NATGUG NEWS

Volume 6 Issue 1 July 1984

OFFICIAL JOURNAL OF THE

National TRS-80

& Genie Users

Group.

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.

The Group maintains two software libraries (Models I and II) which are free to members. Library lists are obtainable from the Secretary.

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

CONTENTS

1. Information on the Group
3. Editorial
3. Guidelines for contributors
3. Library information
4. Members' letters
6. Problem section
7. Booting with drive 0 write-protected using LDOS
8. TRS-80D ???
9. Fitting instructions - High Res. Graphics kit
10. LeScript - a review
19. Floating point accuracy
20. Negative INTegers
20. Compound conditional statements
21. For sale and wanted
22. Review of labelling program

Minor notes from the secretary.

Blandford Forum 19th August 1984

PCW show - Hammersmith September 1984

Swindon Workshop November 1984

Lake District 25/26th August '84

Dunoon ?? September '84

Pyrenees July/Aug '85

Brian Pain.

EDITORIAL

My pessimistic comments about the future of NATGUG seem to have stung one or two of you into activity, resulting in at least one very interesting article that would not otherwise have been produced!

My thanks to Richard Marks for preparing a cumulative index going back to the first issue. We are publishing this separately, so that you do not need an index to find the index, if you see what I mean.

As I write this, we are getting ready for the Milton Keynes workshop, which promises to be a very interesting one, as it will be the first real combined IQLUG and NATGUG workshop, with a reasonable number of QLers prsent. A new concept is my course on the MC68000 processor, which has attracted a lot of interest from both groups, as well as several people who belong to neither group, but wish to learn about this powerful (and surprisingly easy to use) device.

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 hand-written material, write legibly.
6. Do not fold the sheets when posting them. Use a large envelope and keep them flat.

Leon Heller

LIBRARY INFORMATION

Model 1: Leighton Davies, 105 Caerau Road, Caerau, Glamorgan. Tel: (0656) 738337

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

MEMBERS' LETTERS

Congratulations on getting two Newsletters out in three weeks - from a reader who's got no complaints about them or the Group. I can't see that a society with still nearly 500 members is not viable, and I hope it won't fold soon. I couldn't afford another computer, and it would cost me a fortune in hardware & software to get a new system to equal the performance of my Model III for the things I want to do. It's often much faster in practice, though slower in theory, than the 16-bit machines we have in the office - and far more flexible.

I was interested to see comments on PROPASCAL in both newsletters. I've been using it for eight months on a DEC Rainbow, and I find I use it less and less. The time taken to get a short program compiled and debugged is far longer than with Basic, and the string handling facilities are poor. You can't set up a string array - you have to fool the compiler into thinking it's handling an array of record type. You can't use bit setting on strings, as in Basic, and the chr and ord functions (roughly equivalent to CHR$ and ASC in Basic) don't work with dynamic strings. There's no equivalent to the Disk Basic MID$ function; to replace a substring you have to delete the old substring and then insert the replacement at the correct position, which can be a little tricky. The manual doesn't even explain how to output to the line-printer! The procedure, which involves setting up a dummy file, is pretty cumbersome; I might be able to help anyone who gets stuck with it.

I can see that the linking facilities would be useful to a professional programmer who wanted to create his own library for long and complex programs. But an amateur needs to create such a library in order to get quite elementary programs to work. I find I haven't the time.

Has any member tried using BASICODE on the TRS-80? For Model I users it's a bit of a fraud - you can't use it without building your own interface, for which the manual provides a largely illegible circuit diagram, no parts list, and an address in the Netherlands from which you can order the printed circuit board if the docks aren't on strike.

The manual doesn't make clear that this applies only to the Model I - although I guessed from the references to expansion interfaces and 40-pin data buses that that was so, and tried loading the tape on the Model III (you have to use 500 baud).

It turned out that the Model III needs no interface (I suppose the problem must be something to do with the cassette baud rate). The sample programs wouldn't work, however, either under Newdos80 or using the Model III Basic with DOS disabled: the translation is supposed to supply the standard Basicode subroutines (lines below 1000) for the TRS-80, but didn't. I wrote my own subroutines from the Protocol in the manual, and most of the programs now run adequately.

I should be grateful if anyone can supply a listing of the standard subroutines, as mine are pretty inefficient.

C.R.J. Currie 41 Elmar Road,
London N15 5DN.

Your editorial in the June issue of NATGUG News has struck home - more than once I have thought about writing something for the Newsletter and then done nothing about it. To rectify the situation forthwith I enclose the text of my talk about LeScript at the Edinburgh Workshop. I have modified it slightly to suit the Newsletter, and added the result of a comparative assessment I have had the opportunity to make since then.

The general tone of some of your editorials since the QL was announced, and IQLUG formed, has been somewhat anti-Tandy, and one gets the impression that you would not be too surprised or worried if NATGUG folded up. This steady defeatist attitude in the magazine is just the way to precipitate its demise.

The Tandy range is spreading and is now much more business oriented than it was, so, many of our new members are probably first time computer users who are not hobbyists, and are mainly interested in the problems of using software like spreadsheets and other commercial packages. This means that the kind of problems our members have are probably tending to be different. Your experience at workshops, etc., would tell you whether my guess is near or wildly off target, but if I am right, then it may be necessary to ask specific people for specific articles rather than waiting for them to come.

If our membership now has a smaller proportion of pure hobbyists than it used to then we will have more members who will not accept the standard of presentation some articles have. As one of the correspondents in the latest edition of the IQLUG Newsletter put it, when referring to the two issues he had just received, he "read them with interest but some difficulty". I know just how he feels. Sometimes I just skip an article because it is too difficult to read. The one headed "+CODA+" in the June Newsletter is one like that - it is smudged, the type is very small, and the impression is faint. 10 characters to the inch is really the smallest type size that can be photoreduced acceptably. Unless the ribbon is new I would say that copy really should be both emphasised and bold to give a black enough impression.

I certainly would hate to see NATGUG fade away. I have benefitted to the tune of about £200 in discounts because of membership, and the "hot line" through Brian Pain has been a godsend to me on a few occasions, while I, in my turn, have been able to help others through it. I am sure a more positive approach would start the membership curve rising again. After all, all those Model I's and Genies that are being sold are being bought by potential members.

Ian Howard Wright

Rev. J. Wickens , 80 Orchard Road, Sutton, Surrey would like to hear from any EXACTRON users.

I haven't written to you or the Newsletter before though we have spoken on the telephone. It was you who kindly suggested the AMD chip AM7910 when I was interested in constructing a modem some six to nine months ago. I took up your suggestion and designed the modem with it working just about last Christmas - very successful so once again my thanks.

J.B. Moseley

10 Whirlowdale Crescent
Millhouses
Sheffield S7 2NA

(* How about an article on the modem for the newsletter? LFH *)

PROBLEM SECTION

Could I ask you to put a note in the News Letter to see if anybody has a copy of Volume One of the News Letter, issues that I need are 5 and 6: if they would be prepared to lend me those two issues for just a few days so I could read them ? (I recently purchased a complete set of back numbers but as those two are out of print they were not available). I have a spare copy of Volume Three issue 6 if anybody wants it.

Also does any member know of or have any experience of the following American simulations programmes: Forest Fire Fighter, Trucker, and Streets of the City. I saw them advertised in the American magazines about two years ago but can't remember where or who from, so I would be grateful for any information from anybody who has any of them or knows anything about them!

PA Roberts,
11 Spring Gardens,
Parkstone,
Poole,
Dorset.

I am currently using a Video Genie and running Sterm as a communication program to access Bulletin Boards.

I have been trying to use it to access Prestel on the normal 1200/75 service and not the 300 baud service operating on the London computers. However there are obvious problems with the control codes for graphics and colour change. Has anyone solved this problem with Sterm or could suggest an alternative program to enable me to access and subsequently store Prestel frames on disc to print out? No problem with configuring for the different transmit and receive rates.

I should be very grateful for any help!

John Moseley
10 Whirlowdale Crescent
Millhouses
Sheffield S7 2NA.
0742-36831

(* Try Don Bannister on (0232) 668899. LFH *)

BOOTING WITH DRIVE 0 WRITE PROTECTED USING LDOS

I was interested to read Henry Speherd's review of the different DOSs' available for the TRS-80 in the March 1984 issue of NATSUG. I detected a slight bias towrds DOSPLUS, but having never used it I am not in a position to comment on his findings. But having used LDOS extensivley I must disagree with his statements as regards booting with drive 0 write protected. It is done in a similar way to DOSPLUS using the MOD1/DCT driver programme supplied with LDOS.

For a two drive system this is how it is done. Execute the following JCL File using MOD1/DCT.

```
system(drive=0,delay=n)
system(drive=1,delay=n)
system(drive=2,driver="mod1")
1
system(drive=2,delay=n)
system(drive=0,wp=y)
system(drive=3,disabl)
```

This will set up the drives and produce a DEVICE readout as follows:

```
:0WP 5" Floppy No 1, Cyls= 40, Sden, Sides=1, Step=40, Dly=.5
:1   5" Floppy No 2, Cyls= 40, Sden, Sides=1, Step=40, Dly=.5
:2   5" Floppy No 1, Cyls= 40, Sden, Sides=1, Step=40, Dly=.5,
```

The problem now is to write the configuration file to drive 0 which is write protected. When a SYSTEM(SYSGEN) command is issued LDOS first loads Sector 2 in BOOT/SYS. It then checks byte 01. If it is C9 then a configuration file does not exist, otherwise it is 00. It then proceeds to write or delete the configuration file as requested and update BOOT/SYS, Sec 2, Byte 01. The following patch will force LDOS to do all this on Drive 2.

```
.Patch to SYS7/SYS.RS0LT0FF to force
CONFIG/SYS to be written to drive 2.
D06,AB=32
D07,2E=32
D07,7D=32
D07,B7=3A 32 28 49 0D
```

The 32 in each line can be changed to write CONFIG/SYS to any other drive required. You could even have a different configuration file on each of your data disks to be loaded on booting. Of course, in this case, there must be a disk in the designated drive or the system will hang. That problem is easily sorted out by putting a disk in the required drive and pressing Shift/Break.

One other problem is that with Drive 0 being write protected is that any SYSTEM/JCL files will be written to the first non-write protected drive, which will be drive 1. The following patch will force any JCL files to be written to drive 0 via drive 2.

```
.Patch to SYS6/SYS.RS0LT0FF to force
SYSTEM/JCL to be written to drive 2.
D12,B2=3A,32,03,4E,6F,20,66,69,6C,65,20,73,70,65,63,21,20,20,20,20
D12,79=BC
```

Or it could be left as it is and have a JCL file on each of the data disks. The second byte 32 is the one to change for any other drive. To make room for the patch the error message "File Spec Required!" has been changed to "No File Spec!"

One facility that is available, providing type ahead is active, is the abilty to configure a DOS Command. There is no mention of this in the manual and a I suspect it was never intended. After typeing SYSTEM(SYSGEN), while the system is going through it initialisation for creating CONFIG/SYS, a DOS command is typed it will be loaded with the configuration file and executed on boot up similar to an AUTO Command. So, by having CONFIG/SYS and a compiled SYSTEM/JCL file on a data disk, typing SYSTEM(SYSGEN), then DO *, the system can be booted, any parameters set, any filters or drivers loaded and the main programe loaded simply by pressing the reset button.

Henry Shepherd makes no mention of any JCL in his article. I know TRSDOS 2.3 and NEWDOS+ never had one. I have not used the NEWDOS 80 or DOSPLUS JCL, assuming that DOSPLUS does have a JCL. It would be interesting to compare them with LDOSs JCL as I have found it gives me quite a bit of flexability and saves a good deal of typing.

Finally, if you detect a slight bias towards LDOS in this letter, you are probably right.

D S Sutherland
1 CHESHIRE
Stanley Fort
Prop 1

TRS-80O ? ? ?

In view of the pessimism displayed in the last letter concerning the impending demise of the TRS 80 Models 1 & 3 I feel that the enclosed copy of an article that appeared in the July edition of PRACTICAL ELECTRONICS should help to aleviate this pessimism.

Added to this there is the welcome news that Hitachi 256K Drams (48256 16pin) are shortly to become available which will help drag the TRS-80 screaming into the next decade, these rams I am told can be easily fitted into the existing keyboard sockets the same as 64K rams & I think that there is only one more address to decode, the pin being in place of one of the two voltages not required.

The particular Z80O chip that will interest most of us will be the Z8108 as not much surgery will be needed on the existing circuitry.

I have written to Mr Coles via Practical Electronics, asking if he can elaborate on the subject of the 'daughter board' he refers to on the first page of his article, P E have acknowledged my letter saying that they passed it on to Mr Coles & will forward any reply he sends to them.

I have felt that if we take an interest in this development then it may be possible for commercial interests to take it up.

There has been so much hard & software developed that it is completely uneconomical to consider throwing it all away to join the 'larger memory ' brigade, there are still things that can be done with a Model 1 that the newer machines have not got round to yet.

The challenge will be to modify existing software to make use of the extra memory.

If anyone is interested in developing any thing along these lines I would like to hear from them so that we don't all invent the same 4th generation 'wheel'.

Details of the Z800 series can be obtained by contacting HITEK who are the ZILOG distributors in the UK, the phone number is 0954 81996.

E.C.Kilpatrick,
3a Gainsborough St,
Sudbury.
SUFFOLK. CO10 6ET.
Phone 0787 79504.

Fitting Instructions.

HI-RES Graphics Kit.

The modifcation comes in two parts:-

1. A separate, fully tested & assembled, graphics box that plugs into the left hand side of the interface or the keyboard edge connector.

2. An interface printed circuit board, with associated components, that has to be mounted inside the keyboard.

Before attempting the modification, please study the diagrams & instructions carefully. If there are any problems or unclear points please do not hesitate to contact us.

Fitting Instructions.

1. Carefully open the keyboard & position it in a manner that will allow easy access for a systematic approach to the modifications required. If your keyboard is NOT the new style then Z29 will not be in an IC holder, it will need removing & replacing with an 18 pin IC socket. Z46 will also need removing & replacing with a 16 pin IC socket.

A cautionary note. Great care must be exercised when soldering sockets to prevent solder causing shorts between adjacent pins.

2. Two track cuts are necessary as indicated in Figs. 1 & 2.

3. The following IC pins need to be isolated from their curent positions. This can ge achieved by either desoldering the leg & bending it at right angles to the board, or by using a sharp pair of pointed nose cutters, cut the IC leg just above the board & bend the remains of the leg at right angles to the board.

IC	PIN
Z10	14
Z11	9
Z41	7
Z72	14

4. Insert Piggy-backed 21L02 rams into Z46 socket & attach the wires as shown in Fig 3.

WARNING. It is advisable to check for shorts between IC pins before installing holders.

5. Connections.

Link pins 12 & 13 of IC Z26.

Using an insulated wire connect IC Z72 pin 14 (leg only) to IC Z74 pin 9.

Cover the contacts of the Power switch with insulating tape as in Fig 5.

Place the new printed circuit board behind the power, video & cassette sockets as in Fig 5.

Connect the coloured wires to the positions indicated in Fig 4.

Sleeving should be used over connections to bent out legs.

6. Connect 16 pin jumper lead to socket in Z29 as shown in Fig 5., leaving pins 9 & 10 open circuit.

At this point check all connections (Old ROM versions check for punctured wires, connecting the small ROM board stuck to the back of the main printed circuit board to the main board).

7. Connect power & video & test that machine functions correctly (ensuring there are not any metal objects under the exposed board).

The 20 way cable must be routed out of the rear of the two halves of the keyboard case assembly. (See Fig 4.).

Connection to the external graphics box is by way of the 20 way plug with it's locating peg facing upwards.

Ensure that the plug is firmly pushed in & that the socket's ejector lugs are closed so the connection is positive.

8. Carefully fit the machine back into it's case, power up & run the demonstration progam.

EC Kilpatrick,
3a Gainsborough St.,
Sudbury,
Suffolk CO10 6ET.
(0787) 79504

LESCRIPT — A REVIEW

LeScript started off with the unusual name ZORLOF, which sounds more like an adventure game than a word processor, and, not surprisingly, I don't think it sold very well. Then an improved vesion was produced for the Model III and it was renamed LeScript, and sells for £99.

Two or three months ago the Model 4 version became available in this country. And the big advantage of that, of course, is the 80 column by 24 line screen. I think it was the first of the Model 4 word processors to reach the market, although Superscripsit is now available from Tandy at £160, and Prosoft (of NEWSCRIPT fame) has just announced ALLWRITE in separate versions for the Model I, Model III, and Model 4 - but it costs $250.

When you buy LeScript you get a disc and a manual, and unlike most other programmes I have bought, there is no operating system on the disc, so you have to make a copy of your system disc and purge all the programmes you don't need in order to make room for all the LeScript programmes you want. The minimum you need is 27K for the main programme, and the printer driver

programme and perhaps another 5K for LESCRIPT/KSM which I will refer to later. The rest are demonstration programmes or programmes for converting files from other word processors like SCRIPSIT and NEWSCRIPT, or converting a PROFILE III+ data file into a LeScript form letter data file.

These conversion programmes were not all written by Prosoft, and I could not get the NEWSCRIPT one to work. That did not matter because there was no problem transferring my Model I NEWSCRIPT files (on DOSPLUS) to the Model 4, and altering the printer control codes can be done quite quickly.

To get it running you put the system disc in drive 0 and type in "LESCRIPT". If your Model 4 has the extra 64K of RAM then the RAMDISK is automatically brought in and you have about 90K of memory available for your document. This is the size of the biggest individual document LeScript can deal with as far as creating or editing is concerned, but with its CHAIN PRINTING facility it can print up to a total of 20 files at once, and they do not all need to be on the same disc - you just have to put the right disc in as it reads the file. It would be possible, in theory, to instruct it to print a single document 300,000 words long. This 4736 word article uses 31,182 bytes, and there are still 57,233 bytes available in memory.

Now the surprising thing is that it does not matter whether you are using a model 4 DOS or a Model III DOS, LESCRIPT/CMD works equally well with either, despite an entirely different way of converting keystrokes to letters on the screen. I have put it onto a Model 4 TRDSOS6 system disc and also a Model III LDOS5 system disc, and it works just as well with either of them. Put the Model III disc into a Model III and it has the usual 64 column display, but put the same disc into a Model 4 and it comes up with the 80 column display, and, if the extra 64K of memory is installed it uses that as well.

Once the programme is entered you see 3 STATUS lines at the top of the screen, with the LeScript logo at the right end, and 21 lines for you to enter text. These three lines are always there and the text scrolls up the remaining 21. The top status line has 5 fields. Pressing "CLEAR =" puts the cursor into the first one which is for the name of the document. This can be up to 8 characters plus an extension and a disc drive number. Passwords are not allowed, and, in fact, LeScript will not read a password protected file.

The second field is for the width of the text for display or printing purposes - anything from 5 to 255 characters. It is quite useful to set this at "80" while you are composing a document, so that you can see as much as possible on the screen, then change it to the width you want before printing, or viewing as it will be printed. This field can also be used to indicate certain types of file - BAS means a BASIC programme (saved in ASCII form), ASC means an ASCII file such as a file formed by NEWSCRIPT or SCRIPSIT, and EDT and EDR refer to EDTASM files. LeScript can both read and write all these types of files, and it can also read a SCRIPSIT file, but it cannot write one.

The next three fields are for information, and you cannot access them. The first is the word count, the second is the number of lines in the document, and the third is the number of bytes still unused. These three fields are constantly updated as you type. I have not checked the word count, but have found the count of lines to be a bit approximate, partly because it includes the lines with printer control codes which, of course, don't get printed, and you probably don't want them in the count.

The second status line has two fields marked "SH" for SEARCH, and "RP" for REPLACE. If you put any letters or words in this field, then press CLEAR-S the cursor stops at the first occurrence of that group of letters after the cursor position. Pressing CLEAR-S again moves on to the next occurrence. If you are not sure of the spelling you can use the BLANK character (±) as a wild card.

If you want to replace that word or phrase with another you put the replacement field in the "RP" field and when you have located the original word or phrase you press CLEAR-R to replace it. This gives you a selective search and replace, because it allows you to change it or not as you choose. If you press CLEAR-R with the "RP" field blank you delete the word or phrase searched for. To do an automatic SEARCH & REPLACE or SEARCH & DELETE then you use CLEAR-A instead of CLEAR-S.

In LeScript the CLEAR and CONTROL keys both perform the same function. This is very convenient, because they are at opposite ends of the keyboard, so any editing command can be made with one hand. I shall use the word "CLEAR" to mean either.

The third status line serves two purposes. Normally it shows a dot for each column, with the "tens" marked, and two dots for each of the preset tabs - the first one is at 7, and then there is one every 8 columns. If you want you can define your own tab settings, although that is a little fiddly, I believe. The second purpose of this line is to show special messages such as error messages, or comments like "DISK FULL", and so on.

EDITING FUNCTIONS

LeScript has very good editing functions. I used to think that NEWSCRIPT was very good, but LeScript is much more powerful. There are over 60 different functions, which sounds formidable, but they are really easy to use and most of them are linked to obvious letters of the alphabet. I found, that I very quickly got used to the ones I use regularly, and very seldom need to refer to the manual now. In the normal mode the cursor flickers very rapidly but CLEAR-I toggles the insert mode where the cursor blinks more slowly. Cursor movements are precise and fast.

Scrolling can be done quite rapidly three ways. 1. Text will scroll forward with the DOWN ARROW at the foot of the screen or backwards with the UP ARROW at the top. 2. The SHIFTED ARROWS will scroll with the cursor wherever it happens to be on the screen, which can be very useful when, for instance, making changes to a list of items and you want to see what follows the one you are changing. 2. The CLEAR-UP-ARROW and CLEAR-DOWN-ARROW scroll a page at a time. A useful feature in this is that the last line of the old screen becomes the first line of the new one, giving you continuity.

There is full word wrap around, of course, and if the specified width is greater than 80 the whole screen scrolls sideways as you type. Lines are automatically justified as you type or alter text. CLEAR-T gets you to the top of the document at any time, and CLEAR-E gets you to the end with the cursor on the bottom line (i.e., the screen full of text).

Apart from the cursor movements almost all the functions are called with the CLEAR or CONTROL key and just one other. Very conveniently there are two ways of deleting letters. The letter under the cursor is deleted by CLEAR-RIGHT ARROW, and if held down this continues to delete leters to the right. CLEAR-LEFT ARROW deletes the letter before the one under the cursor, and if held down that continues to delete towards the start of the line. I find that very handy as I do a lot of deleting.

Words and lines can be deleted by a command which will repeat if the keys are left down. CLEAR-W for words and CLEAR-L for lines. I was disappointed that there is not a separate delete paragraph function, but successive lines are deleted so rapidly that I do not miss that now. Whole blocks can be deleted by putting a block marker (CLEAR-B) at the end of the block, moving the cursor to the start of it and pressing CLEAR-*. In this case "once more to delete" appears on the third status line, and you either abort it by hitting any other key, or confirm it by repeating CLEAR-*.

This seeking confirmation before making a major boob is useful, and is provided also when clearing the memory to start a new document, killing a file, and exiting to dos.

Moving blocks around the text is very simple. A block marker is put at each end, the cursor is moved to the new location, and CLEAR-M is pressed. I moved a 7K block from near one end of this document to near the other in just a few seconds. The block markers are not automatically removed. That requires CLEAR-Q. At first I thought this was a weakness, but it is probably done on purpose to let you see that the right block has been moved to the correct place. Separate blocks can be saved to disc under individual names if you want to. Using CLEAR-C instead of CLEAR-M will copy the block to the new location leaving the original part unchanged.

LeScript does not like spaces. It deletes all but one for each word when it writes to file or rejustifies, so a carefully spaced out form collapses to the left unless you take steps to prevent it. CLEAR-ENTER inserts a small bar (▬) to indicate an intended space. LeScript automatically adds a space to every fullstop to give two spaces between sentences, but if you want more you have to put them in. This can be done with a global SEARCH & REPLACE, or just typing in two or three mandatory spaces. I found at first that LeScript was a good bit more tedious than NEWSCRIPT when laying out forms, but I soon got the hang of it.

One of the features of LeScript is that SUPERSCRIPTS, SUBSCRIPTS, BOLD printing, UNDERLINING, and ITALICS, can all be shown on the screen. CLEAR and a number from 1 to 5 toggles these on or off. Unfortunately it does not provide directly for printing the whole or a large part of a document in BOLD, which I find useful when the ribbon gets tired, or to get a good dark image to photocopy. However this can be done by defining a special printer control (▮C1B47 - the last four characters are hex for 27 and 71). When any of these are toggled on the text flashes alternately with the indicator - SMALL GRAPHICS BLOCKS for superscripts and subscripts, an ASTERISK for bold, UNDERLINE for underline, obviously, and a SLASH for italics.

It is possible to have the same word or letter shown as bold, italic, underlined, and either subscript or superscript all at the same time! It is quite simple to make individual letters, or even alternate letters print underlined, bold, italic or sub- or superscript - although the printer might get hysterics if there is too much of it. A few of these scattered about the text and it begins to look like a fireworks display! If all this flashing gets too much for you it can be toggled off with CLEAR-SHIFT-DOWN ARROW.

When your text is ready to commit to print or file to disc, or, in fact, at any time you like, it can be viewed as it will be printed by pressing CLEAR-V. It then appears on the screen exactly as it will on paper, with all the print control codes and end of sentence characters, and so on invisible. CLEAR-Z returns you to edit mode where you can alter the width by a few characters, adjust the margins and so on, and then view it again. CLEAR-Z returns you to EDIT mode instantly from any other mode such as printing, viewing, directory, or getting a file (which will stop loading in mid-file), or searching.

When you are ready to print you can either print straight away with CLEAR-P, or file it onto disc first with CLEAR-F. The first time CLEAR-P is pressed LeScript loads the printer driver programme. Thereafter the printer driver remains in memory and it starts printing straight away. There is no need to save your text on disc - it can be printed out and then deleted. It is quite possible to call up a shell for letter or document from file, write it, and print it off, and then call up the shell again for the next one without saving anything because the whole file is always in memory.

Another valuable feature of LeScript is the way that the disc directory can be examined at any time by pressing CLEAR-D. This gives you the directory on drive 0, but pressing "1" brings up the directory on drive 1. Then until you ask for a different drive number CLEAR-D will always give you the last drive asked for. The directory display is excellent. First it shows the number of granules free on the disc, then it lists for each file the exact number of bytes, the granules, and the date when it was last written to file. (The granules, by the way, are about 756 bytes. Pressing the space-bar brings up the next page of the directory. It is an odd thing, but it has to go back to the disc to read the next page of the directory before displaying it. Files can be killed to create space on the disc while in this mode if you want.

CLEAR-Z returns you to edit mode, but if you had wanted to insert another document into the one you were writing, or to append it, you just bring the cursor down to the name of the file and press CLEAR-G. The new document will be inserted wherever the cursor was on the file you were editing. Several files can be inserted one after the other this way. The directory is shown on the left side of the screen only and "PRINT CHAIN" is displayed at the top of the right side. If you want to chain several files together for printing - say something very long, or several standard paragraphs - you bring the cursor opposite each name in turn and press CLEAR-Q, and that name is listed under "PRINT CHAIN". Repeat that for all files to be printed together, changing discs if necessary.

It also does a few other things like semi-automatic hyphenation, renumbering BASIC and EDTASM files, and making the printer pause at chosen places, but the one which I find invaluable is being able to change the case of text without re-typing it. Holding down CLEAR-COMMA and letting the cursor run along the line changes everything to uppercase, while CLEAR-FULL STOP changes everything to lower case.

PRINTING COMMANDS

Now with word processors like NEWSCRIPT you can customise your system disc so that it knows what kind of printer you have, and whether you like to print one or two copies each time, and pause at the end of each page for feeding the next piece of paper in, or print the next page without stopping because you use continuous stationery, etc. Each time you enter NEWSCRIPT it automatically gets everything right, once it has been set up, and it also gives you the chance to change it just before printing if you want to. But with LeScript you have to tell it all this every time. However, this is not nearly as bad as it sounds and, as with so many things in LeScript, there is more than one way of doing it.

I personally have a little file which I call "HEADER", and, after typing in the name of the new document - say "NEWDOC", and entering the width - say "65", I then come down to the first text line and type in "HEADER", press CONTROL-G to GET it, and that file is loaded. This is it:-

```
|k34
|em,v2,i,t1,y66,e60,m6,
|j1
|`1B5330'11B5408'-081B5331'21B54  off
to print a half use the SHIFT-@ (^) as defined two lines higher
|on
```

The first character on each line is a vertical bar, which means that that line contains one or more printer controls, each separated by commas. The first line is a single code - "k34", and that means that the document is to be printed on an EPSON FX-80 printer in mono-type. "k35" would mean proportional spacing on an FX-80. When I got my copy of Lescript in March it supported at that time, 86 models of printer made by 30 different manufacturers. I believe it now supports about 150. There are altogether 58 different printer code numbers.

My header programme's second line has "em" for emphasised printing, "v2" to get two copies or versions printed at a time, "i" for individual sheets - that is stop printing until the next sheet is loaded, "t1" to print on the top line - if the top of the paper is just showing I find that the print head is just far enough down the sheet. Then there is "y66" which means that the paper is 66 lines long, and "e60" for the text to end on the 60th line - in other words, to leave 6 blank lines at the foot of the sheet. And finally, "m6" means a margin of 6 spaces on the left and right edges of the sheet. LeScript is very versatile in dealing with margins - you can set both margins the same, like that, or, if you are stapling together sheets which are printed on one side of the paper you can have a different size of margin on each side, and, if you are binding something together which is printed on both sides sides of the paper, then you can have different margin settings on odd numbered pages from those on the even numbered ones.

The next line is "jl" or "justify left" - each line starting in the first column like this. You can have "jr" for justifying on the right side only, or "jb" to justify both sides, as in a printed document like this, or "jc" for "justify centre" - in other words print everything that follows in the centre of the page. Any of these commands can be put anywhere in the document (on a separate line from the text) and they apply from there until the next change.

The next line in my header file is a mixture of characters and hex numbers. The first one you get by pressing SHIFT-@, and the rest are the hexadecimal values for EPSON FX-80 printer control codes to go into superscript, to print "1", to backspace, to come out of superscript mode, to print a minus sign, to backspace, to go into subscript mode, to print a "2", and to come out of subscript mode - in other words, to print "½". That is the definition how to print a half as a vulgar fraction - something I use quite often. Once that has been defined LeScript will print a "½" every time it comes across the character "\" in the text. You can use any one of 54 different keys to define special function characters that way. Then there is a comment to remind me how to get the half in case I forget. Comments are not nearly as neat as in NEWSCRIPT where you just precede a comment by ".cm". You have to start with "off" to tell the printer to ignore what is coming, then write the comment, then "on" to tell it to resume printing.

The ability to use programmable function characters is really a very powerful thing because it means that if there is anything that your printer can do - anything at all - for which LeScript has not got a direct command, you can get LeScript to do it in the middle of your text - even your own designed graphics graphics.

Well, that's the way I do my printing set-up instructions, but you can also use the KSM, or Key Stroke Multiply, facility. With this you can programme 55 keys so that each one represents a whole phrase, sentence, or even paragraph. You use SHIFT-CLEAR-A-Z SHIFT-CLEAR-1-5, and all the punctuation and other special keys. If you want to include end of line markers - "ENTER", in other words - then you use a ";". So, the whole of my "header" file could be store in the KSM file as a single keystroke (plus SHIFT and CLEAR). They do not say in the manual what the maximum length of each one can be, but the sample one which comes with LeScript contains some which are nearly 500 characters long.

This file, LESCRIPT/KSM, can be created and edited just like any other LeScript document, and if you wish to miss any letters out you just use an end of line marker

There are about 30 separate printing commands covering such things as:-
AUTOMATIC LINE-FEED INSERTION if your printer needs that as well as a carriage R.
The BAUD RATE setting for SERIAL PRINTERS.
CHARACTER DENSITY for PROPORTIONAL SPACING. I understand from a friend that the left & right justified proportional spacing is absolutely accurate to the last dot, which, as a publisher, is one of his top priorities in a word processor. Part of a document can be proportionally spaced if you like, as this paragraph is.

CHARACTER PITCH - that is the number of letters to the inch.
You can instruct the printer to SKIP A SPECIFIED NUMBER OF LINES - like leaving 10 spaces for a sketch - or go to a specified line, like the 20th from the top of the page, or go to line X only if there are less than Y lines left on the page.
There are several HEADER and FOOTER commands for pages, covering where you put titles, page numbers, and so on.
Parts of the text can be INDENTED a chosen number of letters at either or both ends of the line.
Text can be SINGLE or DOUBLE or TRIPLE SPACING or, in fact, anything up to 9 spaces between lines. This can be used in conjunction with special printer control codes to get odd things like 1.5 spaces, and with the SUPPRESS LINE FEED command to get spacing of less than one.

FORM LETTERS

Another very useful facility LeScript provides is the ability to do form letters. That is, standard letters which can be sent out to a lot of different people, each with their own name and address typed in, and the text personalised as required. I use them to tell people how to get to certain locations for meetings - those from the south get one set of instructions, those from the east get another set, while those from the west get a third set. There is no limit to the number of different items you can insert in any document.

You can use this for mailing lists, and you can insert codes in the list of names and addresses so that you can send to only some of the people on your list. I have codes for people who are members, competitors, committee members, and those who receive our Bulletin. So, obviously, some people will have all four codes against their names, and some will have one or two. It is really very versatile.

THE MANUAL

Having been used to the superb manual for NEWSCRIPT, I found LeScript's one the most disappointing part of it. Yet everything is there, but it is not always easy to find. There is a good introduction followed by a clear account of the STATUS lines. Then the editing functions are dealt with, followed by the printing commands. Everything is adequately described, but it is all in alphabetical order of keystroke - CLEAR-A, then CLEAR-B, then CLEAR-C, and so on instead of grouping associated commands together. But this is a minor gripe because there is a very full list of contents at the front, and a good index of editing and printing commands at the end. There is also a good tutorial on creating form letters, with a sample one to try.

Since the Edinburgh Workshop I have been given a copy of a 100-point checklist for evaluating word-processing programmes found in 'Word Processing Buyer's Guide' by Arthur Naiman and published by McGraw Hill. This shows scores for 14 various word processing programmes and dedicated word processors:-

PROGRAMMES

	Marks	Price($)
Pie-Writer	84½	200
Write	82½	400
Wordstar	80	500
Newscript	79½	125
Mince & Scribble	67½	275
Scripsit	61½	400
Magic Wand	61½	400
Easywriter	35	175

DEDICATED WORD PROCESSORS

	Marks	Price($)
CPT 8100	94½	15,000
Dictaphone Dual Dsp	85½	13,500
A.B.Dick Magna SL	83	14,500
Micom 2001	82½	12,000
Wangwriter 5503A	74	6,400
Wang System 5 Mod 3	70	11,500

Using the questions in the checklist, I gave LeScript 85½ points, and there were quite a few features of LeScript, like form letters, and programmable function characters, which were not in the checklist. Obviously this is not infallible. To start with, the values in the checklist depend on what weight the author decided each feature should have, and everyone is unlikely to agree with his weightings. But it does indicate where in the range LeScript lies, and, for a £100 word-processor, it is remarkably high up the list. The list also shows an interesting comparison between relatively cheap programmes and some very expensive ones.

There are just three things it does not do, which NEWSCRIPT does, and which I could find useful:-

1. It does not do "hanging indents" like this. That is, every line in this paragraph, after the first, does not automatically indent a specified number of spaces. That is not much of a problem because it is easy, after the text is composed, to run down it putting tabs down the left side, but NEWSCRIPT saves you the bother.

2. It does not avoid "widows and orphans", or paragraphs which start on the last line of a page, or end on the top line of a new page (or is it the other way round?).

3. It does not do automatic indexing, but I have not so far needed this.

I cannot really sum it up better than the reviewer in the April edition of 80-MICRO edition who said that he did not rush out to buy one once he had reviewed it because he was able to keep the review copy. So he went out and bought a Model 4P to use it on!

I. Howard Wright

17 Greygoran,
Sauchie,
Clackmannanshire,
FK10 3ET.

Telephone
Alloa (0259) 213515

Floating point accuracy

I was interested enough in Chris Oldman's enquiry (in the June NATGUG newsletter) to try his program on my old Genie. Having reproduced the funny results he complains of, I then wrote the following little bit of Level II Basic which calculates the square roots of a series of numbers, each fractionally less than 1 :-

```
90 DEFDBL P-R: DEFSNG X-Z
100 P=0.90000000000: Q=P/10: X=P: Y=Q
110 FOR I=1 TO 12
120 R=SQR(P): Z=SQR(X): PRINT USING "£.£££££££££££  "; P,R,X,Z
130 P=P+Q: Q=Q/10: X=X+Y: Y=Y/10: NEXT
999 END
```

0.90000000000	0.94868338108	0.90000000000	0.94868300000
0.99000000000	0.99498754740	0.99000000000	0.99498800000
0.99900000000	0.99949997663	0.99900000000	0.99950000000
0.99990000000	0.99995005131	0.99990000000	0.99995000000
0.99999000000	0.99999505281	0.99999000000	0.99999500000
0.99999900000	0.99999958277	0.99999900000	1.00000000000
0.99999990000	1.00000011921	1.00000000000	1.00000000000
0.99999999000	1.00000000000	1.00000000000	1.00000000000
0.99999999900	1.00000000000	1.00000000000	1.00000000000
0.99999999990	1.00000000000	1.00000000000	1.00000000000
0.99999999999	1.00000000000	1.00000000000	1.00000000000
1.00000000000	1.00000000000	1.00000000000	1.00000000000

The two left hand columns of results give the fraction and its root as calculated to double length floating point accuracy. The right hand columns do the same with single length floating point. It should be clear from these figures that the accuracy of both methods is limited. Both will eventually give up the struggle - if a number gets close enough to 1, well, lets call it 1! Hence since the root is always marginally larger than its argument, it is the root which is first rounded up to 1.

It is well known, but perhaps not well enough, that it is possible for computer users to suffer from a special form of delusion of grandeur called "delusions of accuracy". Since a computer is able to print results to 16 significant figures as easily as to three or four, it is tempting to believe that such lengthy results are meaningful. Quite often they are not. As the above results show, (look at the double-length root of 0.99999990000), the last few digits can be quite spurious - either because the data supplied is itself insufficiently accurate, or because the numerical method of calculation necessarily employs some approximations, which is evidently the case with Basic's SQR function.

Negative INTegers

A further trap illustrated in Mr Oldman's program concerns the way that the INT function handles negative numbers. The definition of INT(x) is "the next lowest integer smaller than x". Hence INT(-1.5) is -2, because -2 is smaller than -1.5. In converting a negative value in seconds to minutes and seconds via the INT function, the program will report -90 seconds as -2 minutes +30 seconds, which is arithmetically correct but which catches the user unawares.

Compound conditional statements

Mr Oldman also asks why he cannot combine the conditional statements in his lines 140-160 into a single line. The answer is that he can, but not by concatenating statements using the colon alone. In other words, you cannot write:-

```
140 A$=INKEY$: IF A$="" THEN 140: IF A$="F" THEN 200
```

because everything following the first IF is dependent on that IF. The final statement IF A$="F"... will never be reached, since if A$ is null we will GOTO 140, and if it is not we will proceed to the next numbered line.

Once this is understood, it can be exploited to build up very useful compound statements. For example Mr Oldman's lines 140,150,160,170 and 200 can be condensed to one line as follows:-

```
140 A$=INKEY$:
    IF A$="F" OR A$="f"
    THEN CLS: PRINT @535,"NO TIME TO LOSE":PRINT:END
    ELSE IF A$=CHR$(32) THEN GOTO 20
    ELSE GOTO 140
```

I have inserted a few CTRL's to improve the layout and clarify the meaning. Compound statements of this type can be used to reduce the number of GOTO's in a program, with benefit to running time. The GOTO is a particularly inefficient statement in Level II Basic. Of course, speed is unimportant in the above example, which is merely scanning the keyboard for one signal, but in many other circumstances this trick is valuable.

What about our future?

Whilst I am writing, and since the Editor is getting short of material, I will add my gratuitous views to the great debate. I personally take great pleasure, even after four years, in reading my monthly newsletter. Of course much that it contains is of only academic interest to me - it always was, since I have never added discs to my system and the numerous articles on DOS's etc fail to enthuse me. Still, I think I must have found something of value in every issue.

Nevertheless I recognise that in time our TRS-80's and Genies will pass into the micro's Valhalla. But not yet, at least my Mark 0 Genie won't. Yes, I have joined IQLUG, in order to listen in to the birth pangs of what may be a phenomenal new arrival, but I certainly have not placed an order yet for a QL, and perhaps I never will

I think that what attracted me most to the idea of the QL was the sheer audacity of the machine, at such a price. The impact of the specification was immediate. It seemed sure to sweep the country. Now that the technical audacity has been proved to be nearly matched by the marketing audacity of the launch, second thoughts intrude. Of course, there is still time - perhaps a year or more - in which Sinclair can put matters to rights and begin to swamp the market with machines to the original spec. When the early users begin to sing the QL's praises in "Quanta", and a QL Pascal package appears, I may be ready to take the plunge.

Meantime, other new machines come out, and will keep on tempting us. A bunch of literature on a thing called an "Advance 86" popped into my letterbox. What have we here? Another new micro? Designed in Britain; manufactured by Ferranti; marketed by W.H.Smith; IBM-PC compatible. How boring. But it seems the Advance 86a sells at the same price as the QL; comes with more ROM and more RAM; has more keys and more function keys; uses cassette tapes instead of microdrives; and beats the QL for speed on the benchmark tests. The makers, what's more, support an upgrade of the machine to an 86b disc system

I'm glad that I still love my Genie, and have no urgent need to change it. This is not the time to commit oneself to a new home micro unless one has to.

John Gale Hemel Hempstead 0442-53643

FOR SALE AND WANTED

PROGRAMS FOR TRS 80 MODEL 100

'DRAWER' Produces histograms, graphs and other charts from simple data. Stores data and titles for future re-use.

'UPDATE' Keeps track of programs used, and their length. When length changes, UPDATE re-labels and dates the new version. Can be set to run automatically on start-up.

Each program £6.00. Both on one tape for £10

MARTIN 15 MASON ST, WARRINGTON WA1 2JJ

FOR SALE

dBASE II for TRS-80 Model II and Model 4

ONLY £315 + VAT.

WORDSWORTH COMPUTERS LTD, 6 THE SQUARE, ABOYNE, ABERDEENSHIRE AB3 5HT 0339-2070

Tandy TRS-80 Model 1 with numeric keyboard.
Expansion interface high resolution graphics board and BMC green monitor.
Twin disc drive and DMP 100 matrix dot Printer.
Custom desk to house everything Plus all wires cables and Power.
Also included TRS dos 2 Point utility disc N E W dos 80 on disc, ScriPsit on disc, Tandy Scribe on disc, Electric Pencil on disc, Accounts Programme on disc, Tandy Mail on disc and Profile on disc.

Offers in the region of £600.00 complete Package.

T.R. GALLOWAY
67a High Street, Banbury, Oxon.
Telephone: Banbury (0295) 3978

6 Plantation Road,
Ballymacormick Road,
Bangor,
Co Down.
BT19 2AF

11 July 1984

L A B E L M A K E R

The following submission may be a useful review to those members of the user group whose wives are keen jam makers or fruit bottlers or who find the need or desire to print labels for the other thousand and one reasons!

Some time ago I speculated in a piece of software advertised in the MOLIMERX catalogue and have found it one of the most useful practical programs in my collection. We even use the program to pre-print labels for the milkman requesting 'ONE EXTRA PINT PLEASE'. It only costs £18.40 which I feel is a very reasonable sum considering the range of facilities afforded.

The program provides 16 label formats for the standard 15/16" x 3 1/2" labels. A small supply of these labels plus an even smaller supply of cassette labels are enclosed with the program. The program prints only 1 label wide so if you are using labels 2 wide then I find it best to print down one side then turn the labels around and print on the other side.

The master diskette comes on 35 track TRSDOS 2.3 formatted for Model I. The program works on MODEL III and MODEL 4 (in MODEL III mode) and can be easily converted to TRSDOS 1.3, NEWDOS80 V2 or LDOS. The disk comes with the main kernals of two programs 'LABLMAKR' and 'CASSLABL' plus a variety of patches which can be merged with the kernals to cater for the various Epson MX80 / MX100 range of printers. The patches take care of three conditions:

Epson printer with no GRAFTRAX
Epson printer with GRAFTRAX-80 (MX80FT model)
Epson printer with GRAFTRAX plus (MX80/100 III models)

(See later comments regarding FX series)

Installation procedures are extremely well documented and the novice can embark on installing the program with ease. There are in fact individual installation instructions for the following Operating Systems:

TRSDOS 2.3	Model I
NEWDOS+ V.2.1	Model I
NEWDOS80 V.1.0	Model I
NEWDOS80 V.2.0	Model I
TRSDOS 1.3	Model III

After installation and running the program a menu is presented:

1. Index
2. Access Label File
3. Printer Routines
4. Add New Label
5. Install Alternate 'LABLMAKR' diskette

Choice 1 allows a preview of either sample labels displaying the JUMBO (3/8" high) letters or all the labels you have on file.

Choice 2 is used to call a particular label from file and display it on the screen so that it can be either modified and/or printed. Multiple copies of any particular label can be requested.

Choice 3 allows for the printing of a complete set of sample labels, or list of labels on file.

Choice 4 is the main one used to create the text on the labels. Where JUMBO text is required, it is actually displayed as such on the screen using graphics characters. Text can be automatically centred. It is possible to create a file of similar labels with one command.

Choice 5 is called where you have several disks full of label formats and it gives an opportunity to change data disks.

The appendix gives printer DIP switches for the three types of Epson and MICROLINE 82-A together with printed examples of the 16 label types and some application ideas from 'LABLMAKR' users. From trials on an FX100 I find that the output is very erratic. The JUMBO letters don't form completely and only some of the label types work properly. I wrote to Molimerx and they have agreed with this but say that they are contacting the author to see if a suitable version is available.

The program is written Basic and is suprisingly fast considering the work required in computing using dot matrix tables for the JUMBO letters. In summary then, I would recommend this program to all disk users. You can label your disks neatly with it. I enclose a print of the 16 sample labels available.

David Roberts

(0247 - 462564)

Label Type 1

JUMBO 13 CHAR
(Compressed Type; line length maximum = 50 Chars.)
(Compressed Type; line length maximum = 50 Chars.)
(Compressed Type; line length maximum = 50 Chars.)

Label Type 2

JUMBO 13 CHAR
Double Wide; Length = 25
Double Wide; Length = 25
Double Wide; Length = 25

Label Type 3

Double Wide; Length = 25
JUMBO 13 CHAR
Double Wide; Length = 25
Double Wide; Length = 25

Label Type 4

JUMBO 13 CHAR
Double Wide; Length = 25
JUMBO 13 CHAR

Label Type 5

123 (Compressed Type; length = 37 Chars.)
(Compressed Type; length = 37 Chars.)
(Compressed Type; line length maximum = 50 Chars.)
(Compressed Type; line length maximum = 50 Chars.)
(Compressed Type; line length maximum = 50 Chars.)

Label Type 6

123 (Compressed Type; length = 37 Chars.)
(Compressed Type; length = 37 Chars.)
(Compressed Type; length = 37 Chars.)
(Compressed Type; length = 37 Chars.)
(Compressed Type; length = 37 Chars.)

Label Type 7

123 Double Wide = 19
Double Wide = 19
Double Wide = 19
Double Wide = 19
Double Wide = 19

Label Type 8

123 Double Wide = 19
(Compressed Type; length = 37 Chars.)
(Compressed Type; length = 37 Chars.)
(Compressed Type; length = 37 Chars.)
(Compressed Type; length = 37 Chars.)

Label Type 9

<Compressed Type; line length maximum = 50 Chars.>
<Compressed Type; line length maximum = 50 Chars.>
<Compressed Type; line length maximum = 50 Chars.>
<Compressed Type; line length maximum = 50 Chars.>
<Compressed Type; line length maximum = 50 Chars.>

Label Type 10

Double Wide; Length = 25
Double Wide; Length = 25
Double Wide; Length = 25
Double Wide; Length = 25
Double Wide; Length = 25

Label Type 11

Extra Wide 14
Extra Wide 14
Extra Wide 14
Extra Wide 14
Extra Wide 14

Label Type 12

Double Wide; Length = 25
<Compressed Type; line length maximum = 50 Chars.>
<Compressed Type; line length maximum = 50 Chars.>
<Compressed Type; line length maximum = 50 Chars.>
<Compressed Type; line length maximum = 50 Chars.>

Label Type 13

Extra Wide 14
<Compressed Type; line length maximum = 50 Chars.>
<Compressed Type; line length maximum = 50 Chars.>
<Compressed Type; line length maximum = 50 Chars.>
<Compressed Type; line length maximum = 50 Chars.>

Label Type 14

Extra Wide 14
Double Wide; Length = 25
Double Wide; Length = 25
Double Wide; Length = 25
Double Wide; Length = 25

Label Type 15

Double Wide; Length = 25
Double Wide; Length = 25
<Compressed Type; line length maximum = 50 Chars.>
<Compressed Type; line length maximum = 50 Chars.>
<Compressed Type; line length maximum = 50 Chars.>

Label Type 16

Extra Wide 14
Extra Wide 14
Double Wide; Length = 25
Double Wide; Length = 25
Double Wide; Length = 25

BLACKCURRANT

11th July 1984

MAGNETIC MATERIAL

HANDLE WITH CARE

1 EXTRA PINT

TODAY PLEASE

NO MILK TODAY

THANKS

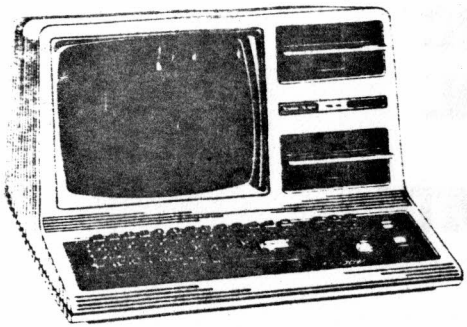
CPM V 2.2

DATA

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

NATGUG NEWS

Volume 6 Issue 2 August 1984

OFFICIAL JOURNAL OF THE

National TRS-80 & Genie Users Group.

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.

The Group maintains two software libraries (Models I and II) which are free to members. Library lists are obtainable from the Secretary.

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

CONTENTS

2. Information on the Group

3. Editorial

3. Guidelines for contributors

3. Library information

4. Members' letters

13. Problem section

16. Answer/originate modem

17. Modem set-up program

17. 80-NET operational again

18. A computer-controlled frequency synthesiser

18. Super Scripsit space code changes

19. Tandy Model 100 and the BBC

22. Spooling Dotprint printer output to disk (part 1)

25. For sale and wanted

30. Tandy DMP2100 printer

32. Typesetting continued

34. Scripsit +

MILTON KEYNES SUNDAY MORNING MEETING
FOR A TRIAL PERIOD UNTIL CHRISTMAS 1984
PLEASE MAKE A NOTE THAT WE WILL BE
MEETING ON THE FIRST AND THIRD SUNDAYS
OF THE MONTH FROM 10.00 to 1.00 TO BE
HELD AT YORK HOUSE, STONY STRATFORD,
MILTON KEYNES. A map can be sent to
those faint of heart at the thought of
entering MILTON KEYNES.
Brian Pain 0908 564271.

****

Swindon Workshop 7/8/9 November 1984.

Topics and speakers required.
Contact Leon Heller or Brian Pain
0903-613004 or 0908-564271.

****

EDITORIAL

Sorry about the lateness of this issue, I've been rather busy lately.

Contributions to the newsletter seem to be increasing. Keep up the good work.

As usual, I will be organising a stand at the PCW show, which has moved to Olympia. The emphasis will have to be on the QL, I'm afraid, but we will be very pleased to see NATGUG members, also. Brian will be organising a "hospitality room" in a local boozer.

I was successful in the recent Radio Amateurs Examination, and am now a licenced amateur, with G1HSM as my call sign. One area of amateur radio that greatly interests me is packet radio, where you have a network of personal computers linked via radio, which in a few years should develop into a world-wide communications system enabling licenced amateurs with micros to communicate with other amateurs anywhere in the world, at minimal cost. These techniques, although legal in the States and Canada, are not yet authorised in this country. It is a very exciting development. If any of you are interested in amateur radio, and are put off by the thought of the examination, go ahead, it is quite easy.

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 hand-written material, write legibly.

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

Leon Heller

LIBRARY INFORMATION

Model 1: Leighton Davies, 105 Caerau Road, Caerau, Glamorgan. Tel: (0656) 738337

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

MEMBERS' LETTERS

Thanks for a superbly organised weekend, I enjoyed myself very much indeed.

Here is the address of the firm who have bought up General Northern's stock:-

Inpholink,
Front Street West,
Bedlington,
Northumberland NE22 5UB.
Tel: (0670) 829009

E.C. Kilpatrick,
3a Gainsborough Street,
Sudbury,
Suffolk CO10 6ET.

(* GNOMIC have been 'liquified'. LFH *)

This letter has been written the day after my arrival back home, after the last NATGUG/IQLUG meeting in York House. I would like for the world at large to hear why I feel that I have just wasted £60 on the above function.

To start with, we can write off £20 for petrol for the car.

Now we come to the meeting itself. The lunch and evening tea should have been priced the same, as it was the same food, with the exception of desert, which would not account for the additional £1 on the price. Whilst on the subject of food, I must also comment on the fact that a salad is nice once in a while, but not for four days on the trot, although it must be said that there was a choice of meat with the salad on two of the days. This was ham or fish. As I am allergic to fish, this left me with a large choice! This amounts to £16.25, although not entirely wasted.

I now move on to the accomodation, leaving the lectures to last. As usual, the hostel at which I stayed, was excellent, but I have one complaint, and that is the time at which we had to be in the hostel: 23.00. This is much too early, but never mind. Cost: £21 for four nights.

Concerning the lectures, luckily I did not have to part with any money for this, as I was supposed to be giving two days of lectures. This left me with enough money to purchase four very low-quality disks from Os, two of which I can only use as TRS-DOS disks.

I was now ready to start my lecture at the time set by Brian (10.00), but no-one knew about this start time, so the lecture was cancelled. Then on Thursday afternoon Brian was asked when I would be starting, to which he replied 'Friday morning, with just the three of you'. Next morning, when a time was announced, we had 10 people in the room almost immediately. Mind you, these people were there the previous day, but we did not know what they wanted, as the booking form did not allow them to specify. A slight oversight on the booking form, that.

So far, this sounds like an all-out attack on Brian, and I apologise; it isn't intended to be. Sorry, Brian. To be totally fair to Brian, you must give concessions for his starting up IQLUG as well, for it must be a lot of hard work and things can quite easily be forgotten.

The Saturday focussed on the QL, and the Sunday was relatively like a normal NATGUG meeting. If things continue in this vein for the next year or so then I'm afraid that I just can't afford to come to a meeting to do as much as I do at home.

Although most of you will quibble and say 'Well, what's £60?', I would like to make it known that I am at college on a grant that entitles me to £8 per week, and on that sort of income £60 is hard to find.

I suggest, for the good of both organisations, that the two groups not split, but rather, to have two separate secretaries to take on the burden, as it is evident that Brian can't be lumbered with it all. After all, I have enough difficulties running my own club, let alone two national ones.

Brian, you have done a great job with NATGUG, and probably IQLUG as well, but don't spoil it by trying to do too much!

P. Knaggs,
12 Seymour Road,
Chippenham,
Wilts. SN15 3NH.

(* I am sorry about Peter's disappointing week-end. His letter is considerably at variance with the one that precedes it. I too thought the food was rather grotty. You must make allowances for the fact that this was the first true joint workshop. We will endeavour to see that future meetings are better organised. What do other members think of the MK week-end? *)

I have just undertaken a major step and taken on Assembly Language Programming from the ground up. Somehow the apparant mystery and complexity of the language made me, and I'm sure, many other members, shy away from it. The opportunity arose because we had not planned any "away" holidays this year. (The weather was too good to waste money this year on the continent!). So with the annual duties of gardening and house painting behind me, I was browsing through some back issues of 80-MICRO and came across a great three-part article on Assembly Language Made Simple by Hardin Brothers. (December 1983, January 1984 and February 1984). Looking back through the newsletters for any hints and tips, I remembered that I bought a copy of Laurie Shields two-part article on Assembly Language Programming. So equipped with EDTASM I made a start.

I have only been at it for a fortnight now and have purchased ZEN84 (Model I and III, ZEN4 for Model 4, is included in the package) for £44.50. So far I have managed to draw a graphic border around the screen and place a smaller rectangle inside it with my name inside the rectangle. Doesn't sound much but believe me, the feeling of excitement at getting it to run makes the effort worth while. More importantly though, there is no better way of learning the language than 'getting stuck in' and writing something - no matter how simple or useless. You will consolidate the knowledge much quicker. There is only one way which I know to consolidate knowledge quicker; that is, try

to teach a fellow 'computerist' what you have learned. It really rams (ugh!) the knowledge home. I quite often use that technique with local users.

Again whilst browsing through back issues I see that Tim Bourne has contributed several articles. The reason that this caught my attention was that he is the author of the PASCAL compiler and editor which is distributed by Molimerx. I would thoroughly recommend his software for those who wish to try PASCAL. The latest release is 6.1 and now includes amongst other facilities, Records. As yet the implementation does not include the type STRING. String handling can only be achieved by means of and ARRAY of CHAR. I hope Mr Bourne enhances his excellent software with this facility.

Whilst on the subject of PASCAL, there is a course presently being run by the Open University (Associate Student Programme) PM683 Structured Programming with UCSD Pascal. The fee is only £65 and is a non-examination course. I applied last year and have been following the course at my own pace. (OK! I hear you say, what about the assembly language programming you are doing!!)

The course is extremely well documented andyou are supplied with:

a course text of five books
an activity booklet containing programming exercises
a guide to the UCSD Pascal environment
a set book - the UCSD Pascal Handbook
a reference manual on data structures
instructions for setting up and doing the problems on your own computer.

It is estimated that the course will take about 100 hours of study. I presently have reached Unit 8 out of a total of 10 Units in the course. So far Mr Bourne's Pascal has served me well in this pursuit as it has the nice facility which compiles the source code to Z80 object code, not P-code as in some other implementations.

The address for those interested in the OU course:

The Admissions Office,
The Open University,
P.O. Box 48,
Milton Keynes,
MK7 6AB

Well, that's all for now.

David Roberts

0232-462564

As promised I enclose hereby a copy of aletter which I have sent to PCW which sets out details of my "struggle" to reclaim £10.89.

Obviously I have been "caught" and I really would not like anyone else to be and feel it is important that both clubs are aware of the position when ordering goods by mail.

I leave it to you, if you wish, to make any additions or alterations as to how you want to put this over in the two newsletters. I have checked with Citizens Advice that as my letter to PCW of the 5th August only contains sections of letters sent to me and I have also not expressed any opinion that might be libelous, it is perfectly in order for me to have written as I have.

I am quite sure PCW will not publish this letter in their magazine but at least by writing it I feel better!

I also enclose the relevant section of the protection scheme for you.

Miriam Pollard

62 Alexandra Park Road
Muswell Hill
London N10 2AD

"CAVEAT EMPTOR

On the 28th November 1983 I placed an order for software advertised in PCW, the company advertising being Silver Fox Communications Ltd., the value of the order being £10.89 which I paid by Access. On the 13th January 1984 I noticed that Access had been debited despite my having in writing cancelled my order.

On the 26th Jannary 1984 I informed Access and on the 9th March they replied stating "unfortunately the retailer has since been closed in our files. In the circumstances may I suggest that if the order was placed due to an advert in a newspaper or in a magazine, that you seek compensation direct from the Newspaper Publishers Association or the Periodical Publishers Association."

I received a further letter dated 19th March stating that "you are quite right under the Consumer Credit Act 1974 equal liability is between the retailer and credit guarantors but for items in excess of £30.(it is now £100). "As the amount of the transaction is under £30 you have a valid claim against the publishers of PCW and if you write to explain the circumstances including details of the advert you responded to then they are obliged to refund the monies to you."

This I did on the 20th March and 26th June and on the 20th July PCW replied as follows: "Our Mail Order Protection Scheme is designed to cover readers who have ordered goods from companies that have been declared bankrupt, Silver Fox Communications have not been declared bankrupt, but has, it seems absconded owing a number of people money, including yourself and VNU. Therefore under the circumstances we are unable to refund your money."

On the 26th July I asked for a copy of the Mail Order Protection Scheme and was advised in their letter of the 30th July - "My advise to you is that you write off the £10.89 and save yourself further expense by refraining from more correspondence."

I used to be a regular subscriber to PCW but you may rest assured that I shall not buy a copy again.

Of course it is your privilege to publish or refuse but I would respectfully ask that you do publish this letter as I feel it better to appear in your own magazine than a different magazine. Please let me know which edition this letter will appear in.

Yours sincerely,

Mrs. M. Pollard

I have recently purchased Scripsit for the Model 100. It operates with the built-in TEXT editor and provides excellent formatting and printer control capabilities including right hand justification and selection of number of copies for printing. It costs £19,95 from Tandy. Well worth it.

That's the good news. The bad news is that it does not run with the standard 'A' ROM as supplied in the UK Model 100s. My son bought the Starblazer game in the States and met the same problem. On enquiring of Tandy they fitted a 'B' ROM for their usual £5 (plus VAT) and all works well. It is essential to have this ROM if you intend to use the Video interface. I have tested this briefly and consider it highly overpriced at around £600 for what is provides.

On another front I have been wanting to move to a 16-bit machine and have reviewed the market for reputable suppliers. At work we are officially IBM-PC and DEC oriented with splatterings of Wang and HP. Some users have themselves acquired Sirius, Apple etc but that's their lookout!

My main requirement was for IBM-PC compatibility. In the end I narrowed the choice down to the Tandy 2000 and the Olivetti M-24. The former, despite its good spec and performance is definitely not PC compatible and I had to dismiss it for this reason.

That left the M-24. I was able to test it extensively with IBM software and could not beat it (by the way do NOT believe the test results in PC User July 1984 issue which comprised a Directory of personal computers. They had to retract their statements at the PC User show and Olivetti, I believe, come close to suing them!). The bottom line is I acquired one with 256K RAM and 2 320K discs drives, mono monitor (the colour one comes out later this year), MSDOS 2.11 and GW-BASIC for under £2,000. It is not simply a PC-clone for its offers increased performance on almost all fronts for less money yet has a distinctiveness all its own including its Italian styling. The M-24 is extremely fast (twice the speed of the IBM-PC), has a genuine 16-bit 8086 chip and offers great expansion possibilities. You have the choice of the terrible IBM-standard layout keyboard or Olivetti's own (you can guess which I chose).

I still have my Model-I, but will probably sell it off in time once programs are transferred. By the way apologies to those who answered my advert for the Model I in the March issue. I did not want to sell until a new machine had been acquired. If anyone is still interested contact me again. I shall put another ad in later, however.

I'm very pleased with the machine and I gather that Molimerx are investigating handling Olivetti software.

If anyone is planning to go Olivetti M-24, let me know as if there is sufficient interest I may start up a Users' Group. By the way the dealer I acquired it from is very helpful - if interested contact Mike Prudence at KGB Micros Windsor (Tel Windsor 50111)

Bob Wiggins Hassocks 3430

The search for a decent general ledger is still on. The choice has been narrowed to Peachtree and Sage. The main problem however is the storage. To assure compatibility with Model III Tandy has provided two 184K drives, which really is not enough to run serious accounts programs. So, perforce, I may have to go for larger storage - a dual, double-sided 80 tracks drive. As the generally recognised minimum storage required for an account program is 250K per drive such drive should provide some capacity to spare. Of course CP/M+ does not support double sided drives (did I hear that somewhere before ?) and I also know that should I replace my original drives with 80 track ones I would not be able to read Model III or TRSDOS 6 disks. It seems to me that although CP/M+ is really very good value for money and though it offers a lot to a programmer Montezuma CP/M is a better bet as a work horse when it comes to the standard CP/M software. Anyone would like to sell me cheaply this program ? Only legitimate copies considered !

Encouraging news in the latest Harding listing. Models 4 and 4P are doing well and there are new independent TRS-80 houses. Of course Tandy have brought the price down - for the period of their sale - by about £300. I wonder. With the NEC business system for £1000 including the printer coming on the market I think that this reduction in the price will be permanent. Anyway for the fun of it I went to Tandy to demand a refund of £300 but no such luck. It is a shame though that 80Micro went on the diet as well. My usual supplier at Victoria near platforms 1 & 2 stopped stocking it though the big Smith on the main concourse still have it. If they also stop, the search for a supplier will be on.

I wonder what is happening in the QLUG camp ? I have not heard anything lately about this wonder - maybe because I no longer take PCW and PC. But I don't see it advertised heavily anymore in the newspapers and Sunday supplements. I have a feeling that the orders, although heavy to start with, may tail off quickly. First time Spectrum owners may upgrade but I don't think that those who went from ZX to Spectrum will want to spend the money on yet another Sinclair. Commodore 64 is doing well though and it is becoming difficult to get. My DEC engineer was looking for one for his nephew and I found him a supplier but it took some time. One can buy it easily but only as a system - including disk drives, printer and some programs.

The June copy of the NATGUG journal has finally reached me (20th July) and I can see that the Sunday of the MK workshop will be shared. Is this wise ? There is this amazing manifestation of "loyalties" in the micro world. "My micro is better than any other and the language I use is the best" and so on. Well micro is micro (so long as it is not a con) and, providing that the keyboard is comfortable and does not require finger acrobatics, the display is clear, there is sufficient memory and a reasonable speed, each one can be usefully employed as a tool. As far as the languages are concerned they are all reasonable or lousy - depending on the mood. I have tried Basic, RM Cobol,

Pascal and Dibol having very quickly discovered that the Assembler is beyond me (and by implication Macros, Forth and C). Basic is all right for quick, short programs but trying to write a longer program is murder. In the early stages and in the days of great enthusiasm I wrote myself an insurance program (some 16K of it). It did everything I could think of including re-assessment for inflation. I now use Data Writer to keep track of my insurance. Then I wrote a general ledger program (some 34K of it and I had to break it into two parts). Again this program did everything I wanted with one serious limitation. The transaction file was loaded into memory which brought with it all sorts of problems like garbage collection etc. This program was discarded some time ago. Then I started on RM Cobol, a self-documenting language but so verbose. Why does one have to say "Display (or accept)...line....position" ? Why could it not be just "display(x,y)...." ? Now with Pascal which, like Cobol, enforces structual programming, there are all these brackets, the begins and ends. The language syntax is very unnatural. Dibol I maybe like most of all - probably because I have not done any serious programming using it. Just small improvements to the programs written for us by the OEM. But, if the machine/language partisanship is going to creep into the joint sessions I may attend future workshops just for the fun of it.

It is astonishing how little the people selling programs know about their wares ! Having decided to acquire CP/M 2.2 (Montezuma) I rang (you know who) to find out a few things about it. After all the only UK supplier will not accept returns if the program is found to be unsuitable and one does not wish to buy a pig in a poke. I found out that this version of CP/M does cater for different types of drives (including 80 tracks double sided) but the dealer has been unable to tell me whether I would be able to define physical drive 3 or 4 as logical A: or B:, whether this CP/M would treat so formatted diskette as one unit of 720K or would it consider each side as a separate unit, or even whether the manual contains the necessary codes to customise CP/M programs !! In fact the attitude was "Don't ask me all these questions - if you want the program send the money Charlie". It would also seem that the US version differs as far as the configuration program is concerned from the version sold in this country. The USA version permits defining the keyboard, assignment of OIBYTE (whatever that may be), and the printer linefeed filter in addition to the options specified in the UK catalogue. Well I have written to Montezuma now and, providing that I get an answer and the answer is satisfactory, I will buy it from the USA (even if this will cost me more). No wonder that people are ripping off the programs. As I say - if the program does what I want I buy it but I must be sure that the program will do just that.

I have been having great fun with the SAGE Try Before You Buy program. If you have not heard of it, for £20 you get a disk and an instruction booklet which takes you through most functions of the full program. If you decide to buy the program the £20 is reduced from the cost and, if not, you lose it. Still this

loss is bearable if one contemplates spending a few hundred Pounds on a program. This booklet also contains a table showing how the volume of the data disk can be calculated. The program is very easy to operate and one can almost use it just by following the menus. One starts with the initialisation and one is asked how many sales, purchases and nominal accounts to open. One must open a minimum of ten purchase accounts, ten sales accounts and ninety nominal accounts. In addition a file of 135 nominal categories is also created. As each sale, purchase and nominal account requires 50 bytes and each nominal category 30 bytes 9550 bytes are required for minimum files which must all reside on drive b:. The remainder of the space on this drive is available for the transactions' file - each transaction requiring 90 bytes. With the CP/M+ formatted disk of 154K this permits some 1,600 transactions per period - and I hope that I am right otherwise I will be in for a shock.

The main menu provides several options and some of the options also have their menus. Choice is on Inkey$ routine with just a slight confusion. For example choice 02 takes one directly to the function but 2 requires a <return>. The same applies to the choice of accounts.

Having initialised the files one then proceeds to allocate the accounts numbers and their descriptions. There are 999 sale/purchases accounts permitted and 999 nominals. Sales & purchases share one sequence of numbers and the purchase accounts take the lower numbers so, if the minimum allocation is chosen, accounts 1 - 10 would be purchase accounts and 11 - 20 sales accounts. Nominals have a separate series from 1 - 999. The nominal numbers of 38, 65, 69 and 89 are pre-allocated for Debtors, Creditors, VAT controls and bank account and one can mess up the system by accidentally allocating these to some other headings. Once these allocations are completed one then proceeds to a misleading function called "Amend layout of a/cs". Nothing of the sort as in fact one creates the format of printed accounts. This is where 135 nominal categories, which were automatically created, are utilised. The only puzzle is that there are eight pages of 15 headings entitled trading account, profit and loss 1, 2 and 3, fixed assets, liquid assets, liabilities and funded by (capital and net profit - the latter being automatically entered by the program). The question is - as these eight pages have 120 categories and 135 categories were created what is happening to 450 bytes ? Maybe I will discover when I get the full program. Having created the accounts one can then print them (not display). The final start up function is to enter the opening balances. Whilst doing so one can see on the top of the screen the net so there is help to see whether one is in balance at the end.

Operation of the program, that is posting invoices, credit notes payments or receipts or entering journal transactions, is an easy matter and one just follows the menu. The program also permits print of transactions either by dates or internal transaction numbers or both. There is no need to maintain any manual records. Trial balance and ledgers can be printed or dis-

played at any time. The accounts (trading, P & L and balance sheet) can be printed at any time and no printer output is provided at the update time. Annual update is a bit clumsy though as one has to transfer, using the Journal, all revenue balances to one profit/loss nominal.

The Try-Before You Buy comes with a well written instruction manual which covers only the aspects available on this test program. I even managed to get the accounts' print which apparently is not possible on this limited version. The manual leads one almost by hand through a specimen set of entries stating what key strokes to make etc. I have not tried to do this specimen set but, instead, I created my own accounts, entered my last month's balances and played with entering transactions until there was no more space. The postings file on the try version is very limited. Yes I think that I like the program and am going to get it. Customisation to CP/M+ is quite simple too, though I believe that the full program has a larger Install section. Well more about SAGE if and when I get it.

The last 80MICRO has a lot of nice things to say about CP/M+. Maybe the bugs, which are coming to light, will be corrected though the Tandy personnel in the branch I visit seem to be quite unconcerned. What Tandy will not do however is to provide support for a larger capacity disk drive. After all if they supported, lets say, a drive giving some 750K their chances of selling hard drives would be greatly reduced. I don't know much about hard drives for micros but I would think that to someone who wants to run Model III programs under LDOS, NEWDOS or DOSPLUS, Model 4 programs under TRSDOS 6.x and also CP/M a hard disk would not be of any use. But then a hard disk is generally beyond the means of the hobbyist and I don't know how many Model 4s are being sold for business use. I should not think that many as Tandy seem to be unable to get away from their hobbyist computer image. Their marketing strategy, in this country at least, is poor or to be more precise non existent. I also notice that there is a great turnover of the sales staff in their branches. The most experienced people seem to be leaving for other companies and they are being replaced by staff who do not know the equipment or the programs.

Well enough for this time - I am trying hard to fill a page or two in the NATGUG. I hate to see it being reduced so much in size. It is sometimes very hard to write especially if there isn't anything to write about (like this time). My communications attempts came to an abrupt halt with the relinquishment of the acoustic coupler, which has not been that satisfactory. I had a visit from my Cable and Wireless friend who produced an acoustic coupler (very much like Tandy's in looks but apparently much cheaper) and we contacted his account at Micronet (?) but I could not raise the Blandford board. There was a ringing tone but no reply. Once I gather some pennies and the modems will drop in price, as they are bound to, I will go for a good one with dialing from the keyboard and auto answer facility.

Anom.

PROBLEM SECTION

I would like to hear from anyone who has used a Silver Reed EX42 typewriter as a printer.

Norman Baust
(0329) 281480

BAR CODE READER

Does any member know where I can acquire (cheaply) a HEWLETT-PACKARD HEDS-3250 Bar Code Wand?

David Roberts

6 Plantation Road,
Ballymacormick Road,
Bangor,
Co Down.
BT19 2AF

Having a pressing need for a decoded version of BASIC, I ordered a copy of "Microsoft Basic Decoded" by James Fervour from Microcomputer Applications of Queen's Road, Blandford Forum. The price was £19.50 plus a £1 credit voucher for having previously bought their catalogue. The book I received was almost useless.

Only on pages 109,137 and 173 was the disassembly listing complete. All the other pages had the op codes listed but omitted the arguments to which the op codes applied. Of course, if you want to call subroutines in ROM you need to know which registers should carry the data on entry, and what is in the registers on Return. With the aid of a disassembler I could have gone through the book filling in the missing data by hand. However, it would have taken several hours, and is not on for a book costing £20.50.

The publisher stated that the book was intended to come apart, and was predrilled for loose leaf filing. The copy I received was not predrilled, but looked as if it would come apart. In the short time that I retained the book I noticed several errors.

I returned the book to the supplier and in due course received a cheque for £19.50, no refund of my £1.30 postage, no word of apology, no helpful suggestions as to where I could obtain a good copy - and they kept my £1 credit voucher! Hardly likely to encourage custom.

So if any member knows where I can get a complete version of "Microsoft Basic Decoded" or has a copy that is no longer needed, I should be pleased to hear from him/her.

Your correspondent Mr Curry was querying the Basicode Interface I agree with him. The circuit in the BBC booklet is deplorably illegible. I ordered the PCB from the Netherlands and received with it a perfectly clear circuit and parts list. I have not completed the board yet as I am having difficulty getting some of the components. I intend to fit it internally

to my GENIE I, but am not clear whether I have to make connections to the cassette recorder. The diagram shows recorder plug connections, but the recorder is a built in unit on the GENIE, which is a different kettle of fish as I am no electronics man. Do I have to connect to the optional external recorder plug? Can anybody help?

B.Mason

'Highlands' Tel:- 098 981 353
Bromsash
Ross-on-Wye
Herefordshire
HR9 7PR

(* Anyone building the BASICODE interface should avoid using the PCB obtainable from the Netherlands. The circuit is unnecessarily complicated. You should use the small circuit board, as described in the newsletter a few issues ago, which you can easily make yourself by painting the pattern onto copper laminate with cellulose paint and then etching in ferric chloride solution. Alternatively, you could wire the circuit up on a small piece of Veroboard. With the interface fitted, if you have a Genie, you must use an external recorder for BASICODE. LFH *)

I have a few queries which I hope someone in the Group may be able to help with.

1. Has anyone managed to run C/PM on a V.G. with the Gnomic 64k RAM extension? Presumably it is just a matter of writing a BIOS but I am rather ignorant on these matters. Any information would be appreciated.

2. There is a circuit for a simple modem in the March '83 Byte which uses the TI TMS99532 chip. It uses the Bell frequencies and I'd like to know what is involved in adapting the circuit to European standards.

3. Is anyone using Homelink connected to their micro? If so how about a review and details of the connection.

I have been considering whether to sell my V.G setup or upgrade, but there is no clear cut choice as far as I can see. The QL is, no doubt, a splendid machine but I fear I should regret going from disk to microdrive. There are more and more attractive machines coming on to the market, but, by the time one has added the price of disks and replaced the software it will be considerably more expensive than adding a hires graphics card or whatever it takes to upgrade the old system, not to mention the familiarity angle.

It would be intersting to see other members views on the subject.

Bernard Mantell

6 Manor Cottages,
Easton,
Winchester,
Hants.SO21 1EQ

Tel. Itchen Abbas 619

(* T.I. now produce a CCITT version of their modem chip. Try Quarndon Electronics on (0332) 32561 for further details. LFH *)

Two little things worry me that perhaps you or a member might be able to help alleviate. I have a Model 1 with Microfirm expansion to 48K, 2 Cumana Disk drives, and an Epson RX80F/T printer. I operate under DOSPLUS and my word processor is NEWSCRIPT.

When writing a collection of sections, I put the text onto a formatted disk in drive 1 that has on it no system code so that the disk takes quite a lot of text. I like to copy each file that I complete onto another format only disk as a backup. The only way I have succeeded in doing this is to have a transfer disk with DOSPLUS only on it, put this in drive 0, use TRANSFER :1 :0 to put the text onto the transfer disk, then put in the backup format only disk in drive 1 and use TRANSFER :0 :1 and the finally KILL the file on the transfer disk in drive 0. Is there a less tedious way? Copy1 is more tedious still.

The other thing is that when setting out forms I often want more than 64 character spaces. This is OK for the printer but means a lot of V+15 to shift the text on the screen and then V* to move it back to normal. Is there any way around this other than parting with Model 1!

Eric Brandes.

46 FIELD WAY CHALFONT ST PETER BUCKS. SL9 9SH
Tel:-Gerrards Cross(0753) 882982

In the May issue of NATGUG News Geof Smith comments that a number of assembler Patches have been Published in Previous issues for ScriPsit. Can anyone helP with a list of these Patches so I can buy any relavant back issues Please.

I use a Seikosha GP100A mk II Printer which will not reco9nise 9raPhic characters. As the LPRINT instruction is vectored it should be Possible to use an assembler Pro9ram to convert TRS-80 9raPhics to the Seikosha 9raPhics mode dot code. Has anyone done this already? If not are there any other Seikosha users who would be interested?

I find the absence of a £ si9n on the screen annovin9 on my Genie I. Is there an alternative UK character ROM available, or has anyone achieved this by any other means?

ALLAN FOWKE
28 QUEENSWAY
SAWSTON
CAMBS CB2 4DJ

(* Eric Hartley's lower-case mod. for the Genie and TRS-80 will give you a pound sign. You also get a much nicer character set into the bargain. The cost is about £12. Fitting involves a little soldering. Contact Eric on (0200) 22980. LFH *)

Having read a recent article in Acorn User regarding the use of a micro to decode and print out radio teletype transmissions, I am trying to find a source of software to work with the Scarab decoder, or any other decoder for that matter, and a TRS-80 Model III with disc drives. I wonder if there is anyone in the TRS-80 User Group who has experience of this application that you could put me in touch with.

Peter Earthy

I am trying to write a comparative report on Database programs so that would be purchasers can get a better idea of the programs that might be suitable for their specific requirements. What I am interested in is the uses of the programs. How well the specific program matches up to the task in real life and any deficiences?

I have copies of Maximanager, Datawriter, Microfiles, ISS, PIMS, SIR and a few homewritten programs for specific purposes. However I have only used these for my own applications and others may have had different experiences.

It is impossible to buy all the programs and to use them for long enough to find out their strengths and weakness and therefore I would welcome other members views. I have no experience of relational databases or the new Profiles. I am also interested in hackers that have modified programs or have used the program beyond the obvious facilities in the specification.

My own Mod 1 has two 40 track double density drives but I have used tapes and a Stringy floppy. How do others feel about speed-ups and small memory machines for database work? How does the TRS-80 compare with other computers? Is the 4P as useful as it appears for this work or are there snags? Do members use more than one program?

Derek Trayler

88. Grosvenor Drive,
Hornchurch,
Essex RM11 1PW.

I was interested in John KilPatrick's comments on Model I software in the May issue of NATGUG News

I have been trying for some time to get a copy of Little Pascal for a tape based Model I without success. If anyone has this program I would appreciate the chance to discuss ways of taking a copy.

Anyone who can help Please ring me on (0223) 832308

Allan Fowke

ANSWER/ORIGINATE MODEM

In the For Sale and Wanted section is an advertisement for Peter Hall's modem PCB. I saw Peter's prototype modem at the MK workshop last year, and tentatively suggested that he might care to make the PCBs available, so that members could build their own. Peter subsequently had 50 boards made, at his own expense. He has made a very nice job of it; the board is neatly laid out, and the assembly and alignment instructions are comprehensive and clear, provided you have some experience of electronic construction. Peter gave me one of his PCBs, and we shall be putting the modem together as a group project at the Swindon Workshop. A nice touch is the use of the TRS-80 (Model I, III or IV) as a tone generator, for setting up the modem. Here is Peter's program:-

```
10 'MODEM SET UP PROGRAM TONE GENERATOR   PETER HALL  JULY 84
20 'TONES WILL BE GENERATED FROM THE GREY "AUX" CASSETTE PLUG
25 'THIS PROGRAM WILL OPERATE ON A MODEL I/III OR IV .
30 DEFUSR0=&H7F04
40 A=&H7F04
50 FORX=1 TO 32
60  READ D
70  POKE A,D
80 A=A+1
90 NEXTX
100 D=30
110 CLS:PRINT"*** MODEM SET-UP TEST PROGRAM ***":PRINT:PRINT
120 PRINT"     1.  1750HZ"
130 PRINT"     2.  1080HZ"
140 PRINT
150 PRINT"     3.  1180HZ"
160 PRINT"     4.  980HZ"
170 PRINT"     5.  1850HZ"
180 PRINT"     6.  1650HZ"
190 PRINT:PRINT:INPUT "SELECTION";S
200 PRINT:INPUT "DURATION";D
210 M=PEEK(84):IF M<>1THEN290
220 IF S=1 THEN F=1750
230 IF S=2 THEN F=1080
240 IF S=3 THEN F=1180
250 IF S=4 THEN F=980
260 IF S=5 THEN F=1850
270 IF S=6 THEN F=1650
280 GOTO 350
290 IF S=1 THEN F=1510
300 IF S=2 THEN F=940
310 IF S=3 THEN F=1020
320 IF S=4 THEN F=860
330 IF S=5 THEN F=1600
340 IF S=6 THEN F=1418
350 IF S<1 OR S>6 THEN 110
360 FC=(1/F-31E-6)/14.66E-6
370 DC=D*F-1
380 POKE &H7F00,DC-INT(DC/256)*256
390 POKE &H7F01,INT(DC/256)
400 POKE &H7F02,FC-INT(FC/256)*256
410 POKE &H7F03,INT(FC/256)
420 X=USR0(0)
430 GOTO110
440 DATA 243,42,0,127,237,75,2,127,6,0,17,255,255,65,62,1
450 DATA 211,255,16,254,65,62,2,211,255,16,254,25,218,17,127
460 DATA 201
```

80-NET OPERATIONAL AGAIN

80-NET, the Group's computerised bulletin board system is operational again thanks to Mac Randle. The number is (0926) 39871, and the system is operational from 23.00 to 00.00 every evening. As before, the relevent parameters are: 300 baud, even parity, seven data bits, one start and one stop bit.

A COMPUTER CONTROLLED FREQUENCY SYNTHESISER

Now that I have an amateur radio licence, I have decided to build my own 2 metre VHF receiver and transmitter.

The heart of the rig will be a microprocessor-controlled phase-locked loop frequency synthesiser. At present I have an experimental 35MHz - 70MHz synthesiser interfaced to my Model I. It uses a Motorola MC145156 phase-locked loop synthesis chip (this device is intended for microprocessor control), an MC1648 ECL voltage-controlled oscillator tuned by a hyper-abrupt variable capacitance diode, an SP8647 ECL dual-modulus prescaler, an op amp active loop filter, a broad-band amplifier and a few other bits and pieces.

Because the Model I generates a colossal amount of RFI, the final version will use a Motorola CMOS processor, but for experimental purposes the Model I works very well, with the MC145156 being controlled via three bits of an output port (I could have used the printer port), with one bit acting as a clock line, another transmitting serial data, and a third bit generating an enable signal. The relatively simple software needed to control the synthesiser was written using Macro-80. If anyone is interested I'll publish the circuit in the next issue. It'll cost about £40 or so to build, and is capable of generating frequencies from LF right up to 250 MHz or so!

Leon Heller

Super Scripsit Space Code Changes

The standard version of Super Scripsit uses an underline character code on the screen as "shorthand" for every two spaces together. This has the disadvantage that the screen layout then does not match the printed layout.

To overcome this, Peter Knaggs has followed up work by Mike Adams to locate the offending code and change it. After amendment, as described below, spaces are not compressed and screen layout matches printer output for all except proportional spaced printing. SCR35/CTL handles the Scripsit/ASCII conversion and SCR64/CTL handles keyboard text input. In each case a conditional jump is replaced by an unconditional jump, removing the test for double space.

Zaps are as follows:

SCR35/CTL FRS1 30H change byte from 20 to 18
Context: from 2EH onwards FE20 200B 2BBE

SCR64/CTL FRS0 38H change byte from 20 to 18
Context: from 36H onwards reads FE20 2007 36F7

Tony Cottingham

21 Sandalwood Road
Loughborough
Leics LE11 3PR

TANDY MODEL 100 AND THE BBC

Having recently obtained a BBC I was soon attempting to sort out the problems of making the two computers talk to each other.

A cable was made up with the required plugs and wired as in the diagram.

Searching through a number of bits of information I tried several things then after a struggle was given a lead,which in fact took me to the Advanced Users Handbook and the following method was arrived at.

My main concern was to be able to type in programs on the 100 ,upload to the BBC and save on disk,to be later downloaded to the 100.

This way the 100 has the facility of a monitor and a disk without buying the expensive Tandy interface (providing that you have a BBC with disk!).

To transmit from the 100 to the BBC

There are two ways;

1) a. Enter TELCOM mode
   b. STATUS 37I1E

   BBC Break
   *FX7,3
   *FX2,1

   ON 100 ENTER UP MODE
   Type in file name(which
   must be a .DO file)
   Press ENTER and file is
   transferred.

   ON BBC press BREAK
      Type OLD
      LIST
   program is there to be saved to disk or if in BBC syntax may be run.

2) If a BASIC program is to be sent then on the 100 you can use,

   SAVE"COM:37I1E"
   and when ENTER is pressed the transfer will be made

   On BBC proceed same as above.

To transmit from the BBC to the 100

On BBC,
   BREAK,Load program to be sent
a) Type *FX8,3 (BAUD RATE 300)
        *FX5,2 (OUTPUT PRINTER)
        VDU2   (ENABLE PRINTER)
     Type L. but do not press

On 100
b) Enter TELCOM MODE
DOWN MODE
Give a File name
Press ENTER

On BBC press RETURN and program
will be transferred.

On 100 press F2(DOWN) to clear. The program will be saved as a .DO file.Look at the .DO file and edit any L. or other spurious characters before the start of the program, then to get it to BASIC, enter BASIC, type NEW to clear any existing program in memory, type MERGE"RAM:progname.DO" and it will be moved to BASIC. For some reason a ?DS error shows, but do not worry. It may now be Listed and Saved as a BASIC program.

I found the 300 Baud speed the most reliable,although transfer from the 100 to BBC can be done at 1200 or 9600 but from BBC to 100 only the 300 rate seems to work every time.

There are supposed to be other ways into the 100 such as

MERGE"COM:37I1E" but this does not appear to work.

Other experiments,
with the 100 in TELCOM/Half mode (Baud rate etc can be 87I1E) and the BBCwith*FX2,1, then the 100 can write direct to the BBC in text or programs direct, even sending control messages such as RUN,CLS,LIST and in fact it appears that most commands can be sent even graphics,so changing the colours on the BBC direct from the 100.

With the 100 in TELCOM/Half mode (87I1E) and BBC with Termi program(see below) running, then the two computers can talk to each other via the keyboards but no direct control commands can be sent,just text,both screens echoing its own keyboard.
NOTE: Line feeds work from the 100 to the BBC but not from BBC to 100 so to line feed from BBC to 100 requires the use of the space bar. From BBC to 100 Delete works but not from 100 to BBC,although the cursor moves back the character is not deleted,so to delete a character requires a backspace then a spacebar.
No limit to the amount of text and all characters from the BBC are accepted by the 100 including curly brackets,square brackets but no pound sign.
If in this mode the DOWN button is pressed anything coming from the BBC will be saved as a .DO file and can then be read latter.

So there we are,I hope that these notes are of use to members.
In fact when you plough your way through the 100 handbook and the BBC User and Advanced User books ,virtually all the information is there but it has to be dug out.

Model 100 see,
LOAD:COM: page 138
SAVE:COM: page160
MERGE:COM: page 142

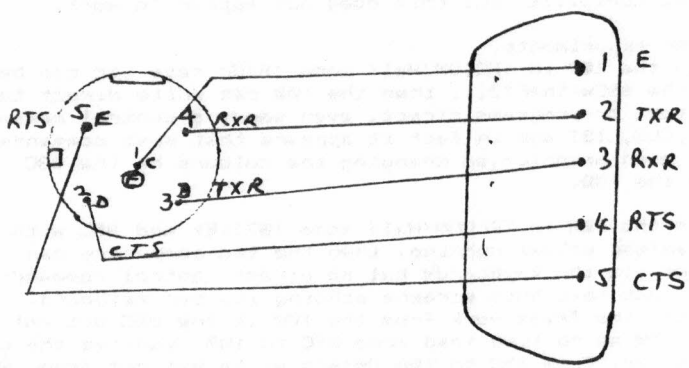
BBC Advanced User book,
*FX2 page 118
*FX5 page 121
*FX7 page 123
*FX8 page 124

TERMI.....

```
10 CLS
20 REPEAT
30 IF(?65032 AND 1)=1 tx=?65033: VDUtx:IF tx=13 PRINT
40 Z$=INKEY$(0):IF Z$<>""  ?65033=ASCZ$:PRINTZ$:IF
Z$=CHR$13 PRINT
50 UNTIL 0
```

Text files are still a problem and any comments or other ideas will be welcome,the 100 is a great machine with good potential. Wonder when ROM for underneath will appear?

RAY COX,LEICESTER



Spooling DOTPRINT printer output to disk (part 1)

This article is the result of one of those things that could only happen at a NATGUG meeting. There I was quietly minding my own business at an early stage of the weekend, when I noticed a printer printing, albeit rather slowly, something that looked very like circuit diagrams. Being both a hardware & software freak, this amazed me and I had to enquire further. It turned out to be an ingenious use of DOTPRINT, where Tony Evetts had gone to a lot of trouble to actually design a font set with fragments of circuit symbols. Not surprisingly, Tony had been asked to give a talk on DOTPRINT. He commented that it was a pity that the program processing was so slow that his lecture was going to take ages. Like an idiot I suggested that it would be possible to do the printing to a disk-file beforehand, and then print this out. I think Peter Knaggs agreed, and before long the idea had become a requirement. Later on in the day, Herr Brian issued a three-line whip about an article, so here we are !!

A couple of comments however. The LDOS cogniscenti said all that was necessary was a bit of device linking. True (nearly - see later), but of course it just happened that we were using NEWDOS/80. Perhaps this caused the use of a bit of a sledge-hammer to crack a nut, but we'll maybe learn something in the process. The other thing is that I've presented the software exactly as written (with some extra comments). There is a temptation to tidy things up and make a few alterations, but I know that this code definitely worked on Tony's Model 1. Unfortunately I can't remember which version of DOTPRINT he had - I think that different releases may require different code.

The first idea was to use NEWDOS's ROUTE PR,MM=hhhhH command which allows a limited amount of device redirection from the printer to a user-written driver at hhhhH. Suffice to say that we actually got this going, and were able to send simple printing to disk. Then came the big test - some DOTPRINTing. Everything seemed to go fine except the "printing" still came from the printer. Oh dear - and then the penny dropped. DOTPRINT's machine code (WPML/CIM) included its own printer driver, something I should really have guessed. Incidentally, this would also have nobbled LDOS LINK idea. If you think about it, bit image printing can produce all of the codes from 0 to 255. I seem to recall someone writing in the Sept '82 issue on the fact that the normal ROM printer driver doesn't cope properly with CHR$(0), (11) and (12).

Anyway, a new approach was now needed. The first thing was to track down the printing bits within WPML. Everybody has their own favourite program and technique. I used TASMON and then SUPERZAP. So as not to take up too much magazine space, I'll not go into this in detail. At the end of the day, all I did was some zaps on WPML so that it referred to my new driver. Before going into the listing, the only other thing concerns closing the disk file. When you write to disk, the file eventually has to be closed, but unfortunately there is no easy way the software can tell when you've finished. To get round this, I added a little extra keyboard driver that looks for a triple key depression <,./>. This stops the writing to file and closes it.

```
;**** DP2DSK/SOF ****
;**** 07/27/84 ****


;********************************
; SET UP DISK & KEYBOARD DRIVER
;********************************

          ORG  0F400H
          EXEC START

START:    EQU  $
          LD   HL,MSG          ;POINT TO THE MESSAGE TEXT
          CALL 4467H           ;DOS CALL TO DISPLAY IT

          LD   HL,(4016H)      ;GET ORIGINAL KBD DRIVER
          LD   (KBDR1+1),HL    ;PUT IT ON THE END OF OURS
          LD   HL,KBDRIV       ;GET OUR DRIVER ADDRESS
          LD   (4016H),HL      ;AND ACTIVATE IT

          LD   HL,SBUFF        ;A BUFFER FOR DISK SECTORS
          LD   DE,FCB          ;FILESPEC WILL BECOME FCB
          LD   B,1             ;A LOGICAL RECORD LENGTH OF 1
          CALL 4420H           ;DOS CALL TO OPEN THE FILE
          JP   Z,402DH         ;BACK TO DOS - NO ERROR FOUND

          JP   4409H           ;DISPLAY ERROR MESSAGE & QUIT


SBUFF:    DS   256             ;TO KEEP 256 BYTE SECTORS
FCB:      DB   'PR2DSK/FIL',13  ;NAME OF THE FILE

          DS   21              ;FCB NEEDS 32 BYTES TOTAL

MSG:      EQU  $
          DB   28,31,'Loading printer data to disk file',10
          DB   10,'Press <,./> to close disk file',13


;******************************
; BIT TO WRITE CHARS TO DISK
;******************************

          ORG  0F800H

DSKPRNT:  EQU  $
          PUSH IY              ;SAVE ALL REGISTERS
          PUSH IX
          PUSH DE
          PUSH BC
          PUSH AF
          PUSH HL

          LD   DE,FCB          ;NEEDED BEFORE FILE ACTIVITY
          CALL 0018H           ;CALL TO WRITE ONE BYTE
          JR   Z,DSKP1         ;NO ERROR

          OR   0C0H            ;TO MAKE THE ERROR DISPLAY WORK
          CALL 4409H           ;DISPLAY IT

          LD   DE,FCB          ;CONFIRM DCB POINTER
          CALL 4428H           ;CALL TO CLOSE THE FILE
```

```
62
63              JP    402DH           ;AND BACK TO DOS
64
65
66 DSKP1:       EQU   $
67              LD    A,(3C3FH)       ;PEEK TOP RHS OF SCREEN
68              XOR   0AH             ;SPACE TO ASTERISK OR VV
69              LD    (3C3FH),A       ;POKE IT BACK
70
71              POP   HL              ;RECOVER THE REGISTERS AGAIN
72              POP   AF
73              POP   BC
74              POP   DE
75              POP   IX
76              POP   IY
77
78              RET                   ;AND BACK WHENCE WE CAME
79
80
81 ;************************************
82 ; THIS IS THE EXTRA KEYBOARD DRIVER
83 ;************************************
84
85 KBDRIV:      EQU   $
86              LD    A,(3820H)       ;LOOK AT A KEYBOARD ROW
87              AND   11010000b       ;MASK OUT THE KEYS WE WANT
88              CP    11010000b       ;WAS IT ",./" ?
89              JR    NZ,KBDR1        ;NO
90
91              LD    DE,FCB
92              CALL  4428H           ;CLOSE THE FILE
93              JR    Z,KBDR2         ;CLOSE WENT OK
94
95              OR    0C0H
96              CALL  4409H           ;DISPLAY THE ERROR
97
98 KBDR1:       JP    $-$             ;OLD DRIVER ADDRESS HERE
99  .
100 KBDR2:      EQU   $
101             LD    HL,(KBDR1+1)    ;GET ORIGINAL DRIVER ADDRESS
102             LD    (4016H),HL      ;RESTORE IT FOR SUBSEQUENT USE
103             JP    (HL)            ;AND GO TO IT FROM HERE
104
105             END
```

Exec Addr F400

I can think of one or two other notes. The code is somewhat untidy - one example is the existence of two origins. I can't really remember why this happened, but as it wasn't too clear where the gaps were in the WPML code, the choice was arbitrary. I put in a little bit that blinks an asterisk at the top right of the screen when file writing is taking place. The experts will recall that there is a bit in ROM that does this already - quite right, but I couldn't remember where it was !! I confess that the whole story is a bit incomplete, but I hope that the bits are of interest. Clearly something more is needed - the bit that prints the disk-files out. I will make this the topic of Part 2.

Don Bannister Tel: 0232- 668899

FOR SALE AND WANTED

Build yourself an answer/originate modem for £30.

PCB with full instructions £7.50 inclusive.

You supply components, case and power supply. No special test equipment needed: filter alignment can be performed using your TRS-80 as a tone generator, with output via the cassette port, using a simple program (listing elsewhere in this issue).

The modem may be used direct-coupled, or acoustically-coupled, to the Public Switched Telephone Network.

Coming soon:-

Bell 103 compatible version.

A version for a CUTS/Kansas City general purpose cassette interface.

RTTY interface.

All these options will only involve changes in component values, the basic PCB remains the same.

Peter Hall,
29 Clyde Road,
Brighton,
Sussex BN1 4NN.
Tel: (0273) 684286

FOR SALE

TRS-80 48K, with monitor, cassette recorder, books and software. 4/5 years old.

D. Doherty,
30 Hoylake Road.
Acton,
London W3.
Tel: 01-743 7633

FOR SALE

VG 3003 + sound (16K) £60.
Expansion box (32K) £80.
Ikegami 9" B/W monitor £60.
Microware twin 40 track drives inc. power supply £180.
Will take £300 for the lot!
Many books and software to go with the above.

Tony O'Donovan,
23 Tintagel Crescent,
London SE22.
Tel: 01-693 3975

SPECIAL OFFER - 20% DISCOUNT

SANYO 16 BIT MICRO WITH MSDOS AND 4 FREE SOFTWARE PACKAGES

The MBC-550 with the Intel 8088 chip as used by the IBM-PC offers 128K RAM (expandable to 256K), two 160K disk drives and comes with 4 major software packages from Micropro FREE: Wordstar word processor, Mail Merge for mailings, Spellstar word spelling checker and Calcstar

The discount price for all this including a monitor is:
£1035.92 with a monochrome monitor

or
£1240.16 with a colour monitor

Both prices INCLUDING VAT.

Only 6 monochrome and 1 colour systems are available at this price which strictly are being sold together (total £7455.68 including VAT) and can only be offered because of stock clearance by a Sanyo micro dealer.

However provided firm orders are received from individuals which will cover the 7 systems, the stated individual prices will be honoured.

The machines will be supplied from the dealer with full 90 day warranty.

Such prices are cash and carry only and apply for payment by cash, bankers order or previously cleared cheque. The normal collection point is the Heathrow area but it may be possible to arrange it elsewhere, for example, the Brighton area. Otherwise delivery can be arranged for a charge.

If interested contact:

R.E.Wiggins on 01-920-8211 (weekdays) or Hassocks 3430 (evenings, weekends).

Or write to "Dilkhush", North Bank, Hassocks, W.Sussex BN6 8JG

WANTED

Copies of 80 Micro from 1st. issue, Jan. 1980 to Jan. 1982 inclusive, in reasonable condition.
Good price paid.

Geoff Bazin,
Morbec Honey Farm,
Arterial Road,
Wickford,
Essex SS12 9JF.
Tel: (0268) 726296

FOR SALE

Tandy Keyboard to Printer Interface cable. Enables connection of Parallel Printer (Centronics Standard) to Model I keyboard WITHOUT THE NEED FOR AN EXPANSION INTERFACE. Genuine reason for sale (I have bought a combined disk/printer interface and no longer need this one).

The interface (Tandy Cat No. 26-1411) is boxed with full instructions and comes complete with approx 6ft printer cable. It takes its power from the keyboard and/or printer and needs no power supply of its own.

Price £30 ONO. Contact Adrian Hancock on Basildon (0268) 288893 after 6.00 pm. for details.

FOR SALE

Two full height 40 track disk drives (TEAC FD-50A) in cabinet with power supply (CUMANA) and interface cable for model 1 expansion interface. - 225 Pounds.

Tandy Quick Printer 2, prints on 60mm wide aluminium coated paper, will connect direct to Model 1 keyboard or via serial or parallel interface. No interface cable but 3 rolls of paper supplied. - 35 Pounds.

Prism acoustic modem 75/1200 baud. - 35 Pounds.

Telephone Graham Yates. (0302) 864279 after 6pm.

RE-INKING SERVICE

**Re-inking service for Printer Ribbons and Cartridges
For Quotation send SAE with details of Printer Model
and ribbon Ref.No.**

**DAVID LOCK, Specialist Technical Services,
23 Regent Road,
Kirkheaton,
Huddersfield,
HD5 OLW Tel.No. 0484-42686**

FOR SALE:

EPSON MX80 F/T printer with Graftrax 80 - £195.

Bob Steed
Kenilworth 511860

WANTED

Any copies of 80-MICRO from issue #1 to #19
&
MACHINE LANGAUGE I/O & OTHER MYSTERIES

Ken Robinson,
65 School Drive
Newton Longville,
Milton Keynes
MK17 ODD (0908) 75479

FOR SALE

Model III or Model 4 Disk Controller Board.
Will control upto 4 5" 40-track or 80-track single or double sided drives in any combination.
Complete with one 40-track Double-sided drive (MPI) in good condition and recently professionally checked for alignment and TRSDOS V1.3 patched for two-sided operation.

£150.00 ovno.
Robert Vieira.
Telephone 01-500 5594

*6, Lechmere Avenue
CHIGWELL
Essex
IG7 5ET*

LAST AND FINAL OFFER

High Resolution Graphics Option for the Model 1.

As sold by Tandy for 169 Pounds !

The original manufacturer of these units wishes to sell the last of them that he has available. Once he has done this there will never be any more available,ever !
So,hurry,this may be your last chance. First come,first served !

You can buy a kit for £59-95 ;this includes full fitting instructions. Or you can have it fitted free of charge if you are nervous of computers !

For further details contact Mike Gibbons 021-747 2260 anytime.

(Yes you did read the price correctly £59-95 including fitting - about a third of Tandy's price !)

FOR SALE. Video Genie with 48k Gnomic ram extension, Gnomic DP2000 single sided double density disk drive, ICL green monitor, SmalLDOS, Word4wordsuperplus, Toolkit, etc. Price £350.
Bernard Mantell
6 Manor Cottages, Easton, Winchester, Hants. SO21 1EQ.
Tel. Itchen Abbas 619

FOR SALE

TANDY TRS-80 MODEL 4

64K 2-Disk Version

This machine was purchased new 5 months ago.

Machine comes complete with original packing,
TRSDOS 6.00/Basic Interpreter,
Disk System/Owner's Manual and Dust Cover....

At Very Reasonable Price..... £925.00

Please give me a call at home :- HATFIELD 68419
(Evenings/Weekends up to 10 August AND
then from 2 September onwards.)

Milton Keynes Workshop July 1984

This was held for the second time at local community centre here in Stony Stratford.

I added a fourth day to try out the idea of putting on three specific courses, with the normal workshop on the Saturday and Sunday. The business content has now been moved to Swindon because of the better facilities available there. If I am honest the idea was not a great seller and perhaps the current membership is just a little too small to support these ventures.

cont P33

TANDY DMP2100 PRINTER

Brian asked me some time ago to write about my impressions.

I am still struggling with it. Using it for printing routine data is fine. Using it for simple letters is OK. **BUT** problem one is the "Top of Form" feature whereby it keeps track of the number of lines printed and if a CHR$(12) is output, throws to a new page. With my Model 1 in BASIC all CHR$(12)'s are converted by the computer into a series of line-feeds by calculating the difference between PEEK 16424 and PEEK 16425. So, no problem. But the Electric Pencil has its own driver and, I assume, DOES put out CHR$(12)'s (or something) which causes the printer to do a page throw at awkward moments. I invested in NEWSCRIPT with which this has been prepared. I was assured it had been modified to cope with a DMP2100. In my opinion, it hasn't. Partly because being written in Basic it is slow and can't feed the printer fast enough, but mainly because any writer of a word-processing program is up against a problem in the DMP2100:

The printer has 2 modes: Data Processing and Word Processing. All print fonts are available in both modes. The results of most paper feed instructions are different in the 2 modes. Half Forward Line Feed, Half Reverse Line Feed and Three-quarter Forward Line (for example) are set under Data Processing Mode when the control characters are output ie. the status of the printer is fixed at say ½ Forward Feed (like this article) until a control code is fed to change it. If the printer is in Word Processing Mode and a control code is fed to it, that code is immediately executed **but the printer stays in the 6 lines to the inch spacing mode.**

NEWSCRIPT puts the printer into the Word Processing Mode so, to print this effort at 8 lines to the inch, I had to incorporate into the text a control code to switch the printer back into Data processing Mode. If I then want to do something like a Half Line Feed at the end of each paragraph, I have to change the printer back to Word Processing Mode; output the Subscript Code available in NEWSCRIPT; switch the printer back to Data Processing Mode. Oddly enough these changes do not change the printer out of 8 lines to the inch mode.

Please would somebody tell me why it is necessary for there to be a Data Processing Mode and a Word Processing Mode. Why could there not have been one Mode with different codes for the paper feeds as to whether the code is to change the status of the printer or the code is to be executed as a one-time action.

Despite the above I like my printer and NEWSCRIPT but would be very wary about buying another word-processor. I read in the July issue about LeScript. **But** with an EPSON.

Also in the July magazine mention was made of the difficulty in reading some of the copy after it had been reduced. The following is page 16 printed on the DMP2100 at 8 lines to the inch and proportionally spaced. As you can see the text fits into very little more space. (The fractions ¼, ½, ¾, are in the DMP2100's repertoire.)

The next line is "jl" or "justify left" - each line starting in the first line like this. You can have "jr" for justifying on the right side only, or"jb" to justify both sides,as in a printed document like this, or "jc" for justify centre" - in other words print everything that follows in the centre of the page. Any of these commands can be put anywhere in the document (on a separate line from the text) and they apply from there until the next change.

The next line in my header file is a mixture of characters and hex numbers. The first one you get by pressing SHIFT-@,and the rest are the hexadecimal values for EPSON FX-80 printer control codes to go into superscript, to print "1", to backspace, to come out of superscript mode, to print a minus sign, to backspace, to go into subscript mode, to print a "2", and to come out of subscript mode - in other words, to print "½". That is the definition how to print a half as a vulgar fraction - something I use quite often. Once that has been defined LeScript will print a "½" every time it comes across the character "\" in the text. You can use any one of 54 different keys to define special characters that way. Then there is a comment to remind me how to get the half in case I forget. Comments are not nearly as neat as in NEWSCRIPT where you just precede a comment by ".cm". You have to start with "off" to tell the printer to ignore what is coming, then write the comment, then "on" to tell it to resume printing.

The ability to use programmable function characters is really a very powerful thing because it means that if there is anything that your printer can do - anything at all - for which LeScript has not got a direct command, you can get LeScript to do it in the middle of your text - even your own designed graphics graphics.

Well, that's the way I do my printing set-up instructions, but you can also use the KSM, or Key Stroke Multiply, facility. With this you can programme 55 keys so that each one represents a whole phrase, sentence, or even paragraph. You use SHIFT-CLEAR-A-Z SHIFT-CLEAR-1-5, and all the punctuation and other special keys. If you want to include end of line markers - "ENTER", in other words - then you use a ";". So, the whole of my header file could be store in the KSM file as a single keystroke (plus SHIFT and CLEAR). They do not say in the manual what the maximum length of each one can be, but the sample one which comes with LeScript contains some which are nearly 500 characters long.

This file, LESCRIPT/KSM, can be created and edited just like any other LeScript document, and if you wish to miss any letters out you just use an end of line marker.

There are about 30 separate printing commands covering such things as:-
AUTOMATIC LINE-FEED INSERTION if your printer needs that as well as a carriage R.
The BAUD RATE setting for SERIAL PRINTERS.
CHARACTER DENSITY for PROPORTIONAL SPACING. I understand from a friend that the left & right justified proportional spacing is absolutely accurate to the last dot, which, as a publisher, is one of his top priorities in a word processor. Part of a document can be proportionally spaced if you like, as this paragraph is.

TYPESETTING CONTINUED

It is just possible that there are still a few readers who are waiting with bated breath to learn how the saga of my typesetting adventures for my Bibliography have been progressing.

Devotees of the subject may remember that I was able, with some difficulty and by taking my machine to the Sun Printers in Suffolk, to tranfer the data files from the RS-232 port in thence via an extremely complicated piece of software and coding to a typsetting machine.

I felt that there must be an easier way. Enter Serious Software of 55 East Street, London N1 (Tel. No. 253 8196. Contact Derek Kaye). Their original solution was to use a 'black box' that claimed to be able to read any disc from any machine. In practice it would only do this when an engineer, flown all the way from America, was holding it's hand, when he left this method no longer worked. Enter now the final panacea XENO-COPY.

XENO-COPY produce a number of programmes that enable alien disc operating systems to be readable by an IBM PC. Amongst these is a modified version of TRSDOS 1.3 called TRSDOS 1.3X. All that is required is for this modified version to be booted-up in a Model III and any data file you want converted COPIED to this 1.3X disc. To all intents and purposes this modified disc behaves just like an ordinary TRSDOS 1.3 in that other discs can be FORMATTED and BACKED-UP to produce additional versions for other data files that require copying.

This 1.3X version is now loaded into an IBM PC which has been BOOTED with the Master XENO-COPY and lo! and behold your data is now loaded in the IBM. This ability could of course be used for a number of other purposes apart from typesetting.

With my text data now on the IBM it can easily be transfered to a LINATRON 202 typesetter using CORA which is a typesetting software interface. The principle is fundamentally very simple as it only requires the insertion of, say {M18H8F88} to produce Measure 18, Height 8 (pica) Font 88=Optima Bold. In addition however all other standard typesetting facilities are available for those who need them. In practice all these codes were added by myself before transfer from TRSDOS.

I enclose a sample of the coded text and its typeset appearance.

The cost of this service is 25p per 1000 characters plus the cost of the Bromide print-out and any initial translation work required if non-standard codes are used for International characters.

Chris Philip
19 Coolhurst Road
London N8

(* An excellent typesetting service is offered by Wordsmith. They can accept Wordstar files, with special Wordstar-like codes, which they can then translate into the codes used by the Linotron photo-typesetter. The price is 75p per 1000 characters. They can even accept files via the 'phone: I used this method recently when I needed a small job doing very quickly. Their 'phone number is (0458) 45359. LFH *)

```
<<
{H11F731}A, J.L.y.{F729}  See {H11F731}CHERTIER, F. M{F729}<
<<
{H11F731}ACKALUSKY, James{F729}<
<
{H10F729}Handbook for Making Rockets and Fireworks.<
{A6}
{H9F731}0000.1<
{F729}Pageant Press: New York 1960<
^T27p. illus.  21cms.<RLA011000<BL\-<++
<
NUC Subject Catalogue. 1960<
<<
{H11F731}ADRIANUS, Romanus.{F729}  See {H11F731}ROMANUS, Adrianus{F729}<
<<
{H11F731}ALBERTI, Guiseppe Francesco Antonio.{F729}  (1712-1768)<
<
{H10F729}La Pirotechnia: o sia Trattato dei Fuochi d'Artificio.
{F730}[Pirotechnics: or a treatise on artificial fireworks.]{F729}<
{A6}
{H9F731}0000.1<
{F729}G. Battista Recurti: Venezia MDCCXLIX (1749)<
^TTP+vi+128+21Pls.+ii(errori).  24cms.<NA0139502<BL8715.ff.38<++
```

A, J.L.y. See **CHERTIER, F. M**

ACKALUSKY, James

Handbook for Making Rockets and Fireworks.

A010.1
Pageant Press: New York 1960
27p. illus. 21cms. RLA011000 BL -

NUC Subject Catalogue. 1960

ADRIANUS, Romanus. See **ROMANUS, Adrianus**

ALBERTI, Guiseppe Francesco Antonio. (1712-1768)

La Pirotechnia: o sia Trattato dei Fuochi d'Artificio. *[Pirotechnics: or a treatise on artificial fireworks.]*

A020.1
G. Battista Recurti: Venezia MDCCXLIX (1749)
TP+vi+128+21Pls.+ii(errori). 24cms. NA0139502 BL8715.ff.38

from p29

Discussing a workshop policy with those members who very kindly stayed behind to clear up it was agreed to have two weekends next year - Swindon in March'84 and Milton Keynes in late August'84. This will space them out at around six months apart

Back to the July meeting. Leon did his two day course on the 68000 and Peter Knaggs with some assistance from Adrian Sims did his course on Z80 graphics. His programs were quite something from a teaching point of view, and the machine code version was quite exceptional in its documentation. My accounts using Sternbergs course had no takers, which I thought may have attracted 3-6 members The systems room over the weekend was a little different to some in the past There was an awful lot of serious activity. Model 4's also in serious use, Model 1 being used as a development system reading Model 2000 and IBM PC disks, machine code routines being written for DOTWRITER. The latter involved Tony Evett, Don Bannister, Geoff Smith and others.

cont'd p35

SCRIPSIT +

Having only just moved from Floppy tape to a disk system, curtesy of Gnomic's £25.00 uncased interface, I thought I would note down some observations on the program SCRIPSIT + as sold by another member, Geoff Smith.

On initalisation Scripsit + asks if you want to try to recover any text lost after reboot. If you do there is a two character entry to try to get the text back. Geoff recommends that you disable the driver within DOS on the Scripsit + disk as this interferes with both this command and the syntax of the help command. Of course Scripsit's own lower case driver takes over.

First of all a help file is called using a new control character, @H and as the file is a file in Scripsit format it can be extended as required.

There is also a word count facility called from the command line using (break) ?L which counts the words up to the current cursor position. A must for some students!

This imolementation of Scripsit has the usual facility of most extended Scripsit versions of being able to inbed control codes witnin a text line. However my printer, a Microline 82 A, is not able to make as full a use of them as would a newer orinter. This facility is used by just typing @P control code @P then carrying on with the text. In order to keep compatability with old files a format line may be entered containing printer control codes using the normal Scripsit format line mode with PC to signify a control code as in this example:) PC=30 .

There is also a neat facility to enter DOS commands from the command line by preceeding them with / as in :- /DIR 1. Having come to Radio Shacks Disk Scripsit from Laurie Shields @Scrip I found the lack of a directory command very frustating. Now this command allows quite a nice trick, you can type:

/LIST filename/ext:1

and the specified file can be read as the NEWDOS LIST command. Pauses are caused by pressing the right arrow, enter causes the listing to continue and the up arrow terminates the list. So you can be working on one text file and read another file part way through without losing your place in text or moving the cursor. How can one do without this? How did I do without it before?

Now we start getting to the bits which Geoff must be pleased with. There is an ability to use Scripsit to build Macro files: these are nothing to do with the discussion 'Macro's in assembly' so loved by Leon; but an ability to write then execute a Scripsit file which is used to modify a previous file already readable by Scripsit.

Say you have been a bit stupid and decided to write a BASIC program using a lot of DATA statements which you've already typed in. You now decide that all the numeric data should have been typed as a string representation of the respective numbers:- ,"33", instead of ,33.. You could of course retype all the data or do what I did.

Save your BASIC program using:-
SAVE"filename/ext:1",A

from BASIC, in this state Scripsit can read the file and operate upon it. Make any modifications within Scripsit you require to other parts of the code, correct typing errors etc. Then execute a macro called say "MACRO/SCR" by typing (break) to get onto the control line then typing X MACRO/SCR. If your macro file contains the correct code all the " are put in for you. To confirm that all is well, the format line :

)Macro has been done

is placed at the start of the Scripsit page. Now this file is saved using:-

S,A filename/ext:1

then loaded back into BASIC as normal. A very good example of a complicated macro file is included with the program and its documents.

He has also added a command @0 which just repeats the last command. This is useful in that after finding one occurrence of say 'Scripsit' using (break) F)Scripsit then pressing @0 all the others can be viewed in order, this may be a faster method of doing a global replace for a small number of items than to use a macro file.

Another of Geoffs little mods is to include a page format preview and in conjunction with this a selective print ability whereby you can print, say, the middle page and the last page of a document.

There are also modifications to the cursor movement commands and a default extension of /TXT appended to an unextended file spec whether for the L or S commands.

After using this program for six weeks I have no hesitation in recommending it to other members of the group who currently use Scripsit to answer their word processing needs and to suggest to those who don't that they should reconsider Scripsit with these enhancements. For £20.00 this program does every thing that is promised and is very good value in these days of software being priced high for a fast profit. K. Robinson

from 33

Pencil 2 seemed to undergoing changes with Peter Knaggs doing the hacking, and John Kilpatrick supplying the wood. I managed to get Laurie Shields to do the Profile and Visicalc sessions which seemed to generate some considerable feedback. In the evening Geoff Smith took members through the techniques of patching Scripsit which again always provokes comment. The Dotwriter involment apparently started after this session.
Don Bannister very kindly did a session with Leo Knaggs on developing a modem into a finished product, with Leo going over the setting up and running of a bulletin board. They were alloted one hour and over-ran into lunchtime by 30 minutes
Carl Pabe also was very kind in doing a short session on the 64K print buffer board, explaining the design and construction. One of our past members markets these boards for around 25.00 Pounds and another can build them or just act as tutor over the Phone.
Sunday consisted of talks on DOTWRITER with Tony Evetts, CP/M with Geoff Smith, Laurie Shields and Robert Vieira.
Blandford Forum August 1984
This was another great day out with talks on Newscript by Rod Bullard, Telex on the TPS by Chris Cain, Data handling Derrick Rowe, Hard Disks on the TPS by Chris Cain and Disk Directories on CP/M by Leo Knaggs. Thanks again to Os and his team for a splendid relaxing day out Brian Cain

NATGUG NEWS

Volume 6 Issue 3 September 1984

OFFICIAL JOURNAL OF THE

National TRS-80

& Genie Users

Group.

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.

The Group maintains two software libraries (Models I and II) which are free to members. Library lists are obtainable from the Secretary.

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

CONTENTS

2. Information on the Group

4. Editorial

4. Guidelines for contributors

4. Library information

5. Members' letters

9. Problem section

10. Micro novice notes 3

13. 48K in the Model I keyboard

15. Sage accounts

18. Printing Genie Hi-Res graphics to scale

22. 'How do you work this computer?'

24. Cumulative index - Part 1

EDITORIAL

Sorry about the non-appearance of the cumulative index prepared by Richard Marks. Rather than publish it separately, I am including half of of it in this issue, and the other half in the next issue.

I have had several lengthy contributions which will be held over until next month.

One of the year's great events takes place next month. I refer of course to the Swindon workshop. The dates are 9/10/11 November, and the venue, as always, the Wiltshire Hotel. IQLUG will also be there.

Tandy are thinking about selling the ACT range in the States. They appear to be abandoning the manufacture of their own machines, since most of the range is now manufactured by other companies and 'Tandyfied' by the addition of a badge and a special colour scheme. Even the 2000 is made by a Japanese company. It looks like they are degenerating into a marketing organisation for other people's products.

See you at Swindon!

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 hand-written material, write legibly.
6. Do not fold the sheets when posting them. Use a large envelope and keep them flat.

Leon Heller

LIBRARY INFORMATION

Model 1: Leighton Davies, 105 Caerau Road, Caerau, Glamorgan. Tel: (0656) 738337

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

MEMBERS' LETTERS

Never having written to a magazine before I hope this assembly of jottings maybe almost as interesting as some of the articles produced each month in the "mag".

It is with great sadness that one reads of the impending doom that surrounds our user group and I'm astounded to find there are only some 500 members in the country. How many TRS80s are there in use I wonder. Nearly everybody knows someone who has one or who has a friend who has one. It perhaps isn't too obvious to a new buyer, or a second hand buyer that the group exists at all. Certainly Tandy's themselves aren't too open about it. Publicity is expensive stuff but I wonder if we shouldn't be doing something to stimulate new membership.

A little background info.

About July 1983 having been running a mail order business with the able assistance of my 13 year old son Adrian and business being good it became apparent that we had to "computerize". The admin effort had become enourmous and we couldn't justify staff. Although I'd been in the electronics industry for years and our mail order biz was that of selling hardware add-ons for home computers the prospect of actually using one of the things hadn't been seriously considered.

Where to start ? The local bookshops had about six million mags, each with roughly that number of ads claiming "we're the best-cheapest-biggest-smallest-most software of them all." No hope there, where to look next ? Simple I thought- ask a friend. Computer owning friends turned out to be whole new experience. Three to five hours ear bending seemed to be the least one could receive in response to a simple request for advice. Finally I decided I didn't have even enough knowledge to know what questions to ask, it seemed that the first machine would inevitably be the wrong one so lets make it fit the following bill, cheap (naturally), well established (not too many bugs left), local supplier (for the questions when it wouldn't work), an all-in-one unit (cant monopolize the telly every night), and twin floppies built in (seemed a good idea). Tandy Computer Centre in Reading was about the fifth shop visited and the first with anyone who had any idea how to assess my problem and offer a solution. I went home after about an hour with a melted Barclaycard in hip pocket and a TRS80 III a DMP100 and Profile III Plus in the boot. About a million manhours and four gross of aspirin later we had issue one of the record keeping for mail order ready to run. Its been a fantastic success and we can't believe how we ever ran the biz without it now. Having used the machine for just over a year now and become familiar with it, it seems we made a really good choice first time. The system has expanded continuously and even been added to with the purchase of another one from a gentleman in Coventry. We're real TRS80 addicts now. Oddly though both machines have had one keyswitch fail, and aren't they a pig to replace ?

Some time later we became aware of the presence of other operating systems and the magic words "LDOS", "NEWDOS", and many others came into the vocabulary, after much head scratching we converted to LDOS 514 but now I'm wondering if it was worth the effort and cost. Is there anyone in the group running Profile on floppies with LDOS ? I'd love to swap notes, we've got a bug or two.

Barry Baker.

318 Kempshott Lane,
Basingstoke.

Now few titles for your Cobol aspirant:

Robert T Grauer
'TRS 80 COBOL'
Prentice-Hall Series
(Most suitable - should be able to get it from their UK branch)

J M Triance
'COBOL PROGRAMMING'
National Computing Centre Ltd 1981 Manchester

William J Harrison
'A PROGRAMMER'S GUIDE TO COBOL'
Van Nostrad Reinhold Co New York (London)

Robert T Grauer
'A COBOL BOOK'
Prentice-Hall Software Series

Khailany and Duplissey
'COBOL FOR SMALL AND MEDIUM-SIZED COMPUTERS'
Houghton Mifflin Co Boston (London)

I have them all. Most of these titles are available from the local libraries - I recommend the first one in addition to the manual.

Anon.

For some months now my eyes were no longer up to the condensed print of my Line VIII printer and, so, I decided to go for a full (132 column) DMP. This decision was not without regret as the Line VIII printer was a really nice tool with many features not available in much more expensive printers. In fact this printer manages all SuperScripsit features with the single exception of double underline.

My choice fell on Epson FX100 which not only gives me 132 columns but also a higher printing speed of 160 cps against 100 cps of Line printer VIII. Though lacking in some of the features (like roll up) it is very fast and quiet. My end of the month accounts' run took much less time with the Epson and, in fact, it was the printer waiting for the disk drives and not the other way round.

Looking at the UK computer magazines you would think that you only need to decide that you wish for something and there are literally hundreds of firms only too eager to provide the goods. Well, nothing of the sort ! It took me two months to get the printer and it had to be flown from the dear old US of A

at the extra shipping charge of £40, which my kind supplier decided to halve with me. And I am not being sarcastic. I wanted the printer quickly and the supplier could have charged the whole shipping cost to me and I would not have complained.

But this goes to show what the adverts such as "available now" or "from our stocks" really mean. They just do not hold the stocks. I have not tested the wonderfully low, advertised prices of £415 for FX100 as a friend in a DEC OEM business, also involved in the micro side, told me that this advertised price was less than they were paying for it themselves. At the danger of becoming even more unpopular in some circles I would say that similar methods are employed by a certain cheap micro producer. I suspect that in spite of any promises of delivery within 28 days he waits until he gets enough orders to justify a big run and only then gives the word to his contractor to start the assembly rolling.

So this is my third printer now. With my second (LPVIII) I have discovered the pleasures and finally the pitfalls of HP. I had then LPVII (slow, cheap and nasty) when I saw LPVIII being offered by Tandy at £299.95 instead of £400+. The reduction was due to the fact that Tandy was discontinuing this model - only the salesman failed to inform me of that. There I was tut tutting over it and feeling that I really had no right to make yet another hole in our joint resources when the friendly salesman proposed that I should make use of their credit scheme. This was my first ever and the last (I hope) venture into the land of never..never. Since opening the account in January 1983 until it's closure at the beginning of April 1984 I bought the goods to the value of £572.80 - the printer being followed by Cobol and then SuperScripsit. I paid out £716.04 of which £87.85 was swallowed by the interest charges and £25 went for insurance. The worst thing is that I am now £20.39 in credit and I have been trying to re-claim that credit without any success since middle of April (It's June now). I have written to the Trading Standards Office in Leeds and now am contemplating taking out the summons in the County Court. The foregoing by way of warning to anyone tempted by the ease of buying.
(I am taking my time over writing this. Since my above bitter remarks, I have received the refund after yet another letter to the Director of Trading Standards.)

Having decided to stop buying Personal Computer World and Practical Computing to save the space in the house I have suffered lately a real deluge of literature from makers of DEC compatibles and other firms (including IBM). Now that the DECUS exhibition and DEXPO are over there should be peace for a while. It is astonishing though that, if one does require information, it is almost impossible to get it. I am getting a bit fed up with my accounting program and so have I written to some ten firms enquiring about their general ledger program. Whilst writing I have enclosed all information about the machine, operating system etc. After two weeks I had only three replies and, in two cases the programs would not do what I have specified and, in the third case the dealer need not have bothered as the storage required is 300K per drive.

As far as the general ledger is concerned I had another answer, this time from Peachtree, to the effect that they do not supply their programs for Model 4. Shortly after that a friend drew my attention to an advert from Guildford Computers that they supply Peachtree software for my machine. I rang them and they say "yes two general ledgers for Model 4 - one at £350 and another at £600". The boss says "if you must have yet another ac-

counting program buy the better one" so I am almost committed to the fuller version subject to check out. This is being done for me by a friend, system analyst working locally on mainframes. Incidentally Guildford Computers are apparently the only authorised Peachtree dealers for Tandy computers.

As far as CP/M is concerned - I like it. I like it very much. As stated in the last Journal CP/M+ has some bugs in it but as an operating system is good, reliable and "user friendly". Having used some five or six operating systems on Tandy I was always confused about the commands. I bought Newdos 80 v2 over two years ago and I must admit that there are many utilities in it but one (at least I) has to search through the manual each time one wants to use any of them. Some utilities and facilities I could not master to this day - like format (I copy with format and then purge) or the different types of files. CP/M is simple. There are not that many commands and I managed to learn all I require in a few hours. I like also the opportunity of being able to buy really good programs. Pro Pascal is a very good and full implementation (with bells and whistles) and I must admit that I prefer WordStar to Superscripsit - maybe I just like using the control key !!!

I had quite fun implementing WordStar on Model 4 and with FX100 I have several types of fonts selectable with Wordstar. There are in total six areas which permit specifying printer codes - not counting PSINIT: which caters for the printer initialization and PSFINI: which does the same for the printer on ending of using WS. Those are PALT: to set alternate character width, PSTD: to reset to standard character width (in each case five bytes) the first one being the number of bytes to send to the printer and then up to four bytes. Thus assuming that we want to choose Epson Elite to be the alternate set we would say PALT: 02, PALT:+1 1BH, PALT:+2 4DH. The other four niches are USER1: USER2:, USER3: and USER4:. In each case up to four bytes can be sent to the printer and the first byte defines how many bytes are being sent. The WordStar manual is very good as far as installation is concerned.

I like WordStar too. The advantages over SuperScripsit are numerous the main one being that, if one edits a file, a backup copy is immediately created on opening of the document. So, if one happens to do anything stupid, as I do occasionally, at most one does lose only the results of the current session. Documents can be read into the file one works on, blocks can be saved as files with different names from the document worked upon and the fonts can be varied in the same document. Also WordStar can print a document whilst one works on another. Those are the main advantages which spring immediately to my mind. The arsenal of WordStar commands is large and I wonder whether anyone knows and uses all of them. But one does not need to use them all to nip along creating the text. A dozen, or so, will do to feel comfortable with this program.

I was looking forward to the workshop and the sessions on CP/M but after all it was not to be. A pity but these things do happen.I wonder what will be the rate of migration from the Tandy specific DOS to CP/M in the brotherhood of Model 4 owners ? The only trouble is that CP/M programs are more expensive than those written specifically for TRS but this is the usual in the computer business. In the mini world one cannot buy a program under £1000 - at least not for DEC. When I enquired for a Data Base for my PDP 11/23 (not really mine - the one I manage in the office) the answer was "we have one. It is German and not expensive - only £11,000". To me this was outrageous so instead we

decided to buy a micro to do the job. One can estimate about £100 to £200 more for a CP/M program doing similar job to a Tandy specific program. But the choice is greater and the programs are probably "more powerful". I have not found any problems with the Copydisk program in CP/M+ but then I normally copy whole disk (selection 3 from Copydisk program and answer the prompt with 0). This copies all tracks, track for track, and all system and data files. The point is that the CP/M+ system resides on track 0 and CPM3.COM and CCP.COM are treated as any other data files. Answering selection 3 with 0 produces immediately bootable disk. In other CP/Ms the first two tracks are system tracks and they contain all system files (including CCP.COM).

Anon.

PROBLEM SECTION

I recently bought two second-hand 40-track disk drives for my Model 1, to replace 35-track drives which decided to self-destruct. The drives were accompanied by some software, which included an MMS Forth package (V 1.9). Unfortunately the previous owner seems to have corrupted the second disk (the utilities package) and I can't persuade it to load, although the system disk can read it if I put it in drive 1. I made a backup using Superbak, but the installation procedure described on the utilities disk just doesn't seem to work. My attempts to purge a duplicate system disk and transfer the utilities to it have also failed, mainly because I don't yet know enough Forth to do the job properly. Is there anyone out there who has the second disk up and running (preferably for 40 track, though this isn't essential) and is prepared to copy their disk onto my corrupt master disk?

Another problem arising from this upgrade is the fact that DIR on NEWDOS 2.0 seems to read all disks as 40-track when PDRIVE is adjusted to regard the drives as 40-track (as in ZAP 029, February 1981). I'm gradually changing all my data disks to 40-track, but quite a lot are still 35-track. Is this normal, and will it cause any problems (if, for example, I try to write to a 35-track data disk using the 40-track DOS)?

Finally, a lot of my older programs are on NEWDOS 2.1, and a few won't work properly with NEWDOS 2.0 . I'd like to continue using them, but want to upgrade to 40 tracks on all DOS's. I've never used PDRIVE or SYSTEM commands with NEWDOS 2.1, and can't find the original instructions to check the appropriate syntax. I only know that it isn't the same as NEWDOS 2.0 . Can anyone tell me how they work on NEWDOS 2.1?

With many thanks, Marcus L. Rowland,
22 Westbourne Park Villas, London W2 5EA

MICRO NOVICE NOTES 3

Relax. This is the last item in this series.

No term as a novice goes on for ever. After a year or two they always either throw you out or move you on. In my case I have to end it myself. Though by most tests I'll be a micro novice for ever, I have decided that this is the last year I'll call myself one.

I have some criteria. Programming skill isn't one of them. I remain one of those for whom computers are strictly a tool. I am dead set against having to learn a thing about them that I don't need for my purposes (my purposes being word processing, database management, and a bit of number-crunching and graph-making). My cry is for human-friendly micros, not micro-friendly humans.

My criteria may be reduced to one: you stop being a novice when, having started with the bare essentials of a micro-system, you move beyond those bare essentials.

For 30 months I stuck it out with Model 1, with 16K of memory, with the tape Scripsit word processing program, with cassette storage, with an Epson MX80/2 printer. I'll never knock it. No one can say I didn't wait long enough to see how much you can do with even the humblest micro means.

Oh, I had my Scripsit Epson patch, thanks to JOHN PHELAN and Molimerx. And I had my B17 program to integrate Scripsit and patch and triple its loading speed, thus turning my biggest headache into pure and pleasant routine, thanks to ERNEST BERBINGTON. And in the last months I had the priceless new knowledge, thanks again to EB and JP in the Nov. '83 Newsletter, that you can exit from tape Scripsit with BREAK and "END", do e.g. a print mode change not possible within Scripsit, and then recover both Scripsit and text with SYSTEM and "/24317". *One more thing that Tandy men had told me couldn't be done!*

All this time, however, I never forgot my shopping list, my unfinished business -- memory expansion, faster storage, better program access to printer capabilities, and some improvement on the printer capabilities themselves. And by last March I had finally screwed myself up to the fateful step, unavoidable in any significant upgrading, of actually interfering with the innards of my system.

AN INTERIM UPGRADE

First came the upgrading of the printer with Graftrax chips. How simple it proved after all the phobia! Thanks to the chips, customized for me by DON BANNISTER, I suddenly found my print modes doubled to 24 with italics, my character set enhanced with some of the continental characters I needed most, and Tandy graphics for the first time at my finger-tips.

Then came the upgrading of my 16K memory to 48K with a Holmes board. To work with Scripsit and 16K is to work with only 4K free for text. That meant page-by-page text editing, which was a lot better than anything less, but left any serious database work totally out of the question. To appreciate having all of 36K free for text or data, there's no better preparation than working with 4K for 30 months.

Simultaneously came the upgrading of my storage with an Aculab "stringy floppy" drive. (Actually what happened was that I bought, in one old and cheap package along with the Aculab, an entire keyboard complete with Holmes board, since I was planning to have 2 systems anyway.)

I never saw Aculab as more than an interim system. You have to trade off relative speed (a quarter of a minute to load Scripsit) and some disk-like facilities (file handling etc.) against tiny capacity (3K to 63K, and taking 2 minutes a turn at the latter size) and high wafer price (depending on length, you could end up getting no more than 10K and as little as 1K for £1). But I find it good enough to make me want to lose no more time in leaving my cassette days behind.

As part of the Aculab package I quite unexpectedly acquired something I hadn't even heard of, though it has almost more interim value for me than Aculab itself (for even without Aculab I could run it, B17-speeded, off a cassette). This was another modification of Scripsit, done by LAURIE SHIELDS, that gave me just the extra facility I needed to access the new capabilities of my printer (or, for that matter, of any printer, since its print mode commands use ASCII code).

How I began my move to disk while all this was going on is another story, not yet finished. Meanwhile, like someone who has climbed a cliff on the bottom 100 metres of Mount Everest, I can look back from the new plateau and say to all who haven't come this way, "It wasn't such a bad climb after all, and it really is a lot more comfortable once you get over the ledge".

User memory capacity boosted by a factor of 9; program loading speeded by a factor of 12; any one of over 160 characters, including over 60 graphics characters, on call from a single key; an old dream fulfilled of having Scripsit access to Graftrax italics, to non-standard characters, and to elementary graphics! I could live with these simple gains for a long time yet, if I were planning for myself alone.

BEYOND THE INTERIM

I have never supposed that any upgrading based on a Tandy Model 1 could be anything but an interim improvement. Nor would I be in the process of acquiring a Model 1 disk system with multiple double density drives if I weren't getting that also so cheap that I'll be ready to supersede even it when the time is ripe.

And when will the time be ripe? I'd have to move on (1) if my Model 1 equipment just packed up and wouldn't work any more (but it shows no sign of doing that); (2) if I badly enough needed some capability to which there is no practical access via Model 1 (some signs of that all right); or (3) if something became available which would be worth getting even at the cost of writing off some or all of the surviving value of present equipment.

When the Sinclair Quantum Leap was announced, like a lot of others I thought, "This must be it!" I gave it a year to get its act together. Now I'm not sure it ever will. I had already concluded of the BBC-B and the IBM PC, "This is not it". Condition 3 will not be fulfilled for me till I am very sure the new system is worth that much more than the old, not only in general, but in regard to certain specific features I have come to see as essential.

But that too is another story. Here I am just discharging my debt of gratitude to those who helped to make tape life livable for me while it lasted. In addition to credit already given in this and previous articles, I must record my appreciation also of trouble taken for me in the past year by LEON HELLER and PETER KNAGGS.

Parig Digan, Dalgan Park, Navan, Ireland

48K in the Model 1 keyboard.

Now that General Northern have gone out of business we thought it would be a good idea to see if it was possible to modify the Model 1 Keyboard to read 48k without having to add a Decoder board.

With considerable help from my good friend Guy Grantham who has already modified his Video Genie we have succeeded & here is how we did it.

The memory signals combined with RAS* & CAS* to read;-

Up to 16K Ram Require	A15 LOW
	& A14 HIGH
From 16K to 32K " "	A15 HIGH
	& A14 LOW
From 32K to 48k " "	A15 HIGH
	& A14 HIGH

Therefore to read all 48k either A14 or A15 or both must be HIGH at the relevant time.

Originally the Computer could only recognise up to 16k of ram because it was only decoded to read memory while A15 was LOW due to Z73 (pins 4,5,6 OR gate) controlling Z21 (2 line to 4 line decoder) at pins 14 & 2.

These signals will be retained for Video, Keyboard & Rom but no longer used to generate the MEM* & RAM* signals by opening links 2,3,4,5 on X3.

New connections are required to generate the MEM* signal which comes from pin 6 Z74.

This signal now has to be active LOW whenever there is a RD* active, excluding calls to reserved memory 3000H to 37FFH required for Disc & Printer I/O or to the Keyboard at 3800H to 3BFFH or again to the separate Video Ram at 3C00H to 3FFFH.

All these invalidating conditions are met when A15 & A14 are LOW & A13 & A12 are HIGH, as provided by the output from Z21 pin 12 to input pin 4 of Z36. So by cutting the trace to pin 5 of Z74 & linking pin 4 of Z36 to this pin the required conditions of MEM* are met for 48K Ram.

Z73 pins 8,9,10 are now 'spare' by virtue of cutting the trace to Z74 pin 5. This gate together with 2 other originally 'spare' gates, Z73 pins 11,12,13 & Z37 pins 8,9,10 will be used to decode the Row & Column addresses of the 64k ram chips.

Pins 9 & 10 of Z73 must have their incoming traces cut & replaced by links to A14 & A15 at Z21 pin 15 & Z73 pin 4 respectively.

Z73 pin 8 outputs the (A15 OR A14) condition required in the first paragraph to address the top 48K of the memory map.

This signal is inverted by the spare NOR gate of Z37 (8,9,10) & combined with RAS* in the spare OR gate of Z73 (12,13) which, in turn from pin 11 feeds the existing RAM* connection at Z74 pin 10.

NOTE, pins 8 & 9 of Z37 must be separated & only ONE of them may be connected to pin 8 of Z73, as although pin 9 appears to be 'floating' it is not possible to to be 100% certain about pin 8 so it must be tied to pin 7 (earth).

A14 & A15 must also be linked to pins 14 & 13 of the Data Selector Z51 whilst its output at pin 12 will be linked to the No 9 pin of each of the 64K rams.

Here are the detailed changes necessary;-

Open up the Keyboard & locate the following chips on the 'trace' side of the board. Z21, Z37, Z51, Z73, Z74, I stuck small number labels on to make sure I could always read them, locate pin No 1 for each chip.

Lay the board component side down with the keys away from you.

Cut the following traces, between pins 8 & 9 of Z37 (a solder sucker is needed here), close to pins 9 & 10 of Z73, (pin 10 on the component side) & at pin 5 of Z74.

Using thin insulated wire connect these pins in this order;-

From Z73 pin 11 to Z74 pin 10.
Z73 pin 12 to Z73 pin 5.
Z73 pin 10 to Z73 pin 4 & on to Z51 pin 13.
Z73 pin 13 to Z37 pin 10.
Z73 pin 9 to Z51 pin 14 & on to Z21 pin 15.
Z73 pin 8 to Z37 pin 9.
Finally Z37 pin 8 to Z37 pin 7.

Now X3 has to be modified as mentioned earlier, starting from No 1, leave that as it is, Nos 2, 3, 4, & 5 have to be open & 6, 7, & 8 closed. A small screwdriver can be used to open & a blob of solder to close any as needed.
Before the 16K Ram chips are removed the system can be tested by switching the computer on in the Level 2 mode & entering 'PRINT MEM', the answer should be either 48340 or 48338 depending on which type of Roms were fitted, the figures are only 2 reflections of the existing 16K & as such cannot be used.

Providing the system works this far then the fitting of the 64k Rams can be carried out.

Remove the 16K Rams & put them away safely, with the board still laid out with the keys away from you cut the 12v trace at pin 8 of Z19, cut the -5v trace at the capacitor pin below pin 1 of Z13, fit a thick wire from the +5v pin above pin 7 of Z1 to the pin above pin 8 of Z17.

Fold the keys over on top of the board & turn the whole assy over, very carefuly bend up pin 9 of each of the 64K chips (not too far or they may break) & insert them in the empty Ram sockets the same way round as the 16K chips, making sure that each pin 9 is clear of the top of the socket as that might be at +5v, (contacts may be slightly proud on some types of socket), wire all the No 9 pins together & connect a wire from pin 12 of Z51 to the nearest pin 9 of the Rams.

Carefuly RECHECK that the +12v trace is definitely CUT & the system is ready to go.

It is not essential to cut the -5v rail but it should be done as a safety measure, cutting the original +5v rail is a very complicated job & is best left alone hence the comments about pin 9 being well clear of the socket as that is still at +5v.

NOTE. Do NOT use Texas 64K Rams as they have a different refresh system & cannot be used without further modification to the Computer circuitry.

E.C.Kilpatrick,
3a Gainsborough St,
Sudbury.
SUFFOLK. CO10 6ET.
19/09/84.
Phone 0787 79504.

CHECK JAN 85 FOR AMENDMENT

SAGE ACCOUNTS

The promotional material, after slightly dying down, has re-started again. I seem to be getting at least two or three letters every day and most of them go straight away into the wastepaper basket. Today however brought, in addition to glossy letters and brochures (which went the usual way), a mug from Autofile and Microcomputer News. I seem to be getting this latter for nothing although it states on it that it is a monthly and it costs £3.

Actually though I would never pay £3 for Microcomputer News I do like it and it does contain a great deal of general information of interest to mini and micro users. What caught my eye in this latest (August) issue was a heading "Software for QL" and I see that CP/M 68K will be implemented on this wonder and the cost of it will be about £50 (cheap!) and also that SageSoft are modifying their accounts to run on QL and that these accounts programs will be "compatible with some of the bundled software".

Well I have now the Sage accounts - the cheapest version at £375 + VAT. The CP/M+ program disk just contains all the programs plus Pip leaving some 4K free - not even enough space for Show.com. Still no matter as the housekeeping can be done with the master disk. The data disk, which must reside on drive B: has 144K available for the transaction file after creation of 10 purchase, 10 sales and 90 nominal accounts. This at 90 bytes per transaction gives enough room for some 1,600+ entries between the updates. As I use this program just to keep check on my home finance I won't be using sale accounts and 90 nominals is just enough. If anyone wants to use it for business with several suppliers and sale accounts this storage would not be enough.

The first step, on receiving the program on IBM format disk (CP/M 86), was to install it. There are four areas which need installation and these are the screen, keyboard, printer and program control.

Screen installation requires clear screen(27 69 0 0 0), bleep(7 0 0 0 0), single cursor, double cursor, cursor lead in(27 89 0 0 0), cursor position separator, cursor position terminator, x & y increment(31 31 0 0 0) and x-y order/IBM/ANSI. All values must be in decimals and in each case five values must be stated, zeros being entered where no value exists. I have not bothered with the single and double cursors, separator, terminator and x-y order.

Keyboard installation requires escape(27 0 0 0 0), return(13 0 0 0 0), delete(8 0 0 0 0), backspace(8 0 0 0 0), up line key(11 0 0 0 0), down line key(10 0 0 0 0), forward key(1 0 0 0 0) and backward key(2 0 0 0 0). Upline, down line keys were sheer guess and forward and backward key codes were entered for the sheer hell of it. No effect on the program and the screen and keyboard seem to behave - break being escape key.

Printer installation requires form feed, condensed print (twice), normal print (twice) and lines per page. On my printer form feed is decimal 12 and, as I don't want condensed print, I filled the condensed and normal print fields (all four of them) with zeros and I decided to have 50 lines per page. As the default CP/M+ setup disables the linefeeds after carriage return I have not experienced any problems with printing.

Finally came the program control and this I set up as: disk allocation 180 0 0 0, sales, purchases, nominals 10/10/90, VAT codes 0-9 I left strictly alone, drive designation is 1 and VAT calculators 62 for net and 60 for gross. VAT accounting I don't propose to use. The calculators < and > do some very clever things on posting by working out the net or gross VAT but so far I have not bothered to read about it. In fact I have not read the manual at all yet. The little knowledge which I acquired whilst playing with Try Before You Buy version is in fact all I need to operate the program.

I had the answer to my queries about CP/M 2.2 though the reply came from Total Access (PO Box 790276, Dallas, TX 75379) and not from Montezuma to whom I wrote. Most succinct reply too on my letter which has been annotated with brief 'yeses' and 'noes' with a few additional explanations . Anyway for those interested - if one wants to add 80 tracks, double sided drives, one is able to format such disks, one can define physical drives 3 & 4 as A: B: C: or D:, a disk is treated as one unit and not as separate sides and the manual contains all the codes necessary to customise CP/M programs and, furthermore, these are different codes from those used by CP/M+ in spite of what the only UK distributor of this program told me. The further advice was that, if I wish to use the 80 track drive as drive A: I must have a normal 180K disk which is used to boot and that the configuration on this disk must then transfer the control to the 80 track drive A:(or words to that effect). On the other hand if my drive 0 should be an 80 track one then I must specify this when ordering. In fact I have received brief and exact information and all my questions were answered precisely. Whilst a slight retrenchment is now the order of the day after the latest purchases (FX100 and Sage) I think that I will get this CP/M but from the dear old USA. They deserve the sale after being so helpful. Obviously our UK distributor must be in clover and does not care whether he sells or not - or is it that he feels that we have no option but to buy from him ?

In the office, with yet another upheaval and new system of subscriptions applicable to some of our subscribers, I find that my subscribers file capable of holding 16,000 records will not be big enough. I was facing rather a tedious job of having to reduce the volumes of the accounts files (suppliers, invoices and transactions) in order to achieve a saving of some 4,000 blocks required for the expansion of the subscribers' files. This is a fiddly job as one has to create new file parameters, play with renaming of the files and then moving the contents into newly laid out files and finally having got the space on the data drive

one has to strip the program drive to make the space for the existing subscribers file and finally move the contents from there to the data drive. Of course the job is done on the spare packs and, as usual, I have about ten of those. Whilst I only require 5 packs for the system (system master & it's copy, file master and copies A and B) I have other copies all over the place. Anyway it now looks that I have managed to convince the directors that we have to expand and that I will get extra £8K on top of my annual allocation of £6.5 to expand the memory to 256K and also buy a third drive - this time 42MB Winchester. This should give plenty of room for the expansion during the next two years and after that I will be put out to grass and will try to earn some pennies on the side with the Sage and woodturning. Anyway no hustle with cutting here and adding there which would take me at least a day if everything would go right. Of course I could have the job done by our OEM but that would cost between £200 - 400 and I don't propose to pay for things which I can do myself.

But back to Sage !

SageSoft have produced three versions of their accounts' program - the basic version being Sage Accounts. This is a fully integrated, one disk system covering sale, purchases, nominal ledger and providing a variety of reports apart from the usual audit trail, trial balance, ledger and accounts prints. When I read up something about this program I will cover the reports in some detail.

The next version is Sage Plus which is based on Sage accounts but permits 9,999 sale/purchase accounts and 9,999 nominal accounts. I should think that such volumes are purely academic and I cannot imagine anyone having requirement for such a vast number of accounts. Where these ceilings may be useful is that one can allocate blocks of numbers say suppliers 500-600 Notts or nominals 2,000 - 3,000 salesmen's travel expenses and so on. But in addition to the increased volumes Sage Plus has budgeting facility and reports budget variances and also has a credit control facility. This version costs £525 + VAT.

The top version is Sage Executive which is Sage Plus plus invoice generation, linkage to stock files, integration to sales ledger, user defined invoice formats, 9,999 stock items (no superfluity here), stock re-order control, stock reports and valuation. This program, still on one disk costs £695 + VAT.

What I find very attractive about the Sage philosophy is that having bought one of the programs one can then upgrade it as required. One can upgrade from Sage Accounts to Sage Plus or Sage Executive and of course from Sage Plus to Sage Executive. There is none of that "Oh yes - if you want your computer to do that you will need another program. We have one ...." and the customer finishes paying for a completely new program and not just an upgrade. I also like the screen presentation and the options of displaying or printing or error correction. This

latter is especially good on sales and purchases functions.

Before I shall continue with Sage I would like to air a few thoughts on copyright etc. Having signed the licence agreement I am prohibited to copy the documentation or any parts of it. Quite understandable as the software piracy is widespread (and it will continue to exist whilst the programs remain so highly priced) and the program without a manual is generally not much use. I say generally because after the bitter experience with the otherwise excellent program which I use in my office I am sure that I would have had less problems if no documentation had been provided at all. But before I signed the agreement I was sold the Try Before You Buy version. The brief manual issued with that version contained sufficient information to permit one to operate the full version as some of the aspects, such as report printing or update, could be deduced from the menus and screen displays when one ventured to enter a function not covered by the manual. Now the question which does bother me is - Is one in breach of the licence agreement if one describes the program operation in such a manner that anyone possessing average intelligence could on reading such desciption operate a ripped off copy of the program ? I shall have a ponder on that and, in the meantime, I would refer anyone wanting to know a bit more about the Sage Accounts to PCW October 1983. I have to add however that some of the critical remarks contained there are no longer valid. Obviously the program has been enhanced since.

Anon.

PRINTING GENIE HI-RES GRAPHICS TO SCALE

In Vol 4 No 5, November '82, the Journal contained a program by Keith Bedford of Lowe Electronics for printing Genie Hi-Res graphics on a Lowe EG 3085 printer. Some time later, Dick Gays sent me a copy modified for use with an Epson printer. With a few changes I was able to make it work on my Shinwa machine. Unfortunately, the size and particularly the shape of the printout do not match those of the screen. Screen circles print as ellipses, and an isometric drawing with lines at 30 degrees is unacceptable when the lines print out at nearer 45 degrees!

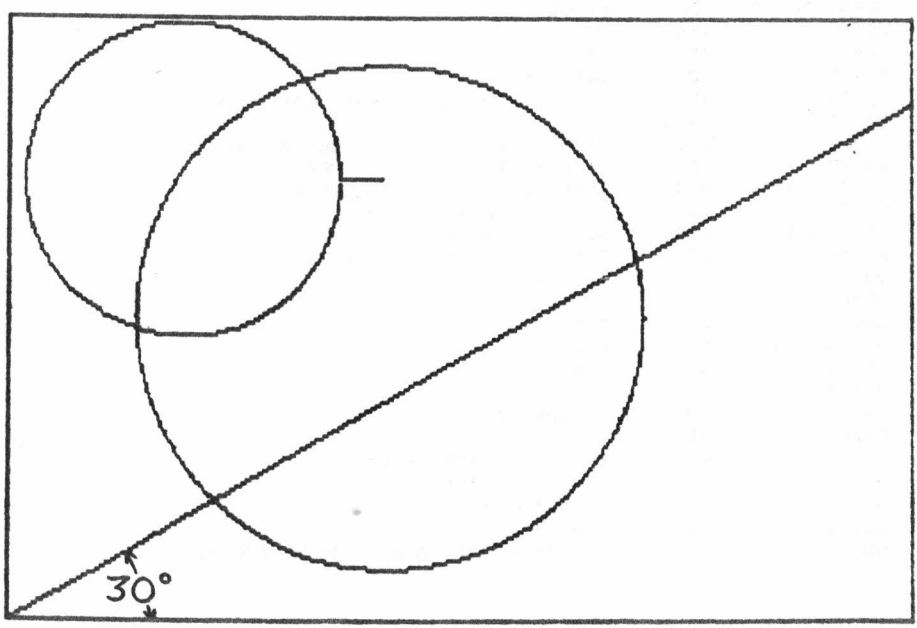
The Genie Hi-Res screen is made up of 192 x 384 pixels. The Shinwa prints either 640 or 1280 dots in a 7.5" line, and prints 8 lines at each carriage traverse. Sending 384 pixels results in a line 2.25" long in the 1280 mode, and 4.5" long in the 640 mode. Printing 192 pixels in a vertical line results in a line about 2.25" long. This length can be increased by multiples of two by sending each pixel to the printer twice, four times, etc.

Doubling the width of the picture by sending each horizontal pixel twice, effectively 768 pixels, would be too many for the 600 dot mode, and would produce a line only 4.5" long in the 1280 dot mode, gaining nothing. In any case, a 9" line would be too long for my paper width. Although a 2.25" square illustration has its uses, 4.5" square is better for general use, achieved by sending horizontal pixels once in the 640 dot mode, and vertical pixels twice.

The maximum rectangle I can draw on my screen is 7.2" x 4.9" approx. In order to get anywhere near these proportions on the printer I need to increase the printed width by 1.5, ie send 576 pixels. One way is to print every alternate pixel twice and hope that the resulting distortion is not noticeable! As you can see from the enclosed example, it works very well. It might have looked odd if the illustration contained vertical lines close together, since some would be twice as thick as others. One way to avoid this is to ensure that verticals appear on either all odd, or all even, pixels. Or draw them all two pixels wide and they will print the same thickness wherever they appear. They need to be thicker anyway to match the horizontal lines.

In order to plot circles which appear circular on my screen I need to use a scale factor of about 1.4 ie plotting x = 1.4y results in a line at 45 degrees. Because my printout still does not quite match the screen, the factor needs to be 1.35 for the printer. For most purposes a value somewhere between is adequate, but where it does matter it is simple to define the factor as a variable at the start of the program. The value can then be changed and a replot done before printing. This method cannot be used however if the "picture" is not plotted by a co-ordinate method.

I enclose a copy of the source program, in case you think it may be of interest to others. The 1.5 times magnification occurs at lines 783 and 785.



```
00120 ;SOURCE CODE FOR VIDEO GENIE HI-RES
00130 ;SCREEN DUMP TO SHINWA PRINTER
00140 ;TREBLE HORIZONTAL AND DOUBLE VERTICAL SCALE
00150 ;FOR 64K MEMORY
00160 ;FOR 16K CHANGE LINE 220 TO ORG  7560H
00170 ;15TH SEPTEMBER 1984
00180
00210
00220         ORG     0EF60H
00230 CR      EQU     13        ;CARRIAGE RETURN
00240 ESC     EQU     27        ;PRINTER ESCAPE CODE
00250 PDATA   EQU     236       ;HIRES DATA PORT
00260 PPOS    EQU     237       ;HIRES POSITION PORT
00270 PLINE   EQU     238       ;HIRES LINE PORT
00280 POUT    EQU     253       ;PRINTER PORT
00290 INIT    EXX               ;SAVE ALL REGISTERS
00295         CALL    SETUP
00300         LD      A,1
00310         LD      (BITMSK),A ;SET BIT MASK TO 1ST POS
00320         LD      A,0FCH
00330         LD      (ROW),A   ;SET ROW TO 1 LINE BACK
00340 MPLOOP  LD      A,(ROW)
00350         ADD     A,4       ;SET ROW TO TOP OF SCREEN
00360         LD      (ROW),A
00370         CP      192
00380         JR      C,MORE
00390         CALL    DESET     ;RESET PRINTER
00400         EXX               ;GET ALL REGISTERS BACK
00410         RET
00420 MORE    CALL    SETHRS    ;SET PRINTER = HIRES MODE
00430         LD      A,255
00440         LD      (COL),A   ;SET TO -1
00450 CLOOP   LD      A,(COL)
00460         INC     A
00470         CP      64
00480         LD      (COL),A
00490         JR      NZ,DOROW;NOT LAST ROW, CARRY ON
00500         LD      A,CR
00510         CALL    PRINT     ;PRINT CARRIAGE RETURN
00520         JR      MPLOOP    ;DO NEXT 4 LINES
00530 DOROW   OUT     (PPOS),A;SET COL IN HIRES
00540         LD      B,6       ;6 BITS WIDE
00550 R6LOOP  PUSH    BC        ;SAVE IT
00560         LD      D,0       ;FORM HIRES BYTE
00570         LD      A,(ROW)
00580         LD      C,A
00590         LD      B,4
00600 BLOOP   LD      A,C
00610         ADD     A,B
00620         DEC     A
00630         OUT     (PLINE),A
00640         OR      A
00650         RR      D         ;SHIFT RIGHT
00660         RR      D         ;TIMES TWO
00670         LD      A,(BITMSK)
00680         LD      E,A
00690         IN      A,(PDATA);GET 6 BITS FROM SCREEN
```

```
00700           AND     E
00710           JR      Z,SKIP  ;CHANGE TO NZ TO REVERSE PRINT OUT
00720           LD      A,D
00730           ADD     A,128   ;SET TOP BIT
00740           ADD     A,64    ;ALSO SET TOP BIT
00750           LD      D,A
00760 SKIP      DJNZ    BLOOP   ;DO 8 BITS
00770           LD      A,D     ;HIRES BYTE
00780           CALL    PRINT   ;SEND IT
00781           POP     BC      ;RESTORE COUNT
00783           BIT     0,B     ;MAGNIFY HORIZONTAL
00785           CALL    Z,PRINT ;BY ONE & A HALF
00790           LD      A,(BITMSK)
00800           ADD     A,A     ;A = A * 2
00810           LD      (BITMSK),A
00830           DJNZ    R6LOOP
00840           LD      A,1
00850           LD      (BITMSK),A
00860           JP      CLOOP
00870 PRINT     PUSH    AF      ;SAVE DATA
00880 PSTAT     IN      A,(POUT);GET PRINTER STATUS
00890           XOR     3FH     ;INVERT MAIN BITS
00900           JR      NZ,PSTAT;0 = OK
00910           POP     AF
00920           OUT     (POUT),A;SEND IT
00930           RET
00935 SETUP     LD      HL,DSMES
00940 SELP      LD      A,(HL)
00950           OR      A
00960           RET     Z       ;END OF MESSAGE
00970           CALL    PRINT
00980           INC     HL
00990           JR      SELP
01000 SETHRS    LD      HL,SHMES
01010 SHLP      LD      A,(HL)
01020           OR      A
01030           RET     Z       ;END OF MESSAGE
01040           CALL    PRINT
01050           INC     HL
01060           JR      SHLP
01070 DESET     LD      HL,SUMES
01080 DSLP      LD      A,(HL)
01090           OR      A
01100           RET     Z
01110           CALL    PRINT
01120           INC     HL
01130           JR      DSLP
01140 SHMES     DEFM    '     ';A MARGIN
01150           DEFB    ESC     ;SET PRINTER TO
01160           DEFM    'K'     ;600 DOTS/LINE MODE
01170           DEFB    64      ;SENDING 576
01180           DEFB    2       ;BYTES OF HI-RES
01190           DEFB    0
01200 DSMES     DEFB    ESC
01210           DEFB    65      ;LINE SPACING
01220           DEFB    8       ;8 DOTS
01230           DEFB    CR
01240           DEFB    0
```

```
01241 SUMES    DEFB    ESC       ;RETURN PRINTER
01242          DEFB    2         ;TO TEXT MODE
01243          DEFB    CR
01244          DEFB    CR
01245          DEFB    CR
01246          DEFB    CR
01247          DEFB    0
01250 COL      DEFB    0         ;STORE COLUMN NUMBER
01260 ROW      DEFB    0         ;STORE ROW NUMBER
01270 BITMSK   DEFB    0         ;STORE BIT MASK
01280          END     INIT
```

B. Mason
'Highlands'
Bromsash
Ross-on-Wye
Herefordshire
HR9 7PR

"How Do You Work This Computer ?"

Had an interesting call about a month ago. It went like this :-

Ring,ring.....ring,ring

"Hello,1234" (this is not my real number !)

"Hello,this is Tandy Corporation here.We have a Model 100 in the Computer Centre and we don't know how to use it. As you belong to the User Group we thought you could help"

(I was a little taken aback by this, I have had calls from Tandy before of a more difficult nature like how to connect disc drives to a computer or what was Newdos and where do you get it from, so I could only repeat what they had said as a question)

"Yes that's right"said Tandy"can you help ?"

Regretfully I told them I could not as I had never seen a Model 100 (except in pictures).

"Oh dear, can you suggest anyone?"

"Seacoal Lane?"

"Where?"

Much as I would like to have continued this conversation the doorbell then rang and I had to say goodbye. I was kind enough to go past the Computer Centre a few days later but I saw it working in the window, so you need not all rush round to your local Tandy's.

Anon.

For Sale:-

Radio Shack TRS 80 Model III Micro-computer with integral twin disc drives.	£650
Radio Shack TRS 80 Lineprinter VI with Tractor unit for continuous print out.	£250
40 Minidiscs in filing containers.	£30
Software:-	
Purchase Ledger III	£30
Visicalc	£30
Stock Control	£30
Superscriptsit	£30
Profile III Plus	£30
10" high pack of folded continuous computer paper.	£5

The above equipment represents a complete business computer system and the seller is prepared to consider offers for the whole package.

Telephone:- Paulerspury (032 733) 602.

LUTON MAILBOX

Tha above has a 300 baud port on its system, the number being 0582~~34111~~ I am adding to the infomation quite regulary now with either the M100 or the M1 from home.
To use the above try the following:-

HELLO c/r
12,110 c/r
566660 c/r
MAILBOX c/r
SUBSCRIBER BRIAN PAIN c/r
COLLECT c/r
READ c/r
WRITE c/r
POST BRIAN PAIN c/r
FINISH c/r
BYEF c/r
The above c/r are carriage return/enter key.
What is happening above is :-
using HELLO to start with, then the account number and password. MAILBOX is opened and subscriber details entered, requesting that mail is COLLECTED for READING.
A message is left by first WRITING it followed by POSTING to the necessary SUBSCRIBER.
Once the session is FINISHED, the user then logs off with BYEF.
I hope the above enables some members to have a go at logging on to the college, and of course you can always try out the equipment in the TANDY store.
The system is used by secondary schools in Bedfordshire who access it with BBC's and 380Z etc.
Brian Pain

CUMULATIVE INDEX

This is the Index for the first four volumes of the Newsletter and parts of volume 5. In time, it will be expanded to incorporate future publications.

Wherever possible, articles have been cross-referenced. Thus an article on using SCRIPSIT with VISICALC can be found under either heading.

I found a large number of short tips on BASIC programming. For convenience, I have lumped these together under the heading BASIC TIPS. A few of these are only applicable to LEVEL 2 or DISK BASIC and these are listed separately.

Readers letters appear in each issue, and have been indexed by their subject. However, letters which ask unanswered questions have been left out, as I couldn't see the point in including them. Similarly, reports from Workshops have been excluded, although they could go into the next index if thought necessary.

The index was composed using AIDS III Database and SCRIPSIT-PLUS, and printed with an Olivetti Bytewriter. Both these pieces of software are highly recommended.

•

Richard Marks.
June 1984.

		Volume	Issue
ACCEL	Compiler review	3	9
ACCOUNTANCY	Sorting Files	4	5
	Using M/C Analysis	4	2
	Electric Accountant	3	5
ACCOUNTS	Ledger prog review	5	4
	Quickmet Review	3	12
ACCULAB	A Eulogy	2	11
	Checking Tape speed	2	11
	Directory	3	3
	and ELECTRIC PENCIL	3	2
	Damaged Directories	2	10
	Finding Directory	2	6
	Moving Directory	2	4
	KILLing Files	2	8
	and MICROTECH	2	10
	" "	2	12
	PURGE and RENAME files	2	6
	Print Buffer	5	9
	Printing Screen Contents	2	8
	Review of Machine	2	3
	Renaming M/C programs	2	8
	Saving M/C files	4	2
	Saving M/C routines	4	9
	Saving screen onto tape	2	11
	with LEVEL3 Basic	2	8
ACCULAB XBASIC	Extended Basic	2	12
ACOUSTIC	Coupler	2	1
ADDITION	in machine code	5	3
ADVANCED VIDEO	AV1 Lower Case review	4	4
	" " "	4	6
ADVENTURES	Transfer from Cass.to Disk	2	6
	SCOTT ADAMS Tips	2	1
AEROCOMP	Double Density review	4	4
AIDS	Database review	5	10
AJEDIT	Word Processor Review/Tutorial	4	1
APL	Language Review	3	8
ASCII	Codes: listing and explanation	2	12
ASSEMBLING	for beginners	4	5
ASSEMBLY	language explained	3	6
ASSEMBLERS	- See EDTASM and ZEN		
AUDIO COMPUTERS	Model I Exp. Interface	4	12
	Memory expansion	4	1
BARBEQUE	- computerised	3	3
BASIC	Programming Tips	3	6
	Programming hints	3	7
BASIC DISK	CMD'O' Tutorial	4	7
	CMD 'O' Tutorial	5	9
	Disabling BREAK	2	1
	NEWDOS/LDOS/DOSPLUS Basics compared	5	9
	Trapping Disk I/O errors	4	4
BASIC EDITOR	Full Screen Editor	3	2
	Review	4	10
BASIC Extended	with SNAPP Programs	4	5
BASIC L2	Adding REPEAT key	2	7
	Decimal to Hex conversion	4	3
	HEX Conversion	4	11

		Volume	Issue
BASIC L2 (cont)	LINE INPUT routine	3	1
	LISTeach page separately	3	5
	Recovering from NEW	2	2
	Set MEMSIZE within a BASIC program	2	6
	Useful PEEKS and POKES	1	6
BASIC TIPS	2 Programs in use simultaneously	4	11
	255 character line	3	8
	CHRS(23) uses	3	8
	Control codes in PRINT	1	4
	Control length of INPUT	2	2
	Counting usage of each variable	4	3
	Cursor character changed	3	5
	DATE using DEFN	4	3
	DOS Exit Addresses	3	11
	Date and Time routine	4	9
	Declaring Variables to speed up execution	3	5
	Disable Break/LIST	3	3
	Doodling on Screen	4	2
	Drawing Pie charts	5	5
	EDIT successive lines of a BASIC program	4	11
	Fast Data Entry	3	3
	Flash asterisk	5	4
	Flashing Asterisk	5	7
	FOR NEXT improvements	3	9
	INPUT Control characters and arrows	2	9
	INPUT expression instead of varaiables	4	4
	Keyboard Addresses	3	11
	LEFTS Tutorial	5	7
	LIST improved	3	11
	LPRINT CHRS(0-12)	4	7
	" " "	5	7
	LPRINT past Tab 64	4	1
	LPRINT changed to PRINT	2	3
	" " " "	3	10
	" " " "	3	11
	M/C Array Summation	4	10
	M/C from BASIC	1	2
	MERGING 2 BASIC programs	1	7
	" " " "	1	9
	" " " "	2	11
	MKSS and Disks	3	7
	Making Printer Paginate	4	3
	Maximum/minimum search	2	12
	More readable LLIST	4	1
	Not using THEN with FOR	2	11
	Omitting Semicolon	1	4
	POKING above 32767	1	5
	PRINTER finding status before printing	5	9
	PRINTUSING with Pounds sign instead of Dollars	3	9

		Volume	Issue
	" " " "	5	6
BASIC TIPS	PRINTUSING Tutorial	4	5
(cont)	PRINTUSING tips	3	5
	PRINTing Graphics	1	8
	Partial screen scroll	3	9
	Possible varaibles	2	3
	RUN Automatically	2	12
	RUN straight after CLOAD	3	8
	Renumber Routine	2	9
	Repetitive INKEYS	4	5
	Reverse Linefeeds	3	10
	STRING Functions	2	5
	Saving Memory	1	8
	Scan for Arrow key	4	7
	Set MEMSIZE in Program	2	7
	Slow LISTing	2	4
	Sort in Infinite BASIC	4	4
	Sort a multidimensional array	4	5
	Speed up execution by declaring variables	4	6
	Storing M/C routines	2	3
	String Handling	3	6
	TAB (132) in Basic	3	10
	Tidying Strings	3	11
	Useful POKES	3	11
	" " "	3	3
	Variables Listing	4	2
	Versatile Keyboard	3	9
BASIC-ENHBAS	Enhanced Basic	3	12
BASIC-PLUS	Enhanced Basic	3	3
BASICODE	On Video Genie	5	10
BBC MICRO	Compared with TRS	5	4
	" " "	5	5
	Interfacing to TRS	4	3
BIG 5 GAMES	Reviewed	3	8
BINARY	Explanation/Tutorial	1	3
BLIND STUDENTS	Assisted Teaching	2	9
BMON Monitor review		2	11
BOOLEAN ALGEBRA tutorial		5	5
BULLETIN BOARD - setting one upw		2	6
BUZZ with Expans.Interface relay		4	7
C COMPILER	Impressions	4	9
C	" " "	5	3
C/80 3.0 LANGUAGE Review /Mathpak		5	4
C PROGRAMS	Counting words in text	4	10
	Generating random Numbers	5	7
	More functionms	5	4
	Purging CP/M disks	5	6
	Terminal Program	4	10
CASH REGISTER	Using a Model 3	5	9
CASSETTE - see under CTR-CASSETTE RECORDERS and TAPE.			
CASSETTE LABELS	in large quantities	5	6
CENTRONICS	737 Printer Review	1	12
CHESS	Comparison of available programs	2	7
CHIP	Labelling before hacking	4	5
CHRS(0-12)	LPRINTING them	4	7

		Volume	Issue
CLEAN SLATE	Wordprocessor review	5	6
CLEAR SCREEN	ROM Calls	1	5
CLOCK	Adding Realtime clock	5	4
	MSF time	2	2
	Real time Battery-powered clock	5	5
	SIMUTEK Speed-up kit	2	5
	Speed Up clock	1	10
	Speed up	3	6
	Speed up	3	9
COLOUR COMPUTER	Assessed	3	10
	and DRAGON 32	4	3
COLOUR PRNTER	Review	3	5
COMMUNICATION	Adding and Testing RS232	4	7
	to BBC Micro	4	3
	Control using C language	4	10
	COMM-80 Interface	2	5
	DIY Terminal program	4	2
	Get BASIC via PRESTEL	5	3
	MODEL II	4	5
	MODEM/CD review	4	5
	Making a Null Modem	5	7
	Model 100 to Model I	5	7
	Model 100 to Model3	5	9
	Model I to II	4	1
	Model 4	5	6
	to NASCOM	2	9
	One TRS80 to another	2	1
	" " "	2	7
	PRESTEL 300 Service	4	7
	PRESTEL with LCOMM	4	11
	to SINCLAIR SPECTRUM	4	12
	ST80 Review	1	6
	ST80-III Review	2	5
	Single chip Modem	4	2
	TELESOFTWARE the reality	4	1
	TRS80 to TRS80	2	1
	" "	2	7
	With British Telecom 2A Modem	4	7
COMPAK Review	(Enhanced Basic)	3	6
COMPILERS	Accel 2 reviewed	3	9
	ZBASIC and Microsoft compared	4	6
COPYART	Wordprocessor review	4	2
CORPLAN	Business game review	3	9
CP/M	Freedom Board	2	5
	On 5 1/4 inch driv	3	1
	On Model 2	5	7
	Model 2,12 and 16	4	5
	Purge Utility in C	5	6
CP/M Users group	Address	4	4
CREATOR	Database review	4	11
CTR41	Cassette recorder Modifications	1	5
CTR80	Cassette recorder Modifications	1	6
CURSOR	Control in M/C	2	7
	Flashing	5	3
	Flashing in Level 2	1	9

	Volume	Issue
CURSOR Moving with DEF FN	3	1
DAISY WHEEL (Tandy) review	2	7
DATA ACQUISITION with a TRS 80	2	10
DATABASES - see individual entries under AIDS, Creator, Datawriter, Hobs, ISS, Maximanager, MDBS, Microfiles, Superpimx and Versafile.		
DATAWRITER Database review	4	12
and some problems	5	4
DATE Display in BASIC	4	2
using DEFN	4	3
DATE Program In BASIC	5	6
DATE ROUTINE Improvements	4	3
DATE and time In BASIC	4	9
DATE check for validity	4	5
DAY/DATE conversion (Basic)	3	12
DAY/DATE another routine	4	4
DAYS Between two dates	3	11
DEBOUNCE Adding to Level2	1	3
M/C Routine	1	4
and Display Delay	1	11
DEBUG (NEWDOS) adding COPY command	2	10
DEBUG " " " "	2	11
DECIMAL to Hex Conversion	4	6
" " In Level 2	4	3
DIGITAL to analogue conversion	1	5
DISK CONTROLLER Disabling	2	7
DISKS Double Density - see DOUBLE DENSITY		
DISKS Care - Model II	3	10
Input/Output without DOS (M/C)	4	5
Input/Output error trapping	4	4
Tranfer of files from Mod I-3	4	6
Without tears'	4	1
DISKS DRIVES Mixing 5 and 8 Inch	4	9
Eight Inch	1	12
Hard VS Floppy	5	6
DISK FILES Sequential	1	7
DISKS TRACK SEEKING Speeded up	2	2
DOODLE/BAS Drawing Lines on Screen	4	2
DOS First Impressions	3	3
Passing data to or from	4	7
DOSES - NEWDOS80, LDOS and DOSPLUS compared	5	9
DOSES - Individual - See under DOSPLUS, LDOS, NEWDOS and TRSDOS		
DOSPLUS compared with others	5	9
Double Density on Genie	3	11
Tutorial	4	1
DOTPRINT review	4	10
DOTWRITER 3 review	5	5
DOUBLE DENSITY Aerocomp board Review	4	4
Do-it-yourself	3	8
DOSPLUS Double Density	3	11
Further trials	4	2
Hardware systems available	3	8
LNW Board Problems	4	2
LDOS D/D using SOLE	3	12
NEWDOS	3	7

		Volume	Issue
DRAGON 32 compared to COLOUR COMPUTER		4	3
DUPLITAPE review		2	12
EASTER calculating date of		5	6
ECOLOGY simulations		3	11
EDGE CONNECTOR Pins explained, and TRS80/Genie differences		3	5
EDGE CONNECTOR Gold plated mod.		4	3
EDIT Basic screen editor		4	10
EDTASM MISOSYS Disk Patch		3	11
	NEWDOS patch	2	6
	Recover crashed text	2	7
	" " " " "	2	8
	" " " " "	4	10
	TANDY - Notes	2	6
EDTASM PLUS for Beginners		3	5
EDUCATION	A Call for Action	2	10
	Anagrams	4	2
	Bingo	4	2
	Bryants Softwear	3	10
	DIRECTORIES	4	5
	Software	3	3
	Software Reviews	3	5
	Teaching Alphabet	4	2
EDUCATIONAL	Software Reviewed	2	12
	Software Word Game	3	2
ELECTRIC PENCIL additions		2	2
	and EPSON printer	4	9
	under LDOS	4	9
	PENCIL PLUS review	5	7
	review	1	2
	Text recovery after reboot or crash	2	9
	" " "	3	5
	" " "	3	6
	Text recovery (ACCULAB)	3	2
	writing Machine code with	2	4
EPROM	Adding to Model I	3	4
EPROM Burner	Hints on Constructing your own	2	7
EPROM Burner	for sale	2	9
EPSON with ELECTRIC PENCIL		4	9
EPSON and LPRINTing CHRS(0-12)		4	3
EPSON FX80	Printer Review	5	6
EPSON MX80	Control Codes	3	6
	Demonstration	3	1
	Graftrax Graphics	3	12
	Initialising	2	11
	Joined up writing	3	5
	Review	3	6
EPSON MX80 FT	Review/Contrl.code	3	6
EPSON MX80 FT	3 review	4	1
EPSON MX80	Printing TRS80 graphic	4	5
EPSON RIBBON	Refills	4	5
	on the cheap	4	1
	revival	4	2
EPSON Setting up desired print type		4	12
EXATRON	Floppy tape assessment	2	12
	Memory Expansion	4	9

		Volume	Issue
EXP INTERF	LNW - Memory Prob	4	11
EXPANSION INTERFACES			
	Audio Computers Board	4	12
	GNOMIC 1000 Review	4	11
	Holmes 48K Board	3	1
	Holmes Board	3	6
	Improving Contacts	4	2
	Microdesign Board	3	1
	Printer Connections	5	5
EXPERTS	The 1982 List'	4	4
EXPERTS	The 1983 List'	4	6
EXTENDED SNAPP BASIC Review		4	5
FASTER Review of BASIC speed-up util.		4	5
But is FASTER really slower?		4	7
FORTH	Books about FORTH	4	2
FORTRAN	Possible Advantages	4	3
	Tutorial 1	4	5
	Tutorial 2	4	10
	Tutorial 3	4	12
	Tutorial 4	5	3
FREEDOM BOARD	CP/M Hardware Mod.	2	5
GAMES	Defence Command review	4	1
	Invaders from Space	2	12
GAMES PADDLE	Hardware Mod	2	2
	Improvements to it	2	2
	Interfacing with TRS	1	12
GARBAGE	Avoiding	4	6
	Cutting down on	4	7
	Removng	3	2
GNOMIC Acknowledgement		5	5
GNOMIC Expansion Interface		4	11
GNOMIC ROM	A-ROM review	3	10
GNOMIC ROM	B ROM Review	3	12
GRAPHICS	BASIC Graphic Editor	2	1
	DOTPRINT Review	4	10
	Displaying graphics	1	7
	Editing graphics	1	12
	Pixel Scrolling	3	8
	Powerdraw / Zgraph review	4	5
GSF and ACCULAB		4	2
GSF Subroutine to enhance BASIC		3	7
HANDICAPPED	Aid with TRS80	4	9
	Aids for	4	2
	Communication Program	4	3
HARDING AJ	Selling Software to him	1	6
HDBS DATABASE review		3	5
HEATHKIT H14	Lineprinter	1	8
HEX TO DECIMAL in BASIC		3	5
HEX to ASCII	Conversion	2	7
HIRES GRAPHICS	80-GRAFIX Review	3	2
	MODEL 3 Microfirm	4	5
	on Video Genie	4	3
	printing screen contents	4	5
HISTOGRAM generating		5	3
HL REGISTER	Saving	3	9
I/O BUS	on Model III	3	12
INDEXING Basic Program		3	5
INKEYS	Analysis and breakdown	3	1

	Volume	Issue
Repetitive use of INKEYS	4	5
INTEL 8253 Timer Chip	5	7
INTEL 8255 Prorammable I/O	2	6
INTERRUPTS Tutorial	5	4
IQLUG - to merge with NATGUG?	5	10
ISS DATABASE Review	3	8
Improvements in V3.0	3	12
JUMBO Flight simulator - amendments	4	5
KARADAWN Computer Review	4	1
KEYBOARD Addresses, memory map	1	7
" " " "	2	12
" " " "	3	11
" " " "	4	9
Contact Cleaning	3	8
and INKEYS used repetitively	4	5
Scanning for Arrow keys	4	7
Video Genie - memory map	2	10
LC COMPILER Review	5	3
LDOS Break key Disabling	3	12
Comparison with other DOSes	5	9
and Video Genie printer port	4	3
LDOS Double Density with SOLE	3	12
LEVEL 2 ROM RUN routine decode	2	12
LEVEL 3 BASIC review	1	5
LINE PRINTER (TANDY) review)	1	10
LINEPRINTER 7 - a modest program	4	5
LISP 80 INTERPRETER review	4	10
LLIST with multistatement lines	4	1
LMOFFSET Getting rid of it	4	1
Tutorial	3	7
LNW Double Density Problems	4	2
LNW INTERFACE Memory Problems	4	11
LOAD 80 Availability	5	3
LOWER CASE AV-1 Board review	4	4
" " "	4	6
Converting ASCII text from Upper to Lower case	2	1
Coverting L/C chars. in string to Upper case	4	5
Doing it yourself'	1	11
Driver routine	4	4
Hardware Lowercase Driver	2	7
NEWDOS lower case driver	2	3
POUNDS sign	2	4
and Reverse Video	2	6
and SCRIPSIT	4	4
Tandy software driver	2	3
Tand Lowercase Hardware	3	2
converting TEXT files	2	11
Adding True Descenders	3	2
on Video Genie	3	5
LPRINT change to PRINT by POKEing	2	3
LPRINTING CHRS(0-12)	4	3
" " "	5	7

NATGUG NEWS

Volume 6 Issue 4 October 1984

OFFICIAL JOURNAL OF THE

National TRS-80 & Genie Users Group.

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.

The Group maintains two software libraries (Models I and II) which are free to members. Library lists are obtainable from the Secretary.

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

CONTENTS

1. Information on the group

3. Editorial

3. Guidelines for contributors

3. Library information

4. Members' letters

6. Problem section

6. Illegal Z80 commands and how to obtain them

7. LDOS, Newclock 80 and SOLE

8. Disk copying between TRS-80, CP/M and MSDOS formats

9. Derived numeric functions

10. Printer review - Quen-Data DWP 1120

12. Review of Floyd Structured BASIC

13. More about pound signs

16. Why you get rounding errors

21. Passing parameters to code from DOS

23. Model I/III incompatibility and garbage collector notes

28. Patches to Misosys disassembler disk version 2.4

29. Spooling DOTPRINT printer output to disk (part 2)

31. For sale and wanted

EDITORIAL

Sorry about the lateness of this issue, but I had a great deal of work to do in connection with the Swindon Workshop.

I am writing this the day after the workshop, which was the most successful one yet, judging by the numbers who attended, and the favourable comments Brian and I received. It was a joint exercise, for both NATGUG and IQLUG members, and members of both groups seemed to get on very amicably. Of course, there were several members like myself who have both a QL and a TRS-80 or Genie, and some experiments were conducted in communicating between a TRS-80 and a QL using modems and the hotel's internal telephone system. Partial success was achieved, after a couple of hours of work, but several problems still remain.

In case you think I have abandoned my Model I and Model II completely, in favour of the QL, I use the Model I monitor on the QL, and the QL uses the Model I as a serial/parallel converter and print buffer! I shall be developing a C compiler for the QL on the Model II, initially as a cross-compiler, and then porting the code onto the QL to create a resident compiler.

My apologies to those of you who sent in long articles for inclusion in the last couple of months, which have not yet been used. I hope to get the next issue out very soon after this one, and your articles will appear then, in the November issue.

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 hand-written material, write legibly.

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

Leon Heller

LIBRARY INFORMATION

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

MEMBERS' LETTERS

I enclose a copy of a letter I sent to PCW, arising out of some advice to buy an IBM PC. Someone has to do Tandy's publicity for them.

If anyone knows of a word-processing program which is compatible with LDOS filters, I should be grateful to hear of it.

Dear Editor

A quick comment on the solution Tony Hethermgton suggested for Spanish and other European accented characters. Tandy offers a system that would be suitable for occasional use, lacking the ability to redefine (easily) the keyboard. The Model 4 with Superscripsit and a Tandy printer can print on paper any of the characters needed for any European language. On the screen however one will only have a code (for instance I see @8 where I want a pound sign in the text). The number of user-definable characters that you can enter at the Model 4 keyboard in this way is limited to 20. However help is at hand - in the Model 100. It can generate any of the extended Tandy character set of graphics and accented characters (coded from 32 up to 255) on screen, but only by using two-key or three-key entry; for example hitting the 'graph' and 8 keys together gives the £ sign on screen, which will also appear on the Model 4's screen when uploaded (after a simple offset has been applied to compensate for the difference of 20hex in the two machines' codes!).

If there exists a word-processing program that uses the DOS keyboard and printer drivers of one of the the improved DOSes, like LDOS, on the Tandy Model 3 or 4, there would be little difficulty or expense in filtering the keyboard or the printer the way one wants - Dvorak, EBCDIC, ... If anyone knows of such a program I would be interested in hearing of it.

Just to exercise your printer, here are a few of the possibilities -
é â § ç ù Ü ñ ¿ ¢ @ ^ ~ A Ü Ö ß .

Yours faithfully

J.A. Negus
Bessas
07150 Vallon Pont d'Arc
France

Thank you for the July and August Natgug News which recently arrived. I had in fact already received them. Perhaps it was a response to my renewal cheque. Unfortunately the envelope was open - the result of some curious form of torture that the French Post Office enjoys inflicting on its passengers. This is of no importance as long as the September issue was not lost en route. That would be serious.

I keep thinking of writing a review of Superscripsit for the News but hesitate as it may be coals to Newcastle; I joined too recently to know what has been published already.

I find Superscripsit a good functional program with only one major fault - its apparent inability, with a Tandy DMP 200 printer, to print out columns and tabs correctly aligned:

200
100

These two numbers were perfectly aligned one above the other on the screen. This fault I am quite prepared to undergo for its proportional printing and other

qualities. I would be interested all the same to hear from anyone who can from experience compare it to other programs. If there was a demand for it I would be happy to keep a file on WP programs at the disposal of members, especially those who contribute.

Does anyone know of comms programs for the III/IV and the 100 capable of handling split baud rates, i.e. 1200/75?

J. Negus
Bessas
07150 Vallon Pont d'Arc
France

(* Try Don Bannister on Belfast (0232) 668899 for 1200/75 baud comms software. LFH *)

I was interested in a couple of articles in the last issue of Natgug in using our machines in Amateur Radio. Apart from Peter Hall, you are the only other licensed amateur that I know of in the group, although I thought that there were more of us in the past.

Regarding Packet radio, I am very interested in a short introductory article by Bob Richardson, an American amateur, in last month's Ham Radio, illustrating the use of a TRS-80 for Packet, including software UART. His company's books used to be imported by GNOMIC, but now they've gone, I am trying to get the ones on Packet Radio direct from him. I'll keep in touch on my progress.

Bob Richardson's books on RTTY are specifically for the TRS-80, Models I and III, but as he again uses the concept of software-generated UARTs, and covers both multi-speed RTTY and CW, they may be of use for a Genie. The volumes covering this are his Disassembled Handbook, Volumes 4 and 5.

My Model 4 generates a fair level of RF noise, even with the additional screening around the circuit boards. With an internal aerial on 2 Metres, I need a strong FM signal to suppress this. SSB signals *have* to be local ones! However, the good news is that virtually all the noise is via the aerial, (the Model 4 is about 6' from the transceiver, on a seperate mains socket) and I believe that most of the interference will go once the aerial is outside.

Robert Vieira (G8AJF)

6, Lechmere Avenue
CHIGWELL
Essex
IG7 5ET

(* There are still quite a few amateurs in the group, I believe. Leighton Davies for one! LFH. *)

I would like to get in touch with other Genie III users.

DV Nudd,
13 Elphinstone Road,
Southsea,
Hants. PO5 3HP.
Tel: (0705) 823052.

PROBLEM SECTION

I am also writing to ask whether you know anything about the Quendata dwp-1120 printer which I have recently purchased. Scripsit drives it without any problems but access to the non-keyboard characters and the printer commands is not possible from scripsit. I have been told that the new super-scripsit package allows control and special characters to be programmed in but as I can't trade my package in for an upgrade the cost puts it beyond my reach.

I have therefore been trying to modify my existing scripsit package, unsuccessfully, and so if you, or any of the club members, can help with modifications to the scripsit package like the Flextext program for the Centronics 737 I would be pleased to know.

I also purchased LDOS recently form Tandy's but am unable to use it with Scripsit. Do you know of any mods to make it run under LDOS? I was particularly looking forward to being able to run the Minidos features with scripsit and so help would be most welcome.

Should anyone be able to oblige with mods for scripsit I will try and write a review for a later issue.

This letter was printed on the Quen-data.

Mr. J.W. Gibbs
16, Romany Close
PORTSLADE
EAST SUSSEX
BN4 2BA

ILLEGAL Z80 COMMANDS and how to obtain them.
==========

The Z80 has a capability for a total of 3756 instructions. if all the bit patterns available are used. However, ZILOG only tell us about 691, thes are known as the legal commands.

This means that there are over 600 combnations left un-assigned. But many of these combinations actually do have functions and work on most later versions of the Z80 microprocessor.

Unfortunately, as these commands are not documented by Zilog, the assemblers would not accept the commands. They are as follows.

The SLL shift logical left family.
(These are available under the SLS command in the new ZEN).

Hex	Command		Hex	Command.	
CB 37	SLL	A	CB 33	SLL	E
CB 30	SLL	B	CB 34	SLL	H
CB 31	SLL	C	CB 35	SLL	L
CB 32	SLL	D	CB 36	SLL	(HL)

There is also a method of accessing the High and Low bytes of the Index registers IX and IY. As there are so many commands that come under this banner I will not list them, but here is a quick method of deriving them.

If you wish to access the IX pair then define the one-byte of DD, or FD for the IY pair. You then simply type in the command as usual but using the H register for the High byte and the L register for the Low byte. Obviously you cannot use the H or L registers in this sub set.

(E.g., To swap the high and low bytes of IY and IX.

Hex	Nmonic	Assembler entry.
DD 7C	LD A,HX	DEFB 0DDH
		LD A,H
DD 65	LD HX,LX	DEFB 0DDH
		LD H,L
DD 6F	LD LX,A	DEFB 0DDH
		LD L,A
FD 7C	LD A,HY	DEFB 0FDH
		LD A,H
FD 65	LD HY,LY	DEFB 0FDH
		LD H,L
FD 6F	LD LY,A	DEFB 0FDH
		LD L,A

The commands available to you are:

LD	ADD	ADC	SUB	SBC	INC	DEC
AND	OR	XOR	CP			

There are two remaining commands which are sub-commands of the functions OUTI, OUTD, OTIR, OTDR and INI, IND, INIR, INDR. These are IN (HL),(C) (Input to location (HL) from port C) and OUT (C),(HL) (Output from location (HL) to port C).

Hex	Command.
ED 70	IN (HL),(C)
ED 71	OUT (C),(HL)

The above are only some of the un-documented commands but are really the only ones worth bothering with.

P.Knaggs.

LDOS, NEWCLOCK 80 and SOLE.

I have been using LDOS for some two to three years on my TRS-80 Model One with no problems whatsoever. Even when I decided to go to Double Density with an Aerocomp Doubler the conversion of my software was a smooth and simple matter even converting LDOS for use with SOLE. For those of you who don't know what SOLE is, it is a very clever piece of software which enables the Model I to Boot in Double Density. Ordinarily the Model I cannot, as the ROM is only programmed to read Single Density.

However one thing that annoyed me was having to enter the Date and, in my case the Time, (as I use the Real-Time clock in several of my programs) every time you Boot. So I eventually decided to buy a Battery Clock and decided on the NEWCLOCK 80, manufactured by Alpha Products, 79-04 Jamaica Ave, Woodhaven, NY 11421. I therefore sent off my order for a Model I NEWCLOCK 80 with instructions to debit my Access card (Mastercard in the States). I checked with Alpha's adverts in an edition of 80-MICRO and, sure enough, the Clock was compatible with LDOS, NEWDOS and DOSPLUS and they could supply a disk of patches, programs etc to run the Clock. Two weeks later the clock arrived. It cost altogether $79.95 which converted on my Access Card to £65.00. Not too bad considering the plunging pound!.

Installing the software to run the clock proved to be not as easy as I had thought, and caused me the most problems I have ever experienced with LDOS.

Having read the instructions thoroughly I eagerly installed the clock and proceeded to Patch my LDOS. First I used the patch they supplied. Dismay, I did'nt work! On Booting the time was initialised but the Date was complete rubbish. It read things like "Thu, Thu, 1986" or "Tue, 0, 1989". Next I tried a Patch supplied by the LDOS people. This had exactly the same effect. I tried to see what the patch did to SYS0 but could not see anything which would affect the operation of the patch. Finally I wrote to Alpha asking them what I could do as I was still having to enter the date manually, although the time was ok. They very promptly replied telling me to use another program they had supplied to initialise the date & time. This program, called HIGHTIME/CMD, was run after booting and sat in high memory, protecting itself, constantly updating the time & date. Thus the problem was solved.

I can recommend the NEWCLOCK 80 to anyone interested in a Battery Clock. Using HIGHTIME or, if you are using single density LDOS, the Patch supplied, the time is constantly updated. If you use Format or Backup the clock is still accurate after you have finished as the clock simply puts the correct time back into the System. If you do decide to purchase a NEWCLOCK 80 for use with a SOLEd LDOS, remember the above, the Model I LDOS patches will NOT work. If any one knows of a patch to use with a SOLEd LDOS for NEWCLOCK 80 I would like to hear from them, as with HIGHTIME it uses up some of my needed memory, whereas the Patch for Single Density LDOS does not. If anyone has any problems with LDOS & NEWCLOCK 80 please contact me, I will do my best to help.

Steve Ord,
22 Lymescote Gdns,
Sutton,
Surrey, SM1 2PN.

```
**************************************************************
*   DISK COPYING BETWEEN TRS-80, CP/M & MSDOS FORMATS        *
*                                                            *
*    I may be able to help anyone who needs to copy          *
*    data files between TRSDOS or NEWDOS-80 and 40-track     *
*    single-sided CP/M or MSDOS/PCDOS formats. Maximum       *
*    file size 128 K.                                        *
*                                                            *
*    This note was written on a Model III using              *
*    Scripsit-Plus; if you are reading it, it has            *
*    been copied onto a CP/M system and printed out          *
*    using Wordstar and a daisy-wheel printer.               *
*                                                            *
*  CONTACT Chris Currie, 14 Keston Road London N17 6PN.      *
*                                                            *
*          Tel.: 01-801-2185 (evenings only)                 *
*                                                            *
*(BRIAN: Please note new address and phone no. this time)    *
**************************************************************
```

DERIVED NUMERIC FUNCTIONS

```
ARCCOS (X)      = -ATN(X/SQR(-X*X+1)) + 1.57079633
       returns the inverse cosine of X; ABS(X) < 1

ARCCOT (X)      = -ATN(X) + 1.57079633
       returns the inverse cotangent of X

ARCCOSH (X)     = LOG(X+ SQR(X*X-1))
       returns the inverse hyperbolic cosine of X; X >= 1

ARCCOTH (X)     = LOG((X+1)/(X-1))/2
       returns the inverse hyperbolic cotangent of X; ABS(X) > 1

ARCCSC (X)      = ATN(1/SQR(X*X-1))+(SGN(X)-1)*1.57079633
       returns the inverse cosecant of X; ABS(X) > 1

ARCCSCH (X)     = LOG((SGN(X)*SQR(X*X+1)+1)/X)
       returns the inverse hyperbolic cosecant of X; X > 0

ARCSEC (X)      = ATN(SQR(X*X-1))+(SGN(X)-1)*1.57079633
       returns the inverse secant of X; ABS(X) >= 1

ARCSECH (X)     = LOG((SQR(-X*X+1)+1)/X)
       returns the inverse hyperbolic secant of X; 0 < X < 1

ARCSIN (X)      = ATN(X/SQR(-X*X+1))
       returns the inverse sine of X; ABS(X) < 1

ARCSINH (X)     = LOG(X+SQR(X*X+1))
       returns the invers hyperbolic sine of X

ARCTANH (X)     = LOG((1+X)/(1-X))/2
       returns the inverse hyperbolic tangent of X; ABS(X) < 1

COSH (X)        = (EXP(X)+EXP(-X))/2
       returns the hyperbolic cosine of X; 1 < X

COT (X)         = 1/TAN(X)
       returns the cotangent of X; X <> 0

COTH (X)        = EXP(-X)/(EXP(X)-EXP(-X))*2+1
       returns  the hyperbolic cotangent of X; X <> 0

CSC (X)         = 1/SIN(X)
       returns the cosecant of X; X <> 0

CSCH (X)        = 2/(EXP(X)-EXP(-X))
       returns the hyperbolic cosecant of X; X <> 0

LOG  (X)        = LOG(X)/LOG(A)
       returns the base A logarithm of X; A > 0 , X > 0

LOG  (X)      = LOG(X)/2.30258509
       returns the common logarithm of X; X > 0

MOD  (X)        = (X- INT(X/A)*A)*SGN(X/A)
       returns X modulo A
```

```
Pi              = 4*ATN(1)
        returns Pi.

SEC (X)         = 1/COS(X)
        returns the secant of X; X () PI/2

SECH (X)        = 2/(EXP(X)+EXP(-X))
        returns the hyperbolic secant of X

SINH (X)        = (EXP(X)-EXP(-X))/2
        returns the hyperbolic sine of X

TANH (X)        = -EXP(-X)/(EXP(X)+EXP(-X))*2+1
        returns the hyperbolic tangent of X
```

Where the constant 1.57079633 has been used it can be replaced by 2*ATN(1) as it represents Pi/2

Of course the above functions will not work if X exceeds the limits for that function.

Ken Robinson

PRINTER REVIEW - QUEN-DATA DWP 1120

When I first bought a printer I could not afford a Daisy Wheel because they cost over a £1000. I settled for the Epson MX80 and it has never had a moments down-time. One of the features of the Epson is that it has a disposable print head but despite getting through many rolls of paper I have not replaced it yet.

The ribbons were not cheap and I tried several ecconomies including getting refills from the States. Many refills seem to have uneven inking and I have since found that Inmac sell the cartridges for less than £3.

Although the Epson MX80 has a reasonable type face it is still "almost correspondance quality" and there are now better quality and faster dot matrix printers, including one from Epson. The better dot mattrix printers have more and smaller pins and can be hard to detect from a Daisy Wheel printout. However they are expensive and very hard on ribbons.

As I needed a better quality than I could get from the MX80 I decided to look at the new cheap typwriter and Daisy Wheel printers. The typewriters look attractive but they have hidden costs in that the interface can cost over £100 and some do not work properly on the TRS-80 without special software drivers.

I decided to look at the Juki printer and eventually came back with a new machine which is called the Quen-Data DWP 1120. At first I thought this was a cheap Qume printer but it turns out that it was untill recently called the Daisy Step 2000. It sells for £250 inc and I got it from Ahters in Harlow. One of the assets of this firm is that you can park nearby (useful in London) and they have a stock which they demonstrate.

The Quen is a 16 cps printer and is therefore very slow compared to the Epson but it can use bi-directional logic seeking printing. In simple terms this means it prints alternate rows from the opposite end and saves having to carriage return to start the next row.

I found that the printer can be set up with DIP switches and codes can be incorporated in the software. The optional settings include the CR+LF which is required by the model 1. The settings also include two modes which seem to baffle me a little. They are called Line and Serial (not serial interface) and as far as I can understand the Serial mode allows you to send out characters one at a time as with a typewriter. However I found this cumbersome to use on the TRS-80 and I believe it is intended to be used with a seperate Keyboard.

This seems to be a future trend and possibly the typewriters of the near future will consists of a printer and a seperate keyboard. At the PCW show I met the MD of Juki who explained that they had a new machine that would have its own Word Processor ROMS which could be used when an optional keyboard was added.

One asset of the Quen is that it can handle 13 inch paper which means that A4 sheets can be inserted sideways. As with many of the these Daisy Wheels it takes single sheets and requires an additional tractor machanism. I found the teletype paper roll will work but a holder is desirable (not avialable). The Epson roll holder works quite well but I guess its not too difficult to make one.

The Daisy wheels are Qume compatable but I did find one Qume wheel that was not in the correct order. However there are a fair number of wheels available at a reasonable price. I wanted a selection of wheels and this was a factor in my choice. Qume wheels are easily obtainable and cheap when compared to other machines. £5 as compared to £15-20.

The instruction book is typical of computer accessories and is very limited. It contains a lot of cryptic codes and I am sure it makes good sense when you get to know it. At least the machine worked from the start but some of the more complex features require further work. Better still it could do with a more user friendly manual as with the Epson.

One of the features of these new printers is that they have typewriter quality but dot mattrix flexiblity. The line spacing can be set in increments of 1/48 which is better than the typewriter single or double line spacing. The character spacing is varyable in 1/120 ths. This seems to have been choosen because it is divisable by 10/12/15 which are the common character sizes. According to the manual it is possible to get proportional spacing 'by software control from the outside computer'. I have not found out what that means and would welcome suggestions.

For normal typewriter usage these features are not very important but for charts and forms it is useful to be able to have variable line spacing and character widths to improve readablity. It is also eccomical for long reports because the 12 cpi wheel can allow more characters per line and more lines per page.

The standard 10 cpi and 6 lpi was orginally used because the type quality was poor. Now with carbon ribbons and even type pressures it is easy to read 12 cpi and 8 lpi. I have only one 15 cpi wheel and it gives excellent numbers but I think the letter spacing is uneven. This may be due to a poor wheel and not a fault of the machine. I thought I had a big use for a 15 cpi wheel as I used condensed a lot on the Epson. It was almost essential for Visicalc but I found that I could get a similar result on the Daisy wheel by having reduced photocopies made. However there are times when this is not practical as you need larger paper for the original copy.

The bottom line is that the Quen Data is a cheap but good quality printer and seems to be compatable with the TRS-80. The Epson is better for listing and quick drafts. The printer has a small buffer but would benefit from the 64K buffer available and I have one under construction. The wheel supplied does not have a £ and the alternative wheel available is worth the extra £5. I pesonally prefer the 12 cpi Prestige Elite but this has no £ sign but has all the ASCII symbols.

Derek Trayler
88. Grosvenor Drive,
Hornchurch,
Essex RM11 1PW.

REVIEW OF FLOYD STRUCTURED BASIC

FSB is a modification of standard Level II/Disk Basic available from Microcomputer Applications for around £14, including VAT and postage. The disk version includes configurations for Mod I or Mod III, tape or disk, and 16K, 32K, or 48K memory sizes. The program works on Dosplus 3.3 and Newdos80 as well as TRSDOS, so it's probably compatible with most Disk Basics. It isn't compatible with the Accel series of compilers. It occupies only 3/4 KB at the top of memory.

The modification allows the following additional statements and functions: REPEAT-UNTIL, WHILE-ENDWHILE, PROC, CALL, and ON-CALL. In addition the IF function is extended so that the apodosis can cover several lines, provided that it ends with an ENDIF statement. The PROCs are simply named subroutines: no parameters or local variables are allowed. If named procedures are used, the program must begin with a :CLEAR statement (note the leading colon). Moreover, procedures, like traditional subroutines, have to follow the main body of the program, whereas a structured language should allow you to define them at the beginning. One can get round this by treating the main body as a procedure and putting a CALL to it near the beginning, followed by the other procedures, e.g.:

```
4 :CLEAR
5 CALL MAIN:END
10 PROC TEST
20 PRINT "THIS IS A TEST"
30 RETURN
35 PROC MAIN
40 PRINT"TESTING"
50 CALL TEST:RETURN
```

It's also unfortunate that ENDWHILE rather than WEND was chosen to terminate a WHILE loop, because the latter would have provided greater compatibility with Model 4 and MBasic.

In spite of its limitations, FSB seems a useful tool for program development, and well worth the money. It should make it easier to convert BBC Basic programs for the TRS-80. Combined with one of the better Disk Basics, e.g. Apparat's, it gives you one of the most flexible Basic interpreters available on any microcomputer.

Chris Currie

MORE ABOUT POUND SIGNS

Thank you for including in the May issue Chris Currie's ingenious contribution on this subject. I have used his method with minor modifications in the BASIC program on the next page. This admittedly trivial program illustrates techniques which may be useful in longer programs to display messages including pound signs on the screen with the option to obtain hard copy if desired. The program as listed runs on the Model 3 and on the Model 4 in Model 3 mode and works both with a Tandy Line Printer VI and with an Epson FX-80 printer. The messages at lines 90, 110,240 & 260 are displayed on the screen with the dollar signs replaced by pound signs. On typing P in response to the message at line 340, a hard copy of the screen display is obtained.

HARD COPY PRINT-OUT. Because Tandy use different ASCII values for the pound sign on the screen and on the line printer, the normal screen print facility (SHIFT, and *) will not work properly in this case. While there may be no universally satisfactory solution to this problem, the method presented provides a general solution in cases where the programmer can foresee that a screen print may be desired. On selecting the screen print option, line 390 calls the assembler subroutine at lines 400-420 which is loaded into a string variable at lines 30-50. Explanations of the assembler subroutines for both Model 3 and Model 4 modes are given in the Appendix.

MODEL 4 MODE. Apart from changing CMD"S" to SYSTEM at line 360, the changes needed to the program for Model 4 mode are confined to lines 400 to the end, viz.

```
400 REM **** DECIMAL VALUES CORRESPONDING TO S/R $PPRSC4 ******
410 DATA 33,0,0,6,1,62,15,239,254,1,32,4,14,163,24,1,79,62,6
420 DATA 239,35,125,254,80,32,233,14,13,62,6,239,36,46,0,124
430 DATA 254,24,32,220,201
440 END
1000 REM *** SUBROUTINE TO PRINT MESSAGE INCLUDING POUND SIGNS *
1010 CR=0 : DP=0 : LG=LEN(MS$)
1020 FOR I=1 TO LG : CR=CR+1
1030 IF MID$(MS$,CR,1)="$" THEN PRINT MID$(MS$,DP+1,CR-DP-1);:
     PRINT CHR$(0);CHR$(PS); : DP=CR
1040 NEXT I
1050 IF DP<LG THEN PRINT MID$(MS$,DP+1,LG-DP);
1060 RETURN
```

HARDWARE VARIATIONS. On visiting a friend with a Model 4P and a Tandy DMP-120 printer, it was disconcerting to find that instead of pound signs, funny foreign characters appeared on the screen and nondescript blobs on the printer. On this system, it was necessary to change PS from 1 to 6 at line 20 of the BASIC program and the relevant bytes from 163 to 195 in the assembler subroutines.

Keith Wagstaff, 16 Majors Place, Falkirk. Tel 0324 22932

```
10 REM*EXAMPLE DISPLAYING POUND SIGNS ON SCREEN & ON HARD COPY (MODEL 3)*
20 CLEAR 500:CLS:PS=1     'ASCII FOR POUND SIGN ; CHANGE TO 6 FOR MODEL 4P
30 SUB$=STRING$(40,0)     'LOAD SUBROUTINE $PPRSC3 INTO STRING
40 V=VARPTR(SUB$):L=PEEK(V+1)+256*PEEK(V+2):IF L>32767 THEN L=L-65536
50 FOR I=0 TO 39 : READ N : POKE L+I,N : NEXT I : POKE L+9,PS
60 PRINT"*************************************************************"
70 PRINT"*     THIS PROGRAM CALCULATES CHANGE FOR A CASH TRANSACTION     *"
80 PRINT"*************************************************************"
90 MS$ = "          ENTER VALUE OF TRANSACTION ($$.PP) "
100 GOSUB 1000 : INPUT VA
110 MS$= "            ENTER CASH OFFERED ($$.PP)          "
120 GOSUB 1000 : INPUT CA
130 IF CA >= VA THEN 150
140 PRINT:PRINT"            ** INSUFFICIENT CASH OFFERED **":PRINT:GOTO 340
150 REM ****CALCULATE CHANGE IN PENCE (INTEGER) **************
160 CP%= INT(100*CA) - INT(100*VA)
170 N5% = INT(CP%/500) : CP% = CP% - 500*N5%
180 N1% = INT(CP%/100) : CP% = CP% - 100*N1%
190 FY% = INT(CP%/50)  : CP% = CP% - 50*FY%
200 TY% = INT(CP%/20)  : CP% = CP% - 20*TY%
210 TE% = INT(CP%/10)  : CP% = CP% - 10*TE%
220 FV% = INT(CP%/5)   : CP% = CP% - 5*FV%
230 TW% = INT(CP%/2)   : P1% = CP% - 2*TW%
240 MS$ = "              NUMBER OF $5  NOTES IN CHANGE IS"
250 GOSUB 1000 : PRINT N5%
260 MS$ = "              NUMBER OF $1  NOTES IN CHANGE IS"
270 GOSUB 1000 : PRINT N1%
280 PRINT TAB(14) "NUMBER OF 50P PIECES IN CHANGE IS"; FY%
290 PRINT TAB(14) "NUMBER OF 20P PIECES IN CHANGE IS"; TY%
300 PRINT TAB(14) "NUMBER OF 10P PIECES IN CHANGE IS"; TE%
310 PRINT TAB(14) "NUMBER OF  5P PIECES IN CHANGE IS"; FV%
320 PRINT TAB(14) "NUMBER OF  2P PIECES IN CHANGE IS"; TW%
330 PRINT TAB(14) "NUMBER OF  1P PIECES IN CHANGE IS"; P1% : PRINT
340 PRINT"     PRESS C TO CONTINUE, P FOR SCREEN PRINT, E TO END";
350 K$=" ":K$ = INKEY$ : IF K$="C" THEN CLS : GOTO 60
360 IF K$="E" THEN  CLS : CMD"S"
370 IF K$<>"P" THEN 350
380 V=VARPTR(SUB$):L=PEEK(V+1)+256*PEEK(V+2):IF L>32737 THEN L=L-65536
390 DEF USR1=L : IDUM=USR1(0) : GOTO 350
400 REM **** DECIMAL VALUES CORRESPONDING TO S/R $PPRSC3 *******
410 DATA 14,16,33,0,60,6,64,126,254,1,32,2,62,163,205,59,0,35,16
420 DATA 243,62,13,205,59,0,13,175,185,32,231,201,0,0,0,0,0,0,0,0,0
430 END
1000 REM *** SUBROUTINE TO PRINT MESSAGE INCLUDING POUND SIGNS ***
1010 X1 = PEEK(16416) : X2 = PEEK(16417)  'CURSOR POSITION
1020 IF X2=63 THEN IF X1>191 X1=X1-64 'BOTTOM LINE SCROLLING ?
1030 PRINT MS$;:PRINT" ";
1040 DU$="":V=VARPTR(DU$):POKE V,64:POKE V+1,X1:POKE V+2,X2:N=1
1050 FOR I=1 TO 64 : X3=INSTR(N,DU$,"$") 'REPLACE EACH $ BY POUND SIGN
1060 IF X3=0 RETURN ELSE X4=X1+X2*256+X3-1:POKE X4,PS :N=X3+1
1070 NEXT I
1080 RETURN
```

APPENDIX - Details of Machine Code Subroutines

```
                00100 ;$PPRSC3 - PRINT SCREEN INCLUDING POUND SIGNS - MODEL 3
0000 0E10       00110 PPSCR3  LD      C,16     ;LINES ON SCREEN,MODEL 3
0002 21003C     00120         LD      HL,3C00H;START OF VIDEO RAM
0005 0640       00130 NXTLN   LD      B,64     ;LENGTH OF LINE ON SCREEN
0007 7E         00140 NXTCHR  LD      A,(HL)
0008 FE01       00150         CP      1        ;MODEL 3 POUND SIGN ?
000A 2002       00160         JR      NZ,PRINT
000C 3EA3       00170         LD      A,163    ;POUND SIGN ON PRINTER
000E CD3B00     00180 PRINT   CALL    003BH    ;$PRCHAR - STANDARD ROM S/R
0011 23         00190         INC     HL
0012 10F3       00200         DJNZ    NXTCHR   ;END OF LINE?
0014 3E0D       00210         LD      A,13     ;CARRIAGE RETURN
0016 CD3B00     00220         CALL    003BH    ;$PRCHAR - STANDARD ROM S/R
0019 0D         00230         DEC     C
001A AF         00240         XOR     A
001B B9         00250         CP      C        ;BOTTOM OF SCREEN?
001C 20E7       00260         JR      NZ,NXTLN
001E C9         00270         RET
0000            00280         END     PPSCR3
00000 Total Errors

                00100 ;$PPRSC4 - PRINT SCREEN INCLUDING POUND SIGNS - MODEL 4
0003 210000     00110 PPRSC4  LD      HL,0     ;TOP LH CORNER OF SCREEN
0003 0601       00120 NXTCHR  LD      B,1
0005 3E0F       00130         LD      A,15     ; @VDCTL
0007 EF         00140         RST     28H      ;RETURN CHAR FROM SCREEN POSN
0008 FE01       00150         CP      1        ;IS IT POUND SIGN ?
000A 2004       00160         JR      NZ,NOFIND
000C 0EA3       00170         LD      C,163    ;POUND SIGN ON PRINTER
000E 1601       00180         JR      PRINT
0010 4F         00190 NOFIND  LD      C,A
0011 3E06       00200 PRINT   LD      A,6      ; @PRT
0013 EF         00210         RST     28H      ; SEND CHARACTER TO PRINTER
0014 23         00220         INC     HL
0015 7D         00230         LD      A,L      ; COLUMN NO. ON SCREEN
0016 FE50       00240         CP      80       ; END OF LINE ?
0018 20E9       00250         JR      NZ,NXTCHR
001A 0E0D       00260         LD      C,13     ; CARRIAGE RETURN
001C 3E06       00270         LD      A,6      ; @PRT
001E EF         00280         RST     28H
001F 24         00290         INC     H        ; LINE NO. ON SCREEN
0020 2E00       00300         LD      L,0      ; START OF LINE
0022 7C         00310         LD      A,H
0023 FE18       00320         CP      24       ; END OF SCREEN ?
0025 20DC       00330         JR      NZ,NXTCHR
0027 C9         00340         RET
0000            00350         END     PPRSC4
00000 Total Errors
```

I have a TRS-80 Model I with RS-232, and a Sharp 1500 (Tandy PC-2) also with an RS-232 interface. I would like to store programs and data for the pocket computer on the TRS-80, for transfer to ESF wafer.

Can anyone help?

John Wickens,
8 Orchard Street,
Sutton,
Surrey.

WHY YOU GET ROUNDING ERRORS

All digital computers are based on the binary number system; the TRS 80 is no exception. The binary system is based on the number two; the decimal or denary system is based on ten and the hexadecimal or hex system is based on sixteen.

Consider 1458 in base 10, it represents 1x10^3 + 4x10^2 + 5x10^1 + 8x10^0, while the number 1011 in base two represents 1x2^3 + 0x2^2 + 1x2^1 +1x2^0.

In base 10,10 represents 1x10^1 + 0x10^0 = 10. Therefore,we need symbols that can occupy the right position to represent zero to nine. The symbols we use are 0,1,2,3,4,5,6,7,8, and 9.

In base two, 10 represents 1x2^1 + 0x2^0 = 2, so in base two arithmetic, we need digits representing only zero and one. It is useful to be able to count in both bases.

base 10	1	2	3	4	5	6	7	8	9	10
base 2	1	10	11	100	101	110	111	1000	1001	1010

One of the niceties of dealing with base two numbers is that there are only four addition facts to remember:

1+1 = 10 1+0 = 1 0+1 = 1 0+0 = 0

Addition in base two is carried out in the same way as addition in base ten. To add 10101 to 11011:

```
    1 1 0 1 1
    1 0 1 0 1
  --------------
  1 1 0 0 0 0
  --------------
    1 1 1 1 1
```

Multiplication in base two requires the following four facts to be grasped:

1x1 = 1 1x0 = 0 0x1 = 0 0x0 = 0

To multiply 111 by 101 :

```
          1 1 1
      x   1 0 1
  --------------
          1 1 1
        0 0 0
      1 1 1
  --------------
    1 0 0 0 1 1
  --------------
      1 1 1
```

An integer word in the TRS 80 consists of two 8-bit integer bytes, a word being the basic unit of memory used

for storage.

Lets calculate the binary equivalent of 56. To do this, we use what some people know as the Chinese remainder theorem. The process is simple:

divide 56 by 2, note the remainder:
divide the quotient by 2 note the remainder:
repeat until a quotient of zero is obtained.

For example:

```
56 /2 equals
28   r 0
14   r 0
 7   r 1
 3   r 1
 1   r 1
 0   r 1
```

As the sixth quotient is zero ,we can terminate our division process. The next step is to assemble the binary number. This is done by writing out the remainders starting with the last one found, which gives 111000. You can double check that 111000 is the binary equivalent of 56 by doing the following:

```
1x2^5 + 1x2^4 + 1x2^3 + 0x2^2 + 0x2^1 + 0x2^0
  32  +   16  +   8   +   0   +   0   +  0  = 56
```

The TRS 80 uses a 16-bit word for storing integer numbers, so we will add enough zeros on the left to give a 16-bit number. This yields 0000000000111000.

Now that we have the binary equivalent of 56 , how does this relate to the line of a basic program saying:

```
10 LET X = 56
```

If we look in memory for the place that X has been stored we find two bytes with the values 56 and 0. How does this compare with our computed binary number? Well, a 16-bit number must be broken into two 8-bit bytes. Doing this with the 16-bit representation of 56 gives 00000000 and 00111000. The byte on the left is called the high-order byte and that on the right is the low order byte.

The TRS 80 uses a CPU called a Z80. As a result the TRS 80 stores the high order byte in the second byte of the word and the low order byte in the first byte of the word. Converting the high order byte to base ten gives 0 and the low order byte gives 56.

Now we've got to the difficult bit. If you read the documentation for the TRS 80 you will notice a restriction on some numbers. A 16-bit word can only represent the range from 0 to 65535. This is the reason that the highest line number is 65525 in Microsoft basic on the TRS 80. How do we get negative numbers with a 16-bit word. The system

known as two's complement can be used to to represent negative integer numbers. What this scheme does is to use half the possible 65535 values to represent positive numbers and the other half for negatives.

So how do we get -56? Recall that 56 was equal to 00000000 00111000. To compute -56, first calculate the one's complement by switching all ones to zeros and all zeros to ones. This gives us 11111111 11000111. To the one's complement add one to give the two's complement. In the case of -56, this yields:

```
    11111111 11000111
+                   1
---------------------
    11111111 11001000
```

If this addition had resulted in a 17th digit for the result the extra 17th bit would have been discarded. Now converting these two bytes to denary gives 255 for the high order byte and 200 for the low order byte. This would be stored in memory as 200 255.

We now need to consider how the TRS-80 handles floating point numbers or numbers consisting of a decimal part and an integer part. To answer that question requires us to analyse the binary number system futher. In binary:

0.1 = 1/2 0.01 = 1/4 0.001 = 1/8 ....

To convert fractions to binary we multiply the decimal fraction by two. If the result is greater than zero, write down the number or else write down a zero.

Continue multiplying only the fractional part of the product until you have accumulated the required number of bits. Let us consider the case of 56.1625, the number 56 gave us 6 bits but a floating point number requires 24 bits for the number and 8 bits for the exponent. Taking the number part we require an additional 18 bits to give the 24 bits required by Microsoft BASIC. So we proceed as follows:

```
0.1625
0.325
0.65
1.3  x 2 gives 2.6 discard the 2 and write
0.6
1.2  x 2 gives 2.4 discard the 2 and write
0.4
0.8
1.6  x 2 gives 3.2 discard 2 and write
1.2  x 2 gives 2.4 discard 2 and write
0.4
0.8
1.6  x 2 gives 3.2 discard 2 and write
1.2  x 2 gives 2.4 discard 2 and write
0.4
```

Now we have 18 additional bits in 56.1625 = 111000.001010011001100110. Now normalise the number by moving the binary point to the leftmost position and put the number into scientific notation (the binary point in the binary system is similar to the decimal point in the decimal system). The result is a binary point followed by 111 i.e .111000001 and so on. We had to move the binary point six places to the left to get this result and this is equivalent to multiplying the 24-bit binary number by 2^6.

We next take these 24 bits and split them into three 8-bit bytes. This yields 11100000 10100110 01100110, in the order most significant byte, next significant byte and least significant byte.

Now take the power of two, in this case 6, and add it to 128. We get 134. For a positive value change the leftmost bit of the most significant byte (the bit closest to the binary point) from a one to a zero. For a negative value leave it as it is. This is the system adopted by Microsoft for the representation of floating point numbers. It is by no means universal. In the case of our number, 56.1625 we have 01100000 for the exponent.

Next convert the three significant bytes to base 10. We get 96, 166 and 102. The number 56.1625 is stored in the RML 380Z as:

```
1st byte ( least significant)  = 102
2nd byte ( next significant )  = 166
3rd byte ( most significant )  =  96
4th byte ( exponent )          = 134
```

We have now got to the point where we can consider why, when the following program is run, do we get an odd result.

```
100 LET A = 20.01
110 LET B = 20
120 PRINT A - B
130 END
```

This program gives a result of 0.0100002 instead of 0.01

To see why, convert both numbers to binary:

```
20 = 10100
.01 = .0000001010001111010
```

In scientific notation:

```
20.01 = .10100000 00010100 01111010 x 2^5
20    = .10100000 00000000 00000000 x 2^5
```

If we subtract the two we obtain:

```
0.01 = .00000000 00010100 01111010 x 2^5
```

This result is normalised resulting in:

```
0.01 = .10100011 11010000 00000000 x 2^-6
```

(the following zeros are added to keep 24 bits of accuracy)

Converting our result back to base ten gives:
(1/2 +1/8 +1/128 +1/256 +1/512 +1/1024 +1/4096)x 2^-6
which all equals 0.0100002.

A further example is the number 0.1. The fraction 1/10 also cannot be held exactly in the computer.

We know that if we multiply 0.1 by 1000 the answer is 100, this is the same as adding 0.1 to itself a thousand times. If we run the following two programs on the RML 380Z we get the following printouts:

```
10 LET A = 1/10
20 PRINT "THE RESULT IS "; A * 1000

THE RESULT IS 100


10 LET A = 1/10
20 LET C = 0
30 FOR I = 1 TO 1000
40 LET S = S + T
50 NEXT I
60 PRINT " THE ANSWER IS "; S

THE ANSWER IS 99.9991
```

Why the two different answers. When BASIC multiplies 0.1 by 1000, BASIC's automatic rounding routine took care of the slight residual error and PRINTed the correct result. But when we added in the slightly incorrect value of A a thousand fold; automatic rounding was insufficient to compensate.

The error introduced into these calculations plagues computers that use scientific notation (the majority of mini and micro computers). One way of avoiding the problem is to use the binary-coded-decimal arithmetic scheme where each decimal digit is represented by four bits and the computer then calculates in much the same manner as one would use a paper and pencil.this technique is fine for buisiness problems, but it uses more memory and is slower. It is almost the obligatory method of carrying out financial calculations.

N.B. This computer printout uses 2^3 to represent two raised to the power three.

Ken Robinson,
65 School Drive,
Newton Longville,
Milton Keynes, MK17 ODD.

Passing Parameters to code from Dos

(Graphics on FED (File Editing Device))

This was supposed to be a simple modification to the program FED but as I was writing it, it came to me to give a description of how this modification works.

This modification allows FED to display graphic characters on the screen. It can be turned on by giving the command FED G. The G will turn on the Graphic mode, if the G is omitted then FED will operate as normal (Not allowing Graphic codes to be displayed)

There now follows a description on how this modification works:

When you type in a command from Dos Ready the Dos will put it into a buffer. If the first name is a program name then the Dos will proceed to load and execute the specified user program.
When the program is entered then the HL register pair are pointing to the character after the program name. This is a useful feature for passing information over into the program.
As I did not always want this modification to be pre sent all of the time I decided to use this method to toggle the graphic option on by passing the parameter G

The FED program first of all checks for a carriage return (Typed FED<Enter>), and if it finds it then it will pass control over to FED.
If it finds the single letter G (Note: all text in the buffer has been converted into Upper case by the Dos) it will then place the modification into memory (Overlapping the FED code) to find out how this works refer to the source code. (Lines: 25 to 48). But if it is not the single letter G then the Graphic option is not required. (Lines: 6 to 23)
After initialising the Graphic option it will then reset the HL register pair to point to the byte after the toggle (G) and passes control over to FED, this is done to allow the user to pass the normal parameters to FED.

Here follows the Source code to give FED a Graphic option:

```
1                ORG   8800H      ; ORG at FED disc I/O buffer
2
3   EXEC:        EXEC  EXEC
4   SKIP:        EQU   0          ; Modifcation Skip code.
5
6                LD    A,(HL)     ; Get next character of command
7                CP    13         ; Is it FED only
8                JR    Z,EXIT     ; Yes - Goto normal FED
9                                 ; All characters al ready in Upper Case
10               CP    'G'        ; No - Is it G
11               JR    NZ,EXIT    ;    No  - Goto normal FED
12               INC   HL         ;    Yes - Move to next letter
13               LD    A,(HL)     ;          Get letter of command
14               CP    13         ;          Is it Enter
15               JR    Z,DOIT     ;          Yes - Patch FED then
16               INC   HL         ;          No  - Move to next letter
17               CP    ' '        ;               Is it a space
18               JR    Z,DOIT     ;               Yes - Patch FED then
19               CP    ','        ;               No  - Is it a comma
20               JR    Z,DOIT     ;               Yes - Patch FED then
21               DEC   HL         ;               No  - Must be a normal parameter
22               DEC   HL         ;                     Move pointers back
```

```
23 EXIT:    JP   5E81H      ;    Goto normal FED
24
25 DOIT:    LD   (RET+1),HL  ; Save DOS Command line pointer
26          LD   HL,DATA     ; Point to the data
27          LD   DE,63E7H    ; Point to the Large screen patch
28          PUSH HL          ; Save Data pointer
29          PUSH HL          ;  & again for all 3 PLACE calls
30          CALL PLACE       ; Place Data at patch
31          POP  HL          ; Recover Data pointer
32          LD   DE,64C7H    ; Point to the Small screen patch
33          CALL PLACE       ; Place Data at patch
34          POP  HL          ; Recver Data pointer
35          LD   DE,613CH    ; Point to the Modify patch
36          CALL PLACE       ; Place Data at patch
37 RET:     LD   HL,$-$      ; Get Command pointer back
38          JR   EXIT        ; Goto normal FED
39          ; Place: HL => Patch code, DE => FED code.
40 PLACE:   LD   A,(HL)      ; Get Data value
41          OR   A           ; Is it a zero
42          JR   Z,NEXT      ; Yes - Skip byte then
43          INC  A           ; Is it End Of Data flag
44          RET  Z           ; Yes - Return to caller
45          LD   (DE),A      ; No - Patch the code
46 NEXT:    INC  HL          ; Move data pointer on one
47          INC  DE          ; Move patch pointer on one
48          JR   PLACE       ; Next byte
49
50
51 DATA:    CP   0C0H        ; Replace BIT 7,A
52          JR   NC,$+2      ; Replace JR NZ,????
53          DB   SKIP,SKIP   ; Skip next 5 bytes
54          DB   SKIP,SKIP   ; CP ' ', JR NC,????
55          DB   SKIP        ; LD A, or LD B,
56          DB   7FH         ; Replace 88H (Nul charter)
57          DB   255         ; The End Of Data flag
58
59          END
```

This is the Zap to give FED a Graphic option:
For information on the Zap please refer to the accompanying text (Above)
Note: The code from byte 15 should replace the 02 02 81 5E on your copy.

FED/CMD Drive 0 Record 53 X'0035'

```
0123456789ABCDEF  BYTE  00 01 02 03 04 05 06 07   08 09 0A 0B 0C 0D 0E 0F
================================================================================
................  <00>  00 00 00 00 00 00 00 00   00 00 00 00 00 00 00 00
......W..~..(..G  <10>  00 00 00 00 00 01 57 00   88 7E FE 0D 28 15 FE 47
 .#~..(.#.  (..,(  <20>  20 11 23 7E FE 0D 28 0E   23 FE 20 28 09 FE 2C 28
.++..^":.!J...c.  <30>  05 2B 2B C3 81 5E 22 3A   88 21 4A 88 11 E7 63 E5
..)....d.>...(a.  <40>  E5 CD 3E 88 E1 11 C7 64   CD 3E 88 E1 11 3C 61 CD
>.!....~.(.(.=.#  <50>  3E 88 21 00 00 18 DC 7E   B7 28 04 3C C8 3D 12 23
.....0..........  <60>  13 18 F4 FE C0 30 00 00   00 00 00 00 7F FF 02 02
................  <70>  00 88 80 80 80 80 80 80   80 80 80 80 80 80 80 80
................  <80>  9C 8C 8C 8C 8C 8C 9C 8C   8C 8C 8C 8C 8C 8C 8C AC
.Graphic.option.  <90>  95 47 72 61 70 68 69 63   80 6F 70 74 69 6F 6E AA
..Conception.&..  <A0>  95 80 43 6F 6E 63 65 70   74 69 6F 6E 80 26 80 AA
..Implemented...  <B0>  95 80 49 6D 70 6C 65 6D   65 6E 74 65 64 80 80 AA
......By........  <C0>  95 80 80 80 80 80 42 79   80 80 80 80 80 80 80 AA
.Peter.J.Knaggs.  <D0>  95 50 65 74 65 72 80 4A   80 4B 6E 61 67 67 73 AA
..(C).23/09/84..  <E0>  95 80 28 43 29 80 32 33   2F 30 39 2F 38 34 80 AA
................  <F0>  83 83 83 83 83 83 83 83   83 83 83 83 83 83 83 83
```

Peter Knaggs.

MODEL-I/III INCOMPATABILITY and GARBAGE COLLECTOR NOTES

Firstly, I would just like to thank those members who answered some of my queries in June's NATGUG News. In particular, the information which was quoted to me from "Model-III Rom Routines Documented" explained the weird problems I was having with one of my programs. If there is anyone who has this book, or the alternative - "Model-III Rom Commented" and wants to sell it, I would like to buy it, as I now have some Model-i programs I wrote which won't run on the III!

I would like to ask those members who have used LDOS or DOSPLUS 3.5 on Model-III for their comments on each of these DOSes, as I am considering changing to one of these. DOSPLUS looks good but is a bit expensive when compared to LDOS now that Molimerx have slashed the prince. In particular, I need to be sure that existing software will work on the one I choose. Also, I don't want to get bogged-down having to read through half of a massive volume of documentation before I dare boot it up.

Now, regarding compatability, The Models I and III are not as compatable as one is lead to believe, (This also applies to model-4 when it's in model-III mode), although, at least, BASIC programs don't get converted into electronic wallpaper, as when going from I/III to 4! (To save space, I'll just use "M" for "Model" from now on).

First of all, it appears that, when manipulating the screen with machine-code routines, there is some hardware bug which can cause misplacement of charecters going to the screen in 32-char. mode, even to the point of bombing part of the Operating System and crashing the whole machine as well! Even if you are only reading from the screen, what you get back isn't what you expected to find! My answer to this one was do the screen-zapping in 64-char. mode and then PRINT CHR$(23) afterwards to get the correct display, and when reading back from the screen, PRINT CHR$(28) first. This obviously imposes some limitations, but the mc is so fast you hardly notice it happening. Incidently, this problem does not happen on the 4 in III mode.

Another problem arises with the TAB command, again only when using 32-char. mode. I think this is because Tandy changed the address of the flag that tells it what print size is in use, and they forgot to change the video driver routine to suit. In any event, all TABs except the first on the line only move half the distance intended. Very annoying, as it means juggling about with PRINT USING to get the same effect TAB used to produce on the MI.

There are two problems which can occur with programs which have sound output. If you have transferred games from MI onto the III, you may have been suprized to find that with some of them the sound effects have gone mute, despite the fact that others work perfectly! I fould this out with the SPACEVADERS game from Molimerx. There answer was "Sound works on the MI and the Genie, but doesn't work on the III, and we don't know why!" Well, if anyone wants to know, here is the reason: On MI, port FFH controls the tape relay and the video mode as well as the audio I/O. If a program allows or 32-chr. mode and/or tape relay switching, a "mask" at 403DH which echoes the current status of relay and video (as long as these are switched by the usual ROM calls) has to be ORed

with the value going to port FFH to maintain tape and video as before. Now, on MIII and 4, the tape relay and video control use a different port, and the address of the control byte has been changed. Unfortunately, if the software uses the 403DH flag, the value which will be ORed with the sound bits will always be C3H (for TRSDOS 1.3 anyway), which results in the bits controling the sound remaining unchainged, regardless of their intended value. As a result, no sounds are produced! The solution, of course, is to delete the instruction which ORs the contents of 403DH, and the sounds will magically reappear! However, that is not the end of the story. The MIII automatically turns interrupts back on whenever anything is dispayed using the cursor, whereas MI of course it does not, which makes sound output a bit rough. If there is room in the program, you should patch in a "DI" instruction in the sound effects routine. If you are a tape user, you can type POKE 16402,201 before loading the tape. Note that the cursor stops flashing when you do this, as this operation causes an imediate return on the next interrput, which forces it to stay off. You could put the necessary instructions to do this at the start of a machine-code program, which will work for disk users, but you should reboot when you have finished with the program rather than use a "Return to DOS" option, as the disks may not operate properly with interrupts "locked-out". Incidently, some mc programs from MI which go bananas on MIII can be fixed by this POKE. It has something to do with the real-time clock, but I'm not sure what. This problem with interrupts is also responsible for crashing some programs which read tapes and display information from them as they are loading.

For those of you playing SPACEVADERS, here are the necessary patches to get sound effects to work on Model-III:

PATCH SPACEV/CMD (ADD=7171,FIND=ED583D40,CHG=F3000000)
PATCH SPACEV/CMD (ADD=7179,FIND=B3,CHG=00)
PATCH SPACEV/CMD (ADD=731D,FIND=CD33,CHG=0000)

The last one stops the misplacing of the letters when printing "G E T R E A D Y" by keeping it in 64-Char. mode.

If you read my previous article, you'll remember I promised to send in details of patches to the Misosys Disassembler so that it works with the Series-I Assembler. This I have now done, so it should appear elsewhere in this magazine. Also, you will remember my reservations about using SNAPP VI (Garbage Collector) on my model-III because of the need to use 42-track disks, which might prove unreliable. Well, I have decided not to use this program, so anyone who is using TRSDOS 1.3 can have it at the Molimerx sale price, if they are prepared to take a chance (or have got 80-track drives!). From my point of view, it wasn't a complete waste of money though, because it comes wit a useful tutorial on controlling the Microsoft garbage collector, and the disk has a number of useful patches for TRSDOS 1.3 which, as it is a modified TRSDOS system disk, can be copied off. These are useful even if you don't use the SNAPP package itself. If I get lots of requests for details, I'll send in a printout of the patch files, and a photocopy of their descriptions to be printed in the mag; SNAPP gives blanket permission for this. (Talking of patches, a fairly long list of them for TRSDOS 1.3 is available from Tandy Customer Service. these include everything from fixing spelling errors in the HELP file to making the software write-protect work like the hardware one does. There is also an attempt at fixing BASIC's flaky startup routines, which is only partially successful). Also on the SNAPP disk is a utility which allows you to kill selected TRSDOS modules

to gain more space, and the manual details the functions of each of the 14 modules used. I also noticed that the disk has a BASIC program called "GENERATE/BAS" on it, and there is also a file called "BOB/MAP". I tried running the program but couldn't make too much sense of it, and when I looked at the directory I found I had created an empty file called "ROY/MAP". I assume this is some kind of program generator - has anybody got any idea if it has any uses, and can it be used independantly of the SNAPP BASIC Extentions?

As I am not going to use SNAPP VI, I have now purchased Prosoft's Trashman. I have tried this out and it really is amazing! Unfortunately, when I ran the TMTIMER program I forgot to DO the "HERZ50" patch so my timings are all wrong, but in percentage terms the savings were as follows:-

Strings	Saving
100	26.32%
250	82.45%
500	96.50%
750	97.52%
1000	97.89%
1500	98.60%
2000	98.88%

The more strings you have, the more it likes it! Of course, this is only the saving in garbage collection time, your programs don't all run 98% faster! (shame, shame!). The only problem is that attempts to automate the use of Trashman by way of a DO file could run into trouble because DO on TRSDOS 1.3 insists on using (and changing) HIMEM which means that it might not work properly, and even if it does, I won't know where to call Trashman when in BASIC. A possible solution might be to have a DO file containing the following:-

```
TM
BASIC -F:3
X=PEEK(&H4411)+PEEK(&H4412)*256+1:IF X)32767 THEN X=X-65536
DEFUSR9=X:CLS:X=USR(n):IF X()0 THEN PRINT"ERROR!":END
RUN"FILENAME/BAS"
```

Where "TM" calls Trashman and "n" is the number of bytes required (see documentation). This works on the premise that, imediately after activating Trashman, HIMEM is always 1 byte below the "entry point" as displayed when TM is entered. (This will NOT be true once TM has been called from BASIC). The section from "CLS" could be part of the program instead of being in the DO file, but then it would have to test if USR9 has been set to avoid a crash if the program is called manually without Trashman. Any suggestions?

While on the subject of the dreaded garbage collection, some members may be interested in a couple of techniques I have used to reduce it's occurance without the use of an enhancement program like Trashman. These methods apply equally to MI and MIII users. Firstly, here is one for those programs that use INKEY$ in a loop, such as an editor or a screen-drawer. Emagine that you have a program segment such as:

```
100 K$=INKEY$:IF K$()"" THEN 120
110 ....code to do if no key pressed.
120 IF K$=CHR$(13) THEN.....and so on.
```

The problem with this is, of course, that lot's of little one-charecter strings gobble up the string space until it run's out, and we all know what happens then! Now, my replacement would look like this:

```
100 K=PEEK(&H4099):IF K)0 THEN 120
110 ....code to do if no key pressed.
120 POKE &H4099,0:IF K=13 THEN.....and so on.
```

This works because location &H4099 keeps track of the last key pressed, even if INKEY$ isn't being used, but as we are assigning a numeric value, no string collection takes place. If you leave out the POKE, the program will keep responding to the last key pressed until you press something else, which might be useful on some occasions. It may be necessary to POKE &H4099,0 before the routine that uses this is called, to ensure that unwanted key-strokes don't get through.

My second technique is useful in those routines where the following situation arises:

```
100 FOR I=1 TO 2000:INPUT#1,A(I),T$,B(I),C(I):NEXT
```

Here, we are stuffing arrays with numerical values, but in the process we are having to read past a string we don't want, with obvious consequences.

Now, it is possible to arrange things so that the previous string to be assigned is overwritten by the next one, instead of just being added at the bottom of the string space. Try my test program: (Leave the apostraphe's in!)

```
10 CLEAR 2000:DIM A$(200)
15 FOR I=1 TO 100:A$(I)=STRING$(20,I+32):NEXT
20 GOSUB 60
25 'A=PEEK(&H40D6):B=PEEK(&H40D7)
30 INPUT T$
40 'POKE &H40D6,A:POKE &H40D7,B
50 GOTO 20
60 ? PEEK(&H40D6)+PEEK(&H40D7)*256:RETURN
99 FOR I=1 TO 100:? A$(I):NEXT:GOTO 20
```

This prints the current address of "Next Available String" (NAS), and then prompts for a string input. Type in any ten-Char. string and watch the numbers. As you would expect, NAS reduces until the collector pulls it back up. Now, take out the apostraphes in lines 25 and 40, and run again. You will notice that now, NAS remains unchainged. (You may BREAK and GOTO 99 to prove that the contents of the array arn't being scrambled). Now, answer the prompt with ten letter "F"'s and BREAK. Some manual string assignments at this point will show what happens to T$: After Z$="ZZZ",'T$ becomes FFFFFFFZZZ. If we now do H$="HHHH" we get FFFHHHHZZZ. In other words, normal string assignments overwrite the last T$ assignment. We could do T$=T$ to keep it, or T$="" if we don't care: I suggest one or other of these is done to avoid problems. In any event, we keep T$ from filling-up the string space and keep the collector at bay! In a "real-world" application, there are two things to bear in mind: Firstly, execute a FRE("") just before the section which uses this technique, to get rid of any garbage which may already be present, and secondly, make sure that the result from FRE("") is at least 256, to leave room for the longest string that might be read in. Oh, and remember to make sure the variables you use to save NAS arn't used elseware within the routine, or you'll confuse the daylights out of the machine!

I developed this method when expanding the Tandy Budget Management software on a 48K Genie system. I had installed a routine to use short leaders on tape files and to keep the tape running continiously to stop the relay clicking on and off like a telephone ex-

change, only to find that when running the cumulative reports module, it kept missing records from the tape. The garbage collector turned out to be one of the culprits. Using the above method, in conjunction with a fix for the other causes, the problem was solved.

Now then, will those of you with a Model-III (or 4), please sit infront of your computer, boot TRSDOS 1.3 and try this:

LOAD 123 (enter)

"That's illegal!" you'll say! Yes, but try it anyway! What's this, no error message? In fact, the only error message you'll get from LOAD is ERROR 10 (Required Command Perameter Not Found) if you use illegal charecters in the filespec, otherwise, you can use illegal filenames, nonexistant drive numbers, invalid passwords nonexistant files or non-machine-language files, and no error report is given, despite the operation being aborted. What is worse is that no I/O errors that might occur will be reported. Tandy, amazingly, say that no-one has ever reported this before and that they are "looking into it". Has anyone found out how to fix it?

Finally, a patch to remove the dotted-line under "TRSDOS Ready":-

PATCH *1 (ADD=4E75,FIND=063E,CHG=180C)

You'll appreciate this more if you use DUAL a lot, as it get's rid of the lines of dots which preceed each command on the printout.

Roy Sainsbury, Park Farm Nurseries, Landford, Salisbury.
Telephone: Romsey (0794) 390319 after 6pm.

P.S. - After printing this article, I realised that the DO file for calling Trashman won't work because BASIC scans the keyboard after every statement so input from DO will be missed and you'll probably get a SYNTAX ERROR. To solve this, either put the setup statements into a program and run it from the DO file, or have the DO file write the program in imediate-mode with RUN as the last command.

P.P.S. - I would like to exchange programs with the library. The only snag is that you seem to have dropped the Model-III library. Has it been merged with the Model-I library? Any programs I may contribute I would try to make compatable with both machines, but there is the small point of incompatable disk formats (unless you have LDOS of course, which as yet I don't).

(* It is not so much that we have dropped the Model III library, as the library dropping us. The Clays, who used to run the library, got an IBM PC, and we couldn't get anyone else to take it on. LFH. *)

PATCHES TO MISOSYS DISASSEMBLER DISK VERSION 2.4

Those of you who bought the Misosys Disassembler (Disk version 2.4), thinking it would be compatable with the Tandy Series-I Editor/Assembler will know that in fact it is not, except by using the rather unsatisfactory method of sending the output to tape and then to use the TPSRC utility to make a disk file out of it. I was particularly annoyed because I already had the tape version!

However, PATCH to the rescue! Below is a list of 6 patches which can be made to version 2.4. Patch 1 is the one which makes it's files compatable with the Series-I Editor/Assembler. The others correct a number of minor buss which are not critical to the operation of the disassembler, but improves its operation. For the purposes of these notes, it is assumed that the '32K' version will be used, that is, the version which loads from address 9FFFH. If you want to use the "16k" or "48k" versions, you'll need to subtract or add respectivly 4000H to each address given (except those in ROM of course!).

NO.	ADDR.	CONTENTS	CHANGE TO
1.	ABD2	CD 68 AA	00 00 00
2.	AE09	0D	03
3.	AA48	50 C3	00 00
4.	A00F	03	0D
5.	AA73	F2	56
6.	AA00	CD 2B 00 FE 50	3A 01 38 FE 01

The functions of these changes are as follows:

1. Make files compatable with Series-I Editor/Aaaembler. When outputing to disk, use the "D" option. This patch works by stopping the disassembler sending the file header used by tape files (and obviously some assemblers), which is rejected by Series-I.

2. Prevent scrolling after the "Depress (CLEAR)" message so that the top line of output will not be lost if this message has to be printed on the last line on the screen during "screen" mode.

3. Increase time delay between tape files to maximum. It is still not equal to five seconds, as indicated in the documentation, but is the best that can be done.

4. Ensure that the first "Writing BLOCKX to cassette" message begins on a new line instead of being appended to previous text (This applies to the MODEL-I only).

5. Correct file headers written to cassette so that they read "BLOCKA, BLOCKB..."etc, instead of "MOSISY".

6. Corrects the unreliability of "SHIFT-@" and "CLEAR" during review mode, caused by double-scanning of the keyboard. When this patch is applied, use "@" WITHOUT shift to pause. As a bonus, it allows single-stepping down the listing if "@" is pressed successively. Pressing any other key (except CLEAR) restarts normal scrolling. (On Model-III, the auto keyboard

re- peat causes a "slow-scroll" effect if you hold "@" down).

If you are using Model-I you'll need to use the program on TRSDOS 2.3B in order to load it's output into Tandy's assembler. Although it has a patch command, I have found that it often renders programs unloadable, so it is better to ignore PATCH and to modify the program in memory using DEBUG. (This cannot be done if you use the "15K" version - but I bet now I've said that, someone will prove me wrong!

Roy Sainsbury

Spooling DOTPRINT printer output to disk (part 2)

Onward then with the story. We left part one with DOTPRINT's printing going to a disk-file. The files will always be called "PR2DSK/FIL", although they can obviously be renamed. All that remains is to do the actual printing from the disk-file. To be more useful, this part prompts for a filespec.

One of the nice-to-haves that Tony mentioned, was that it should be possible to call the program from within BASIC (eg DOTPRINT) on some occasions, and directly from DOS READY on others. To make this workable, I have origined the code in high memory. When run, the resident part is protected by setting HIMEM just below it. BASIC (or indeed SCRIPSIT - see later) will in turn respect this HIMEM and not clobber our code. The odd value of origin is chosen to make the actual entry point an even FE00H, which is more easily remembered. The need to detect the end of data does not arise when the file is being read. All that is necessary is to detect either an end-of-file or past end-of-file condition.

```
 1 ;**** DSK2PR/SOF ****
 2 ;**** 07/27/84 ****
 3
 4           ORG  0FDB7H
 5
 6 HIMEM:    EQU  4049H          ;POSITION FOR MODEL I
 7
 8
 9 START:    EQU  $
10           EXEC START
11
12           LD   HL,MSG1        ;POINT TO PROMPT TEXT
13           CALL 4467H          ;DOS CALL TO DISPLAY IT
14
15 LOOP1:   EQU  $
16           CALL 0049H          ;ROM CALL - WAIT FOR A KEY
17           CP   0DH            ;WAS IT ENTER ?
18           JR   Z,PROG         ;YES - RUN THE PROGRAM
19           CP   01H            ;WAS IT BREAK ?
20           JR   NZ,LOOP1       ;NO - NOTHING ELSE ALLOWED
21
22           LD   HL,PROG        ;START OF CODE TO PROTECT
23           DEC  HL             ;SAVE ONE EXTRA BYTE
24           LD   (HIMEM),HL     ;SET HIMEM
25           JP   402DH          ;AND BACK TO DOS
26
27 MSG1:    DB   28,31,'Press <BREAK> to load only ,'
28           DB   ' <ENTER> to run',13
29
```

```
30
31 ;***********************************
32 ; THIS IS THE ACTUAL PRINTING CODE
33 ;***********************************
34
35 PROG:     EQU  $
36          LD   HL,MSG2        ;"DUMPING ..." MESSAGE
37          CALL 4467H          ;DISPLAY IT
38
39          LD   HL,KBUFF       ;LINE INPUT BUFFER
40          LD   B,24           ;NR OF CHARS TO ACCEPT
41          CALL 0040H          ;ROM CALL - LINE INPUT
42          JR   C,QUIT         ;BREAK WAS HIT
43
44          LD   A,B            ;GET NR OF CHARACTERS ENTERED
45          OR   A              ;SET FLAGS (WAS IT ZERO ?)
46          JR   Z,PROG         ;YES - START AGAIN
47
48          LD   HL,KBUFF       ;RE-POINT TO KEYBOARD BUFFER
49          LD   DE,FCB         ;FILESPEC TO GO HERE
50          CALL 441CH          ;DOS CALL TO EXTRACT FILESPEC
51
52          LD   HL,SBUFF       ;BUFFER FOR SECTORS
53          LD   DE,FCB
54          LD   B,1            ;RECORD LENGTH OF 1
55          CALL 4424H          ;OPEN EXISTING FILE
56          JR   NZ,ERROR2      ;ERROR FOUND
57
58
59 LOOP2:   EQU  $
60          LD   DE,FCB
61          CALL 0013H          ;READ A BYTE FROM FILE
62          JR   NZ,ERROR1      ;THERE WAS AN ERROR
63          PUSH AF             ;SAVE THE CHAR
64
65 LOOP3:   EQU  $
66          LD   A,(37E8H)      ;"PEEK" THE PRINTER STATUS
67          AND  0F0H           ;LOSE LOW STATUS BITS
68          CP   30H            ;PRINTER READY ?
69          JR   NZ,LOOP3       ;NO
70                                        .
71          POP  AF
72          LD   (37E8H),A      ;"POKE" CHARACTER TO PRINTER
73          JR   LOOP2          ;KEEP GETTING CHARS TO PRINT
74
75
76 ERROR1:  EQU  $
77          CP   1CH            ;IS IT END OF FILE ?
78          JR   Z,DONE         ;THIS "ERROR" IS PERMITTED
79          CP   1DH            ;IS IT PAST END ?
80          JR   Z,DONE         ;AND THIS ONE IS TOO
81
82 ERROR2:  EQU  $
83          OR   0C0H           ;RETURN AFTER MESSAGE
84          CALL 4409H          ;SOME OTHER ERROR MESSAGE
85          RET                 ;BACK TO WHERE WE CAME FROM
86
87
88 DONE:    EQU  $
89          LD   HL,MSG3        ;"DUMP COMPLETE ..."
90          CALL 4467H          ;DISPLAY THE MESSAGE
91 QUIT:    EQU  $
92          RET                 ;BACK AGAIN
93
94
```

```
 95 KBUFF:     DS   25            ;BUFFER TO KEEP THE KEYSTROKES
 96 SBUFF:     DS   256           ;BUFFER FOR THE SECTOR READS
 97 FCB:       DS   32
 98
 99
100 MSG2:      EQU  $
101            DB   28,31,'Dumping disk data to printer',10,10
102            DB   '<BREAK> to quit , <ENTER> to restart',10
103            DB   'Filespec to print > ',3
104
105 MSG3:      EQU  $
106            DB   'Disk dump complete',13
107
108            END
```

Exec Addr FDB7

You will notice that all exits from the program are through RETs, so that you should get back whence you came !! I wasn't too careful about saving registers, as BASIC's USR call does this for you. All that's needed is a DEFUSR=&HFE00 and a matching I=USR(0) The very simple printer driver (for a Model I) will pass all characters from 0 to 255 inclusive.

I think the biggest surprise of all was what could be achieved with Geof Smith's SCRIPSIT-PLUS. There was a little bit of fiddling about with the intention of printing DOTPRINT "printed to file" headers from within SCRIPSIT, to be followed by ordinary printing. Using one of the features of Geof's special version, it was possible to invoke the above program and then return safely to SCRIPSIT. Tony has the details (I think) of how this was done. Although the programs may be missing the last few percent of information to be of immediate use, I hope that aspiring machine code programmers may find a few bits that can be understood. The disk file handling is all pretty standard stuff, and could well be used in other applications. Please feel free to get in touch if there are queries.

Don Bannister Tel: 0232- 668899

FOR SALE AND WANTED

* FOR SALE *

I am now upgrading to a QL, and have the following for sale:-

TRS-80 Model I Level 2
Tandy expansion interface
Single Teac drive
LDOS, tape EDTASM, some other software.

Offers to:-

PJ Thompson,
8 Blenheim Close,
Haverhill,
Suffolk CB9 0AD.
Tel: (0440) 704614

* FOR SALE *

32K Genie I with sound, excellent condition.
Plus RS-232, printer and interface.

Offers to:-

SJ Macdonald,
10 Laurel Drive,
Northwood,
Colerne,
Wilts. SN14 8QH.
Tel: (0225) 743192

FOR SALE

Tandy TRS-80 Model III Micro-computer with integral twin 184K disk drives and green screen monitor. Exellent condition.

£100$ Software including:

TRSDOS
LDOS
NEWDOS
Fortran
Visicalc
Superscripsit
Profile III Plus
Tandy Writer
Money Manager
Numerous games

£650 ono

Telephone: 0625-26979 (after 6 pm).

— £650 —
Video Genie - Expansion box 48k - With additional 30 keys to control datawriter etc. - Twin full height Teak drives 180k each with powerpack - Single 80 track doublesided 400k with seperate powersupply - Centronics 737 printer - 12" green screen Monitor - Datawriter - Visicalc - WBW - Tandymail - Three DOS systems - 24" of Manuals etc — Donald HARTLEY 35 Dacres Road London S.E.23 2NB -01-699-5662....

FOR SALE

Tandy Model II complete with 3 disk expansion drive, daisy wheel II and matrix V printer, and also complete with covers, desk, printer stand and spares, Scripsit and spreadsheet programs, TRSDOS and CP/M operating systems.

All in very good condition.

Offers, please, to P.M. Giles on 073 081 (Fridhurst) 3583.

FOR SALE MODEL1 HARDWARE AND SOFTWARE

(All in working order)

Hardware

Model 1, 48K, Expansion Interface, RS232. Numeric keypad. Green screen monitor. Level I and II manuals. Technical reference manual. Learning level II book by D.Lien. Atari type joystick. Mini amplifier. Plus TRSDOS and LDOS

TOTAL £100

Teac twin disk drives plus cable to expansion interface.

TOTAL.....£190

Tape-based Software

Scripsit
Tape-mailing list
In-memory information
Level II Basic Instruction tapes (2)
Cop 16 copying utility
Casino Games pack
Micromovie
Meteor mission (joystick/sound)
Blackjack and backgammon
Alien Defence (joystick)

TOTAL.....£40

Disk-based Software

EDAS Editor/Assembler	£35	(RRP £53.48 Molimerx)
Superutility plus	£40	(RRP £65.55 Molimerx)
Inside super-utility + series 3 (book)	£5	(RRP £13 Molimerx)
Macroman monitor	£15	
Fortran (ex Tandy)	£20	
Sorceror of Siva game	£8	
Cyborg	£8	(RRP £17.25 Molimerx)
Ultra Trek	£5	(RRP £14.95 Molimerx)
Fast Basic book + disk pub by Wiley	£10	(RRP disk alone £15.35)
Basic faster and better book by Rosenfelder + disk	£15	(RRP £52.10 Molimerx)
Blink (passes control between Basic programs	£5	

TOTAL.....£140

(including numerous misc. games et al)
OR will split at above prices.

Publications - most as new

	RRP (where known)	Sale
Microsoft Basic Decoded by Farvour	£19.90	£5
The Custom TRS-80 by Kitsz	£19.90	£5
TRS-80 Disk & other mysteries by Pennington	£15.52	£5
Programming in Style by Dwyer	£7.99	£3
TRS-80 Basic (Wiley self teach)	£6+	£3
More TRS-80 Basic (Wiley self teach)	£7+	£3
Basic computer games (Tandy 62-2004)		£2
Business Programming Applications (62-2074)	£4.79	£2
TRS-80 Assembly language programming (62-2006)		£2
Using the TRS-80 in your home (62-2069)	£6.99	£3
Intro to TRS-80 graphics by Inman		£1
32 Basic programs for TRS-80 level II		£1
Games for the TRS-80 (Virgin books)	£2.95	£1
Programming techniques for level II by W.Barden jr (62-2062)		£1
	Total	£35

OR will split at above prices.

IF THE TOTAL PACKAGE AS ABOVE IS PURCHASED, PRICE IS £450
(as against £505)

- A REAL BARGAIN FOR A FULLY WORKING SYSTEM!

CONTACT:

BOB WIGGINS
'Dilkhush'
North Bank,
Hassocks,
W.Sussex
BN6 8JG

Tel: 07918-3430
(evngs/we)

01-920-8211
(weekday)

TRS-80 MODEL 1 LEVEL2
32K EXPANSION INTERFACE
TWIN DISK DRIVES;;LINEPRINTER 7
TAPE RECORDER ;;V.D.U
RS-232 BOARD;;HI -RES GRAPHICS BOX
100+GAMES ON DISK AND CASSETTE
DOZENS OF BOOKS;;BUSINESS PACKAGES
650 POUNDS
PHONE JIM MULVANEY ON 051-486-7030 (AFTER 4 P.M WEEKDAYS)

NATGUG NEWS

VOLUME 6 Issue 5 NOVEMBER 1984

OFFICIAL JOURNAL OF THE

National TRS-80 & Genie Users Group.

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.

The Group maintains two software libraries (Models I and II) which are free to members. Library lists are obtainable from the Secretary.

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

CONTENTS

2. Information on the Group
4. Editorial
4. Guidelines for contributors
4. Library information
5. Members' letters
7. Oggy oggy oggy! or, notes from the West Country
9. Prestel and the Nottingham Building Society
10. Breaking the systems barrier or the convolutions of data transfer
11. Using block graphics
13. Getting to know the Model 4P
18. Fun with a modem
21. Displaying menu on boot-up
24. Model 100 ROM functions
30. RAM files routine
32. For sale and wanted
33. Cumulative index - Part 2

TRS-80 Model 4
CP/M Software

Sage A/C £375 + VAT
Sage Plus £695 + VAT
Sage Pay £195 + VAT
Supercalc £150 + VAT

Reduced Prices
all include VAT & Postage

DOSPLUS 3.5	£89.95	(III/IV)
DOSPLUS 4	£99.95	(IV)
Newscript	£79.95	(I/III)
Zerloff	£59.95	(III)
LeScript	£99.95	(IV)
Pascal 80	£59.95	(I/III)
Tallymaster	£69.95	(I/III)
Elec Webster	£99.95	(IV)
Hyphenation	£29.95	(I/III/IV)
Grammatical	£29.95	(I/III)

Secondhand Software
all inclusive prices

Data Writer	£50.00	(III)
Scripsit	£25.00	(I)
Visicalc	£30.00	(I)
Renassen	£15.00	(I)
DOSPLUS 3.4	£30.00	(I, S/D and D/D)
NEWDOS 80/2	£25.00	(I)
NEWDOS 80/2	£25.00	(III)
Tandywriter	£15.00	(III)
Tandymail	£10.00	(I)
Modem 80	£20.00	(I/III)
Inventory	£10.00	(I/III)
Mfg Inventory	£10.00	(I/III)

Wordsworth Computers Ltd
36 High Street
Banchory
AB3 3SR
(03302) 2563 or 4168 (after hours)

EDITORIAL

I have had to hold a couple of lengthy contributions from Roy Barber and Laurie Shields over until the next issue, because I wanted to include the second part of the cumulative index.

We will probably be going bi-monthly, starting with the next issue, as it is getting difficult sometimes filling up a 36 page issue, and we feel that a large issue every two months is better than a flimsy issue every month. It will also reduce costs considerably, which is important in view of the falling membership.

Although the QLers all seemed to enjoy the Swindon workshop immensely, I have heard that some NATGUG members were less than ecstatic. It is getting extremely difficult now to put together an interesting programme for TRS-80 users, because we are running out of topics. In a way we are a victim of our own success, having done it all, in a manner of speaking. I did think of getting someone from the CP/M Users Group (such as Derek Forgrad, their software Librarian and Membership Secretary) along to the next workshop, now that many members have CP/M on their Model 4s, but Brian tells me that CP/M talks have not been very popular in the past. If anyone has any bright ideas on how to re-vitalise the workshop programs please get in touch.

80-NET, the Groups Computerised Bulletin Board system is now in the tender hands of Mac Randle. It is operational most evenings from 23.00 to midnight, and the number is (0926) 39871.

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 hand-written material, write legibly.
6. Do not fold the sheets when posting them. Use a large envelope and keep them flat.

Leon Heller

LIBRARY INFORMATION

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

MEMBERS' LETTERS

Many thanks for publishing my query in the September NATGUG news; unfortunately no-one has replied as yet, but I'm still hoping. I'm writing to ask how long is left on my subscription, and to make a modest proposal;

My interests include science fiction and wargaming, and I subscribe to a number of magazines, fanzines, and journals. Most of these publications put subscription information on the envelope, below or beside the address. This is usually presented in one of two ways, either by saying how many more issues are left on the subscription;

"Issues Left: 2"

or by giving the number of the last issue to be received on the subscription;

"Last Issue: No. 29"

The latter form has the advantage of only needing to be updated when a subscription is renewed, and is more suitable for use with a word processor or a dedicated mailing list program. However, a data base could easily handle either form. Please would you consider adding this information to the mailing labels on future issues.

Are Kansas City Software still producing TRS-80 programs, or are they just selling old stock? The last time I saw their stand (at a computer exhibition at the Barbican last year), all the TRS-80 programs were at least a year old, and their main thrust seemed to be into BBC software. I've not seen any of their advertising in a long time. If Kansas are dropping out of the TRS-80 market, or have gone out of business, maybe some kind soul will relent and tell us how to copy their tapes onto disk. I'm NOT interested in piracy; however, I have some good Kansas games which I would like to play more often, but tend to ignore in favour of disks. Maybe this problem has been solved by some NATGUG members; if so, please could someone tell me what to do! Most TRS-80 software manufacturers accept that copyright lies in the program, not the media, and will allow personal backups and transfers from tape to disk. Some will even transfer software for customers, for a fee. Most pundits say that the TRS-80 is obsolescent, and represents a dwindling market. Isn't it time that Kansas, and other manufacturers with similar policies, did something nice for the people who set them up in business?

Finally, another silly problem arising from my conversion from 35-track to 40-track drives. The FREE command on DOSPLUS gives a track allocation chart for a specified drive, which I sometimes find extremely useful. However, while my DOSPLUS disk is 40-track, and reads itself as 40-track, FREE persists in mapping drive 1 disks as 35-track. This isn't causing any loading problems, but is annoying. I suspect that there is a zap for this, but can't seem to find it.

With many thanks, Marcus L. Rowland, 22, Westbourne Park Villas, London W2 5EA, England

If a Model 4 in III mode is any help in keeping up the Model III library, I'd be glad to help, unless all the programs have gone with the wind. I still have the Model I and can link the two via the RS232, which may be of use, and I also have a modem (which so far I've not succeeded in linking to anyone else - but haven't tried very hard due to pressure of other things!)

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

Many congratulations on passing the R.A.E. and getting your class B licence. My XYL and I have had our licences for about 4 years, she is Pat, G4PDA and I am Bas, G4PAY. We really aquired our TRS-80 Model 1 Level 11 16K in early 1981 to aid us in learning Morse, in which role it was invaluable and as a result we had only a year as B licencees. We obtained our G4's in May 1982 thanks to TRS-80, and at that time we envisaged using the TRS-80 for RTTY and CW and later fo AMTOR. The big snag is that it has proved impossible to overcome the interferance which the machine throws out right acoss the HF bands and across 2 Metres as well!

We have external antennae on a mast at about 35 feet above ground level, there is an HQ1 for HF with a crossed 5 element Yasi for 2 Metres on a rotator, and a W3DZZ trap dipole erected as an inverted V. Maximum distance of the antennae from the TRS-80 is 15 feet. The tranceivers are within 2 feet of the computer.

Before I joined the Colchester Users Group which meets at John Kilpatrick's abode I was a simple soul and not in the least aquisitive and I had found that there was no interference picked up by the various radios when the antennae were disconneted and the antennae sockets were left open.

Therefore the interference was being picked up on the external antennae at a distance of 15 feet throush brick walls and roof tiles! Simple problems call for simple solutions so I lined the case of the TRS-80 with Aluminium Foil and the problem was solved.!!!

That was fine until my aquisitive nature (or my gullibility) took over and I aquired from Arnold Handley his old printer, from Brian Pain a parellel interface, from Peter Inglis a TCS and from Tandy at a bargain price of £37.00 the high resolution graphics mod. Take it from me it is quite impossible to screen that lot, so I am seriously considering changing to another breed of machine but no one publishes any measurement to give even a hint of the amount of RF hash that these machines, disk drives, printers and modems throw out. A friend operates CW from a BBC 2 alongside the rig with no trouble but he has only cassette I/O, others operate Spectrum without problems, but the Tandy sounds like a Chinese Laundry on overtime!

In the Wireless Telegraphy Act which came to the Statute Book this year there are provisions for the Secretary of State to make an order requiring Manufacturers of receiving apparatus and audio systems including Organs, to ensure that

their products do not respond to out of band signals, and likewise to ensure that their products do not radiate interference outside of certain prescribed limits.

If you or any member of the groups (TRS-80 & QL) has any information on these issues which are of interest to Radio Amateurs I will be glad to hear from them, other than sussestions of operating th TRS-80 in a metal lined case/room or 200 feet from the antennae, because I would love to use mine on the air.

Incidentally if anyone can tell me how to modify this "Pencil" program (Tape) to overcome the problem of dropping a letter at the beginning of each line I will be most grateful, HELP HELP HELP !!!

Basil Caines.
10 Shears Crescent,
West Mersea,
Essex CO5 8AR.

OGGY OGGY OGGY!
or, notes from the West Country

The penalty for having too much to say is that certain people demand that you write an article - largely and mainly I suspect to save a certain gentleman from having to provide his own space fillers. Oh well, let's get on with it - don't expect anything clever, 'tis just a collection of ramblings.

First and foremost, a very great **THANK YOU** to those who organised Swindon. A really great session and one that proved that the TRS-80's aren't finished yet. There were a few famous faces missing though, and one can only hope to meet them next time. And on the subject of Swindon, shame on you all - it was left to a **Queer Lugger** to be the last to reluctantly unplug his system and go home! The weekend was planned around that popular Spanish DOS, and by the sight of all those 4's in attendance, it was a needed subject area. Good old Dotprint was in evidence too, with Dave Holman giving another of his popular talks, this time concentrating on the Letterset Manipulation Utilities although he successfully managed to avoid Wiggle! One would have thought that Dave had enough punishment at his work, but he went on to give an introduction to CP/M on the 4; both he and Laurie Shields are working on ways to transfer CP/M to TRSDOS 6.0 and vice-versa. (I shall avoid the temptation to say that the latter deserve one another).

Did you see the reference to Laurie in 80-Micro? The lad has made it to the top at last! And as a little ad for him, by the time that you read this then ZEN84 will be available for CP/M. The more sober among you might like to know that he is selling a Utility disk containing DI, Verf, and Comma-Edit for £19.50. DI is a directory sorting program for Newdos80, Verf is a file verifying utility similar to Superzap's VDS, and Comma-Edit gives you that ND80-type edit facility IN DOS LEVEL! These utilities are for either Newdos80 or TRSDOS 6 (although the latter doesn't need DI) and the Model III version incorporates a self-booting ND80 for the Model 4P. Laurie's number is 0246 75568.

Part of my recent redundancy package was the chance to buy an old two-drive terminal for £75. However, although Os couldn't find me a reasonably priced EI for a second Model I system, he could find me a nice cassette Model 4 that the drives would fit

into! So, picture the scene: it's 8.00pm and with a keen sense of anticipation our new owner has driven on a wet and windy night to collect his new machine. Calamity! On asking for a M4 DOS, it is found that the drives do not run at the usual 6ms, and worse still, despite assurances to the contrary, El Dose will NOT re-configure itself to a slower stepping rate. ZAP Enter Zapman Leo! Now if anyone wants something to marvel at, it just has to be either Leo or Laurie using Dissassembler, Scripsit, Zen , all cobbled together with a JCL file! Anyway, Leo took SYSO/SYS to bits, re-assembled it, tested it, and found the B U G. Sounds simple doesn't it? Well it was 2.15am when I left Blandford with a working DOS, and I am eternally grateful to Leo. But, I have to ask, how is it that that software was published with what turned out to be a really silly error in it, and how does Leo get the recognition that he deserves? The same problem apparently exists in 6.2, but I am assured that the same genius has fixed that too!

Another commercial, much belated, is for Mr. Matthews from Dorchester who has presented NATGUG, via the writer, with a superb collection of BYTE magazines. Thank you Sir, and ONE day I WILL have them all indexed! For the benefit of any new members, I try to maintain a magazine library and hold, in addition to BYTE, copies of all 80-Micro to date. An index is available from Commissar Pain, against which photocopies may be ordered; we don't loan out the magazines because they wouldn't stand the handling, and you wouldn't like the postal charges!

The definition of True Joy: booting up Model IV Visicalc on a 128k machine! If you hadn't seen it, would you believe that that little number up in the RH top corner (you know, the one that rapidly changes to OM) now says 90. The problem now is, have I got enough disk space to save the file! Dave Holman gave me a copy of his JCL program that puts the system into the Memdisk drive 2, and I thought that quite impressive, but Visicalc just runs as normal with just that magic 90 proving the difference.

Another commercial, this time for Os House. (Incidentally all these ads are totally unsolicited so sorry if I get anything wrong). Os attends all our workshops seemingly regardless of the commercial viability of same so it seems to me that we should all give him all the business possible. I mention this because I hear of so many people ordering from a firm in Bexhill when they could buy the same thing from Os - and at least 10% cheaper thanks to his NATGUG discount scheme. Os's number is 0258 53737, or you could always leave a message on the TBBS Blandford Board, 0258 54494.

And another! The people who are always asking me where to buy coloured ribbons for the Epson; I bought mine from a guy who once advertised in NATGUG - he trades as Workforce, of 140 Wilsden Ave, LUTON, Beds., and his phone number is 0582 454456. Last prices that I had were 2.50 black and 3.50 colours (Red,blue,green,brown) + VAT but that was last year.

Sad to read in this month's 80-Micro of the demise of CLOAD - still, and another ad for the most deserving, I still have several CLOAD tapes and disks, all brand new. Tapes sell for £1.50 and disks for just £2.50, plus p&p. I also have a few AI and Big5 tapes left, together with copies of May 83 and June 83 80-Micro (when it used to be quite thick!). These mags are £1.50 each + 70p p&p.

Dotwriter has certainly aroused a lot of interest in the Tandy together with a dot-addressable printer, so much so that Brian has thrown down the challenge: We've put up with "SARAH" ever since the earliest Model I and Lineprinter II days - you've

got until March 1st to produce a really superb high-res pin-up! Warts and pimples are allowed, but block and ASCII characters are definitely not! There will naturally be a worthy prize for the best entry, rumour has it that we are negotiating a Page 3 contract with someone called Murdoch.

That reference to March 1st is your early warning of the next NATGUG workshop; including the AGM it will be held on March 1st,2nd, and 3rd at the Wiltshire Hotel, Swindon. There is NO January Milton Keynes event due to reported lack of demand - Brian's number is 0908 564271!

Another advert that has disappeared from the American magazines is Banner Machine - has anyone ever used this, and if so, with what results? According to the ads, this is not to be confused with the old Banner program, but is a high-res printing package presumably similar to Dotwriter. I would be very interested to hear from any users out there.

Well, that's enough space filled for one month, so let me leave you with one cheery thought - it's time to send in your subscription renewals before all goes on Christmas pressies!

David Washford,6 Houston Way,Frome BA11 3EU 0373 72739

Prestel and the Nottingham Building Society

The Nottingham Building Society have provided access through Prestel to many of their services. The facilities which they offer include:-

a) A statement of your account, together with details of the previous transactions.
b) Give instructions for payments to be made to third parties.
c) If you have a mortgage with the NBS, check on the balance outstanding and the interest added.
d) Transfer sums to your account with the Bank of Scotland, or vice versa.

Obviously it is necessary for there to be passwords to ensure that only you can make payments from your account. This is done by using a 'one time pad' in that a four figure code is used for each transaction. Each code is used only once and then discarded. Repeated attempts to issue instructions using the wrong code, end up by barring any more attempts. Or if you suspect something is not right you can issue a 'stop' instruction yourself, which can only be removed by your written authority.

Payments can only be made to third parties, by prior arangement and then only to organisations such as the Electricity Board or British Telecom. (When I asked why this was I was told that it was to further prevent unauthorised access and clearly without this restriciton, the interest of hackers would increase.)

If you do not have a Prestel modem, the NBS will loan you a Tandata unit, for either a small rental or nothing provided your account is above a substantial figure. Alternatively you can buy it, but it is expensive. Technically the unit is interesting in that it has a detachable keyboard, linked by an infra red beam to the modem itself.

If you have any queries you can send messages (or vulets as the NBS call them). You can ask them to a) telephone you or

b) write to you or c) mailbox you. Replies come back very quickly by mailbox, often in aa matter of minutes, between 9 to 5 any day, including weekends.

If you want to see how they operate you should look at page 444 on Prestel and if you want further information you can ask them to send it to you (or mailbox me on 277453106).

The NBS are constantly thinking of more financial facilities to offer and they are just trying out a link with the stockbrokers, Vickers da Costa, but it looks a bit ponderous

Richard Creak.

Breaking the Systems Barrier

Or

The Convolutions of Data Transfer

Venue :- Swindon Meeting
Time :- Getting on for Midnight on Friday

There I was, sitting talking to Mike Gordon-Saker about something of absolutely no consequence at all, when along came this young chap with a beard.

" I've been told you've got a 'Prom programmer." he said. " Yes, that's right." said Mike. " Well. I've got a problem that you might be able to help me with...."

Introducing himself as Dave Beales, he went on to explain that he had a QL on which he had written an assembly language program for a Centronics Parallel printer interface. (Apparently, if you hang a Rom and a bit of extra hardware onto the QL bus, you can add a new device called 'PAR' into the system.) The problem was, no-one has got a Prom programmer for the QL.

" O.K.," said Mike, " You give me a hex dump of your code and I'll program it for you."

" Fine." said Dave and trotted off to his QL.

After a while, I went over to see how he was doing. Dave had written a small Basic routine to peek the code he had assembled and print it out as a series of decimal values, via the Ser1 channel, to a serial printer. Then came the problem.... The printer wouldn't Line-Feed.

After inserting various CHR$(9)s and 13s and God Knows What, we <u>still</u> couldn't get anything sensible. The printer would merrily churn away, busily overprinting everything. " Let's try another printer." I said. So we did. And another QL, but still no joy. " Why don't we send it direct to my system via the RS232." said Mike, who had wandered over to see what we were playing at.

There was the next problem. Mike's Prom programmer is part of his Model I system and it hasn't got an RS232 on it (It's been back to Tandy three times and still doesn't work). " O.K. Send it down to my Model 4P and I'll copy it across." he said. So we did.

We connected the QL directly to the 4P and Mike got his manual out to find out how to use the comms. Considering the fact that Mike had only had his 4P for 10 days and had never ever done any comms before, it all went fairly easily. We only had to reboot twice and try again before we managed to produce an Ascii file containing the data.

Moving over to the Model I, Mike used SuperUtility to move the file onto a Newdos 80 disk so that we could play

around with it. We then LISTed the file to the printer to have a look at it. We found that there were a few errors in it that had occured during the serial transfer. We decided that it would be best to edit the file to correct them. Mike loaded in AJEDIT, since that was the only editor he had got on that particular system disk, and we then found our next problem. By the look of the results, there weren't any Carriage-Returns in the file and AJEDIT's line buffer was overflowing every so often and losing some of the data.

" I know," I said, " we can use Basic to reformat the data."

" You're the Basic Hacker," said Mike, grinning 'Go to it."

After a quick splurge on the keyboard, I wrote a routine to read the file as a Basic Data file then write it back out with each value on a separate line. We then went back into AJEDIT to have a play around. We corrected all the errors, i.e. we read all 197 values off the QL screen and manually checked each one) then we removed the byte counts from the file. (Dave's routine had sent the count alongside the data to give us some points of reference.)

"O.K." I said," Lets do a Global Insert with DEFBs and turn the data into an assembler file." Mike looked horrified, " You can't do that in AJEDIT !" he shrieked. "Alright, I'll do the bloody job in Basic !!!" I said and proceded to do so.

Ending up with a file full of DEFBs, we decided to assemble it. Neither of us could remember how to use EDTASM so Mike got out his brand spanking new ZEN84 that he'd just bought from Laurie Shields and went at it like a madman. After chasing his tail for 10 minutes and getting nowhere he decided it might be a good idea to ask for some help. Laurie and Don Bannister happened to be floating around just then, so we conned them into assembling the data and producing a BIM file for us.

Finally, Mike then got out his Zero Electronics Eprom Programmer, connected it up and blew the data into Dave's 27is.

" Great ! Thank's a lot ! I'll buy you both a pint tomorrow." said Dave and we closed down the systems and went upstairs to bed. It was 3.45 am !

Ian (Jonno to his friends) Johnson
28 Campion House
Bowthorpe
Norwich
NR5 9BW

USING BLOCK GRAPHICS

The following Basic program is used with a machine code program to simplify and speed up the use of block graphics.The Basic program is an example,to show the printing of a sine wave:-

```
5 DEFINT I-N
10 DEFUSR0=&HD000
20 FOR I=1 TO 100
30 FOR J=0 TO 7
40 K(J)=450*SIN((J+I*8)/50)+480
50 POKE &HE000+2*J,PEEK(VARPTR(K(J)))
60 POKE &HE000+2*J+1,PEEK(VARPTR(K(J))+1)
80 NEXTJ
90 X=USR0(I)
92 LPRINT
95 NEXTI
110 END
```

The machine code to go with it in assembler is as follows:-

```
ORG 0D000H
LOAD 0D000H
START: EXEC START
TXT: EQU 0E000H
LENGTH: EQU 960
PUSH HL
PUSH IX
PUSH DE;JUST IN CASE OF BASIC INTERACTION
LD HL,LENGTH;LINE LENGTH
PUSH HL;WE NEED IT FOR OTHER THINGS
LD E,27
CALL LP
LD E,"A"
CALL LP
LD E,8;LINE SP=8 DOTS
CALL LP
LD E,27
CALL LP
LD E, "L"
CALL LP
LD E,LENGTH-768
CALL LP
LD E,3
CALL LP;SETS UP BIT IMAGE DATA
LOOP1: LD IX,TXT;START OF TEXT
LD D,128
LD E,0;RESET
LOOP: LD L,(IX+0)
LD H,(IX+1)
LD A,H
CP 0
JR NZ,AL
LD A,L
CP 0
JR Z,BETA
AL: DEC HL
LD A,E
OR D
LD E,A
BETA: LD (IX+0),L
LD (IX+1),H
SRL D
JR Z,END1
INC IX
INC IX
JR LOOP
SRL D;PUSH BIT TO right IN D
JR Z,END1;8 BYTES PROCESSED
INC IX;NEXT BYTE OF GROUP OF 8
INC IX
JR LOOP
END1: CALL LP;NOW PRINT.8 BYTES DONE
POP HL;WHERE ARE WE ALONG LINE?
DEC HL
LD A,H
CP 0
JR NZ,LOOP2
LD A,L
CP 0
JR Z,END2
LOOP2: PUSH HL
JR LOOP1
END2: LD E,13
CALL LP
```

```
POP HL;RETURN TO BASIC
POP IX
POP DE
RET
LP: LD A,(37E8H)
CP 63;PRINTER READY?
JR NZ,LP;NO
LD A,E
LD (37E8H),A;YES
RET
END
```

The program assumes an Epson type dot matrix printer,with the appropriate control codes.As it is configured here,the output appears as a solid line up the value to be displayed .The assembler used was Laurie's ZEN84,and after a little time to get used to the way it works I am very pleased with it.For anyone who is interested,but has not got the benefit of an assembler, the data that needs to be poked from &HD000 to &HD088 is given below.I have also included a sample print out.

Alan Malvern Tel: Ascot 26202

```
 229  221  229  213  33  192  3  229  30  27  205  125  208  30  65  205  125
08  30  8  205  125  208  30  27  205  125  208  30
 76  205  125  208  30  192  205  125  208  30  3  205  125  208  221  33  0  2
4  22  128  30  0  221  110  0  221  102  1  124
 254  0  32  5  125  254  0  40  4  43  123  178  95  221  117  0  221  116  1
203  58  40  16  221  35  221  35  24  220  203  58
 40  6  221  35  221  35  24  210  205  125  208  225  43  124  254  0  32  5
25  254  0  40  3  229  24  184  30  13  205  125
 208  225  221  225  209  201  58  232  55  254  63  32  249  123  50  232  55
201
```

Getting to know the Model 4P

Back in March I bought a Model 100, for the specific task of transferring data between my (home) Model I and the (office) North Star Horizon, which uses Hard Sectored Disks and is impossibly incompatible diskwise. By converting files to ASCII and sending them through the RS232 from Model I to Model 100, and then from Model 100 to Horizon, I achieved my object with some difficulty, particularly with long Basic programs. Eventually I got fed up with the Model 100 and decided we must part! My main objections were 40 column display, very slow display, Selectric style keyboard with lots of things in the wrong place, and a nasty habit of thinking it was an 8k machine instead of 24k every time I plugged anything into it. Probably as an only computer it would be very good, but it was too unlike a Model I and I spent most of my time correcting mistakes. Also, I wasn't using most of its facilities.

So, I turned it in for a Model 4P. (Part of a 4P, anyway). Financially it was pretty disastrous, as I gather I was the fourth person in a month to do the same swop! However, apart from the finance, I am delighted with it.

Nobody seems to have written anything about the 4P as distinct from the 4, so I am going to! As everybody knows from the ads, it looks rather like a sewing machine when being carried (and weighs just about as much, too). The magazine that described it as the "luggable" model 4P had a nice descriptive touch, I think. It does weigh 26 lbs, but feels more like 260 lbs after 100 yards or so. Nevertheless, it can be carried in one hand by the very nicely balanced handle (a bit sharp in places) which is in the back of the case and very firmly fixed. You certainly don't feel it is going to break. One criticism is that there are holes to let the rain in and no waterproof cover that I know of.

Inside the sewing machine is a complete computer with two disk drives, screen, and one of the nicest keyboards I've met. My particular joy is that everything is in exactly the same place as on the Model I - no more miss-hit keys because they aren't where they ought to be. The keyboard is separate, on a flexible cable about 18" long which you can't unplug: only about 1" thick, but with legs at the back so that you can tilt it if it suits you better (as it does me). Very nicely positioned keys, with a light touch that makes it a joy to use. Some people may say it is noisy as the keys do make quite a loud tap when they bottom, but I don't find it annoying.

The screen measures 7.75" x 5.25" but doesn't seem small as the definition is exceptionally good. I find it easier to read than the Model I 12" screen simply because it is so much clearer. Oddly, not all the screen width is used - the line length is only 6", but this gives a nicely framed "picture".

I had expected all manner of incompatibility problems on changing from the Model I, but they haven't materialised. I have for two years been using double density, double sided drives with NEWDOS80 v 2.0, LDOS 5.1.3, and DOSPLUS 3.5 - anything, in fact except TRSDOS 2.3! I find that most things will work in the Model III mode without trouble, provided you keep to the same DOS. (Can't say about DOSPLUS - don't have it)

The Model 4P is really two computers in one, which makes it quite difficult to describe: however, I will try and do this in two stages.

Model III mode.

The 4P is not supplied with any Model III DOS, which is rather surprising. What it does have is a disk marked MODELA/III, which contains the Model III ROM image. When you boot with this disk it takes around 20 secs "Loading ROM image - please wait". Having done so it then announces "Not a SYSTEM disk", and hangs up. If you now insert any Model III system disk and press the reset button, it will behave as though it is a Model III. Visicalc (model III version) was being given away with it when I bought mine, and this disk contains TRSDOS 1.3, so I have a DOS, but no manual for it! More to the point, NEWDOS80 v 2.0 is available by changing the /SYS files (only) to the Model III version. When you do this all the other programs on the disk work. So do all my Basic programs written in NEWDOS80, and so do nearly all of my other programs. Molimerx' Pascal works perfectly, and as far as I can see any disk written to use NEWDOS80 on the Model I will work using the Mode III version on the 4P.

Exceptions to this rule are machine language programs which use their own Keyboard, Display, and Printer routines, notably SCRIPSIT, and (as far as I can see) the EDIT, MSO, LBO, etc bundle of programs making up the Disk Editor Assembler (otherwise known as MACRO-80).

Files and programs stored under TRSDOS are unreadable to NEWDOS80, except for a special FDRIVE format which enables you to copy them to a NEWDOS80 disk and then read them. This is described in FDRIVE option TI=AM of NEWDOS80. I used this to copy Model III Visicalc over to my system disk.

For TRSDOS and LDOS users the position is much easier. If you copy the MODELA/III file on to your system disk it becomes

self-booting and you don't have to use two disks to boot. Incidentally, once you have loaded the ROM image you don't need to load it again unless you switch off or load something else over it (for example TRSDOS 6 in the model 4 mode).

One very welcome discovery is that the ROM printer driver no longer filters out Chr$(0) etc - anything you want to send to your printer you can send!

Model 4 mode.

As well as the MODELA/III disk, you get another marked TRSDOS 6.0, and this is used only to give you the Model 4 mode. In this mode the lines are 80 characters long instead of 64 and there are 24 lines per page. TRSDOS 6.0 is in fact LDOS 6.0, and from the users point of view is almost exactly the same as LDOS 5.1.x. This makes changing to a new system very painless so long as you knew LDOS' The Manual is very thick, but not set out nearly as well as the real LDOS one, and I find myself using that one rather than the one supplied. Some commands (for example those for double sided disks) are not documented in the Tandy book but are still in the DOS, and work. Passwords have been changed, which is irritating, particularly as they are not given in the users manual, but only in the Technical Manual, which is another £16.95 (and even heavier than the users handbook). I think the Technical Manual is superb - except that it doesn't mention the 4P at all!!! From the software point of view, if you are going to do any assembler programming the Technical Handbook is absolutely essential.

From the point of view of internal coding, the 4/4P are totally different from the I/III. No entry points for machine routines are given - instead you have to load a number (called the supervisory call number or SVC no) into the A register and then do an RST 28H. A very comprehensive list of SVC's is given in the Tech H'book, and without them you'll get nowhere at all. Even poking to the screen is not on any more, as a number of people have observed! The advantage of these SVC's is that programs should be fully transportable between current and future models of the 4/4P, and, if you've been using LDOS 5.1.3 or 5.1.4, with the Model I and Model III also. SVC's have been a feature of LDOS available for the models I/III for some time now. A disadvantage is that you can no longer forage around in the ROM looking for useful routines, but have to use the 104 SVC's: they are, however, very comprehensive in their scope.

Model 4 Basic is different from I/III basic, and is almost pure Microsoft Basic 80 or MBasic. Since I have been using MBasic on the Horizon for some time, I have not found the change difficult. As for the spaces between keywords, I add these almost automatically and find it difficult not to put them in. The ability to use sensible variable names (up to 40 characters long) makes it easier to write and understand Basic programs, and I think the extra time in typing a few more letters is saved later on when you want to alter something and have forgotten how the program works.

What I did miss (but not any more) are the abbreviated commands. Having to type EDIT 100 instead of E100 or just finding the line and typing a comma is a chore! However, I now have the DOSPLUS "PLUS 6.0" enhancements to Mod 4 Basic, which puts all those features back in again. It also adds an array sort which is very powerful and fast, labels (GOTO DISPLAY is

legal), and DOSPLUS' Search and Replace function which is marvellous for editing programs. Alas, I believe I bought Wordsworth's last copy! Well worth buying if you can find it.

There are a lot of very odd features about the 4P, particularly regarding the character set. ASCII 1 to 31 are the usual control characters, but only some of these can be used in the direct mode (i.e. by pressing the control key and a letter key). You can only find out which by experimenting, as the book tells you about this, but doesn't tell you which keys! You can use them from programs by using the PRINT CHR$(n) command, just as you could on the Model I. One of the most useful, return to start of line without erasing it, is missing, and to get the same result I have used PRINT STRING$(POS(0),24). 24 is a single backspace without erase.

Perhaps the oddest idea is that a Null, or CHR$(0), when sent to the screen, says that the next character sent to the screen, if an ASCII 1 to 31, is printable instead of a control code. There are special characters in these slots, mostly the foreign vowels with dots and accents over them; one good point is that one of them is the £ sign. I find this whole idea infuriating in practice; any program that sends a null to the screen is liable to produce gibberish, particularly as a null itself has a special character allocated to it. The result of this is that a string of n nulls sent to the screen, instead of printing nothing, now prints n/2 capital A's with two dots over them! Maybe I'm unusual in sending nulls to the screen, but they are very useful sometimes in producing formatted displays.

Still more quirks of the character set - CHR$(21) toggles the codes from 199-255 between space-compression codes as used on the Model I, and a "special" character set which includes all the ones produced by 1-31 and a few more. When in this mode, sending CHR$(22) to the screen toggles between the "special" set and an "alternate" set which includes all the Greek alphabet and then some. You can't have both special and alternate characters, though, CHR$(22) changes ALL the "special" characters on the screen to the "alternate" characters.

Finally, there is a reverse video mode which I haven't quite fathomed yet. If I do reverse the video the cursor (normally a flashing underline) turns into a flashing zero! Why can't they leave well alone?

Nearly all the peculiarities previously induced by the odd ROM errors seem now to have been eliminated. Routeing is now possible in all variations (previously there were some things you couldn't Route); the ability to send any control character to the printer makes life a lot easier. There seem to be one or two additions to the DOS above the previous LDOS standard, mostly fairly trivial ones. Sadly, the beautful layout of the LDOS manual has been completely messed up, although the Tandy version does have an excellent index.

It is very easy to criticise and tear holes in anything new, so I must add that despite my criticisms I am delighted with my 4P, and would strongly recommend it to anybody who wants a compact, luggable, and very versatile computer but doesn't want or need the complications of colour graphics. The 4P is a work-horse, rather than a toy, and I'm really only starting to find out about it so far.

Two final boobs - 3 function keys are provided with no apparent way of programming them, and the printer and external I/O bus edge connectors are, believe it or not, solder coated!

Postscript.

The above was written in October, and I have found out a few more oddments since then. There are bugs in the basic! Those I have found or been told about are: the WIDTH command doesn't work and causes a syntax error. OPEN"E", ... doesn't work unless the file has already been created, although the book says it will create a new file. Most annoying bug is that (some) syntax errors in Basic return you straight to TRSDOS. This can be disastrous if you are writing a program, as there is no Basic * command to get your program back. I managed to recover one large lost program by DUMPing most of memory to disk and then sorting out the wheat from the chaff, but it took an age. Moral - "save" frequently!

The good news is that TRSDOS 6.01.02 cures these problems ....the bad is that PLUS 6.0 won't work with it, so on balance you are worse off! Only the Basic appears to be changed in TRSDOS 6.01.02 as far as I can see. This has been so thoroughly shuffled that I've not found a way to modify it or PLUS 6.0 to make them compatible. Has anyone else out there?

I now have the second 64k of memory installed. Very useful with a MEMDISK on it for two reasons: first, you can instal a whole TRSDOS+Basic system on it and all the disk accesses every time you use a library command disappear. Much quicker and quieter. Second, although you are told you cannot access it from Basic, you can indirectly; by putting a data file on your Memdisk and GETting and PUTting records from this file, it acts just like a huge 64k array. If you SYSRES sys files 1,2,3,10, and 12 then getting data from the second 64k is just as fast as getting it from an array, but doesn't use up Basic's program space. Incidentally, because basic loads from 3000H up instead of from 5200H, there is more space available anyway.

Henry Shepherd.

It is with much personal sadness I have to inform members of the User group that Nigel Hyde died suddenly last week. To all of us who had the privile ge of knowing and working with him it has come as a great shock. I first met Nigel some five years ago when he was using a Model-1 and typically, even in those early days, he was not prepared to accept the limitations of the available software. He was an active member of the group confining his interests mainly to the commercial application of computers but was not unaware of their educational and leisure potential.

Nigel was also an accomplished double-bass player performing with a number of orchestras, an enthusiastic member of a local amateur dramatic society appearing recently in a pantomime as the Wizard in the "Wizard of Oz". He was a member of the Royal Institution and also the St John's Ambulance.

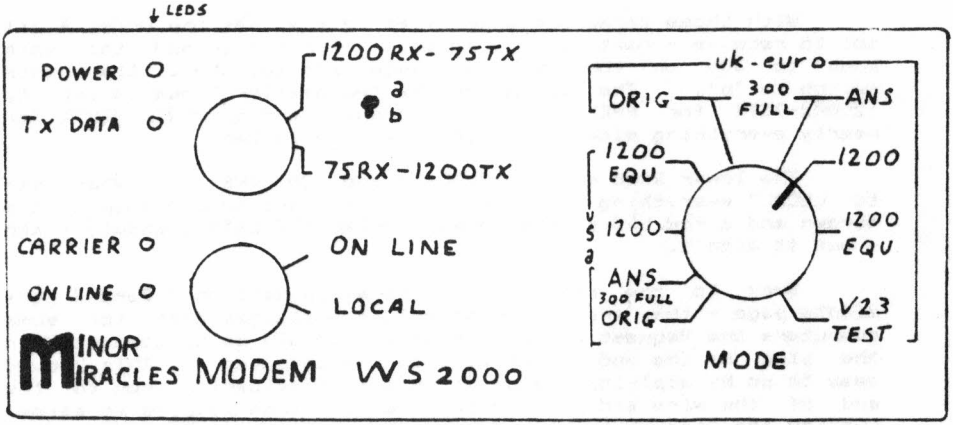
He will be greatly missed, not only by his wife, four children and other members of family but by all of his colleagues who had the pleasure of working with him.

Laurie Shields. 30th November, Chesterfield.

FUN WITH A MODEM

For some months I have had a HOMELINK account with the Nottingham Building Society (NBS) who have provided me with a Tandata modem. This means that by connecting the modem to a television set and a telephone jackpoint I can not only call up my NBS account to pay regular bills directly from it, and to transfer money to and from my Bank of Scotland current account, but I can also get the full use of PRESTEL, including Micronet and other computer associated pages.

This was fine, but I could not get a print-out of what was on the screen. The Tandata modem can print only to their own thermal printer. Just at that time I saw the Minor Miracles W2000 modem described in the Personal Computer World - a multi-rate modem for £125 + VAT.



From:- MIRACLE TECHNOLOGY (U.K.) LTD.,
P.O. BOX 48,
IPSWICH IP4 2AB

It comes in a box 6" wide x 2½" high x 6¼" deep - just big enough to sit my telephone on. It has a reasonably good manual which, very helpfully, has the more important paragraphs in heavy type.

Needless to say, everything was not plain sailing. The first problem was fitting it. The RS232 interface edge connector is underneath the Model 4, and so has to have a plug with the cable coming out at right angles to the connector, but the plug supplied by Minor Miracles for the Model 4 protrudes nearly an inch from the base. The plug can be changed to one of the right kind with a little care, or you can get a short (7½") extender cable from Tandy with a right angle female plug on one end and a male plug on the other for about £15.

Once that problem was solved I thought I was in business. I connected one cable of the modem to the RS232 interface of the Model 4, another to a power point, and the third to the telephone jackpoint, and then plugged the phone into the modem. The phone is now left permanently connected like this, where it performs normally until the modem is switched from "local" to "on line", even if the modem is not plugged in.

Next, I looked up the communications library command "COMM" in the TRSDOS 6 manual. In fact, to get things operational you

have to look up several different sections of the manual (it wouldn't do to make things too simple!), but it boils down to entering the following three commands from DOS:-

SET *CL TO COM/DVR (page A-114)

SETCOM (BAUD=300) - or whatever rate you want (page 1-140)

COMM *CL (page 1-41)

This was where the next snag emerged. PRESTEL sends data down the telephone line at 1200 Baud, but you send back your messages at 75 Baud, and nowhere in the manual did it suggest how TRSDOS 6 deals with two different Baud rates at the same time (that's because it doesn't, as I discovered later). So I settled for starting by trying to log onto a bulletin board at 300 Baud for both input and output.

With those three commands typed in that was the Model 4 all set to receive - next was the modem. First I turned the main knob to 300 on the UK-Euro range, and set the little toggle switch to "b". The top one of the two smaller knobs is set to 1200RX-75TX for PRESTEL and Micronet, and to 75RX-1200TX for nearly everything else, so I set it to the latter.

The lower knob selects "LOCAL" and "ON LINE". When set to "LOCAL" everything you type on the keyboard should show on the screen and a red LED on the modem, marked "TX DATA", should flash - but it didn't.

Back to the manual. There, in Section 3 Part 3 (the middle page - they are not numbered) it explained that for some computers the Request to Send (RTS) pin should be disconnected in the plug on the end of the lead to the computer. This proved easy to do by applying the point of a fine soldering iron to the end of the wire and pulling it free, and then tucking it aside. Cutting the wire would have done just as well, but this way made it easier to reverse if that was not the right solution.

This time success! The letters I typed appeared on the screen and the red light winked away!

The next step was to try it out by calling up a bulletin board. Minor Miracles have very thoughtfully provided a list of telephone numbers for 25 British, 19 USA, and 5 European ones. Our own FORUM-80 in Milton Keynes, and Frederick Brown's FORUM-80 in Hull were both engaged, so I dialed up Peter Toothill's MAILBOX-80 in Liverpool. When I heard the high pitched tone that indicated data was coming down the line I turned the bottom small knob from "LOCAL" to "ON LINE", and sure enough, a welcoming message appeared on the screen. It was all working at last! I was even given a number as a registered user of MAILBOX-80.

The next few minutes was spent working my way through the MAILBOX-80 HELP file, pausing every screenful to press CONTROL* to get a hard copy from the printer.

So, the modem was giving me access to bulletin boards, but what about PRESTEL? That is where NATGUG's hot line to Brian Pain came to my rescue once again. He gave me the names and phone numbers of several people who might help, and my first call, to Don Bannister in Belsfast, struck oil.

The TRS-80 operating systems do not support the split Baud rates needed for PRESTEL, but he had written a communications

programme for the Model I and the Model III which does, and which also overcomes the other problems those models have with PRESTEL, namely displaying the 24-line PRESTEL page on the 16 line TRS-80 screen. I could have used that in the Model III mode but, even better, he was actually working on a version for the Model 4 and expected it to be ready in about a month's time.

A few weeks later a disc arived from Don containing his programme which also writes the information on the screen to a disc file if wanted, a separate programme for printing out the disc files, a disc file of sample PRESTEL pages, and a text file of instructions. All this for £20 (I believe the Model I and Model III versions are a little cheaper).

I immediately made a backup disc and then printed out the file of instructions and viewed the sample pages, but I waited until after 6 p.m., when the phone calls are cheaper and the PRESTEL computer is free, before trying out the main programme.

Nothing but a blank screen. I tried every combination of modem setting that seemed at all likely, trying them with and without the TRSDOS 6 COMM command operting, and also with DOSPLUS IV. All to no avail. Finally I had to admit defeat and phoned Don Bannister for help. He was very surprised, but after some discussion we tried it out at 300 Baud by connecting our two computers together over the phone. This worked well, and we spent a few minutes conversing via the keyboard before he pointed out that it was not doing my phone bill any good. It was most interesting to see how we could switch from from speech to computer-to-computer connection and back just by switching the modem between "ON LINE" and "LOCAL".

This proved that the programme worked at 300 Baud, whether the DOS has a communication utility or not, and I was left to try again at 75/1200.

All this had, naturally, been done using the backup disc, so, after I hung up I tried the original disc Don had sent me, and up came the PRESTEL logo on the screen! Perhaps the backup was faulty, although that had never happened before. Then I remembered I had used a shortcut which I had just discovered - with a system disc in drive 0 and a formatted disc in drive 1 I entered "BACKUP :0 :1", and then when the question "Diskette contains data, use or not?" comes up on the screen I replaced the system disc with the source disc and answered "Y". This saves all the hastle of constantly swopping discs during a backup.

Well, there is a notice beside the computer in the office which says "when all else fails - consult the manual", so I did just that, and there, near the end of the BACKUP section, is the warning that what I did "will cause incorrect information to be written to the destination disc that will later corrup data". A simple explanation of why my short cut is not worth taking.

Now, thanks to Don Bannister, I can get PRESTEL on my computer instead of commandeering the family television when I want to pay a bill or book my holidays, and perhaps the above saga may help someone else negotiate the tricky path between buying a modem and actually getting everything to work. (By the way, I've discovered that Super Utility Plus lets you back up data discs without having to keep swopping them).

Ian Heward Wright,
17 Greygoran,
Sauchie,
Clackmannanshire,
FK10 3ET.

DISPLAYING MENU ON BOOT-UP

Ever wanted to disply a menu with single key entry on boot-up and still retain the ability to enter a DOS Command. After such experimenting with JCL Files and Auto Commands, I finally hit on the following method which involves the creation of another SYSTEM file. With LDOS 5.1.4. this will be SYS13/SYS.

When the system is booted SYS0/SYS is loaded and executed. After all the initilisation and configuration is done there is a jump to 4400H, a series of overlay load requests. The overlay load request consists of a LD A,xx and an RST 28 instruction. The value in A determines which overlay is loaded and with which option code.

The value in A is a bit record, broken down as follows:

Bit	0-2	Sector No of Dir Sec containing overlay to be loaded.
Bit	3	Offset in to directory sector for file entry.
Bit	4-6	Overlay Request Option
Bit	7	Must always be Zero.

So a LD A,93H will decode as follows:

93H = 10010011

Bit 0 - 2 = 011 = 3	Equals the fourth sector after the GAT and HIT sectors.
Bit 3 = 0	Equals the first entry in the Dir Sec. Consequently only the first two enteries in each sector are available for System Files.
Bit 4 - 6 = 010 = 2	Which equals option code Two. More on this later.
Bit 7	Must always be one. If not an error is generated.

On examining the directory with a zap utility it will be found that with LDOS 5.1.4 there is only one slot available for another SYSTEM file which is the second entry in Sector Nine.

To load SYS13/SYS the bit record will be as follows.

Bit 0 - 2 = 7 = 111	The ninth sector of the directory.
Bit 3 = 1	The second entry in that sector.
Bit 4 - 6 = 011	We will use option code three.
Bit 7 = 1	

Which equals 10111111 = 0BFH

So if x'4401' is zapped with a 0BFH, the system will attemp to load the system file whose directory entry is in in the second slot of the ninth directory sector. With long directories there is no increase in the available space for system files as the maximum number is determind by the the method described above.

On entry into a system file the first command is AND 70H. This isolates the option code still in the A register.

So 0BFH ANDed with 70H will produce a value of 30H

```
          10111111  = BFH
 AND      01110000  = 70H
          --------
          00110000  = 30H Option Code three.
```

or

```
          10010011  = 93H
  AND     01110000  = 70H
          --------
          00010000  = 10H Option Code one.
```

To create a new system file the disk must be zapped directly. Starting at sector nine byte 20H, zap the following values:

```
    5F 07 F4 00 00 53 59 53  31 33 20 20 20 53 59 53
    AE 01 F5 9C 04 00 FF FF  FF FF FF FF FF FF FF FF
```

Also zap the HIT sector byte 27 with A6. This is the hash code for SYS13/SYS. This will create SYS13/SYS with a record length of zero. When the source code is assembled and saved it will then use the existing directory entry. The following zap will direct SYS0/SYS to SYS13/SYS:

```
.Patch to SYS0/SYS.RS0LT0FF
.Forces SYS0/SYS to load SYS13/SYS.
D1,AA=AF  )  This modifies the 402DH and 4030H vectors to return to
D1,AD=AF  )  SYS13/SYS
D2,58=BF  )  This is the zap to 4401H
```

The following zap by-passes the initialisation routine duplicated in SYS13/SYS.

```
.Patch to SYS1/SYS.RS0LT0FF
.By-passess initilisation routines
D0,45=48,FE,20,20,44
```

A few points on the SYS13/SYS Source Code. The ORG address must be 4E00H as this is the DOS Overlay Area. As many files can be added to the Menu providing that the end address does not exceed 51FFH. The number of files that can be used therfore is limited and is dependant on their legnth. I have go 12 on ay file will some quite long multiple commands. If 51FFH is exceeded strange things will start happening.

Cpl D S Sutherland
1 CHESHIRE
Stanley Fort
BFPO 1

```
4E00            00010           ORG     4E00H
4E00 E670       00020           AND     70H             ;ISOLATE OPTION
4E02 FE30       00030           CP      30H             ;OPTION THREE
4E04 CA1E4E     00040           JP      Z,CLEAR
4E07 FE20       00050           CP      20H             ;OPTION TWO
4E09 C0         00060           RET     NZ              ;RET TO CALLER. UNKNOWN OPTION.
4E0A 212344     00070           LD      HL,4423H        ;KEYBOARD FLAG
4E0D CB7E       00080           BIT     7,(HL)          ;IS TYPE AHEAD ACTIVE
4E0F 2016       00090           JR      NZ,CON          ;IF SO JUMP TO CONT
4E11 21154F     00100           LD      HL,MES1         ;SPACE BAR PROMPT
4E14 CD6744     00110           CALL    4467H           ;DISPLAY IT
4E17 CD2800     00120 LOOP      CALL    28H             ;KEYBOARD SCAN
4E1A FE20       00130           CP      ' '             ;IS IT THE SPACE BAR
4E1C 20F9       00140           JR      NZ,LOOP         ;IF NOT, LOOP TO KEYBOARD SCAN
4E1E CDC901     00150 CLEAR     CALL    1C9H            ;CLEAR SCREEN
4E21 216C4E     00160           LD      HL,MENU         ;MAIN MENU
4E24 CD6744     00170           CALL    4467H           ;DISPLAY IT
4E27 31E041     00180 CON       LD      SP,41E0H        ;INITILISE STACK
4E2A FB         00190           EI                      ;INTERRUPTS ON
4E2B 3A0F43     00200           LD      A,(430FH)       ;SYSTEM FLAG
4E2E 07         00210           RLCA                    ;BIT SEVEN INTO CARRY FLAG
4E2F 3EC3       00220           LD      A,0C3H          ;OP CODE FOR JP
4E31 3801       00230           JR      C,KEY           ;TEST FOR DEBUG ACTIVE
4E33 AF         00240           XOR     A               ;A TO ZERO
4E34 321543     00250 KEY       LD      (4315H),A       ;CONTRUCT A JUMP IF DEBUG ACTIVE
4E37 211843     00260           LD      HL,4318H        ;COMMAND LINE BUFFER
4E3A 063F       00270           LD      B,3FH           ;MAXIMUM NUMBERS OF CHARACTERS
4E3C CD4000     00280           CALL    40H             ;CALL LINE INPUT ROUTINE
4E3F 38DD       00290           JR      C,CLEAR         ;CARRY FLAG SET IF BREAK PRESSED
4E41 78         00300           LD      A,B             ;NUMBERS OF CHARACTERS INPUT
4E42 FE01       00310           CP      1               ;IS IT ONE CHARACTER. IF MORE THAN
4E44 C2694E     00320           JP      NZ,SYS1         ;ONE IT MUST BE A DOS COMMAND
4E47 7E         00330           LD      A,(HL)          ;FIRST CHARACTER IN BUFFER
4E48 CBAF       00340           RES     5,A             ;CONVERT TO UPPERCASE
4E4A FE41       00350           CP      'A'
4E4C 2807       00360           JR      Z,ONE
4E4E FE42       00370           CP      'B'
4E50 2808       00380           JR      Z,TWO
4E52 C31E4E     00390           JP      CLEAR           ;NOT AN ALLOCATED LETTER
                00400 ;         And as many files as required.
4E55 21344F     00410 ONE       LD      HL,FONE
4E58 1805       00420           JR      MOVE
4E5A 213D4F     00430 TWO       LD      HL,FTWO
4E5D 1800       00440           JR      MOVE
                00450           ;Duplicates number of files in previous section.
4E5F 111843     00460 MOVE      LD      DE,4318H        ;COMMAND LINE BUFFER
4E62 D5         00470           PUSH    DE              ;SAVE IT
4E63 013F00     00480           LD      BC,63           ;63 CHARACTERS TO MOVE
4E66 EDB0       00490           LDIR                    ;MOVE FILE NAME INTO BUFFER
4E68 E1         00500           POP     HL              ;COMMAND LINE BUFFER ADDRESS
4E69 3E93       00510 SYS1      LD      A,093H          ;LOAD AND EXECUTE SYS1/SYS
4E6B EF         00520           RST     28H             ;WITH OPTION CODE 2
4E6C CC         00530 MENU      DEFB    0CCH            ;SPACE COMPRESSION CODE
4E6D 44         00540           DEFM    'Donald Sutherland's Disk Operating System'
4E96 0A         00550           DEFB    10              ;LINE FEED
4E97 CC         00560           DEFB    0CCH            ;SPACE COMMPRESSION CODE
4E98 2D         00570           DEFM    '----------------------------------------'
4EC1 0A0A       00580           DEFW    0A0AH           ;TWO LINE FEEDS
4EC3 DB         00590           DEFB    0DBH            ;SPACE COMPRESSION CODE
4EC4 4D         00600           DEFM    'Main Menu'
```

```
4ECD 0A        00610          DEFB    10              ;LINE FEED
4ECE DB        00620          DEFB    0DBH            ;SCPACE COMPRESSION CODE
4ECF 2D        00630          DEFM    '----------'
4ED8 0A0A      00640          DEFW    0A0AH           ;TWO LINE FEEDS
4EDA 41        00650          DEFM    'A - Scripsit'
4EE6 C5        00660          DEFB    0C5H            ;SCPACE COMPRESSION CODE
4EE7 42        00670          DEFM    'B - Format'
4EF1 C5        00680          DEFB    0C5H            ;SPACE COMPRESSION CODE
               00690          ;And again as many files as required.
4EF2 45        00700          DEFM    'Enter file required or DOS Command
4F14 0D        00710          DEFB    0DH             ;CARRIAGE RETURN
4F15 0A0A      00720 MES1     DEFW    0A0AH           ;TWO LINE FEEDS
4F17 50        00730          DEFM    'Press Space Bar to continue.'
4F33 0D        00740          DEFB    0DH             ;CARRIAGE RETURN
4F34 53        00750 FONE     DEFM    'SCRIPSIT'
4F3C 0D        00760          DEFB    0DH             ;CARRIAGE RETURN
4F3D 46        00770 FTWO     DEFM    'FORMAT :1 (NAME,SIDES=2,CYL=40,STEP=2)
4F63 0D        00780          DEFB    0DH             ;CARRIAGE RETURN
               00790          ;Enter file names as you would from DOS Ready.
4E00           00800          END     4E00H           ;END
00000 TOTAL ERRORS
29931  TEXT AREA BYTES LEFT
```

```
CLEAR   4E1E 00150    00040 00290 00390
CON     4E27 00180    00090
FONE    4F34 00750    00410
FTWO    4F3D 00770    00430
KEY     4E34 00250    00230
LOOP    4E17 00120    00140
MENU    4E6C 00530    00160
MES1    4F15 00720    00100
MOVE    4E5F 00460    00420 00440
ONE     4E55 00410    00360
SYS1    4E69 00510    00320
TWO     4E5A 00430    00380
```

MODEL 100 ROM FUNCTIONS

LCD FUNCTIONS

FUNCTION NAME	DESCRIPTION	HEX ENTRY ADDRESS
LCD	Displays a char on LCD @ current cursor position (also RST 4) Entry condition: A = Char displayed Exit condition: None	4AFA
PLOT	Turn on pixel at specified location Entry condition: D = x coord. (0 - 239) E = y coord. (0 - 63) Exit condition : None	70FA
UNPLOT	Turn off pixel at specified location Entry condition: (as for PLOT) Exit condition : None	70FB
POSIT	Set Cursor Position Entry condition: H = Column No (1 - 40) L = Row No (1 - 8) Exit Condition : None	4258

EASCA Send specified Escape Code sequence 424C
Entry Condition: A = Escape code
Exit Condition : None

ROUTINES FOR GENERATING COMMON LCD FUNCTIONS AND ESCAPE CODES

FUNCTION NAME	DESCRIPTION	HEX ENTRY ADDRESS	EQUIV ESC
CRLF	Generate a Carriage Return	41FE	-
HOME	Move cursor to home pos'n (1,1)	4209	-
CLS	Clear Display	420D	-
SETSYS	Set system line (lock line 8, LABEL)	4211	T
RSTSYS	Reset sys. line (unlock line 8,LABEL)	4216	U
LOCK	Lock Display (no scrolling)	421B	Y
UNLOCK	Unlock Display (scrolling)	4220	W
CURSON	Turn on Cursor	4225	P
CUROFF	Turn off cursor	422A	Q
DELLIN	Delete line at current cursor pos'n	42FF	M
INSLIN	Insert blank line at cursor pos'n	4234	L
ERAEOL	Erase from cursor to end of line	4239	K
ENTREV	Set reverse character mode	4245	P
EXTREV	Turn off reverse character mode	424A	q

VARIABLE AND STATUS LOCATIONS

NAME	CONTENTS	ADDRESS
CSRY	Cursor Position (ROW)	F640
CSRX	Cursor Position (COLUMN)	F641
BEGLCD	Start of LCD Memory	FE00
ENDLCD	End of LCD memory	FF40

KEYBOARD FUNCTIONS

FUNCTION NAME	DESCRIPTION	HEX ENTRY ADDRESS
KYREAD	Scan keyboard for key, return with/without one Entry condition: None Exit condition : A = character, if any Z flag - set if no key found - reset if key found Carry - set (char in code table below) - reset (normal char set code) When Carry is set (1), Register A will contain one of the following:-	6EF2

REGISTER A	KEY PRESSED
0	F1
1	F2
2	F3
3	F4
4	F5
5	F6
6	F7
7	F8
8	LABEL
9	PRINT
0A	SHIFT-PRINT
0B	PASTE

CHGET — Wait and get character from keyboard — 12D4

Entry condition: None
Exit condition : A = character code
Carry - set if special character
- reset if normal character
(<F1> - <F8> return pre-programmed strings)

CHSNS — Check keyboard queue for characters — 13E4

Entry condition: None
Exit condition : Z flag set if queue empty,
reset if keys pending

KEYX — Check keyboard queue for chars of BREAK — 6F1F

Entry condition: None
Exit condition : Z flag set if queue empty,
reset if keys pending
Carry - set when BREAK entered
- reset with any other key

BRKCHK — Check for BREAK chars only (CTRL-C or -S) — 6F32

Entry condition: None
Exit condition : Carry - set if BREAK or PAUSE entered,
- reset if no BREAK chars.

INLIN — Get line from keyboard (terminated by ENTER — 45F8

Entry condition: None
Exit condition : Data stored at location F68C

USING FUNCTION KEYS ROUTINES

The function table consists of character strings to be used by the keyboard drive when process <F1> - <F8> keys. The strings have maximum length of 16 characters and are terminated by a "80" (Hex) code. If the last character of the string is OR'ed with 80, the character will also serve as a terminator. The entire string will be placed in the keyboard buffer when the appropriate strings for all 8 keys is pressed. You must specify character strings for all 8 function keys (use the terminator byte for any you wish to ignore).

EXAMPLE OF FUNCTION TABLE

```
FCTAB   DEFM    'FILES'     ; F1
        DEFW     0D80
        DEFM    'LOAD'      ; F2
        DEFB     80
        DEFM    'SAVE'      ; F3
        DEFB     80
        DEFM    'RUN'       ; F4
        DEFW     0D80
        DEFM    'LIST'      ; F5
        DEFW     0D80
        DEFB     80          ; ignore F6
        DEFB     80          ; ignore F7
        DEFM    'MENU'      ; F8
        DEFW     0D80
```

FUNCTION NAME	DESCRIPTION	HEX ENTRY ADDRESS
TFNK	Set function key definitions Entry condition: HL = Address of funct. table Exit condition : None	5787
LRFLK	Clear funct. key def'ns table (fills with 80's) Entry condition: None Exit condition : None	5784
SPNFK	Display function keys Entry condition: None Exit condition : None	4284
TDSPF	Set and display function keys Entry condition: HL = Start address of funct. table Exit condition : None	4281
RAFNK	Erase function key display Entry condition: None Exit condition : None	4266
FNKSB	Display function table (if enabled) Entry condition: None Exit condition : None	57A9

PRINTING ROUTINES

PRINTR	Send char to line printer Entry condition: A = char to be printed Exit condition : Carry - set if cancelled by BREAK - reset if normal return	6A45

PNOTAB	Print char without expanding tab characters Entry condition: A = Character to be printed Exit condition :	1484
PRTTAB	Print a char expanding tabs to spaces Entry condition: A = Character to be printed Exit condition :	4B0B
PRTLCD	Print contents of LCD Entry condition: None Exit condition : None	1E29

RS232C and MODEM ROUTINES

RCVX	Check RS-232c queue for characters Entry condition: None Exit condition : A = Number of chars in queue Z flag - set if no data - reset if characters pending	6A75
RV232C	Get a character from RS-232C receive queue Entry condition: None Exit condition : A = Character received Z flag - set if OK - reset if error (PE,FF,or OF) Carry - set if BREAK pressed else reset	6A86
SENDCQ	Send an XON resume character (CTL-Q) Entry condition: None Exit condition : None	6B13
SENDCS	Send an XOFF character (CTL-S) Entry condition: None Exit condition :	6B26
SD232C	Send a character to RS232 (with XON/XOFF) Entry condition: A = character to be sent Exit condition :	6B3A
BAUDST	Set Baud rate for RS-232C Entry condition: H = Baud Rate (1-9,M) Exit condition : None	6B7D
INZCOM	Initialise RS-232C Entry condition: H = Baud Rate (1-9,M) L = Uart configuration code (see UART byte desc. below) Carry - Set for RS-232C Exit condition : None	6BAE

BIT(S)	Description
0	Specifies number of stop bits : 0 = 1, 1 = 2
1-2	Parity Setting : 00 = None, 01 = Even, 10 = Odd
3-4	Word Length : 00 =6, 01 = 7, 10 = 8

The byte is ANDed with 1FH to ignore Bits 5-7. The string containing the current STAT setting is located at F662 (5 bytes) : Baud, Length, Stop Bits and XON/XOFF switch.

SETSER — Set serial interface parameters and acttivate RS-232C — 17C8

Entry condition: HL = Start address of ASCII string containing parameters terminated by a binary Zero ('78E1E',0). Syntax same as in telecom's STAT

Carry - Set for RS-232C

Exit condition : None

CLSCOM — Deactivate RS-232C — 6BE4

Entry condition: None

Exit condition : None

CASSETTE RECORDER ROUTINES

DATAR — Read character from cassette (no checksum) — 6CD3

Entry condition: None

Exit condition : D = Character from cassette

CTON — Turn motor on — 1484

Entry condition: None

Exit condition : None

CTOFF — Turn motor off — 1436

Entry condition: None

Exit condition : None

CASIN — Read char from cassette and update checksum — 14BC

Entry condition: C = current checksum

Exit condition : A = Character

C = contains updated checksum

CSOUT — Send char to cassette and update checksum — 14CD

Entry condition: A = Char to be sent

C = current checksum

Exit condition : C = updated checksum

SYNCW — Write cassette header and sync byte only — 6BEF

Entry condition: None

Exit condition : None

SYNCR — Read cassette header and sync byte only — 6C2E

Entry condition: None

Exit condition : None

DATAW — Write a char to cassette (no checksum) — 6C04

Entry condition: A = Character to be sent

Exit condition : None

RAM FILES ROUTINE

The Directory Table (located at F969) contains all file location, type and stat information. Each file is managed by an 11 byte directory entry in the format:

```
Byte 1        Directory Flag (for file type and status)
Bytes 2-3     Address of file
Bytes 4-11    8 Byte filename
```

The Directory Flag contains the following information:

```
Bit 7 MSB                  1 if valid entry
Bit 6                      1 for ASCII text file (DO)
Bit 5                      1 for Machine Language (CO)
Bit 4                      1 for ROM file
Bit 3                      1 for Invisible file
Bit 2                      Reserved
Bit 1                      Reserved
Bit 0                      Internal use only
```

```
MAKTXT   Create a text file                                    21E7
         Entry condition: Filename (max 8 bytes must
                          be stored in FILNAM(OFCA5). 'DO'
         Exit condition : HL = Top address of new file
                          DE = address of directory entry (FLAG)
                 Carry - set if new file already exists
                       - reset if new file
```

```
CHKDC    Search for file in directory                          57B4
         Entry condition: DE = address of filename to find
                          (ASCII filename + 0 byte terminator)
         Exit condition : HL = Start address (TOP) of file
              Z  Flag - 0 (file found)
                      - 1 (file not found)
```

```
GTXTTB   Get top address of file                               57EE
         Entry condition: HL = Address of directory
                          entry for file
         Exit condition : TOP start address of file
```

```
KILASC   Kill a Text (DO) file                                 1F90
         Entry condition: DE = File TOP start address
                          HL = Address of directory
                               entry (Flag)
         Exit condition : None
```

```
INSCHR   Insert a character in a file                          6869
         Entry condition: A = Character to  insert
                          HL = Address to inser char
         Exit condition : HL = +1 ,Carry set if out of memory
```

MAKHOL	Insert specified no of spaces in file Entry condition: BC = Number of spaces to insert HL = Address to insert spaces Exit condition : HL & BC are preserved Carry - set if out of memory	6875
MASDEL	Delete specified number of characters Entry condition: BC = No of chars to delete HL = Address of deletion Exit condition : HL & BC are preserved	68A7

OTHER ROUTINES

INITIO	Cold start reset Entry condition: None Exit condition : None	69DE
IOINIT	Warm start restart Entry condition: None Exit condition : None	69E8
MENU	Go to main menu Entry condition: None Exit condition : None	54A4
MUSIC	Make tone Entry condition: DE = frequency B = duration Exit condition : None	6F73
TIME	Ready system TIME Entry condition: HL = Address of 8 byte area for TIME Exit condition : HL = > TIME (hh:mm:ss)	18D7
DATE	Read system DATE Entry condition: HL = Address of 8 byte area for DATE Exit condition : HL = > DATE (dd/mm/yy)	18F7
DAY	Read system DAY of the week Entry condition: HL = Address of 3 byte area for DAY Exit condition : HL = > DAY (ddd)	192C

David Roberts

FOR SALE AND WANTED

PRAXIS MEDICAL SOFTWARE

OUTRIGG ST. BEES CUMBRIA CA27 0AN
Tel. No. (0946) 822326

SPECIAL DEMONSTRATION OFFER.

The only way to assess adequately the merits of any computer system is to have "hands-on" experience for a reasonable amount of time. To simply spend an hour or two having the system demonstrated to you is totally inadequate. To allow doctors to examine the PRAXIS system in detail we are currently offering (for £35, incl. V.A.T.) a demonstration copy of our system, consisting of a program disc, two data discs and a comprehensive manual, to doctors owning or having access to a TRS80 Model III or IV. Doctors having access to the necessary hardware can thus spend as long as they wish trying out the system and deciding whether or not they would like to install it in their practices. Should they decide to do so, the cost of the operational system will be £750 (plus V.A.T.); and the £35 paid for the trial package will be credited on return of the package.

The software provided for the home demonstration will function exactly as the standard issue system, but certain protection devices have been added to prevent these copies being used for any purpose other than demonstration.

Please note that the software has to be supplied on a LDOS system disc. We therefore need proof of ownership of LDOS or, if you do not already own LDOS, we can supply it for £60.00, inclusive of V.A.T. and postage. The cost of this will also be credited if a system is subsequently purchased from us. Orders for this system must be on the enclosed form.

GENIE I incl. Sound and Cassette 16k
Expansion Interface - 32k
includes RS232 Printer and Double Density Disc Interface
2 40 Trk Disc Drives with integral Power Supplies
Exatron Stringy Floppy Drive with Power Supply and Tapes
Large Quantities of Disc & Tape software
incl. LDOS,NEWDOS80
EDTASM,MACRO80 Assembler
Tape And Disc Versions Scripsit
Many Games & Utilities
Large no of Books & Manuals
ALSO A Seperate RS232 Interface with Software control of Speed etc.

Whole Package $500 but prepared to sell individual items

Barry Reynolds
17 Cresswells Mead
Holyport,Maidenhead
Berks. SL6 2YP

Tel.Maidenhead(0628) 39595

	Volume	Issue
MACHINE CODE Adding Single Precision numbers.	5	3
Beginners guide	3	6
Easy Graphic titles	3	10
From BASIC	3	8
Glossary of Codes	3	5
In BASIC DATA statement	2	2
Right Shift 16-Bit	3	12
Tutorial	1	4
Writing with PENCIL	2	4
and BASIC	1	9
MADELEINE Daisy Wheel Font	2	10
MAILBOX Review	3	5
MAILING LIST - tape based program	4	5
MANUALS Suggested guidelines	4	9
MAPLIN MODEM Carrier Detection	5	10
Construction Tips	4	10
MATHS GAMES revieweded	2	9
MAXIMANAGER Database Comments	4	7
MC68000 System Observations	5	10
MDBS Database Review	3	12
MEM - how it is calculated	4	3
MEMORY EXPANSION Audio Computer Board	4	1
DIY on Model I	1	3
" " "	1	4
" " "	1	5
on Video Genie	2	9
MEMORY SIZE setting within program	2	6
MENU in Machine code using ZEN	5	9
MERGING two BASIC programs (see also Basic Tips)	2	11
MICOS EPROM Tape I/O speed up mod	4	4
MICROFILES database review	4	9
MICROLINE 80 Printer Review	3	3
MICROLINE 82 Linefeed Selection	4	7
MICROLINE Control Codes	4	6
MICROMOVIE in school	3	3
MICROMUSIC saved to disk	2	9
MICRONET 800 Review	5	3
MICROSOFT Compiler review	4	6
MICROTEK - a warning'	2	12
MODEL I imminent demise	2	7
MODEL 100 Disk Interface	5	10
Interface to Model 1	5	7
Interface to Model 3	5	9
ROM Disassembled	5	7
MODEL 2 Direct File Access	5	5
Disc problems	4	5
and a Modem	4	5
Some experiences	2	6
Interfacing with Model I	4	1
Software Library	3	12
TRSDOS on the Model 2	3	1
MODEL 3 first impressions	2	9
disk transfer to/from Model I	4	6
interface with Model 100	5	9
Input/Output bus	3	12

	Volume	Issue
MODEL 3 Memory map	3	2
(cont) use in Business	4	12
MODEL 4 first Impressions	4	12
Initial Appraisal	5	3
MODEM	5	6
Observations	5	7
PAL Chips	5	10
PDRIVE specs. under NEWDOS	5	4
Reviewed - plus how to enable two screens	5	3
with STERM communications prog	5	6
tips	5	4
two months later'	5	3
MODEL 16 first impressions	4	5
preview	3	9
" "	4	4
MODEMS a Line-powered modem	5	3
British Telecom 2A Connections	4	7
Modtech Modem	2	9
null Modem	5	7
single chip device	4	2
testing your own	2	11
MODEM/CMD Review	4	5
MODTECH MODEM Review	2	9
MONITOR - DIY simple monitor program	4	7
BMON review	2	11
MON3 review and criticism	2	12
see also:- ZEN and DEBUG (Newdos)		
MORSE Code hardware modification	1	8
MSF CLOCK NPL	2	2
muMATH Language reviewms	2	4
and criticismse Review	2	12
MUSIC write your own with CELESTE	4	6
review of ORCHESTRA 80	4	9
NASCOM Interfacing to TRS 80	2	9
NEWDOS PLUS Adding BOOT Commnd	4	10
NEWDOS 80 Assessment	2	4
Compared with other DOSes	5	9
Double Density	3	7
adding Drivers	2	4
EDTASM patch	2	6
Make files INVISIBLE	4	11
Ignoring TIME? prompt	4	12
JCL Files speed-up	4	11
JKL with Epson printer	4	5
LIST - a bug?	4	5
Lower Case Driver	2	3
MODEL 4 PDRIVE's	5	4
Model 1to 3 Transfer	4	6
PDRIVE Tutorial	3	9
Personalising Disks	4	2
" "	4	4
" "	4	5
Tutorial Part 1	2	7
Tutorial Part 2	3	1
Tutorial Part 3 (including SYSTEM and NEWDOS BASIC	3	10
Zapping	2	11

	Volume	Issue
NEWDOS DEBUG Adding COPY funct	2	10
More Modifications	2	11
NEWSAGENTS database program	4	3
NEWSCRIPT Word processor review/tutor	4	3
" " " " "	4	4
NULL MODEM construction	5	7
NUMERIC KEYPAD Installation	1	6
OLDEST TRS80 Is it mine?	4	2
OMIKRON Mapper Review	2	4
OMIKRON CP/M and Mapper Review	3	11
OPEN UNIVERSITY and TRS 80	2	2
ORCHESTRA 80 Music program review	3	7
another review	4	9
PASCAL Cassette I/O	4	9
with EPSON printer	4	9
with LinePrinter	1	10
Print@ Equivalent	4	9
Tiny PASCAL	1	6
Tutorial 1	3	9
Tutorial 2	3	10
UCSD Pascal review	1	9
availability	2	4
and Video Genie	4	6
PASSWORDS removed by ZAPS	1	7
PERCOM Separator review	2	5
PERMUTATION Generator	2	1
PHOTOPEN review	3	6
PLOTTER/PRINTER (Tandy) review	3	3
POUNDS SIGN accessing on Model I	2	5
POWERDRAW review	4	5
PRESTEL 300 BAUD Service	4	7
via Acoustic Coupler	4	11
BASIC program loading	5	3
GATEWAY system description	5	3
with ST80 III	4	7
PRIMARY Schools software	2	11
PRINT BUFFER Board	5	9
PRINT Screen Contents	2	2
PRINTER PEEKing Status	5	9
Pagination	4	3
Set-up program	4	12
PRINTERS - Two printers per computer	2	4
PRINTUSING with pounds not dollars	5	6
PROBABILITY FUNCTION	2	11
PROFIT and Loss Sorting /Selecting	4	5
PROTECTED software Comments	4	3
PROZAP review (Disk Zapper)	3	7
PURGE Disks routine	3	7
QUICKPRINTER Review	2	1
RANDOM NUMBER GENERATOR in the ROM	2	1
RANDOM NUMBERS Generating using C.	5	7
RANDOM NUMBERS Generating in BASIC	1	11
RELAY replacement	2	10
Use as a buzzer	4	7
RENUMBER Basic programs in Basic'	2	9
(correction)	2	11
RESEARCH MACHINES 380Z and TRS80	3	3
RESET without losing Basic text	3	5

	Volume	Issue
ROM CALLS (useful ones) on Model I	1	9
ROMAN NUMERALS conversion	5	3
ROMS Identifying Level 2 Roms	4	3
RS232 Interface Building and Testing	4	7
Testing	4	9
NATGUG's RS232 Interface	3	12
with serial printer	1	8
RUN in L2ROM Explained	2	12
SCREEN Flicker cure	3	12
SCREEN INPUT under Cursor control	2	7
SCREEN-MEMORY TRANSFER Direct	1	12
" " "	2	1
" " "	2	2
BASIC using strings	2	10
In M/C (fully explained)	2	2
SCREEN PRINTING contents of	2	2
with EPSON graphics	4	5
Genie HiRes graphics	4	5
SCRIPSIT		
Reviews of SCRIPSIT	1	10
" " "	1	12
Compared to other wordprocessors	4	2
SCRIPSIT		
Systems:-		
Running under NEWDOS	3	11
Getting DIRectory of disk	4	1
Printing with EPSON	2	11
Printing with Daisy Wheel	2	8
Cassette Scripsit	4	4
Model 3 Cassette (1500 baud)	3	6
and VIDEO GENIE	3	12
SCRIPSIT		
Uses:-		
As a BASIC Editor	2	3
with a DISASSEMBLER	3	1
for SORTING tables	4	12
with a typesetter	5	7
with VISICALC	4	11
SCRIPSIT		
Counting words:-		
In a text file (using M/C)	3	12
From within SCRIPSIT, by patching ?L to do count	4	3
Tape SCRIPSIT	5	4
Adding a ?Z instruction	5	5
Recovering lost text:-		
by dumping memory to disk	2	7
by re-entering SCRIPSIT	2	8
by dumping to disk using ZEN	2	9
from ACULAB SCRIPSIT	3	2
Printing Printer Control Codes:-		
Patch to allow in-line printer control codes	4	11
Using LDOS print routines	4	3
Another 2 methods!	2	2
and yet another!!	2	5

	Volume	Issue
SCRIPSIT (cont) Other modifications:- Addresses used by SCRIPSIT	5	6
Changing Control key (hardware mod)	2	7
Changing control codes (soft)	3	12
Partial screen scroll	2	5
Printing POUNDS sign	2	8
SERIAL INTERFACE (cheap)	1	3
Cheaper	2	11
Intelligent	3	8
SERIAL PRINTER INTERFACE (easily built)	1	6
SERVO Interfaced to Model I	3	6
SLAVE Z80 interfaced to TRS80	2	6
SLIDE PROJECTOR Controller	1	8
SNAPP BASIC Review	4	5
SOFTY EPROM Programmer	1	10
SORTING BASIC Integers	1	10
by RANK	5	3
using CMD 'O' - tutorial	4	7
using CMD 'O' - more help	5	9
multidimensional arrays	4	5
using TASORT - review	4	2
using tables from word proces.	4	12
SOUND EFFECTS CELESTE Music Prog review	4	6
With ORCHESTRA 80	4	9
from BASIC	3	8
Wiring into Video Genie	2	10
SOUND GENERATOR interfacing	2	6
SOUND I/O Simple Circuit/Prg	2	1
SPACEV/CMD On Disk	2	11
With LDOS	3	6
SPECTROSCOPY ATOMIC solving	3	7
SPECTRUM (SINCLAIR) and TRS80 linked	4	12
SPEECH I/O Hardware Mod	3	5
SPEECH SYNTH (Hardware)	1	8
SP 0256 Chip	5	10
Using GI SP0256 chip	4	12
VOXTRAX Chip Review	4	3
SPEEDY tape loading program review	3	9
SPELLING CHECKERS compared	3	5
ST80 (SMART TERMINAL) review	1	6
STRUCTURED programming guidelines	2	2
using Level 2'	2	10
SUPERPIMI Database review	3	8
SUPERSCRIPT review	2	9
patch for Daisywheels	4	2
POUNDS sign display	3	5
typeface codes	5	7
SUPERUTILITY Review	3	9
SUPERZAP - sending output to SCRIPSIT	4	2
SWIMMING and recording Results	3	6
TANDYMAIL review	4	3
TANDYSCRIBE printing control character	3	1

		Volume	Issue
TAPE	Automatic RUN after SYSTEM load	3	3
	B17 speed-up program	3	7
	COPY 16K tape copier review	3	12
	Copying tape to tape directly	2	6
	Formatting of tapes	1	4
	Identifying unknown tapes	3	5
	Loading improvement	2	2
	Mail list program	4	5
	find Name of SYSTEM tape	3	6
	Saving data blocks	2	1
	Saving data files	1	5
	Speed up using MICOS EPROM	4	4
	TC-8 Fast Cassette	2	3
	Tandy's improvements	4	9
	Unlock CLOAD jams	3	7
	Using VIDEO GENIE 2nd tape port	4	7
	VTAPE Utility review	3	2
TAPE	RELAY reducing the strain	2	9
	replacing	2	10
TBUG	Modifying registers	1	2
TEAC	DRIVES exposed!	2	7
TELESOFTWARE	the realities	4	1
TELETYPES	Using with TRS80	4	4
TEMPERATURE	Farenheit to Cent.	3	10
TERMINAL	PROGRAM (simple)	4	2
	(see also COMMUNICATIONS)		
TEXT	FILES converting to Lower case	2	11
TIMING	using INTEL 8253 chip	5	7
TIMING	your program using the Clock	2	5
TOOLKIT	Enhanced Basic - review	2	10
	another one	3	2
TRACKCESS	Review	3	9
TRASHMAN	Discussion	4	10
TRSDOS	- a bug?	4	2
	Backup/Format patch on GENIE	5	5
	Faster disk Disk I/O	2	2
	Forty tracks patch	3	12
	" " "	5	3
	reformatting used disk	4	5
	single key commands	4	6
TRSDOS 1.3	Disable DATE request	5	10
TRSDOS 2.3	Review	1	7
TRSDOS	Model 2	3	1
TRSDOS 6.1	Hints/Patches	5	7
TSHORT - BAS	IC Keywords Instan	2	12
TYPING	Direct from keyboard to printer	4	6
UNIVAC	With Model I	3	11
VARIABLES	counting usage by BASIC	4	3
	listing all used	4	2
VARIDISK	Bugs solved	4	3
VERSAFILE	Database Review	4	2
VIDEO GENIE	Adding CLEAR key	2	10
	Adding hex keypad	3	1
	Edge connector pin	3	5
	HIRES modification	4	3
	Inverse Video	3	8
	Keyboard Decoded	2	10
VIDEO GENIE	Port FD and 37E8	4	12

		Volume	Issue
(cont)	Printer port	4	12
	Printer port under LDOS	4	3
	Screen Clearing	2	7
	Selecting Cassette port	4	6
	Using 2nd Cassette port	4	7
VISICALC	Review and two tutorials	3	9
	Sorting VISICALC data	4	1
	Printing with EPSON	4	1
	Printing via SCRIPSIT	4	11
	VCFORMAT Modification	3	5
VOLTAGE	reduction to prevent reboots	2	2
	more comments	2	10
VOXTRAX	Speech synthesiser review	4	3
VTAPE	Tape loading Utility review	3	2
VTOS/LDOS	uses	3	1
WALLPAPER	Electronic	2	7
WANG OIS	Micro and TRS 80	5	3
WARRANTY	The Tandy warranty (joke)	3	11
WORD 4 WORD	Review and Tutorial	4	6
	Printing CHRS(0-12)	4	9
WORD PROCESSOR	for sorting tables	4	12
WORD PROCESSORS	- see CSW, COPYART, ELECTRIC PENCIL, SCRIPSIT, NEWSCRIPT, WORD-4-WORD.		
	For a comparison of some, see	4	2
XBAS	ACCULAB Extended Basic review	2	12
ZBASIC	Compiler review	4	6
ZEN	(Original version) review	2	3
ZEN 82	Improved Version'	3	5
	and more improvements	3	6
	comments	3	9
ZEN 84	review	5	5
ZGRAPH	review	4	5
Z80	- running a slave from a TRS 80	2	6
68000	PROGRAMMING Tutorial	5	6
68000	Processor and the Model I	4	6
80 GRAPHIC	HIRES Board Review	3	2
80 GRAPHIX	notes	3	5
80 NET	Instructions	3	9
80NET	announcement	4	4

==========

NATGUG NEWS

Volume 6 Issue 6 December 1984

OFFICIAL JOURNAL OF THE

National TRS-80
& Genie Users
Group.

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.

The Group maintains two software libraries (Models I and II) which are free to members. Library lists are obtainable from the Secretary.

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

CONTENTS

2. Information on the Group
4. Editorial
4. Guidelines for contributors
4. Library information
5. Members' letters
6. Accessing the video on the Model 3 and 4
12. Easy labels
13. Using long record lengths with TRS-80 random files
15. Data management with SIR
19. Some Cs for Model 4
24. For sale and wanted
24. Mods to Peter Hall's modem
25. Re-inking printer ribbons
26. Update on illegal commands
26. Windows on the TRS-80

EDITORIAL

We haven't heard any howls of protest about the suggestion that we will probably be going bi-monthly, due to the lack of contributions and the declining membership. We will leave things as they are until the end of the current volume, and probably go bi-monthly with the start of the new volume in July.

If any members have upgraded to 80-track double-sided drives, and still have their old 40-track single-sided TEACs or whatever gathering dust, you might be able to sell them to impecunious IQLUGgers, now that a disk interface is available for the QL. Please drop me a line if you have any drives you want to get rid of, and I'll advertise them in the IQLUG newsletter.

We will be holding three Swindon weekend workshops in 1985, one in March (March 1/2/3), one in the summer, and the usual autumn gathering. We have decided to go for three meetings because of the interest shown by the members of IQLUG. We won't have any problems organising programmes for the IQLUGgers, but, as I said in the last issue, we are running out of interesting topics for NATGUGgers, so please let Brian know if you have any suggestions, or better still, if would like to give a presentation yourself.

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 hand-written material, write legibly.

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

LIBRARY INFORMATION

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 I library lists are available from Brian Pain.

MEMBERS' LETTERS

A couple of weeks ago I sent you a letter which mentioned Kansas computer software. At that time I didn't know if the company was still in business. Since then I've managed to make contact, and have established the following facts:
1: Kansas are still in business, mainly selling BBC programmes, but still sell their old range of TRS-80 software. They currently sell such software at half the last catalogue price. eg, now £4.75 for most arcade games.
2: At one time Kansas were prepared to sell unprotected versions of some of their games (mainly older ones, such as "U Boat Attack"). However, the unprotected tape masters have now been "lost", and are unavailable.
3: Kansas still keep stocks and masters of their protected tapes, to honour their lifetime guarantees.

This seems to clarify the position on copying to disk; Kansas still aren't going to tell anyone how to crack their security, although they evidently don't intend to return to the TRS-80 market. This is a pity, because some of their arcade games are extremely good.

Marcus L. Rowland

22, Westbourne Park Villas,
London W2 5EA,

Last August I wrote to Mr. Leighton Davies at the address given in July NATGUG NEWS asking for a list of Model 1 software and enclosing a large stamped self-addressed envelope.

I know that voluntary organisations with honorary officers cannot be expected to operate at the speed of light, but I do feel that if the Welsh wizard of the Model 1 was ever going to function at all he would have done so by now!

I have a number of queries on commercial Model 1 software which I possess and had hoped for some help in dealing with them. My TANDYMAIL program, for instance, was written by Understanding Ltd., who seem to have gone out of business. Their software bugs, however, have not!

I would be most grateful if you could let me know whether the Model 1 library is still functional, and if so to whom queries should be addressed.

Gordon Millington,
126 Grange Road,
Guildford,
Surrey GU2 6wP.

(* Library lists are available from Brian Pain, not Leighton. LFH *)

Accessing the Video on the Model 3 and 4

At a recent club meeting I was asked to produce a solution to 'PEEK'ing and 'POKE'ing the Model 4's screen. After the initial reaction of 'impossible' I recalled one of the SVC calls in the Dos that could copy the video memory to an accessible part of user Ram and also copy it back. With a little bit of machine code it should be possible to set up two USR's using these SVC's and thus provide the long lost utility that no TRS80 programmer should be without. The first problem was testing the logic and as I only had a Model 1 to work with we ended up with versions for both machines, but don't discard the Model 1/3 routine as un-necessary as there are some hidden benefits.

In order to use the technique a block of memory equal to the size of the video Ram is needed and the simplest way of allocating part of user memory for this purpose is to 'DIM'ension an integer array. Finding the size couldn't be simpler since each integer requires two bytes of memory the whole Model 1 screen (1024 bytes) would need 512 integers. To avoid any wasted memory we will use the zero element in the array to store the first & second bytes of the screen so dimension statement will be: DIM SCR(511). The machine code routine itself will need about 22 bytes of memory and we again have a number of choices, POKEing into reserved memory, packed strings, adding another 11 elements to the array for storing the screen, etc. but the neatest way would be to set up another smaller integer array just big enough to hold the program. Let's call this one USR(n) where n will be decided after writing the routine.

On the Model 1/3 the screen is easily accessed being memory mapped from 3C00H to 3FFFH inclusive. To copy the contents to or from our array we only need to set up the HL and DE registers as pointers, load BC with the number of bytes (1024 dec) and execute an LDIR instruction. To keep the USR calls simple we will include the parameter 0 or 1 to indicate whether the the data moves 'Out' or 'In' relative to the array. Since a block move Usr routine might come in useful for other purposes we will make sure that the locations within the array for these source, destination and byte-count values are such that they can be easily modified from Basic.
Now for the assembler part, this was written with Zen the layout of the listing being slightly different from what you may be used to.

```
                ;Screen/a
                USRIN:    EQU   0A7FH      ; Model 1/3 Rom routine
                VIDEO:    EQU   3C00H
                
                          ORG   0A000H     ; could be anywhere
                          LOAD  $          ; load code into memory
                
A000 CD7F0A     USR:      CALL  USRIN      ; get User parameter in HL
A003 7D                   LD    A,L        ; put into A ignoring H
A004 B7                   OR    A          ; check if 0 or 1
A005 210000               LD    HL,$-$     ; get Varptr(SCR(0)) into HL
A008 00                   NOP              ; padding
A009 11003C               LD    DE,VIDEO   ; screen location into DE
A00C 00                   NOP              ; padding
A00D 010004               LD    BC,1024    ; prepare to move 1024 bytes
A010 2801                 JR    Z,MOVE     ; if move Out then skip
A012 EB                   EX    DE,HL      ; switch registers for move In
A013 EDB0       MOVE:     LDIR             ; move data/screen contents
A015 C9                   RET
                          END
```

For those who have never used machine code programs stored in integer arrays the following diagram should make everything crystal clear. The memory addresses are shown as an offset from the first byte of array US%(10) found from VARPTR(US%(0)).

Memory Address	Decimal Value	Array Subscript	Array Value	Z80 Code	Instruction Mnemonic
+ 0	205	0	32717	CD	CALL
+ 1	127			7F	User Parameter
+ 2	10	1	32010	0A	to HL
+ 3	125			7D	LD A,L
+ 4	183	2	8631	B7	OR A
+ 5	33			21	LD HL,
+ 6	0	3	0	0	Varptr of
+ 7	0			0	US%(0) to be supplied
+ 8	0	4	4352	0	NOP padding
+ 9	17			11	LD DE,
+ 10	0	5	15360	00	Video Screen
+ 11	60			3C	Address
+ 12	0	6	256	00	NOP padding
+ 13	1			01	LD BC,
+ 14	0	7	1024	00	1024 Byte
+ 15	4			04	Count
+ 16	40	8	296	28	JR Z,
+ 17	1			01	to MOVE
+ 18	235	9	-4629	EB	EX DE,HL
+ 19	237			ED	MOVE:LDIR
+ 20	176	10	-13904	B0	2 byte instr.
+ 21	201			C9	RET

The tricky part is line 11 where we will load the HL register with the start of the integer array used for storing the screen. Since this array could be located virtually anywhere in memory this value can only be supplied at run-time. Conveniently however this address is stored in the 6th and 7th bytes of US%(n) thus forming US%(3).
To put the address of first byte of SC%(0) there we use the BASIC statement: US%(3) = VARPTR(SC%(0))

As the assembler has usefully put the object code into memory we can use a little Basic program to Peek the bytes out in decimal format for inclusion into a Data statement. We may as well resolve the problem of integers bigger than 32767 right from the start, so the program is:

```
100 'Getcode
110 A=&HA000
120 INPUT"Number of bytes ";B
130 FOR N = 0 TO B STEP 2
140 L = PEEK( A + N ): H = PEEK( A + N + 1 ): C = 256 * H + L
150 IF C > 32767 THEN C = C - 65536
160 PRINT C; ",": NEXT
```

Which produces this output:

```
32717, 32010, 8631, 0, 4352, 15360, 256, 1024, 296, -4629, -13904,
```

The Basic program incorporating the above as a Data statement to test the logic on the Model 1, was:

```
100 'Stest/Bas
110 DIM X%, US%, US%(10), SC%(511)
120 FOR N% = 0 TO 10: READ US%(N%): NEXT
130 DATA 32717, 32010, 8631, 0, 4352, 15360, 256, 1024, 296, -4629, -13904
140 END
300 US% = 1: GOTO 320 '  Save screen      'I'nto the array
310 US% = 0:          '  Restore screen   'O'ut of the array
320 US%( 3 ) = VARPTR( SC%( 0 )):
    DEFUSR = VARPTR( US%( 0 )):
    X% = USR( US% ):
    RETURN
```

Run the program and after Ready is displayed key GOSUB 300, saving the screen contents; key CLS followed by GOSUB 310 and the screen is restored. The video is now very easily transferred back and forth to the array SC% without any need for setting memory size or loading m/c utilities. Even better the whole of the screen can be written off to disk or tape as a file, or as part of a file containing a number of different video displays. Such a technique could be used in a menu driven program where two or three different menus are used during different stages of the program's execution and rather than clearing the screen and printing the menu each time the USR routine would give and immediate refresh. The time taken on reading the data from disk into the integer arrays during program initialisation would hardly be noticed.

However when I started to write such a program to demonstrate the idea we came across the rather protracted procedure invented by Microsoft for transferring numeric quantities to and from file buffers. To do the job according to the book we would have had to write FIELD statements covering 128 integer elements with the associated 128 CVI, RSET and MKI statements. Newdos80 programmers would have it a lot easier using "FF" files and Igels but even with that superb operating system the process was rather slow compared with the speed at which the data was moved to and from the screen. If only we had a foolproof way of finding out whereabouts in memory Disc Basic had allocated the file buffer then we could use our own block move Usr and save messing about with FIELD, CVI, MKI etc.
In the end finding the buffer turned out a lot easier after a bit of thought. Simply FIELD a one byte string, 1 AS UB$, and then ask Basic for the VARPTR(UB$). The value returned is the address of a 3 byte block of data giving the length of the string, the least and then the most significant bytes of the address of the string itself. These latter two values are the ones we need so the address of the file buffer is given by:

```
140 OPEN "R", 1, "SCREEN/DAT"
    FIELD 1, 1 AS UB$
    UB% = 256 * PEEK( VARPTR( UB$ ) + 2) + PEEK( VARPTR( UB$ ) + 1
```

Armed with this information and remembering that the buffer is only 256 bytes we can write a routine to send the contents of the array to disc as follows.

i) set the indicator US% = 0 for output from array
ii) set the element US%(5) = the file buffer address
iii) set the element US%(7) = 256; the size of the block to be moved
iv) re-define the USR address in case Basic has moved the arrray
v) set up a FOR/NEXT loop count of 0 to 3
vi) set the element US%(3) = appropriate position in array US%(n)
vii) activate the Usr to block move 1/4 of array to file buffer
ix) PUT the file buffer to the appropriate disc file record
x) Loop until executed four times.

To input from disk it is only necessary to do the reads before the block moves with the appropriate indicator in US%. Combining the disk I/O and the Usr sections together generated a program that would very successfully executed block moves in any possible combination. What was lacking however was a routine that enabled the video screen to be edited before being saved. The simple one line word processor:

```
10 IN$=INKEY$: IF IN$="" THEN 10 ELSE PRINT IN$;: GOTO 10
```

showed signs of promise and a bit of experimenting indicated that the shifted-arrow keys performed a lot better than the unshifted. Note that on the Model 3 it is necessary to press Shift, Down arrow & Letter Z to generate the Shift-down of the Model 1. So the next stage was to test for CHR$(91), CHR$(10), CHR$(9) or CHR$(8) and change the values to 27, 26, 25 or 24 as appropriate. Contolling the cursor location by manipulating the values in &H4020/21 to the top, bottom, right or left hand edges of the screen in response to shifted arrow-keys if they were pressed ended up with a simple screen editor to cover all our needs. The shifted '@' is used to toggle a graphics mode whereby 64 is added to characters for printing, and one last problem, discovered later was the need to inhibit printing in the bottom right hand corner of the screen as this forces the Rom to scroll the screen up one line and ruin the display.

The final version of the Model 1/3 demonstration program is shown in Listing 1. The main features have already been described but a more detailed explanation of some of the finer points may be required. Firstly a listing of all the variables and their use:

C4$	Cursor on	Chr$(14) Used by the screen editor.
C5$	Cursor off	Chr$(15) " " " " "
CU%	Cursor pos.	&H4020 stored as variable for speed.
GR%	Graphic switch	Used to generate grapics characters.
IN$	Inkey string	All keyboard input.
N%	Loop counter	Used during disk I/O as sector count.
SC%(511)	Screen array	Integer array to store screen data.
SE$	Editor string	Used by editor checking input for cursor keys.
UB$	Fielded var.	Used to find file buffer.
UB%	Buffer addr.	Address of file buffer.
UN%	Screen Number	Counter for screens in file Screen/Dat.
US%	Usr parameter	Sets Into or Out from array SC%
US%(10)	Usr array	Integer array storing machine code Usr.
X%	Dummy	Used to activate USR routine.

Taking the program in sections:

Lines	Function
100 - 140	Program initialisation.
150	Generating the character string SE$. The screen editor checks all input characters against this string or validity and deciding appropriate action.
160 - 180	Printing the menu.
190	Switch on the cursor, call the INKEY$ subroutine, switch off the cursor, test for validity and Goto as required. The +1 forces a Goto 190 if non-valid key pressed.
191 - 198	Calls to subroutines as required ending with Goto menu.

```
100 'Dscreen/bas
110 DIM IN$, C4$, SE$, N%, US%, X%, UN%, UB%, C5$, UB$, CU%, GR%, US%(10), SC%(511)
120 FOR N%=0TO10: READ US%(N%): NEXT
130 DATA 32717,32010,8631,0,4352,15360,256,1024,296,-4629,-13904
135 CLS:PRINT:LINEINPUT"Filespec for screen data (SCREEN/DAT) : ";IN$:IFIN$="" THEN IN$="SCREEN/DAT"
140 OPEN"R",1,IN$:
    FIELD1,1ASUB$:
    UB%=256*PEEK(VARPTR(UB$)+2)+PEEK(VARPTR(UB$)+1)
150 CU%=&H4020:C4$=CHR$(14):C5$=CHR$(15):
    SE$=CHR$(13)+CHR$(31)+CHR$(91)+CHR$(10)+CHR$(9)+CHR$(8)+CHR$(27)+CHR$(26)+CHR$(25)+CHR$(24)+CHR$(96)
160 CLS:
    PRINTTAB(15)"Screen to Disk Utility":
    PRINTTAB(15)"======================"
170 PRINT"
<1>  Screen to Memory          <2>  Memory to Screen
<3>  Disk to Memory            <4>  Memory to Disk
<5>  Screen to Disk            <6>  Disk to Screen
<7>  Edit Memory               <8>  Edit Disk"
180 PRINT"<9>  Display Disk Screens      <A>  Memory to printer
<B>  Disk to printer           <@>  Finish":PRINT:
    PRINT"Select  ? ",
190 PRINTC4$;: GOSUB230: PRINTC5$;:
    ON INSTR("123456789AaBb@",IN$)+1 GOTO 190,191,192,193,194,195,196,197,198,199,205,205,206,206,210
191 GOSUB400: GOSUB300: GOTO160
192 GOSUB310: GOSUB230: GOTO160
193 GOSUB220: GOSUB300: GOTO160
194 GOSUB220: GOSUB340: GOTO160
195 GOSUB220: GOSUB400: GOSUB300: GOSUB340: GOTO160
196 GOSUB220: GOSUB330: GOSUB310: GOSUB230: GOTO160
197 GOSUB310: GOSUB410: GOSUB300: GOTO160
198 GOSUB220: GOSUB330: GOSUB310: GOSUB410: GOSUB300: GOSUB340: GOTO160
199 FOR UN%=0 TO LOF(1)/4 -1
200 GOSUB330: GOSUB310: PRINT@60,UN%;: GOSUB230
201 IF IN$="[" THEN UN%=UN%-1: IF UN%>0 THEN 200 ELSE UN%=0: GOTO200
202 IF IN$="@" THEN UN%=LOF(1)/4-1: GOTO204
203 IF IN$="E" OR IN$="e" THEN GOSUB310: GOSUB410: GOSUB300: GOSUB340
204 NEXT: GOTO160
205 GOSUB550: GOTO160
206 GOSUB220: GOSUB330: GOSUB550: GOTO160
210 CLOSE: END
220 PRINT: INPUT"Screen save number ";UN%: RETURN
230 IN$=INKEY$: IFIN$="" THEN 230 ELSE RETURN
300 US%=1: GOTO320:     ' Screen to array
310 US%=0:              ' Array to screen
320 US%(5)=15360: US%(7)=1024: US%(3)=VARPTR(SC%(0)):
    DEFUSR=VARPTR(US%(0)): X%=USR(US%): RETURN
```

```
330 US%=1: GOTO350:      ' Disk to array
340 US%=0:               ' Array to disk
350 US%(5)=UB%: US%(7)=256: DEFUSR=VARPTR(US%(0))
360 FOR N%=0TO3:
        US%(3)=VARPTR(SC%(N%*128)):
        IFUS%=1 THEN GET1,1+N%+4*UN%: X%=USR(US%)
        ELSE X%=USR(US%): PUT1,1+N%+4*UN%
370 NEXT: RETURN
400 CLS:                 ' Screen Editor
410 PRINTC4$;: GOSUB230
420 ON INSTR(SF$,IN$) GOTO 440,450,460,470,480,490,500,510,520,530,540
430 IF PEEK(CU%)=255 AND PEEK(CU%+1)=63 THEN410
    ELSE IF NOT GR% THEN PRINT IN$;:    GOTO410
         ELSE PRINT CHR$(ASC(IN$)+64);: GOTO410
440 PRINTC5$;: GR%=0: RETURN
450 PRINTCHR$(30);: GOTO410
460 PRINTCHR$(27);: GOTO410
470 PRINTCHR$(26);: GOTO410
480 PRINTCHR$(25);: GOTO410
490 PRINTCHR$(24);: GOTO410
500 PRINTC5$;:
    POKE CU%,PEEK(CU%)AND63: POKE CU%+1,&H3C:
    GOTO410
510 PRINTC5$;:
    POKE CU%,PEEK(CU%)OR192: POKE CU%+1,&H3F:
    GOTO410
520 PRINTC5$;:
    POKE CU%,PEEK(CU%)OR63:
    GOTO410
530 PRINTC5$;:
    POKE CU%,PEEK(CU%)AND192:
    GOTO410
540 GR%=NOT GR%: GOTO410
550 FOR N%=0 TO 15:
        FOR X%=0 TO 31:
            GR%=SC%(N%*32+X%):
            LPRINT CHR$(GR% AND 255);:
            GR%=GR%/256:
            GR%=GR%-256*(GR%<0):
            LPRINT CHR$(GR%);:
        NEXT: LPRINT:
    NEXT: LPRINT " ": RETURN
```

For Sale: Prism Acoustic Modem (75/1200 Baud) for Micronet, Prestel, etc. BT approved. With transformer and RS232 lead, instructions for Model 3, but no driving software. Note that this won't work with the standard Model 1 communications package, which has a 300-baud minimum and requires send and receive at the same rates. Mistaken auction purchase, selling very cheaply because I can't test or guarantee.
Only **£20 inc P&P** (PO or cheque) from:
M.L.Rowland, 22 Westbourne Park Villas, London W2 5EA.

199 - 204	Option 9, Display Disk Screens allows backstepping to a previous screen, editing any screen whilst on display and premature exit with the '@' key if required.
210 - 230	Standard routines.
300 - 320	Block move of screen as described earlier but video address and byte count values updated in case of alteration by the disk I/O section.
330 - 370	Disk I/O section. As described earlier with the variable US% being used to test whether Read from or Write to disk.
400 - 540	The screen editor, each part is explained.
400	Entry point for a completely fresh screen.
410	Edit screen entry point. Switch on cursor and get character.
420	Test input key for control code and if not then key must be character for printing and drop through to next line.
430	If cursor position is lower right hand corner then 410 Otherwise if in graphics mode add 64 before printing.
440	Exit from editor. Switch off cursor and reset graphics.
450	Clear key, Chr$(31), used to clear to end of line.
460 - 490	Change unshifted arrow keys to shifted.
500 - 530	Moving cursor to screen limits in response to the shifted arrow keys.
540	Switching the value in GR%, this line is in effect: IF GR% = 0 THEN GR% = -1: GOTO140 ELSE GR% = 0: GOTO140

Run the program and what you end up with is limited solely by your imagination. The potential is there to create any combination of text and graphics on multiple screens and store them safely on disk for re-use in other programs with a stripped down version of the Usr calls to read from disk, store in arrays and put on the screen without any delay whatsoever. So far the routine has proved very useful during discussions with commercial clients in creating displays for stock control, order entry, customer detail files and the like, where the ultimate user has the chance of seeing how his data will appear on the screen when the final package is presented.

Laurie Shields

EASY LABELS

There are some job which can be done better without using computer assistance even though they appear to fit in with a software package. Given the need to send out several hundred duplicated letters it would appear to be ideal for a mail type program. However the labels can come unstuck and jam in the printer and the whole job can take several hours.

As the mailing list was fairly static it was easier to print a master sheet on plain paper and to get labels produced on a photo copier. If a name or address changes it is simple to stick a new label over the old one on the master sheet and this saves having to reprint the whole thing. The method allows a simple check when the mailing is done in batches. If the label is still on the sheet then the mail shot has not gone out.

Derek Trayler

USING LONG RECORD LENGTHS WITH TRS-80 RANDOM FILES (without span-blocking)

I shall first refer to MBasic-86, then return to the TRS-80. MBasic-86 allows 32K record lengths but the manual doesn't explain how to field such a large buffer. The conventional field statement would only cope with about 7K, even with all fields of 255 bytes. I found a way round it by setting up an array for the field variables and then poking the string pointers of the array to point at the buffer.

It occurred to me that one could do the same on the Model I/III/4, but it's not quite so simple because the file buffers are not contiguous, being separated by FCBs and so forth. I therefore set up the record buffer in an area of protected memory and then mapped the file buffer onto it. I append a demo program to this note. It assumes a DOS default record length of 256, which is OK for Model III DOSes but perhaps not for some on the Model I. It also assumes that your Basic allows variable-length records.

To use this technique you must be prepared to sacrifice an amount of memory at least equal to the record length of the monster file, which might be as much as 20K. Always enter Basic with enough variable files and the memory size set to more than the record length below the top of RAM or the bottom of any user routines, whichever is the lower. The demo program assumes only one file, with giant records, using buffer #1; if you want to use other normal-length files, use the other buffers for them.

Now to explain what we are going to do. We are going to set up the fields as an array in protected memory. These are the actual logical fields required for your application. The demo program uses all string fields, but they could equally be two, four, or eight-byte numeric fields. We shall use two other field variables, FB$ and FC$. FB$ is the actual file buffer, which is all one field: if the logical record length is an exact multiple of 256, the buffer length is 128; otherwise it is calculated by the initialization routine at line 6000. A second field variable FC$, identical in length to the first, is set up to form a window on the protected memory; it is moved while the logical record is read from or written to disk. Once we have set up the fields, all we need to do to write a record to disk is to LSET or RSET the variables into the FA$ array, then use a loop to move the FC$ window, LSET FC$ into FB$, and PUT the subrecord FB$ buffers to disk. No further calculations are required. To read a record (as in the demo - a text file is a good test unless you are using TRSDOS 1.3) you use a loop to GET the subrecords from disk into the file buffer, point the window, and LSET FB$ into FC$ in its current position; after that, the variables can be extracted from the FA$ array. Remember NEVER to use any elements of the array or the field variables in the left-hand side of an assignment statement.

I don't know enough about the Mod 4 to know if this technique will work on it, but I see no reason why it shouldn't. It should also work under Mbasic and CP/M:- The method in line 5 for finding protected memory should work on any fairly standard version of Microsoft Basic.

A few bytes of disk space will generally be wasted at the

end of each logical record, but the wastage should always be less than 1 per cent of space used and the granules required for a file should be no more than with conventional span-blocking.

It is worth noting that the technique could be extended to provide virtual memory and to handle several such long records in the same area of protected memory, by making the FA$ array two-dimensional: one dimension would correspond to the file buffers. The size of the other dimension would be determined by the file with the most fields. The buffer lengths and the FB$ fields would each be in a one-dimensional array, and the initialization routine would form a loop to set up all the buffers. Since the string pointers of a 14 by 100 string array occupy less than 6K, one could in theory write a program to handle 14 files each with a record length of over 20K- the data for the number and lengths of fields would have to be read sequentially from a fifteenth file, since there would not be room for enough data statements!

```
4 REM DEMO LONG BUFFER PROGRAM
5 CLEAR 5000: DEFINT F:A$=A$:
    HI=CSNG(PEEK(VARPTR(A$)+1)+PEEK(VARPTR(A$)+2)*256):HI=HI+10
10 REM*** FOR MOD I/III
       HI=PEEK(16561)+PEEK(16562)*256:HI=HI+10'SAFETY
15 REM*** CHANGE THE SAFETY MARGIN ADDED TO HI TO SUIT YOUR
    SYSTEM, E.G. 500 BYTES FOR MBASIC-86
20 DEF FNMD%(A,B%)=A-(INT(A/B%)*B%)
22 CLS
25 LINE INPUT "NAME OF FILE TO ACCESS: ";F$
26 KW$=TIME$
30 GOSUB 6000: REM*** INITIALIZE BUFFERS
40 DIM B$(FO)
299 REM*** DEMO READ (E.G. OF EXISTING TEXT FILE)-1ST RECORD
300 KT$=TIME$:RC=1:GOSUB 1000:GOSUB 700:FOR N%=1 TO FO:PRINT
       B$(N%):NEXT:CLOSE:PRINT"START INITIALIZE AT:
       ";KW$:PRINT"START RECORD READ AT ";KT$:PRINT"STOP AT
       ";TIME$:END
499 REM WRITE VARIABLES TO MEMORY BUFFER: DEMO ONLY
500 FOR N%=1 TO FO:LSET FA$(N%)=B$(N%):NEXT:RETURN
699 REM READ VARIABLES FROM MEMORY BUFFER:DEMO ONLY
700 FOR N%=1 TO FO:B$(N%)=FA$(N%):NEXT:RETURN
999 REM READ RECORD FROM DISK-INVARIANT
1000 FSTART!=HI:FOR N%=1 TO NBUF%
1020 GET 1,(RC-1)*NBUF%+N%:REM*** RC=LOGICAL REC. NO.
1025 GOSUB 20050:REM *** MOVE WINDOW THROUGH MEMORY BUFF.
1030 LSET FC$=FB$:NEXT
1050 RETURN
1999 REM WRITE RECORD TO DISK-INVARIANT
2000 FSTART!=HI:FOR N%=1 TO NBUF%:GOSUB 20050:LSET FB$=FC$
2010 PUT 1,(RC-1)*NBUF%+N%:NEXT
2020 RETURN
5999 ' BUFFER INITIALIZATION ROUTINE:INVARIANT
6000 FSTART!=HI
6005 READ RL:'RECLENGTH
6010 READ FO:'NUMBER OF FIELDS LOGICAL
6020 DIM FA$(FO):'ARRAY FOR FIELDS
6030 NBUF%=INT(RL/256):IF FNMD%(RL,256%)=255 THEN RL=RL+1:GOTO
       6030
6040 IF FNMD%(RL,256%)=0 THEN NBUF%=NBUF%*2:BU%=128:GOSUB 20015
       ELSE GOSUB 20000:REM FIELD FILE BUFFER AND DUPLICATE
       BUFFERS
6049 REM NOW FIELD LOGICAL FIELDS
```

```
6050 FSTART!=HI
6060 FOR N%=1 TO FO:READ A%:TP=VARPTR(FA$(N%)):POKE
     TP,A%:GOSUB25000:FSTART!=FSTART!+A%
6070 NEXT N%
6100 RETURN
19999 REM *** FILE BUFFERS FOR IRREGULAR RECORD LENGTH
20000 NBUF%=NBUF%+1:BU%=INT(RL/NBUF%)
20010 IF FNMD%(RL,NBUF%)<>0 THEN BU%=BU%+1
20015 FC$=FC$
20020 OPEN"R",1,F$,BU%
20030 FIELD 1,BU% AS FB$
20040 RETURN
20050 TP=VARPTR(FC$):POKE TP,BU%
20060 GOSUB 25000
20070 FSTART!=FSTART!+BU%: RETURN
24999 REM  POINT STRING AT MEMORY BUFFER
25000 POKE TP+1,FNMD%(FSTART!,256):POKE
     TP+2,INT(FSTART!/256):RETURN
29999 REM DEMO DATA:CHANGE AS APPROPRIATE
30000 DATA  1066:'RECORD LENGTH
30010 DATA 6:'NO. OF LOGICAL FIELDS
30020 DATA 101,151,202,203,204,205:'FIELD LENGTHS LOGICAL
```

Chris Currie
14 Keston Road
London N17 6PN.

DATA MANAGEMENT WITH SIR

ISS was one of the first programs that I got to do anything useful on the 16K model I and through various upgrades it has improved with disk versions and extra memory. Although the program is still available there is a "son of" program called SIR. Although it retains the advantages of its predecesor it is a big improvement in its capacity and flexiblity. In fact within my own company it is a big talking point that my micro is more flexible than the mainframe.

Any Database is usually limited by the memory of a micro and uses various dodges to get round the restrictions. One way is to keep a keyfile in memory and access the discs to find the records as required. Another method is to cut the file into blocks of manageable size. These two compromises have advantages and disadvantages which make them more suitable for some applications than for others. If you have blocks then you waste a lot of time if you want to access data in alternate blocks and the keyfile method can be slow to search a lot of records. One second per record is reasonable but you soon get up into longish delays with large files.

SIR allows you to have some fields from all your records or all the fields of some of your records. Thus you may decide on the composition of you own blocks. Given a very large personal file; you could load in the age col and select a specific age group. These are then marked and all the other cols (fields) can be loaded in. You could also select by name, sex of any other parameter.

One of the nice features is the way you can use several methods as applicable. For instance, you might want most of a file and to exclude a few special ones. It is then quicker to manually delete these temporarily rather than try and select all but those few records. On the other hand you may only want

to find a few and it is quicker to select these by hand. Whatever method you choose it is possible to see the result in seconds. In fact sorting is over in a few seconds at the worst case. Most times it is too quick to see.

With some programs you can use very powefull conditional searches which will select files that meet those criteria. SIR will only sort or select on only one field at a time but these can be accumlative. If you select males or females, you can go on to select on age, name or any other classification that you have set up. Each one refines the selection a stage further and you are able to see the consequences of your selection. This method is aided by the rapid responce and saves the problem of conditional operators. If you have ever seen the consequences of mixing (&) for (OR) or (<) instead of (>) you will appreciate the point. If the answer is going to "none found" it is better to know quickly.

SIR has very strict rules on data entry and this sometimes causes problems. You can have a numeric col which will not accept letters or a letter col which will not accept numbers. This allows sorting to be accurate without leading zeros. However it precludes code numbers, so another type of field can be "mixed". That still requires you to specify what the first two characters will be. This is probably no problem with data that you know but can cause problems with miscellaneous input.

One field that caused no end of problems was the one used for address. One address may be 14 Oxford St. and another may be White Wings or Toad Hall. You tend to find yourself using dummy numbers to get round the entry problem. As it is not sensible to sort on address you are given the facility to specify non-sorting which allows you to enter any characters including spaces and numbers.

SIR allows you to change you mind to quite a degree and this is a big help when starting out. You can increase the width of cols and add new ones. You can change the order of cols quite easily and this creates a lot of benefits because sorts are affected by the relative position of the cols. If you have first name, second name it makes the names more readable but sorting would be on first name and Angela Brown may come next to Archibald Smith and not Betty Brown.

When you are setting up a database you may have one column which is only relavant on infrequent occassions. Therefore it goes on the end and is only loaded in when required. When you do need it, the column can be loaded in first and it becomes the primary search. There seem to be no penalties for having all fields sorting and the time taken relates strictly to the number of records taking part. As you narrow the number taking part, the sort gets even faster. If however you do not like the result of your search, you have only to specify a (N)ew sort and and all the records are back in the file.

As with most programs there are several tricks that you can practise to get even more out of your program. These are not documented and you use them at your own risk. The program sets up its file using ASCII which is relatively inefficient in using disc space. A figure is stored as a string of characters and not as an integer. This may be inefficent but it is much easier to use and it is helped by allowing you to rewrite the file as a text file with each line terminated. This will allow you to access the data with very simple BASIC programs or a word processor. You can also modify you data using a word processor but again you may cause some problems.

Manipulating the data got me round the only bug I have ever found in the program. When I converted a large ISS file using the convert utility I had a field with code letters. The program accepted 010 but if you edit the field it removes the leading zero and this caused it to sort out of order. If you enter a figure with a leading zero it stays but in the edit mode it is stripped. Not much of a problem and only likely to occur on rare occasions. I got round it by using Scripsit to replace the missing figure and save it back in ASCII.

SIR is one of the finest database programs for use on the screen but is limited in its printout facility. This may or may not be a problem. If you want a mailmerge or labels program you will have to write it yourself. This is not as much of a problem as it would appear. Owing to the ASCII format it is very easy to understand and the ablity to generate text files make it fairly staightforward.

There is one disadvantage when writing text files for use with a word processor. Each field is pushed up to the next without seperation. If spaces were added as with normal printout it would take up more room than the data file. This would cause problems if you had a maximum size file. On the other hand it is less practical when you want a quick printout using printcodes and page instructions which are not available in the main program. On the other hand if you had a text file that big it would not fit into most wordprocesors. My solution is to use a BASIC program to open out the field prior to printing.

If you need to update your data you can waste a lot of time making non entries. For instance it is some times necessary to work you way through all the fields of a record to get to the only one that requires alteration. Sir allows you to select a batch of records for updating and to work you way through one field only. Each time you have the option to alter the other fields with the minimum number of keystrokes. Professional users with large regular updates will appreciate the saving. There is nothing worse than the saga of "another record - yes - which one - 123 - is it this one - yes" and then require (enter) to be pressed 6 times to move through the fields to the one and only field to be modified.

One of the problems with ISS and still in SIR is the mandatory record numbers. You can select which fields are required as long as the first is the record number. This has led to much discussion. Why is the first record on a print out number 67? Because they are sorted alphabetically. But why is Angela number 67. Because she was the 67th name entered. Why is my number 71? Because it was the 71st name entered. Why did you enter her name first? !!!!!.

In practise, I learnt to cut the numbers off the print out and I really thought the record numbers were vital to the program. However when I tried to write a program which accessed the file by BASIC and treating the text as a Random Access file, I found that the numbers did not agree. If you delete a record, the SIR number is eliminated but the Random Access numbers close up. As an experiment I put blanks into the first 4 characters of each record and the program works wihout the annoying record numbers printing out. However if you add a new record it will have a number added.

Another bodge is obtainable via the Parameter file. Each data file has a parameter file which keeps a record of the position, length, type and heading for each field. The number

if fields is limited to 26 (one for each letter). If this is a serious problem you could consider having different Par files. For instance if you want to sort on dates you need to have the date in YYMMDD order. If you use three cols you can have them in DD MM YY order for reading and reverse them for sorting or ranging. Now this is fine if you have sufficient cols to spare. If you have two files with different headers you can access the same data in different way. You merely rename the text to match the header or copy it to another disc. This technique allows a club to keep extensive figures with nearly twice the number of cols.

SIR allows you to have a col or field name and also a three character header. When using single character fields you find the headers mergee into one line. Sometimes with non intended results especially when moving cols about. There is no facility to change headings and the offical way is to delete the col and insert a new one but this means re-entering all the data. A program such as Super Utility can make suitable modifications to the Par file.

SIR works on a 35 track single density model I or a 40 track double density model III. The size of the data files is limited by the disc capacity but never exceeds those stated. It ignores 80 tracks and crashed when trying to add records on a double density model I. It does not support multiple data drives and will not work on single drives. Although it will work on LDOS it does not seem to honor the high string and writes over LDOS filters. It will work on a model 4 in III mode but not in TRSDOS 6 in model 4 mode. This is a big pity because the model 4 is much more suitable for this type of program. The RAMDISC and extra memory would be very effective.

The Bottom line is that this is a very good program for certain applications but is not the all singing all dancing type. If you like ISS and have two drives then SIR is much better. If there were snags with ISS and you have not found an alternative then SIR might provide the answer. Like a Sports car it is fantastic on the open road being both practical and fun but it runs into its limitations when you drive off the road. Its no use for form letters or for long fields. It is great for getting quick answers and especialy using trial and error.

I have heard some people say that its best to write your own program to do the job you require in the way that you need it done. In that case SIR might be worth considering as a data validation and sorting subroutine. Its fast, efficient and well proven. For the less experienced programmer it produces text files that do not need data conversion and unpacking and a simple MID$ can work wonders.

Derek Trayler

Some 'C's for Model 4.

From Roy Barber.

It seems about time I did a bit more for the mag.(We cant leave all the work on model 4 to Anon can we.).

I have been looking for a suitable alternative to Basic and assembley language for some time now. I used Pascal for a while and as I have stated before, was very impressed with the Borland CP/M Turbo Pascal. I still think the Borland implementation is first class. However Pascal is not a flexible or free enough language for my tastes and after a quick look over Lisp, Modula 2, C and a few others I rapidly reached the conclusion that C was the way to go. C allows the user all the freedom to communicate with the system (mainly through its use of pointers) and allows function libraries to be built up and modules from it to be included in ones programs as required. The user can name the functions himself and through their use can even make the language look like another language altogether. I particularly like this feature as I can include functions I am used to on the HP systems I use at work. A couple of simple examples of this are 'beep' for a beep and wait(n) for a delay of n millisecs.

Please dont get the idea I am one of those characters who seem to love to knock Basic. I would not like to be without Basic, particularly when I want to do something like quick calculations or try out an idea straight from the keyboard. I believe Basic has its place in computer languages and think we would loose many newcomers to computing if they could not begin with Basic.

To get back to C. This will not be a tutorial on C, just a few tips which may help anyone with a model 4 or CP/M system to benefit from my experience.

The first thing a real beginner needs to know is what book to buy. I have seen most of them and read in detail at least five of these. My personal recommendation is for the Purdam book The C Programing Guide. It follows the Kernighan and Ritchie standards closely and all the program examples should be runnable as written on the majority of C compilers. The text is easy to follow and it is written as a teaching book rather than as a C reference. Some of the other books about seem to include examples which call functions not immediately available on the cheaper C systems. It is most frustrating to a beginner to carefully type in one of the first few examples of a two line program only to find the compiler consistently reject it with an error message. The beginner will probably not yet realise how such a simple example can be wrong but if the example is calling a function which is not in your library you have to write it or use another similar function that you have. The K and R book, The C Programming Language is always recommended as a must and most C suppliers documentation refer to it. It is certainly useful as a reference but in my opinion most beginners could get by without it, at least for the first few months if they have the Purdam book.

What about the language versions?. I have now used C/80 from The Software Toolworks, Eco-C and Misosys LC in that order. C/80 and Eco-C are CP/M programs and LC (or elsie as Misosys lovingly refer to it) is for running under LDOS. Pro LC is the same thing for TRSDOS 6.
C/80 is my favourite, for a beginner it is near to the K and R standard, close enough to satisfy most users. Eco-C is also close to the K and R standard and has a larger UNIX compatible library. It is quite a bit more expensive than C/80. LC is good but lacks a few quite important features. It also works differently to the more common CP/M versions and anyone using it without the experience of a CP/M C will not appreciate the advantages of the CP/M versions.

I suggest that if you have CP/M for your model 4 (and who could be withought it ?) and you wish to try C then C/80 offers the opportunity to get into the C world for a minimum outlay of about £50. The main omissions from C/80 are that no relocating assembler, library handler or linking loader are provided with the package. Floating point and long integers are also an extra at about a further £30. If you have CP/M Plus as I did you have the necessary RMAC, LIB80 and LINK with your CP/M and can build a quite sophisticated C system with full library facilities from C/80. The additional floating point and long package allows quite extensive maths operations to be performed with up to 24 bits of precision. If you do not have the DRI RMAC assembler set or the equivalant Microsoft M80 Package you can still learn C with C/80 with no additional expense. A quite good, purpose written assembler is include called AS. This will allow perfectly adequate programs to be produced. As the assembler is not relocating and a library handler is not included, program modules or library functions have to be included manually into ones programs instead of automatically as when RMAC or M80 used. This is not to imply that you have to write them out in full each time. All that is necessary is that the module or function source exists on the disk. One simply puts in ones program a line: #include "myfunctions.c" and the compiler does the rest, however then the source code is included and compiled with the program, not as efficient as the RMAC/M80 method where only the machine code module would be tacked onto ones program.
The documentation provided with C/80 is it weak point. Software Toolworks keeps its cost down for C/80 by only providing loose sheets of instructions instead of the bound handbooks of the more expensive versions. The necessary information is all there but I must admit is sometimes well hidden.

Time precludes me from going into too much detail on these programs as when discussing three packages in one go there is quite a lot of ground to cover, however anyone can phone me for more if they are really interested.
The beginner user of C/80 will probably not be too worried by the omissions from the full C with C/80. For guidance the main omissions are: no #line, no typdef and no bitfields. These omissions seem common to most non UNIX C packages and they are missing from all three of the packages I am discussing. Do not worry about them unless compatibility with a full UNIX C is your ambition. Such more important features as random file access facilities can be obtained with all three C,s.
With C/80 and Eco-C no text editor is provided for entering your

programs. If you have Wordstar this can be used but I find it frustratingly slow. If you have a good, fast text editor for your CP/M you will be happy to use it. If you have to use CP/M,s ED then I am sorry for you.
I (and a number of my colleagues) use the excellent editor supplied with Turbo Pascal. Borlands should supply this as a separate package as it is so good for writing source code. My method of handling the system with C/80 on the model 4 is by using submit files (do files if you are not into CP/M). At power up my system loads Turbo Pascal into Ramdisk. I write my code and save it on Ramdisk. I then switch to drive B on which my C system resides with all the libraries and assembler package. I then type: s c filename m (s for submit renamed s) and submit does the rest. If successful the outcome of the submit file is a working com or machine code program on the Ramdisk ready to be tried. This can then be run and if as is usually the case, modifications are required the source still on ramdisk can be reloaded int the editor, corrected, recompiled and retried until satisfactory. When the required result is obtained the source and the final com file is then copied to a seperate disk for permanance.

The compilation, assembly and linking is reasonably fast with C/80 and Eco-C. With short programs I can just about get my pipe filled and running nicely whilst the computer is doing the work (No comments about smoking at the computer please, I believe the smoke lubricates the heads !). For long programs I wait through the compilation phase as that is where most errors occur. If this is succesful I then go and make a cup of tea while the rest is handled automatically by the submit file. Up to now with C/80 and Eco-C the job has been completed before my tea is ready but C programs tend not to be extremely long.

Eco-C is similar in overall features to C/80 but has got some plus,s. The main advantages of Eco-C are: better documentation, you usually get the Purdam book with it. Jack Purdam is, I believe the author of Eco-C as well as the book although the book does not publicise the fact too obviously. A further plus to my mind is the provision of the M80 assembler package. This allows the use of Z80 code if you want to include assembly language in your coding. It also makes for slightly faster operating speed when required. The M80 package alone costs about £180 so at about £190 for the complete Eco-C system one cannot complain at the cost. I like Eco-C but must admit the main reason is because I can use The Z80 code. (You dont have to Know assembler to use C but if you do it will eventually prove usefull.). As an example of the need to use assembler, I interfaced my RTC and wrote a better random number generator using it as in line code in my C functions. The code produced by Eco-C appears slightly more efficient than that from C/80 but not significantly so. Compilation is slower but assembly faster. Overall there is not a great difference if RMAC and LINK is used with C/80. You still need a word processor for writing your programs.

Pro LC from Misosys is quite different. It runs under LDOS or TRSDOS 6. This can be a great advantage and combined with the true, memory mapped screen display instead of the slower serial type displays under CP/M produces cleaner results on all display

functions.
The disadvantages of LC against one of the CP/M versions are quite numerous however. First and foremost is that you have no true linking, relocating machine code capability. A library facility is provided and is quite usable but it is not in the form UNIX or CP/M users know it. Relocatable modules are not produced. LC relies entirely on the file include facilities of the EDAS assembler provided with the package. A library of ASM files can be prepared in assembler source form This is then linked to the compiled C code within another ASM file which is provided. The whole lot is then assembled by EDAS to produce a runable command file. If you are going to use your library on your own model 4 this is no disadvantage but if you want to share your library or use someone elses functions you must do it at the source code level. In C of course the ASM files you would be using would probably have been produced automatically by the C compiler from your C coded functions.
The real disadvantage of LC is its speed of assembly and linking. Typically, for a small program calling about ten functions from the libraries I can not only light my pipe while it is going on but I can smoke it and make and drink a cup of tea and sometimes it is still not finished !. Alright, thats laying it on a bit but for identical programs LC takes about ten times longer than for C/80 or Eco-C. I should point out that as the programs get longer the difference gets less but as most of the work with C is in compiling and assembling small functions the time factor is important.
A further minus for LC is the lack of some important features. Those refered to for C/80 and Eco-C are still missing from LC but two of the important basic features, structures and unions are also absent from LC. This can cause the need to write considerably longer code to compensate for the omissions. Also the beginner will not be able to experiment directly with the more advanced examples in the text books.
The documentation provided with LC, although quite comprehensive must get a further minus from me. It is not always as clear as it should be. On the subject of making libraries (which is a feature of C.) and including them it can be particularly confusing for the newcomer. I had been using the CP/M C,s for about nine months but had considerable problems. I found it more confusing than the limited information provided with C/80. For example, LC provides no step by step procedure on how to include or get a user constructed library permanently included during assembly. Also the procedure for producing a library is confusing due to the omission of tabs which must be included in the example but are not shown. If you know EDAS you may not suffer these problems but many, like me may be new to EDAS. After you have found how to modify the ASM file to include your library you may find you can not rewrite the necessary ASM file back to the disk as it needs the password to allow overwriting the file. It is necessary to rewrite the file as the filename must be unchanged in order to work with the JCL provided. The password is provided but it is hidden in a text file provided with the system. This caused me some delay in getting the full system up and running. The information is there and you may well not fall into the trap I did but that information should really be all together in one place, in the manual.
The use of Pro Pads, a form of compressed library of files or partitioned data set is recommended by Misosys and makes for more compact libraries which can be searched for functions to be

included. This costs about a further £30. Its use is not essential but with the rather poor library facility of LC it helps to offset some of the disadvantages of the system. For some strange reason the manual is clearer on using Pro-Pads than about using the inbuilt library facility.
Finally some plus,s for LC. The graphics functions provided allow use of TRS80 graphics, with the memory mapped screen access capability this can make for some nice fast, screen displays. As TRS80 graphics are non portable anyway some may question the advantage of this. The run time code is reasonably fast. The EDAS assembler is a good absolute assembler with full macro facilities but is not as useful as a true relocatable assembler like M80.
EDAS does include an editor like the old EDTASM. It is a line editor as opposed to the full screen editors like Wordstar but this may be considered an advantage if you are used to editing in BASIC. It is used to enter your C source program or for writing in assembler.
LC costs about £130 and can be considered a complete package. The Pro Pads library program can be considered an optional extra which allows for more compact libraries which can be added to in any order and can be quickly searched by the assembler.

My opinion of the C,s discussed, based only on my limited experience is that if you have CP/M and an acceptable text editor and do not need TRS80 graphics then C/80 is the most cost effective way to experience C on the TRS80. If you need Z80 code and have no CP/M I hope you enjoy making cups of tea. You will probably get into the habit of doing something while you wait for your runnable /CMD program to emerge from the frantic disk activity which goes on while LC is assembling and searching libraries. Errors at assembly time can make you grow old quickly with LC. What price a 256K Ramdisk add-on ?.
An identical, integer only program compiled using C/80 produced 3k of code. The identical program done with LC produced 6k of code.

Would any professional user of C like to add or contradict my views ?. That is what we need to get a better perspective of implementations of a language like C.

For a good summary of the CP/M implementations of C get Grey Matters handout on the C,s they sell. These cover the specifications of C/80, Aztec C, BDS C, Q/C and Eco-C and are quite informative.

ON ANOTHER SUBJECT,
some of you may remember I asked for information on directly accessing the Model 4 keyboard and video about six months ago. Nothing was forthcoming so I worked it out for myself. I had it written up for the newsletter but blow me,- 80 Micro went and published almost the identical information in their September issue and for me to have sent mine in it would have looked as if I had made a direct cribb of the 80 Micro article so I didnt send it. Now if you want the info you will have to read 80 Micro.

Roy Barber (Bexleyheath, 01-304-3856)

(* You CAN use Z80 code with C/80! Just put .Z80 at the beginning of of the assembler section, and .8080 at the end. This only works if you are using M80, however, for assembling the compiler output. LFH *)

FOR SALE AND WANTED

* * * FOR SALE * * *

Hardware

Model I keyboard, expansion interface with RS-232, 48K RAM, tape recorder and cables, monitor, two drives in Blandford box,

Software

Newdos-80 original disk and documentation, Microsoft FORTRAN original disk and documentation

Books

Microsoft BASIC Decoded, TRS-80 Disk and Other Mysteries, TRS-80 Technical Reference Handbook, BASIC Computer Games (TRS-80), Machine Language Programming From The Ground Up (Howe), TRS-80 Assembly Language Programming (Barden).

Documentation

RS-232 board, Editor/Assembler, MON3 and MON4, TRS-80 Pascal Version 5.0.

Also 48 disks, five unused, with lots of game programs.

I am prepared to sell all the above for £450 o.n.o. I am also prepared to sell bits and pieces at proportional prices.

Please note that I will be away during January, home in February, away in March, and at home after that.

David Epstein,
23 Inchbrook Road,
Kenilworth,
Warwickshire CV8 2EW.
Tel: (0926) 53955 (before 10 p.m.)

MODS TO PETER HALL'S MODEM

The filter section can be made more stable by placing a 220pF ceramic capacitor across R50. This might not be needed in every case, but it is probably wise to include it anyway.

The V23 version (Prestel 1200/75 baud) is now running, but giving this a really good dynamic check on a BB proved to be a problem, as most are 300 baud only, but I did manage to access CBBS and PSS on 1200/75 baud.

The mods are as follows:-

1) Filter values

R45 = 15K	R47 = 5K1	R49 = 6K8
R46 = 1K8	R48 = 3K3	R50 = 1K5//1K5
R51 = 180K//390K		R52 = 68K//100K
R53 = 56K		

R54, R55, R56 = short circuit and TR8, 9, 10 can be left out.

2) XR2211 Bandwidth

C11 = 4700pF C12 = 2200pF R22 = 68K

Again, TR3 collector and emitter can be shorted with a link, and R24, VR7, R25 and ANS/ORIG switch can be omitted.

3) Transmitter

R32 = 67K (68K//390K) giving 450Hz
R34 = 77K (100K//270K) giving 390Hz

Here also R33, VR3, TR7, R35, VR5, TR6, R36, R37, D2, D3 and the TEST/RUN switch may be left out.

A note on Receive calibration.

VR6 is set for 1750Hz (it should be 1700Hz) but with the increased bandwidth of V23 it should not matter. This was to keep the same Rx set-up program.

Peter Hall,
29 Clyde Road,
Brighton,
Sussex BN1 4NN.

RE-INKING PRINTER RIBBONS

Normal stamp-pad ink is water-based, and will cause pins on a DM printer to rust, even with WD-40, though that should delay things somewhat. I use stamp-pad ink designed for use with metal stamps which is oil-based and quite immiscible with water. Using this ink I don't use WD-40 at all. The one I use is a REXEL product and has "For metal numbering machines, daters, and price markers" printed on the side. Also "Shake well"!

By the way, have you heard of anybody using a DM printer to cut stencils with? If so, how does the print head stand up to it? I'm reluctant to experiment with new print heads at £30+.

Phil Quatermain,
138 Three Bridges Road,
Crawley,
West Sussex.

(* Brian used to cut BANDA stencils with his DM printer, and it didn't seem to affect it adversely. I feel that a daisy-wheel would be better if you can lay your hands on one, both from a reliability point of view, and because of the improved quality. LFH *)

An Update from Peter Knaggs on ILLEGAL COMMANDS

I have recently received the ZILOG data book for 1983/84 (They seem to be giving them away).

In there I looked up the Z800 processor (all 70 Pages of it). To my amazement I have found the Zilog have announced that some the the illegal Z80 commands are now legal for the Z800, namely the Half-index register accessing.

These are new nemonics published by Zilog:

In my article	Zilog Z800 spec.
LD HX, n or r	LD IXH, n or r
LD LX, --''--	LD IXL, --''--
LD HY, --''--	LD IYH, --''--
LD LY, --''--	LD IYL, --''--

If you want more information Zilog can be found at:- and I can be found at:

Zilog (UK) Limited,
Zilog House,
Moorbridge Road,
MAIDENHEAD,
Berkshire
SL6 8PL
(0628) 39200

P. Knaggs,
12 Seymour Road,
CHIPPENHAM,
Wiltshire.
SN15 3NH.
(0249) 654940

Peter Knaggs

Windows on the TRS-80 ????

I have been asked by Mister Brian Pain to write some windowing software for the TRS-80. The object being to beat all these newfangled machines back a bit. No, the real object being to write a BASIC then a Machine Code version for another of my fabulous lectures on how to fool the machine from code.

First: The LOGIC.

When you send a character to be printed it will be placed onto the screen via a POKE command, this is done so as not to confuse the Poor ROM's cursor (Never intended for this kind of thing) After the character has been placed onto the screen then the cursor needs to be moved on a character (So as not to overtype all the time). If the cursor has NOT reached the last column all the work is done so return from the routine. Otherwise, it is moved to the start column (Note - Column and not line as they are not necessarily the same) and then moved down to the next line.

If on the event of that being the last line then the cursor is either reset to the top of the window or the window is scrolled, dependent of the type of window it is.

The Scroll:

1. Find the start of the window.
2. Then copy the text from the last line to this line.
3. Now move down a line.
4. Repete until all lines copied up (Except first line)
5. Now blank out the last line.
6. Reset cursor position to start column of last line
7. END

Second: The BASIC Program.

Here I use the variable SC to tell the system which screen to use. The routines need various information about the screens; this is stored in the array SC(x,y) where x is the screen number and y is the data item.
The contents of the array are:
The start cursor location in Line/Column.
The end cursor location in Line/Column.
The Current cursor loc. in Line/Column.
The Screen type:-0 - Off
1 - No scroll/Overwrite
2 - Scroll

Lines 10 - 70: Clear memory and set up five screens.
Lines 110 - 180: Small Program to Print characters on screen 0 and then on screen 1
Lines 65100 - 65170: Routine to Print text in TX$ onto screen SC.
Lines 65210 - 65270: Display the character and update cursor location.
Lines 65310 - 65360: Scroll the current screen contents.

Lines 65370 - 65430: Blank out last line of current screen. and reset cursor location.
Lines 65480 - 65520: Screen information data.

Peter Knaggs 0249-654940.

```
1 'WINDOWS in BASIC by Peter Knaggs at the request of Brian Pain
2 'discussed at the Swindon Workshop November 1984
3 '*********************************************
5 CLS
10 CLEAR MEM/2              'Clear out the memory
20 DIM SC(4,6)              'Assigne the window array
25       ' Assigne data to window information
30 FOR X = 0 TO 4           'Loop for 5 windows (0-4)
40    FOR Y = 0 TO 6        '  Loop for 7 Data bytes (0-6)
50       READ SC(X,Y)       '    Read data byte into array
60    NEXT Y                '  Next Data byte
70 NEXT X                   'Next Window
80 '
90 ' This is a small test/Demo Program
100 ' ''''''''''''''''''''''''''''''''''''
110 FOR T = 32 TO 191       'Loop for Printable characters
120    A$ = CHR$(T)         ' Set Character
130    SC = 0               ' Set to window 0
140    GOSUB 65210          ' Display on the window
150    SC = 1               ' Select window 1
160    GOSUB 65210          ' Display on the window
170 NEXT                    'Next character
180 GOTO 110                'Repeat until break Pressed.
190 '
200 '
210 '
65100 ' Display string TX$ on window SC.
65110 ' ''''''''''''''''''''''''''''''''''''
65120 IF SC(SC,6) = 0 THEN RETURN  ' Return if window off
65130 FOR XX = 1 TO LEN(TX$)       ' Loop for string lenght
65140     A$ = MID$(TX$,XX,1)      '  Get charater from string
65150     GOSUB 65210              '  Display the character
65160 NEXT                         ' Get next character
65170 RETURN                       '  Return to caller
```

```
65180 '
65190 'Display character in A$ onto screen SC
65200 '*************************************
65210 POKE 15360 + SC(SC,4) * 64 + SC(SC,5), ASC(A$)    'Poke A$ on screen
65220 SC(SC,5) = SC(SC,5) + 1                           'Move cusror on
65230 IF SC(SC,5) < SC(SC,3) THEN RETURN                'Is it at end of line ?
65235                                                   'No - Return.
65240 SC(SC,5) = SC(SC,1)                               'Reset line position
65250 SC(SC,4) = SC(SC,4) + 1                           'Move next line down
65260 IF SC(SC,4) <= SC(SC,2) THEN RETURN               'Return if on screen
65270 IF SC(SC,6) = 1 THEN SC(SC,4) = SC(SC,0): RETURN 'If Screen type 1,
65275                                                   ' Re-set line No.
65280 '
65290 'Scroll the text in the window
65300 '*****************************
65310 FOR X = SC(SC,0) TO SC(SC,2)                      'Loop for # lines - 1
65320    PO = (X * 64) + 15360                          ' Set POKE address
65330    FOR Y = SC(SC,1) TO SC(SC,3)                   ' Loop for # characters
65340       POKE PO + Y, PEEK(PO + Y + 64)               '  Move chr up a line
65350    NEXT                                           ' For all of line
65360 NEXT                                              'For all -1 lines
65363 '
65365 ' Blank out last line
65367 '********************
65370 PO = SC(SC,2) * 64 + 15360                        'Set addr to last line
65380 FOR X = SC(SC,1) TO SC(SC,3)                      'Loop for # characters
65390    POKE PO + X, 32                                '  Make chr into a space
65400 NEXT                                              'For all of line
65410 SC(SC,4) = SC(SC,2)                               'Set current line
65420 SC(SC,5) = SC(SC,1)                               'Set current character
65430 RETURN                                            'Return to caller
65440 '
65450 '
65460 'SCREEN DATA
65470 '***********
65471 ' DATA format :-
65472 'Start line #, Start column #, Stop line #, Stop column #,
65473 'Current line #, Current colum #, Screen type
65475 'Screen type: 0 - Screen off, 1 - Non-scroll screen, 2 - Scroll screen
65477 '
65480 DATA 10,  1, 15, 30, 10,  1, 1    :'Screen 0
65490 DATA  5, 31, 10, 60,  5, 31, 2    :'Screen 1
65500 DATA  0,  0, 10, 64,  5, 16, 2    :'Screen 2
65510 DATA 15,  0, 15, 64, 15, 22, 0    :'Screen 3
65520 DATA  0,  0,  0,  0,  0,  0, 0    :'Screen 4
```

Windows on the TRS-80 (Part II)
"""""""""""""""""""""""""""""""

Third: The Machine code Program.

This is simply a machine code version of the BASIC Program presented in the last article. I decided to use the Index register as a base to the screen information, there by allowing the Programmer to Place the information at any suitable location.

In this Program the data is stored at the start.

Lines 15 - 19: Screen data information.
Lines 27 - 43: Test Program (Same as BASIC).
Lines 50 - 58: Print string of text.
Lines 66 - 96: Display the character and update cursor location.
Lines 102 - 127: Scroll the current screen contents.
Lines 130 - 145: Blank out last line of current screen.
Lines 154 - 165: Calculate Peek/Poke Position of line A.

Fourth: Machine code on the Model 4.

This I have written by reference to the Model 4 Program reference manual and no contact whith a Model 4 It is there and of course not guaranteed to work but I see no reason why it should not.

The Program itself is exactly the same as the Model I/III version in function and method of function. The one major difference is that the Model 4 works in Rows and Columns instead of memory locations, so I do not have to calculate the PEEK/POKE address of the screen.

Lines 9 - 14: SVC vector EQUates.
Lines 24 - 28: Screen data information.
Lines 30 - 37: Subroutines used by software.
Lines 41 - 67: Test Program (Same as BASIC).
Lines 74 - 82: Print string of text.
Lines 90 - 125: Display the character and update cursor location.
Lines 129 - 154: Scroll the current screen contents.
Lines 157 - 174: Blank out last line of current screen.

=====> The Programs <=====
-> Model I/III machine code.

```
 1      ; WINDOWS/M13
 2      ; Screen windowing routines.
 3      ; By : Peter J Knaggs
 4      ; For the TRS-80 Model I/III computers.
 5      ORG 0A000H ; Some where to put it
 6      LOAD $ ; Load it into memory there as well
 7      ;
 8      ; Screen data
 9      ; """""""""""
10      ; Date format :-
11      ; Start line, Start Column, End line, End column
```

```
12      ; Current line, Current column, Screen type
13      ;  Screen types => 0  Screen  off,  1  Screen overwrite, 2
Screen scroll
14      ;
15 SCREEN0 DB 00,00,15,30,00,00,02 ; Screen 0
16 SCREEN1 DB 05,30,10,60,05,31,01 ; Screen 1
17 SCREEN2 DB 00,00,10,64,05,16,02 ; Screen 2
18 SCREEN3 DB 15,00,15,64,15,22,00 ; Screen 3
19 SCREEN4 DB 00,00,00,00,00,00,00 ; Screen 4
20      ;
21 DOS'EXIT JP 402DH ; Return to DOS
22 BREAK EQU 14400 ; Keyboard matrix for Break key
23 CLS EQU 1C9H ; Clear screen routine
24 VDCHAR EQU 33H ; Print character on screen routine
25      ;
26      EXEC $ ; Set execution address
27      CALL CLS ; Clear the screen
28      LD A,15 ; Turn cursor off character
29      CALL VDCHAR ; Print it on the screen
30      LD HL,BREAK ; Point to keyboard matrix
31      ;
32 START LD A," " ; Set lowest character
33      LD B,191-32 ; Set counter (No. Printable characters)
34 LOOP LD IX,SCREEN0 ; Point to Screen 0's information
35      CALL DISP'CHR ; Print the character on screen 0
36      LD IX,SCREEN1 ; Point t screen 1's information
37      CALL DISP'CHR ; Print the character on screen 1
38      BIT 2,(HL) ; Is the BREAK key Pressed ?
39      JR NZ,DOS'EXIT ; Yes - return to DOS
40      INC A ; No - Move on to next character
41      DJNZ LOOP ; Repeat for all characters
42      ;
43      JR START ; Go round again (Until BREAK Pressed)
44      ;
45      ;
46      ;DISP'STRING
47      ; HL => String to be Printed, Ending with an X'00'
48      ; IX => Screen Data information
49      ;
50 DISP'STRING LD A,(IX+6) ; Get screen type
51      OR A ; Is screen on ?
52      RET Z ; No - Return to caller
53 DISP'S LD A,(HL) ; Get character
54      INC HL ; Move Pointer on
55      OR A ; Is it End of Text flag ?
56      RET Z ; Yes - Return to caller
57      CALL DISP'CHR ; No - Display the character
58      JR DISP'S ; Repeat until end of text (0) reached
59      ;
60      ;
61      ;DISP'CHR
62      ; A  =  Character to Print
63      ; IX => Screen Data information
64      ; AF' BC' DE' HL' destroyed
65      ;
```

```
 66 DISP'CHR PUSH AF ; Save Print character
 67     EXX ; Save all other registers
 68     LD A,(IX+4) ; Get Current line
 69     CALL DISP'CALC ; Calculate PEEK/POKE location
 70     LD A,(IX+5) ; Get current column
 71     LD E,A ; Add it to PEEK/POKE location
 72     ADD HL,DE
 73     POP AF ; Retrieve Print character
 74     LD (HL),A ; Put the character onto the screen
 75     EX AF,AF ; Save the character
 76     LD A,(IX+5) ; Get the current column position
 77     INC A ; Add one to it
 78     LD (IX+5),A ; Put it back
 79     CP (IX+3) ; Is it over limit ?
 80     JP C,DISP'EXIT ; No - Return to caller
 81     LD A,(IX+1) ; Get Start column
 82     LD (IX+5),A ; Reset current column
 83     ;
 84     LD A,(IX+4) ; Get current line
 85     INC A ; Add one to it
 86     LD (IX+4),A ; Put it back
 87     DEC A ; take it back again
 88     CP (IX+2) ; is it over limit ?
 89     JP C,DISP'EXIT ; No - Return to caller
 90     LD A,(IX+6) ; Yes - Get screen type
 91     CP 2 ; Is it a scroll screen ?
 92     JR Z,DISP'SCROLL ; Yes - goto scroll routine
 93     LD A,(IX+0) ; No - Get start column
 94     LD (IX+4),A ; Reset current column
 95     ;
 96 DISP'EXIT EX AF,AF ; Retrieve Print character
 97     EXX ; Retrieve user registers
 98     RET ; Return to caller
 99     ;
100     ;
101     ;
102 DISP'SCROLL LD A,(IX+0) ; Get Start line
103     LD (DISP'LOOP+1),A ; Set line value
104 DISP'LOOP LD A,0 ; Get line value
105     CP (IX+2) ; Is it end of scroll
106     JR NC,DISP'BLANK ; Yes - Blank out last line
107     CALL DISP'CALC ; Calculate PEEK/POKE location
108     LD A,(IX+1) ; Get start column
109     LD E,A ; add it to PEEK/POKE location
110     ADD HL,DE
111     LD A,(IX+3) ; Get end column
112     SUB (IX+1) ; Subtract start column from it
113     LD B,A ; Set counter to result
114     ;
115 D'LOOP PUSH HL ; Save pointer
116     LD DE,64 ; Add 64 to pointer
117     ADD HL,DE ; (Move pointer to next line)
118     LD A,(HL) ; Get the character
119     POP HL ; Retrieve the pointer
120     LD (HL),A ; Set the new character
```

```
121      INC HL ; Move pointer on
122      DJNZ D!LOOP ; Repeat for all of line
123      ;
124      LD A,(DISP!LOOP+1) ; Get line value
125      INC A ; Add one to it
126      LD (DISP!LOOP+1),A ; Put it back
127      JR DISP!LOOP ; Repeat for all lines
128      ;
129      ;
130 DISP!BLANK LD A,(IX+2) ; Get End line
131      CALL DISP!CALC ; Calculate PEEK/POKE location
132      LD A,(IX+1) ; Get start column
133      LD E,A ; add it to pointer
134      ADD HL,DE
135      LD A,(IX+3) ; Get End column
136      SUB (IX+1) ; Subtract Start column
137      LD B,A ; Put result into counter
138 DISP!BL:LD (HL),32 ; Put space on screen
139      INC HL ; Move to next character
140      DJNZ DISP!BL ; Repeat for all of line
141      LD A,(IX+2) ; Get End line
142      LD (IX+4),A ; Set current line
143      LD A,(IX+1) ; Get Start column
144      LD (IX+5),A ; Set current column
145      JR DISP!EXIT ; Return to caller
146      ;
147      ;
148      ; DISP!CALC:
149      ; A  = Value to Multiply by 64
150      ; Exit:
151      ; D  = 0
152      ; B  = 0
153      ; HL = A * 64 + 15360 (Start of screen)
154 DISP!CALC:
155      LD H,0 ; Zero out H
156      LD L,A ; Place A into HL
157      PUSH HL ; Copy it over to DE
158      POP DE
159      LD B,63 ; Set counter
160 DISP!L:ADD HL,DE ; Add HL to DE
161      DJNZ DISP!L ; Repeat 64 times (Inc 0)
162      LD DE,3C00H ; Point ot start of screen
163      ADD HL,DE ; Add it to pointer
164      LD D,0 ; Zero out D
165      RET ; Return from routine
166      ;
167      ;
168      END
```

->Model 4 machine code.

```
1      ; WINDOWS/M4
2      ; Screen windowing routines
3      ; By : Peter J Knaggs
```

```
 4      ; For the TRS-80 Model IV (4) computer.
 5      ORG 0A000H ; Some where to put it
 6      LOAD $ ; Load it into memory there as well
 7      ;
 8      ; Equates
 9 @FLAGS EQU 101 ; Point IY to system flags
10 @EXIT EQU 22 ; Exit back to TRSDOS Ready
11 @VDCTL EQU 15 ; Send video control byte.
12 @DSP EQU 2 ; Display character at cursor.
13 @KEY EQU 1 ; Wait for key press from *KI
14 @ERROR EQU 26 ; Post Error message.
15      ;
16      ; Screen data
17      ; ************
18      ; Date format :-
19      ; Start line, Start Column, End line, End column
20      ; Current line, Current column, Screen type
21      ; Screen types => 0 Screen off, 1 Screen overwrite,
22      ;                 2 Screen scroll.
23      ;
24 SCREEN0 DB 00,00,15,30,00,00,02 ; Screen 0
25 SCREEN1 DB 05,30,10,60,05,31,01 ; Screen 1
26 SCREEN2 DB 00,00,10,64,05,16,02 ; Screen 2
27 SCREEN3 DB 15,00,15,64,15,22,00 ; Screen 3
28 SCREEN4 DB 00,00,00,00,00,00,00 ; Screen 4
29      ;
30 DOS!EXIT LD A,@EXIT ; SVC Number
31      RST 28H ; Return to DOS
32      ;
33 ERROR LD C,A ; Put error number
34      RES 7,C ; Set Return to DOS flag
35      SET 6,C ; Only display Error message string
36      LD A,@ERROR ; SVC Number
37      RST 28H ; Display Error and return to DOS
38      ;
39      EXEC $ ; Set execution address
40      ;
41 CLS LD A,@DSP ; SVC Number
42      LD C,28 ; Character code (Start of screen)
43      RST 28H ; Print it
44      LD A,@DSP ; SVC Number
45      LD C,31 ; Char code (Clear to end of screen)
46      RST 28H ; Print it
47      ;
48      LD A,@DSP ; SVC Number
49      LD C,15 ; Char code (cursor off)
50      RST 28H ; Print it
51      ;
52      LD A,@FLAGS ; SVC Number
53      RST 28H ; Set IY to flags
54      RES 0,(IY+10) ; Reset BREAK Pressed flag
55      ;
56 START LD A," " ; Set lowest character
57      LD B,191-32 ; Set counter (No. Printable characters)
58 LOOP LD IX,SCREEN0 ; Point to Screen 0's information
```

```
59      CALL DISP!CHR ; Print the character on screen 0
60      LD IX,SCREEN1 ; Point t screen 1's information
61      CALL DISP!CHR ; Print the character on screen 1
62      BIT 0,(IY+10) ; Is the BREAK key pressed ?
63      JR NZ,DOS!EXIT ; Yes - return to DOS
64      INC A ; No - Move on to next character
65      DJNZ LOOP ; Repeat for all characters
66      ;
67      JR START ; Go round again (Until BREAK Pressed)
68      ;
69      ;
70      ;DISP!STRING
71      ; HL => String to be Printed, Ending wite an X'00'
72      ; IX => Screen Data information.
73      ;
74 DISP!STRING:LD A,(IX+6) ; Get screen type
75      OR A ; Is screen on ?
76      RET Z ; No - Return to caller
77 DISP!S:LD A,(HL) ; Get character
78      INC HL ; Move Pointer on
79      OR A ; Is it End of Text flag ?
80      RET Z ; Yes - Return to caller
81      CALL DISP!CHR ; No - Display the character
82      JR DISP!S ; Repeat until end of text (0) reached
83      ;
84      ;
85      ;DISP!CHR
86      ; A  =  Character to Print
87      ; IX => Screen Data information
88      ; AF' BC' DE' HL' destroyed
89      ;
90 DISP!CHR:EXX ; Save all other registers
91      PUSH AF ; Save Print character
92      LD A,(IX+4) ; Get Current line
93      LD H,A ; Set cursor
94      LD A,(IX+5) ; Get current column
95      LD L,A ; set cursor
96      POP AF ; Retrieve Print character
97      LD C,A ; Set Print character
98      LD A,@VDCTL ; SVC Number
99      LD B,2 ; VDCTL function number
100     RST 28H ; Display character
101     JP NZ,ERROR ; Go - If error
102     EX AF,AF ; Save the character
103     LD A,L ; Get column
104     INC A ; Add one to it
105     LD (IX+5),A ; Set current column
106     CP (IX+3) ; Is it over limit ?
107     JP C,DISP!EXIT ; No - Return to caller
108     LD A,(IX+1) ; Get Start column
109     LD (IX+5),A ; Reset current column
110     ;
111     LD A,H ; Get line
112     INC A ; Add one to it
113     LD (IX+4),A ; Set current line
```

```
114      DEC A ; take it back again
115      CP (IX+2) ; is it over limit ?
116      JP C,DISP!EXIT ; No - Return to caller
117      LD A,(IX+6) ; Yes - Get screen type
118      CP 2 ; Is it a scroll screen ?
119      JR Z,DISP!SCROLL ; Yes - goto scroll routine
120      LD A,(IX+0) ; No - Get start column
121      LD (IX+4),A ; Reset current column
122      ;
123 DISP!EXIT:EX AF,AF ; Retrieve print character
124      EXX ; Retrieve user registers
125      RET ; Return to caller
126      ;
127      ;
128      ;
129 DISP!SCROLL:LD A,(IX+0) ; Get Start line
130      LD H,A ; Set cursor
131 DISP!LOOP:LD A,H ; Get line value
132      CP (IX+2) ; Is it end of scroll
133      JR NC,DISP!BLANK ; Yes - Blank out last line
134      LD A,(IX+1) ; Get start column
135      LD L,A ; Set cursor
136      LD A,(IX+3) ; Get end column
137      SUB L ; Subtract start column from it
138      LD B,A ; Set counter to result
139      ;
140 D!LOOP:INC H ; Move pointer to next line
141      LD A,@VDCTL ; SVC Number
142      LD B,1 ; VDCTL control function
143      RST 28H ; Get the character from the screen
144      JP NZ,ERROR ; Go - if error
145      DEC H ; Retrieve the pointer
146      LD B,2 ; VDTCL control function
147      LD C,A ; Set character
148      LD A,@VDCTL ; SVC Number
149      RST 28H ; Put the character onto the screen
150      INC L ; Move to next column
151      DJNZ D!LOOP ; Repeat for all of line
152      ;
153      INC H ; Move to next line
154      JR DISP!LOOP ; Repeat for all lines
155      ;
156      ;
157 DISP!BLANK:LD A,(IX+2) ; Get End line
158      LD H,A ; Set cursor
159      LD A,(IX+1) ; Get start column
160      LD L,A ; Set cursor
161      LD A,(IX+3) ; Get End column
162      SUB L ; Subtract Start column
163      LD B,A ; Put result into counter
164 DISP!BL:LD BC,0220H ; Set character and function
165      LD A,@VDCTL ; SVC Number
166      RST 28H ; Display space of screen
167      JP NZ,ERROR ; Go - if error
168      INC L ; Move cursor on
169      DJNZ DISP!BL ; Repeat for all of line
170      LD A,(IX+2) ; Get End line
171      LD (IX+4),A ; Set current line
172      LD A,(IX+1) ; Get Start column
173      LD (IX+5),A ; Set current column
174      JR DISP!EXIT ; Return to caller
175      ;
176      END
```

Peter Knaggs