Appendix A – VDU Code Summary

This Appendix describes the functions performed by the whole of the character set when printed using VDU or PRINT CHR$. Note that several ones are labelled expansion. This means that they will only be effective if the associated expansion modules are connected. Dec hex CTRL + bytes function

dec	hex	CTRL+	bytes	function
0	0	@	0	Does nothing
1	1	A	1	Send character to printer (expansion)
2	2	B	0	Enable printer (expansion)
3	3	C	0	Disable printer (expansion)
4	4	D	0	Write text at text cursor
5	5	E	0	Write text at graphics cursor
6	6	F	0	Enable VDU drivers
7	7	G	0	Make a short bleep (BEL)
8	8	H	0	Move cursor back one character
9	9	1	0	Move cursor forward one character
10	A	J	0	Move cursor down one line
11	B	K	0	Move cursor up one line
12	C	L	0	Clear text area
13	D	M	0	Carriage return
14	E	N	0	Pagedmodeon
15	F	0	0	Paged mode off
16	10	P	0	Clear graphics area
17	11	0	1	Define text colour
18	12	R	2	Define graphics colour
19	13	5	5	Define logical colour
20	14	T	0	Restore default logical colours
21	15	U	0	Disable VDUdrivers/delete current line
22	16	V	1	Select screen MODE
23	17	W	9	Re-program display character
24	18	X	8	Define graphics window
25	19	Y	5	PLOT K,X,Y
26	lA	Z	0	Restore default windows
27	lB	[	0	Reserved
28	1C	,	4	Define text window

29	1D	–	4	Define graphics origin
30	lE	.	0	Home text cursor to top left of window
31	1F	/	2	MovetextcursortoX,y.
32–126				Complete set of ASCII characters
127	7F	DEL	0	Backspace and delete
128-223				Normally undefined (define using *FX20)
224–255				User defined characters

Appendix B - PLOT numbers

0	Move relative to last point
1	Draw relative to last point in current foreground colour
2	Draw relative to last point in logical inverse colour
3	Draw relative to last point in current background colour
4	Move absolute
5	Draw absolute in current foreground colour
6	Draw absolute in logical inverse colour
7	Draw absolute in current background colour

Higher PLOT numbers have other effects which are related to the effects given by the values above.

8-15	Last point in line omitted when ‘inverted’ plotting used
16-23	Using a dotted line
24-31	Dotted line, omitting last point
32-63	Reserved for Graphics Extension ROM
64-71	Single point plotting
72-79	Horizontal line filling
80-87	Plot and fill triangle
88-95	Horizontal line blanking (right only)
96-255	Reserved for future expansions

Horizontal line filling

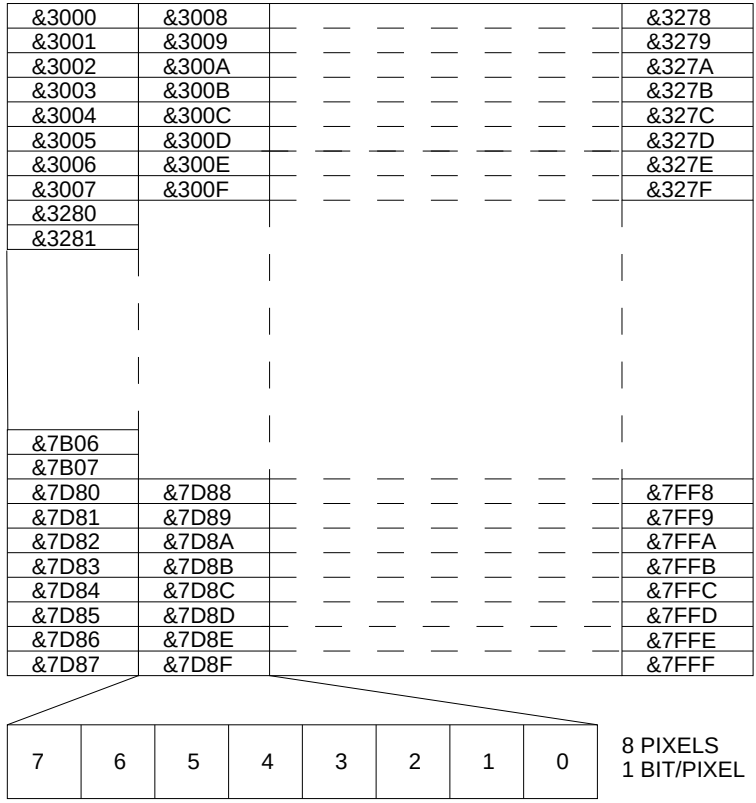
These PLOT numbers start from the specified X,Y co-oridnates. The graphics cursor is then moved left until the first non-background pixel is encountered. The graphics cursor is then moved right until the first non-background coloured pixel is encountered on the right hand side. If the PLOT number is 73 or 77 then a line will be drawn between these two points in the current foreground colour. If the PLOT number is 72 or 76 then no line is drawn but the cursor movements are made (these may be read using OSWORD call with A=&D/13, see chapter 4).

Horizontal line blanking right

These PLOT numbers can be used to *undraw* an object on the screen. They have an the opposite effect to those of the horizontal line filling functions except that the graphics cursor is moved right only. PLOT numbers 91 and 95 will cause a line to be drawn from the specified co-ordinates to the nearest background coloured pixel to the right in the background colour. PLOT numbers 89 and 93 move the graphics cursor but do not cause the line to be blanked.

Appendix C -- Screen mode layouts
MODE 0 Screen layout

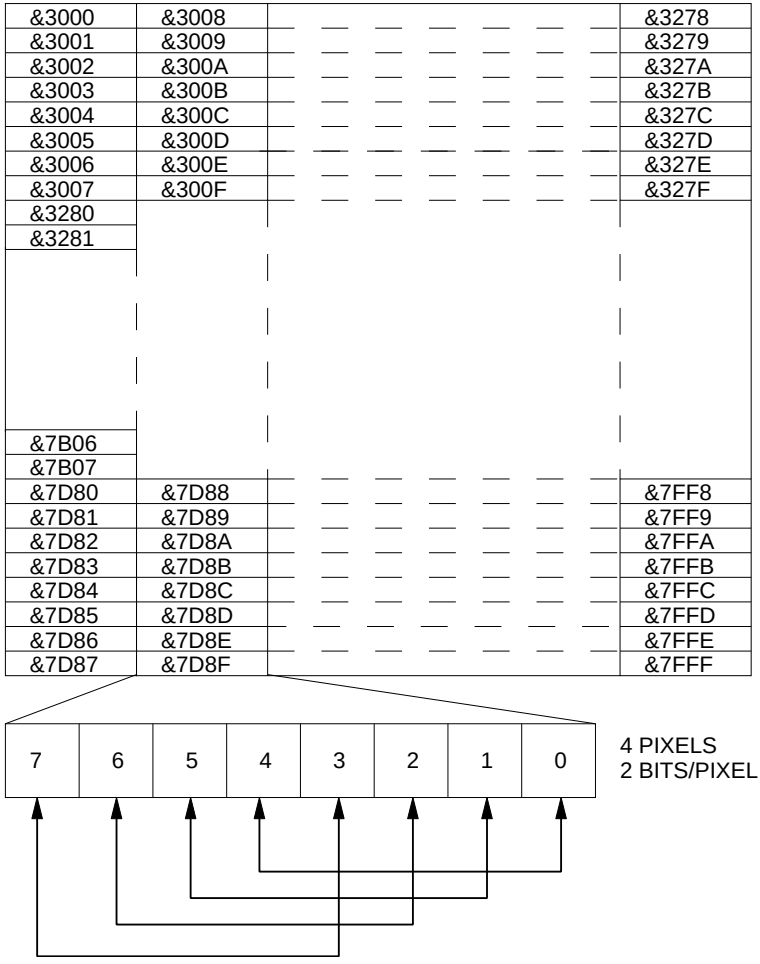
Graphics 640x256
Colours 2
Text 80x32



Note that the screen layout is only as shown after a **CLS** and will change as the screen is scrolled.

MODE 1 Screen layout

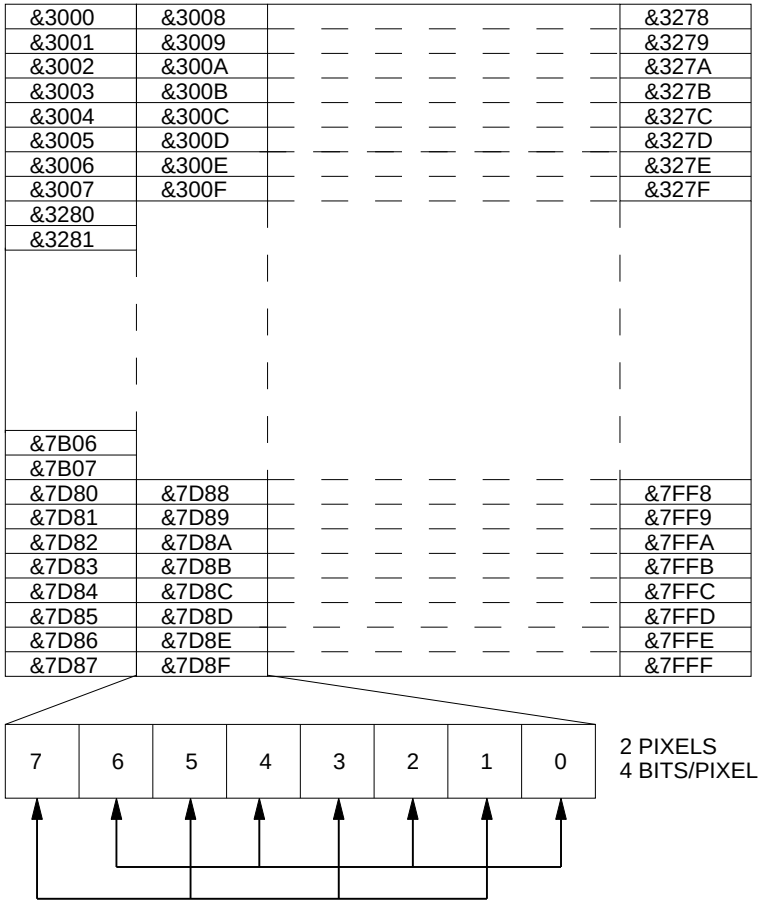
Graphics 320x256
Colours 4
Text 40x32



Note that the screen layout is only as shown after a **CLS** and will change as the screen is scrolled.

MODE 2 Screen layout

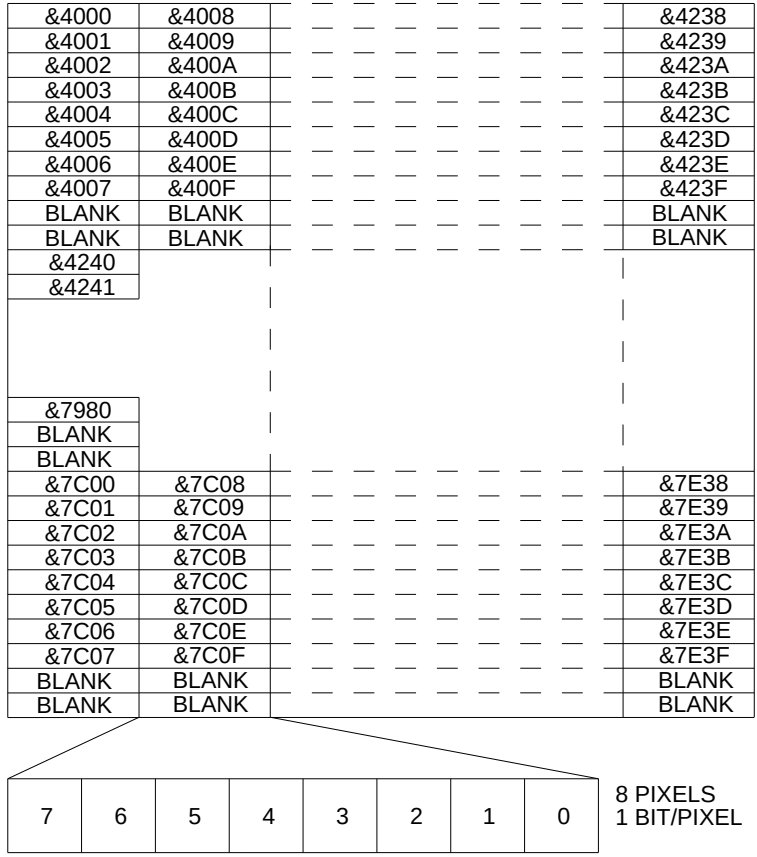
Graphics 160x256
Colours 16
Text 20x32



Note that the screen layout is only as shown after a **CLS** and will change as the screen is scrolled.

MODE 3 Screen layout

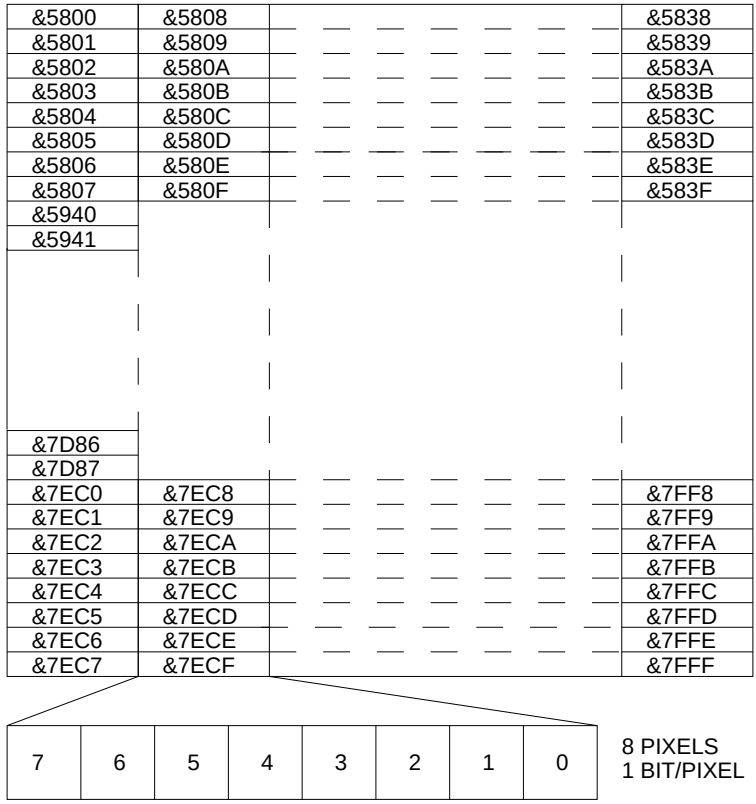
Graphics Not available
Colours 2
Text 80x25



Note that the screen layout is only as shown after a **CLS** and will change as the screen is scrolled.

MODE 4 Screen layout

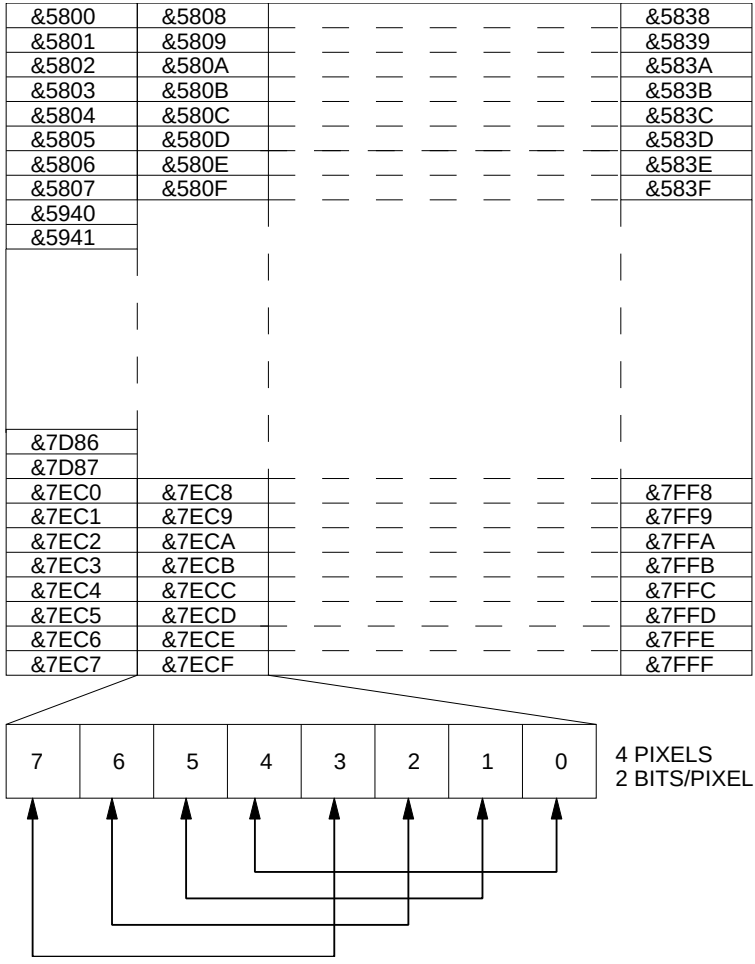
Graphics 320x256
Colours 2
Text 40x32



Note that the screen layout is only as shown after a **CLS** and will change as the screen is scrolled.

MODE 5 Screen layout

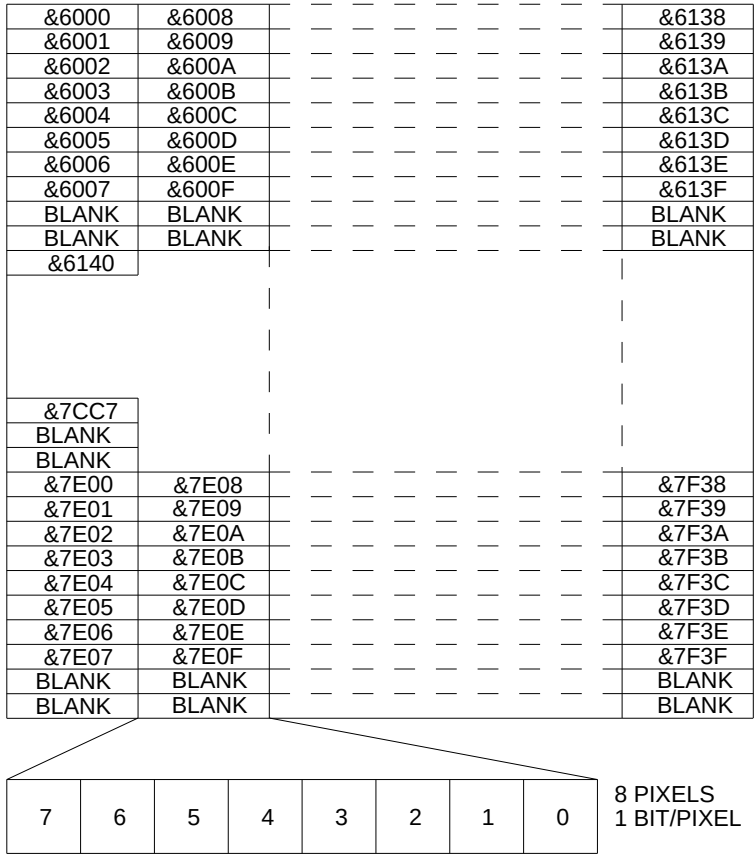
Graphics 160x256
Colours 4
Text 20x32



Note that the screen layout is only as shown after a **CLS** and will change as the screen is scrolled.

MODE 6 Screen layout

Graphics Not available
Colours 2
Text 40x25



Note that the screen layout is only as shown after a **CLS** and will change as the screen is scrolled.

Appendix D - Operating System Calls and Vectors

Routine		Vector		Function
	Addr	Name	Addr	Name
		USERV	200	The user vector
		BRKV	202	The BRK vector
		IRQ1V	204	Primary interrupt vector
		IRQ2V	206	Unrecognised IRQ vector
OSCLI	FFF7	CLIV		Command line interpreter
OSBYTE	FFF4	BYTEV	20A	*FX/OSBYTE call
OSWORD	FFF1	WORDV	20C	OSWORD call
OSWRCH	FFEE	WRCHV	20E	Write character
OSNEWL	FFE7	-	-	Write LF,CR to screen
OSASCI	FFE3	-	-	Write character, &0D=LF,CR
OSRDCH	FFE0	RDCHV	210	Read character
OSFILE	FFDD	FILEV	212	Load/save file
OSARGS	FFDA	ARGSV	214	Load/save file data
OSBGET	FFD7	BGETV	216	Get byte from file
OSBPUT	FFD4	BPUTV	218	Put byte in file
OSGBPB	FFD1	GBPBV	21A	Multiple BPUT/BGET
OSFIND	FFCE	FINDV	21C	Open or close file
		FSCV	21E	File system control entry
		EVNTV	220	Event vector
		UPTV	222	User print routine
		NETV	224	Econet vector
		VDUV	226	Unrecognised VDU commands
		KEYV	228	Keyboard vector
		INSV	22A	Insert into buffer vector
		REMV	22C	Remove from buffer vector
		CNPV	22E	Count/purge buffer vector

		IND1V	230	Spare vector
		IND2V	232	Spare vector
		IND3V	234	Spare vector
NVRDCH	FFCB	-	-	Non-vectored read char.
NVWRCH	FFC8	-	-	Non-vectored write char.
GSREAD	FFC5	-	-	Read char. from string
GSINIT	FFC2	-	-	String input initialize
OSEVEN	FFBF	-	-	Generate an event
OSRDRM	FFB9	-	-	Read byte in paged ROM

Appendix E – Plus 1 ROM slot

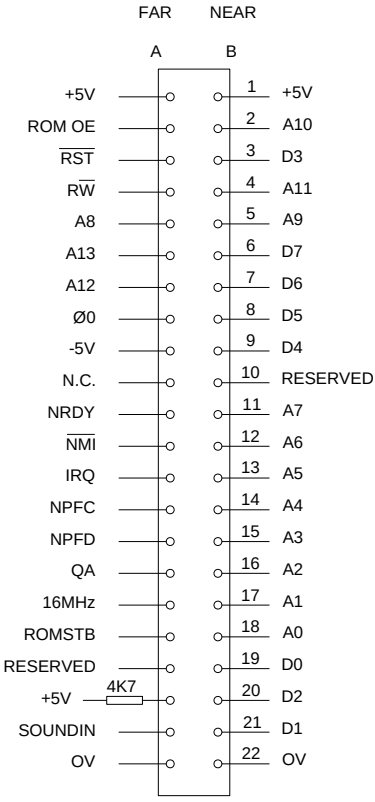


Figure E.1 – The Plus 1 ROM slot connector

Note that most of the standard BBC Micro 1MHz bus signals are available from this slot. However, some of the uses are marginally different to the BBC 1MHz bus. A full specification for producing suitable add-ons is available from Acorn Computers Limited.