USEFUL INFORMATION ABOUT BBC MICROCOMPUTERS

The Computing Service accepts no responsibility for the accuracy of this information. It's often by word of mouth but does appear to be correct. Contact John Matthewson (MJM3, P14, ext 623) to notify of errors, suggestions etc.

This file was last updated 01-01-83. don't believe this.

Fitting A to B Upgrade Kits

Users fitting their own A to B upgrade kits to the BBC micro should exercise care when plugging the extra chips into their sockets (remember to turn the power off!) the sockets are not always the right way round; if you are sitting at the keyboard then all chips have pin one (marked by notch or other molding) directly away from you or to your left.

Fitting Disc Interface to Issue 3 Boards

Snip pin 9 of IC 27. Connect leg of IC to right hand contact of S9 (which should be broken). Break tracking coming out between pins 9 & 10 if IC 27.

Sideways Language ROMs on Issue 3 Boards

Some issue three boards do not see the language ROM unless its inthe left hand most sideways ROM socket (ie one next to OS). If this is the case check S12 and S17. Break them if they are connected.

Updating from EPROM OS0.1 to ROM 1.0

If the operating system is in four 2732 EPROMS then they will need to be removed to make room for sideways language ROMs. To upgrade get an OS which looks like one ROM, ie a ROM or 4 2732s or 2 2764s in carriers. Remove the EPROMs, remove the BASIC chip (extreme left socket) and put it in one of the other sockets. Put the OS ROM in where the BASIC came from. Configure the jumpers as follows:

S18 - N ; S19 - S ; S20 - N ; S21 - 2xE/W ; S22 - N
S32 - W ; S33 - W

Speeding up the OS

S18 and 19 control the speed of access to the ROM sockets. Setting them at N & S configures for fast access. Some 2732 EPROMs can keep up with this though its working outside their specification.

Differences Between BASIC 1 & 2

These are the differences between BASIC 1 and BASIC 2:

1. ELSE no longer leaves a byte on the hardware stack in ON..GOTO/GOSUB.

2. INSTR no longer leaves the shorter string of INSTR("AB","ABC") on the software stack.

3. EVAL("TIME") now works.

4. Bug in ABS mended - PRINT -ABS1 works.

5. ASC":" now works inside the assembler.

6. New statement OSCLI which takes a string expression and passes it to the OS command line interpreter.

7. OPENIN now opens files for input only. New keyword OPENUP opens for updating but has the same token that OPENIN had.

8. Next bit of OPT controls whether code will be put at the program counter P% or the code origin O%.

9. LN and LOG recoded - more accurate.

10. INT 1E30 now works.

11. SIN/COS now more accurate.

12. binary to decimal conversion changed. Can now print 10 figures. 7.7 (recurring binary fraction) now prints as 7.699999999.

13. EQUB, EQUW, EQUD, EQUS now in assembler. EQUS does not put CR on end of string. Assembler now has macros by usinfg FN:

```
    DEF FNosbyte(A,X,Y)
    [OPT Z:LDA#A:LDX#X:LDY#Y:JSR&FFF4:]:=""

    [OPT Z
     EQUS FNosbyte(1,2,3)
     ......
```

14. REPORT gives (C)1982 after start up.

15. ON ERROR GOTO 9999 works.

16. Change of MODE resets COUNT.

17. BASIC only executes if A contains 1, no entry at 8003.

18. BASIC is of type 60 - to do with tube.

19. Version number 01.

20. In INPUT ";" is introduced and functions as "," does.

21. Errors with ERR zero cause ON ERROR OFF.

22. STOP is now fatal.

23. "No room" is fatal.

24. Error handling does not use stack.

25. Allocation of strings more efficient. Can now use REPEAT A$=A$+"*": UNTIL LEN A$ =255.

26. Error 45 is "Missing #" after PTR, EOF etc.

27 DIM P% -2 gives Bad DIM.

Bug in OS1.2 ?

OSWRCH appears to occasionally corrupt the X register. The effect seems to occur just after or during changing the text background or foreground colours.

Looking at the Keyboard

In OS1.2 at least, the most significant bit of location &EC is set if a key is depressed at that moment. This feature should not be used as it is not guaranteed to be supported except where speed is required. Polling the keyboard generally takes ~2 ms which can be too long when handling 9600 baud lines.

RS423 Handling

Note that operating system 1.0 or later is required for a full implementation of the RS423 line.
Use the OSBYTE call *FX156,format,mask where 'format' is the bytes to be written to the 6850 register and 'mask' is the mask as used to select which bits are set/unset (see page 430 in the manual). Writing direct to the command register will not work as the operating system will change the byte from time to time.
BREAKs are a bit of a problem. If you set the relevant bits in the command register to put a BREAK level on the line, the OS will clobber this every centisecond when it services the clock interrupt and checks the CTS state. Only way to get around this is to disable interrupts during the BREAK routine. However, this means any incoming characters will be lost, the clock will be incorrect etc.
The RS423 buffering system is enabled by *FX2,2. To send/receive use *FX138,0 (note this call is not limited to the keyboard buffer) and *FX145,1.
Note that the cassette interface not only uses the same hardware (the 6850) but also has the same buffer space as the RS423 so there are very unpleasant side effects if the two are used at the same time!

Reading the keyboard from machine code

If a soft key is hit which expands to a string of text then calls to OSRDCH will return that string WHATEVER the selected source. That is if a soft key containing "ABCD" is hit and OSRDCH is called "A" is returned. If the source is then changed to the RS423 for example, using FX2,1, then OSRDCH will return "B", not the character waiting on the line. If soft keys are required to work use calls to OSRDCH or FX129 (read key with time limit). The RS423 line should then be read using FX145,1 (but with the RS423 enabled, FX2,2). If the soft keys are not to be expanded use *FX145,0.

TERMINAL SOFTWARE FOR THE BBC MICROCOMPUTER

This file contains information about using a BBC microcomputer as a Phoenix terminal. It will be updated from time to time so watch this space! This version was last modified 10-01-82. Enquiries should be made to John Matthewson (MJM3) or Mick Seaman (MJS3) who are both on extension 623. Unless urgent, we would prefer messages via the 3081 rather than telephone calls.

Purchasing

See MICROLIB.DISCOUNT for details of where and how to purchase BBC microcomputers. A Model B is needed for terminal use.

The Machine Operating System (MOS)

All software is currently being developed under operating system 1.2 and is not guaranteed to work under earlier versions (and indeed generally does not). This is because earlier versions either do not support the RS423 interface or have bugs associated with it. As yet no machines are supplied with this version. Machines either have version 0.1 or 1.0. Acorn say that they will be receiving MOS 1.2 in ROM soon (Christmas?) and it will probably cost about 10 pounds. However, delivery is uncertain and it may be some time before the ROMs are actually available. In the meantime we can be of some help.

You can find out which version of the MOS your machine has by using the command "*FX0" which displays the version number on the screen. Machines are currently being delivered with 0.1 in ROM unless they have a disc interface in which case they have 1.0.

If your machine has MOS 1.0 then we can update it by a simple chip swap. Contact us to arrange a time at which to bring the machine in for modification.

If your machine has 0.1 then the MOS may be either in ROM or EPROM (ignore the fact that "*FX0" always implies that it's EPROM!). To find out which remove the cover and look under the right hand side of the keyboard printed circuit board. You will see either two shiny black integrated circuits and three empty sockets, or one shiny black one and four dull grey ones with labels stuck on. If you have the former your MOS is in ROM, but the latter implies it's in EPROM.

If you have a ROM then you can wait until Acorn can supply 1.2 in ROM or wait a shorter time for us to have 1.2 available in EPROM - this will cost 25 pounds and we hope will be available soon. Watch this space. If you have EPROMs then they can be updated at no charge. However, all the sideways ROM sockets are then used and the terminal software cannot be resident. As an alternative, Acorn will provide, when available, a 1.2 ROM free of charge in exchange for the EPROMs. For urgent cases we will be able to update to 1.2 at a charge of 15 pounds.

As an alternative if you have a ROM system, you could have a disc interface installed. This would provide you with 1.0 which we can update. The Disc Filing System (DFS) comes in an EPROM which we can then program with the terminal software. The cost of the upgrade kit compares well with that of us providing both 1.2 in EPROM and the terminal software (45.50 pounds + VAT of 35+15 pounds). Note that the DFS EPROM is not required for operating cassettes and is best removed since it causes the default disc filing system to be non-existant discs. If the DFS is required at a later date it can be provided. This last option is the one we recommend as it will probably be the most convenient one for both you and ourselves.

Wherever possible we will undertake to update operating systems and install terminal software ourselves (at the Computer Laboratory) but we can not accept responsibility for any accidental damage or loss of warranty incurred by this.

Availability of the Terminal Software
=====================================

Terminal software is still under development and is only distributed on the understanding that it almost certainly will contain bugs! At present the software not only allows terminal emulation but can preform file transfer between the 3081 and either disc or cassette. Graphics is also built in but the setup will only work in conjunction with the VIEW program in the 3081.

A BBC microcomputer is currently in use in the User Area as a graphics terminal and people interested should use this machine for demonstration purposes.

The software is available either on disc or cassette (please send the relevant medium to us with your request for the software). At present we can only support single side/single density discs and double side/double density (for the Torch disc drives as well as for the Acorn 3000). It is not expected that we will be able to support other formats - you may need to lend us your drive!

The software is also available in EPROM at a cost of 15 pounds. This allows the program to be resident in the machine so that it does not need to be loaded from disc or cassette each time it is required. Note that the machine can still be used as a standard BBC micro when this chip is present.

The software is provided under the usual conditions of the Computing Service (see MICSFLIB.SOFTWARE.LICENSE) and users will be required to sign an agreement before receiving it. Note that we reserve the right to withhold the software either in whole or part.

Future plans include implementing a subset of the Tektronix standard in order to emulate Tektronix graphics terminals for the VIEW program which will give enhanced plotting speed, but this will not be a full implementation and so will not necessarily work on other machines. It is also hoped to provide a facility for providing a screen dump on dot-matrix printers. This will be useful for obtaining rapid but low quality hard-copy of graphics. This feature will only be provided for use with the Epson MX80 printer (see MICSFLIB.DISCOUNT).

Miscellaneous Items
===================

Member PROG

```
C Execute by commanding file camplot.c:run
C Key labels for beeb
      INTEGER LU(10),LS(10),LC(10),LCS(10)
      INTEGER UKEY(4,10),SKEY(4,10),CKEY(4,10),CSKEY(4,10)
      READ (5,60)
      WRITE(6,10)
   10 FORMAT(' Verify strings read correctly:')
      DO 50 I=1,10
      READ (5,70) LU(I),(UKEY(J,I),J=1,4),LS(I),(SKEY(J,I),J=1,4),
     & LC(I),(CKEY(J,I),J=1,4),LCS(I),(CSKEY(J,I),J=1,4)
      WRITE (6,71) LU(I),LS(I),LC(I),LCS(I),(UKEY(J,I),J=1,4),
     &(SKEY(J,I),J=1,4),(CKEY(J,I),J=1,4),(CSKEY(J,I),J=1,4)
   50 CONTINUE
   60 FORMAT ( 3X)
   70 FORMAT (2X,4(I2,4A4))
   71 FORMAT (' ',2X,4(I2),4('1',4A4))
    1 NKEY = 11
      PW = FLOAT(NKEY+2) * 10.02
      TW = 7.0
      NTIMES = 3
      NBOLD = 3
C
      CALL GLPC2S
      CALL GLPS2S('PLOT    ', PW, 1.0)
      CALL GLWV4S(0.0, 1.0, 0.0, 0.15, PW)
      CALL GLTXDS(1, 8)
C
      CALL GLWV2S(-1.0, FLOAT(NKEY+1), -0.53, FLOAT(NKEY)+1.57)
      CALL GLTF2S(2, 1.0/TW, 0.0, 0.0, 1.0/TW, 0.0, 0.0)
C
      DO 40 IPIC = 1, NTIMES
      IF (IPIC .NE. 1) CALL GLDV3S
      DO 40 IBOLDS=1,NBOLD
      CALL GLTXDS (1,10)
      CALL KEY(0, 'Ctrl/shift', 10, 'Control', 7,
     &  'Shift', 5, 'n', 0)
      CALL GLTXDS (1,8)
      DO 80 I=1,10
   80 CALL KEY(I,CSKEY(1,I),LCS(I),CKEY(1,I),LC(I),
     &SKEY(1,I),LS(I),UKEY(1,I),LU(I))
C
      NN = NKEY+1
      DO 20 I = 2, NN
      CALL GLMT2S(FLOAT(I-1), 0.0)
      CALL GLDT2S(FLOAT(I-1), 1.0)
   20 CONTINUE
      DO 30 I = 1, 5
      CALL GLMT2S(0.0, FLOAT(I-1)/4.0)
      CALL GLDT2S(FLOAT(NKEY), FLOAT(I-1)/4.0)
   30 CONTINUE
C
      XL = -0.5
      XH = FLOAT(NKEY) + 0.2
      YL = -0.32
      YH = 1.15
      XD=.2
      YD=.2
      CALL GLMT2S(XL-XD, YL)
      CALL GLDT2S(XH+XD, YL)
      CALL GLMT2S(XH, YL-YD)
      CALL GLDT2S(XH, YH+YD)
      CALL GLMT2S(XH+XD, YH)
      CALL GLDT2S(XL-XD, YH)
      CALL GLMT2S(XL, YH+YD)
      CALL GLDT2S(XL, YL-YD)
   40 CONTINUE
C
      CALL GLPS6S
      CALL GLPC4S
      STOP
      END
C
      SUBROUTINE CENTRE(ITEXT, L, X, Y)
      DIMENSION ITEXT(1), A(2)
      CALL GLMT2S(X, Y)
      CALL GLTXES(ITEXT, 1, L, A)
      CALL GLMT2S(1.5*X - 0.5*A(1), 1.5*Y - 0.5*A(2))
      CALL GLTXPS(ITEXT, 1, L)
      RETURN
      END
C
      SUBROUTINE KEY(N,IT1,L1,IT2,L2,IT3,L3,IT4,L4)
      DIMENSION IT1(1), IT2(1), IT3(1), IT4(1)
      X=FLOAT(N)+.5
      CALL CENTRE(IT1, L1, X, 0.82)
      CALL CENTRE(IT2, L2, X, 0.57)
      CALL CENTRE(IT3, L3, X, 0.32)
      CALL CENTRE(IT4, L4, X, 0.07)
      RETURN
      END
```

We will soon have available leads suitable for connecting a BBC microcomputer to a Phoenix line. Price to be announced.

Ordering

If there is a charge for the items that you require send an official order made out to the Computing Service. Wherever possible an Internal Debit Note will be issued (to University Departments) but where this is not possible an invoice will be issued and the above prices will be subject to VAT.

If you are purchasing a BBC microcomputer, send the two orders together with a cover note requesting that we make the necessary modifications.

Supply of software to commercial users is subject to negotiation.

Description of the Terminal Program

Running the Program

Unless the micro is to be used only as a terminal, some familiarity with the BBC microcomputer is required and so will be assumed here.

The program is run by executing the command '*RUN PHX' for the cassette versions and '*PHX' for disc or ROM versions. The program is then loaded into RAM starting at location &1900 (& will be used to signify hexadecimal) and executed - the execution address is also &1900. If the message 'Not connected/rotate plug' is displayed then the CTS and RTS lines on the RS423 interface are not connected together. This is usually caused by the interface cable not being connected or the RS423 plug is upside down - it has two-fold rotational symmetry and can be inserted in either of two orientations. When the correct connections are made a title and version number are displayed. The machine is then in terminal mode

The BREAK key causes the program to terminate and BASIC is re-entered. Note that the RAM area &1900-&2FFF is used by the program and &3000-&7FFF by the screen so BASIC programs or data may be corrupted.

From now on the 'break key' does not refer to the hardware reset key labelled 'BREAK' but the special function key, f0, which sends a break down the Phoenix line and is akin to the break key on other terminals.

Terminal Mode

Note that all messages generated by the BBC microcomputer's operating system or the program are displayed in inverse video. All characters received from the Phoenix line are in normal video. The pound key generates the ASCII code for a grave accent and for compatibility with Phoenix a grave is displayed whenever this code is received from the line. However, for compatibility with the BBC microcomputer system it is displayed as pound sign under all other circumstances (eg in a file name) and will appear in inverse video. All pound signs will be turned into graves when sent to Phoenix.

When in terminal mode all keys behave like those on a terminal except the special function keys and the cursor control keys. 'Clear' (shift-f0) clears the screen and 'BREAK' (f9 or shift-f9) issues a line break. The cursor control keys and the COPY key (khaki key caps) behave in the normal way for a BBC microcomputer - a secondary cursor can be moved around the screen with the cursor controls and the character immediately above the cursor is appended to the current input line when the COPY key is depressed.

The special function keys f0-f9 may be programmed to contain commonly used strings. See the section on system commands.

Other modes of operation of the program may be entered using the shifted special function keys.

Graphics Mode

the terminal program has graphics capabilities when used in conjunction with the VIEW program on the IBM3081. The device name supplied in the OPT string to VIEW is SBBCMICROn where n can take the values 2, 4, 8, or X (or can be omitted to give the default of 4) and is the number of available colours (X is 16). The pen numbers 0-15 refer to the colours

0	White (background colour of this device)
1	Black
2	Red
3	Green
4	Blue
5	Yellow
6	Magenta
7	Cyan
8	Flashing black/white
9	Flashing red/cyan
10	Flashing green/magenta
11	Flashing yellow/blue
12	Flashing blue/yellow
13	Flashing magenta/green
14	Flashing cyan/red
15	Flashing white/black

Colours 8-15 do not follow the Cambridge conventions. Only 2, 4 and 8 colour modes give standard results.

Note that the string 'SBBCMICRO' is conveniently held in shift-f1.

The number of colours available is traded off against resolution - the resolution is 640x256 for 2 colours, 320x256 for 4 colours and 160x256 for 8 and 16 colours.

The screen is cleared back to 80 column text by the 'Clear' key.

File Transfer Mode

Text files may be transferred between the microcomputer and the IBM3081. The 'Send file' key (shift-f2) initiates sending a file to the IBM and the 'Get file' key (shift-f3) causes a file to be transferred down to the micro. Pressing either key causes the program to prompt for both the Phoenix and BBC microcomputer file names. Answering either of these questions is aborted using the 'Escape' key which is enabled at this stage and the terminal mode is re-entered. Note that the BBC microcomputer file name refers to the currently logged in filing system which may be changed using a system command (q.v.). Only the disc and cassette filing systems are supported at present.

Only text files are permitted for transfer (i.e. not SAVEd BASIC programs etc.) and may only contain the ASCII printable characters (codes &20-&7E), tab (&09) and linefeed (&0A). BBC microcomputer files for shipping up to the mainframe should be in byte rather than string format (i.e. generated by BPUT# or *SPOOL or *BUILD but not PRINT#), should have lines seperated by carriage return (&0D) and the line length should not exceed the logical record length of the Phoenix file and anyway should not exceed 255 minus the number of '@' characters in the line. '@' is treated textually and is automatically expanded to '@@' before transmission.

The line length for files transferred from the IBM should not exceed 255 characters and should not have significant carriage control characters.

The program attempts to detect all errors and if it cannot recover safely will abort with a message indicating what the error was. The 'BREAK' key will abort a transfer prematurely - all other keys have no affect in this mode. A break is sent out on the line to halt further output or to exit from INPUT. However, the terminal status will be left with TERMINAL WIDTH 256, TERMINAL NOMESSAGE and %PWARNLEVEL set at 255.

After the transfer is completed or aborted the terminal mode is re-entered.

Note that a motor control is essential on cassette recorders.

System Command

The 'System' key (shift-f4) prompts for a line of text which is then sent to the BBC microcomputer's operating system command line interpreter. Any "star" command (e.g. *FX, *KEY, *CAT etc.) may be executed. The 'Escape' key is enabled during this mode so that *CAT for the cassette filing system can be terminated!

This facility is provided to change the transmit and receive baud rates from their default values of 1200 (*FX7 & 8), programming the special function keys f0-f9 (*KEY), changeing the filing system (*DISC, *TAPE) and cataloguing files (*CAT). Certain commands will have catastrophics effects, e.g. most of the *FX commands will interact with the program in unpredictable ways and some disc commands will overwrite the program in RAM (e.g. *COMPACT, *COPY etc.)

It is possible to type control codes on the keyboard while entering the command line which are forwarded to the VDU driver. Thus things like the mode and colour of the screen can be changed but this feature was not deliberately implemented and should be treated with caution!

Updates

```
HASP-II***HASP-II.......END JOB  138.....4.28.24 PM  2 FEB 83....P
```

```
=======
     No updates - current version is the first release.
```

Member BLANK

```
KeyLname                       Lname                     Lname                     Lname
 0 0                           0                         0                         0
 1 0                           0                         0                         0
 2 0                           0                         0                         0
 3 0                           0                         0                         0
 4 0                           0                         0                         0
 5 0                           0                         0                         0
 6 0                           0                         0                         0
 7 0                           0                         0                         0
 8 0                           0                         0                         0
 9 0                           0                         0                         0
```

Member INFO

This file contains a program for plotting a 'birdcage' for labelling the BBC micro special function keys. The program is member PROG. To execute it use 'C CAMPLOT.C:RUN' and answer the questions. The data must be in the correct format - see the data files in this PDS for examples.

The members of this file are:

```
 INFO - this information
 PROG      - the program for plotting the birdcages
 BLANK     - data for producing a blank birdcage
 PHX004    - data for the Phoenix terminal program version 0.04
 PHX005    - data for the Phoenix terminal program version 0.05
```

Member PHX004

```
KeyLname                       Lname                     Lname                     Lname
 0 0                          10Clear scrn               0                         0
 1 0                           9SBBCMICRO                0                         0
 2 0                           9Send file                0                         0
 3 0                           8Get file                 0                         0
 4 0                           6System                   0                         0
 5 0                           0                         0                         0
 6 0                           0                         0                         0
 7 0                           0                         0                         0
 8 0                           0                         0                         0
 9 5BREAK                      5BREAK                    0                         0
```

Member PHX011

```
KeyLname                       Lname                     Lname                     Lname
 0 0                          10Clear scrn               0                         0
 1 0                           9SBBCMICRO                0                         0
 2 0                           9Send file                0                         0
 3 0                           8Get file                 0                         0
 4 0                           6System                   0                         0
 5 0                           9Scrn dump                0                         0
 6 0                           0                         0                         0
 7 0                           0                         0                         0
 8 0                           0                         0                         0
```

Specification of BBC micro RS423 cable

The cable described here is the standard cable provided by the Microprocessor Support Service for connection to University Data Network and other serial lines at a cost of 5 pounds. Orders should be made out to "The Computer Laboratory", and sent to Mick Seaman at the Computing Service wherever possible an Internal Debit Note will be issued rather than an invoice so that VAT need not be paid.

Cable - 6 core screened 7/0.1 (BS 367-375), 2 metres long

Connectors - 25 way D type plug, plastic headshell
5 way "domino" DIN plug

Assembly - The core colours of the cable are Red, Blue, Green Yellow, White and Black. White is not used and is trimmed back at both ends of the connector as is the screen.

The D-25 plug is connected as follows:-

Pin 2	- Yellow	Transmit data from BBC (DTE)
3	- Green	Receive data
4	- Blue	Request to send (from BBC)
5	- Red	Clear to send (to BBC)
7	- Black	Common Return

Pins 2,3 and 7 are covered with rubber sleeving. Pins 4 and 5 are connected together with an extra wire inside the headshell. The cable is covered with a rubber sleeve at the point at which the cable clamp acts to prevent cable damage and connection breakage.

The 5 pin "domino" DIN plug is connected as follows:-

Pin 1 (center)	- Black
2	- Red
3	- Yellow
4	- Green
5	- Blue

Each pin is individually sleeved and the whole cable is sleeved within the plastic cover. The cover is inscribed "TOP" in line with the location cut out (using a soldering iron).

Use - The cable can be inserted into the RS423 socket at the back of the machine in 4 orientations (daft). It should be inserted with the locating cutout at the top. If it is inserted upside down the standard communications program (see MICROLIB.BBCMICRO.TERMINAL) detects this and prints a message.
Clear to Send (CTS) and Request to Send (RTS) are connected inside the D 25 plug headshell as required for UDN connections where these signals are not used. The wire making the connection made need to be cut for use on other machines.

HASP-II***HASP-II.......END JOB 172.....4.31.53 PM 2 FEB 83....POST

```
MPT2ooo J0172MPT2ooooooooooooooooooooooooooooooooooHASP-II***HASP-II
```