Support Group Application Note
Number: 030
Issue: 1
Author:

BBC B+ 128K - Additional User Information

The following instructions are intended to assist you in making the best use of the 64K bytes of sideways RAM installed in your computer. They are intended to be used in conjunction with both the BBC Micro User Guide and the Acorn Disc Filing System User Guide.

Applicable Hardware : BBC B+

Related Application Notes: Information on the B+ 128K

Support Group
Acorn Computers Limited
Acorn House
Vision Park
Histon
Cambridge CB4 4AE

Introduction

Generally, there are two ways in which to use the 64k bytes of sideways RAM provided with the computer. Firstly, the RAM may be viewed as four 16k byte paged blocks into which suitable ROM images may be loaded. The second way is to view the 64k bytes of sideways RAM as one contiguous block of memory for use as BASIC program space. These two uses form the basis of the two different methods of addressing the RAM.

For those whose only intention is to use the sideways RAM in one of the two ways mentioned above, the section that follows is intended as a quick guide to its use. For the more adventurous, there then follows a more in-depth explanation of the means by which the sideways RAM may be utilised.

GETTING STARTED

Having set up the computer in the way described in the User Guide you should proceed as follows:

Loading a ROM image
To load a suitable ROM image from the currently selected filing system into the computer you should enter:

*SRLOAD <FILENAME> 8000 W Q then press the RETURN key.

This will cause the ROM image, specified in the filename parameter, to be loaded into sideways RAM in the first 16k block of memory.

Repeating the above command with difference filenames, and incrementing W through X, Y and Z, will allow you to fill the three other blocks of sideways RAM with ROM images.

Having loaded the image(s) into sideways RAM you must press the BREAK key. This caused the newly loaded ROM image(s) to be initialised and they may then be accessed in the same way as a sideways ROM.

Using the sideways RAM for BASIC programs
The 64k Sideways RAM Support Disc should be inserted into the disc drive attached to your machine. You should then enter the following:

*BAS128 then press the RETURN key.

This will load a disc-based version of BBC BASIC into your machine which will in turn allow you to make use of the 64k of sideways RAM as BASIC program space. For details of BAS128 see below.

DETAILED DESCRIPTION

If the sideways RAM is not being used in conjunction with BAS128 the user has two different methods of addressing the sideways RAM. These are referred to as the ABSOLUTE method and the PSEUDO method. The absolute method treats the sideways RAM as four separate pages, referring to them with ROM IDs, the pseudo method treats the sideways RAM as one contiguous block addressed in the normal way.

Absolute Addressing
This form of addressing sees the computer's memory map in the normal way with the 64k of sideways RAM being split between four 16k pageable blocks. These four paged blocks of RAM may be paged in at &8000-&BFFF in the same way as the paged ROMs.

The four pages of sideways RAM are identified as pages W, X, Y, and Z, which correspond directly to ROM pages C, D, 0 and 1 (see note 1). As such, they may be used for storing ROM images loaded in from disc or network filing systems.

To read from or write to these pages of memory, when in the absolute addressing mode, a ROM ID parameter must be specified when issuing the appropriate command.

This means of addressing is mainly for manipulating data that conforms to the same format as a paged ROM. Also the size of the code cannot exceed 16k in length. Where the area of memory required exceeds 16k, the absolute addressing method is inappropriate and the pseudo addressing method should be employed.

Pseudo Addressing
With this form of addressing, the four 16k blocks of sideways RAM are regarded as one contiguous 64k block (see note 2).

The blocks are configured in the manner shown in figure 1. As you can see each of the 16k pages have a range of address allocated to them, and it is these hexadecimal addresses that specify the area of sideways RAM to be used and not the ROM ID.

To use these pages for data it is necessary to first specify that this is the use to which they are to be put. This is carried out using the *SRDATA command (see below). Therefore, to use the range of addresses &00 through to &3FEF a *SRDATA W command must be issued. To use the entire 64k block, the *SRDATA command must be issued for W, X, Y and Z in turn.

It should be noted that, subject to their allocation using the *SRDATA and *SRROM commands, each block may be specified for use with either ROM images or DATA irrespective of the use allocated to the other blocks.

Note 1: At the point of manufacture the ROM numbers allocated to BASIC are E and F thereby giving it the highest priority. It is possible to change this by altering the position of the hardware link S13 located on the main PCB. If this priority change is performed, the sideways RAM will occupy pages C, D, E and F, with BASIC occupying pages 0 and 1. This will also mean that ROM IDs Y and Z will correspond to ROM IDs E and F respectively.

Note 2: When used as a contiguous block, the sideways RAM will provide 65.472kbytes of RAM and not a full 64kbyte block. This is due to 64 bytes of RAM being allocated for use with the ROM header information.

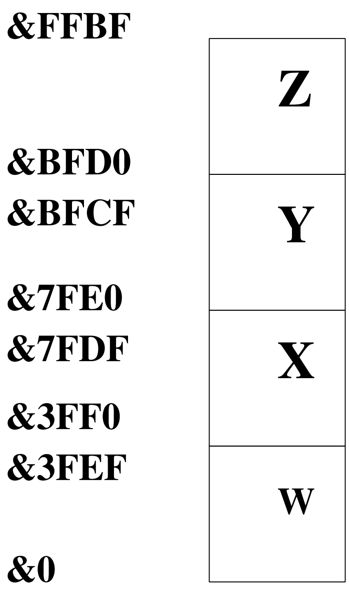

Figure 1

Sideways RAM Utilities

In view of its nature, sideways RAM can prove difficult to use. To help you, there are six utilities which are resident in the 1770 disc filing system ROM already present in your machine. These utilities are as follows:

UTILITY	PARAMETERS
*SRSAVE	<FILENAME> <START ADDRESS> <END ADDRESS> (<ROM ID>) (Q)
*SRLOAD	<FILENAME> <LOAD ADDRESS> (<ROM ID>) (Q)
*SRWRITE	<START ADDRESS> <END ADDRESS> <START ADDRESS> (ROM ID)
*SRREAD	<START ADDRESS> <END ADDRESS> <START ADDRESS> (ROM ID)
*SRROM	<ROM ID>
*SRDATA	<ROM ID>

The parameters used with these utilities, with two exceptions, should in general be familiar to you. The two new parameters, ROM ID and Q, are explained below, with explanations of the utilities following thereafter.

It should be noted that the END parameter requires the address of the byte immediately after the last byte of the block of memory involved. In other words, for <END ADDRESS> read <LAST ADDRESS+1>.

The ROM ID parameter

The ROM ID parameter allows a particular page to be specified for use. There are two different ways in which this may be done:

ROM ID in the range &0 to &F

This is the basis of the absolute addressing mode. the pages of memory specified in this range correspond directly to the sideways memory pages recognised by the computer's operating system.

Note that although any ROM ID in the range &0 to &F may be entered, only C, D, 0 and 1 (see note 1) are pertinent when considering this particular application. Use of any parameter, other than those referring to sideways RAM pages, is precluded during all read operations because of the need to protect ROM based software.

<u>ROM ID in the range W to Z</u>
The ROM IDs W, X, Y and Z correspond directly to ROM IDs C, D, 0 and 1. They are provided as a 'non-machine-specific' way of specifying four memory pages. As with ROM IDs &0 to &F, the use of these ROM IDs is part of the absolute method of addressing.

<u>ROM ID left unspecified</u>
Not specifying a ROM ID constitutes use of the pseudo addressing method. The operation will be performed with regard to the specified address(es) only.

The Q parameter
The Q (Quickly) parameter allows the speed to a *SRLOAD or *SRSAVE operation to be increased. As sideways RAM occupies the same address space as the currently selected filing system, loading and saving to and from sideways RAM is noticeably slower than the same operation performed with normal RAM. To speed up the operation, the use of the Q parameter will designate normal RAM between OSHWM and the bottom of screen memory as a buffer. This dramatically increases the size of the buffer available for transfers between sideways RAM and the currently selected filing system, thereby speeding up the load or save operation.

It must be born in mind that, although the greater speed is desirable, the use of the Q parameter is not always a good idea. Data resident in this area of normal RAM (this includes BAS128 when loaded) will be overwritten when a 'Quick' load or save command is performed.

THE SIDEWAYS RAM UTILITIES

<u>*SRSAVE <FILENAME> <START ADDRESS> <END ADDRESS> (<ROM ID>) (Q)</u>
This command performs the same function as the *SAVE command (see the Disc Filing System User Guide) except that the area of memory to be saved to the filing system is resident in the sideways RAM.

As with the *SAVE command the end address may be substituted with the length of the block of memory concerned. If this is used the syntax becomes:
*SRSAVE <FILENAME> (<START ADDRESS>+<LENGTH>) (<ROM ID>) (Q)

<u>*SRLOAD <FILENAME> <LOAD ADDRESS> (<ROM ID>) (Q)</u>
This command functions is the same way as the *LOAD command (see they Disc Filing System User Guide) except that the area of memory into which the specified file is read, is resident in the sideways RAM.

Note: having loaded a ROM image into sideways RAM it is necessary to press the BREAK key. This is because certain types of OS call directed to the specific page into which the image is being loaded, are disabled by the write operation. These calls are not reinstated until a reset is performed. The same applies to ROM images loaded using *SRWRITE.

*SRWRITE <START ADDRESS> <END ADDRESS> <START ADDRESS> (<ROM ID>)
This command writes a copy of a specified area of memory contained within normal RAM to a specified position in sideways RAM.

The fist parameter is the address of the start of the block of memory in normal RAM that is to be transferred. The second parameter is the address in normal RAM of the first free byte after the end of block that is to be transferred.

The second parameter may be replaced with the length of the block to be transferred. If the length is substituted for the end address of the block, then the syntax of the command becomes:
*SRWRITE (<START ADDRESS>+<LENGTH>) <START ADDRESS> (<ROM ID>)

The third parameter is the address of the start of the area in the sideways RAM into which the block is to be transferred.

*SRREAD <START ADDRESS> <END ADDRESS> <START ADDRESS> (<ROM ID>)
This command performs the reverse function of the *SRWRITE command. It reads a specified block of sideways RAM and copies it onto a given area of normal RAM>

The first two parameters point to the start and end of te block in NORMAL RAM into which the information is to be transferred. The third parameter points to the start of ge area of SIDEWAYS RAM that is to be transferred.

It is important to realise that the first address parameters given with both *SRREAD and *SRWRITE refer to normal RAM.

*SRROM <ROM ID>
This command is used to specify the use to which a page of memory is to be put. It reserves the page of sideways RAM specified by the ROM ID parameter, for use with the absolute addressing mode.

This is the default condition for all of the sideways RAM pages and is only reset upon power-up.

*SRDATA <ROM ID>
This command performs the same function as *SRROM except that the specified page if sideways RAM is allocated for use with data. If this command is issued in respect of one or more pages of sideways RAM, the specified page(s) must then be addressed using the pseudo addressing method.

Important Notes:
Users should note that th images of some commercially available ROMs will not function in the sideways RAM area due to protection mechanisms employed by some software houses.

The sideways RAM utilities are only resident in version 2.2 DFS and above.

BAS128 - DISC BASED BBC BASIC

This version of BBC BASIC gives the user access to a full 64k of memory for BASIC programs.

BAS128 is loaded from the disc by typing : *BAS128

The screen will be cleared and the banner BBC BASIC will appear together with a bytes free message 64k.

The ">" prompt will be displayed and commands can be issued in the same way as for the ROM based BASIC.

If you are a new user to the BBC Micro you may wish to read the User Guide before using this disc-based version of BBC BASIC. If you are familiar with the use of BASIC and intend to write your own programs, you may need to be aware of the technical differences between the ROM-based BASIC and the disc-based BAS128. They key point is that this disc-based version is loaded into the RAM area in which a user program would normally reside. The 64k area of additional "sideways" RAM is then made available for the user program.

The detailed differences between the two BASICs are as follows:

1. When using BAS128, the 64k of additional memory pretends to be a continuous address space above the normal 64k address space of the computer. Thus addresses &000000 to &00FFFF refer to normal memory and addresses &010000 to &01FFFF refer to the 64k bytes of sideways RAM.

   PAGE defaults to &10000
   HIMEM defaults to &20000

   Similarly, LOMEM and TOP are addressed in this way.

2. Indirection Operations.

   Statements such as DIM code 100 will assign a 17-bit value to "code". Indirection operators may then be used to read and write data into the area of memory reserved by the DIM statement: eg:

   !code=27127, PRINT code?3 or $(code+27)="HELLO"

   Indirection operators will address normal memory if the calculated pointer is in the range &000000 to &00FFFF and will address sideways RAM if the pointer is in the range &010000 to &01FFFF.

3. The Assembler

   This will operate using 0% and P% as 17-bit pointers.

   a) The Destination Pointer
   If the destination for the code is in the range &000000 to &00FFFF, then normal memory is referenced. Addresses from &010000 to &01FFFF refer to sideways RAM.

   b) The Program Counter
   The code produced is adjusted so that address references in the range &010000 to &01FFFF refer to sideways RAM addresses &008000 to &00BFFF. References from one bank of sideways RAM to another are trapped giving a BANK error; eg:

   P%=&13F00:[JSR &1CF34

   will give a BANK error.

Note that, when BAS128 is resident in user RAM, the Q parameter detailed above should NOT be used as BAS128 may be overwritten.

When compiling, any code generated over a bank boundary is trapped, resulting in a WRAP error. Thus, the maximum amount of code that can be generated at one time is 16kbytes; eg:

```
P%=&13FFC:[LDA #0 :TAY:SEC:SBC &70
```

will give a WRAP error since execution cannot pass from one bank to the next directly.

4. SHADOW mode will be forced on so that the BAS128 can reside in RAM between &3000 and &8000. The remaining memory from OSHWM to &3000 may be used by BAS128 for workspace.

5. The LOAD, SAVE and CHAIN commands make use of the OSGBPB routine which is not available with the TAPE filing system on OS 2.00. Thus these commands will not work if the TAPE filing system is selected. They will work with DISC, ADFS or ECONET.

ADDITIONAL OPERATING SYSTEM COMMANDS

There are four new Operating System calls which have been provided for use with the sideways RAM. These are as follows:

OSWORD 66 Block Transfer
The parameter block:-

XY+	0	bit 7 clear/set for READ/WRITE
		bit 6 clear/set for ABSOLUTE/PSEUDO
	1-4	Main RAM address
	5-6	Block Length
	7	ROM ID 0-&13 0-F, W, X, Y, Z, (ABSOLUTE only)
	8-9	SRAM address

OSWORD 67 Load Save
The parameter block:-

XY+	0	bit 7 clear/set for SAVE/LOAD
		bit 6 clear/set for ABSOLUTE/CONTIGUOUS
	1-2	File Name address
	3	ROM ID 0-&13 0-F, W, X, Y, Z (ABSOLUTE only)
	4-5	File length (SA+E only)
	6-7	Buffer Start address
	8-9	Buffer length, 0 for default,
		>=32K for buffer from OSHWM to bottom of screen memory

OSBYTE 68 Test RAM Presence
Returns a byte in X

7 0
* - F E 1 0 D C RAM Present flags

* - bit 7 set if E/F are being used for 3rd and 4th pages.
Bits 0-5 set if RAM pages found in given slot (see above), on last BREAK/POWER ON.

OSBYTE 69 Test Pseudo/Absolute Use of Page
Returns a byte in X.

Bit 7 as for OSBYTE 68,
Bits 0-5 set if RAM in given slot (see above) is being used in PSEUDO addressing mode.