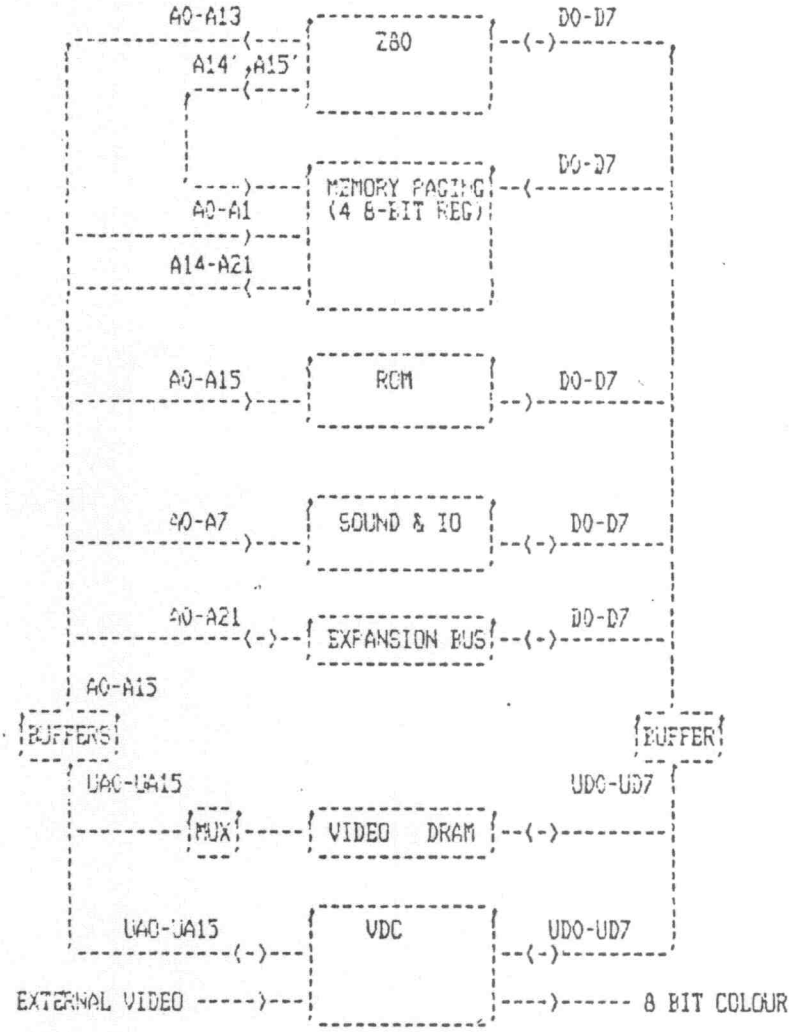
INTRODUCTION

The VDC (ELITE) chip is designed as the video display generator in a paged memory Z80 personal computer. The full addressing range of the computer is 4Megabtyes and the top 64K is allocated to the Video display (although in a minimal system it may also be use for programs and variables).

A typical system use is as follows:



THIS IS OLD INFORMATION BUT MAY BE OF USE.

The VDC generates a complex video display from information placed in the Video DRAM by the processor. The Z80 must load a 16 bit pointer to "line parameters" in the VDC.

Whenever the Z80 needs to access the video RAM or the VDC its clock is stretched until a "slot" is available. Note that the video display and the Z80 normally run asynchronously.

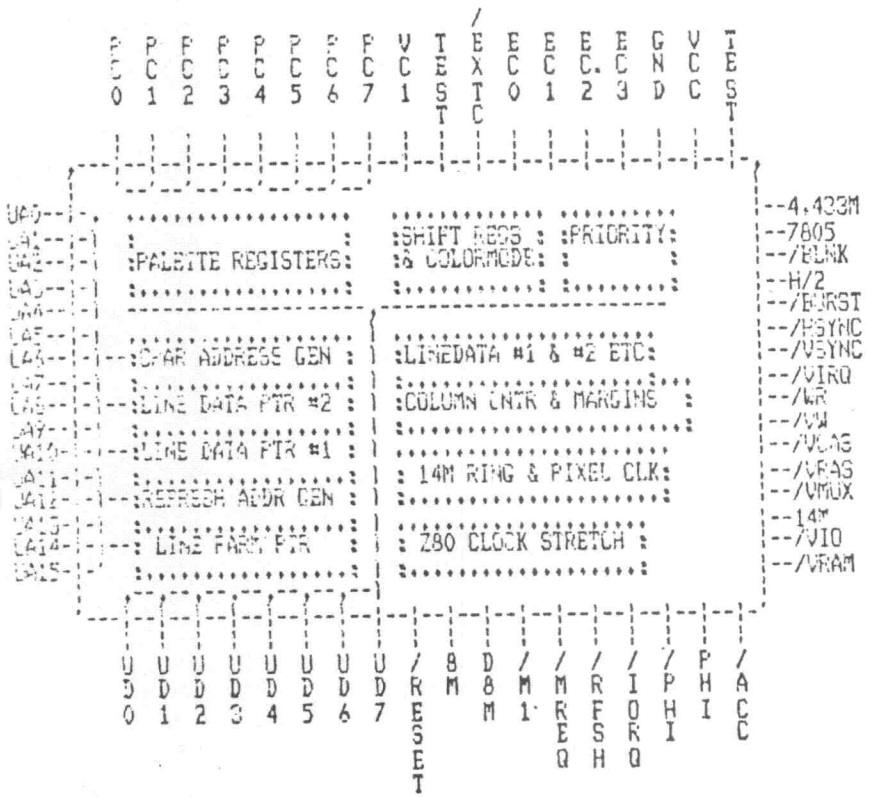
Video dynamic RAM is automatically refreshed.

In contrast to the majority of video display devices which allow the user various modes for the whole display the VDC allows many different modes in the same display frame:

* Mixed mode displays
* User definable characters from fonts of 64, 128, and 256
* 8-bit colour output (256 colours)
* 2,4,16, and 256 colours per line chosen from 256
* Maximum resolution (using interlace) 672 * 512
* Cell based graphics, bitmap and characters
* Characters any height from 1 to 256 scanlines
* Choice of 256 border colours
* User defined screen width and height
* External colour input (unlimited sprites or TV camera)
* Efficient use of RAM (can work with (1K of RAM)
* 4 colour 84 column text mode (uses pixel mode)
* Special uses of bits to increase colour options
* Phase-locked PAL

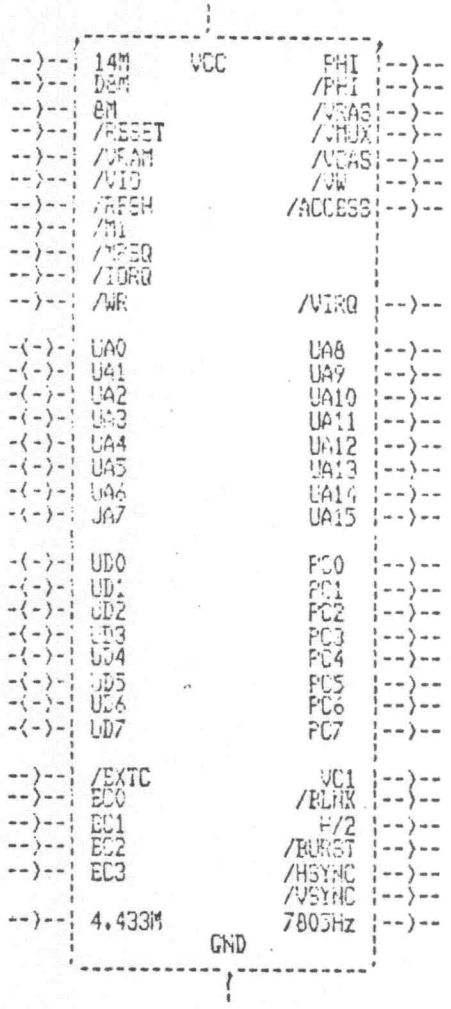
BLOCK DIAGRAM

* 66 of 68 pins are used leaving 2 for TEST
* this diagram is intended as a guide to chip layout
* this pinout (or similar) seems to make internal routing easier
* presumably it is best to put GND and VCC on opposite sides
* refer to DPCH/VDC/4 for more details



In the above diagram the internal address bus (A-BUS), internal data bus (D-BUS) and physical colour bus (PC-BUS) are shown.

PINOUT



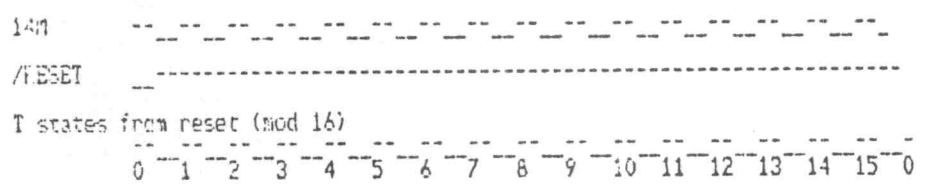
* 2 pins spare for TEST
* /BLNK might be released if more test pins are required

SIGNAL DESCRIPTION

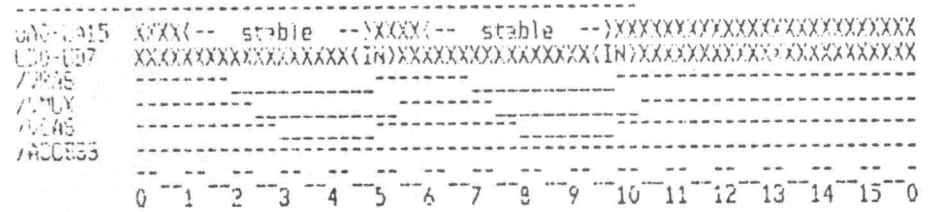
14M	Input. 14.3MHz master video clock.
8M D8M	Input. 8MHz (or 12MHz) master processor clock. Input. Delayed 8MHz (or 12MHz)
/RESET	Input. Reset input from sound chip (phased to /M1).
/VRAM	Input. Active low if Z80 requesting access to video RAM.
/VIO	Input. Active low if Z80 requesting to write to VDC registers (Address 0H80-0H8F gated with /IORQ).
/RFSH	Input. Active low if Z80 refreshing memory.
/M1	Input. Active low if Z80 opcode fetch.
/MREQ	Input. Active low if Z80 accessing memory.
/IORQ	Input. Active low if Z80 accessing IO.
/WR	Input. Active low if Z80 writing to RAM or IO.
/VRAS	LSTTL output. Active low RAS strobe to video memory.
/VMUX	LSTTL output. MUX strobe to video RAM multiplexers.
/VCAS	LSTTL output. Active low CAS strobe to video memory.
/VW	LSTTL output. Active low write strobe of video memory.
/ACCESS	LSTTL output. Active low if Z80 accessing video RAM or video registers. Used to control external buffers.
PHI /PHI	FAST LSTTL output. Z80 clock. FAST LSTTL output. Inverted Z80 clock.
/VIRQ	LSTTL output. Active low if video interrupt requested.
UA0-UA7	LSTTL drive tri-state input/output. Low address bus. (8 lines)
UA8-UA15	LSTTL drive tri-state output. High address bus. (8 lines)
UD0-UD7	LSTTL drive tri-state input/output. Data bus. (8 lines)
/EXTC	Input. Active low to request external colour input.
EC0-EC3	Input. Active high. External colour input. (4 lines)
PC0-PC7	LSTTL output. 8 bit colour output. (8 lines)
/HSYNC	LSTTL output. Active low Horizontal Sync.
/VSYNC	LSTTL output. Active low Vertical Sync.
/BURST	LSTTL output. Active low Colour Burst strobe.
/VC1	LSTTL output. General purpose output. (Used for colour kill.)
/BLNK	LSTTL output. Active low for video blanking.
H/2	LSTTL output. Used by PAL modulator.
VCC GND	Power input at 5V. Ground connection.
4.433?	LSTTL input. Divided by 4 * 71 * 2 to give 7505Hz. [illegible]

EXTERNAL TIMING

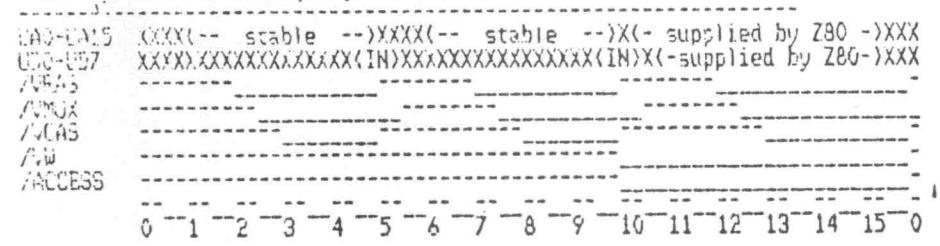
TIMINGS RELATED TO 14M clock



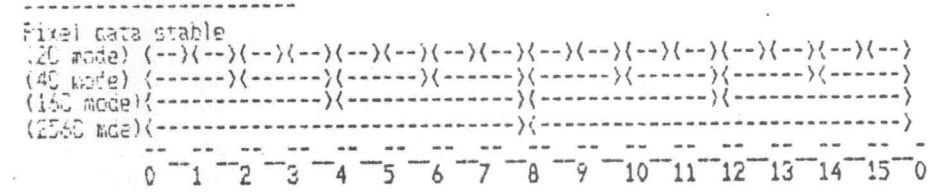
Ideal Dynamic RAM timing signals with no Z80 access.



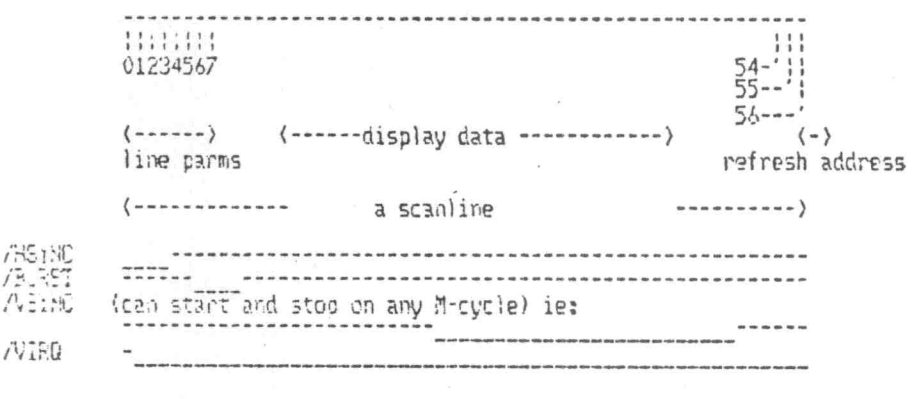
Ideal Dynamic RAM timing signals with Z80 access for video write.



Physical colour output.



16 T-states make up a memory cycle.. an M-cycle. There are 57 M-cycles (or 912 T states) in a scanline.



Allowable departure from ideal spec (in nS):

Name	Transition		Min delay		Max delay		Remarks
/RAS	+-	-+	10	10	30	30	
/CAS	+-	-+	10	10	30	30	
/WE	+-	-+	10	10	30	30	
/OE	+-	-+	10	10	30	30	
/ACCESS	+-	-+	10	10	30	30	
/RES	-+		-70		70		Apply feedback to FF
PC0-PC7	-+	+-	10	10	40	40	Want consistency here
/INVID	+-	-+	-70	-70	70	70	Apply feedback to FF
EC0-EC3	-+	+-	-70	-70	70	70	Apply feedback to FF
/EXTC	-+	+-	-70	-70	70	70	Apply feedback to FF
/HSYNC	-+	+-	0	0	400	400	
/BURST	-+	+-	0	0	400	400	
/VSYNC	-+	+-	0	0	400	400	
/VIRQ	+-	-+	0	0	400	400	

Other timings are highly conditional on previous inputs and do not lend themselves to expression in the above format. Generally they involve asynchronous signals which are associated with flipflops with feedback. It does not matter which clock transition the changes are clocked in on.... only that no indeterminate states are generated ie: use feedback.

DEVICE OPERATION

There are two master clock inputs. The 14M signal determines the timing of the video display and the 8M signal the clock of the Z80. The two clocks run asynchronously.

The 14M signal is not exactly 14MHz. It is the pixel dot rate in the highest resolution display modes and allows for a horizontal resolution of 672 pixels. The line period is obtained by dividing the master video clock by 912. (A line period consists of 57 memory cycles of 16 master clock periods.)

The 8M signal may be 12MHz in some machines. It is twice the clock rate of the Z80 and is used to toggle a flip flop which generates the Z80 phi signal. ie: 6MHz for a 4MHz Z80A and 12MHz for a 6MHz Z80B. Clock stretch is achieved by inhibiting this toggle action.

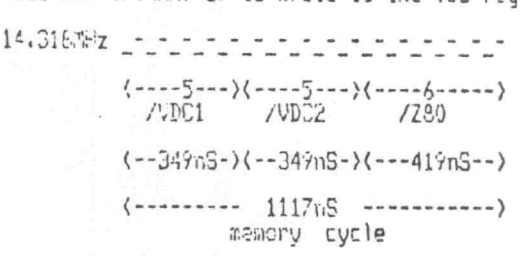
For an NTSC (American) display the master clock is chosen to be four times the colour subcarrier ie: 14.318MHz (3.58MHz *4) which gives a line period of 63.7uS.

A PAL (UK and German) display has a line period of 64uS which would suggest a master clock of 14.25MHz. In fact unless this was phaselocked to the 4.433MHz colour subcarrier a modulated display would show shifting "combing" which greatly degrades display appearance.
There are two possibilities:
1/The 4.433MHz is divided by 568 to form a 7805Hz square wave which is used with H/2 to phaselock the master video clock to 14.2375MHz. This gives a line period of 64.0uS.
2/A 0.8MHz output is provided which can be used to kick a 4.433MHz PAL crystal at 1/5th its frequency. In this case the 14M clock is 14.198MHz and the line period is 64.23uS which would give 280nS chrominance resolution error in a PAL-D receiver.

The SECAM (French) display will probably use a 14.25MHz master clock though it may prove rather cheaper to use 14.101MHz.

In the following description timings relate to the NTSC machine with a Z80A processor running at 4MHz.

The VDC controls Z80 accesses of the SLOW or VIDEO address and data buses. Access of video memory occurs in cycles of 16 master video clock periods. There are 57 such cycles in a video scanline and each cycle last for 16 master video clock periods. In a memory cycle there are three distinct memory accesses. The first two 'slots' are used by the VDC to read data in the video RAM while the last slot may be requested by the Z80 to read from or write to the video RAM (/VRAM) or to write to the VDC registers (/VIO).



In one memory cycle the VDC reads 2 bytes of data from the video RAM and simultaneously clocks out 16, 8, 4, or 2 pixels of previously loaded data to the video display (the number of pixels depends on the display mode).

The Z80 running speed is slowed when it tries to access the video RAM or VDC as it has to wait until a slot is available. In practice the slowing should be about 60% when accessing the video RAM.

It can be assumed that the VIDEO address bus is stable some 50nS after the start of a slot and that the data is read at the end of the slot. Precise hold times for data read are undefined as yet but should be of the order of 20nS.

The hardware generates a /HSYNC pulse of 4.47 uS duration every 63.7 uS. A colour burst strobe /BURST of 4.47uS (for PAL) is generated 2.23uS after the rising edge of /HSYNC output. /BURST is inhibited in the VSYNC video mode. Note that the /VSYNC output is controlled by the data stored in RAM (see below).

The 16 VDC slots after the beginning of the /HSYNC pulse are used by the VDC to load up 16 registers in the VDC from the LINE PARAMETER table. The last 6 VDC slots in a scanline are used by the VDC to refresh the video RAM. The other VDC slots may be used by the VDC to access display data in RAM.

CONTROL REGISTERS

The following registers control the operation of the VIDEO DISPLAY CONTROLLER. They are addressed as IO and the Z80 clock is stretched to synchronise with the Z80 access slots on the video address bus. Addresses 080H to 08FH are reserved for VDC registers in this and future enhanced versions. The following registers are write-only.

(Addresses 090H to 0FFH are reserved for the memory paging, tape, RS423, keyboard, sound etc: (see below))

080H /FIXBIAS

d7 VC1 output used to kill external colour

(d6,d5) (PRIOR1,PRIOR0) external colour priority

=00 EC0-EC3 select corresponding palette colour whenever the display is active and /EXTC is low.

=01 The external colour on EC0-EC3 selects the corresponding palette colour if /EXTC is low and the internal display is generating a logical colour in the range COL8-COL15.

=10 The external colour on EC0-EC3 selects the corresponding palette colour if /EXTC is low and the internal display is generating a logical colour in the range COL8-COL15 OR the external colour is in the range COL0-COL7 (EC3 low).

=11 The external colour on EC0-EC3 selects the corresponding palette colour if /EXTC is low and the internal display is generating a logical colour in the range COL8-COL15 OR the external colour is in the ranges COL0-COL3 or COL8-COL11 (EC2 low).

(d4,....d0) colour bias for logical colours 0-15

081H /BORDER

(d7,....d0) 8 bit border colour

082H /LPL

(d7,....d0) (a11,...a4) of pointer to line parameter table in video RAM. The index into an entry of 16bytes, (a3,..a0), is generated byt the hardware.

083H /LPH

d7 /(load line parameter base) normally 1

d6 /(clock in line parameter base)

(d3,...d0) (a15,..a12) of pointer to the line parameter table in video RAM.

The video display is controlled by values loaded into the video RAM segment (up to 64K at the top of the 4M space) by the Z80. Once this 'line parameter table' has been loaded in and the line parameter base register has been loaded the display requires no more action on the part of the Z80.

The visible display is split up into 'video mode lines'. These 'modelines' are made up of 1 to 256 scanlines. (A scanline is one scan of the electron beam across the CRT and takes about 64 microseconds.)

The following 16 registers are loaded from the line parameter table before each modeline:

SC	scanlines in this modeline (two's complement)
MB	the MODEBYTE (defines video display mode)
LM	left display margin etc:
RM	right margin etc:
LD1L	(a7,...a0) of line data pointer LD1
LD1H	(a8,..a15) of line data pointer LD1
LD2L	(a7,...a0) of line data pointer LD2
LD2H	(a8,..a15) of line data pointer LD2
COL0	8 bit value of logical colour #0
COL1	" #1
COL2	" #2
COL3	" #3
COL4	" #4
COL5	" #5
COL6	" #6
COL7	" #7

BUS ACTIVITY FOR THE VARIOUS MODES

The possible video modes are:

VSYNC — no border colour and use margin information to control positioning of the vertical sync pulse. This gives considerable interlace flexibility.

PIXEL — use information pointed to by LD1 as a bit mapped display.

ATTR — use information pointed to by LD2 as a 2-C bitmap display and information pointed to by LD1 as cell- based graphics attributes (ie: to define paper and ink colours in the cell)

CH256 — use information pointed to by LD1 as indices of characters in a font of 256 characters pointed to by LD2. These characters can be any number of lines deep (up to 256). NB: offsets in the font pointer define which line of the character to start on.

CH128 — As above but assumes a font of 128 characters.

CH64 — As above but assumes a font of 64 characters.

LPIXEL — As for pixel mode but with half the horizontal resolution.

Note that all these modes may be mixed on the same screen and that one has the choice of 2-C, 4-C, 16-C and 256-C colour modes for the PIXEL, LPIXEL, CH256, CH128 and CH64 modes. Also note the special interpretation of certain bits of display data described below.

Details of bus use during a memory cycle in the various modes:

PIXEL	/VDC1	/VDC2
Address	LD1(15,....,0)	LD1(15,....,0)
Data into	BUF1(7,...,0)	BUF2(7,....,0)

BUF1 and BUF2 are loaded sequentially into the shift register and clocked out MSB first ie: both are display bytes. The line data pointer LD1 is incremented twice in each memory cycle. Screen data is fetched from memory and the LD1 counter is incremented only in the active part of the display ie: between the left and right margins of a scanline. The scanline count loaded at the beginning of the PIXEL modeline determines how many scanlines this mode last for.

ATTR	/VDC1	/VDC2
Address	LD1(15,....,0)	LD2(15,....,0)
Data into	BUF1(7,....,0)	BUF2(7,....,0)

Cell based (Spectrum type) graphics. LD1 is used as a pointer to the colour array (paper and ink colours) and LD2 points at 2-C pixel data ie: the display bytes. LD2 is incremented once every memory cycle while the display is active and keeps incrementing up for all scanlines in the modeline. LD1 restarts from the same address for each scanline so attribute data in BUF1 applies to cells which are 8-bits wide and have a depth of the number of scanlines in the modeline.

The character modes involve indirection through the character font. A different font may be defined for each modeline and line by line vertical scrolling is obtained by offsetting the original index.

```
CH256              /VDC1          /VDC2
Address       LD1(15,....,0)  LD2(7,....,0),BUF1(7,....,0)
Data into     BUF1(7,....,0)  BUF2(7,....,0)
```

LD1 is reloaded at the start of each scanline and acts as a pointer into a section of RAM containing the indices of the characters to be displayed. It is incremented once in each memory cycle. LD2 is a pointer into the character font to be used and is incremented at the start of each scanline (it points to a row of a character in the font). Thus the font consists of 256 bytes defining the first row of each character and then another 256 bytes for the next row of each character etc: If the characters are 9 lines deep this requires 2304 bytes of character font (256*9). The data in BUF2 is loaded into the shift register.

```
CH128              /VDC1          /VDC2
Address       LD1(15,....,0)  LD2(8,....,0),BUF1(6,...,0)
Data into     BUF1(7,...,0)   BUF2(7,....,0)
```

This is basically the same as the 256 character font mode but note that the font for 128 9 line deep characters only requires 1152 bytes of memory.

```
CH64               /VDC1          /VDC2
Address       LD1(15,....,0)  LD2(9,....,0),BUF1(5,...,0)
Data into     BUF1(7,....,0)  BUF2(7,....,0)
```

This is basically the same as the 256 character font mode but note that the font for 64 9 line deep characters only requires 576 bytes of memory.

```
PIXEL              /VDC1          /VDC2
Address       LD1(15,....,0)  LD1(15,....,0)
Data into     BUF1(7,....,0)  BUF2(7,....,0)
```

This is much the same as the PIXEL mode except that the LD1 pointer is only incremented once in each memory cycle and the BUF2 data is not used. This gives half the horizontal resolution of the PIXEL mode.

VSYNC

No use is made of the information loaded from memory. It is equivalent to the LPIXEL mode.

THE LINE PARAMETER REGISTERS

SC This is a 2's complemented count of the number of scanlines in the modeline, ie: OFFH for one scanline in modeline (ie: one line of graphics).

M5 d7 If set this takes the VIRQ interrupt line low

(d6,c5) Defines the colour mode:
00 2-C Two colour mode. If a bit in the byte of display data is 1 a pixel of logical colour #1 is output and if 0 a pixel of logical colour #0. The bits are output to the screen in the following order:

d7	d6	d5	d4	d3	d2	d1	d0

01 4-C Four colour mode. Pairs of bits in the byte of display data define the colour of the pixel displayed. 00 for logical colour #0, 01 for logical colour #1, 10 for logical colour #2, and 11 for logical colour #3. The pixels are displayed in the following order:

(d7,d3)	(d6,d2)	(d5,d1)	(d4,d0)

10 16-C Sixteen colour mode. Groups of 4 bits in the byte of display data define the colour of the pixel displayed. 0000 for logical colour #0 up to 1111 for logical colour #15. The pixels are displayed in the following order:

(d7,d5,d3,d1)	(d6,d4,d2,d0)

Note that logical colours #0 to #7 have 8-bit values loaded from the line parameter table at the start of each scanline but that logical colours #8 to #15 have 8-bit values defined as follows:

logical colour #8 = (f4,f3,f2,f1,f0,0,0,0)
#9 = (f4,f3,f2,f1,f0,0,0,1)
....
logical colour #15= (f4,f3,f2,f1,f0,1,1,1)

where (f4,f3,f2,f1,f0) are the low 5 bits of the FIXBIAS register.

11 256-C Two hundred and fifty six colour mode. In this mode the byte of display data defines the colour of a single display pixel.

The actual color produced is as follows:
RED = [b0]*(4/7) + [b3]*(2/7) + [b6]*(1/7)
GREEN = [b1]*(4/7) + [b4]*(2/7) + [b7]*(1/7)
BLUE = [b2]*(2/3) + [b5]*(1/3)

d4 =0 for /VRES. In VRES mode the LD1 and LD2 data pointers are reloaded at the start of each scanline and so the same display pattern is repeated for each scanline of the modeline.

(d3,d2,d1) defines the video display mode (see above):
000 VSYNC mode
001 PIXEL mode
010 ATTR mode
011 CH256 mode
100 CH128 mode
101 CH64 mode
110 unused at present
111 LPIXEL mode

d0 If 1 this forces a reload of the line parameter base register. This will normally occur at the end of each video frame.

LM d7 =1 for MSBALT ie: if the top bit of the display byte is 1 this causes logical colours #2 and #3 to be selected instead of #0 and #1 in the 2-C display mode. If the top bit is 0 logical colours #0 and #1 are used as usual. In both cases the top bit seen by the shift register is forced to 0. This mode is useful in simulating an 80 column VDU in the PIXEL mode. Since the msb or LmS of any character is 0 for character spacing it can be used to highlight areas of text.

d6 =1 for LSBALT ie: if the bottom bit of the display byte is 1 this causes logical colours #4 and #5 to be selected instead of #0 and #1 in the 2-C display mode. If the top bit is 0 logical colours #0 and #1 are used as usual. In both cases the top bit seen by the shift register is forced to 0. This mode is useful in simulating an 80 column VDU in the PIXEL mode. Since the lsb or LmS of any character is 0 for character spacing it can be used to highlight areas of text.

(d5,..d0) define the left hand margin of the active display. In practice this value will not be below 10 for the left hand edge of the CRT. The display changes from being border colour at the left hand margin and the display data counters start being incremented. The left hand margin defines the start of the vertical sync pulse in the VSYNC video mode.

RM d7 ALTIND1 If a 2-C character mode is selected this will cause characters with an index above 080H to have a paper of logical colour #2 and an ink of logical colour #3 instead of #0 and #1.

d6 ALTIND0 If a 2-C character mode is selected this will cause characters which have their next to most significant bit set to swap logical colours as follows:

#0 -) #4
#1 -) #5
#2 -) #6
#3 -) #7

(d5,..d0) These bits define the right hand side of the active display. The maximum value is normally 54 for the right hand edge of the CRT. The display returns to the border colour at the right hand margin and the line data pointers are not incremented until the next left hand margin. In the VSYNC video mode the right hand margin defines the end of the vertical sync pulse.

LD1L This 8-bit value defines the starting value of (a7,...a0) of the line data pointer LD1.

LD1H This 8-bit value defines the starting value of (a8,...a15) of the line data pointer LD1.
LD1 is used as a pointer to the next byte of display data in the PIXEL and LPIXEL modes. In the CH256, CH128 and CH64 modes it is the index of a character in the character font. In the ATTR mode it points to attribute information.

LD2L This 8-bit value defines the starting value of (a7,...a0) of the line data pointer LD2.

LD2H This 8-bit value defines the starting value of (a8,...a15) of the line data pointer LD2.
LD2 is used as a pointer for pixel information in the ATTR display mode and as a pointer to the character font in the CH256, CH128 and CH64 modes. It is not at present used in the PIXEL and LPIXEL modes but it is hoped that is will define vertical pixel resolution at a later date.

COL0 Logical colour #0. This is the paper colour in C-2 modes though note the exceptions above.

COL1 Logical colour #1. This is the ink colour in C-2 modes though note the exceptions above.

COL2 Logical colour #2 (Alternate paper)
COL3 Logical colour #3 (Alternate ink)
COL4 Logical colour #4
COL5 Logical colour #5
COL6 Logical colour #6
COL7 Logical colour #7