

# INDEX



## INSTRUCTIONS

File the index page from every issue at the back of your Fast Access binder for a handy reference to your library of programs. The pages can be ordered by Icon or by issue, it's your choice.

## 80 TRACK DRIVES

Insert FAST ACCESS into your disk drive and type:

\*CHANGE <RETURN>

The program will prompt you to insert a formatted 80 track disk when necessary. Single drive users will have to swap the disk several times.

TITLE	PROGRAM FILES	TEXTFILES
TERMINAL	TERMNAL	T.TERMINA
COMPANDER	TERM/V6	
TOPOLOGIKA	COMPAND	T.COMPAND
	TOPDEMO	
	AVON	
DOUBLE FONT	STOCK	
	DFONT	
GALLERY	DOUBLE	
	GALLERY	
	GALSCR3	
	COMP	
GAMEPLAN		T.GAMEPLN
INVENTORY	INVENT	
SPACE ATTACK	SPACEAT	
	ATTACK	
	ATCODE	
	SPRDATA	
TWO BY TWO	TWOBTWO	
CELLS	CELLS	T.CELLS
SPEEKIT	SPEEKIT	T.SPEEKIT
	SPEECH!	
ADDICT'S ANTHEM	ADDICTF	
PERPLEXITY AD	PERPLEX	
	PERP	

## ALL USERS

Make backup copies of both disks and keep the originals in a safe place with Write-Protect tabs on. Use only the copies, as many of the programs write to the disk, which will diminish the usefulness of the originals. For specific filing system information, please refer to the help file on disk.

## NEW USERS

Don't Panic!. First find out whether you have 40 or 80 track drive(s) attached to your computer. Then go to your User guide or Welcome Manual and find out how to use the \*COPY command. Next re-read the section above All Users, and then continue reading down from this point.

## 40 TRACK DRIVE SYSTEMS

FAST ACCESS is supplied on 40 track disks and will work on any 40 track BBC Micro system straight away. Remember to make a working copy before use.

## 80 TRACK DRIVES

If your filing system allows double-stepping, we recommend using the system's own command. As a general rule, built-in 40-to-80 track converters should be used where available; the documentation for your filing system or utility ROM will give full instructions.

## ADDRESSES

If for any reason your copy of Fast Access will not work on your system then please carefully re-read the instructions given above. If you still experience problems then return it to: INFONET LTD, 5 River Park Estate, Billet Lane Berkhamsted, Herts HP4 1HL. TEL 0442 876661

Advertising enquiries to FAST ACCESS, Argus House, Boundary Way, Hemel Hempstead, HP2 7ST. TEL 0442 66551.

Editorial and technical enquiries to FAST ACCESS, 6C Belgic Square, Padholme Road, Peterborough PE1 IXF TEL 0733 53355.

Contributions should include full source code and instructions file on disk. Payments are extremely competitive.

The contents of this publication including all articles, designs, plans, drawings and programs and all copyright and other intellectual property rights therein belong to Argus Specialist Publications Ltd. All rights conferred by the Law of Copyright and other intellectual property rights and by virtue of international copyright conventions are specifically reserved to Argus Specialist Publications Ltd. Any reproduction requires the prior written consent of Argus Specialist Publications Ltd.

Typesetting and artwork by Simon Fifield.  
Printed by Loxleys, Sheffield.

Duplication by Direct Disk Supplies, Teddington, Middlesex.

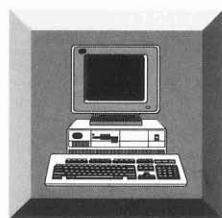
©Argus Specialist Publications Ltd 1989

## RESOURCE FILES

TITLE	FILES	TYPE
Menu	FAMENU	BASIC
	GOSPR	Data file
	GOCODE	M/ code
MenuFix	MEND	BASIC
	A.FAMENU	BASIC
	B.FAMENU	BASIC
Info Desk	IDESK	BASIC
Change	CHANGE	M/code
Help	T.HELP	Text file



# RS423 ASCII TERMINAL



## AUTHOR

S.L. DICKINSON

## FILES

'TERMNAL' ..... BASIC  
'TERM/V6' ..... BASIC  
'T.TERMINA' ..... TEXT

This program is designed to provide a flexible means of accessing on-line databases and bulletin boards which use the scrolling ASCII (as opposed to the viewdata) format. The software can also be used for direct file transfers between two computers using the RS423 port. The program allows the user to select options from a main menu. These include receive & transmit baud rates, word characteristics (data, parity and stop bits), choice of screen mode, logging directly to a disk file (limited only by disk capacity), logging to printer, error trapping and monitor echo. A

file transfer option is also provided which allows previously prepared messages or data to be transferred directly from disk.

## BAUD RATE.

Both receive and transmit baud rates are selectable between 75 and 19200 baud. Most modem users however, will be restricted to 300/300, 1200/75 or 1200/1200.

## WORD CHARACTERISTICS.

Selecting word type before contacting a database is very important. because an incorrect choice will provide nothing but garbled characters. There is little consistency in the use of word types by major host systems. For example, VIEWDATA systems use 7 data, even parity & 1 stop bit whereas DISTEL uses 8 data and 2 stop bits. Although the selection need only be made once (as with baud rates) the

program will allow the user to change protocol while still holding the line open.

### LOGGING TO FILE

The purpose of this function is to store the entire record of a communication with a database. The program operates much faster than a modem, so the time spent paying for the telephone call should be kept to an absolute minimum.

### LOGGING TO PRINTER.

Data can also be 'logged' directly to a printer. Users must insert their own printer handling routines in the relevant procedure. It should not need stating that only parallel printers can be used. For those with serial printers, hard copy can easily be made whilst off-line from a disk file.

### GOING ON-LINE.

'Duplex' communications are those which occur in two directions. When data can travel in both directions simultaneously, the connection is referred to as full-duplex. If transmissions can only be made by either terminal alternately, the connection is half-duplex.

### FILE TRANSFER MODE.

File transfer mode differs from duplex mode in that it is designed to send or receive an entire file. Once communications have been opened, a file may be directly read from or to disk. This enables data or messages to be prepared off-line and checked before dialling up a terminal.

### ERROR TRAPPING

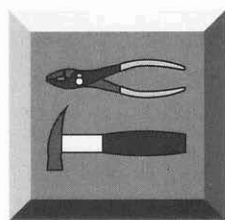
Three types of error are detected.

1. Framing error, indicates that a byte was incorrectly framed bystart and stop bits.
2. Parity error, odd or even parity bit does not correspond withthe data received.
3. Receiver overrun error, Characters have been received that have not been read from the 6850 ACIA registers.

For a more detailed explanation of the menu options please refer to the text file on disk B.



# TEXT COMPRESSION



## AUTHOR

ULTAN HENRY

## FILES

'COMPAND' ..... BASIC  
'CPACTER' ..... M/CODE  
'PCODE' ..... M/CODE  
'T.COMPAND' ..... TEXT

When writing adventure programs, or indeed any program involving large amounts of text, it soon becomes apparent that the BBC Micro has insufficient memory available to store the amount of data required. One of the simplest solutions to this problem is to compress or *squeeze* this data into a smaller section of the computer's memory. The most efficient methods of text compression are largely dependant upon the content of the text in question. For example, if the text contains a large proportion of numeric digits

(0,1,2,3,...etc.), each digit could be assigned its own 4-bit code so that two numerals may be stored in a single, 8-bit, byte.

Another possible method is to identify the 255 most common words in the text and code them into single bytes (1 to 255), with the zero byte being reserved to signify "output the next character as literal ASCII".

## SETTING UP

Since the method of compaction involves a large number of calculations, the program has had to be written in machine code to reduce the amount of running time needed. This also means that, although the source code is rather long, (and also heavily annotated), the machine code it produces is relatively short - allowing a larger amount of uncompactd text to be entered into the machine.

To generate the machine code, RUN the program from

the menu. This will generate two pieces of code, 'CPACTER' and 'PCODE', which should be saved on a separate disc when prompted.

## USING THE COMPACTER

To use the compacter program, type

**\*RUN CPACTER (or simply  
\*CPACTER)**

to load and execute the machine code.

You may now enter up to 255 strings, each containing up to 255 characters (or until the computer's memory is full!). If you wish to enter fewer strings, simply press <RETURN> to the prompt and the compaction process will begin. As you are entering the text, you should note the number assigned to each string, since you will need this later with the expander.

A **rough** indication will be given of the length of time required to analyse the text entered. If you have entered a large quantity of text, be prepared to wait!

When compression is complete, the program will output the text, through the expander, so that it may be checked for errors. The program will then

print (in decimal) the memory location immediately after the text. This number should be noted.

Finally, the compacted text is stored under the filename 'DATFILE' together with a copy of the expander code.

## USING THE COMPACTED TEXT

To use the compacted text, 'DATFILE' must be loaded into memory at location &1900 using the command:

**\*LOAD DATFILE 1900**

If your program is to be in BASIC, PAGE must be reset above the expander code by typing:

**PAGE = XXXXX + &100**

where XXXXX is the decimal value of the memory location above the code as given by the compacter.

To print a string of compacted text:

(i) from BASIC enter:

**X%=<number of string>**

**CALL &1900**

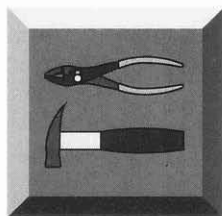
(ii) from machine code enter:

**LDX #<number of string>**

**JSR &1900**



# DOUBLE FONT



## AUTHOR

IVOR HEWITT JNR

## FILES

'DFONT' ..... BASIC  
'DOUBLE' ..... M/CODE

DFONT is a routine for those who wish to use the high resolution graphics capabilities of Mode 0 but also want to print textual information on the screen, which can be read on a normal T.V. set.

DFONT provides the user with the ability to display 40 columns of text in any of six thicknesses and two shades. The program is designed to work in Mode 0 but will also work in any other mode except Mode 7 in which case it switches itself off.

When run DFONT will create a machine code file DOUBLE which must be loaded every time DFONT is to

be used in a program. To incorporate DFONT in a program a number of procedures must be added to the end of the program, these procedures are included at the end of the DFONT program.

After DFONT has created the machine code file DOUBLE a short demo will run displaying the different shades and thicknesses that can be achieved and how the procedures can be easily incorporated into a program.

The program works by trapping the write character vector, checking to see if the VDU code is a character and if so displaying two pixels for every one in the character.

The fonts are turned on by PROCfonton and off by PROCfontoff

PROCfont(t,s) where t is the thickness(0-6) and s the shade(0-1) selects the desired font.



# FAST ACCESS MENU FIXER



Please accept our sincere apologies for the problems many of our subscribers had with the menu on ***FAST ACCESS issue 1***. The problem has been rectified from issue 2 onwards and all users should have no more problems at all with the ***FAST ACCESS*** menu.

On disk B of issue 3 you will find a program called 'MEND'. It is not available from the menu so you will have to type:

**CHAIN "MEND"**

**<RETURN>** to run it.

The program will tell you what to do to mend the faulty menus on **issue 1** of ***FAST ACCESS***. Please follow the instructions carefully and use only backups of the ***FAST ACCESS*** disks NOT the originals.

**TRACE**

Some of our subscribers

have had trouble using the TRACE utility. The problem is caused by the fact that the machine code is assembled into memory but not activated. To do this you need to run the TRACE program by selecting it from the menu. When the screen has become still type:

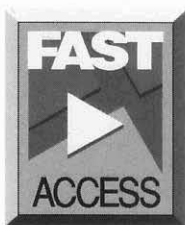
**CALL &900 <RETURN>**

This will activate the code and you can now use the **\*TRACE** command to trace the flow of a BASIC program.

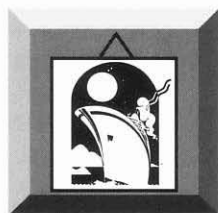
**SAFETY FIRST**

Before using any program you should **ALWAYS** make a **BACKUP** of the ***FAST ACCESS*** disks. Only use the backup disks not the originals. This will prevent you from damaging your original disks. If your copies become corrupted you can then make fresh copies from your original disks.





# GALLERY



## ARTIST

'STEVEN NISBET' ..... WITCH HUNT

## FILES

'GALLERY' ..... BASIC

'GALSCR3' ..... COMPRESSED

'COMP' ..... MACHINE CODE

This is the third in a series of digital pictures created on the BBC micro. Future issues of FAST ACCESS will feature work of equally high quality from other computer artists.

Steven Nisbet, the originator of our first three pictures creates his artwork on paper first, then reproduces them on the screen with his own custom paint program.

If you have drawn any screens of a similar quality do not hesitate to send your pictures on a clearly labelled disk to our editorial address given on the Index page. You should also include a text file on the disk giving as much detail as pos-

sible about your work. This should include sources of inspiration, techniques used, hardware and software used and so on. We are particularly looking for work which could be structured as a graphics series, or tutorial examples. Don't forget, we offer a cash payment for any picture published in FAST ACCESS.

## CLIP ART

We are also looking for clip art; that is small pictures or line drawings suitable for use in DTP packages. In this area, quantity and variety of subject matter rank equal in importance with quality. So if you have anything you think may fit the bill, then send that in too!



# DISK FORUM



Welcome to the third issue of FAST ACCESS magazine. This one we hope will work on all machines, and arrive in pristine condition. We welcome all the feedback from our readers whether praise or criticism. You see, without your opinions we could not have created this magazine in the first place!

## COMING SOON

Here are a few things likely to be in forthcoming issues:— Arcade game, working demos and a new series from Peter Scott

Powerful WIMP style database Commz— look out for a special modem offer with this sophisticated communications package Yukon gold, from the author of Cordelia

## READERS SUGGESTIONS

*...A painting/drawing program, a reasonable CAD, how about a spelling checker, and what about a debugging utility.. What about a database. The program TRACE does not work, although the BASIC program appears to.*

*Having tried the suggestion of loading FAMenu, then \*TRACE, the program hangs up with a flashing cursor.*

*F.T Crapper*

The problem with TRACE is that the machine code needs to be activated with CALL &900. The \*TRACE command can now be used. For more details please read the Help text file on disk B.

*May I put in a strong plea for a tutorial on the details of using the CASDATA files. I for one am not sufficiently au fait with the technicalities of tape to disk transfer. I am sure there are many of us with old tape software dying to get it onto disk. More sampled sounds please and a tutorial on MIDI please. Could you put FAST ACCESS on reversible disks?*

*John S. Aston*

Never use your master disks as working disks, and yes we get the message, lots more info on technical programs like CASDATA and TRACE



tising or duplicating expenses could cost you dear. *Editors note: Last year's best selling BBC game, Last Ninja written by Peter sold around 20,000 copies. Work that out for yourself!*

The software house may offer you an exclusive contract, to work for them and no-one else. They may jack up their royalties quite substantially to justify this. My advice is not to sign anything like this.

I know from experience this is not a wise move. The company may go bust, reject your games, change ownership, fail to publicise your games properly, or any one of a number of other things. What happens to you and your work then?

I had to give a year's notice before quitting my three year exclusive contract with ASL. I was lucky in that I could do work other than games writing, and had little time for games due to my university course, but you may not be. The situation over what happened when I left university was not clear in my contract, which I disliked.

After completing any modifications asked for, you'll probably be asked for an Electron version of your BBC game. If

you've never used an Electron before, tell the software house this. It shouldn't make much difference. The Electron, being simpler than a BBC, is correspondingly easier to learn to program.

I started out on an Electron, and therefore am somewhat biased towards the strange little machine. If you've written your game around the limits I've mentioned in the previous articles, you should find that an Electron version is a piece of cake!

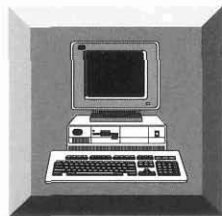
## CONCLUSION

I hope I've been of some help to all you budding programmers out there. I've been in your situation, as have all games programmers at one time or another, and it isn't an easy one. Keep plugging away and you'll succeed.

Once you become established, you'd be surprised how easy the whole business becomes. Except the ideas, that is. The programming side becomes less of a challenge, except you tend to try more ambitious ideas. The graphics and sound become more difficult to get just right. Did I say it became easier? Well, there again...



# INVENTORY



## AUTHOR

MICHAEL WILSON

## FILES

'INVENT' .....BASIC

Inventory is a straightforward database program with extensive on screen help, and so is particularly suitable for the novice user. Simply follow the instructions given to create your own lists of information.

Unlike some other databases which restrict you with pre-defined fields, Inventory allows you to control this when you create your database. Don't worry if you don't understand what some of the terms used mean, a few trial runs should make everything clear.

The strength of this program is it's flexibility, and example applications include business stock, names and addresses, collector's pieces, or anything

which might be recorded in an inventory. Inventory is also suitable for use in the classroom, helping children to understand databases and their uses.

The main menu displays all the options which include:

1. Load in file
2. Save file
3. Enter data
4. View records
5. Print records
6. Change fields
7. Quit
8. Information

When an option is selected details are given on what to do at each stage. This makes the program user friendly.

To run the program, select it from the FAST ACCESS menu. Once in memory and running it does not require FAST ACCESS disk A in the drive. However, there is no room on disk A for data files anyway, so remove disk A and replace with a blank formatted disk before loading or saving a file.



### Special FAST ACCESS Reader Offer: **SAVE £3!**

There's no mystery to "X" BASIC, a new addition to the familiar BBC BASIC inside your machine. Your current programs won't even know that you've upgraded your BASIC but any new programs written with XBASIC will benefit.

#### **Fully compatible with all 8-bit versions of BBC BASIC.**

XBASIC has been written by a professional programmer to improve BBC BASIC programming with faster routines, more modern structures and helpful utilities.

If you enjoy programming in BBC BASIC then you'll enjoy XBASIC. See the Article in Programming Plus

in the January issue of A&B Computing.

Order now and you'll receive:

- **16K ROM**
- **Examples Disk (specify format)**
- **Full Manual On Disk**

Order Product Number EB4 and specify 40 track or 80 track DFS disk. Price: £12.00 including postage and packing. (Overseas customers please add £1.00 for postage.) Send your order to: Readers Services, Argus House, Boundary Way, Hemel Hempstead, Herts, HP2 7ST. Visa and Access credit card orders can be taken on 0442 66551.

## LET US KNOW !

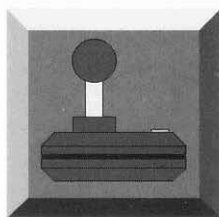
We would like to know what you think of your new disk magazine, so let us know about it.

- Is there a program which you think is excellent?
- Do you think that there should be more of a particular type of program?
- Do you need more information?
- Are having compatibility problems with a program?
- Do you have something to say about Fast Access in general?
- Do you have any bug fixes or improvements to published programs?

Write To: FAST ACCESS, 6C Belgic Square, Padholme Road,  
Peterborough. PE1 5XF.



# SPACE ATTACK



## AUTHOR

EWEN SETTI

## FILES

'SPACEAT' .....BASIC  
'ATTACK' .....BASIC  
'ATCODE' .....M/CODE  
'SPRDATA' .....DATA

An Enemy U.F.O. has entered the planets defence installation's power cells. The main worry is that it may damage the thin protective walls surrounding the unit.

You and another player control moveable shields, in which you must deflect the

intruder, and stop it from doing any serious damage.

If a player misses the UFO, the other player will receive a penalty point.

The 1st player to reach a score of ten points, wins the game.

Each player has 2 controls to move their shield up/down. Player one's shield is on the left hand side, and player two's shield is on the right hand side.

There is a choice of 8 different speeds, 1 is very slow, up to 8 which is quite fast. To start each game, press the speed you require.

## CONTROLS

### PLAYER 1

A ..... UP

Z ..... DOWN

S ..... SOUND ON

Q ..... SOUND OFF

Escape ..... ABORT GAME

### PLAYER 2

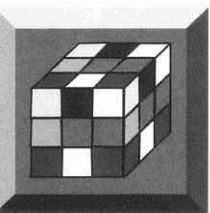
\*

?

S

Q

Escape



# TWO BY TWO



## AUTHOR

DES CATLIN

## FILES

'TWOBYTWO' ..... BASIC

Two By Two is another of Des' excellent graphic puzzles. The object of the game is to move the coloured boxes on the screen until all the boxes of the same colour are together in one

large square. To move a row of boxes simply press the number or letter which is shown at each end of a row.

There are three play options. The first, a practice mode where the completed puzzle is shown will help in getting used to how the boxes move. The next option is the easy one. Here four colours need to be sorted out. The last option is the hardest where five colours need to be sorted out.

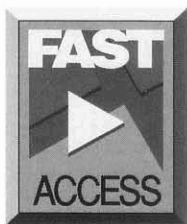
# WANTED

If you have written any *useful, good quality* programs for the **BBC MicroModel B/ B+/ Master 128/ Archimedes** why not send it to us for publication in Fast Access.

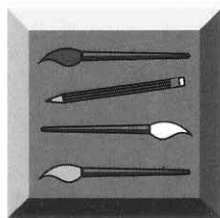
Send it to:

**FAST ACCESS**  
6C Belgic Square,  
Padhome Road,  
Peterborough.  
PE1 5XF.





# CELLULAR AUTOMATA



## AUTHOR

PHIL BENDER

## FILES

'CELLS' ..... BASIC  
'T.CELLS' ..... TEXT

**They are as fascinating as the Mandelbrot Set, and don't keep you up all night waiting for a result.** Scientists can use them as models of complex processes in Nature. The patterns they produce are beautiful, spectacular, and unexpected. What are they? They are the one-dimensional cellular automata, 1D cousins of John Conway's famous 'Life'.

The program is easy to use, with most of the instructions on screen. Escape has been redefined as <CTRL>+<@> so that the Escape key will not halt the program if pressed accidentally. You are asked for values for S

and N at the beginning of the program, so to change them you will have to run it again. I find it best to concentrate on a particular 'family' of automata at one time. S is obviously limited to 4 since the program runs in Mode 1; N is kept to 5 so that the rule will fit on one line. The rule starts out set to all 0s, but subsequently the previous rule is returned for you to edit. You can also choose the starting seeds: random gives a random scattering of random cells; you can recall the old start configuration, thus saving time; or you can enter a new one. In these latter cases you can edit the cells, the screen acting as a 38 cell window which can be swung left or right. Enter a cell by pressing the appropriate number, or remove it by pressing 0.

You can change the colours before pressing <RETURN> to

start the automaton. Pressing <Q> will halt the automaton at the end of the current plot line. When the screen is full you can change colours, or press <D> to save it under the filename 'CELSCRN', while pressing <S> will allow the automaton to continue, scrolling up the screen until a new screenful is displayed.

Rather than calculate the sum of the neighbours of each cell the program actually works out the contribution each cell

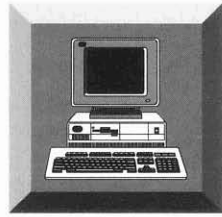
makes to the sums of its neighbours. If a cell's current value is zero then it has no contribution to make, and can be safely ignored. Likewise, a cell with value '0' need not be plotted, as the background is already in colour 0. These two amendments mean that the program runs up to four times as fast as the more obvious approach. In the ultimate case, a rule consisting entirely of 0s, it zips through the 256 generations in less than two seconds.

## RULES TO TRY

0 102 203 210	0 102 330 210	0 203 011 023
0 310 202 031	0 102 231 023	0 103 022 103
0 210 233 002	0 310 202 301	0 102 300 120
0 103 022 303	0 211 031 203	0 310 202 310
0 102 300 130	0 103 102 003	0 211 303 201
0 310 202 311	0 102 301 230	0 103 132 003
0 211 303 203	0 310 202 323	0 102 302 100
0 103 202 313	0 211 322 203	0 320 122 310
0 102 302 130	0 130 202 103	0 301 123 103
0 320 310 120	0 102 302 131	0 130 202 132
0 301 210 123	0 320 310 210	0 102 320 210
0 130 202 133	0 301 210 323	0 322 202 330
0 102 323 201	0 130 320 120	0 301 313 213
0 323 101 303	0 102 330 210	0 131 011 203
0 310 011 000	0 330 102 303	0 102 330 210
0 201 302 301	0 310 202 030	0 331 310 203



# SPEEKIT



## AUTHOR

JOHN DEWAR

## FILES

'SPEEKIT' .....BASIC  
'SPEECH!' .....M/CODE  
'T.SPEEKIT' .....TEXT

*Speekit* uses SPEECH! by Superior Software to provide an audible listing of any BASIC program stored on disk. It will read any part of a program, all you have to do is specify the start and finish lines. It gives you the spelling of all the variables and even tells you if the letters are upper or lower case. Print statements such as "Please wait" are spoken properly but in certain circumstances you may prefer to spell these out too.

All the BASIC keywords are pronounced as they should be, with obvious exceptions such as ACS and SGN. Even REM

statements can be read out or ignored. Speekit also provides a formatted listing on the screen which you can pause at any time by pressing the space bar.

## RUNNING THE PROGRAM

When you run the program it will instruct you to wait while SPEECH! is loaded into memory. There is also another short interval while the program creates an array and fills it with BASIC tokens.

When this is done, you will be asked to enter the name of the file that you wish to check, this should of course be a BASIC program. Up to 12 characters are accepted so that DFS users can specify a drive and directory. Alternatively you can enter a \*Command such as \*DIR, \*PITCH, \*CAT et cetera.

Having selected a file you are then asked for the first line to be checked. This start line can be anywhere in the program and allows specific sections and

procedures to be examined.

Finally you are asked to enter the last line to be checked. Obviously this must be equal to or greater than the first. If line numbers are not entered default values of 0 and 32767 will be assumed. When both line numbers have been entered you will be presented with the options menu.

### OPTIONS MENU

This menu is simply a box at the bottom of the screen which displays four options, Lines, Prints, Rems and Space, with a letter 'Y' (ON) or 'N' (OFF) next to each. Pressing L, P, R or S will toggle this setting. Each option effects the way in which Speekit reads a file.

With 'Lines' on, Speekit will read out the line numbers as it progresses through the program. With this option turned off line numbers will be ignored but the computer will emit a short bleep to indicate the start of each new line.

The second option 'Prints', determines how a print statement is read and allows the following to be pronounced properly. PRINT "Press space bar to continue". The alternative is to read out each individual

character.

Since REM statements have no effect on a program they can be ignored using the third menu option. However, you may prefer to use both the 'Prints' and 'Rems' options so that the REM statements within a program are read out as intended.

The final option prevents Speekit from saying 'Space'. The reason for this only becomes apparent when reading a number of lines with plenty of spaces in them!

To exit from the menu press the RETURN key and Speekit will start reading the selected file. Notice that as Speekit spells out the names of the variables in the program it will also tell you when the letters change from upper to lower case and vice versa.

It is possible to return to the menu at any time by simply pressing the space bar. However, while SPEECH! is 'talking' the normal interrupts are turned off and it can be difficult to get a response from the keyboard. For this reason it may be necessary to hold down the space bar until the menu appears.