



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
*Command Creator	CREATE	T.CREATE
Last Ninja 2 Demo	BEEP NINJA NINJA1 NINJA2 NINJA3 1.LEVEL-A 1.LEVEL-B	T.NINJA
Programming The 6845	CURSOR 6845BAS	T.PR6845
Screen Translator	SCTTRANS	T.SCTTRANS
Track Read/Write Util	+T-WRITE +T-READ	T.TRACKS
Home Help	ACCT CALENDA INPOUTP MTHPLAN TITLE CREDIT DIARY GAS/ELE HIST MENSEL OV2 OV2A OV2B OV2C OV3 OV4 GLPCODE EPSCODE	T.HHINTRO T.HHPART2 T.HHPART3

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 Disk User will not work on your system then please carefully re-read the instructions given above. If you still experience problems then return it to: SELECT SUBSCRIPTIONS LTD, 5 River Park Estate, Billet Lane Berkhamsted, Herts HP4 1HL. TEL 0442 876661 to 4

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.

Page layouts by Simon Fifield.

Printed by Loxleys, Sheffield.

Duplication by Direct Disk Supplies, Teddington, Middlesex.

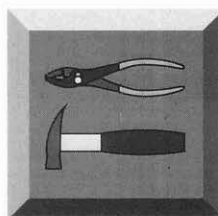
©Argus Specialist Publications Ltd 1990

RESOURCE FILES

TITLE	FILES	TYPE
Menu	FAMENU	BASIC
	GOSPR	DATA
	GOCODE	M/CODE
Info Desk	IDESK	BASIC
Change	CHANGE	M/CODE
Help	T.HELP	TEXT
Domino D.	DOMINO	BASIC
Hostages Ad	HOSTAGE	BASIC
	HOST	DATA
Garfield Scr.	GALLERY	BASIC
	Comp	M/CODE
	GALSCR4	DATA



* **COMMAND CREATOR**



AUTHOR

PETER DEW

FILES

'CREATE'BASIC
'BEEP'M/CODE
'T.CREATE'TEXT

An incredibly useful and versatile software feature of the BBC Micro is ***COMMAND**. It allows easy access to useful machine code routines, normally in ROMS. For example, the disk filing system and built in operating system have a range of commands, but it's not just roms that can have their own commands, in fact any machine code program can.

The coding needed to make additional commands however, is fairly complicated, and must be done in 6502 machine code. Presented here is a program that will convert a standard 6502 M/C program into one that, having been loaded, can be accessed

instantly by a single command. **USING THE PROGRAM**

When the program is run it asks for the **ORIGINAL** machine code program, the new ***COMMAND** and the destination program filename. It automatically loads in the original, adds the extra code and saves to the destination filename.

The new machine code program can be loaded in the normal way with ***RUN**, but instead of jumping to the main program, the ***COMMAND** is simply 'installed' and control returns to BASIC. Then typing the appropriate ***COMMAND** will instantly access the main program. If however, when the program is loaded you want it to install the command **AND** jump to the routine, it only requires the addition of a single line in the creator program :-
245 JMP routine

Attributes can easily be passed to the original machine code routine because when the routine is entered after a ***COMMAND** the contents of

X & Y are passed, so that the full command line can be analysed. The command interpreter used in the program accepts upper and lower case which is converted to upper case.

Having instant *COMMAND access could be particularly useful for applications like RAM based languages, or perhaps a wordprocessor.

HOW IT WORKS

When the original machine code program is loaded into memory the extra code is assembled in memory directly above the original program. The original program's execution address is the address used when the new command is recognised. The new program's execution address points to the extra code. This code works by changing the **CLI (Command Line Interpreter)** vector to it's address, and compares the command requested with the new command, if they match the original program is called, but if not the old CLI is jumped to.

All of the above is done in only about an eighth of a kilobyte, leaving loads of space for your original program! That's about it, one final thing - the machine code program must start above **TOP**. If you're

desperate for space you could try lowering **PAGE** to about &1200, providing you're not going to use random access filing. To do this type:-

```
PAGE=&1200 <RETURN>
```

Obviously if you're using a Master, **PAGE** will be at &E00, and will not need lowering. What you use it for, is up to you...

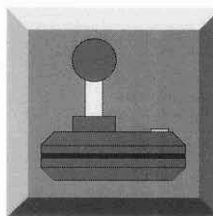
An example would be the machine code program **BEEP**. So here's how you'd use the program :-

Select the *Create icon from the **FAST ACCESS** menu and enter the following when the program prompts. (Note - 40 track DFS users will first have to copy the **CREATE** and **BEEP** files to a blank disk, and then load the BASIC program by typing **CHAIN "CREATE"**)

```
Original filename?  
BEEP  
Destination filename?  
BEEP2  
New *command? BEEP  
Now when in BASIC  
type the following:  
>*RUN BEEP2  
>*BEEP  
(BEEP!)  
>*BEEP  
(BEEP!)
```



LAST NINJA 2 DEMO



AUTHOR

PETER SCOTT

FILES

'NINJA' BASIC
'NINJA1' M/CODE
'NINJA2' M/CODE
'NINJA3' M/CODE
'1.LEVEL-A' DATA
'1.LEVEL-B' DATA
'T.NINJA' TEXT

This exclusive demo of the new 'Last Ninja 2' game from Superior Software written by Peter Scott contains the first ten

screens. This is only about 10% of the complete game. You can hack at the files all you like as the data files for levels two to six are not there!

NOTE FROM PETER

This game is my most recent release, and continues on from the original 'Last Ninja', the biggest selling BBC and Elec-

OTHER CONTROLS

P PICK UP, BLOCK
SPACE. SELECT WEAPON
H HOLD OBJECT
COPY/DEL PAUSE ON/OFF
S/Q SOUND ON/OFF
SHIFT/ESC QUIT GAME

THE GAME CONTROLS

MOVEMENT

Z NORTH WEST
X SOUTH EAST
* NORTH EAST
? SOUTH WEST
J SMALL JUMP
K MEDIUM JUMP
L LARGE JUMP

NO WEAPONS

RETURN/*PUNCH
 RETURN/?KICK
 RETURN/ZPICK UP/BLOCK
 RETURN/XPICK UP/BLOCK

tron game of 1989. It is also Superior Software's 100th release (*congratulations Superior - Ed*).

Producing a conversion is not an easy task. Many people think it is - no wrangling with original ideas, and so on. Well, this is true, but with an original game, you can compromise with anything too complex or difficult to program. With a conversion, you can't.

'Last Ninja 2' is a good example. The original used the Commodore 64 with hi-res 320x200 graphics in 16 colours.

This game had to be converted to run on the BBC and Electron. In 4 colour mode 5, due to memory needs.

Many of you may be thinking that 'Last Ninja 2' must have been easier to write because I converted 'Last Ninja 1'. So did I. This was a **BIG** mistake! I used quite a few routines I'd sweated over in the original. Modifying year old code wasn't easy. It ended up taking longer to re-use old code than it took to write the original stuff!

In future Issues of **FAST ACCESS** we hope to bring you more playable demos of new games so that you can play them for yourselves before making the decision to buy. Why rely on second rate reviews when you can try the game out yourself? Remember only **FAST ACCESS** disk magazine provides this service!

STAR • SMOKE BOMB

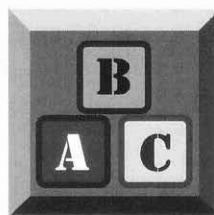
RETURN/ZTHROW
 RETURN/XTHROW

SWORD • NUMCHUKAS • STICK

RETURN/*HIT OR STAB UP
 RETURN/?HIT OR STAB DOWN
 RETURN/ZHIT OR STAB ACROSS
 RETURN/XHIT OR STAB ACROSS



PROGRAMMING THE 6845



AUTHOR

MOSTYN HELLARD

FILES

'CURSOR'BASIC
'6845BAS'BASIC
'T.PR6845'TEXT

BBC BASIC provides an easy way to program the 6845's registers in the form of the VDU23 command. This command takes the form;

VDU23;R%,V%;0;0;0

where 'R%' is the register number (0 to 17) and 'V%' is the new value to be written to that register. Reading from the registers is more involved because BASIC does not provide a command to do so directly. A function that will allow us to read from the register; 'R%' is listed in figure two. The function is called in the following manner;

V% = FNpeek (R%)

where 'R%' is the register to be read and 'V%' is the variable

that will contain the register's value after the call.

REG 0: HORIZONTAL TOTAL

This register contains the total number of displayed and non - displayed characters per horizontal scan line (counted from 0 so less 1). Do not confuse a 6845 character with an ASCII character, the 6845 character is in fact a byte. So this register contains the number of bytes (minus 1) used to represent a row in the current screen mode. This register is write only but you can see the default values for each mode in figure three.

REG 1: CHARACTERS PER LINE

This register contains the number of (6845) characters to be displayed. This total lies in the range 0 to the horizontal total (held in register 0). This register is write only.

REG 2: HORIZONTAL SYNC POSITION

Contains the position of the

first displayed character from the left. The default values are shown in figure three. This register is write only.

REG 3: HORIZONTAL SYNC WIDTH

Contains the duration of the horizontal sync pulse, alteration of this write only register will result in a display collapse.

REG 4: VERTICAL TOTAL

The value in this register determines the number of displayed and undisplayed character lines (minus one), and so the frame refresh rate. This write only register is 7 Bit.

REG 5: VERTICAL SYNC ADJUST

This five Bit, write only register is a finer version of register 4, allowing accurate adjustment of the frame refresh rate to be made.

REG 6: ROWS PER FRAME

This seven bit, write only register controls the number of lines displayed on the screen. For default values see figure three, altering this register when in mode 7 (or 135) will produce some strange if not useful, results.

REG 7: VERTICAL SYNC POSITION

This seven bit, write only register determines the position

of the vertical sync pulse and normally contains the number of lines on the screen plus two. Default values are shown in figure three.

REG 8: INTERLACE MODE

This register accepts values in the range 0 to 3 and determines whether the display is interlaced or not. This register should not be altered in mode 7 (or 135). The values are;

- 0 - Non interlaced display
- 1 - Interlaced sync display
- 2 - Non interlaced display
- 3 - Interlaced sync and video display

REG 9: SCAN LINES PER ROW

Contains the number of scan lines per displayed character, including blank spacing, less one. The values in this register (see figure three) explain the unused spaces between lines in the text only modes (3 and 6). This register is 5 bit and should not be altered in mode 7 (or 135).

REG 10: CURSOR START AND STATUS

This register controls the start position of the cursor relative to each displayed character, the value given must be less than the number of scan lines per row in register 9. Reg-

ister 11 controls the line on which the cursor stops. Register 10 serves another purpose, it controls the cursor's format. In order to program this register you must firstly decide the cursor start line then;

Add 0 for a static cursor

Add 32 for an invisible cursor

Add 64 for a quickly blinking cursor

Add 96 for a slowly blinking cursor (default)

The register is 7 Bit, write only.

REG 11: CURSOR STOP

Used in conjunction with register 10 this 5 Bit, write only register allows you to specify the depth, and position relative to each character, of the cursor. It's value must not exceed the number of scan lines per character (found in register nine), and must not be lower than the value of register ten (minus 0, 32, 64, or 96 as applicable).

REG 12: SCREEN START - HIGH

This register contains the most significant byte of the address of the start of screen memory. Altering the value stored here will cause the computer to display another area of memory on the screen! Values assigned to this register must first be divided (DIV) by eight.

REG 13: SCREEN START - LOW

This register contains the least significant byte of the address of start of screen memory. Values assigned to this register must first be as above.

REG 14: CURSOR ADDRESS - HIGH

This register contains the most significant byte of the cursor address in mode 7 (or 135). In the other modes it contains the most significant byte of the cursor address, times eight. This register is read/write.

REG 15: CURSOR ADDRESS - LOW

This register contains the least significant byte of the cursor address in mode 7 (or 135). In the other modes it contains the least significant byte of the cursor address, times eight.

REG 16: LIGHT PEN - HIGH

This register (read only) returns the most significant byte of the current light pen position (if fitted).

REG 17: LIGHT PEN - LOW

This register (read only) returns the least significant byte of the current light pen position.

FIGURE ONE

NUMBER	REGISTER TITLE	STATUS
0	Horizontal Total	W
1	Characters per Line	W
2	Horizontal Sync Position	W
3	Horizontal Sync Width	W
4	Vertical Total	W
5	Vertical Sync Adjust	W
6	Rows per Frame	W
7	Vertical Sync Position	W
8	Interlace Mode	W
9	Scan Lines per Row	W
10	Cursor Start & Status	W
11	Cursor Stop	W
12	Screen Start - High	W
13	Screen Start - Low	W
14	Cursor Address - High	RW
15	Cursor Address - Low	RW
16	Light Pen - High	R
17	Light Pen - Low	R

Figure One shows the register titles and if they can be read or written to.

FIGURE TWO

```
DEF FNpeek(R%)
  ?&FE00=R%
  =?FE01
```

Figure Two: Function to read 6845 register, called with command; 'V%=FNpeek(R%)'. Where 'R%' is the register to be read and 'V%' is the variable to contain it's value.

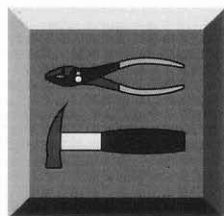
Figure Three shows the default values of each register in each mode. Note values the same with or without Shadow memory.

FIGURE FOUR

REGISTER ...	MODE NUMBER							
•	0	1	2	3	4	5	6	7
0	127	127	127	127	63	63	63	63
1	80	80	80	80	40	40	40	40
2	98	98	98	98	49	49	49	51
3	8	8	8	8	4	4	4	4
4	38	38	38	30	38	38	30	30
5	0	0	0	0	0	0	0	2
6	32	32	32	25	32	32	25	25
7	34	34	34	27	34	34	27	27
8	1	1	1	1	1	1	1	3
9	7	7	7	9	7	7	9	18
10	103	103	103	103	103	103	103	114
11	7	7	7	9	7	7	9	18
12	Varies when screen scrolls							
13	Varies when screen scrolls							
14	Varies when cursor moves							
15	Varies when cursor moves							
16	Only if light pen fitted							
17	Only if light pen fitted							



SCREEN TRANSLATOR



AUTHOR

TERRY BLUNT

FILES

'SCTTRANS'BASIC
'T.SCTTRANS'TEXT

The program on this disc is designed to take graphic screen files, display them defining a window on the screen, then produce a new file consisting of a stream of VDU codes that can be sent straight to any Epson compatible printer. All this without the need of any supporting program.

It was originally designed for creating small graphic strips for use as letterheads in any wordprocessor that supports OS commands. The translation process is quite slow, but the resultant file, having no processing overheads, will run as fast as the disk system and your printer will allow!

The translator program is Tube compatible and will work with any version of BASIC, with shadow screen systems and with The Model B, Master and Electron (with disks).

When running the program you are prompted for the source screen file. This can be any MODE 0,1,2,4 OR 5 picture that has been *SAVED. A non-existent file entry will cause the disc catalogue to be displayed and a repeat prompt.

Next you are asked for the name of the file that will contain the VDU stream.

The screen MODE number is prompted for as the program has no way of knowing which is required from the file.

The next prompt is for positive or inverse print. Half tones are not supported, the lower half of the colour range for the given MODE (disregarding flashing colours) being regarded as 0 and the remainder being 1.

Width can be set as wide or narrow. Wide print will stretch the display to fill the width of normal A4 paper, whereas narrow is useful if you want normal text from your wordprocessor in part of the heading. The display will obviously be distorted by these adjustments and you will have to decide which is most acceptable.

If you have selected narrow width you then have a choice of printing the display to the left of the paper, the right, or centred.

Up/Down Cursor keys are used to set the wire-frame window position that marks the area that will be translated and with SHIFT to alter the vertical size. The latter can only be done in 8 dot high jumps. This represents one pass of the print head in bit image mode.

Pressing the COPY key will send the marked area to a file.

Translation of a screen is done using the BASIC POINT function to determine 8 pixel high rows of the screen that build up individual bit image data bytes. Each of these is preceded by the length data and the whole file has the bit image code put on the front and the cancellation code put on the end. In order to prevent the unpredictable effects that would be caused by any of these codes being sent to the screen, every byte of the file is preceded by a 1. The only exceptions are the very first byte, which is a 2 to switch on the printer and the very last which is a 3 to turn it off again.

Those with 24 pin printers should be able to alter the printer codes to allow for this, and by taking a larger vertical slice it should be possible to get a better and faster output.

WANTED

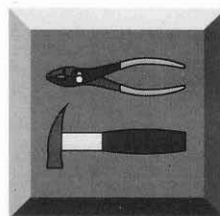
If you have written any useful, good quality programs for the BBC Micro why not send it to us for publication in Fast Access.

Send it to:

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



TRACK READ/ WRITE UTILITY



AUTHOR

MARK GIDLEY

FILES

'+.T-WRITE'BASIC
'+.T-WRITE'BASIC
'T.TRACKS'TEXT

Here are two useful disk utility programs which can be used for a variety of applications. It is important that you do **NOT** use these programs with your **FAST ACCESS** disks in the drive. **Dfs** users will have to insert a blank disk to hold the machine code files generated. **ADfs** users are advised to simply press SPACE to save the machine code to the same disk. This prevents a 'Disc changed' error message.

Both programs are constructed to allow them to be called as a * command from disk with relevant attributes.

The first program, Track Write has the syntax :

```
*TWRITE <trk> <afsp>
```

It can be used to write a file directly to disk without the need of a filename on the disk to which it is written. For example, to write the file TEST to track 9 onwards, the following command should be issued :

```
*TWRITE 9 TEST
```

Of course, files on other disks, with file directories can be used. For example, to write the file \$.TEST from drive 1 to track 22 onwards, the following command should be issued :

```
*TWRITE 22 :1.$.TEST
```

The length of the file is limited only by memory and this is &6300 bytes as the file loads in at &1400 (this can be altered, see below) and can load up to the top of screen memory in mode 7 (&7C00). You are advised to be in MODE 7 when using the utility, as the program will give directions, and so could corrupt

the loaded program if, say MODE 1 were used.

This program has a variety of uses. For example, it could be used as a protection method for hiding files from nosy people, because once the file is on the disk, there will be no trace of it on the disk catalogue. This type of protection is very common amongst software companies. The file can then be read back using the TREAD utility as explained later.

The second program, Track Read can be used to read tracks directly from a disk and has the syntax :

```
*TREAD <trk> <number>  
<addr>
```

If, for example, you wanted to read tracks 10 through to 14 from a disk, into memory from &2000 onwards, then the command needed to do this would be :

```
*TREAD 10 5 2000
```

It will read in a minimum of one track (excluding zero tracks!) and so will load in all 10 sectors of that track. Thus one track will use &A00 bytes of memory. It may be useful to note this so that when using the utility, you do not read over programs still in memory !

Together with the protection use of this program, it can be used to retrieve files on disks that have been deleted by using the ***WIPE** , ***DELETE** or ***DESTROY** DFS command.

This is because when these commands are carried out, it is only the filename which is deleted from the disk, not the actual data. Please note that this will probably only work if you have not since saved other files to the disk.

Both programs are fully relocatable by altering the value `assm%`. They are set up to assemble at &900, reaching up to &C00, so writing over the soft key buffer (&B00-&BFF) and the expanded character definitions (&C00-&CFF).

The TWRITE utility will currently load the program file to write to the disk at &1400 but this may be altered by changing the value of `buffer%`. If it is increased, then the length of the file which can be written to the disk will be &7C00-`buffer%`.

Both programs use memory locations &380-&3FF as a string buffer, which again can be altered by changing the value of `store%`.



HOME HELP



AUTHOR

NAME

FILES

'ACCT'	BASIC
'Calenda'	BASIC
'InpOut'	BASIC
'MthPlan'	BASIC
'Title'	BASIC
'Credit'	BASIC
'Diary'	BASIC
'Gas/Ele'	BASIC
'Hist'	BASIC
'MenSel'	BASIC
'OV2'	BASIC
'OV2A'	BASIC
'OV2B'	BASIC
'OV2C'	BASIC
'OV3'	BASIC
'OV4'	BASIC
'GLPcode'	M/Code
'EPscode'	M/Code
'T.HHINTRO'	TEXT
'T.HHPART2'	TEXT
'T.HHPART3'	TEXT

OVERVIEW

Home Help is a comprehensive suite of programs compatible with the complete BBC range from Electron to Master 128. Disk based, it works with DFS, ADFS and the 2nd Processor.

The package is designed to allow you to preview your financial position throughout the year and make managing your income much easier. Producing a complete and useful record of all your purchases for future reference, Home Help is very useful when making those tricky financial decisions. The program consists of several linked modules with a consistent user interface. Unlike other home accounts packages, it is not essential to constantly update your records every time a bill comes in. Of course, the better your record-keeping the more accurate Home Help's predictions will be.

WHAT IT IS NOT

A complete financial accounting system. Home Help will not provide the sort of information required by most businesses. Nor will it satisfy the law. However kept alongside traditional records, Home Help can be a powerful tool.

FEATURES

Perpetual Calendar

Diary

Month Planner

Gas & Electricity records

Credit card/ loan repayment schedules

On-screen help

Screen graphs, or dump to printer

Instant financial summaries

Data security option

STARTING HOME HELP

To get the most of Home Help, you will need to copy all the files as listed above on disk A, plus the file ACT0 from disk B to a freshly formatted disk. Create a !BOOT file with the line:

```
CHAIN"TITLE"
```

and set the !BOOT option with:

```
*OPT4,3
```

Pressing<SHIFT><BREAK> will now take you into Home Help. The program runs in

mode 7 on the BBC or mode 6 on the Electron.

HINTS & TIPS

If your Electron has been expanded so that page for ADFS is at E00, or the Mode 7 unit is fitted then the extra memory will be used by the program..

NOTE. Any files Saved using Mode 7 or the 2nd Processor will not load into the Un_expanded Electron.

To lower the screen display, type *TV255,1 then press <BREAK>.

Do this before data has been entered or it will be lost.

Use <ESCAPE> at the Menus to catalogue the disk, to handle files or use *FX6,0 when a printer line feed is needed. If more than one drive is available data can be stored on a selected drive, but the program disk must always be in drive 0. ADFS users must first create directories A, D and C on their data disk. If using a single drive with ADFS a backup copy of the program will be required for each set of data files. Press <ESC> then *MOUNT the new disk.

Start a new disk if space on the disk is close to 64K bytes as this amount is claimed for each new file. Compact the disk if

space is spread over the disk or a disk full error occurs. (Save your Data on another disk first before exiting the program.)

Backup copies are strongly recommended.

SECURITY SYSTEM

Home Help will protect your data from prying eyes with it's own built in security system. To use the system you must enter a code when Saving or Loading a file. The code must be three numeric digits followed by an optional name if required, eg 123FRED. If no code is required, just press <RETURN>. When saving data, you need not enter a code except when you wish to change it.

It is essential to remember the code you use.

USING THE BUDGET PLANNER

The Program is very easy to use so familiarise yourself by first trying all the options.

Note: SAVE cannot be selected until some new data has been entered.

If LOAD is selected you are given the chance to save your new Data before proceeding, and <M> will return you to the Main Menu. All calculations are made as Credits (money re-

ceived) and Debits (money paid out). You should enter Credits first.

Your present Status will be shown clearly for any month of the Year when viewing from the Monthly Menu. There are three pages for each month.

Press (H)elp for a list of keys when viewing months. You can move quickly to any page and also Print the page display if the Main Menu Select Printer option has been set
OPTION ON E: VIEW ACCOUNTS

The most frequently used option; you will use it to keep a constant check on the state of your finances. Selecting this will display:

Average week for the year followed by: **Month Menu**. Select Annual Balance or Month to be viewed. Month will show:

- 1.Regular bills with sub totals
- 2.Sundry items 1 & 2 and their totals
- 3.Your month's credit
- 4.Total to be paid out that month
- 5.Balance
- 6.Accumulated Credit balance
- 7.Accumulated debit for bills to be paid later

8.Credit balance available for the following month

If 5.is minus, 6.credit is used
MAIN MENU OPTIONS

There now follows a brief summary of Main Menu options 1 to 6

1.LOAD Old Acct & 2.SAVE New Acct

Enter Filename "ACT"+ref (eg 01.) Hit <ESCAPE> to list files on disk and type in *COMMANDS

3.Credit received

Enter your year's credit along with credit carried forward from last year. This should be the net amount in your bank balance on January 1st. Your average monthly credit (wages) will then be set this can be changed as necessary later.

4.Annual Debit

Estimate and enter all regular bills using last year's bills as a basis. The actual amounts are entered when you receive the bill. Calculations always reflect *present status*.

5.Quarterly Debit

Due every 3 months and can be entered for any single quarter. The amount will be split into the three months prior to the due date. Due dates may be

changed by first entering the cost for any quarter. If already set just press <RETURN>

6.Monthly debit

Always enter for the complete year in January. Use the automatic option to set for the whole year. Changes can easily be entered later.

All names can be deleted by entering <*> and zero for cost.

Note: <RETURN> bypasses any option

<=Back or Forward=>: Change month

Warning indicator: Memory Required

Reduce names mainly Sundry items for more room and save the file. Press <BREAK> to RUN the program, then re-load your datafile. Extra space will only be available when you re-load the reduced file. Bytes free are displayed when entering data.

MAIN MENU OPTIONS 7-D

There now follows a brief description of the remaining menu options.

7 & 8. Sundry debit 1 & 2

For any additional purchases made each month.

<= Back or Forward =>:
Change month

One way to use Sundry 1 & 2 is

79

to separate credit and cash purchases. When an item is selected, names can be changed using the editing keys so mistakes are easily amended. Sundry items may be deleted for the year by entering <WIPE> in January.

Delete the current month by entering <*> Enter <*> to delete a single item and enter zero for amount.

9.SELECT Printer

First checks if the printer is on line, if not, a warning is given.

**<Draft or Emphasized>:
Select print style.**

The Accounts Record heading is then printed, followed by Average Week and View Accounts. Press 'P' to print the display being viewed. Selecting Main Menu or <ESCAPE> will deselect the Printer.

Keys used when viewing or printing:

**< C ontinue / B ack / N ext
month >**

**<Last month / Menu / Help /
P rint>**

P is only shown if the Menu option has been selected 'Due' will indicate the item to be paid that month 'New' shows a

monthly item that has changed since last month

**A.DIARY: Filename
"Diary"+ref(eg 01.)**

This has a separate file for any other information. Diary Month List:for month and year selected Further Events: Additional notes

A. for annual items shown yearly

Q. for quarterly interval events

M. monthly for every year

S. for single events

Also:

Appointments : single events

Celebrations : annual events

Vacations : single events

Selected separately from the Menu, these may be recorded for any year.

Calendar: indicates if there are items entered for the month selected. A copy of the Month or Diary List may also be printed.

View the calendar to check for any entries with the following keys

< F A C V >: for above items

**< L ast N ext year / M enu /
Curs ors >: to change month**

< H elp Menu >

**< D iary list for month se-
lected >**

B.Years Calendar

Will display or print any year and the dates for the usual annual holidays. A section for notes is provided below the calendar when printed. Planner (Selected from Calendar.) A Monthly Calendar page for making notes may be Printed for any year or month.

C.Gas/Elec: Filename

"CHART"+ref (eg 01.)

Stores a record of your fuel/cost and will calculate and store current prices. The average cost and units used can also be seen and used in the Budget Manager.

The Barchart presents a comparison of all your Gas and

Electricity. bills over the past seven years. All charts have a print option. Files may be checked with the <ESCAPE> key.

D.Credit calculator

For credit cards and Bank loans. This option will show the schedule of costs, APR, annual interest, and Total to be paid on a monthly basis.

HISTOGRAM

This option is available only from the scrolling list under the Main Menu. Selecting it will list to screen or printer your monthly expenses and the items as a percentage cost of your year's credit.



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.

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 a bit of 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)
- Manual

Order product number 'EB4' and specify 40 track or 80 track DFS disk. Price £12.00 to FAST ACCESS subscribers only. **SAVE £3** on normal price. Includes 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 orders can be taken on 0442 6655