About This Remade Document

Title: Master Series Compact Application Note, Issue D
Last remade: 30-Mar-2025
Repository: https://github.com/acheton1984/AcornDocsRemade

This is not quite the beautifully "remastered" document as seen elsewhere, but it is intended to be very similar to the original in an easy-to-read and searchable PDF format.

Reconstruction Notes

Source scan: http://8bs.com/othrdnld/manuals/essentialsmaster.shtml

* Mention of this was first spotted on a Stardot thread - https://www.stardot.org.uk/forums/viewtopic.php?p=439204#p439204
* Recreated from a scan of the original document.
* Font sizes, margins, line spacing, and character spacing are only approximate. Some words may appear on different lines, but the overall layout and pagination are preserved.
* Spelling errors have been corrected where spotted.
* A title and remade document history page have been added, along with a link to this repository.

Disclaimer

The underlying content is reproduced or adapted from the original documents for educational purposes and ease of access.

All rights remain with their respective owners.

---- original document follows ----

Acorn Computers Ltd.
British Broadcasting Corporation
Master Series Compact

Scope:

This document is intended to explain the differences apparent to the user and applications software writer between the Master Series 128 and the Compact computer. It covers most, but not necessarily all, of these differences.

Hardware:

Interfaces:

The following interfaces on the Master 128 are deleted or changed:

Cassette	- Connector and internal hardware deleted.
Tube	- Connector and internal hardware deleted.
1 Mhz bus	- Connector and internal hardware deleted.
User I/O port	- Connector deleted. The internal 6522 User VIA connections to the original 20 pin connector are now split as follows: Joystick/mouse - PB0 thru PB4 + 2 ctrl bits. Expansion port - PB5 thru PB7.
Disk	- Now 25 Pin D-type socket. Note that there is no hardware support for a third drive
Printer	- Now 24 Pin Delta-ribbon socket.
RS 423	- Now optional and RS 232 specification. The upgrade consists of plugging in four IC's (5, 9, 13 & 14).
A to D	- Connector and internal hardware deleted. (See Joystick/Mouse port).
Audio (external)	- Connector and internal hardware deleted.
Composite Video	- Now Black/White only. Cannot be colour.
TV	- Connector and internal hardware deleted.
Cartridge sockets	- Connectors deleted. Potential capability through the expansion port.
Internal Modem	- Internal connector deleted.
Aux power out	- Connector deleted. No PSU in computer case.

The following interfaces are added on the Compact Computer:

Joystick/Mouse	- Suitable for one digital joystick (Atari compatible) or Mouse with suitable pinning. A Trackball can also be used.
+5V DC	- Power input to the computer.
Expansion Port	- This interface is similar, but not identical to a Master 128 Cartridge socket. It can support Sideways ROM's 0 & 1 when link PL 11 is set North. A 2Mhz bus is provided by this port as in the Master 128. The port must be used with care as lines are not necessarily buffered. Only a limited amount of +5V power is available, and demand should be kept below 200mA total for this connector, the RGB connector and the Joystick/Mouse port.

Functions:

In addition to the functional changes implied by the interface changes mentioned above, also note:

Real-time clock	- Deleted.
CMOS RAM	- Deleted. Function replaced by an EEPROM device which does not need battery back-up. This device is socketed and has a maximum

number of 10000 write cycles per location.

Sideways ROM sockets - In addition to the system ROM, there are four 28 pin sockets. Three take 16K ROM's:

IC 23 - ROM number 2
IC 17 - ROM number 3
IC 29 - ROM number 8

and one takes a 16K or 32K (IC 38, ROM numbers 0 & 1). The latter socket must be enabled by setting link PL 11 South. It is normally set for the "external" ROM(s) to be active for test purposes.
Note that "Paged" EPROM's such as the 27513 and 27011 cannot be used internally.

System ROM - Link PL 12 is set North for a 64K EPROM, and South for a 128K ROM.

Links (misc):

Inverse Video - PL 9 (not fitted) is normally tracked East.

Inverse Sync - PL 10 (not fitted) is normally tracked East

Sound volume - VR 1 (10K) may be fitted.

Expansion Port pinout

(See Master Cartridge Application Note for detail)

SOLDER SIDE

PIN	PROJ - BB	MASTER 128
1a	SCREEN (0V)	+5V
2a	+5V	AT13
3a	AT13	(neg) RST
4a	(neg) RST	AA15
5a	AA15	A8
6a	A8	A13
7a	A13	A12
8a	A12	PHI 2 out
9a	PHI 2 out	-5V
10a	N/C	(neg) CSYNC/0
11a	N/C	BR/(neg)W
12a	BR/(neg)W	(neg)NMI
13a	(neg)NMI	(neg)IRQ
14a	(neg)IRQ	(neg)INFC
15a	(neg)INFC	(neg)INFD
16a	(neg)INFD	AA14
17a	AA14	(neg)8/16Mhz
18a	(neg) 8Mhz	CRTC (neg)RST
19a	0V	ANOUT
20a	PB7 USER	AGND
21a	PB6 USER	SPEECH
22a	PB5 USER	0V
POLARISATION SLOT		
24a	0V	
25a	SCREEN (0V)	

COMPONENT SIDE

PIN	PROJ - BB	MASTER 128
1b	SCREEN (0V)	+5V
2b	+5V	A10
3b	A10	CD3
4b	CD3	A11
5b	A11	A9
6b	A9	CD7
7b	CD7	CD6
8b	CD6	CD5
9b	CD5	CD4
10b	CD4	LPSTB
11b	LPSTB	BA7
12b	BA7	BA6
13b	BA6	BA5
14b	BA5	BA4
15b	BA4	BA3
16b	BA3	BA2
17b	BA2	BA1
18b	BA1	BA0
19b	BA0	CD0
20b	CD0	CD2
21b	CD2	CD1
22b	CD1	0V
POLARISATION SLOT		
24b	0V	
25b	SCREEN (OV)	

Remade from a scan of an original document. This PDF is hosted at:
https://github.com/acheton1984/AcornDocsRemade

Firmware:

ADFS:
1) *DRIVE has been added to the ADFS to assist with compatibility in file conversions from DFS. *DRIVE n is equivalent to *DIR :n. As ADFS only has two drives, if n<4 it is forced to drive 4 or 5. If n>=4 it is rejected. *DRIVE should not be used in new applications.

2) *COPY/*COMPACT/*BACKUP use Shadow RAM if available, and will not corrupt user workspace. If Shadow RAM is not available, the utilities will first consider using unclaimed Filing system RAM, and then finally will force Mode 135. The commands force *FX112,0 to avoid overwriting their own buffer.

*COMPACT no longer takes parameters and ADFS will issue an error message to remind the user that the memory specified will not be used.

3) *FORMAT/*VERIFY/*BACKUP are contained within the ROM.

*FORMAT takes parameters <drv> <siz> where <drv> is the drive number (0 or 1, 4 or 5), and <siz> is S,M or L for 40Tr, 80Tr single-sided and 80Tr double-sided respectively. 40Tr is provided for use only where a 5.25" single-sided 40Tr drive is fitted. The user must ensure that the syntax chosen is suitable for the drive type being used. The use of *FORMAT does not corrupt user workspace i.e. it uses 2 pages of utility workspace at &DD00. Sector skew is now 4 (it was 7 in the Master 128). This results in slightly faster disk performance with the 3.5" drive fitted as standard.

4) OSGBPB calls 6 & 7 return a zero byte after the CSD name or LIBRARY name to be compatible with the ownership byte returned by the Net Filing System.

5) CLOSE#0 no longer produces "Channel on Channel 57" when following an EXEC sequence.

6) Modifications have been made to the Floppy driver software in ROM which results in a noticeable speed-up in disk operation compared with the Master 128.

7) *CONFIGURE FDRIVE now uses write pre-compensation on all four parameter values. This is applied to tracks 32-79 and 112-160. The four FDRIVE step rates are: 0 - 6mS 1 - 12mS 2 - 2mS 3 - 3mS.

8) The 40 Trk limitation which caused OSWORD &72 (*LOAD/SAVE etc} to generate an error when an attempt was made to read the last track of a 40 Trk disk has been removed.

9) The TUBE and Winchester support code has been removed to provide space for the utilities.

10) A head settle has been added to cover the situation when doing a *BACKUP between two 5.25 drives and:
The head is on the right track, the other drive has just been used and the motor is still on. A disk error 48 might otherwise be issued.

11) *BYE now closes all files when in a "No directory" state.

12) *RENAME Wildcards are always rejected.

MOS:
The Operating System is effectively compatible with that of the Master 128. All of the extended graphics features are available as for the Master 128.

1)The Real-time clock is not present, and calls to this will return a year of "1999". i.e. "Fri,31 Dec 1999.23:59:59".

2)The Configuration system is similar for *CONFIGURE and *STATUS, but the latter lists in alpha order. References to Tube/Notube/Extube/Intube have been deleted. New keywords have been added as follows:

SWITCHED - makes stick default to switched mode (0/&7FFF/&FFFF).
Currently affects bit &20 of default *FX190 value.
PROPORTIONAL - makes stick give values in the range 0 thru &FFFF.
Currently affects bit &20 of default *FX190 value.
STICK <decnum> - makes stick have speed <decnum>.
Currently affects bit &1F of default *FX190 value.

The default takes effect after power-up, CTRL + BREAK or BREAK.

An EEPROM is used instead of the Master 128 CMOS RAM. This is normally 128 bytes, but a 256 byte version may be fitted later.

An OSBYTE call with A = 161, X = 255 yields the following:

Y=0 indicates no EEPROM present.
Y=&7F 128 byte EEPROM present.
Y=&FF 256 byte EEPROM present.

Writes to EEPROM address 128 using *FX162 will be ignored. A read from 128 is allowed.

3)The A to D port is not present, and hence analogue joysticks cannot be used. The new digital Joystick/Mouse port is introduced, and this is a sub-set of the previous User Port connections. The User Port is no longer present as such. The connections for this 9 way D-type connector are:

Joystick D-type pins:	6522 connections:
1 Up (-ve true)	(PB3)
2 Down (-ve true)	(PB2)
3 Left (-ve true)	(PB1)
4 Right (-ve true)	(PB4)
5 No joystick connection	(CB1)
6 Fire	(PB0)
7 +5V	
8 0V	
9 No joystick connection	(CB2)

On power-up, CB1 and CB2 interrupts are enabled. A sideways ROM must be present that can process these interrupts if a mouse or tracker-ball are fitted.

When such an interrupt is confirmed, the sideways ROM can set the top bit of OSBYTE variable 190 to disable MOS processing of ADVAL values, then every clock tick, service call &2C is offered sideways. In the Y register is an offset from &0200 to the following workspace:

+0 ADVAL lo-byte	(ADVAL hi-byte from OSBYTE var 188)
+1 Xlo-coord	(x-coord returned as ADVAL1)
+2 Xhi-coord	
+3 Ylo-coord	(y-coord returned as ADVAL2)
+4 Yhi-coord	
+5 spare	
+6 spare	

(This workspace is subject to confirmation)

OSBYTE variables 188 and 189 have their normal meanings.

If the top bit of OSBYTE variable 190 is set, the MOS will not update ADVAL values from the digital joystick or cursor keys. This is designed only for external ROM's wishing to control ADVAL values e.g. mouse/trackerball software. Note that by just setting the top bit of this option, the old value may conveniently be restored by simply resetting the top bit.

*FX190,64 This option enters a key into the keyboard buffer according to bits set in ADVAL0 (lo-byte). The character "typed" is as follows (in order of priority):

bit &80 (right)	cursor right
bit &40 (up)	cursor up
bit &20 (down)	cursor down
bit &10 (left)	cursor left
bit &08	delete key
bit &04	return key
bit &02	copy key
bit &01 (fire)	copy key

The characters are typed with (almost) the same effect as typing them at the keyboard (i.e. within a centisecond or two). Auto-repeat is supported. In this mode ADVAL1 and ADVAL2 will not reflect the state of the "joystick" position.
Bits &08, &04 and &02 are never set by the digital joystick, but may be set if a mouse/trackerball is supported.

*FX190,32
This option affects the digital joystick and ADVAL. It is designed for games that used the analogue joystick as switches. It has the following effect:

ADVAL1	Xleft &FFFF	Xcentre &7FFF	Xright	&0000
ADVAL2	Ydown &0000	Ycentre &7FFF	Yup	&FFFF

*FX190,1 (or,2,3,4,5,6,7)
This is a feature whereby the speed of the analogue simulation of the joystick may be adjusted.

*FX190,1	make left/right & up/down sweeps slow.
(*FX190,2 or ,3,4,5,6 are progressively faster).	
*FX190,7	make left/right & up/down sweeps fast.

"Standard" settings *FX190,0, *FX190,8 & *FX190,12 use the speed selected by *FX190,3

*FX4,3
The *FX4,3 option makes the cursor keys have joystick-like effects:

left cursor	moves joystick left
right cursor	moves joystick right
up cursor	moves joystick up
down cursor	moves joystick down
copy key	makes joystick fire

The state of the real joystick and cursor keys (in this mode) are ORed together. This has the primary advantage that either the real joystick or the cursor keys may be used to affect ADVAL values. When this option is selected, pressing a cursor key does not enter a code into the keyboard buffer. If a value is "poked" into the keyboard buffer, RDCH will assume the code to represent a softkey (rather like *FX4,2). If a mouse or trackerball is connected, this option has no effect (the mouse/trackerball takes priority).

(See BASIC section below for ADVAL implications).

4) Some TUBE code has been removed, but:
a) The TUBE flag accessible from OSBYTE 234 remains and indicates No TUBE.
b) Service call &FE remains.
The command *X (controls an external Tube splitter) has been removed.

5) SRAM utils and Ellipse code are now within the MOS ROM area. A bug with long, thin ellipses has been fixed. A facility to load an SRAM image and update the MOS ROMtype table has been added. An "I" should be added to the *SRLOAD command.

6) *BUILD/*APPEND now allow top-bit set characters to be input.

7) The keyboard layout has been changed as follows:
a) The "@" character has been moved to the "Shifted-0" position as for the Electron. SHIFT + 0 gives "@" (&40), and CTRL + 0 gives NUL (&00).

b) The key position previously used for the "@" character is now used for "CODE" input and is marked with two squares (set vertically). The use of CTRL + SHIFT + CODE preceding any ONE key stroke will cause that key stroke character to be entered with the top-bit set.

Top-bit set characters must not be used within file names.

8) The first call of JSR BREAK in the MOS to allow break indirection has been changed to preserve ROMID.

9) INKEY-256 now returns 245 (&F5).

10) *FX16,0 suppresses ADVAL support as usual, and reduces interrupt processing overhead accordingly. The default number of channels has been altered to 2.

11) Key interpretations set by *FX 221 thru 228 have been extended:
Options 0 and 1 of *FX 22x remain as before. The meaning of value 2 has been changed. It used to mean "use 2 as a base". It now means return a code representing the key preceded by a NUL. For example:

NUL &80 means f0	(if *FX225,2 set)
NUL &89 means f9	(if *FX225,2 set)
NUL &91 means SHIFT + f1	(if *FX226,2 set)
NUL &A3 means CTRL + f3	(if *FX227,2 set)
NUL &B6 means SHIFT + CTRL + f6	(if *FX228,2 set)

Values other than 0,1 and 2 remain unchanged.

Note that when a NUL is entered at the keyboard it is supplied as NUL NUL.

This extension of the *FX calls enables applications (such as the revised version of VIEW in the machine) to continue using function keys extensively but also handle characters with the top-bit set.

Operation of the keyboard is transparent to the MOS RDCH routine. For example, pressing CTRL + 0 on the keyboard results in a NUL being returned to RDCH. Similarly entering a top-bit set code results in the return of that single byte through RDCH.

The next section describes keyboard buffer input and output changes. Note that the use of NUL in the two operations is two COMPLETELY separate uses

and they should not be confused.

Keyboard buffer input rules (via *FX138 etc.):
Normal codes &01 thru &7F are entered using 1 byte (as normal).

Special keys (e.g. function & cursor keys) are entered using 1 byte (as normal).

Extended printable codes (&80 thru &FF) are entered as 2 bytes i.e. NUL followed by extended (top-bit set) code. This is a new feature.

A NUL must be entered using 2 bytes NUL NUL. This is a new feature.

Note that when CTRL + 0 is entered into the keyboard, the MOS automatically supplies NUL NUL to the keyboard buffer.

Keyboard buffer output using RDCH:
RDCH automatically converts codes leaving the buffer as follows:
Normal codes &01 thru &7F are removed as 1 byte (as normal).

Special keys have their usual special effects e.g. key expansion (as normal).

Extended printable codes are returned in a SINGLE byte. This is a new feature.

A NUL NUL is returned as a single NUL (as normal).

This means that legal calls continue to work as before, only NUL's poked into the keyboard buffer may have strange effects.

When the new *FX22x,2 is in effect:
Special keys may expand to 2 bytes (NUL followed by &80 thru &BF).

NUL is returned as 2 bytes (NUL NUL).

12) Previously the VDU drivers made calls to the user printer vector and the extension vector, without allowing for the possibility that these may page-in the FSRAM. This has been corrected.

13) *ROMS now indicates whether a slot is ROM or RAM.

14) *TAPE and *MOTOR commands are supported, but have no effect. "-CFS-" and "-TAPE-" are not supported.

15) To provide for additional fonts in the future (e.g. the ISO-font), an additional parameter value 8 has been added to *FX25 to specifically select the Master 128 font i.e:
*FX25,8 forces the Master (Series) font.
*FX25,0 continues to reselect the default font.

16) *TIME will attempt to get the time from the Network if the ANFS is in use. The "day" will be filled with three SPACE characters. Application writers please note.

17) A fix to prevent spurious 1770 NMI's has been added.

18) *BUILD/*APPEND now allow 8-bit characters to be entered.

19) *SHOW without a parameter now displays all soft keys.

BASIC.
The version of BASIC fitted is IV with improvements to accuracy and speed of transcendental functions.

1) ADVAL is implemented via the digital joystick port as follows:
ADVAL0 returns:
Hi-byte of 16-bit value - last channel to convert now totally bogus but provided for compatibility.

Lo-byte of 16-bit value - bits (msb to lsb)
PB4 PB3 PB2 PB1 0 0 0 FIRE

ADVAL1 ADVAL3 ADVAL5 return:
x coordinate in range &0000 thru &FFFF

ADVAL2 ADVAL4 ADVAL6 return:
y coordinate in range &0000 thru &FFFF

The x and y coordinates are supported in an Acorn-compatible fashion i.e:
Left - x value increases
Right - x value decreases
Up - y value increases
Down - y value decreases

ADC events are still supported.

2) TIME$ returns the dummy time "Fri,31 Dec 1999.23:59:59", unless ANFS is present and active, in which case an attempt will be made to get the time from the Network.

3) *BASIC uses *FX142 to change language.

General Guidelines for software writers:
There is no Cassette Filing System fitted to the machine, and the RS 232 port is not necessarily fitted. Avoid calls to hardware, as there are differences between this machine and the Master 128.

Applications writers are recommended to rewrite Model B applications where relevant to make use of the Compact/Master 128 enhanced features. e.g:

* Ability to Write/Read and switch two screens flexibly. This can enhance animation effects.
* The position of BASIC "PAGE" can be assumed to be at &E00, with HIMEM at &8000 if Shadow screen in use.
* Greatly improved graphics primitives.
* Extended Colour fill, with flood fill.
* Extended cursor control, window scrolling and window text control.

Applications for this machine should normally be written in ADFS format on 3.5" disks (not in DFS format as this Filing system is not fitted or recommended for use with 3.5" media). Programmers may wish to maintain compatibility with both this machine and the Master 128 to support users with 3.5" drives fitted to a Master 128. Disk copy protection routines should not make ANY assumptions about the hardware - it may change!

This information is subject to change without notice. No responsibility can be taken for any errors or omissions. The user or program writer should verify that any application program is suitable for the intended environment(s).

D.J.Bell