

# **PRES**

PO Box 319, Lightwater,  
SURREY GU18 5PW

## **Advanced Plus 7 User Guide**

Copyright (C) 1988 PRES Ltd  
All rights reserved

# CONTENTS

---

How to fit the AP7 module	3
Electron	3
Master	3
Master Compact	4
BBC A and B	5
How to use AP7	7
Software utilities	7
Saving the programs in AP7	7
Saving ROMS onto Tape or Disc	7
Loading ROMS into AP7	8
Zero RAM	8
Lock	8
Unlock	8
Printer Buffer	8
MakeROM	9
Using AP7 from Machine Code	12
Write Protecting AP7	13
LOCK and UNLOCK from Machine Code	13
The Internal Battery	14
Links	14

## WARNING

**Save the software first!** This unit contains software. Please read the following instructions carefully before using the cartridge. (See section on 'Saving the programs in AP7' before loading data into the cartridge.)

**NOTE: Keep a backup copy of your ABR utilities.**

## HOW TO FIT THE AP7 MODULE

---

### ELECTRON

The AP7 has been designed to work with the AP6. It will not work with other ROM expansion boards and no support can be given. If AP2 is fitted, it should be version 1.22 or higher.

1. Fit the **orange, yellow and blue** leads into the **four and two** way connector housings as shown in **fig 1** and push onto the connector block, again shown in **Fig 1**.
2. Change the **jumpers** to their correct positions as shown in **Fig 1**. Two only at this stage.
3. The remaining three **jumpers** can be left as shipped, **jumpers** on KL MN and OP, see **Fig 5**. For more information on these **jumpers**, see page 12.
4. Turn off the power to your Electron and remove any cartridges that you may have fitted. Remove the Plus 1 from the Electron by removing the two large fixing screws. Remove the lid of the Plus 1 by removing the three fixing screws in the base.
5. The AP7 will fit into either socket 6 or 15. The correct orientation is with the notch of the ICs on the AP7 facing the back of the Plus 1, the same way you would normally fit a ROM or EPROM. **Fig 7** shows how ICs are marked and numbered.
6. Connect the **two** way housing to the two nearest pins on the AP6, see **Fig 6**. (Pins marked A)
7. Reassemble the equipment and turn on. Do not forget to save the software that is contained in the AP7.

### MASTER

1. Disconnect the power lead from the mains outlet. Remove any peripherals that are connected to the computer. Turn the computer upside down and remove the four screws marked **fix**. Carefully re-invert the computer, taking care to hold the two halves together. Remove the upper half of the case and place face down on a firm surface.
2. Check link 19, located to the left of the IC 37, the third IC north of the cartridge slots at the right hand side of the circuit board. It should be in the **east** position. If not, carefully remove it and replace it in the correct position.
3. The **red** lead, supplied with the AP7, is dual length. The short version is needed for the Master so cut off the excess but leave the crimped on terminal. **Double check before cutting**. You need to be left with a length of wire with a small crimp terminal on one end and the miniature clip on the other.

4. Fit the **orange, red and black** leads into the **four** way connector housing as shown in **Fig 2** and push onto the connector block, again shown in **Fig 2**.
5. Change the **jumpers** to their correct positions as shown in **Fig 2**. Two only at this stage.
6. The remaining three **jumpers** can be left as shipped, **jumpers** on KL MN and OP, see **Fig 5**. For more information on these **jumpers** see page 12.
7. Fit the AP7 into the socket marked IC 27, the third socket up on the right hand side of the computer, counting from the cartridge slots. The notch or pin one markings on the AP7 should be aligned with the notch on the IC socket. **Fig 7** shows how ICs are marked and numbered.
8. Connect the **red** lead to **pin 20** of IC 15 and the **black** lead to **pin 29** of the same IC. IC 15 is located 2" west and 1.5" north of the socket that AP7 uses. **NOTE:** The **black** wire connected to **pin 29** of IC 15 can foul on the lid mounting. To avoid this, lean the chip **east** over the small relay.
9. Replace the upper half of the computer and refit the four **fix** screws, the longer screws being fitted in the rear positions. Do not forget to save the software contained in the AP7.

#### **MASTER COMPACT**

1. Disconnect the power lead from the rear of the keyboard. Remove any peripherals that are connected to the keyboard. Turn the computer upside down and remove the four screws as marked on the serial number label. Carefully re-invert the computer, taking care not to hold the two halves together. Remove the upper half of the case and move to the left hand side as far as possible.
2. The **red** lead, supplied with the AP7, is dual length. The short version is needed for the **Master Compact** so cut off the excess but leave the crimped on terminal. **Double check before cutting.** You need to be left with a length of wire with a small crimp terminal on one end and the miniature clip on the other.
3. Fit the **white, red and black** leads into the **four** way connector housing as shown in **Fig 3** and push onto the connector block, again shown in **Fig 3**.
4. Change the **jumpers** to their correct positions as shown in **Fig 3**. Two only at this stage.
5. The remaining three **jumpers** can be left as shipped, **jumpers** on KL MN and OP, see **Fig 5**. For more information on these **jumpers** see page 12.
6. Move the AP7 into a position so that the small pin on the end of the white wire can be plugged into **Pin 22** of IC 23. See **Fig 7** for a graphical view of the marking and pin numbering of ICs. IC 23 is located, measured from the south east corner of the printed circuit board, at 3.25" north and 1.75" west.

7. Fit the AP7 into the socket marked IC 17, the location of this socket, measured from the south east corner of the printed circuit board, is 4.25" north and 1.75" west. The notch or pin one markings on the AP7 should be aligned with the notch on the IC socket. **Fig 7** shows how ICs are marked and numbered.
8. Connect the **red** lead to **Pin 20** of IC 22 and the **black** lead to **Pin 29** of the same IC. IC 22 is located immediately west of IC 23.
9. Replace the upper half of the keyboard and refit the four fixing screws. Do not forget to save the software contained in the AP7.

**BBC A and B (NOTE: Not suitable for Model B+)**

1. Disconnect the power lead from the mains outlet. Remove any peripherals that are connected to the computer. Turn the computer upside down and remove the two fixing screws nearest to the front edge and the two on the back of the computer to release the lid. Carefully re-invert the computer, taking care to hold the two halves together. Remove the upper half of the case and place face down on a firm surface. Now remove the two sets of screws and nuts that hold the keyboard in place. Position the keyboard so that the IC sockets at the right hand side of the main circuit board can be accessed.
2. The **red** lead, supplied with the AP7, is dual length. The long version is needed for the BBC A and B, so you will need to insulate the crimped terminal in the centre of the lead. Use a small piece of insulating tape or sellotape.
3. Fit the **white**, **red** and **black** leads into the **four** way connector housing as shown in **Fig 4** and push onto the connector block, again shown in **Fig 4**.
4. Change the **jumpers** to their correct positions as shown in **Fig 4**. Two only at this stage.
5. The remaining three **jumpers** can be left as shipped. **Jumpers** on KL MN and OP, see **Fig 5**. For more information on these **jumpers**, see page 12.

**NOTE:** The AP7 should be fitted into either IC socket IC 88 or IC 101, the first or third socket counting from the right hand side.

6. Move the AP7 into a position so that the small pin on the end of the white wire can be plugged into **Pin 20** of IC 52, if IC 88 is used, or **Pin 20** of IC 100, if IC 101 is used. See **fig 7** for a graphical view of the marking and pin numbering of ICs.
7. Fit the AP7 into the chosen socket, either IC 88 or IC 101. The notch of pin one markings on the AP7 should be aligned with the notch on the IC socket. **Fig 7** shows how ICs are marked and numbered.

Connect the **red** lead to **Pin 8** or IC 77 and the **black** lead to the + end of D14. IC 77 is located 1.5" east and 3" south of the north west corner of the circuit board. D14 is the southerly most diode of a pair located south and slightly west of the row of IC sockets that AP7 is fitted into.

9. Replace the keyboard and the upper half of the computer using the two sets of screws and nuts and four fixing screws respectively. Do not forget to save the software contained in the AP7.

**PLEASE NOTE**

It has not been possible to test AP7 with the  
large number of ROM/RAM expansion boards.

## HOW TO USE AP7

---

After fitting, the RAM in AP7 will be mapped into the computer's memory as two of the 16 sideways ROMs. The ROM numbers of AP7 will depend upon which socket in the computer the AP7 has been fitted. (See below.) You will need to know which ROM number AP7 is in when loading software into the RAM.

<u>ELECTRON</u>	AP7 will be mapped into either sockets 5 & 6 or 14 & 15.
<u>MASTER</u>	AP7 will be mapped into sockets 6 & 7.
<u>MASTER COMPACT</u>	AP7 will be mapped into sockets 2 & 3.
<u>BBC A and B</u>	AP7 will be mapped into either sockets 12 & 13 or 14 & 15.

### ABR software utilities

The new PRES utilities are supplied in a ROM filing system image loaded in one of the AP7 banks when it is purchased. This image contains seven utilities, plus an additional program to transfer the utilities onto tape, DFS or ADFS to free the AP7 bank for other uses.

### Saving The Programs In AP7

To transfer the programs you should first prepare your media, e.g. making sure a tape is in the cassette recorder, or that a correctly formatted disc (DFS or ADFS) is in the drive.

When the media is ready type \*ABR and press <RETURN>. This will cause the ROM filing system to be selected and the transfer program to be executed. It will ask you which filing system the programs are to be saved on. Press T, D or A as appropriate.

You should also keep a copy of the whole suite of utilities, as supplied in the AP7. This may be achieved by using SaveROM which is one of the programs which has just been transferred.

If using ADFS it can be a good idea to put some of the utilities into the library directory then they can be used from any directory, assuming the library directory has been set up with the \*LIB command.

There are seven utilities supplied with the AP7: SaveROM, LoadROM, Zero, Lock, Unlock, PrntBuf and MakeROM which are described in the following sections.

### SaveROM

This program is used to save a RAM/ROM image to the current filing system.

*Syntax:*

\*SaveROM <filename> <bank> (S). (OR from TAPE \*/SaveROM <etc.>)

The bank number must be included, and must be in hexadecimal.

The program defaults to a fast algorithm which corrupts main memory. If you want to preserve the memory you should type S after the bank number, which uses a slow algorithm leaving memory intact.

### **LoadROM**

This program is used to load a RAM bank with an image stored on the current filing system.

#### **Syntax:**

\*LoadROM <filename> <bank>. (OR from TAPE \*/LoadROM <etc.>)

Bank number is optional, but must be in hexadecimal if present.

If no bank number was given, the program will test each bank, working down from bank F to find ones which are RAM. If the bank is occupied it will ask if you want to overwrite the image currently loaded into it. If the bank is empty the image will be loaded into it.

If the bank number specified is already in use you will also be asked whether you want the present occupant overwritten.

### **Zero**

This program is used to wipe a RAM bank, i.e. to tell the OS that it is no longer present, and to zero the bank.

**NOTE:** In some cases, a <CTRL><BREAK> will be necessary after using this program to completely disengage links to some programs. This applies essentially to filing systems.

#### **Syntax:**

\*Zero <bank> (OR from TAPE \*/Zero <bank>)

It has one parameter, a bank number in hexadecimal, which must be present.

### **Lock**

The RAM in AP7 can be made to behave like the memory in ROM, i.e. the memory can only be read and not written back to. This can be done by write protecting the RAM. This is useful when programs which normally run from ROM try to write back to themselves as a means of software protection.

This program is used to turn on the write-protect facility in the AP7. Its parameters are one or more bank numbers in hexadecimal. If no parameters are given, all AP7 banks will be locked.

**NOTE:** Although the programs lets you enter any of the sixteen permissible bank numbers, the design of the AP7 is such that the only thing of consequence is whether it is an odd or even number. The effect of this is that if you have an AP7 & ABR fitted, the odd banks will always be locked or unlocked together (whichever odd bank number you entered) and the same will apply to the even banks.

#### **Syntax:**



\*Lock <rom of bank no.> (OR from tape \*/Lock <rom or bank no.>)

AP7 will normally be locked at switch on:-  
although this is not guaranteed

### **Unlock**

This program is used to turn off the write-protect facility in the AP7. Its parameters are one or more bank numbers in hexadecimal. If no parameters are given, all AP7 banks will be unlocked.

*Syntax:*

\*Unlock <rom or bank no.> (OR from tape \*/Unlock etc>)

See the note above, which applies here also.

### **Printer Buffer**

This program is a sideways RAM image, which provides a printer buffer, to supplement the meagre one used normally. Before it can be used it must be loaded into a RAM bank by using LoadROM.

The program responds to \*PB, or \*PPB to avoid any possible clashes.

There are five commands, which may be entered on the same line as a \*PB or \*PPB:

F - Turn off buffer program (only allowed if buffer is empty)  
N - Turn on buffer program  
C - Clear buffer's contents (only allowed if on)  
P - Purge buffer if <ESC> is pressed (only allowed if on)  
K - Keep buffer intact if <ESC> is pressed (only allowed if on)

Any number of parameters may be included on the same line, e.g. \*PB FK. If a parameter is not recognised a list of the parameters will be displayed. If an unknown parameter occurs in a multi-parameter line, all preceding parameters will not be processed.

\*PB and \*HELP PB or \*HELP PPB produce a buffer status report, and the list of commands. The buffer status report gives you four pieces of information about the program, a) whether it is on or off (the remaining reports will be omitted if the buffer is off), b) the buffer size (14K in 16K image, or 6K in 8K), c) whether the buffer is empty, and d) whether the <ESCAPE> key will cause the buffer to be cleared.

### **MakeROM**

This program is used to frame one or more files in a ROM Filing System image, for loading from a RAM bank (i.e. the way these utilities were supplied).

The ROM Filing System is activated by \*ROM, and behaves like the Tape system, except that only reading is permitted.

This program is a sideways RAM image, and must be loaded into a RAM bank using \*LoadROM.

Start the program with \*MakeROM or \*PMakeROM. Either form of the name may be abbreviated by putting a full stop after the characters.

The program can only be run in the host processor, so Second Processor users should run this program with the Tube off. If a Tube is active the program will refuse to respond to \*MakeROM/\*PMakeROM.

Once the program has been started, it will ask you for a title for the image about to be created. This name will appear on \*ROMS as the title of the ROM, and will also appear on \*HELP. It can be up to 20 characters long, and should only include alphanumeric characters.

You will then be presented with a list of options to choose from, which are:

- T Select Tape and then load a file
- D Select DFS and then load a file
- A Select ADFS and then load a file
- R Load a file from current filing system
- \* Operating system command
- F Start a fresh image
- S Save the current image
- Q Quit the program

Options <T>, <D> and <A> will select the relevant filing system, and then ask for a filename (<R> just asks for the name, keeping the current filing system). The filename must be just a filename, of one to ten characters. It cannot include any kind of directory specification or other special characters.

The file will then be read and converted into ROM filing system format. An error message will be displayed if the file would cause the image to overflow, and the file will not be included.

Once the conversion has been completed or aborted the option list will be displayed again.

<\*> is provided for Operating System commands to be entered. Care should be taken that memory is not corrupted, otherwise the image being built may be overwritten. When the OS command has completed, press the <RETURN> key and the option list will be displayed again.

<F> starts the program again with an empty image. The program will ask you to confirm that you want to start afresh by pressing Y. Press <ESCAPE> if you have second thoughts.

<S> saves the image to the current filing system. You will be asked for a filename.

<Q> leaves the program. You will be asked to confirm that you want to leave by pressing Y, or <ESCAPE> if you have had second thoughts. When you leave MakeROM you will be returned to BASIC.

The image produced by this program may be loaded into a RAM bank by using \*LoadROM.

Normally files in a ROM filing system image cannot be executed with \*name but must be run with \*/name or \*RUN name. Images produced by this program **will** permit programs to be run by \*name.

It is possible to have a !BOOT file in an image which can be started by <SHIFT><SPACE><BREAK>. However, this program must be in machine code, and is called before Basic is started. To make the facility more useful a program called !Boot is included in the AP7. This program starts up Basic and then CHAINS a program called BOOT. Thus to set up an auto-boot sequence you should include the !BOOT provided and a Basic program BOOT of your own.

## USING AP7 FROM MACHINE CODE

---

### Reading bytes from AP7

The RAM in AP7 can be read like any other sideways ROM.

There is a routine in the computer's Operating System which can be used to read a byte from any of the ROMs, including AP7. This routine is called OSRDRM and its address is &FFB9. The following assembly instructions could be used to read a byte from AP7.

```
.program
    LDX #address MOD256
    LDY #address DIV256
    JSR read_AP7                byte read from AP7 returned in A
    :
    :
    rest of program

.read_AP7                      subroutine to read a byte from AP7

    STX &F6                    initialise low byte address
    STY &F7                    initialise high byte address
    LDY #ap7rom                Y=ROM number of AP7
    JMP OSRDRM                 call MOS to read a byte from AP7
```

### Writing bytes to AP7

Unlike OSRDRM there is not an Operating System routine that can be used to write bytes to Sideways RAM. The following assembly program can be used to write a byte to AP7.

```
.program
    LDX #address MOD256
    LDY #address DIV256
    LDA byte
    JSR write_AP7              write byte in A to address X,Y in AP7
    :
    :
    rest of program

.write_AP7                     subroutine to write a byte to AP7
    STX &F6                    initialise low byte address
    STY &F7                    initialise high byte address
    TAX                       save data in X
    LDA &F4
    PHA                       save current ROM on stack
*   LDA #12                   must first deselect Basic
*   STA &F4                   (always use bank 12 when
*   STA romsel                AP7 is fitted)
    LDA #AP7_rom
    STA &F4
    STA romsel                now page in AP7 ROM
    TXA                       restore data in A
    LDY #10
    STA (&F6),Y              store byte to AP7
    PLA                       get old paged ROM
```

```

STA &F4
STA romsel          select current ROM
RTS

```

\* Electron only

(It is recommended that Basic is only deselected by ROM 12)  
romsel - Electron = &FE05, Master and Compact = &FE30

**NOTE:** Writing to AP7 can only be done after the RAM has been unlocked. See Write Protecting.

### Write Protecting AP7

As mentioned previously the RAM in AP7 can be made to behave like the memory in ROM. Two memory addresses are used to lock the RAM and two addresses are used to unlock the RAM.

```

Lower bank  = UNLOCK - &FCD8
              LOCK   --- &FCD9
High bank   = UNLOCK - &FCDA
              LOCK   --- &FCDB

```

For example:

**ELECTRON and BBC A or B** to lock the RAM, write to addresses &FCD9 or &FCDB e.g.. ?&FCD9=0 (lower): ?&FCDB=0 (higher) or use the utility LOCK.

AP7 in the **MASTER and MASTER COMPACT** needs the following extra statements as the AP7 uses the internal 1MHz bus:

```
A%=?&FE34: ?&FE34=?&FE34 OR &20: ?&FCD8=0: ?&FCDA=0: ?&FE34=A%
```

(Entries 3 and 4 will unlock both upper and lower banks of RAM.)

### LOCK and UNLOCK from Machine Code

This routine can be used in your own machine code programs:

```

.program
    LDA &FE34          read ACCCON register
*    PHA              save
*    ORA #&20         set bit 5 for cartridge access
*    STA &FE34        save to ACCCON register
    STA @@@@         lock/unlock RAM
*    PLA              recover
*    STA &FE34        restore ACCCON register
    :
    :
rest of program.

```

\* Master and Master Compact only.

where @@@@ is the address to either lock or unlock the RAM, for example, STA &FCD8 to unlock lower RAM bank.

## Internal Battery

The AP7 is fitted with a nickel-cadmium rechargeable battery which is recharged when AP7 is used in the computer.

## Battery Life

If the AP7 module is used regularly, for example, at least 2 hours a week, then the life of the battery will be greater than 5 years. If used less than 2 hours a week, recharging should be carried out at intervals of not less than 3 months. A full recharge period takes 14 hours. See the following section on **link** options if continuous usage is greater than say 24 hours.

**NOTE: The battery is not user-replaceable.** In the event of any technical support or queries please contact PRES service department for advice (Tel. 0274 580519)

## Links

There are **three** extra links on AP7 that are user configurable. See **Fig 5**. They are KL MN and OPQ.

KL - In place when shipped. This link controls the charging rate of the battery. When fitted, it provides a high charging rate suitable for intermittent use of the AP7. If the AP7 is to be powered up continuously (e.g. every three days or more) then link KL needs removing.

MN - In place when shipped. This link, when removed, will remove all power to the RAM. Useful if a corrupt image has been installed in the RAM or if, in the unlikely event of a ROM image has been corrupted. **Warning:** the contents of both RAM banks will be lost if link MN is removed.

OPQ - Fitted to OP when shipped. This link is for technical service only.

**Warning:** Do not adjust this link unless you have been instructed to do so. If the link is set in the wrong position, erratic operation could result from your computer.



PRES LTD, PO Box 319, Lightwater, SURREY GU18 5PW

Tel: 0276 752046. Fax: 0276 51427

