Acorn
The choice of experience.

VIDC Datasheet

Acorn Advanced R&D

ISBN 1 85250 027 1

Published by:
Acorn Computers Limited, Fulbourn Road, Cherry Hinton, Cambridge CB1 4JN, UK

Contents

1. Introduction

The Video Controller (VIDC) accepts video data from memory under DMA control, serialises and passes it through a colour look-up palette, and converts it to analog signals for driving the CRT guns. The chip also controls all the display timing parameters and controls the position and pattern of the cursor sprite. In addition, the VIDC incorporates an exponential DAC and stereo image table for the generation of high quality sound from data in the memory.

The VIDC requests data from the memory when required, and buffers it in one of three FIFOs before using it. Note that the addressing of the data in memory is controlled elsewhere in the system (usually in the Memory Controller, MEMC). Data is requested in blocks of four 32-bit words, allowing efficient use of paged-mode DRAM without locking up the system data bus for long periods.

The VIDC is a highly programmable device, offering a very wide choice of display formats. The pixel rate can be selected in a range between 8 and 24MHz and the data can be serialised to either 8, 4, 2, or 1 bit per pixel. The horizontal timing parameters can be controlled to units of 2 pixels, and the vertical timing parameters can be controlled to units of a raster. The colour look-up palette which drives the three on-chip DACs is 13 bits wide, offering a choice from 4096 colours or an external video source.

Extensive use is made of pipelining throughout the device.

The cursor sprite is 32 pixels wide, and any number of rasters high. It can be positioned anywhere on the screen. Three simultaneous colours (again from a choice of 4096) are supported, and any pixel can be defined as transparent, making possible cursors of many shapes.

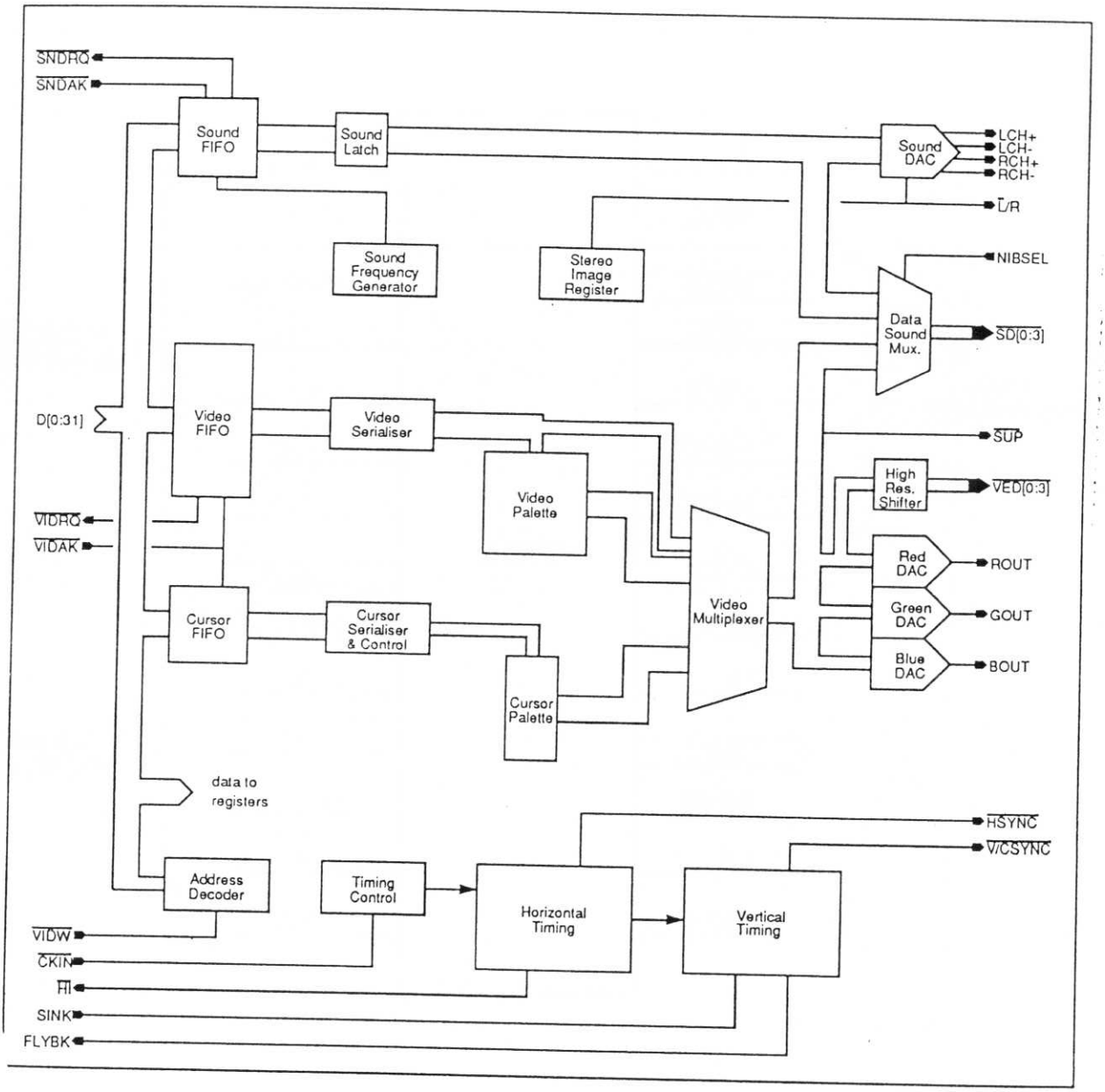
The sound system implemented on the device can support up to 8 channels, each with a separate stereo position.

It should be noted that there are two variants of the VIDC, designated VIDC1 and VIDC2. The two devices are identical apart from two aspects: the sense of the video DACs; and the order of the bits in the sound DAC. See sections 6.5 and 6.10.

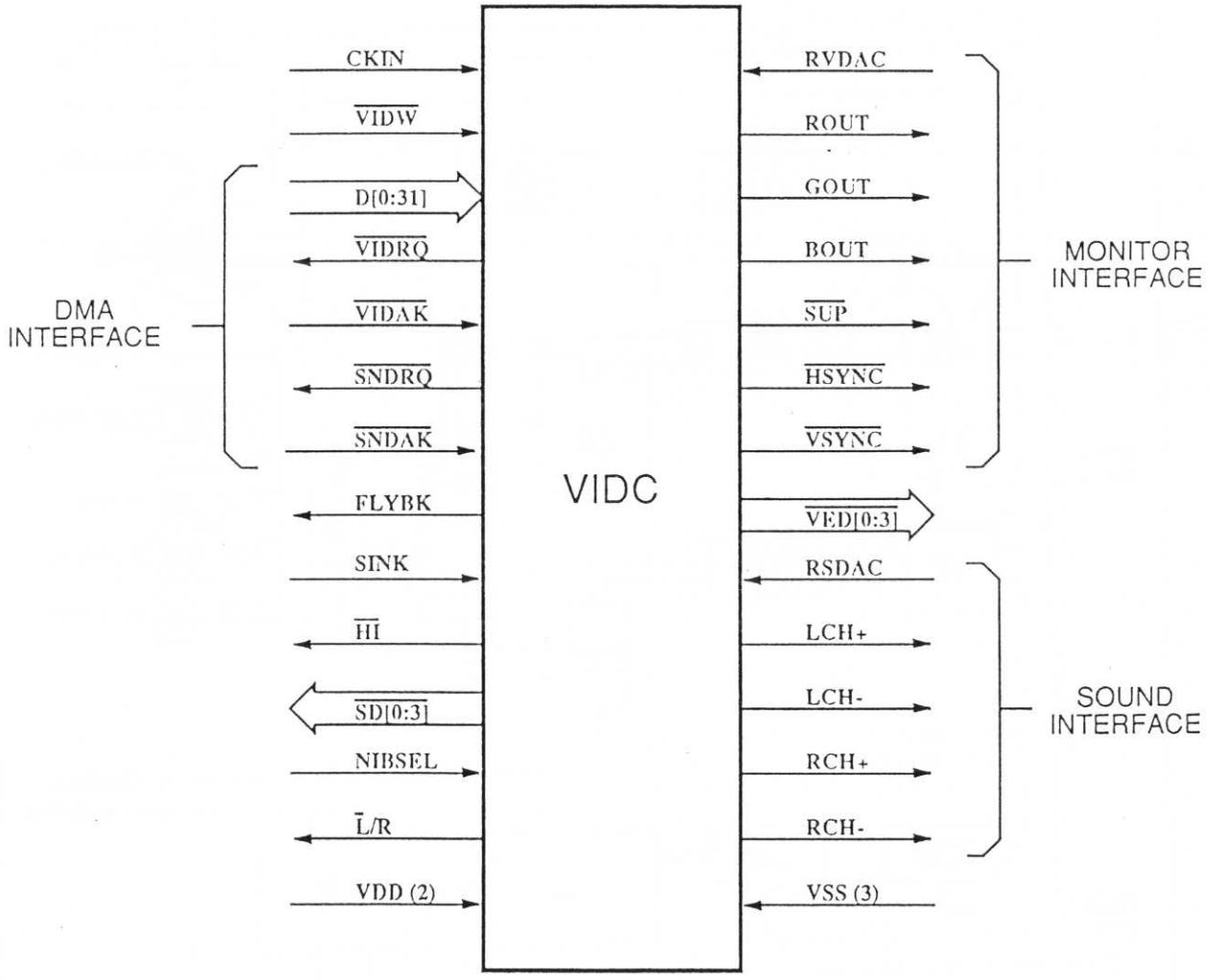
FEATURES

* pixel rate selectable as 8, 12, 16 or 24MHz
* serialises data to 1, 2, 4, or 8 bits per pixel
* 16 word by 4096 colour look-up palette
* 4-bit DACs for each CRT gun
* highly programmable screen parameters
* border facility
* cursor sprite
* optional interlaced sync. display format
* allowance for external synchronisation
* very high resolution monochrome mode
* high quality stereo sound generation
* fabricated in CMOS for low power

2. Block Diagram



3. Functional Diagram



4. Description of Signals

Name	Pin	Type	Description
CKIN	19	IT	Master clock input (typically 24MHz).
$\overline{\text{VIDW}}$	27	IT	Register write strobe. A LOW on this line writes data into one of the VIDC registers. The address of the register is supplied on the upper bits, and the data to be written is on the lower bits of the data bus.
D[0:31]	44-68,1-7	IT	Data input bus. This bus carries data for register writes, video DMA, Cursor DMA and Sound DMA, according to which type of data strobe is present.
$\overline{\text{VIDRQ}}$	23	OC	Video data request. This signal is driven LOW when the VIDC requires another block of 16 bytes of video data (when **HSYNC** is HIGH) or cursor data (when **HSYNC** is LOW). It is driven HIGH again by the first valid $\overline{\text{VIDAK}}$.
$\overline{\text{VIDAK}}$	8	IT	Video data acknowledge. A LOW on this signal strobes a data word into the video or cursor FIFO depending on the state of **HSYNC** when the request was made. Note that a LOW on $\overline{\text{VIDRQ}}$ signifies a request for 4 words of data, and so $\overline{\text{VIDAK}}$ must go LOW 4 times to service each request.
$\overline{\text{SNDRQ}}$	24	OC	Sound data request. This signal is driven LOW when the VIDC requires another block of 16 bytes of sound data. It is driven HIGH again by the first valid $\overline{\text{SNDAK}}$.
$\overline{\text{SNDAK}}$	9	IT	Sound data acknowledge. A LOW on this signal strobes a data word into the sound FIFO. Note that a LOW on $\overline{\text{SNDRQ}}$ signifies a request for 4 words of data, and so $\overline{\text{SNDAK}}$ must go LOW 4 times to service each request.
FLYBK	22	OC	Vertical flyback. This signal is driven HIGH when the display is in vertical flyback. Specifically, it is set HIGH at the start of the first raster which is not display data, though may be border, (at the bottom of the screen), and is cleared down at the start of the first raster which is display data (at the top of the screen).
SINK	20	IT	External Synchronisation pulse. A HIGH on this signal resets the vertical timing counter, and if interlaced display format is being used, the odd field is selected. The horizontal timing counter, and all other registers, are unaffected by this signal.
$\overline{\text{HI}}$	21	OC	Horizontal interlace marker. Test pin. When an interlaced display format is selected this signal is driven LOW half way along, and stays LOW until the end, of each raster.
$\overline{\text{SD[0:3]}}$	37-34	OC	Multiplexed sound data output. Test pins. These pins are used for testing the digital data paths through the chip, and should be used in conjunction with test mode 3 and **NIBSEL**. For more information on test mode 3, refer to the *control register*, section 5.23.

NIBSEL	33	IT	Sound data output selector. Test pin. When this signal is LOW, the sound data bus port outputs the inverse of the green DAC data, or the low nibble of the sound data. When **NIBSEL** is HIGH, the sound data bus port outputs the inverse of the blue DAC data, or the high nibble of the sound data.
$\overline{L}$/R	17	OT	Left / Right output. Test pin. This signal is driven LOW when the sound output is steered to the left output port, and is HIGH when the sound output is steered to the right output port. In test mode 3, the pin changes its function, and outputs the sound sampling clock instead.
RVDAC	43	IA	Video DAC reference current. A current must be fed into this pin to set the output current of the video DACs. The full scale output current is 15 times this current. In most applications a resistor from **VDD** to this pin is sufficient to set the current.
ROUT	39	OA	Red analog output. The output to the CRT guns is a current sink. On VIDC1 "black" is defined as 15 times the reference current, and on VIDC2, "black" is defined as zero current. Level shifting and buffering is normally required to drive the lines to the CRT.
GOUT	40	OA	Green analog output. As for ROUT
BOUT	41	OA	Blue analog output. As for ROUT
$\overline{SUP}$	28	OC	Supremacy output signal. This signal is used to control a multiplexer between the output of VIDC and an external source when video mixing is required. If bit 12 of the video or cursor palette for any logical colour is set, $\overline{SUP}$ is driven LOW when that logical colour is displayed. In this way any logical colour can be defined as being supreme or not, on a pixel-by-pixel basis.
$\overline{HSYNC}$	25	OC	Horizontal synchronisation pulse. This signal is required by some monitors. It is also used by the MEMC to discriminate between cursor and video data requests. The pulse is active LOW, and the pulse width is programmable in units of 2 pixels, though there are certain system-related restrictions. See section 6.2.
$\overline{V/CSYNC}$	26	OC	Vertical / composite synchronisation pulse. Depending on bit 7 in the *control register*, this pin can be either the vertical sync. pulse, or a form of composite sync. pulse. The vertical sync. pulse width is programmable in units of a raster and, if selected, is active LOW. The composite sync. pulse is the exclusive-NOR of $\overline{HSYNC}$ and $\overline{VSYNC}$.
$\overline{VED[0:3]}$	32-29	OC	Video external data output. The inverse of the 4 bits of data which are fed to the red DAC are output on these pins. With an external serialiser, this data can be used to produce very high resolution monochrome displays.
RSDAC	12	IA	Sound DAC reference current. A current must be fed into this pin to set the output current of the sound DAC. The full scale output current is approximately 32 times this current. In most applications a resistor from **VDD** to this pin is sufficient to set the current.

LCH+	13	OA	Left channel positive sound output. The sound output is in the form of a current sink which is switched to one of 4 pins (pins 13 - 16). The left channel signal is produced by externally integrating and subtracting the two signals **LCH+** and **LCH-** . Similarly, the right channel signal is produced by externally integrating and subtracting the two signals **RCH+** and **RCH-** .
RCH+	14	OA	Right channel positive sound output. See description of pin 13.
LCH-	15	OA	Left channel negative sound output. See description of pin 13.
RCH-	16	OA	Right channel negative sound output. See description of pin 13.
VSSd	18	PWR	Digital ground. This pin is the ground supply to the digital circuits in the device.
VSSs	10	PWR	Sound ground. This pin is the ground supply to the sound DAC in the device. It must be connected to the pin **VSSd** outside the chip.
VSSv	42	PWR	Video ground. This pin is the ground supply to the video DACs in the device. It must be connected to the pin **VSSd** outside the chip.
VDDd	38	PWR	Digital supply. This pin is the positive supply to the digital circuits in the device.
VDDs	11	PWR	Sound supply. This pin is the positive supply to the sound DAC in the device. It must be at the same potential as **VDDd**, and should be decoupled to **VSSs**. Note that the sound reference current input and the sound analog output currents are all referenced to this signal.

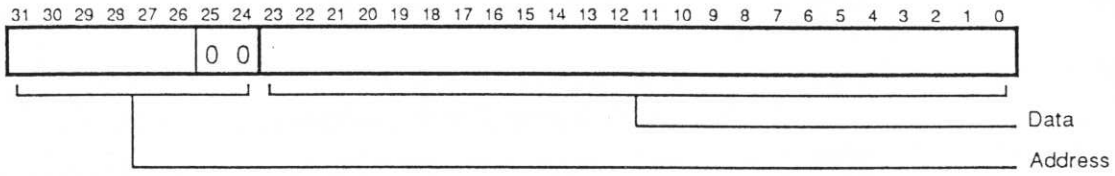
Key to Signal Types:

IT	TTL Compatible Input
OC	CMOS Level Output
IA	Analog Input
OA	Analog Output
PWR	Supply

5. Programming Model

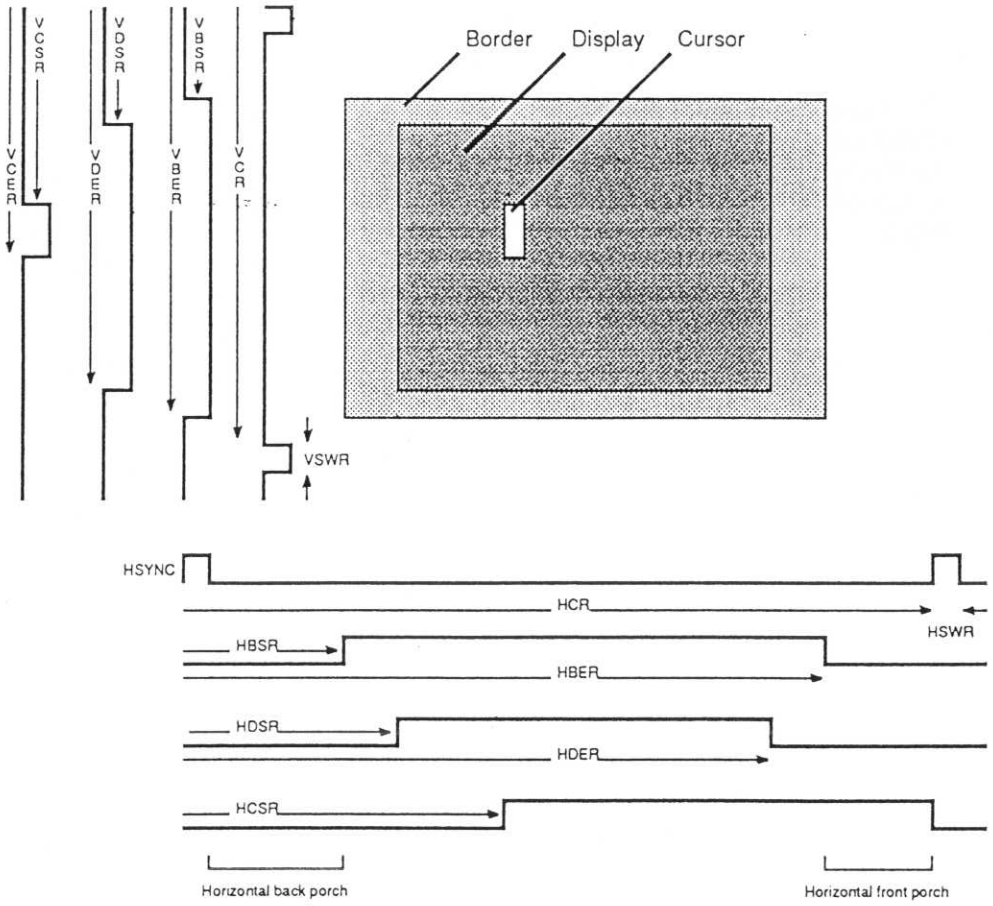
5.1 Register Overview

Apart from the three 32-bit wide FIFOs (Video, Cursor and Sound), the VIDC contains 46 write-only registers of up to 13 bits each. In all cases the address of the register is contained in the top 6 bits (26-31) of the data field. Bits 25 and 24 are not used. The actual data bits are distributed among the remaining 24 bits of the data field according to the register in question.



Treating bit 24 as the least significant address bit, the register map is shown in Table 1. Note that there are 18 *reserved* locations. These locations should never be written to as they may actually contain other registers (some of the registers are dual-mapped within the device).

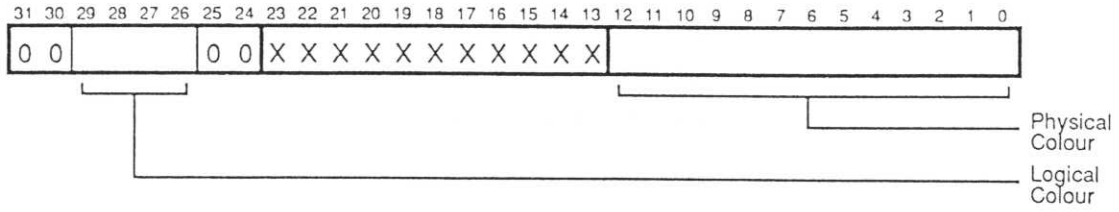
In order to define the display format correctly, eleven registers need to be programmed as shown in the diagram below.



Address (Hex)	Register
00	Video Palette logical colour 0
04	Video Palette logical colour 1
08	Video Palette logical colour 2
0C	Video Palette logical colour 3
10	Video Palette logical colour 4
14	Video Palette logical colour 5
18	Video Palette logical colour 6
1C	Video Palette logical colour 7
20	Video Palette logical colour 8
24	Video Palette logical colour 9
28	Video Palette logical colour A
2C	Video Palette logical colour B
30	Video Palette logical colour C
34	Video Palette logical colour D
38	Video Palette logical colour E
3C	Video Palette logical colour F
40	Border Colour Register
44	Cursor Palette logical colour 1
48	Cursor Palette logical colour 2
4C	Cursor Palette logical colour 3
50-5C	*reserved*
60	Stereo Image Register 7
64	Stereo Image Register 0
68	Stereo Image Register 1
6C	Stereo Image Register 2
70	Stereo Image Register 3
74	Stereo Image Register 4
78	Stereo Image Register 5
7C	Stereo Image Register 6
80	Horizontal Cycle Register
84	Horizontal Sync Width Register
88	Horizontal Border Start Register
8C	Horizontal Display Start Register
90	Horizontal Display End Register
94	Horizontal Border End Register
98	Horizontal Cursor Start Register
9C	Horizontal Interlace Register
A0	Vertical Cycle Register
A4	Vertical Sync Width Register
A8	Vertical Border Start Register
AC	Vertical Display Start Register
B0	Vertical Display End Register
B4	Vertical Border End Register
B8	Vertical Cursor Start Register
BC	Vertical Cursor End Register
C0	Sound Frequency Register
C4-DC	*reserved*
E0	Control Register
E4-FC	*reserved*

Table 1: Register Allocation

5.2 Video Palette Logical colours 0-FH : Addresses 00H-3CH

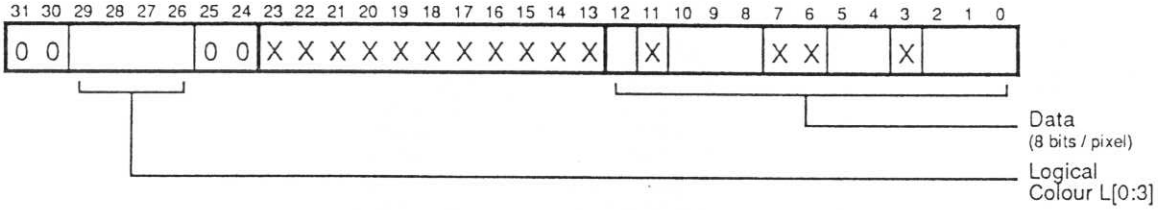


In 1,2 & 4 bits per pixel mode, data bits **D[0:12]** define the physical colour corresponding to that logical colour.

D[0:3] define the Red amplitude. **D[0]** least significant.
D[4:7] define the Green amplitude. **D[4]** least significant.
D[8:11] define the Blue amplitude. **D[8]** least significant.
D[12] defines the supremacy bit for that colour.

SUP.	BLUE				GREEN				RED			
D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0

In 8 bits per pixel mode, only 9 bits of the palette are used.



The palette outputs define the least significant bits of each colour.

SUP.	BLUE			GREEN		RED		
D12	D10	D9	D8	D5	D4	D2	D1	D0

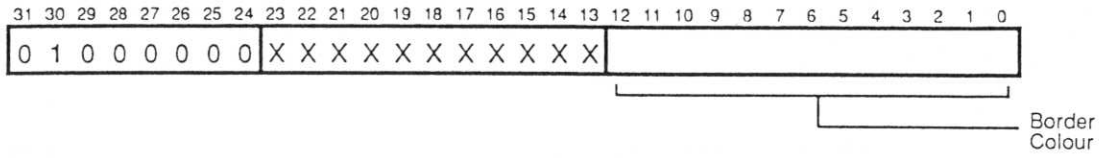
The most significant bits for each colour now come directly from the upper 4 bits of the logical colour field, giving the physical data field as follows:

SUP.	BLUE				GREEN				RED			
D12	L7	D10	D9	D8	L6	L5	D5	D4	L4	D2	D1	D0

D_n: These bits come from the palette field.
L_n: These bits come from the logical field.

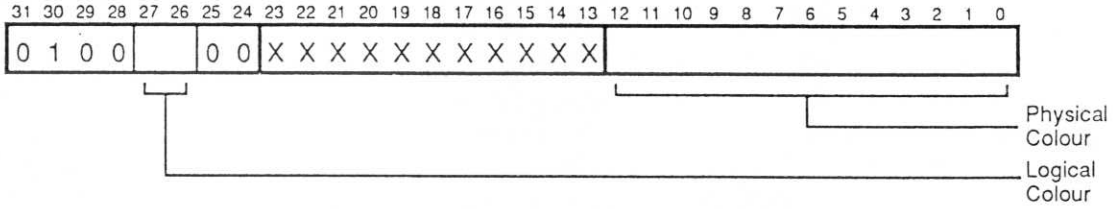
In 4 and 8 bits per pixel mode, all 16 locations should be programmed.
In 2 bits per pixel mode only colours 0, 1, 2 and 3 need to be programmed.
In 1 bit per pixel mode only colours 0 and 1 need to be programmed.

5.3 Border Colour Register : Address 40H



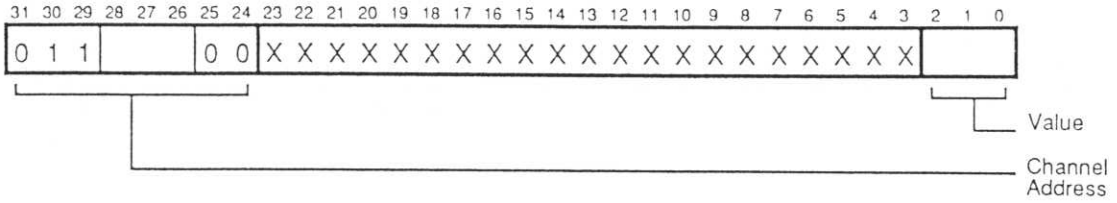
In all modes this register defines the physical border colour.
D[0:3] define the Red amplitude. D[0] least significant.
D[4:7] define the Green amplitude. D[4] least significant.
D[8:11] define the Blue amplitude. D[8] least significant.
D[12] defines the supremacy bit for the border.

5.4 Cursor Palette Logical Colours 1-3 : Addresses 44H-4CH



In all modes these registers define the physical cursor colours corresponding to the logical colours. Note that cursor logical colour 00 is defined as being transparent (ie. no cursor display), and this location is used for the Border Colour Register.
D[0:3] define the Red amplitude. D[0] least significant.
D[4:7] define the Green amplitude. D[4] least significant.
D[8:11] define the Blue amplitude. D[8] least significant.
D[12] defines the supremacy bit for that cursor colour.

5.5 Stereo Image Registers, Channels 0-7 : Addresses 60H-7CH



These 8 registers define the stereo image position for each of the 8 possible channels as shown in Table 2.

Address (Hex)	Register
60	Stereo Image Register 7
64	Stereo Image Register 0
68	Stereo Image Register 1
6C	Stereo Image Register 2
70	Stereo Image Register 3
74	Stereo Image Register 4
78	Stereo Image Register 5
7C	Stereo Image Register 6

Table 2: Stereo Image Register Allocation

When only 4 channels are used, registers 4,5,6,7 should be programmed to the same values as registers 0,1,2,3 respectively.

When only 2 channels are used, registers 0,2,4 & 6 pertain to one channel, and so should be programmed to the same value, and registers 1,3,5 & 7 pertain to the other channel.

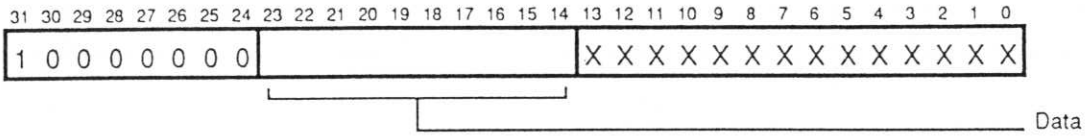
When only one channel is used, all 8 registers should be programmed to the same value.

The 3-bit value is defined in Table 3.

Value	Effect
0	*Undefined*
1	100% left
2	83% left
3	67% left
4	centre
5	67% right
6	83% right
7	100% right

Table 3: Stereo Image Register Values

5.6 Horizontal Cycle Register (HCR) : Address 80H



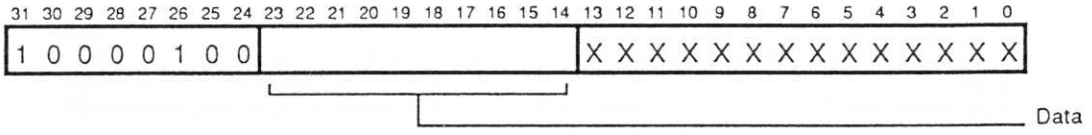
This register defines the period, in units of 2 pixels, of the horizontal scan. ie. display time + horizontal retrace time.

If N pixels are required in the horizontal scan period, then value (N-2)/2 should be programmed into the HCR [N must be even].

If interlaced display is selected, N/2 must also be even.

This is a 10 bit register, with bit 14 the least significant.

5.7 Horizontal Sync Width Register (HSWR) : Address 84H



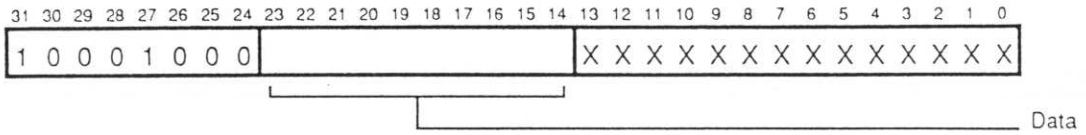
This register defines the width, in units of 2 pixel periods, of the HSYNC pulse.

If N pixels are required in the HSYNC pulse, then value (N-2)/2 should be programmed into the HSWR. [N must be even].

The minimum value programmed may be 0, but system constraints impose a larger minimum value. See section 6.2.

This is a 10 bit register, with bit 14 the least significant.

5.8 Horizontal Border Start Register (HBSR) : Address 88H



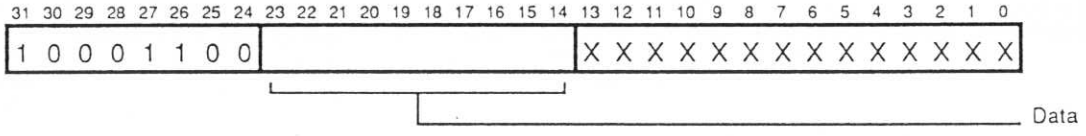
This register defines the time, in units of 2 pixel periods, from the start of the HSYNC pulse to the start of the border display.

If M pixels are required in this time, then value (M-1)/2 should be programmed into the **HBSR**. [M must be odd].

Note that this register must always be programmed, even when a border is not required. If a border is not required, then the value in the HBSR must be such as to start the border in the same place as the display start. ie. $M_{HBSR} = M_{HDSR}$.

This is a 10 bit register, with bit 14 the least significant.

5.9 Horizontal Display Start Register (HDSR) : Address 8CH

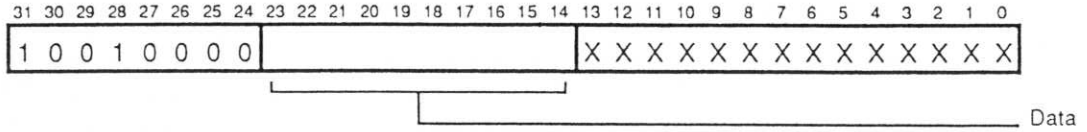


This register defines the time, in units of 2 pixel periods, from the start of the HSYNC pulse to the start of the video display.

The value programmed here depends on the screen mode in use. If M pixels are required in this time, then:
in 8 bits per pixel mode, value (M-5)/2 should be programmed into the **HDSR**. [M must be odd].
in 4 bits per pixel mode, value (M-7)/2 should be programmed into the **HDSR**. [M must be odd].
in 2 bits per pixel mode, value (M-11)/2 should be programmed into the **HDSR**. [M must be odd].
in 1 bit per pixel mode, value (M-19)/2 should be programmed into the **HDSR**. [M must be odd].

This is a 10 bit register, with bit 14 the least significant.

5.10 Horizontal Display End Register (HDER) : Address 90H

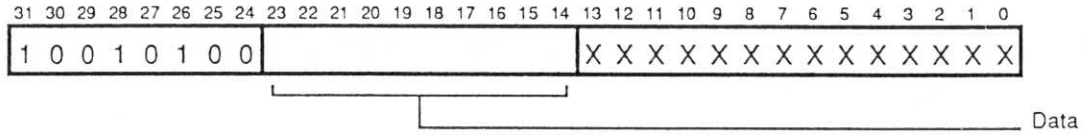


This register defines the time, in units of 2 pixel periods, from the start of the HSYNC pulse to the end of the video display. (ie. the first pixel which is *not* display).

The value programmed here depends on the screen mode in use. If M pixels are required in this time, then:
in 8 bits per pixel mode, value (M-5)/2 should be programmed into the **HDSR**. [M must be odd].
in 4 bits per pixel mode, value (M-7)/2 should be programmed into the **HDSR**. [M must be odd].
in 2 bits per pixel mode, value (M-11)/2 should be programmed into the **HDSR**. [M must be odd].
in 1 bit per pixel mode, value (M-19)/2 should be programmed into the **HDSR**. [M must be odd].

This is a 10 bit register, with bit 14 the least significant.

5.11 Horizontal Border End Register (HBER) : Address 94H



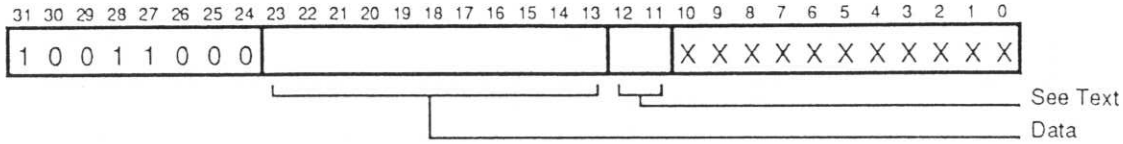
This register defines the time, in units of 2 pixel periods, from the start of the HSYNC pulse to the end of the border display. (ie. the first pixel which is *not* border).

If M pixels are required in this time, then value (M-1)/2 should be programmed into the **HBER**. [M must be odd].

Again, if no border is required, this register must still be programmed such that $M_{HBER} = M_{HDER}$.

This is a 10 bit register, with bit 14 the least significant.

5.12 Horizontal Cursor Start Register (HCSR) : Address 98H



This register defines the time, in units of single pixel periods, from the start of the HSYNC pulse to the start of the cursor display.

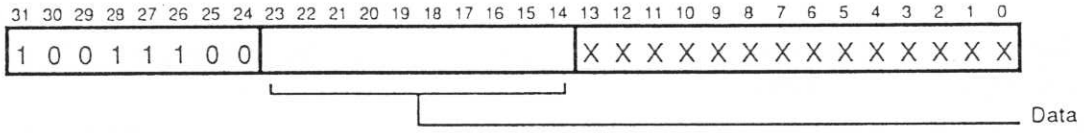
If M pixels are required in this time, then value (M-6) should be programmed into the **HCSR**.

This is normally an 11 bit register, with bit 13 the least significant. Bits 11 and 12 must be zero except in the High Resolution mode. See section 6.6.

In this mode, where each 24MHz pixel is further divided into 4 pixels, the cursor sub-position can be defined by programming bits 11 & 12 of the **HCSR**, which will move the cursor position within the 24MHz pixel. Refer to section 6.6.

Note that only the cursor start position needs to be defined, as the cursor is automatically disabled after 32 pixels. If a cursor smaller than this is required, then the remaining bits in the cursor pattern should be programmed to logical colour 00 (transparent).

5.13 Horizontal Interlace Register (HIR) : Address 9CH

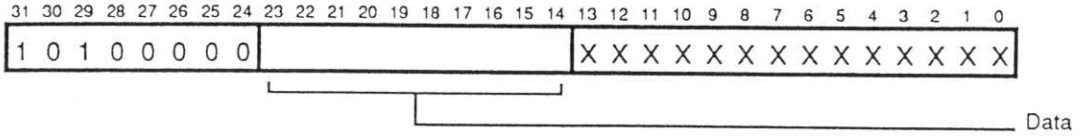


This register must be programmed if an interlaced sync. display is required. Otherwise it may be ignored.

If value L is written into the **HCR**, then value (L+1)/2 should be written into the **HIR**. [L must be odd].

This is a 10 bit register with bit 14 the least significant.

5.14 Vertical Cycle Register (VCR) : Address A0H



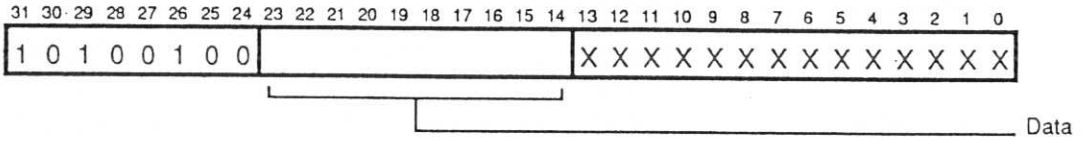
This register defines the period, in units of a raster, of the vertical scan. ie. display time + flyback time.

If N rasters are required in a complete frame, then value (N-1) should be programmed into the VCR.

If an interlaced display is selected, (N-3)/2 must be programmed into the VCR. [N must be odd]. Here N is still the number of rasters in a complete frame, *not* a field.

This is a 10 bit register, with bit 14 the least significant.

5.15 Vertical Sync Width Register (VSWR) : Address A4H

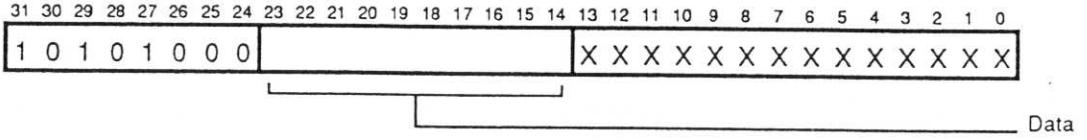


This register defines the width, in units of a raster, of the VSYNC pulse.

If N rasters are required in the VSYNC pulse, then value (N-1) should be programmed into the VSWR. The minimum value allowed for N is 1.

This is a 10 bit register, with bit 14 the least significant.

5.16 Vertical Border Start Register (VBSR) : Address A8H



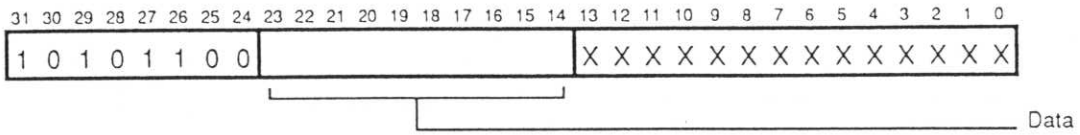
This register defines the time, in units of a raster, from the start of the VSYNC pulse to the start of the border display.

If N rasters are required in this time, then value (N-1) should be programmed into the VBSR.

If no border is required, then this register must still be programmed, in this case to the same value as the VDSR.

This is a 10 bit register, with bit 14 the least significant.

5.17 Vertical Display Start Register (VDSR) : Address ACH

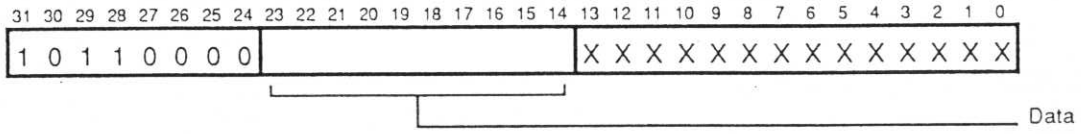


This register defines the time, in units of a raster, from the start of the VSYNC pulse to the start of the video display.

If N rasters are required in this time, then value (N-1) should be programmed into the VDSR.

This is a 10 bit register, with bit 14 the least significant.

5.18 Vertical Display End Register (VDER) : Address B0H

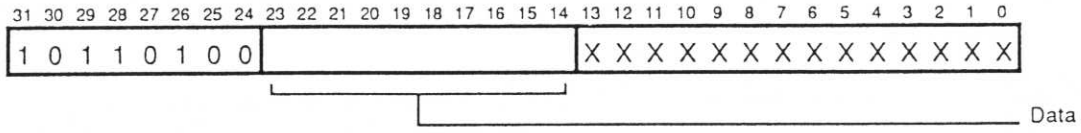


This register defines the time, in units of a raster, from the start of the VSYNC pulse to the end of the video display. (ie. the first raster on which the display is *not* present).

If N rasters are required in this time, then value (N-1) should be programmed into the VDER.

This is a 10 bit register, with bit 14 the least significant.

5.19 Vertical Border End Register (VBER) : Address B4H



This register defines the time, in units of a raster, from the start of the VSYNC pulse to the end of the border display. (ie. the first raster on which the border is *not* present).

If N rasters are required in this time, then value (N-1) should be programmed into the VBER.

If no border is required, then this register must be programmed to the same value as the VDER.

This is a 10 bit register, with bit 14 the least significant.