I load it into a part of RAM reserved for DOS so that the code is always protected from being overwritten by BASIC.

```
100 'This program extends the EDIT command so that
101 'SHIFT<ENTER> puts the computer into edit mode for
102 'the next line.
103 '
110 DEFINT A-Z
120 DATA 58,128,56,31,208,237,91,236,64,19,205,44
130 DATA 27,197,225,35,35,205,63,27,195,102,46
131 '
132 'Load machine code into RAM.
140 CHECK=-2534
150   FOR ADDR=16477 TO 16499
160 READ MCODE:POKE ADDR,MCODE:CHECK=CHECK+MCODE
170   NEXT
180 IF CHECK THEN PRINT "Error in DATA statement.":END
181 '
182 'Link machine code with keyboard input routine (ROM)
183 'via DOS exit at 16815.
190 POKE 16815,195:POKE 16816,93:POKE 16817,64   'JP 16477
200 END
```

Run the program each time the computer is powered up; it can then be deleted. In the event of a 'crash' (MEMORY SIZE?) the POKE statements in line 190 will have to be re-entered. Note that repeated use of SHIFT<ENTER> acts as a single-step LIST command, which is a useful bonus feature.

John Couch

REVIEW OF GALTIC'S DP 1000 DISC & PRINTER UNIT

There must be quite a number of TRS-80 & GENIE owners for whom the cost of the expansion interface boxes and extra memory has been a deciding factor against upgrading from Level II. Internal keyboard memory expansion at a reasonable price has of course been available from a number of U.K. sources, but on its own such a modification is of limited use to the majority of owners. Recently however, General Northern Microcomputers, of Peterlee, Co.Durham, have introduced their DP 1000 unit to fill this need. Having previously fitted internally 48K of RAM and "The Rom" plus "Feature Rom A" from the same stable, I had no qualms about phoning my credit-card order through. In due course "IT" and the disc-drive, ordered at the same time, arrived. Alas : There had been a slight misunderstanding (my fault) and there was no DOS in the parcel, but another phone call to the proprietor Mr.Owens fixed that little problem, and despite the imminence of the Christmas holiday, he managed to make special arrangements and the DOS arrived with the minimum delay.

As expected everything worked first time. The unit is housed in a black steel box about 27cm X 15cm X 6cm and has its own integral mains power supply fed via about a meter of cable. A 50-way edge connector is provided on the front panel to interface to the keyboard, and two 34-way edge connectors are located at the rear, one for connection to the disc drives and the other for joining up to a printer. These are gold-plated on double-sided glass board and should not add to the problems experienced with some TRS80 keyboards. The unit can be supplied to suit

either the Genie or the TRS-80, and supports both the Genie AND the TRS-80 methods of addressing the printer port. A 25ms interupt driven heartbeat is built in. Single Density is standard but Double Density can be catered for with an optional adapter. Future developments include an add-on RS-232 module.

At £76.00 (& VAT in U.K.) the unit represents excellent value and has revealed no vices after three months extensive use, at home and in a school environment. It is not often that I feel able to unreservedly recommend a product, but this is one of the exceptions and anyone seeking a viable alternative to conventional disc upgrade should give this little box serious attention.

Richard Stockwell. Guernsey(forest 36291)

PRESTEL VIA LDOS AND ACCOUSTIC COUPLER.

After practising on the Maplin computer I finally decided I must get on to the 300 baud rate Prestel. The main factor was the loan of an accoustic coupler. I had some time ago borrowed the group's modem and RS 232 and had some fun, but, having lashed out on the Tandy RS232 when it was on offer recently and unearthing an under-utilised coupler from a friend who does this sort of thing for living, now was the time to try the big stuff.

The method I am using is to set up a 'DO' file (JCL) on LDOS to set the RS232, then to call up LCOMM *CL. The first thing then is to set up a receiving disk file (CLEAR/6 and CLEAR/9 then CLEAR/6 and CLEAR/:) to record everything for later reviewing. All that remains is to dial Prestel and away we go. It can be operated in a fairly slick manner, particularly if you have some idea of where you want to go. When you finish you turn off and close the file (CLEAR/6 and CLEAR/- then CLEAR/6 and CLEAR/0). SHIFT/CLEAR/= takes you back to LDOS and then, having set up the printer with PR/FLT (using another 'DO' file) you can print the whole operation to screen or printer by means of LIST FILESPEC or LIST FILESPEC (P). You can of course use SCRIPSIT instead if you insist.

Large snag --- I have to phone London at present as 300 baud is only on one Prestel computer at present. Secondary snag --- Connectors of course. I have to clean the edge connectors and the Mickey Mouse RS232 connector every time.

I include a printout of a session to show what the output looks like on the TRS-80. You will see that there is already some activity with computers. Have we (NATGUG) as a group any plans for utilising this facility so that we can talk to it and each other via it ? It would be a very good idea to get something going. I sent a reply-card to MICRONET 800 but they have not acknowledged it. I rang subsequently and was told there was a hold-up on software for the TRS-80. An advantage with MICRONET would be the possibility of using the MAILBOX utility and also the Birmingham computers.

Mac Randle.

```
****************************************
          **************
          **************
          *  *************
****************************************
          DRYDEN COMPUTER
 Please enter below your Customer
Identity.(Key ** if you make a mistake)
 You may program this Identity into
your set by going to PAG 924 (after
receiving the 'Welcome to Prestel'
frame).
          CUSTOMER IDENTITY ----------
                            ---------


                    P
                  P R E
                P R E S T
              P R E S T E L
              E S T E L
                T E L
                  L
  DRYDEN                      COMPUTER
                    L
 Please enter your Personal Password.
 Key ** if you make a mistake.
                   ---


              _  SUN 24 APR 1983 13:33
    GOOD AFTERNOON
    MR M J RANDLE

                    P
                  P R E
                P R E S T
              P R E S T E L
                E S T E L
                  T E L
                    L


    This is Dryden computer
    You last used it SUN 24 APR 08:47
    Press _ for Main Index                (( Should print # ))

    #

P R E S T E L              0a            0p
  Main Index
  ************************************
1 SPECIAL FEATURE   ABC TRAVEL GUIDE
  Scheduled Air services from every UK
  airport to any airport in the world
  ************************************
2 INFORMATION News,Weather,Travel,Sport
  Advice,Whats On,Ads,Local,Amusements.
3 PRESTEL CITISERVICE & OTHER BUSINESS
```

```
  INFORMATION
4 ALPHABETIC INDEXES to subjects & IPs
5 WHAT'S NEW                April 22nd
6 TALKING BACK TO PRESTEL Mailbox,
  Teleshopping & other message services
7 INFORMATION FOR PRESTEL USERS
  ************************************
9 MICRONET 800 Telesoftware,news,views
  & products for microcomputer users
0 HOMELINK Information on Homebanking &
  Teleshopping services
  ************************************
```

((Here you can find out about MICRONET 800 by pressing 9
however this is not yet available to TRS-80 and I did not
get a reply to my enquiry.
So type :- *10149#))

```
    *10149#

P R E S T E L           10149a       0p
************************************
COMPUTER ORGANISATIONS & CLUBS
************************************
1 VIEWFAX Micro clubs
2 ASSOCIATION OF LONDON COMPUTER CLUBS
3 ICPUG Microcomputer user groups
4 COMPUTER RETAILERS ASSOCIATION
  Microcomputer dealers and service
  organisations - members
5 BRITISH APPLE SYSTEMS USER GROUP
0 COMPUTERS MAIN INDEX
************************************

    1

V I E W F A X  258      258206a      0p

        **********  ** ********
        ******* **  * ***** ***
        ******* **  ***********
         ******** **********  *
         ******** ********* * *
If you are involved with a user group
or computer club, and would like your
own free pages on Viewfax 258, contact:
Viewfax 258
205'219 Corporation Street
Birmingham B4 6SE
or call us on 021'705'0258
    Key 1 for National Clubs/Groups
    Key 2 for Local Clubs/Groups
  0 Micro Magazine        9 Front Page
```

```
V I E W F A X  258      2582061a      0p
  *  *MICRO CLUBS** APRIL
  *  ******** ** * ***** *
  ************************
  ************** * *******
****************************************
Sharp MZ'80 Users Club SAE to Paul
Chappell,Computer Centre, Yeovil
College, Yeovil, Somerset.
Independent Commodore Prods. User Group
30 Brancaster Road, Newbury Park,
Ilford. IG2 7EP
Atom User Group SAE to Richard Meredith
Sheerwater, Yealm View Road, Newton
Ferrers, Devon.
Amateur Computer Club SAE to J McDonald
1 Carlton Court, Studley Grange Road,
London W7 2LU.
                    _ for More Clubs
  0 Club Index          9 Micro Index

    #

 V I E W F A X  258      2582061b      0p
  *  *MICRO CLUBS** APRIL
  *  ******** ** * ***** *
  ************************
  ************** * *******

UK101/Superboard User Group SAE to
Adrian Waters, 9 Moss Lane, Romford,
Essex.
Tangerine Users Club Please send SAE to
1 Marlborough Drive, Worle, Avon
BS22 0DQ
CP/M Users Group SAE to 11, Sun Street,
Finsbury Square, London. EC2M 2PS
British Apple Systems User Group SAE to
John Sharp, PO Box 174, Watford, Herts.
                    _ for More
  0 Club Index          9 Micro Index

    #

V I E W F A X  258      2582061c      0p
  *  *MICRO CLUBS** APRIL
  *  ******** ** * ***** *
  ************************
  ************** * *******

ZX80/81 National Software Association
SAE to 15, Woodlands Road, Wombourne,
Staffs.
```

Compucolor User Group SAE Bill Donkin,
19, Harwood Avenue, Bromley, Kent.
National Personal Computer User Assoc.
SAE to NPCUA, 11 Spratling Street,
Ramsgate, Kent.

Powertran Users Club SAE to Philip
Probetts, 50 Cromwell Road, Wimbledon,
London. SW19 8LZ

_ for more
0 Club Index 9 Micro Index

#

V I E W F A X 258 2582061d 0p
* *MICRO CLUBS** APRIL
* ******** ** * ***** *
****
********** * *******

Compukit User Club SAE to P. Crabb
21 Jones Close, Yatton Avon.
National Acorn Atom User Group SAE to
Alan Carr, 105 Fairhole Avenue, Gidea
Park, Romford, Essex
National TRS'80 User Group SAE to Brian <<------<<<< ((Here we are))
Pain, 40a High Street, Stony Stratford
Milton Keynes.

National ZX80 & ZX81 User Club SAE to
44 Earls Court Road, London W8 6EJ

_ for more
0 Club Index 9 Micro Index

0

V I E W F A X 258 2582062a 0p

********** ** ********
******* ** * ***** ***
******* ** ***********
********* ********** *
******** ********* * *

If you are involved with a user group
or computer club, and would like your
own free pages on Viewfax 258, contact:
Viewfax 258
205'219 Corporation Street
Birmingham B4 6SE
or call us on 021'705'0258
Key 1 for National Clubs'Groups
Key 2 for Local Clubs'Group
0 Micro Magazine 9 Front Page

```
V I E W F A X  258      2582060a     0p

                Key Number for
               area required.
          1
          4          2
                                    **CLUB
                                    * 403
                                      AREA
                                      Key 5
  0 Clubs Index          9 Micro Index

    5
```

```
V I E W F A X  258      2582066a     0p
  ****************************************
  * *  * * & CLUB 403  **              *
  * *  * * **** *   *  *  **** **** * *
  * **** * **** * * * *** *  * **** * *
  * **** * **** *****  *  **** **** * *
  ****************************************
      MICRO CLUBS IN THE 403 AREA
****************************************
Key for
11  Atom User Group
12  West Midlands Amateur Computer Club
13  Kingswinford BBC Micro User Group
14  West Midlands Vic Users Group
  0 403 Index          9 Viewfax Index
  8 UK User Groups

    12
```

```
   V I E W F A X  258      25820662a    0p
****************************************
  *** **                  * * *
     *   WEST MIDLANDS AMATEUR * * ***
     *       COMPUTER CLUB     * * *
  *** **                  * * *
****************************************
The West Midlands Amateur Computer Club
has over 150 members with a wide range
```

```
of interests and machines.
Club meetings take place n the second
and fourth tuesday of every month at
7.30 pm in the Physics Lab of Elmfield
School, Love Lane, Stourbridge and
visitors are always welcome.
                More _
  0 Inde          9 Viewfax Micro Mag

    9

V I E W F A X  258       258200a       0p
  ************************************
  * *  * * MICR MAG    ** 24/04/83  *
  * *  * * **** *   *   *  **** **** *
  * **** * **** * * * *** *  * **** *
  * **** * **** *****  *  **** **** *
  ************************************
        ALL OUR PAGES ARE FREE
****************************************
Key for:
 1  Latest News        91 ZX Index
 2  Viewfax fun        92 Apple Index
 3  Mode 7 art         93 Acorn/BBC Index
 4  Client Index       94 Our Address
 5  Clubs & Groups     95 Our Rates
 6  Sales & Wants      96 What's New
 7  Monsters Sale!     97 Thank you !
 8  Telesoftware       99 2 Hour Micro
  _  Numbers etc           prog from BBC
        MicroGnome - Key 98
           0 Front Page

    94

V I E W F A X  258       258293a       0p
  ************************************
  * *  * * VIEWFAX 258 **           *
  * *  * * **** *   *   *  **** **** *
  * **** * **** * * * *** *  * **** *
  * **** * **** *****  *  **** **** *
  ************************************

      205-219 Corporation Street
             Birmingham
               B4 6SE
            021 705 0258

          Mailbox:  021704115

     KEY ANY NUMBER FOR FRONT PAGE
```

```
V I E W F A X  258      258a          0p
  ************************************
  * *  * * FRONT PAGE  ** 24/04/83  *
  * *  * * **** *   *  *  **** **** *
  * **** * **** * * * *** *  * **** *
  * **** * **** *****  *  **** **** *
  ************************************
    Your Software on Prestel? Key 8
**************************************
Key:
11 Micro Magazine      16 About Viewfax
12 Latest Micro News   17 What's New
13 Telesoftware        18 Client Index
14 Our Address         19 Complaints Box
15 Write to us          _ New Readers
  (alpha boards only)
0 Prestel Micro Index
   YOUR COMPANY ON PRESTEL?  - KEY 5


    17


   V I E W F A X  258      258294a       0p
  ************************************
  * *  * * WHAT'S NEW  ** 19/04/83  *
  * *  * * **** *   *  *  **** **** *
  * **** * **** * * * *** *  * **** *
  * **** * **** *****  *  **** **** *
  ************************************
Key for
 1  News from the World of Computers
    (Updated Daily)
 2  "MicroGnome"  Alternative Micro Mag
    (19/04/83)
 3  Micro Quiz returns!
    (05/04/83)
 4  Index to West Midlands User Groups
    (22/03/83)
 5  Tandy Telesoftware Index
    (15/03/83)
 6  "Pictures" via Prestel
    (08/03/83)
  0 Viewfax Front Page
  9 Prestel What's New Index


    5


V I E W F A X  258      25821507a     0p
   ********              ***
   ********              ***
   ************ *****   ************
    ******************************
     ******************************
     ************ ****************
     ****** ***** ******  ********
     ************ ****************
     ************ *** ************
```

```
          Telesoftware Index     **
                                 **
        Key the required number
11 DSTOCK2     (Stock Control)
12 CHASE       (Game)
13 MATCHES     (Game)
14 STUNT       (Game)
15 STREK       (Game)
16 QUADRANGLE  (Game)
17 TELDIR      (Telephone Index)
18 CONVERT     (Hex ] Decimal Converter)
  0 Telesoft Index        9 Micro Mag

    18

V I E W F A X  258       258217442a   0p
****************************************
  TANDY TELESOFTWARE - CONVERT
****************************************
NAME: CONVERT
AUTHOR: J.W.B.Dunn
SUPPLIER: N.E.TRS-80 User group
COST: FREE
DOCUMENTATION: Non required
Key 0 Tandy Program Index
Key 6 Viewfax Telesoftware Index
Key 9 Viewfax Micro Magazine
Key _ to continue
****************************************
    TELESOFTWARE FROM VIEWFAX 258
****************************************

                                  (( Now change course again ))

    *1014#

P R E S T E L          1014a        0p
****************************************
COMPUTERS
****************************************
1 MICROCOMPUTERS/MICROCOMPUTING
2 COMPUTER NEWS & MAGAZINES
3 COMPUTER ORGANISATIONS & CLUBS
4 COMPUTER EDUCATION,TRAINING &
  RESEARCH
5 ON-LINE INFORMATION SERVICES
6 COMPUTER JOBS
7 PRESTEL/VIEWDATA SERVICES
9 PRESTEL ADVERTISER:Computer systems
  & equipment,services & bureaux,
  consultants,viewdata equipment
****************************************

                                  (( Now Log Out ))

    *90#
```

PRESTEL Op

P

((There is supposed to be a farewell))

DEFLUX YOUR RAMS

I noticed in the current issue of the news letter that Derek Trayler complained about memory problems with his L.N.W. expansion board. The most likely cause for this is residual flux around the pins on the 4116 memory sockets. These are dynamic RAMS and depend on the storage of charge in some incredibly small capacitors. It requires very little leakage to upset these devices so it is essential to thoroughly clean off any residual flux after soldering. This should be done as soon as possible after the soldering operation since rosin fluxes become progressively more difficult to remove the longer they are left. The best method for removing flux is to scrub the surface with a small brush after wetting with meths or isopropyl alcohol (IPA). Finally wash the surface with clean meths or IPA and dry off with clean tissue and allow to air dry. If you have a suitable dish it helps to soak the board in meths for a short time but remember these solvents are highly flammable so No Smoking or naked flames.

These comments apply to any high impedance system with potentially floating inputs. Flux residues will not normally cause problems with TTL circuits where impedances are generally low. If in doubt wash, if that has no effect you probably have a dodgy joint either in the cable or at a solder joint.

Ron Everitt

FOR SALE
========

ACULAB Floppy Tape Drive with 21 Wafers - Programs include ØZEN & Extended BASIC both with Manuals, LOCASE and many others.

ACULAB TRS-1 Parallel Port Decoder(Printer Interface)

Set of 4 Cassettes - Level II Instruction Course Part II

Any reasonable offer considered for all three items or separate.
Eric Brandes Tel:Gerrards Cross(0753) 882982

TRS-80 Model I, 48K, 2 disks, lower case,
Scripsit, Visicalc, Lineprinter II.
£ 950 or [illegible]ight split.

John Mills - Bradford 682594.

NATGUG SWINDON '83

by Andy Bell

Having spent a most enjoyable day at the Wiltshire Hotel, Bob asked me to put fingers to keyboard and write about some of Saturdays events.

Various lectures were going on amidst the flashing screens and whirring disk drives and I decided to go to the lectures given by Laurie Shields, the subject of which was mainly machine code.

Laurie is obviously a very experienced machine code programmer, but surprisingly his lectures were very "Listenable" to !

The first lecture in the morning was on DOS routines and it is the first of these I will try to explain. the program was called SPLIT and its purpose by the use of Shift/Break is to enable the programmer to shift between two basic programs in memory at the same time. The listing follows :-

```
100 MEM48K   EQU   0F8A4H
110          ORG   4049H
120          DEfW  MEM48K-2          ;THIS SET MEM SIZE
130 KBD      EQU   4016H             ;DRIVER ENTRY ADDRESS
140          ORG   MEM48K
150 SPLIT    LD    HL,3C00H
160          LD    DE,VSAVE
170          LD    BC,1024
180          LDIR
185          LD    HL,401DH          ;VIDEO DCB
190          LD    BC,42E8H-401DH
200          LDIR
210          LD    (SPSAVE),SP
215 ; THE ABOVE SAVES PARAMETERS OF SCREEN AND CURSOR
220          LD    HL,(4013H)        ,KBD DCB
225          INC   HL
230          LD    (HL),0
235          INC   HL
240          LD    (40A4H),HL        ;START OF BASIC
245          LD    HL,(KBD)
```

NATGUG SWINDON '83 cont

```
250           ND    (SKBD+1),HL
255           LD    HL,IKBD             ;NEW KBD DRIVER
260           LD    (KBD),HL
265           DEC   HL
270           LD    (40B1H),HL          ;TOP OF BASIC MEM
275           LD    DE,0FFCEH           ;SETS UP 50 BYTES
280           ADD   HL,DE               ;STRING SPACE IN
285           LD    (40A0H),HL          ;SECOND AREA
290           CALL  1B4DH               ;CLEaNS UP ALL THINGS
295           JP    066CH
300 ;ALL OF THIS UP TO NOW IS THE INITIALISING ROUTINE
305 IKBD      LD    HL,SKBD             ;TELL DOS TO LOOK FOR
310           LD    (KBD),HL            ;SECOND KEYBOARD
315           CALL  SWAP
320           JP    066CH
325 SKBD      CALL  $-$                 ;CHAR INPUT ROUTINE
330           CP    04H                 ;BREAK PRESSED ?
335           RET   NZ
340           LD    A,(3880H)           ;THIS TESTS THE SHIFT
345           AND   1                   ;KEY PRESSED 0350
350           XOR   1
355           RET   NZ
360           CALL  SWAP
365           XOR   A
370           RET
380 SAVE      LD    HL,3C00H
385           LD    DE,VSAVE
390           LD    BC,1024
395           CALL  SWAP                ;THIS TAKES ONE BYTE
t00           LD    HL,401DH            ;AT A TIME AND SWAPS
405           LD    BC,42E8H-401DH      ;BOTH SCREENS AT THE
410           CALL  SWAP                ;SAME TIME
415           POP   BC
420           LD    HL,(SPSAVE)
425           LD    (SPSAVE),SP
43p           LD    SP,HL
435           LD    A,(403DH)           ;LOCATION OF 32 OR 64
440           OUT   (255),A             ;MODE Of SCREEN.TESTS
445           PUSH  BC                  ;AND RESTORES
450           RET
```

NATGUG SWINDON '83 cont

```
455 SWAP      LD    A,(DE)
460           LDI                        ;ONE BYTE AT A TIME
465           DEC   HL
470           LD    (HL),A
475           INC   HL
480           JP    PE,SWAP
485 VSAVE     DEFS  1024
490 BSAVE     DEFS  42E8H-410DH
495 SPSAVE    DEFS  2
500 ;EXECUTION ADDRESS SHOULD BE aT 420DH
505           END
```

Assemble the code as a CMDFILE and after typing SPLIT/cmd enter BASIC and Initialise from basic with SYSTEM ?* /63652 .The screen should clear and pressing SHIFT/BREAK will swap between the two basic programs running at thg same time !!. Incidentally as you shift from one basic program to the other, the first will stop untill you re-activate it with Shift/Break.

Laurie Shields assembled this program on NEWDOS 80 v 2.0 and appeared to have no problems, but I am unsure of what will happen to LDOS users as I think LDOS uses the break key for something. This would be easy enough to change, just alter the compare at line 330.

The source code here is just some of the many notes I have made at the weekend and this letter is a bit rushed to meet the next newsletter

NATGUG (PART II)

On Saturday afternoon, Laurie Shields took us through another routine which is very handy for Level II users wh don't have some of the Level 3 commands and indeed some of these features are not available to disk users!

The features this routine allows is Hex, Octal and Binary conversion and also two byte peeking and poking of memory. All these commands use the format ?(&H3C00) or A=(&H4142) or ?(&B110011) or ?(&PEEK(16767)) etc, etc.

NATGUG SWINDON '83 cont

It is important to set Memory Size to 32640 (7F80H) or the routine will crash and be lost forever. So here is the source code:-

```
1 ; AMPS/SCE          4194H
2           ORG       4149H
3           JP        START
4           ORG       7F80H
5 START     RST       10H         ;GET CHAR 1 OF INPUT BUFF
6           CP        E5H         ;IS IT "&"
7           JR        NZ,NOTPK
8           RST       10H
9           CALL      252CH       ;ROM ROUTINE
10          PUSH      HL
11          CALL      0A7FH
12          LD        E,(HL)
13          INC       HL
14          LD        D,(HL)
15          LD        (4121H),DE
16          POP       HL
17          RET
18 NOTPK    LD        B,1
19          LD        C,2
20          CP        'B'         ;IS IT "B" FOR BINARY
21          JR        Z,CONT
22          LD        B,4
23          LD        C,10H
24          CP        'H'         ;IS IT "H" FOR HEX
25          JR        Z,CONT
26          DEC       B
27          LD        C,8
28          CP        'O'         ;IS IT "O" FOR OCTAL
29          JR        Z,CONT
30          DEC       HL
31 CONT     RST       10H
32          CP        '('         ;BRACKET THERE ?
33          JR        NZ,NOTEXP
34          PUSH      BC
35          CALL      252CH       ;EVAL CONTENTS IN BRACKET
36          CALL      0AF4H       ;ERROR CHECK
```

NATGUG SWINDON '83 cont

```
37          POP     BC
38          PUSH    HL
39          LD      HL,(4121H)
40          LD      E,(HL)
41          LD      D,0
42          INC     HL
43          LD      A,(HL)
44          INC     HL
45          LD      H,(HL)
46          LD      L,A
47          EX      DE,HL
48          ADD     HL,DE
49          EX      DE,HL
50          CALL    CONT2
51          POP     HL
52          RET
53 NOTEXP   LD      DE,0FFFFH
54 CONT2    DEC     HL
55          PUSH    HL
56          LD      HL,0
57          CALL    0A9AH           ;SET UP BASIC
58          EX      (SP),HL
59 NXTDGT   POP     IX
60          RST     10H             ;ANALYZE TEXT
61          EX      AF,AF'
62          RST     18H             ;CHECK FOR END OF STRING
63          RET     NC
64          EX      AF,AF'
65          JR      C,DIGIT
66          CP      'A'
67          RET     C               ;IF NO CARRY CONT
68          SUB     7
69 DIGIT    SUB     30H
70          CP      C
71          RET     NC
72          PUSH    IX
73          EX      (SP),HL
74          PUSH    BC
```

NATGUG SWINDON '83 cont

```
75 TIMES2  ADD     HL,HL          ;MULT HL BY 2
76         JP      C,07B2H        ;JUMP TO BASIC ERROR MSG
77         DJNZ    TIMES2
78         LD      C,A
79         ADD     HL,BC
80         LD      (4121H),HL
81         POP     BC
82         EX      (SP),HL
83         JR      NXTDGT
84         END     06CCH          ;EXEC ADDRESS IS 06CCH
```

These statements can be incorporated into basic statements and strings etc, and should be of help to all.

Here is a very handy routine for users of NEWDOS v 2.0 (if there is anyone else out there who uses it !!!).

You can create a Chain File as a CMD file using NEWDOS 80 v 2.0 's inherent Dos call to location 4419H (called DOSCALL). Here is the code :-

```
10 DOSCAL  EQU     4419H          ;DOSCALL LOCATION
20         ORG     5200H
30 START   LD      HL,COM1
40         CALL    DOSCAL
50         LD      HL,COM2
60         CALL    DOSCAL
70 COM1    DEFM    'DIR A,I,S'    ;WHATEVER YOU LIKE
80         DEFB    13             ;CARRIAGE RETURN
90 COM2    DEFM    'COPY,0,0'
100        DEFB    13
110        JP      402DH          ;FINISHED, GOTO DOS
120        END     START
```

As many commands can be chained together like this as you like and it is much faster than a /JCL file. Unfortunately NEWDOS 80 v 2

Andy Bell.

QUICK PRINTER II

Howis the carriage return suppressed, when the 16 char mode. I want to use this printer for labels at our local youth club.

Brian Pain

New member in Guernsey/Jersey Mike Allister 0481-21591 or call sign GU4EON with VGI + D/D

Bill Mason Ross on Wye on 098 981 353 has problems with his VG1 & HI-RES Can somebody give him a call

Has anybody a mint ND80 2.0 Model iii I require one for college.

Brian Pain

A Tandy customer wanted to interface from Miii to a telex. Has anybody any experience of this.

Brian Pain

MAILWRITER

Michael Bootes is happy to help members over the 'phone on this program.

HAUNTED HOUSE

Mr Jack Deul of Belgium is stuck in this game, has 6 items so far. All helpful suggestions to me, to be forwarded onto the above. Brian Pain.

Alf Robinson 11 Signals Unit RAF HEHN., BFPO 40 would like help on an EPROM board for the MI, and running a SIEMENS T150 teleprinter at 75 baud.

ROBOTICS

Has anybody got any form of the above working from their TANDY, as I am sure some very simple forms would be of interest to the schools and likeminded parents. Now I am returning to MKMUG perhaps I had better finish off what I started with K9-G7 using the model car chassis that got so far, and peter ed out. How many of us are guilty of half-finished projects? Brian Pain

Reg Kendall from Ipswich would like to chat with members using the NEC printer. Copdock 454

TANDY of LEICESTER are interested in a repeat prescription program.

COMPUTING TODAY JUNE '83

Has anybody typed in the above program for the MI, I would like a copy for college admin.

Brian Pain

SPECTRUM to MI or MIII

Has anybody got the above working or bought one of the RS232 interfaces available, as advertised in the SINCLAIR USER.

Brian Pain

HI-RES MODEL I

Norman Baust would like help over the phone on the boards that members have working. Call 0329-281480

FOR SALE

MI with 32k, L/case, green screen, MX80FT II Printer, cassette Dave Richards 061-624 8944

GAMES BOOTH

My summer project is the above, creating an environment with water for submarines, hot air for spacecapsules, vibrations for car type games. I have the water working using the game from the LII manual. Trevor Grover supplied the circuit to control the external relay for switching a winderscreen washer off & on. The current problem is how to avoid electrocution! I have a SPECTRUM which runs on 9volts..... Brian Pain.

COMPUTRONICS mags given to the group. For your free issue send 50p to cover env. & postage, with a label for sticking to env.

NEW BOOKS

FROM PRENTICE-HALL INTERNATIONAL

Assembly Language Graphics for the TRS-80 Color Computer

Don Inman, Kurt Inman and Dymax

The authors use sound+graphics to show how 6809 Assembly Language can be used to perform tasks that would be difficult or impossible with BASIC. The book contains video screen diagrams.

Spring 1983 280 pages pb. 8359-0317-6 £12.70

Clean Slate Word Processing for the TRS-80

Henry Milton

Commands, examples, hints and shortcuts for word processing using the TRS-80 are explained in this handy manual. Information is also given on original design source code so that you can implement a custom version of Clean Slate.

Summer 1983 376 pages pb. 672-22005-9 £15.25

Computer Art and Animation for the TRS-80

David Heiserman

David Heiserman shows you how to develop BASIC programs that produce interesting and useful graphics for a variety of applications.

Spring 1983 246 pages pb. 13-164731-8 £11.00

Hardware Interfacing with the TRS-80

John E. Uffenbeck

For the TRS-80 Model I and III, this book provides a detailed introduction to microcomputer interfacing. 14 practical hardware experiments which use BASIC are included.

Spring 1983 240 pages pb. 13-383869-2 £11.85

TRS-80 COBOL

Robert T. Grauer

For the TRS-80 Models II and III, this book explains ANS-74 COBOL as well as the characteristics of the operating system, TRSDOS.

Spring 1983 352 pages pb. 13-931204-8 £14.40

TRS-80 Models I, III and Color Computer Interfacing Projects

William Barden, Jr.

Careful step-by-step instructions and numerous illustrations are provided in this practical handbook. The author explains how to interface the computer to a wide range of devices, such as telephones, audio inputs, clock timers and temperature sensors.

Summer 1983 284 pages pb. 672-22009-1 £12.70

Prices and publication dates are correct at the time of going to press but may be subject to change.

Prentice/Hall International

66 Wood Lane End, Hemel Hempstead, Herts. HP2 4RG

INNOVATIVE
TRS 80-GENIE SOFTWARE
from the professionals



ACCEL 4

Another ACCEL to continue the evolutionary trend of these unique basic compilers! The first ACCEL a tape orientated compiler, was first introduced by Chris Paradine at the end of 1979. Since then it has gone through a number of enhancements, chiefly in speed and the number of basic statements which it will compile.

ACCEL 4 continues this trend, but its principle feature is its ability to leave memory untouched, except for about 127 bytes, at run time. In other words essentially all user memory is available for the compiled program. So far as is known this is the first time that such a claim can be made for a basic compiler available for the Tandy and Genie machines. It is indeed, therefore, "A Horn of Plenty".

Furthermore by reason of some very clever programming, ACCEL 4 approaches the other ultimate aim of a compiler, namely that the compiled program should be the same or less length as the original. It is inherent in compilers that the Object code is longer in length than the Source code. It is difficult to see how this cannot be so, but in one or two programs ACCEL 4 even achieves this aim.

It is impossible to give performance figures for a compiler in respect to speed of execution. This would vary with some programs to a quite startling factor of may be 20 or 30 times faster than the original and at the other end of the spectrum some software is not worth compiling. The criterion is, of course, the amount of time the program spends in accessing the peripherals. No compiler can change the length of time it takes a printer to print, nor the time a disk drive takes to read or write. In the worst case with a program whose sole object is to read or write to disk, and assuming that this is being done by the program continuously in its uncompiled form, then little or no speed increase will be achieved. A program, however, that spends its time doing calculations, manipulating strings or other work that is a function of the CPU, will benefit greatly.

Like its forefathers, ACCEL 4 supports Microsoft Basic and compared to its antecedents has very few restrictions. When ACCEL meets a Basic statement that it does not wish to compile or is unable to compile, it merely leaves it as it is and when such a statement is met at run time, ACCEL passes control back to the machine's interpreter for that line. One finishes up therefore with a rather unique combination of machine code with a sprinkling of Basic. ACCEL 4, however, supports pretty well all of the important Basic statements and commands.

ACCEL 4 is compatible with TRSDOS, LDOS and smal-LDOS on Model I and Model III Tandy machines, together with the original Video Genie, the Genie I and Genie II.

ACCEL 4 ... ... **£59.95** inc. V.A.T.
P. & P. 75p.

MOLIMERX LTD
A J HARDING (MOLIMERX)

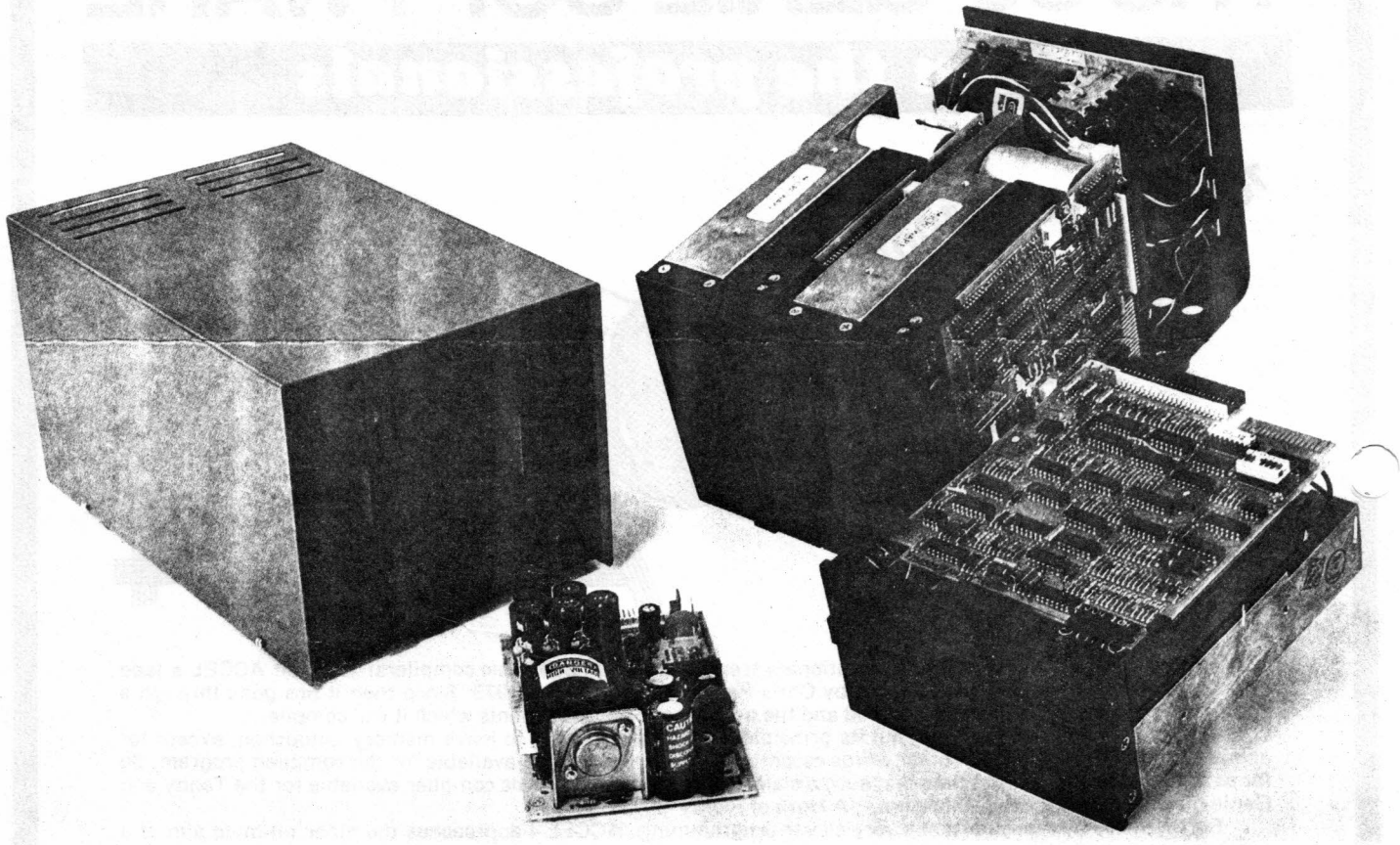
1 BUCKHURST ROAD, TOWN HALL SQUARE, BEXHILL-ON-SEA, EAST SUSSEX.

TEL: [0424] 220391 / 223636 **TELEX 86736 SOTEX G**

TRS-80 & VIDEO GENIE SOFTWARE CATALOGUE £1.00 [**refundable**] plus £1 postage.

Microware (London) Ltd.,
637a Holloway Road, London N19 5SS.
Telephone 272 6398/6237

MICROWARE ANNOUNCE THE Z/L RANGE OF FLOPPY DISC SUB SYSTEMS



The Control Data Family of Flexible Disk Drive is a single- or double-sided, random-access, low-cost storage device. Maximum storage capacity 1 megabyte on a 133.4-millimetre (5.25-inch) interchangeable disk. This disk drive is interchangeable with comparable products, providing industry compatibility in size, mechanical mountings, electrical interface, power requirements and physical appearance. CDC® FDD operates in single- or double-density formats. Single-density operation is achieved by using Frequency Modulation (FM) encoding, and in double-density operation uses Modified Frequency Modulation (MFM).

OPERATIONAL CHARACTERISTICS

Capacity	9408	9409	9409T
Unformatted			
Per disk	250.0 kbytes	500.0 kbytes	1 megabyte
Per track	3.1 kbytes	6.2 kbytes	6.2 kbytes
Formatted (16 sectors,	128/256 bytes)		
Per disk	163.84 kbytes	327.68 kbytes	655.36 kbytes
Per track	2.1 kbytes	4.1 kbytes	4.1 kbytes
Code	MFM	MFM	MFM
Transfer Rate	125 kbits/s	250 kbits/s	250 kbits/s
Average latency	less than 100 ms	less than 100 ms	less than 100 ms
Seek Time			
Track to track	less than 5 ms	less than 5 ms	less than 5 ms
Average Access	less than 80 ms	less than 80 ms	less than 132 ms
Setting time	less than 15 ms	less than 15 ms	less than 15 ms
Head Load Time (OPT)	less than 50 ms	less than 50 ms	less than 50 ms
Media	hard/soft sector	hard/soft sector	hard/soft sector
Rotational Speed	300 r/min		300 r/min
Track Density	48 TPI		96 TPI
Flux Reversal Density (track 39, side 1)	5876 FRI		5922 FRI
Number of Tracks	40		80
Inner recorded radius (side 0)	1.437 in (36.50 mm)		1.385 in (35.2 mm)
Outer recorded radius (side 0)	2.250 in (57.2 mm)		2.250 in (57.2 mm)
Inner recorded radius (side 1)	1.354 in (34.39 mm)		1.344 in (34.1 mm)
Outer recorded radius (side 1)	2.167 in (55.0 mm)		2.167 in (55.0 mm)

THE Z/L RANGE

Type	Capacity	Tracks	No. of Drives
ZL141	250 k	40	1
ZL142	500 k	40	2
ZL241	500 k	80	1
ZL242	1 Mb	80	2
ZL291	1 Mb	160	1
ZL292	2 Mb	160	2

Microware Disk Drive Subsystems are plug compatible with the following:–
TANDY, VIDEO GENIE, BBC MODEL B, NABCOM, SUPER BRAIN, IBM/PC. and many more.

Microware also supply floppy drives, Winchesters and sub systems to OEM's and the trade at very aggressive prices, offering the best price performance ratio available. Call and ask for details.

CABINETS AND PSU

Cabinets of steel, aluminium or plastic are available in a choice of colours, and each unit is fully guaranteed.

AC 8151 40 WATT SERIES

The AC 8151 Switch-Mode Power Supply has been designed for use in small terminals and other similar equipment. The AC 8151 has dual line inputs and regulated outputs of: +5V at 2.5A, +12V at 2.0A, −12V at 0.1A. This compact unit features low magnetic radiation and is built to conform to International Safety and RFI Regulations.

Name _____

Address _____

_____

YOUR LOCAL DEALER IS:–

NATIONAL TRS-80 & GENIE USER GROUP Newsletter

VOLUME 4 ISSUE 12 JUNE 1983

CONTENTS

2. Information on the Group
3. Editorial
3. Guidelines for contributors
4. Members' letters
6. Problem section
10. Data Writer review
12. Port FD and address 37E8 for the EG3013
12. Cable problem
14. FORTRAN tutorial part 3
17. Zap for Newdos/80 to allow null reply to TIME?
18. Spectrum/TRS-80 exchange of data
19. Printer set-up program
24. First impressions of the Model 4
26. Model III in business
27. TRS-80 Model 4
28. Upgrading Model I (Level II) 16K
29. Sorting tables prepared on a word processor
37. Speech synthesis using the G.I. SP0256-AL

NATGUG

INFORMATION ON THE GROUP

Membership of the Group is by subscription to the Newsletter, which is published monthly. Membership details are obtainable from the Group Secretary. Membership of the Group is open to anyone with an interest in the TRS-80 range of microcomputers, and compatible systems such as the Video Genie.

Details of the Group accounts and the constitution of the Group are obtainable from the Secretary.

Members requiring assistance with problems related to the TRS-80/Video Genie may call the Secretary. An attempt will be made to put them in touch with a member who can help with the problem.

Workshops are arranged from time to time in various parts of the country.

Sub-groups exist in many areas. A list is provided at the back of the Newsletter from time to time. Sub-groups also exist for medical and educational users.

The Group runs 80-NET, a computerised bulletin board service that is available to all users of microcomputers.

The Group maintains three software libraries (Models I, II and III) which are free to members. Library lists are obtainable from the Secretary. The medical and educational sub-groups maintain their own libraries.

A membership list is obtainable on disk from the Secretary.

Back numbers of the Newsletter are available.

Please send all contributions for the Newsletter to the Editor.

Chairman and Newsletter Editor:-

Leon Heller,
8 Morris Walk,
Newport Pagnell,
Bucks. MK16 8QD.
Tel: (0908) 613004

Secretary and Newsletter Publisher:-

Brian Pain,
24 Oxford Street,
Stony Stratford,
Milton Keynes.
Tel: (0908) 564271

Librarians:-

Model I: Leighton Davies, 105 Caerau Road, Caerau, Glamorgan. Tel: (0656) 738337.

Model II: Jim Hutton, 25 High Street, Stroud, Glos. Tel: (045 36) 4423.

Model III: Nick Clay, 6 Rectory Lane, Byfleet, Surrey. KT 14 7LL. Tel: (09323) 42348.

?!? SEPTEMBER 4 Blandford Forum
Come & see Os House @ home with dems
on M4 & M100 & Hard disc. Lunch in
the sun - and the coast is just down
the road. Lectures of course! ?!?

EDITORIAL

We had a stand at the recent London Computer Fair at Earls Court. My thanks to everyone who helped look after the stand. The exercise was moderately successful in that we gained something like 30 new members, and several of those who took away membership application forms will probably join eventually. Tandy were not exhibiting, but they loaned me a Model 16 to have on the stand. My thanks to Vince Moore, Ian Sandilands and Steve Sturgess of Tandy Corp. for arranging things.

You should shortly receive details of the Milton Keynes July Workshop. It will be devoted to DOSs.

It is subscription renewal time again! Brian will be sending out renewal forms separately from the Newsletter. It costs a bit more but people are more likely to notice it and act upon it than if we included it with the Newsletter.

We are proposing to change the Newsletter to an A5 booklet format. This will make it much easier to read, and will make my job a bit easier. The print size will be the same as at present. Can someone provide me with some art work for the front cover, A4 size, to be reduced down to A5 for printing?

Leon Heller

GUIDELINES FOR CONTRIBUTORS

1. If possible, send material in printed form.
2. Scripsit or Pencil disks/cassettes are acceptable.
3. Ensure your ribbon is in reasonable condition.
4. Printer output should be on A4 paper if possible.
5. If you send in handwritten material, write legibly.
6. Do not fold the sheets when posting them. Use a large envelope and keep them flat.

Leon Heller

WORKSHOP PROGRAMME

JULY 29/30/31	Milton Keynes
August 6/7	London BASIC
August 13/14	London Machine Code
August 21/22	London Word Processing
September 28-October 2	London PCW show
October 9	London Build a Maplin Modem
November 18/19/20	Swindon.
December 17/18 or January 7/8	London Model 16 & its Operating Systems.
February 1984	Milton Keynes Workshop and AGM.

MEMBERS' LETTERS

Thank you for your note about my letter in the Newsletter.

I have, in fact, got round the variable difficulty. The problem arose because I have four programs, three of which are theoretically a subsidiary to the fourth one, but in practise are sometimes called on their own. This means that each program must be provided with the necessary clear and dimension strings.

The solution, as is so often the case, proved to be very simple - I added a line before the clear and dimension string which checks to see whether certain key variables are equal to 0 or to some other value. If they are all equal to 0 then it runs through the dimension line, if any of them is not equal to 0 then it skips that line and goes into the body of the program. So far, I am glad to say, this has been a triumphant success.

I have also found a way round the Command 0 difficulty. This is simply to add an additional variable which is given a value from the initial position of each string in the list. Thus, in the example I sent, using B as the variable, B(1) would equal 1 and be attached to A$(1). Sorting both these by the strings produces B containing the true order of the strings.

I am grateful for your help.

I read with some interest Mr. Willey's letter and I can cast some light on his query about the use of more than one Delete command.

I ran into a problem with my program for listing variables which has to be merged on to the end of the program to be dealt with and in a recent case I found myself with insufficient memory to do the necessary manipulations.

After some thought I inserted a dummy line simply consisting of a line number and REM in the middle of the program and then added to Varilist the lines shown on the attached slip. I then loaded both programs and deleted the second half of the program to be dealt with. The effect of these lines is first of all to delete the part of the program that has been dealt with, then to merge the whole program and finally to delete again the part that has been dealt with. In the end I got the program to run right through, although at the first attempt it stopped just before the end, having run out of string space. I was however able to correct this by increasing the string space using Command F. Had I thought of it I could have inserted that in the program as well. (I now have).

It seems to me that the method might well be used to meet Mr. Willey's requirements of changing program lines, unless of course he actually wants to make up the program line as he goes along. Even this might be possible if you start the program with a line containing a string of variables with initial values for the dimensions and in the dimension line put, for example, DIMX(Y). You could then do a clear during the program, provide new values for all the variables and rerun the dimension line.

```
60110 IFAD=0ORLN>ETHEN60400
60112 IFLN<>4010THEN60120          Line 4010 is a dummy '4010REM' & not in the filed program
60113 CMD"F",DELETE5-4010          When we get to it delete the part we have done
60115 MERGE"CASHBOOK               Merge the whole program
60116 CMD"F",DELETE5-4000          Delete the part we do not want
60117 CMD"F=SASZ",10000            Increase the string space
60118 GOTO60040                    Get on with the good work!
60120 J=AD+4:IFJ>32767THENJ=J-65536
```

M. C. MATTHEWS, B.Sc., F.R.I.C.S.

I have just received the May Newsletter & notice there are one or two things about which I would like to comment.

Concerning the article about 'CREATOR', the simplest way to arrange the line numbers of the 'created' program is to ensure that it is loaded on to a TRS-DOS disc, then load the created program & without running it, save straight back on to the same disc.

TRS-DOS BASIC will automaticaly rearange the line numbers, it can then run under any DOS. NDBOV2 does not rearange the line numbers but can run CREATOR. The same goes for REPORTOR.

For the man with the problem with Scripsit & his MX80 there is a good program produced by Apparat called 'FLEXTEXT 80' that will give him all the control he wants, it is provided with it's own Modify program, a friend of mine uses it for all his letters & they are full of all sorts of changes to the script.

Reference GNOMIC'S various items, I wrote to them, enclosed an SAE, asking several questions that were not in their adverts, all I got a few weeks later was a Standard printout without any comments concerning the questions I had asked. I wrote them a second letter enclosing a copy of the first one pointing out that they had failed to answer any of my questions, that was at least two weeks ago but still no reply. What is the matter with them, don't they want my custom?

I notice that the end of Andy Bell's letter on page 35 is missing, you can tell him that there are quite a number of people who prefer NDBOV2 to any other DOS, I for one would like to know what's missing from his letter.

In the April issue of 80 Micro was a machine language program called VCMOD, as it was not on the Load 80 Tape I typed it in & assembled it (my first effort), it gives improved editing facilities to VISICALC as well the ability to send print control codes to a MX80, if anyone who wants a copy of the code sends me a disc & postage I will put it on for them, (either TRSDOS or NDBOV2).

I hope to be able to come to the meeting at the end of July but as usual it will depend on someone else's transport, the next Swindon meeting I shall be able to go down by bus I hope.

Last but not least, tell Brian that I have typed both the Basic as well as the machine code for the program from Computing Today (April & May issues not June) but as yet have not been able to get it to run, it crashes on the third statement in line 430 & I am certain that I have created the file 'USR/DAT' correctly but if he wants it, send me a disc & he can have it.

E. C. KILPATRICK.
3A GAINSBOROUGH ST
SUDBURY.
SUFFOLK,
CO10 6ET

FOR SALE

TRS-80 LII, 48k on board with L/C Tandy Green Screen VDU. Centronics P1 printer +interface. ESF floppy tape. CTR-80 Cassette Recorder. System Desk. 35 MICRO -80 plus several 80-MICRO MASSES of Sofware ring for Detail. Mike Long 01-228-34[illegible] Clapham.

PROBLEM SECTION

I am writing to you as the result of an enquiry which I made to Lowe Eletronics regarding the Genie I computer. They were unable to help me in this case but suggested that I contact you.

I am very interested in running CP/M on my Genie Computer and believe that it should be possible to implement on this system. My original thought was that it would be possible to swich out the resident ROM by using a memory card in the S-100 bus of the EG 3014 Expander. I had thought that this was the way in which the advertised CP/Genie operated. However, on writing to Lowe Electronics, I was informed that the CP/Genie had never really got of the ground.

They did, however, suggest two possible alternatives. Firstly, a version of CP/M for the TRS-80 which uses memory from 4000H up and which used the existing 64 x 16 screen. Secondly, a plug in board for the TRS-80 (Omikron?). which acts as a memory mapper and allows both 64K CP/M or normal operation.

I would be most obliged if you could let me know if you have any information on the products mentioned and whether or not you have heard of these working with a Genie.

D S Marsland
14 Canning Street
EDINBURGH
EH3 8EG

THANK YOU FOR YOUR HELP ON THE TELEPHONE, I HAVE NOT BEEN ABLE TO CONTACT MR CARNE AS YET. I MENTIONED IN OUR CONVERSATION THAT I HAVE A TRS-80 MODEL III WITH TWIN DISC DRIVES, AND WHAT I WISH TO DO IS TO MANIPULATE A SERIES OF CIRCLES FROM THE SAME ORIGIN, AND ROTATE THEM ALL OR INDIVIDUALLY THROUGH VARYING DEGREES TO ELLIPSES. I DO NOT WANT PARTICULARLY TO CLUTTER THE SCREEN WITH THESE, MERELY TO MOVE A POINT OR POINTS ALONG THE ORBITS AT VARIABLE RATES, AND HOLD THEM IF NECESSARY AT PARTICULAR POINTS.

I WOULD ALSO LIKE TO BE ABLE TO MEASURE THE ANGLES BETWEEN OBJECTS ON DIFFERENT ORBITAL PATHS. AS A LATER DEVELOPMENT I WOULD LIKE TO USE THE SAME AS THE ABOVE BUT FOR CIRCLES WITH SEPERATE ORIGINS.

THE PARTICULAR USE I WOULD LIKE TO USE THIS FOR IS TO DIPLAY ASTRONOMICAL INFORMATION AND PERHAPS TO DEVELOP A GAME.

THANK YOU FOR YOUR HELP IN THIS MATTER ALSO I WOULD BE PLEASED TO HAVE ANY MORE INFORMATION ON THIS SERVICE YOU PROVIDE AND DETAILS OF ANY MEMBERSHIP FEES YOU HAVE TO HELP YOU TO DEFRAY EXPENSES,

F. E. DRAKE
8 LEACROFT CLOSE
STAINES
MIDDX

From time to time notes appear in the Newsletter saying "If only beginners would letus know what they want.....". Here are two suggestions:-

Having just purchased an Expansion box for my V-G I would like to know more about S10C boards. Do they really work if they are just plugged in or is software needed as well.....well obviously for some applications I know. Where does one get them from? If they are so standardised why aren't they more advertised and cheaper?

Also I wouldlike to know something about Eproms. I have looked through back numbers of the Newslettters but can find no mention of them (nor of S100s). Though there are references to Eproms at workshops, but there that is no help to us out in the sticks!

Yours

CHRIS LOVELL.

(* A friend of mine, Gordon Wheelwright, imports S100 boards from the States, and sells them quite reasonably. Give him a ring on (0908) 613209.

S100 boards are available for practically any function you can think of. Some examples are music synthesis, RAM, EPROM, parallel I/O, serial I/O, Analogue-Digital conversion, modem, high. res. graphics etc. Prices range from about £100 - £500.

Generally, some special software needs to be written to use these boards, although some are supplied with software, which would probably need modifying for use on the Genie.

A major problem with S100 is the fact that it is not really a proper standard, and many boards would need some hardware mods. to work on the Genie. Carrying these out would probably require a considerable amount of digital design experience. The first step would be to obtain a complete description of the Genie S100 interface, in particular, detailed timing diagrams would be essential for both the Genie and the S100 board you wish to interface. Unless you really know what you are doing I'd forget about S100!

I'll write something about EPROMs in the next issue. LFH *)

Thank you for putting my letter in the news sheet a while back. Two or three members offered loans of a Service Manual, and I have one from a local member on 'long term loan'. Leighton Davies phoned about a loan of the manual, and I have half-promised him a program for the library when I can get around to tidying it. Stephen Carter wrote with interesting ideas from a Model III system for a text search program.

In the last Newsletter, John Couch (no address given) sketched a very interesting program which seemed well worked out. When I ran it my machine crashed and I'm a bit at a loss to know what to do about it. Do Model Is differ? Does it matter that I have no disk system.

Thanks for all the help, anyway.

Peter Knewstubb,
175 Huntingdon Road,
Cambridge.

(* The above letter illustrates how effective the group can be in helping members to solve problems with their systems.

If an address is not published, just give Brian a ring and he will let you have the information. Some members do not give their addresses, and I do not really have the time to find it out and insert it in the Newsletter.

John Couch's program worked fine on my system, under Level II BASIC. My machine is a very old one with the original ROMs. Perhaps the program is not compatible with the new ROM machines. LFH *)

WORD-4-WORD PATCH TO PRINT CHR$(0, 11 12)

In your March issue you published a letter by Jim Bartholomew giving a solution to my exact problem... except that I use Word-4-Word Plus, and it's not the same. Has anyone out there worked out a version of the patch to suit Word-4-Word Plus, please?

John Imber,
247 Kennington Road,
London SE11 6BY.
Tel: 01-582 5127

EXPANSION BOX RESET PROBLEM FOR VG

The 'answer' on P14 of October '81 Newsletter is not an answer at all unless one has the tape! I have written to Mac Randle, but to have to load a tape just for this seems ridiculous - surely there is something better. An EPROM? It might as well include the fix for CHR$(0) etc. as well.

Chris Lovell,
5 Montpelier Road,
Ilfracombe,
Devon.
Tel: Ilfracombe 63162

Although I attended the workshop at Swindon I missed Laurie Shields' lecture on m/c. Could someone please confirm the listing given with the May Newsletter is correct for the &H routine on page 33. Ignoring the deliberate mistake in line 2 which should be "ORG 4194H", I find that I have three problems.

1) What is the 2 byte POKE syntax?
2) The 2 byte PEEK use subtracts the adjacent locations, instead of multiplying the MSB by 256 and adding the LSB.
3) The following program operates in a strange way!-

```
10 INPUT A$
20 F=&H(A$)
30 PRINT F:GOTO 10
```

The first result is always 0, but the second and subsequent results are all correct. Have I discovered a software bug?

Comments please.

Bob Robinson,
6 Picket Piece,
Andover,
Hants.

* * * FOR SALE * * *

TRS-80 Model I, Level II, 48K with upper/lower case, monitor, CTR-80 recorder, sound box, DP1000 printer/disk interface, joy stick, over 40 tapes (games, utilities etc.), many books. List price about £850, asking £450 o.n.o.

Please telephone Martin Speakman on 061-748 8062 (Urmston, Manchester.

* * * WANTED URGENTLY * * *

Tandy tape recorder, CTR-80 preferably, or CTR-41.

Phone Mike Gibbons on (0604) 494038 or 021-747 2260 if no reply.

FOR SALE Very good condition 'GREEN SCREEN' monitor.Little used over two years. £60 o.n.o. RUGBY (0788) 812839 after 6.30

S.M.FURBER

FOR SALE COMPUTRONICS BACK ISSUES

I have a number of these mags from around mid81 to late /82.
They are available 50p each to cover p&p etc
Brian Pain.

DATA-WRITER REVIEW

I have read with great interest the article in March 1983 issue about the manuals, SAM and Data-Writer. It was round that time that I ordered my copy of this program. Now, after using it for a little while, I would like to add a few words to what has been said and maybe offer some useful tips.

The first suggestion is, that if you have not got it already, buy also a 40-track TRSDOS from Molimerx. As far as I can make out TRSDOS minimum system is smaller than Newdos 80 v2. I managed to squeeze all Data-Writer programs on four diskettes. The actual split is:

Disk one	Entry, Edit, Stats & Sort.
Disk two	Sort, Select, Manage, Maths.
Disk three	Labels, Letters, Report.
Disk four	Access.

Having Sort on disks one and two is quite useful.

Data can be entered using a word-processor but the program has also an Entry module. This module creates Data base control file and also formats the screen for entry and validates entries as described in the manual. Beware however using <return> ! If you do so after completing a line the cursor will move to the next, as it should do, but on saving the file you may find that you have lost part of it. Movement between the lines should be using up and down arrows. After last entry in a record the down arrow brings you to a choice of edit, ending the session with writing the last record, ending the session without writing the last record, or writing the record and returning for more entries. Shift - down arrow should delete the character under the cursor but this does not work !

The Edit module works as described and has a very useful search facility based on Instring routine. This routine differentiates between capital and lower case, but maybe one ought to compile any data base in one case only - capitals. The only "funny" in this module appears at the end of the program with the option to run the program again. One has to answer "N" twice to get out to DOS.

Stats is a very important program as it not only finds the format errors in the data base but also reports number of records, field names and the longest entry in each field. Always use this program before any data manipulation.

Sort is very good, works on two levels, and within each level permits still another option. It is fairly fast - my 531 records covering DIY between 1978 and 1982 were sorted by two labels, ascending alpha order in each in about 50 seconds.

Manage does permit to divide or combine data bases, add or delete fields, swap fields etc. Word of warning: when adding fields to my address data base I have lost six records out of 98. It was just as well that I ran Stats before and after.

Maths does not permit conditions (if, and, or) nor does it permit to zero the field. The way to overcome this is to enter a figure and then to deduct this figure. To illustrate: Infl-rate = 1, Infl-rate = Infl-rate - 1. This will zero the inflation rate field after data in it is no longer necessary.

FOR SALE

TRS-80 16K LEVEL II
+ TANDY MONITOR SCREEN
+ MANY BOOKS (MANUALS ETC)
+ DOZENS OF PRE-RECORDED PROGRAMS
A REAL BARGAIN AT ONLY #250
TE[illegible] DON BENLOW ON 0545 870206

Select works as described and using Select and Maths I have divided my Insurance Contents schedule by different types of possesions, entered the appropriate inflation rate for each type (Manage permits you repetition of entry on field creation), Maths worked out new insurance values, after combining the data base again, the Report produced very nice schedule for the Insurers. I rather like Report module and so far I have not encountered any difficulties with it.

Labels does permit typing different number of labels from different files, but this does not mean that, if you ask for one label for one file and two labels for the second file, these labels will be printed in one run. First one label for each record in both files is produced then second printing run starts to produce one label for each record in the second file. Do not be alarmed when, at the start of printing, you get one line of labels containing gibberish. This is a test run and printing proper starts after it.

Acess gives an impression of being written as an afterthought and in a hurry. To start with I could not run it at all. The program was returning to DOS with the message that key files did not exsist. I wrote to Molimerx about it and I received a reply, which I quote! "I am afraid I cannot properly look into your problem until you send me the disk containing your database."
Well, I did not send my disk but experimented instead. Without going into the details, if your data base has been created by a Word processor then run Entry to create control files. Access module will then create key files for your data base. These files have to be deleted and re-created if you add or delete records. One label only, chosen by the user, is used. String search does not differentiate between capital and lower case. If label contents repeat the last record is always found first. Using left arrow key one can then see earlier records. Display is limited to nine fields only and, if your record contains 20 fields - which is the maximum, you do not see eleven fields from field 10 onwards.

When quitting the program you are asked whether you wish to update the file. If you do not wish it the program ends leaving you in MEMSIZE !

On the whole I think Data-Writer a worthwhile program, even though slightly over-enthusiastically described by Molimerx. The manual is quite good though some things one has to find out through bitter experience. So far, apart from the already mentioned address file, DIY index, and schedule of house contents I also compiled an index to selected articles from different computer publications covering Aug 81 to Jan83 and I fully intend to use it for every kind of data base.

THE ALTERNATE SOURCE

David Washford
1 Alexandra Road
Bournemouth
BH6 5JA
England

Dear David,

I am sorry to say but we are going to discontinue the Alternate Source Programmer's Journal. I can send you our Bound Volume II which includes issues 7-12 for $19.95 plus shipping, but before I can process this order I will need the expiration date on your Visa card. Thanks in advance. I might also mention that we will be publishing a Bound Volume III which includes issues 13-18, the list price is $19.95 but if you put one on reserve now the price is only $15.95. This bound volume will be out sometime this summer. You should also specify whether you want your order shipped via surface or airmail.

PORT FD & ADDRESS 37E8 for the EG3013

Owners of the Video Genie EG3013 expansion box have to live with modifying TRS80 software which ADDRESSes the printer at 37E8H when their system PORTs the printer at FDH. A very simple hardware modification (2 diodes) will allow the use of software which uses either address or port. No more Superzap or Prozap, searching for 32 E8 37 or 3A E8 37 to change them into 00 D3 FD and 00 DB FD. Whichever system the software uses this little mod will accept.

I won't go into details of how it works just accept the fact that it does. It has saved me quite an amount of work as, being a user of data bases and word processors, I use the printer more than most. The first job is to get inside the expansion box itself and this is done by removing 5 screws from the bottom. Gently remove the top cover to reveal the PCB. Turn the box so the edge connectors are facing you. From the centre of the connector that goes to the computer, going towards the back of the PCB is a row of ICs the one nearest being a 74LS156 and the one behind a 74LS155. These are Z33 and Z34 respectively. Owners of the service manual should have no difficulty in locating these components.

What is required now is (a) A good fine soldering iron (b) Two IN4148 diodes and (c) A steady hand. Join the anode ends of the two diodes together and connect to pin 6 of Z33 the 74LS156. Take one of the diode cathodes to pin 5 of Z34 and the other to pin 11 of Z34 the 74LS155. Thats it ! You have now completed the necessary modifications. Ensure that switches 6 & 7 of DP1 are set to OFF (DP1 is the switch with the red spot) Before putting the cover back on, with everything connected, from DOS basic type the following --

```
FOR X=1 TO 25 : POKE &H37E8,13 : OUT253,13 : NEXT
```

You should get 50 carriage returns, 25 from 37E8 and 25 from port FD. Box it all up, forget about any software mods, sleep happy at night and good printing.

Ken Grey

PS Thanks to Mike Morris for passing along the gen on the mod.

CABLE PROBLEM

We had a problem with the special interface cable that we were using. On checking up we discovered that the connector had been damage and that one of the contacts was bent. There was nothing to do but replace it. The actual cable was a special two for one cable that allows two devices to share the same connector. The unit had come with our ESF but has other uses. To save sending to America we bought one locally.

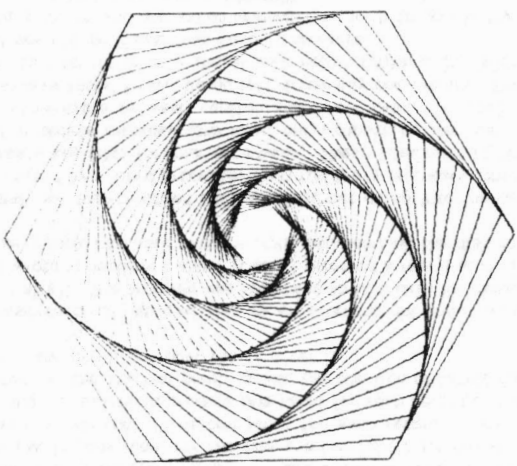
Fitting it caused no great problem but the cable did not work. Checking it with an AVO showed that all lines were conducting. The mystery deepend as wiser heads were baffled. The reason turned out to be simple and surprising. The leads were conected the wrong way up. The tracks on the top of the board were connected to the bottom of the extension and vice versa. It appeared that the connector was the wrong way up and merely required turning over. Later we discovered that the orientation makes no difference and that there are two systems which are similar but not compatable. It does not matter which you use as long as both ends are the same.

The ones normally supplied in the UK are good but do not seem to have a male version in the board edge IDC range. This makes them OK for printer and disc cables which are female both ends but you cannot make up the disc extender cables or the two for one cables because you need male on one end and female on the other. I can recommend the disc extender cables for any one that needs to move their system and has to take the top off to unplug the discs.

A small tip for the less affluent printer users. It is possible to reload the Epson cartridge although it has to be said that the cartridge was not designed for re-use. The cartridge has clips which can be undone and there is no difficulty in removing the old ribbon. Fitting the new ribbon is not difficult but it is fiddly. The best way seems to be to slip the ribbon in between the pinch cogs and thread it out the other end. By turning the white knob on the cartridge it is possible to wind the rest of the ribbon into the cardridge. Care should be taken not to unwind too much ribbon of the reel as there is plenty to tangle and it will make a mess if it contacts anything.

Refills seem to cost around £2.50 for the black ribbons and colours can be got for £3.50. This represents a big saving and worth the effort. I have only just started and have reloaded all my old cartridges. So far I have had no great problems but the refills have varied in quality. Some have been every bit as good as the Genuine Epson cartridges while others have appeared patchy.

Derek Trayler



```
5 '*******..COLOR PRINT BY KEITH BLOUNT ...16/03/83
10 CLEAR100 : DEFINTA,R,S : DEFSTR P
15 R#=240                              ' R = RADIUS
20 S#=0                                ' S = THE OFFSET ANGLE
25 LPRINTCHR$(10);LPRINTCHR$(17);LPRINTCHR$(16) 'GRAPHICS MODE
30 LPRINT"M240,-240";LPRINT"I";LPRINT"M242,0"   'SET POSITION
35 FOR A#=S# TO S#+300 STEP 60                  'SET SHAPE
40 P$="D"+STR$( R# * (COS (A# * .01745 )))      'X AXIS
45 P$=P$+","+STR$(  R# * (SIN (A# * .01745 )))  'Y AXIS
50 LPRINTP$ : NEXT A#
55 S#=S#+5                                      'INC. OFFSET ANGLE
60 R#=R#*SIN(1.047)/SIN(115)*.01745:            ' NEW RADIUS
65 P$="M"+STR$( R# * (COS (S# * .01745 )))      'NEW X AXIS
70 P$=P$+","+STR$( R# *(SIN (S# * .01745 )))'NEW Y AXIS
75 LPRINTP$                                     'MOVE TO INNER POS.
80 IF R# > 30 GOTO 35                           'GO AGAIN
85 LPRINT"M-240,-300" : END
```

FORTRAN tutorial PART 3

CORRECTION

In the last tutorial, on page 24 of the April 83 issue, the typical program using subroutines should be as follows:-

```
C         Main Program
          ................
          ................
          CALL ROTATE(FLAG,NUMBER)
          ................
          ................
          END
C
C
        SUBROUTINE ROTATE(METHOD,NUM)
        ................
        ................
        RETURN
        END
```

12.0 FUNCTIONS.

These are similar to the DEF FN word in Basic. That is it allows the programmer to simulate any built in function that the compiler does not offer. This is most useful particularly when a commonly used maths function is not available.

As an example I shall use a function to calculate the cost of an item plus VAT. This will be called FUNCTION VAT and the amount needing calculation passed inside the brackets. Thus the function and simple testing program could be like this:-

```
        FUNCTION VAT(AMOUNT)
C
        PERCNT = AMOUNT * 0.15
        VAT = AMOUNT + PERCNT
        RETURN
        END
C
C
        PROGRAM VATTST
C
        WRITE (5,5)
5       FORMAT (' ENTER THE AMOUNT')
        READ (5,10)FRED
10      FORMAT(F9.2)
        TOTVAT = VAT(FRED)
        WRITE (5,15)TOTVAT
15      FORMAT (' THE TOTAL WITH VAT=',F9.2)
        END
```

12.1 Things to notice about functions.

a) The value returned is the assignment, to the name of the function, in the last line before the RETURN. In this example the Function VAT has VAT as the left

hand side of an assignment before the return. This must always happen otherwise the function does not get its value. This value is then transmitted back to the statement that references the function.

b) In the example the main program has a simple VAT(FRED), which gets the total VAT price of FRED. However it could easily be VAT(FRED+JIM) or VAT(FRED+JIM/JEFF*LEON). That is a value is passed and not an actual variable.

c) Functions may have more than one parameter, although again this is not shown in the example. In fact this is clearly illustrated by the built in function AMIN. This finds the minimum of any number of parameters e.g. AMIN (57,2*52,7).

d) Functions are written in the same way as Subroutines. That is header statement followed by your various declarations and then the body of the Function. Oh and don't forget the RETURN and END !! It should be noted that a function can only pass back one value. The one indicated by its name, in the example VAT.

12.2 Differences between Subroutines and Functions.

The main difference between Subroutines and Functions is shown in d) above. Functions can only pass one value. Subroutines can pass as many as you like via its parameters. The use of either is a personal preference and I'm afraid to say that most programmers like Subroutines. However functions are great for mathematical operations while subroutines are better for actual transfer of control.

13.0 CHARACTERS IN FORTRAN

When Fortran 66 was just a babe a lot of processing of characters was considered a criminal waste of time. After all, computer time was expensive enough without offering nice prompts to tell people what to do. Thus this version of Fortran was never endowed with proper character manipulation. In fact they did not even call them characters but Holleriths. Nevertheless character manipulation is very easy provided they are treated as integers and you are reasonably careful. It was, after all, never designed for making a full-blown word processor. Since they are treated as integers it is easy to add, subtract divide and even raise them to a power. Naturally this can lead to errors that are difficult to trace but used properly they are no more difficult to use than strings in Basic.

FORTRAN tutorial PART 3

13.1 Input and Output of Characters

13.11 Input

For one character input the following example would work:-

```
      READ (5,5) INPUT
5     FORMAT (A1)
```

Notice the A1 for the Format descriptor and that INPUT is an integer. The format descriptor stands for Alpha-numeric and as usual the number after stands for the amount. However most people will require more than one character. In this instace only the format descriptor needs to be changed to A4. In order to read in more than four letters in then an array is used. Arrays are exactly the same as in Basic and need no further introduction. I suppose you have probably guessed that a Dimension statement is used, however that is not only the case. Arrays can be specified by type e.g Integer Fred(20) or Real JEFF(200).

An example to read 10 letters in:-

```
C
      BYTE FRED(10)          ! BYTE=INTEGER *1
      READ(5,5)FRED          ! FRED=Whole of array
5     FORMAT(10A1)           ! 10 of A1
```

Although it is short there is a lot of explaination required. Firstly anything following an exclaimation mark is a comment. Secondly Byte is another type and is mearly a shortened form of Integer taking up one Byte (suprise,suprise). It could easily be replaced by an Integer but Byte arrays are more efficient taking up one byte instead of 4. Thirdly the format is a repeating A1 format as used previously and I suggest if you don't understand it then reread section 10.5.

13.12 Output

After reading the it in - it is easy to push it out! Simply reverse the Reads for the Writes in every previous example. That is all.

13.2 Comparisons and Substrings

13.21 Comparisons

These are the usual If statements. However it must be restated that characters are treated as integers. Thus the following would work by comparison with the ASCII value of the letter as well.

```
      IF ( FRED .eq. 'K') GO TO 99
```

13.22 Substrings

Getting parts of a word is quite easy provided you store the word in an array as illustrated above. Then by careful manipulation of the array subscripts - substrings can be formed. For example if BYTE array WORD=FREDERICK then WORD(1) to WORD(4) is equal to FRED and

WORD(6) to WORD(9) is equal to RICK.

14.0 PROBLEM

As the last tutorial contained no problem I'm going to set a difficult one in this tutorial. I believe that you learn more by doing the program than by reading about how it is done. So although you may not complete it or get my solution, the amount you learn will benifit you in the long run. I will be giving a full run down of the program and the method behind it in the last tutorial.

The problem is:-

For an decimal integer between 1 and 9999 convert this to a roman numeral in the appropriate range. For example 8 is equal to VIII.

To get an idea of how it works try a few conversions. Remember their is a function called MOD. Characters can be passed as parameters through subroutines in ' '. Also that the computed Goto can be quite useful. These are all the hints I'm Giving.

Jeffrey Goldberg
11 Greenhalgh Walk,
London N.2.
01-458 3025

P.S. This is written using a modified version of Newscript. I have added the following major commands:-
Z - Another macro.
MS - Macro Save; saves the macros X,Y,Z to a file called Macro.
MR - Macro Recall; Does the opposite of above.
SP - SPy; allows you to look at another file whilst editing another document.
I have also done several other modifications too minor and numerous to mention. If someone wants to contact me on the above mods I am there after 7pm.

ZAP FOR NEWDOS/80 TO ALLOW NULL REPLY TO TIME? WHEN BOOTING

I use NEWDOS/80 exclusively as my pet DOS, as it seems very reliable, supports double density, and has most of the facilities I currently need. One little thing started to niggle me a bit however.

I like to put a valid date on any of my disk backups and therefore use SYSTEM option AY=Y to get the computer to ask for the date when first booting. Nine times out of ten, however, I'm not interested in the time but NEWDOS/80 insists that a valid time is entered, even if it is 00:00:00. The idea for the following patch came when I saw someone using MULTIDOS (thanks Don H), which quite happily took <ENTER> as an answer to its TIME? prompt, taking it to mean 00:00:00

Although the memory resident part of NEWDOS/80 SYS0/SYS has virtually no space left for patching (see end of Zap 060), there is plenty of room left in the initialising code which is only executed while booting. As the DATE? & TIME? dialogue is part of this latter code, a zap is easily installed and it doesn't matter that it is subsequently overwritten in RAM.

As it says in the book, no responsibility is taken if a later official Apparat Zap uses the space taken up by this patch. Additionally, DON'T, DON'T modify your original, valuable or only copy of NEWDOS/80. Zap as follows -

```
SYS0/SYS,12,6BH
      from  50 CD 64 4F 01     to  50 C3 B0 50 01

SYS0/SYS,13,CDH
      from                                              00 00 00
            00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
            00 00

        to                                              CD 64 4F
            3A 1B 43 FE 0D C2 56 4F 21 43 40 97 77 2B 77 2B
            77 C9
```

Now hopefully when you do a cold boot, you should be able to answer the TIME? prompt with <ENTER>, and get 00:00:00. You can still put in the actual time if your use requires it, of course.

Don Bannister 41, Myrtlefield Park, BELFAST, BT9 6NF
Home Tel: 0232- 668899

From J.R. Keneally.

SPECTRUM/TRS80 Interchange of data.

It can be quite useful to be able to dump an area of memory in the TRS80 across to a Spectrum. This can be done without any extra hardware apart from a cassette recorder. Just connect the TRS80 cassette output to the Aux or Mic input to the recorder, and the Spectrum Ear input to the Ear jack. Signals will then get across when you put the recorder on record, without a cassette in. Most tape recorders will do this, and you dont need to worry that you cant hear the signals through the Spectrum loudspeaker (or should I say soft-speaker). You will need some software, of course, but it's simple as long as you remember the Spectrum might be running twice the speed of the TRS80. For a copy, send a cassette+postage to me (see address for Camel Micros advert).

PRINTER SET-UP PROGRAM

The program, called SET/CMD simulates a DOS command and provides an extremely easy way to "set" your printer for emphasised or condensed print etc, without having to go into Basic. It was originally designed for my own setup which is as follows;

TRS80 model 3
TRSDOS 1.3
Epson MX80F/T 3

The program is simple (and virtually crash proof) and can easily be adapted for other machines and DOS's.

The command syntax is simple too, just enter from DOS ready, the command SET, followed by a space, followed by any of these parameters C L D E S A U P (in any order). The program will automatically reset the printer and then set the selected modes etc. If no parameters are included then the printer will just be reset.

C	Turns on the condensed character printing mode.
L	Turns on the large character printing mode.
D	Turns on the double character printing mode.
E	Turns on the emphasised character printing mode.
S	Sets a skip over perforation mode.
A	Converts printer to American character set.
U	Turns on the undirectional printing mode.
P	Deselects paper end detector.

Command examples ;

SET	With no parameters, this will simply reset the printer.
SET UC	This will set the printer to produce unidirectional condensed print.
SET SA	This will set skip over perforation and select the American character set. (Ideal for those program listings.)
SET LC	This will set the printer to produce Large/condensed print (8 characters per inch).

On receipt of the command the program will do its bit and return to DOS with a simple message such as "Skip over set." or "Condensed mode on.".

If the printer is switched off or is not ready for some reason, then the program will return to DOS with the message "*** printer not ready ***".

How the program works.

The program evolves around the fact that all DOS command lines are placed into a buffer. In TRSDOS 1.3 this buffer starts at address 16933 decimal (4225 hex). When a program is called, the register pair HL contain the address of the first nonblank character. So when we enter "SET SL" this is what the buffer contains ;

Address	16933	16934	16935	16936	16937	16938	16939	16940	
Contents- Hex	53	45	54	20	53	4C	ØD		
Ascii	S	E	T		S	L	<E>		<E> = enter

In this case HL will point to address 16937, which contains the letter "S". Lines 660-820 interpret the contents of this address and the program then jumps to the relevant subroutine which sends the codes and prints the messages.

The register pair HL is then incremented by 1, pointing now to the letter "L" and the process repeats. It will continue looping like this, collecting the next parameter, sending the codes and printing messages until it comes to the <E> (enter).

This indicates the end of the command line and the program, with no more parameters to work on, returns to DOS (lines 310-330).

There is a bit of printer checking at the start of the program and that's about it.

The listing is fairly self explanitory.

Modifications.

As far as I know only two modifications are needed to make this program compatable with the model 1.

Firstly, model 1 sends characters to the printer by loading them into address 37E8H (14312 decimal). Model 3 however, sends them via the port 0F8H. So change line 610 to this ;

```
        LD      (37E8H),A       ;Get printer status
```

Secondly, the program uses a ROM subroutine to print messages onto the screen (021BH), this is absent in the model 1 ROM.

Model 1 users will have to delete line 190 of the listing and add the following code;

```
VDLINE  LD      A,(HL)          ;Get character
        INC     HL              ;Point to next char
        CALL    0033H           ;Display character
        CP      0DH             ;Is it "enter"
        JR      NZ,VDLINE       ;Repeat if not
        RET                     ;Return
```

To avoid any conflictions enter the above code between lines 631-639. That should settle any computer differences.

As far as printers are concerned it should be a simple case of altering the codes and messages. Incidentally the MX80F/T3 is very similar to the earlier Epson printers with Graftrax or Graftrax plus so it may run on these without modifications.

To add further features such as italics or underline print etc, just add a routine to the end of the program which does what you want and insert two lines in the "filter" (lines 630-790) to send the program to your subroutine.

As for different DOS's there should be no problems but I cannot say for sure.

By the way, when you assemble this program be sure to call it SET/CMD otherwise change the command syntax to that of your program name.

For those with a tight schedule, if you send me a TRSDOS formatted disk and the return postage I will be happy to return the assembled program along with the source code. For those with plenty of time on their hands, if you have any problems give me a ring. (01 854 2171).

Michael Tomlinson.
60 Brookhill close,
Woolwich,
London.
SE18 6UD.

```
                00100 ;***********************************************
                00110 ;*                                             *
                00120 ;*          SET/CMD PRINTER TYPE SETTER        *
                00130 ;*                By Mike Tomlinson            *
                00140 ;*       Version 1.0            13/05/83      *
                00150 ;*                                             *
                00160 ;***********************************************
                00170 ;
                00180 ;
021B            00190 VDLINE    EQU     021BH
                00200 ;
                00210 ;
8000            00220           ORG     8000H           ;Start program here
8000 E5         00230 START     PUSH    HL              ;Save pointer
8001 3AE837     00240           LD      A,(37E8H)       ;Get printer status
8004 CB5F       00250           BIT     3,A             ;Is printer switched on ?
8006 2805       00260           JR      Z,PRTOFF        ;If not then exit
8008 CD2D80     00270           CALL    RDYTST          ;Is printer ready ?
800B 2808       00280           JR      Z,RESET         ;Skip next 11 bytes if so
800D 21EB80     00290 PRTOFF    LD      HL,ERROR        ;Load error message
8010 CD1B02     00300           CALL    VDLINE          ;Print it
8013 E1         00310 EXIT      POP     HL              ;Replace pointer
8014 AF         00320           XOR     A               ;Clear register A
8015 C32D40     00330           JP      402DH           ;Return to DOS
                00340 ;
```

```
8018 CD3580   00350 RESET   CALL    SENDC           ;Send control code
801B 3E40     00360         LD      A,64            ;Reset printer
801D CD3780   00370         CALL    SEND
8020 7E       00380 LOOP    LD      A,(HL)          ;Get parameter
8021 FE0D     00390         CP      0DH             ;Is it a C/R ?
8023 28EE     00400         JR      Z,EXIT          ;Return to DOS if so
8025 E5       00410         PUSH    HL
8026 CD4180   00420         CALL    FILTER          ;Go to it !
8029 E1       00430         POP     HL
802A 23       00440         INC     HL              ;Increment pointer
802B 18F3     00450         JR      LOOP            ;Return for next
              00460 ;
              00470 ;       *****     PRINTER READY TEST      *****
              00480 ;
802D 3AE837   00490 RDYTST  LD      A,(37E8H)       ;Get printer status
8030 E6F0     00500         AND     0F0H            ;Mask Low nibble
8032 FE30     00510         CP      30H             ;30H =  printer ready
8034 C9       00520         RET
              00530 ;
              00540 ;       *****     PRINTER ROUTINES      *****
              00550 ;
8035 3E1B     00560 SENDC   LD      A,27            ;27 = Epson control code
8037 F5       00570 SEND    PUSH    AF
8038 CD2D80   00580 LOOP1   CALL    RDYTST          ;Is printer ready ?
803B 20FB     00590         JR      NZ,LOOP1        ;If not, loop until it is
803D F1       00600         POP     AF
803E D3F8     00610         OUT     (0F8H),A        ;Printer port (Mod 3)
8040 C9       00620         RET
              00630 ;
              00640 ;       *****     IDENTIFY PARAMETERS     *****
              00650 ;
8041 FE43     00660 FILTER  CP      'C'
8043 281D     00670         JR      Z,CONDEN
8045 FE4C     00680         CP      'L'
8047 2825     00690         JR      Z,LARGE
8049 FE44     00700         CP      'D'
804B 2835     00710         JR      Z,DOUBLE
804D FE45     00720         CP      'E'
804F 2840     00730         JR      Z,EMPHAS
8051 FE53     00740         CP      'S'
8053 284B     00750         JR      Z,SKIP
8055 FE41     00760         CP      'A'
8057 285B     00770         JR      Z,AMERIC
8059 FE55     00780         CP      'U'
805B 286B     00790         JR      Z,UNIDIR
805D FE50     00800         CP      'P'
805F 287B     00810         JR      Z,PAPER
8061 C9       00820         RET
              00830 ;
              00840 ;       *****     PRINTER SET ROUTINES      *****
              00850 ;
8062 3E0F     00860 CONDEN  LD      A,15            ;Condensed routine
8064 CD3780   00870         CALL    SEND
8067 210581   00880         LD      HL,CMESS        ;Point HL to message
806A CD1B02   00890         CALL    VDLINE          ;Print it
806D C9       00900         RET
              00910 ;
806E CD3580   00920 LARGE   CALL    SENDC           ;Large print routine
8071 3E57     00930         LD      A,'W'
8073 CD3780   00940         CALL    SEND
8076 3E01     00950         LD      A,01H
8078 CD3780   00960         CALL    SEND
807B 211981   00970         LD      HL,LMESS        ;Point HL to message
807E CD1B02   00980         CALL    VDLINE          ;Print it
8081 C9       00990         RET
```

```
                01000 ;
8082 CD3580     01010 DOUBLE  CALL    SENDC           ;Double print routine
8085 3E47       01020         LD      A,'G'
8087 CD3780     01030         CALL    SEND
808A 212981     01040         LD      HL,DMESS        ;Point HL to message
808D CD1B02     01050         CALL    VDLINE          ;Print it
8090 C9         01060         RET
                01070 ;
8091 CD3580     01080 EMPHAS  CALL    SENDC           ;Emphasised print routine
8094 3E45       01090         LD      A,'E'
8096 CD3780     01100         CALL    SEND
8099 213A81     01110         LD      HL,EMESS        ;Point HL to message
809C CD1B02     01120         CALL    VDLINE          ;Print it
809F C9         01130         RET
                01140 ;
80A0 CD3580     01150 SKIP    CALL    SENDC           ;Skip over perforation
80A3 3E4E       01160         LD      A,'N'
80A5 CD3780     01170         CALL    SEND
80A8 3E04       01180         LD      A,4
80AA CD3780     01190         CALL    SEND
80AD 214F81     01200         LD      HL,SMESS        ;Point HL to message
80B0 CD1B02     01210         CALL    VDLINE          ;Print it
80B3 C9         01220         RET
                01230 ;
80B4 CD3580     01240 AMERIC  CALL    SENDC           ;Set American mode
80B7 3E52       01250         LD      A,'R'
80B9 CD3780     01260         CALL    SEND
80BC 3E00       01270         LD      A,0
80BE CD3780     01280         CALL    SEND
80C1 215E81     01290         LD      HL,AMESS        ;Point HL to message
80C4 CD1B02     01300         CALL    VDLINE          ;Print it
80C7 C9         01310         RET
                01320 ;
80C8 CD3580     01330 UNIDIR  CALL    SENDC           ;Unidirectional printing
80CB 3E55       01340         LD      A,'U'
80CD CD3780     01350         CALL    SEND
80D0 3E01       01360         LD      A,1
80D2 CD3780     01370         CALL    SEND
80D5 217081     01380         LD      HL,UMESS        ;Point HL to message
80D8 CD1B02     01390         CALL    VDLINE          ;Print it
80DB C9         01400         RET
                01410 ;
80DC CD3580     01420 PAPER   CALL    SENDC           ;Paper end detecter off
80DF 3E38       01430         LD      A,'8'
80E1 CD3780     01440         CALL    SEND
80E4 218C81     01450         LD      HL,PMESS        ;Point HL to message
80E7 CD1B02     01460         CALL    VDLINE          ;Print it
80EA C9         01470         RET
                01480 ;
                01490 ;       *****   MESSAGES   *****
                01500 ;
80EB 2A         01510 ERROR   DEFM    '*** Printer not ready ***'
8104 0D         01520         DEFB    0DH
8105 43         01530 CMESS   DEFM    'Condensed print on.'
8118 0D         01540         DEFB    0DH
8119 4C         01550 LMESS   DEFM    'Large print on.'
8128 0D         01560         DEFB    0DH
8129 44         01570 DMESS   DEFM    'Double print on.'
8139 0D         01580         DEFB    0DH
813A 45         01590 EMESS   DEFM    'Emphasised print on.'
814E 0D         01600         DEFB    0DH
814F 53         01610 SMESS   DEFM    'Skip over set.'
815D 0D         01620         DEFB    0DH
815E 41         01630 AMESS   DEFM    'American mode on.'
816F 0D         01640         DEFB    0DH
8170 55         01650 UMESS   DEFM    'Unidirectional printing on.'
```

```
818B 0D         0166            DEFB    0DH
818C 50         0167    MESS    DEFM    'Paper end detector off.'
81A3 0D         01680           DEFB    0DH
8000            01690           END     START
00000 TOTAL ERRORS
31497  TEXT AREA BYTES LEFT


AMERIC 80B4 01240   00770
AMESS  815E 01630   01290
CMESS  8105 01530   00880
CONDEN 8062 00860   00670
DMESS  8129 01570   01040
DOUBLE 8082 01010   00710
EMESS  813A 01590   01110
EMPHAS 8091 01080   00730
ERROR  80EB 01510   00290
EXIT   8013 00310   00400
FILTER 8041 00660   00420
LARGE  806E 00920   00690
LMESS  8119 01550   00970
LOOP   8020 00380   00450
LOOP1  8038 00580   00590
PAPER  80DC 01420   00810
PMESS  818C 01670   01450
PRTOFF 800D 00290   00260
RDYTST 802D 00490   00270 00580
RESET  8018 00350   00280
SEND   8037 00570   00370 00870 00940 00960 01030 01100 01170
                    01190 01260 01280 01350 01370 01440
SENDC  8035 00560   00350 00920 01010 01080 01150 01240 01330
                    01420
SKIP   80A0 01150   00750
SMESS  814F 01610   01200
START  8000 00230   01690
UMESS  8170 01650   01380
UNIDIR 80C8 01330   00790
VDLINE 021B 00190   00300 00890 00980 01050 01120 01210 01300
                    01390 01460
```

FIRST IMPRESSIONS OF THE MODEL 4

Looking very much like a Model III in a fawn grey housing, the most obvious difference is that there are more white keys on the keyboard. All keys which are not printable characters are now white, including the new control, capitals lock and function keys. All codes can be generated by key manipulation, graphics from the keyboard even. But looks aside, how does it match the claims made of it ?

Leaving for later the operation of Model III software, the first thing that I did was boot the TRSDOS 6.0 operating system disc supplied. Reports are that this system is from Logical Systems, the authors of LDOS, and first sight certainly supports this. I have referred to the LDOS system manual for information which I have found to be of direct application to TRSDOS 6.0. Model 4 ram starts at 0000H, not at the 4000H of the Model I/III, so the absolute addressing information from the LDOS manual is not applicable, but the general information on the organisation of the operating system and its management of discs and hardware has applied in each instance at which I have looked.

The library commands appear to work exactly as described in the manual, and the LDOS like features of TRSDOS 6.0 which I have had time to try, such as Memdisc and the Job Control Language seem to work as they should, although I confess to a less than thorough test of the JCL since it has some very advanced features.

I have not tried them, but the disc has spooling software as well as a general RS232 communications package. An additional module is included for communication with the Model 100 to come.

The TRSDOS manual describes full I/O control similar to that of LDOS. In fact the claims made of 'All these things and more' does seem to be justified. Short of the Model II range, the manual is the thickest yet from Tandy, and manages to be so in spite of not going into the technical details which are in an LDOS manual. This will be a shortcoming to those amongst us wanting intimate details of the operating system, but in fairness the information is available from other sources. As a manual for the new user, it contains plenty to be absorbed without the added confusion which a detailed technical work would provide. Whilst a good programmer will relieve the user of the need to know, I feel that there are so many possibilities and options that the newcoming programmer may need assistance.
(NOTE- a golden opportunity to continue the good work of our Group).

The facility for an extra 64K of ram has not been tested, because although it is there the extra ram isn't and even if it was I have no software to use it. I think that this will be a tempory situation only, and that Visicalc etc will be made available soon.

The new Microsoft Basic will give Model I/III users some surprises. The first is that entry with no file handling reserved leaves only 29424 bytes for program and data storage. This does not compare with the 38263 of my trusty Model I, although no indication is given as to the potential expansion of this when the extra ram is fitted.
The next surprise is that it is the 'standard' Microsoft basic, not the one with which we are familiar. This means that key words (FOR, PRINT, etc) must always be followed with a character which is not allowed as part of a variable name, (such as a space). The facility for long variable names is fine, but I am used to my machine knowing 'FORX=ATOB' without my having to spell out 'FOR X=A TO B', which needs more thought and typing. Is this the price of CP/M compatability ? (Incidently, CP/M for the Model 4 is not yet released.)

The Tandy specification claims that Model III software can be used on the Model 4 without change. What better way to test than a Model III Newdos 80 into drive 0 and reboot. Having booted, I found no difference between operation on the Model III and the Model 4 in Model III mode. Similarly, Model III TRSDOS, DOSPLUS and Super Utility also worked on the Model 4 exactly as they did on the Model III. I have even had a play with a Model 4 pretending that it is a Model III with hard disc drive attached. It worked. What else is there to say ?

Unfortunately, when in Model III mode, the Model 4 CPU is limited to the Model III standard speed of 2 Mhz, or at least so the specification says. With Neil's help, I have Model III software running with the CPU speed of 4 Mhz, although the actual processing speed is restricted to 3 Mhz by the slow speed of the ROM. Even so this is 50 percent up on the standard Model III, and comes at no extra cost.

Conversion from the Model I or III

For LDOS users the Model 4 can read your discs immediately. A feature which I have not tested, but believe to be true.
(You earned it by learning LDOS in the first place).
For the rest of us, file conversion from Model III TRSDOS is provided by a utility called CONV which will convert Model III TRSDOS discs to Model 4 format - if the Model III mode will not do for you. (Model III TRSDOS was/is unique to itself). Conversion from Model I or other Model III systems is by a utility called REPAIR, which converts the disc director[illegible]o be readable by

TRSDOS 6.0. Files may then be used or copied.
These utilities are supplied and carry out file or directory conversion only. Neither will convert your program code to Model 4 Basic format, and for obvious reasons, neither can translate peeks/pokes or screen handling, but then you must have some fun left for yourselves. It may be worth repeating that no conversion of Model III software is essential, it will run in Model III mode as is is (peeks/pokes and all). Conversion of Model I software should be well known by now, and needs nothing more than for the Model III.

In summary, YES PLEASE, I want one. I shall be converting to the Model 4, which I shall use as a fast Model III until software arrives to take advantage of the extra ram available in the Model 4 mode. I do not expect this to be long.
Whilst I am at it, I shall also have an Os carrying case to move it about in safely.

Leo Knaggs.

MODEL III IN BUSINESS — JOHN JEFFERY

****

Like Julian we purchased our Model III about eighteen months ago,for use in our Business.We had previously used a 16k model one(level II).Originally I wrote the Business Programm in Basic for the Level Two and it consisted of six modules linked together by courtesy of a Kansas City Programme called PROGRAMMER.It was Menu driven and served us very well for a while.

However with the advent of Model III and VISICALC I immediatly set about recasting the entire Financial System.The major problem was not the Computer or the Programme,but the Principles of Accountancy which had been left to our Secretory for several years.Initial forays into this field produced either progressive bankruptcies or made us paper millionaires:indeed any resemblance between the balances in our Books and in the Bank was purely coincidental.

Our initial aim was to produce weekly accounts by a combination of the Simplex Account Book and our standard trade VAT Book.I might add we call ourselves Dairymen (what our customers call us is not for publication) and we run a Family Business of three Milk Rounds,with only two employees plus half the local schoolboys.Again as we work seven days per week (we get Christmas Day off),we (that means me) have to do the Books either afternoon or evenings.

This means we must have a system that does not require much time,and we find that using Visicalc (and Scripsit) we catch up on any backlog of work.Indeed not only are we now almost up to date,but have gone back over a previous years accounts and done them (in the process exposing the most glaring arithmetical and positional errors) using Visicalc which can add up if instructed correctly.

Whilst Visicalc is indeed a very powerful accountancy tool and shows great versitility in operation,we really require a full understanding of the DIF function in order to access the discs to provide a full years summary of the accounts.With this in view we are still awaiting the latest release from Tandy of the enhanced version which was promised as long ago as March,it is now June.Telephoning Walsall has produced only promises of "it will be in the post tonight";this refers to the instructions for dealing with the DIF files.

However to get back to our system, it is integr[illegible] to the extent that a figure is entered only once and is then processed on its way down the sheet. For example consider the purchase and sale of goods bearing VAT, both transactions occurring in the same week. The number of articles purchased is entered in the appropriate column and the purchase price appears at the side; at the bottom the total price of goods plus VAT and of VAT is shown. This number is then transferred to the Sales Section which shows total selling price and the amount of VAT to be paid. In addition it is loaded into a section marked Vat Analysis Summary which indicates into which Box Number the amounts should be entered on the VAT Sheet.

Finally in general we are pleased with our Model III and look forward to developing our system further.

TRS - 80 MODEL 4

1. The Model 4 can run all Model III programs without change.
2. The Model 4 has a new TRSDOS and Microsoft BASIC of its own.
3. The Model 4 has LDOS compatability
4. The Model 4 has CP/M capability.

With the Model III BASIC held in ROM, compatability with existing software is built in, providing immediate access to a large range of working programs for all purposes.

The new TRSDOS 6.0 includes features such as

- Memory Disk, which simulates a disc drive in memory giving super fast access.
- a Job Control Language for automatic action of a sequence of operations, which includes Sound Alerts.
- a Complete communications program for the growing world of data access
- a Print Spooler allowing print of one job results whilst doing another
- Forms Control to give better control over the printed page
- Input and Output Routing to connect different sources and destinations of information, including links and filters.
- Compatability with LDOS files From Model III or Model I

And more.

The new Microsoft Disc BASIC includes

- Access to an 80 chr x 24 line screen (which can be 40 x 24 if required) with full access through PRINT @,
- CHAIN and COMMON commands for large programs, Variable names up to 40 characters long,
- A check for 'Endless loops', WHILE...WEND structure for continued operation until a condition changes,
- A sound feature with control of tone and duration, Even print line width setting.

Model III BASIC and MODEL 4 BASIC programs are not interchangeable, but they will BOTH run on the same machine in Model III or Model 4 modes.

The operating Mode of the machine is governed by the disc used to 'BOOT' it into life. Available NOW are TRSDOS 1.3 for Model III applications, TRSDOS 6.0 for Model 4 applications, and comming soon is CP/M Plus to give access to the range of programs available for this operating system.

FOR SALE.

COMPLETE SET OF 10 COPIES OF WAYNE GREEN'S "ENCYLOPEDIA FOR THE TRS-80", SOFT COVER, IN MINT CONDITION. £26 inc. post.

Telephone David Law, LUDLOW (0584) 4338 evenings.

Technical

Processor	Z-80A at 4 MHz (Model 4) or 2 Mhz (Model III mode)
Memory	Rom - 14K. Ram - 16 to 64K Ram, depending upon model, expandable to a maximum of 128k in all cases.
Keyboard	70 key typewriter style, incl 3 programmable function keys, Capitals lock key, Control key and numeric pad.
Display	80 x 24 (40 x 24) or 64 x 16 (32 x 16), Upper/Lower case, reverse, 96 text, 64 graphic, 96 special chrs.
Discs	Std = 0-4 drives, 40 track, double density, Single side 184K - Special = Please ring to discuss your needs.
Cassette	Operation at 500/1500 baud in Model III mode.
Sound	Including tone and duration controls, with Keyclick available.
Printer	Parallel interface (Compatable with all previous models).
RS-232C	Serial interface available or supplied fitted.
Size	12.1/2" x 18.7/8" x 21.1/2"

On House

UPGRADING MODEL I (LEVEL II) 16K — JOHN JEFFERY

Those of us who have a Model I, whilst being reluctant to get rid of it find it extremely difficult to upgrade it in such a way as to be able to interchange discs and programmes with Model III. What is required is a cheap way to increase the memory to 48K, to read and record discs in double density (as in III) and a cheap RS232 interface. All these would appear to be available.

Audio Computers of Southend produce a 32K add-on memory board - 46pounds inclusive of post and Vat. This is only one of their products, one can obtain a 128K Ram Card with P.I.A. (peripheral interface adapter) built in for around 120 pounds. Instructions for fitting are provided and although this involves some cutting of wires and soldering some members have successfully fitted them (P9 Nat.User Group/July 1982).

The Exatron Memory Expansion is a U.S. Import produced by the makers of Stringy Floppy. This consists of a 64K RAM which overlaps all ROMS, and is in one continuous block, but the sections covered by ROMS are not used. As this is a U.S. import there are Air Mail charges and Customs and handling charges. (NATGUG March83/P15)

A separate add-on Disc Controller is in the pipeline from the same firm, this may be useful if it provides double density compatible with Model III.

In the case of the RS232 Interface I see one is available at 10.50pounds from E.Hartley, 91 Quebec Road, Blackburn. BB2 7DP., who also has lower case kits at 15.50inc. With the RS232 it would appear possible to link the Model I with the Model III via the Maplin Modem which we are constructing.

To come back to my first point - is there a cheap and efficient way to increase the power of Model I to at least the power of Model III, to enable software to be freely interchangable; and if so is it the way I have indicated or are there some horrible snags lying in wait. Any solutions to the problem eagerly explored, and I may be buying your old Model I's.

11 ROWAN DRIVE,
KILBURN, DERBYS.
DE5 OPG.

TEL DERBY 881239

SORTING TABLES PREPARED ON A WORD PROCESSOR

My research frequently involves sorting out tabular, or tabulatable, material and getting the results out onto paper. The data is mainly string or non-decimal numeric (e.g. acres, roods, and perches). The tables are not long - anything from 30 to 300 lines - but the number and width of columns varies much. Generally I need to sort by several columns independently, but sometimes I need a multi-key sort on two or three columns. I imagine that other members may often have a similar need, so may be interested in the methods used. Having done such tasks by hand for years, I decided it was time to try sorting on my Model III. There seemed to be three choices:

1) A commercial data-base program. This would be expensive for the job and need elaborate file definition for each table: moreover, data entry record by record is slow.

2) A separate program for each table. Too time-consuming.

3) Preparation of tables on a word processor and sorting them with a general-purpose Basic program. Data entry would be quick and easy, and the whole file could be scanned for errors before sorting. Unfortunately, I knew of no such Basic program; there are commercial sorting utilities, but most seem to be designed for interfacing with Basic for each application. I therefore wrote my own, using the CMD"O" indirect sort available on Newdos-80. At the end of this note is the current, and for me final, version.

SYSTEM REQUREMENTS ETC.

TABLSORT/BAS occupies about 1.7K when compressed and was written for use with Scripsit and NEWDOS/80 on the Model III. It should run on Model I NDBB, and can be adapted for 32K by changing the CLEAR statement in line 10. I gave up trying to write a version for TRSDOS 1.3 when the variable-length-record files wouldn't work. It could be done with sequential files. You can use any word Processing Program which:

1. has a tab function;
2. can save and load files in ASCII;
3. Permits a forced carriage return, stored as CHR$(13) -0DH- in such ASCII files;
4. uses a single byte as an EOF marker.

The Program uses a single-key in-memory sort: the whole table is sorted by the key field and output can be in ascending or descending order. The NEWDOS multiple-key sort would require either a fixed number of columns (fields) in all tables or a line-generating routine, which is too slow. Moreover the array Pointers would take up too much memory. If you need a multiple-key sort, you can either use Scripsit to Prepare a DO file which will run the Program repeatedly, or group fields for sorting, or both. The maximum line (i.e. record) length is 255, and, subject to that, the maximum number of columns (fields), and the maximum field length, are both 255. The maximum number of lines (records) will vary according to the line length and the number of fields. The following short routine calculates it (rather cautiously: you might get more in), and also tells you whether to alter the CLEAR statement in

line 10 of TABLSORT:

```
5 REM routine for calculating maximum no. of sortable recs.
10 INPUT"NO OF FIELDS?";F
20 INPUT "RECORD LENGTH";L
30  N=INT((36357-8*F-2*L)/(L+11)):IF((N+2)*L+24)>32767  THEN
N=INT((32748/L)-2)
40 PRINT"MAX NO OF RECORDS IS ";N
50 PRINT"STRING SPACE IS";(N+2)*L+24
60 GOTO 10
```

PREPARING THE TABLES

This is the tricky bit. The program requires that all lines in a given table are of equal width and all columns of equal length; each column has a standard width including trailing spaces (entries are left-justified on the column). Each column must have a heading on the first line. To determine field lengths required, scan each column of your draft table to determine the item of maximum length and add 1 (a trailing space for clarity). Add a further 1 to the right hand column, and write down the results (TABLSORT needs to know them). Then set your word processor tabs to correspond with the cumulative field lengths, and add a tab for the last character of the last field. That will be a carriage return, which is included in the field-length count. If you are using Scripsit, the tab settings give the number of characters to the left of the tab. Thus if you are sorting 3 columns, and the longest items in each column are 19, 19, and 18 characters respectively, then the field

lengths are 20, 20, and 20 and the tab settings 20, 40, and 59. Having done this, type in your headings and then tab out to the last tab position and press ENTER; then type in the second line and so on. The carriage return must always come at the last tab position, NOT immediately after the last text character. When finished, the file should contain no print format codes or anything but the table, and must be saved in ASCII format.

SPECIAL PROBLEMS: NUMBERS, LONG LINES, AND LONG HEADINGS

Scripsit ASCII files produce string data, and TABLSORT sorts string arrays. Numeric data are treated as strings and sorted in ASCII order. This is OK if you remember that 12 in ASCII is less than 8, but 08 is less than 12: i.e. ensure that in columns of numbers the decimal point is always the same distance from the left margin of the column, and pad out any gaps with leading zeroes. You can always edit them out of the file after sorting.

Scripsit's maximum line length is 132. If your tables have longer lines, split each line into two. Set the paragraph indent to 1, and use the paragraph marker to end the first line and the carriage return to end the second line. The paragraph marker must be the last character of the last field on the first line, so set a tab point for it and include it in your field-length count. Add 1 to the next field length to allow for the indented space.

If the heading of a column won't fit on one line, put the extra on line 2, indenting it by one space from the tab position. Don't forget to put a carriage return on the last

tab Position of that line.

USING THE PROGRAM

Set BASIC 1V to obtain the maximum memory. The Pro9ram will ask you for a file name and will check the record len9th. It then asks you for the number of fields and the len9th of each. (At this Point you can decide to 9rouP fields, either to save keyboard inPuts or to obtain a multi-key sort of adjacent columns. SuPPose your table had 7 10-character columns, and you want to sort by columns 3, 4, and 5 in that order of Priority. You enter 5 as the number of fields, and 10, 10, 30, 10, 10 as the field len9ths.) The Pro9ram then disPlays the headin9s (or as much of them as fits on the first line) and asks you which you want to sort by. If you have so many columns that the headin9s scroll uP the screen, try Puttin9 a comma before the last colon in line 180 of the Pro9ram. Finally, outPut can be in ascendin9 or descendin9 order, and can be to screen (no scroll Protection, but useful for a dummy run), Printer, screen and Printer, or disk (for reloadin9 into ScriPsit). The last record of the file consists solely of SCRIPSIT's EOF marker. TABLSORT replaces that with a carria9e return in the sorted file.

Now for the Pro9ram:

```
5 REM TABLSORT/BAS
10 CLEAR 32767:
DEFINTA-Z:DIMN,DD,CH,NF,MA,RL,DD$,KF,NM$:DIMSF(3)
20 CLS:LINE INPUT"NAME OF FILE TO BE SORTED? ";NME$
30 OPEN "I",1,NME$
```

```
40 LINEINPUT#1,LE$:RL=LEN(LE$)+1:CLOSE:CMD"F=ERASE",LE$
50 INPUT "NO. OF
FIELDS?";NF:DIMFA$(-(NF*(NF>2)+3*(NF<3))),LF(NF),TITLE$(NF)
60 CHECK=0
70 FOR N=1 TO NF
80 PRINT"LENGTH OF FIELD NO.";N;:INPUT
LF(N):CHECK=CHECK+LF(N):NEXT
90 IF CHECK>RL PRINT "TOTAL FIELD LENGTHS EXCEED LINE
LENGTH.";:ELSE IF CHECK<RL PRINT"TOTAL FIELD LENGTHS LESS
THAN LINE LENGTH";
100 IF CHECK<>RL THEN PRINT "TRY AGAIN":GOTO 60
105 CLS:PRINT"FILE OPEN. DON'T REMOVE DISK"
110
OPEN"R",1,NME$,RL:MAX=LOF(1)-1:DIMA$(MA),B$(MA),B(MA),C$(MA)
:
120 REM FIRSTFIELD
130 DD=0:FOR N=1 TO NF:FIELD 1,DD-0 AS DD$,LF(N) AS
FA$(N):DD=DD+LF(N):NEXT
140 REM GET HEADINGS
150 GET 1,1
160 FOR N=1 TO NF:TITLE$(N)=FA$(N):NEXT
170 REM ASK FOR KEY FIELD
180 PRINT:PRINT"SORT BY:":FOR N=1TONF:PRINT N;"
";TI$(N):NEXT
190 PRINT "ENTER NUMBER OF YOUR CHOICE (1 TO ";NF;")":INPUT
KF
200 REM REFIELD BUFFER
210 SF(1)=0:SF(2)=LF(KF):SF(3)=0
220 IF KF=1 THEN SF(3)=RL-LF(KF)ELSEIF KF=NF THEN
```

```
SF(1)=RL-LF(KF) ELSE GOSUB 380 'SUB NEWCOLS
230 CMD"F=ERASE",LF( ):DD=0
240 FOR N=1 TO 3
250 FIELD 1,DD-0 AS DD$,SF(N) AS FA$(N):DD=DD+SF(N):NEXT
255 REM READ IN TABLE FROM DISK
260 FOR N= 2 TO MA
270 GET 1, N
280 A$(N)=FA$(1):B$(N)=FA$(2):C$(N)=FA$(3):NEXT N:CLOSE
290 REM SORT
300 PRINT"SORTING":CMD"O",MA,*B(1),B$(1)
310 REM PRINT
315 CLS
316 PRINT"OUTPUT IN:1 ASCENDING  2 DESCENDING ORDER"
317 INPUT"ENTER NUMBER";T:T=T AND 1:PRINT
320 PRINT "OUTPUT TO: 1 SCREEN 2 PRINTER 3 SCREEN & PRINTER
4 DISK"
330 INPUT "ENTER NUMBER";N
340 ON N GOSUB 400, 440, 460, 480 'RT1,RT2,RT3,RT4
350 END
360 REM SUBROUTINES
370 REM SUB NEWCOLS
380 FOR N=1 TO
KF-1:SF(1)=SF(1)+LF(N):NEXT:SF(3)=RL-SF(1)-LF(KF):RETURN
390 REM SUB RT1
400 FOR N=1 TO NF:PRINTTITLE$(N);:NEXT
410 IFTTHENFOR N=1 TO MAELSE FOR N=MA TO 1 STEP-1
420 PRINTA$(B(N))B$(B(N))C$(B(N));:NEXT:RETURN
430 REM SUB RT2
440 CMD"ROUTE DO PR":GOSUB 400:GOSUB 540:RETURN
```

```
450 REM SUB RT3
460 CMD"ROUTE DO DO PR":GOSUB 400:GOSUB 540:RETURN
470 REM SUB RT4
480 LINE INPUT "FILENAME OF NEW FILE? ";XF$
490 OPEN"O",1,XF$
500 FOR N=1 TO NF:PRINT#1,TITLE$(N);:NEXT
505 IFTTHENFOR N=1 TO MAELSE FOR N=MA TO 1 STEP-1
510 PRINT#1,A$(B(N))B$(B(N))C$(B(N));:NEXT
515 PRINT#1,CHR$(13);
520 CLOSE:RETURN
530 REM SUB RT5
540 CMD"ROUTE DO DO":RETURN
```

Christopher Currie

ND/80 v2.0 MERGE, RUN-V, DO files.
Merge is used for my labelling to members every month-100% record.since'82
RUN-V is tremendous for modular, structured programming .
Do files are fine, & I use DOSPLUS for building them, for use with ND/80.

Brian Pain.

FOR SALE

48k TRS-80 Model I, with twin disks & Video. Will consider different micro as part exchange. £650.
Peter Wood Milton Keynes 0908 564863

MODEL 100 & MODEL 4

Reviews & Comments please.

SOLOLOAD

Mr A.R.D. Smithard of West Sussex is using one of the above units.

FOR SALE

New Epsom MXF/T1
2 Off
Ring Brian Pain

Isleworth Area

E.J. Grigg would like meet others in this area
Contact him on 01 560 2405

SPEECH SYNTHESIS USING THE G.I. SP0256-AL

Using the General Instrument SP0256-AL chip a simple speech synthesiser can be built for around £20. It works fine on my Model I, and should work on the Model III, with some minor changes to the software.

The SP0256-AL can be obtained from Watford Electronics. The crystal is only available from Semiconductor Specialists (089 54) 46415, as far as I know. Although they are a large distributor, they do not seem to mind small orders.

The SP0256-AL is similar to the Votrax SC-01 in that it synthesises speech by generating phonemes. Thus any word can be produced, you just output the appropriate sequence of phoneme codes to the device.

In fact, the device uses allophones, that is, phonemes which sound alike. There are about 64 allophones in the English language, so the data rate is very low, and can easily be handled by simple BASIC programs.

Full details on how to program the device are in G.I.'s publication, VSM2128 Voice Synthesis Module User's Guide, obtainable from Semiconductor Specialists.

The following test program will output all the allophones in turn:-

```
10 FOR I=5 TO 63
20 OUT 0,I
30 IF INP(0)=254 THEN 30 'WAIT UNTIL LRQ GOES HIGH
40 NEXT I
```

The following program converts a file containing allophone codes into appropriate numerical codes. A sort of speech assembler! It is rather slow, as it uses a sequential search technique, but this does not really matter.

```
10 '                    * * * SPEECH/BAS * * *
20 '
30 'PROGRAM TO CONVERT ALLOPHONES IN SYMBOLIC FORM
40 'TO NUMERICAL CODES.
50 '
60 'VERSION 1.0 - 29/5/83
70 'WRITTEN BY LEON HELLER
80 '
90 CLS
100 DIM A$(64),C(64)
110 FOR I=0 TO 63
120 READ A$(I),C(I)
130 NEXT I
140 INPUT "SOURCE FILE";N1$
150 INPUT"DESTINATION FILE";N2$
```

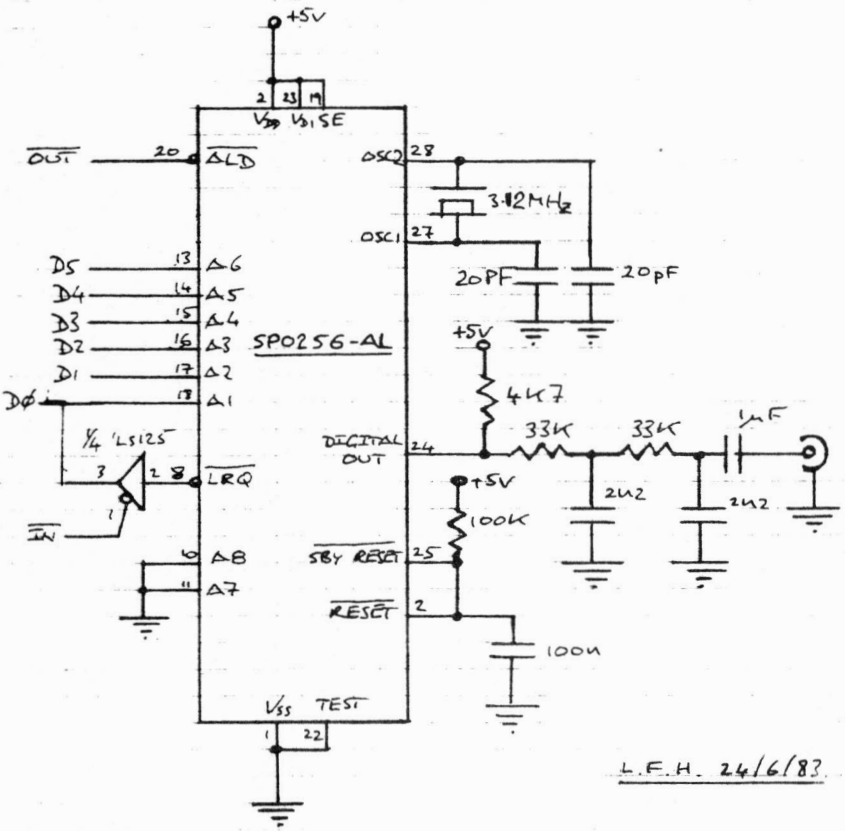
```
160 OPEN "I",1,N1$:OPEN "O",2,N2$
170 IF EOF(1) THEN 290
180 INPUT£1,C$
190 IF C$="" THEN 290
200 N=0
210 IF A$(N)=C$ THEN 260 ELSE 220
220 N=N+1
230 IF N>63 THEN 240 ELSE 250
240 PRINT "CANNOT FIND CODE: ";C$
250 GOTO 210
260 PRINT A$(N),C(N)
270 PRINT£2,C(N)
280 GOTO 170
290 PRINT£2,64
300 END
310 '
320 'SYMBOLIC AND NUMERICAL ALLOPHONE CODES
330 DATA AA,24,AE,26,AO,23,AR,59,AW,32,AX,15,AY,6
340 DATA BB1,28,BB2,63,CH,50,DD1,21,DD2,33,DH1,18,DH2,54
350 DATA EH,7,EL,62,ER1,51,ER2,52,EY,20,FF,40
360 DATA GG1,36,GG2,61,GG3,34,HH1,27,HH2,57,IH,12,IY,19
370 DATA JH,10,KK1,42,KK2,41,KK3,8,LL,45,MM,16
380 DATA NG,44,NN1,11,NN2,56,OR,58,OW,53,OY,5
390 DATA PA1,0,PA2,1,PA3,2,PA4,3,PA5,4,PP,9
400 DATA RR1,14,RR2,39,SH,37,SS,55,TH,29,TT1,17,TT2,13
410 DATA UH,30,UW1,22,UW2,31,VV,35,WH,48,WW,46
420 DATA XR,47,YR,60,YY1,49,YY2,25,ZH,38,ZZ,43
```

This program reads numerical allophone codes from a disk file, and then outputs them continuously to the synthesiser.

```
10 '                        * * * TEST2/BAS * * *
20 '
30 'TEST PROGRAM FOR SP025-AL SPEECH SYNTHESISER BOARD
40 'VERSION 1.0 - 29/5/83
50 'WRITTEN BY LEON HELLER
60 '
70 DIM A(100)
80 CLS
90 INPUT"ALLOPHONE FILE";N$
100 PRINT
110 OPEN "I",1,N$
120 N=0
125 IF EOF(1) THEN 190
130 INPUT£1,A(N)
150 IF A(N)=64 THEN 190
155 PRINT A(N);
160 N=N+1
170 GOTO 125
180 '
190 N=N-1
200 PRINT:PRINT:PRINT "PRESS ANY KEY TO STOP"
210 FOR I=0 TO N
220 K$=INKEY$:IF K$="" THEN 230 ELSE 300
```

```
230 OUT 0,A(I)
240 IF INP(0)=254 THEN 240
250 NEXT I
260 OUT 0,0
270 FOR J=1 TO 500:NEXT J
280 GOTO 210
290 '
300 END
```

Leon Heller



FOR SALE

TRS-80 16k LII, Expansion Interface 32k, Sekosha GP80 Printer, Cassette Lots of Software. £450 ONO or will split
Mike Russell 0908 604398

FOR SALE

Gold plated edge connectors
40 way £3.20 34 way £3.20
Printer cable kits also available.
Richard Creak Danbury (024541) 3725

CAMEL MICROS

33a COMMERCIAL ROAD, WEYMOUTH, DORSET, U.K.

Tel.: (03057) 70092

SOFTWARE
SPECTRUM and TRS80 Mod. 1/3

GRASP

General purpose graph plotting program with facilities for scale change, axis inversion, histograms or graphs, multiple graphs, and screen dump to MX80 printer with graphics, etc. Full manual included. SPECTRUM version has full colour options, high res. plots, function and transforms generators, image ghosting, over 50 different options. Requires 48K Spectrum.
Price £6.50

FILEMASTER

For TRS80 disc users only, this program can map almost any disc file format (fixed record size, variable record size, part fixed part variable, or serial files) into any other. Data on two input files can be merged to the output file, and arithmetic relationships can be set up. Ideal for linking data across programs, or producing listing files or command fales to drive other programs. Capable of producing automatically, for example, a file which can be listed on a printer, used in Scripsit, or used as a command file, direct from a Visical /DIF file. All redundant data is stripped, full control over format of listing is possible, and titles (or commands) etc. can be included horizontally or vertically. Also has a manual mode for examining file structure, and manually extracting data onto a different file.
Price £11.90

SMUGGLERS COVE

Brand new Adventure game, 48K TRS80 only. Looks impossible until you spot the obvious, we'll say no more! There is more than one way to win, some more elegant than others. Machine coded for speed, with over 30 locations, 70 objects, and 50 verbs.
Price £5.50 (on cassette).

All prices include VAT and postage.

NATIONAL TRS-80 & GENIE USERS GROUP

The group offers support on the following Tandy micros:-
a)Model 1. b)Model 2. c)Model 3. d)Model 4. e)Model 16. f)Model 100 and the Genie 1,2 & 3 from Lowes.

Back Issues, software library lists and publications are available. Back numbers of specialist journals are also kept and the group offers access to these at various meetings.

The group publishes a monthly newsletter, holds meetings on a variety of topics, and can put members in touch with others who may have an answer to a particular problem.

Chairman Leon Heller Secretary Brian Pain

I would like to join/renew my subscription to the Group(Please delete).

Surname:.....Initials:.....

Title:.....

Address:.....

.....

.....

Postcode:.....Telephone:.....

Equipment:.....

DOS (if applicable).....WP:.....

Interests: Business....Scientific....Educational....
Hobbyist....BASIC....Machine Code....
Medical....A/Radio....W/Proc....
Fin Mods....Database....Comms....
Hard DIY....Soft DIY....Langs....
Games....Graphics....
Others.....

Volume 5 Issues	7 - 12	Jan'84 - Jun'84	7.00 ....
5	1 - 6	Jul'83 - Dec'83	7.00 ....
4	1 - 12	Jul'82 - Jun'83	14.00 ....
3	1 - 12	Jul'81 - Jun'82	14.00 ....
2	1 - 12	Jul'80 - Jun'81	12.00 ....
1	1 - 12	Jan'79 - Jun'80	10.00 ....

Assembler Notes I by Laurie Shields	1.49 ....
do do II do do	1.49 ....
the above are for beginners.	
Modifications to the V/Genie I by Ken Robinson	1.49 ....
Disk File handling Model I & III by Jim Norris	1.49 ....
Notes on CP/M for beginners by Leon Heller	1.00 ....
Maths I & II programs on Disk/Cassette	1.00 ....
Software List for Model I	1.49 ....

Enclosed -
Cheque:.....(Payable to NATGUG)

Access:.....Expires:....

Barclaycard:.....Expires:....

Send to Brian Pain, 24 Oxford Street, Stony Stratford, Milton Keynes. MK11 1JU
(0908 - 564271)

MILTON KEYNES WORKSHOP 29/30/31 JULY 1983
YORK HOUSE, STONY STRATFORD, MILTON KEYNES

FRIDAY 2.30 - 5.30 CP/M - COLIN ASHFORD (01-462-7296)
An Introduction to its Good & Bad Features.
An overview of commercial & Public domain Programs in a business environment.

6.30 - 9.30 CP/M & WORDSTAR LEON HELLER (0908-613004)
Comparisons with Tandy wordProcessors.
Essential Features to understand.
Interactive "Tools" - MAILMERGE & SPELL CHECKERS.
Going through a hands on session.

The above is being covered as so many business users are now moving over to CP/M, and require assistance.

SATURDAY - ROOM 1
10.00 - 1.00 DOSPLUS PETER TOOTILL (051-428-2733)
1. Main Points - 1 hour.
2. In depth review.

2.30 - 5.30 LDOS Alan Redhouse (07593 - 502)
1. Main Points - 1 hour.
2. Getting inside - eg hooking 8" drives on.

7.00 - 1.00 CORPLAN BUSINESS GAME.
12 decisions with a minimum of 3 teams required.

SATURDAY - ROOM 2
2.30 - 4.30 Writing I/O Routines without DOS
John Newgas (01-539-7071)
4.30 - 5.30 MULTIDOS John Newgas
1. Main Points.
6.30 - 9.30 DISK PROTECTION John Newgas.

SUNDAY ROOM 1 10.30 - 12.30 & 2.30 - 4.30
NEWDOS/80 V2.0 & the BASIC Programmer
Leo Knaggs (0258 - 53737)

ROOM 2 11.00 - 1.00 & 2.30 - 4.30
NEWDOS/80 V2.0 & the Machine Code Programmer
Laurie Shields (0246 - 75568)

---

* Free Coffee will be available over the weekend
* A map will be sent on receipt of the booking form.
* York House is 5 miles from the M1 & is on the A5.
* There is a direct bus from Central Milton Keynes.
* There will be a barbeque on the Sat. evening .
* There is space for campers in the buildings grounds.
* Thee building has been booked from Friday morning to Sunday evening. It will not be closed & we are the caretaker!
* Os house will be in attendance, with his own workshop.
* Leighton Davies, our long-standing librarian, will be there in his own room.
* Members who do not book, will be placed in a secondary position in the systems area. This step is essential after the Jan meeting.
* About 3 years supply of 80-MICRO will be available to members.
** As an incentive to members who live some distance away,
** Please deduct 20% from lecture fees if over 100 miles from MK,
** and 40% if over 200 miles. There is a 'special' for over 500!

NATIONAL TRS80 & GENIE USERS GROUP
BOOKING FORM - FOR THE WEEKEND 29/30/31 JULY 1983

1.	Friday Lectures only	5.00
2.	Saturday Lectures only	3.50
3.	Sunday Lectures only	2.50
4.	Friday Lunch	2.50
5.	Saturday Lunch	2.50
6.	Sunday Lunch	2.50
7.	Friday Supper/Tea	2.00
8.	Saturday Supper/Tea	2.00
9.	Fri Evening & Sat. Morng B&B Hostel	5.00
10.	Sat Evening & Sun. Morng B&B Hostel	5.00
11.	Fri Even. & Sat. Morng B&B S/Stratford	11.50
12.	Sat Even. & Sun. Morng B&B S/Stratford	11.50
	Total :-	=====

To NATGUG, c/o Brian Pain, 24 Oxford Street, Stony Stratford, Milton Keynes. (0908 - 564271)

From :.....

Address:.....

.....

.....

Telephone :.....

Please find enclosed cheque for :.....

Please charge my ACCESS/BARCLAYCARD ACCOUNT

.....EXPIRES .....

FUTURE WORKSHOPS

AUGUST 6/7 London off Baker Street
BASIC in a Business Environment
For the beginner & intermediate.
There will be no game/graphic content.
A specific syllabus will be available.
Around 15 are needed to make this viable,
with a cost of around 25 pounds.

AUGUST 13/14 Venue as above
MACHINE CODE beyond Laurie Shields I & II
This weekend will be for those who have done
some work but would like to go further.
Again a specific syllabus will be drawn
up and the cost will be around 25 pounds.

AUGUST 21/22 Venue as above.
All weekend on WORD PROCESSING
Model 1,2,3,4 with HARD disk, and 3-4
printers. Interfacing WP to VISICALC, plotters,
Database and running over WORDSTAR, SCRIPSIT &
SUPERSCRIPSIT, NEWSCRIPT & WORD4WORD. Geoff Smith
on getting inside SCRIPSIT, Derrick Rowe on
interfacing WP to database, Leon Heller on
WORDSTAR & lots more.
Again the cost around 25 pounds.

SEPTEMBER PCW SHOW & 1 Day on the MAPLIN MODEM with ACC.

NOVEMBER 18/19/20 SWINDON.

DECEMBER 17/18 or January 7/8
London off Baker Street.
Model 16 and its operating systems
Jim Norris, Derrick Rowe & Leon Heller
will take members through the M16 in true
workshop style. We expect considerable support
on this from members & TANDY themselves.

FEBRUARY 1984
Milton Keynes at Woughton Campus OR
the original campus now taken over by BRIT. TELECOM.

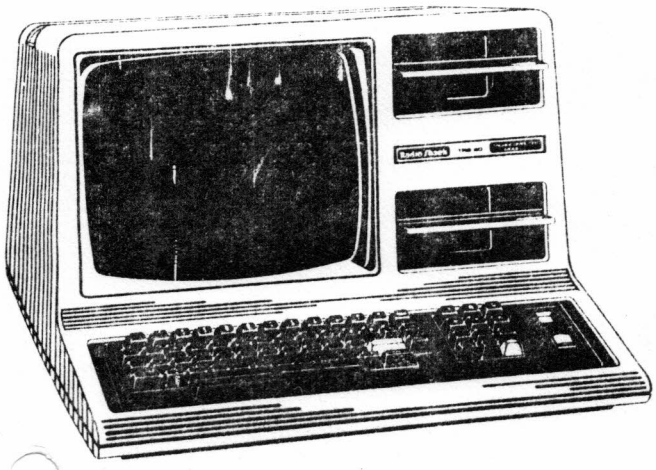
********** DISCOUNTS FOR THOSE TRAVELLING OVER 100 MILES **********

BLANDFORD FORUM SEPTEMBER 4 BLANDFORD FORUM
SEPTEMBER 4 BLANDFORD FORUM

BLANDFORD COMPUTERS

Tandy®
AUTHORISED DEALER

THE COMPUTER DIVISION OF O. & J. HOUSE LIMITED



NEW

TRS-80 Model 4

In stock NOW!

NEW "BLANDFORD CARRYING CASE FOR MODEL 4 AND III

Powerful Model 16 **Model 12**

disks £18 For 10

ALL PRICES INCLUDE VAT!

Disk Drives FROM £165 **HARD DISK SYSTEM**

SOFTWARE

Books BACK IN STOCK:-
"MACHINE LANGUAGE DISK I/O AND OTHER MYSTERIES

REPAIRS - UPGRADES - PART EXCHANGES

SECOND HAND TANDY EQUIPMENT AND SOFTWARE BOUGHT AND SOLD

Save -USER GROUP-QUANTITY DISCOUNTS

HIGHER SHAFTESBURY ROAD · BLANDFORD FORUM · DORSET · DT11 8SS

TEL. BLANDFORD 53737 (STD 0258) 5 LINES INCLUDING EVENINGS & WEEKENDS